

Spectrum™ Technology Platform

Version 12.0 SP1

API-Handbuch



Inhalt

1 - Erste Schritte

Allgemeine Schritte für die Verwendung der API	5
Weiterleitung von Daten an einen Dienst	6
Mikro-Batch-Verarbeitung	8
Unterstützte Compiler	10
Bibliotheken von Drittanbietern	14
Netzwerkprotokolle und Ports	14
Verwenden von Beispielanwendungen	15
Verwenden von HTTPS	15
Erhöhung des Timeout-Wertes	16

2 - Die C-API

Einführung in die C-API	18
Server	34
Service	38
Message	39
DataTable	53
DataRow	61

3 - Die C++-API

Einführung in die C++ API	75
Server	89
Service	92
Message	93
DataTable	104
DataRow	111

4 - Die COM-API

Einführung	123
Server	127

Service	130
Message	131
DataTable	138
DataRow	144
Map	152

5 - Java-API

Einführung	157
Server	161
Service	168
Message	169
DataTable	175
DataRow	181

6 - Die .NET-API

Einführung	189
Server	193
Service	196
Message	197
EnhancedDataTable	203

7 - ManagementAPI-Methoden (veraltet)

Einführung	208
GetLicenseInfo	208
GetVersionInfo	209

8 - Moduldienste

Address Now-Modul	212
Enterprise Geocoding-Modul	267

GeoConfidence-Modul	354
Universal Addressing-Modul	357
Universal Name-Modul	564

9 - Info zu Spectrum™ Technology Platform

Was ist Spectrum™ Technology Platform?	576
Architektur der Unternehmensdatenverwaltung	577
Architektur von Spectrum™ Technology Platform	581
Module und Komponenten	586

Kapitel : Anhang

Anhang A: ISO-Ländercodes und Modulunterstützung	593
---	-----

1 - Erste Schritte

In this section

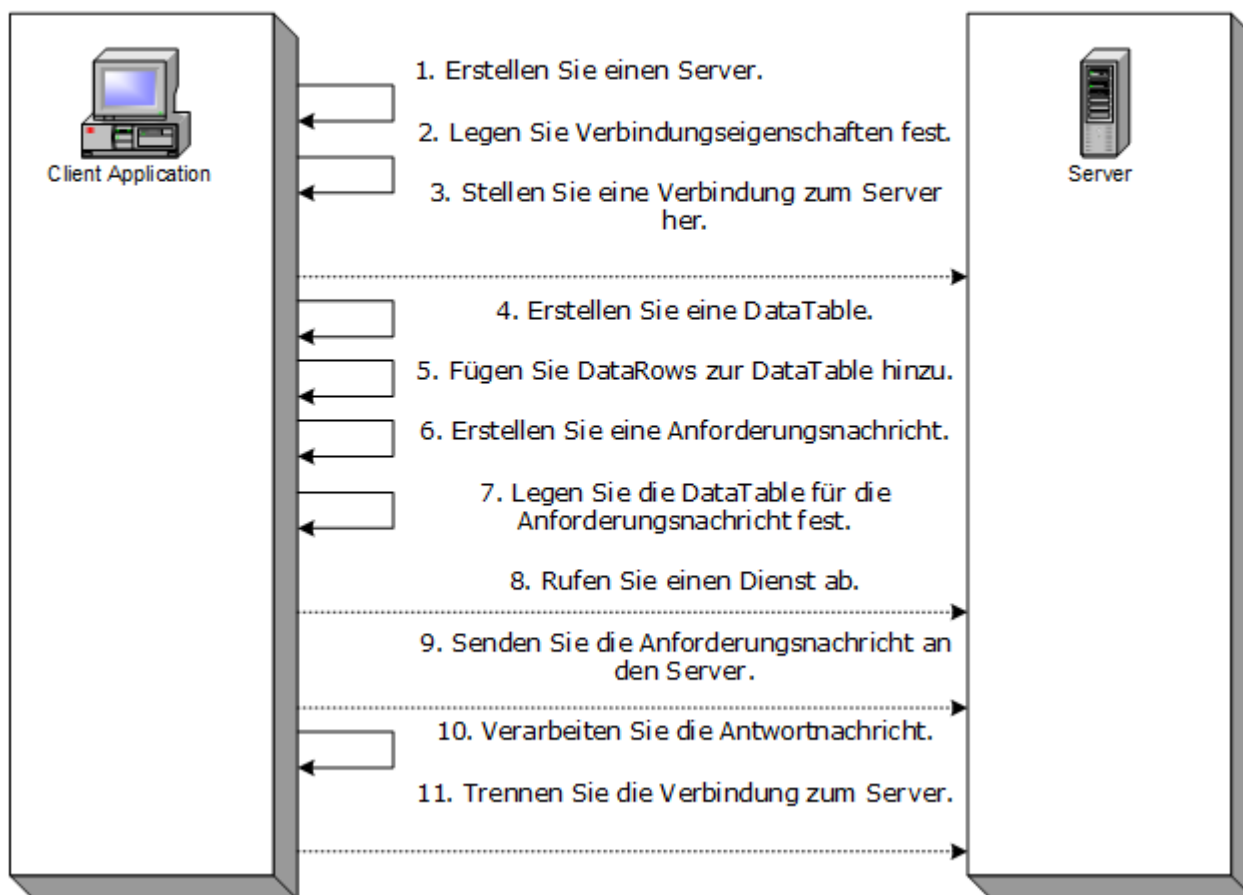
Allgemeine Schritte für die Verwendung der API	5
Weiterleitung von Daten an einen Dienst	6
Mikro-Batch-Verarbeitung	8
Unterstützte Compiler	10
Bibliotheken von Drittanbietern	14
Netzwerkprotokolle und Ports	14
Verwenden von Beispielanwendungen	15
Verwenden von HTTPS	15
Erhöhung des Timeout-Wertes	16

Allgemeine Schritte für die Verwendung der API

Die grundlegenden Schritte für die Verwendung der Spectrum™ Technology Platform-API sind die folgenden:

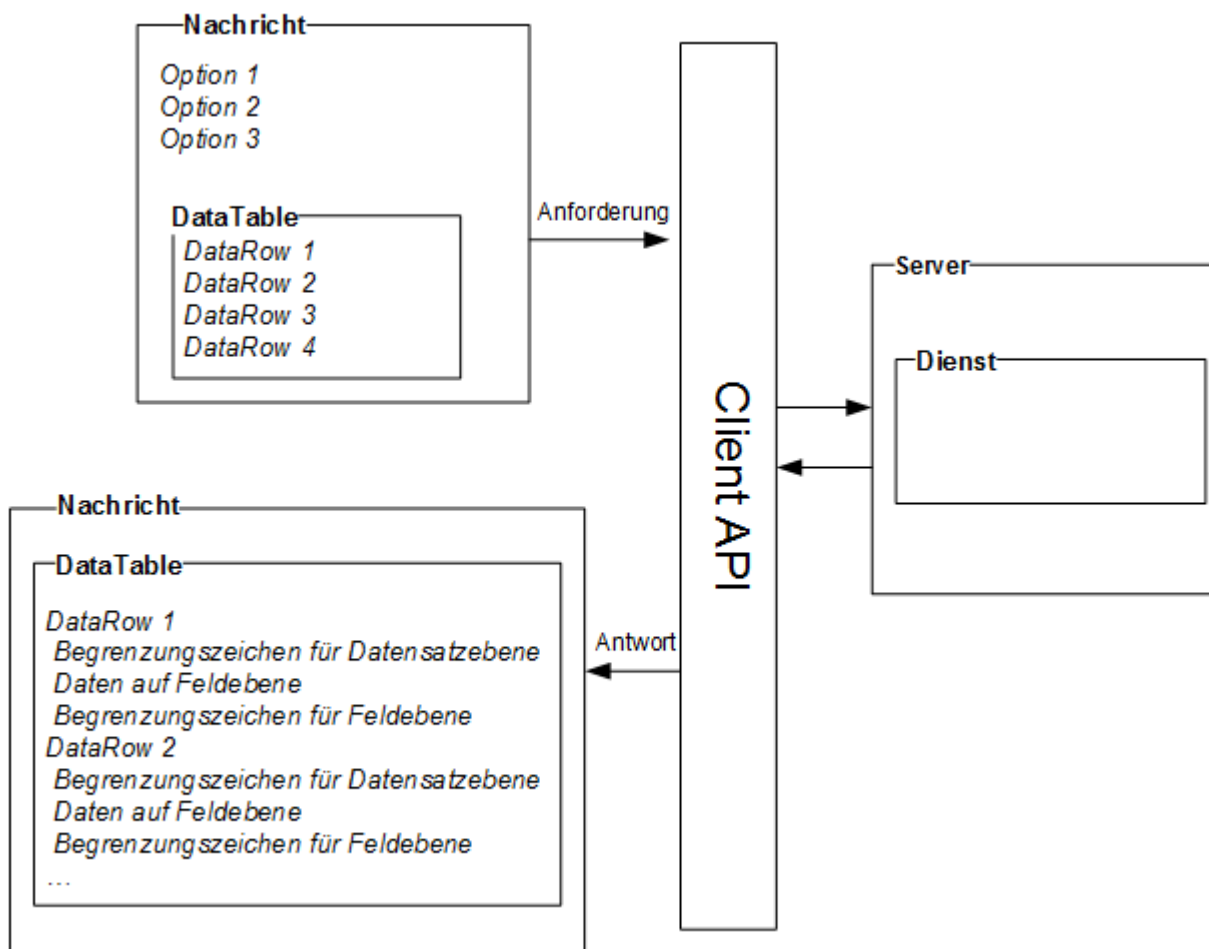
1. Erstellen Sie eine Serverinstanz.
2. Legen Sie Verbindungseigenschaften fest (Verbindungstyp, Host, Port etc.).
3. Stellen Sie eine Verbindung zum Server her.
4. Erstellen Sie eine DataTable.
5. Fügen Sie Datensätze zur DataTable hinzu.
6. Erstellen Sie eine Anforderungsnachricht.
7. Legen Sie eine DataTable für die Anforderungsnachricht fest.
8. Rufen Sie einen Dienst ab.
9. Senden Sie die Anforderungsnachricht an den Server.
10. Verarbeiten Sie die Antwortnachricht.
11. Trennen Sie die Verbindung zum Server.

Schritte für die Verwendung der Client-API



Weiterleitung von Daten an einen Dienst

Im folgenden Diagramm wird die Weiterleitung von Daten an einen Dienst über die API dargestellt:



Nachricht

Mithilfe der Message-Komponente können Sie Ihre Eingabedaten an den Spectrum™ Technology Platform-Dienst senden und Ausgabedaten von dem Dienst empfangen.

Mithilfe der Message-Komponente können Sie auch die Standardverarbeitungsoptionen eines Diensts überschreiben. Die Standardoptionen für einen Dienst werden in der Management Console festgelegt. Der Dienst „ValidateAddress“ kann beispielsweise Ausgaben in Groß- und Kleinbuchstaben oder nur Großbuchstaben erzeugen. Gehen wir davon aus, dass Sie in den meisten Fällen Ausgaben in Großbuchstaben benötigen. Allerdings muss die Ausgabe für eine Ihrer Anwendungen in Groß- und Kleinbuchstaben erfolgen. In dieser Situation würden Sie den Standard für die Schreibweise im ValidateAdress-Dienst als Großbuchstaben einstellen und die Standardeinstellung der Schreibweise für diese einzelne Anwendung über die API überschreiben. Für Anforderungen, die gemäß den angegebenen Standardwerten verarbeitet werden sollen, müssen Sie keine Eingabeoptionen in Ihrer Anforderung bereitstellen.

Die Eigenschaften für „Message“ umfassen Kontexteigenschaften wie Konto-ID, Kontokennwort, Dienstname und Dienstmethode, sowie Optionseigenschaften, bei denen es sich um die dienstspezifischen Laufzeitoptionen handelt, und Fehlereigenschaften, bei denen es sich um die Fehlerklasse, Fehlermeldung und Fehler-Stacktrace handelt.

DataTable

Die Komponente „DataTable“ enthält die Datensätze für Ihre Eingabe- und Ausgabedaten. Mithilfe der dieser Klasse zugeordneten Methoden definieren Sie die Spaltennamen für Ihre Ausgabe und fügen dem Dataset Datensätze hinzu. Mithilfe der „Reset“- und „Next“-Methoden werden die Ergebnisse durchlaufen, die in einer Antwort des Servers zurückgegeben werden.

DataRow

DataRow enthält Schemainformationen und eine Liste von Datenzeilen. Einzelne Datensätze befinden sich in Datenzeilen. Für jede Ausgabedatenzeile sind Begrenzungszeichen auf Datensatzebene, Daten auf Feldebene und Begrenzungszeichen auf Feldebene vorhanden.

Begrenzungszeichen auf Datensatzebene beschreiben die Verarbeitung des Datensatzes. Begrenzungszeichen auf Datensatzebene umfassen den Status der Anforderung (Erfolg, Fehlschlagen oder Fehler) und die Zuverlässigkeit im Hinblick auf die Genauigkeit des Ausgabedatensatzes.

Daten auf Feldebene umfassen den geprüften, standardisierten oder erweiterten Datensatz.

Begrenzungszeichen auf Feldebene enthalten zusätzliche Daten zu einem bestimmten Feld. Der Typ eines Postfachs gemäß der Kategorisierung des USPS ist beispielsweise ein Begrenzungszeichen auf Feldebene.

Server

Die Server-Komponente stellt den Spectrum™ Technology Platform-Server dar. Mithilfe einer Server-Komponente stellen Sie eine Verbindung zu einem bestimmten Dienst des Servers her, trennen diese oder greifen auf den Dienst zu.

Dienst

Die Service-Komponente wird zur Verarbeitung der gesendeten Nachricht verwendet (d. h., die Eingabenachricht wird versandt und die Antwort empfangen). Die Service-Komponente verfügt über lediglich eine Methode: Nachricht verarbeiten.

Mikro-Batch-Verarbeitung

Bei der Mikro-Batch-Verarbeitung handelt es sich um eine Methode, bei der Sie mehr als einen Datensatz in eine einzelne Dienstanforderung aufnehmen können. Wenn Sie eine umfangreiche Sammlung an Datensätzen über einen Dienst verarbeiten, können Sie die Leistung bedeutend verbessern, indem Sie mehrere Datensätze in eine Anforderung aufnehmen, statt separate Anforderungen für jeden Datensatz auszustellen. Spectrum™ Technology Platform unterstützt die Mikro-Batch-Verarbeitung für REST- und SOAP-Webservices sowie für das Client-SDK.

Mikro-Batchgröße

Die Anzahl der in eine Anforderung aufzunehmenden Datensätze ist unbegrenzt. Im Allgemeinen erhalten Sie die beste Leistung, wenn Sie zwischen 50 und 100 Datensätze in einem Mikro-Batch senden. Wir empfehlen, Mikro-Batches unterschiedlicher Größen zu testen, um die optimale Mikro-Batchgröße für Ihre Umgebung zu ermitteln. Hinweis: In einigen Fällen erhalten Sie eventuell mehrere Datensätze in der Antwort für jeden Eingabedatensatz. Wenn Sie beispielsweise eine Adressenüberprüfung durchführen und 10 Adressen in den Mikro-Batch aufnehmen, wobei jede Adresse mit zwei möglichen überprüften Adressen übereinstimmt, würden Sie 20 Datensätze statt nur 10 in der Antwort erhalten.

Gehen Sie vorsichtig vor, wenn Sie Mikro-Batches und mehrere Threads für Anforderungen an Spectrum™ Technology Platform verwenden. Mehrere Threads überfordern das System eventuell, wenn die Mikro-Batchgröße jedes Threads zu groß ist.

Verwenden einer Datensatz-ID

Es ist eventuell hilfreich, jedem Datensatz in einem Mikro-Batch eine ID zuzuweisen, sodass Sie die Datensätze in der Anforderung mit den in der Antwort zurückgegebenen Datensätzen in Beziehung setzen können. Führen Sie dies anhand von Benutzerfeldern durch.

Mikro-Batch-Verarbeitung in der Client-API

Senden Sie mehrere Datenzeilen in der Anforderung, um die Mikro-Batch-Verarbeitung in einer API-Anforderung an einen Dienst durchzuführen. Beispiel: Bei der folgenden .NET-Klasse werden zwei Zeilen in der Anforderung gesendet:

```
using System;
using System.Collections.Generic;
using System.Diagnostics;
using System.Linq;
using System.Net;
using System.Text;
using ConsoleApplication1.ValidateAddress_Reference;

namespace Test
{
    class Program
    {
        static void Main(string[] args)
        {
            var validateClient = new ValidateAddress {Credentials = new
NetworkCredential("admin", "admin")};

            var address1 = new input_portAddress
            {
                AddressLine1 = "1825B Kramer Lane",
                AddressLine2 = "Suite 100",
                PostalCode = "78758",
                City = "Austin",
                StateProvince = "Texas"
            }
        }
    }
}
```

```

};

var address2 = new input_portAddress
{
    AddressLine1 = "100 Congress",
    PostalCode = "78701",
    City = "Austin",
    StateProvince = "Texas"
};

var addresses = new input_portAddress[2];
addresses[0] = address1;
addresses[1] = address2;

var options = new options {OutputCasing = OutputCasing.M};
output_portAddress[] results =
validateClient.CallValidateAddress(options, addresses);

for (int i = 0; i < results.Length; i++)
{
    System.Console.WriteLine("Record " + (i+1) + ":");
    System.Console.WriteLine("AddressLine1=" +
results[i].AddressLine1);
    System.Console.WriteLine("City=" + results[i].City);
    System.Console.WriteLine("StateProvince=" +
results[i].StateProvince);
    System.Console.WriteLine("PostalCode=" +
results[i].PostalCode + "\n");
}

    System.Console.Write("Press any key to continue...");
    System.Console.ReadKey();
}
}
}

```

Tip: Trennen Sie die Verbindung zwischen Anforderungen nicht. Durch das Trennen und Herstellen einer Verbindung wird die Leistung eventuell reduziert.

Unterstützte Compiler

Das Client-SDK von Spectrum™ Technology Platform wird mit den hier aufgelisteten Compiler- und Runtime-Mindestversionen unterstützt.

Java

Paketverzeichnis des Client-SDK: `clientSDK/platforms/java`

Für das Client-SDK ist das Java JDK ab Version 1.4 erforderlich. Diese ist nicht auf dem Client-SDK installiert.

Windows 32-Bit

- JDK: 1.4
- C-Compiler: MSVC 6.0 SP3, MSVC 2003, MSVC 2005, MSVC 2008
- C++-Compiler: MSVC 6.0 SP3, MSVC 2003, MSVC 2005, MSVC 2008
- C# .NET: Microsoft .NET Framework 1.1
- Visual Basic: MS Visual Basic 6.0

Windows 64-Bit

- JDK: 1.4
- C-Compiler: MSVC 2005, MSVC 2008
- C++-Compiler: MSVC 2005, MSVC 2008

HP-UX RISC

- JDK: 1.4
- C Compiler: cc: HP92453-01 A.11.01.21 HP C (Gebündelt) Compiler
- C++ Compiler: aCC: HP aC++ B3910B A.03.30 HP aC++ B3910B A.03.27

Die clientSDK-32-Bit-Bibliothek ist mit diesen Bibliotheken verknüpft:

- libpthread.1
- librt.2
- libnsl.1
- libxti.2

Die clientSDK-64-Bit-Bibliothek ist mit diesen Bibliotheken verknüpft:

- libpthread.1
- libnsl.1
- librt.2
- libdl.1
- libc.2
- libxti.2
- libdl.1

HP-UX Itanium

- JDK: 1.4
- C Compiler: cc: HP aC++/ANSI C B3910B A.06.05
- C++ Compiler: aCC: HP aC++/ANSI C B3910B A.06.05

Die clientSDK-32-Bit-Bibliothek ist mit diesen Bibliotheken verknüpft:

- libpthread.so.1
- libnsl.so.1
- librt.so.1
- libxti.so.1
- libdl.so.1

Die clientSDK-64-Bit-Bibliothek ist mit diesen Bibliotheken verknüpft:

- libpthread.so.1
- libnsl.so.1
- librt.so.1
- libxti.so.1
- libdl.so.1

Red Hat (32-Bit)

- Betriebssystem: Red Hat Linux 2.4.9-e.65smp
- C Compiler: gcc version 2.96 (gcc 4.1 für das „Address Now“-Modul erforderlich)
- C++ Compiler: g++ Version 2.96

Die clientSDK-Bibliothek ist mit diesen Bibliotheken verknüpft:

- libstdc++-libc6.2-2.so.3
- libm.so.6
- libc.so.6
- ld-linux.so.2

Red Hat (64-Bit)

- Betriebssystem: Red Hat Linux Version 2.6.9-34.0.2.ELsmp
- C Compiler: gcc Version 3.4.5
- C++ Compiler: g++ Version 3.4.5

Die clientSDK-Bibliothek ist mit diesen Bibliotheken verknüpft:

- libstdc++.so.6
- libm.so.6
- libgcc_s.so.1
- libpthread.so.0
- libc.so.6
- ld-linux-x86-64.so.2

SuSE

- Betriebssystem: SuSE SLES 8 (gehostet von UnitedLinux 1.0) (i586)\nKernel 2.4.21-295-smp (0).
- C Compiler: gcc Version 3.2.2
- C++ Compiler: g++ Version 3.2.2

Die clientSDK-Bibliothek (32-Bit) ist mit diesen Bibliotheken verknüpft:

- libstdc++.so.5
- libm.so.6
- libgcc_s.so.1
- libc.so.6
- ld-linux.so.2

Solaris

- Betriebssystem: Solaris 5.8
- C Compiler: cc: Forte Developer 7 C 5.4 2002/03/09
- C++ Compiler: CC: Forte Developer 7 C++ 5.4 Patch 111715-16 2005/04/28

Die clientSDK-32-Bit-Bibliothek ist mit diesen Bibliotheken verknüpft:

- libpthread.so.1
- libsocket.so.1
- libnsl.so.1
- librt.so.1
- libc.so.1
- libdl.so.1
- libmp.so.2
- libaio.so.1
- libc_psr.so.1

Die clientSDK-64-Bit-Bibliothek ist mit diesen Bibliotheken verknüpft:

- libpthread.so.1
- libsocket.so.1
- libnsl.so.1
- librt.so.1
- libc.so.1
- libmp.so.2
- libmd5.so.1
- libscf.so.1
- libaio.so.1
- libdoor.so.1
- libuutil.so.1
- libm.so.2
- libc_psr.so.1
- libmd5_psr.so.1

AIX

- Betriebssystem: AIX Version 5.1.0.0

- C Compiler: xlc 6.0 Visual Age C 6.0
- C++ Compiler: xlc 6.0 Visual Age C++ 6.0

Die clientSDK-32-Bit- und clientSDK-64-Bit-Bibliotheken sind mit diesen Bibliotheken verknüpft:

- libC.a
- libc_r.a
- libpthread.a
- librtl.a

Bibliotheken von Drittanbietern

Die Spectrum™ Technology Platform API verwendet die folgenden Bibliotheken von Drittanbietern:

- Apache Commons Pool 1.6
- ICU 3.2.0
- Jakarta Commons HttpClient 3.1
- OpenSSL 1.0.2L
- OpenTop 1.5.3
- POCO 1.3

Netzwerkprotokolle und Ports

Die API kommuniziert über HTTP, HTTPS oder SOCKET mit dem Spectrum™ Technology Platform-Server. Spectrum™ Technology Platform verwendet in der Regel den Port 8080, um HTTP-Anforderungen zu empfangen, und den Port 443 für HTTPS-Anforderungen. HTTP- und HTTPS-Features werden auch in den C, C++, COM, Java und .NET APIs unterstützt. .NET, Java und COM APIs unterstützen Unicode. C und C++ APIs unterstützen sowohl ASCII als auch Unicode.

Neben HTTP unterstützt Spectrum™ Technology Platform auch eine permanente SOCKET-Verbindung. Die SOCKET-Hochgeschwindigkeitsverbindung bietet eine schnellere Leistung als das herkömmliche HTTP. Spectrum™ Technology Platform verwendet in der Regel Port 10119, um SOCKET-Anforderungen zu empfangen.

Verwenden von Beispielanwendungen

Im Client-SDK sind Beispielanwendungen für alle unterstützten Sprachen enthalten. Die Beispielanwendungen rufen einen Beispieldienst auf dem Spectrum™ Technology Platform-Server auf, der die Schreibweise der Eingabedaten in entweder Großbuchstaben oder Kleinbuchstaben ändert.

1. Kopieren Sie die Datei `casing-<version>.car` von `ClientAPI\common\lib` in den Ordner `server\app\deploy` auf dem Spectrum™ Technology Platform-Server.

Der von den Beispielanwendungen verwendete Schreibweisendienst wird nun auf Ihrem Spectrum™ Technology Platform-Server bereitgestellt.

2. Suchen Sie im Ordner `ClientAPI\platforms` nach dem Unterordner `samples` für Ihre Plattform, und öffnen Sie die Datei `readme.txt`, um weitere Anweisungen zur Verwendung der Beispielanwendungen zu erhalten.

Anmerkung: Sie können die Beispielanwendung ändern, damit sie einen der von Ihnen lizenzierten Dienste verwendet, und die Ausführung neu kompilieren.

Verwenden von HTTPS

In dieser Prozedur wird beschrieben, wie die HTTPS-Kommunikation zwischen Ihrer Anwendung und dem Spectrum™ Technology Platform-Server verwendet wird.

1. Geben Sie die Stamm-CA an, die für die Kommunikation zwischen Ihrer Anwendung und dem Spectrum™ Technology Platform-Server verwendet wird, indem Sie wie folgt vorgehen:

Option	Bezeichnung
Wenn Sie nicht wissen, welche Stamm-CA verwendet wird	<p>Kopieren Sie die Datei <code>ca-bundle.pem</code> in Ihr Arbeitsverzeichnis. Bei C/C++, COM und ASP befindet sich die <code>.pem</code>-Datei an dem Speicherort, an dem Sie das Client-SDK installiert haben, in dem folgenden Ordner:</p> <pre>Spectrum Client SDK\ClientAPI\platforms\windows\c-c++\<32or64>\<version>\lib\openssl</pre> <p>Bei ASP lauten einige Beispiele für einen Arbeitsordner wie folgt:</p> <ul style="list-style-type: none"> • Wenn Sie für die Ausführung von ASP Internetinformationsdienste verwenden, kopieren Sie <code>ca-bundle.pem</code> in das Windows-Systemverzeichnis (z. B. <code>C:\Windows\system32</code>).

Option	Bezeichnung
--------	-------------

-
- Wenn Sie für die Ausführung von ASP Internet Explorer verwenden, kopieren Sie `ca-bundle.pem` in das Standardarbeitsverzeichnis von Internet Explorer (z. B. `C:\Documents and Settings\\Desktop`).
-

Wenn Sie wissen, welches Stamm-CA verwendet wird Geben Sie in Ihrer CA-Bundledatei das Stamm-CA-Zertifikat an.

2. Legen Sie den Verbindungstyp in Ihrer Anwendung auf HTTPS fest, wenn Sie eine Verbindung zum Server herstellen.

Erhöhung des Timeout-Wertes

Wenn es zu Timeouts zwischen dem Client und Server kommt, können Sie den Timeout-Wert für den Client erhöhen.

- Legen Sie den Timeout-Wert mithilfe der `setConnectionProperty`-Methode fest.

2 - Die C-API

In this section

Einführung in die C-API	18
Server	34
Service	38
Message	39
DataTable	53
DataRow	61

Einführung in die C-API

Die C-API besteht aus folgenden Strukturen:

- Server
- Service
- Message
- DataTable
- DataRow

Anmerkung: Bei der C-API handelt es sich um einen C-Wrapper um den C++-Code. Unter Unix können Sie einen C++-Compiler verwenden, um Ihre C-Anwendung zu erstellen. Dies ist der bevorzugte Ansatz. Ein C-Compiler kann jedoch auch direkt unter Linux und Solaris verwendet werden. Unter HP-UX und AIX müssen Sie alle erforderlichen C++-Bibliotheken verlinken, wenn Sie den C-Compiler verwenden. Führen Sie dazu `ldd ./batch in`
`....samples/batch/bin/` aus, um eine Liste aller abhängigen Bibliotheken abzurufen, und führen Sie diese im Linkbereich Ihres makefile auf.

Unterstützte Bibliotheken

Spectrum™ Technology Platform stellt eine C-API in einer ASCII- und Unicode-Version bereit, wobei die Unicode-Version so kompatibel wie möglich mit dem ursprünglichen Design der ASCII-Version bleibt. Spectrum™ Technology Platform wendet International Components for Unicode (ICU) in der API an, um das Unicode-Feature zu unterstützen. Bei ICU handelt es sich um eine ausgereifte und weit verbreitete Gruppe von C/C++-Bibliotheken zur Unicode-Unterstützung, die von IBM entwickelt wurde.

Der Unicode-Standard definiert eine Standardcodierung basierend auf 16-Bit-Codeeinheiten. Diese wird in ICU durch die Definition von UChar als vorzeichenloser 16-Bit-Integer-Typ (unsigned short *) unterstützt. Dies ist der Basistyp für Zeichenarrays für Zeichenfolgen in ICU. Spectrum™ Technology Platform verwendet UChar als die Unicode-Zeichenfolgendarstellung in unserer C-API.

Anmerkung: Nicht alle Dienste unterstützen den vollständigen Unicode-Zeichensatz. Der „Validate Address“-Dienst unterstützt beispielsweise den Zeichensatz ISO 8859-1 für US-Eingaben und für internationale Eingaben und Ausgaben sowie den Zeichensatz CP 850 für kanadische Eingaben und Ausgaben. Die Unicode-Bibliotheken sollten jedoch immer verwendet werden, wenn Ihre Eingabedaten möglicherweise Zeichen enthalten, die nicht zum ASCII-Zeichensatz gehören, auch wenn der zugrunde liegende Dienst den vollständigen Unicode-Zeichensatz nicht unterstützt.

Detaillierte Informationen zu UChar finden Sie auf den folgenden beiden Websites:

- icu.sourceforge.net/userguide
- www.ibm.com/software/globalization/icu

Unter Windows unterstützte C-Bibliotheken

Jede API-Konfiguration erzeugt Bibliotheksdateien mit einem gemeinsamen Basisnamen (g1client), jedoch mit einem eindeutigen Suffix und gegebenenfalls Präfix (im Fall von statischen Bibliotheken „lib“). Die Bibliothekssuffixe geben Folgendes an:

```
<lib>g1client<S><U><D>.<lib|dll>
```

- lib: Gibt eine statische Bibliothek an.
- dll: Gibt eine dynamische (freigegebene) Bibliothek an.
- S: Verweist auf einen Single-Thread-Build. Wenn „S“ nicht vorhanden ist, handelt es sich um eine Multi-Thread-Version.
- U: Verweist auf einen Build der UNICODE-Version. Wenn „U“ nicht vorhanden ist, handelt es sich um ein ASCII-Build.
- D: Verweist auf einen Debug-Build. Wenn dieses Suffix nicht vorhanden ist, wird ein optimierter Release-Build angegeben.

Um die UNICODE-Version zu aktivieren, muss die Makrodefinition „LIB_UNICODE“ in Ihrem Projekt vorhanden sein.

Um die UNICODE-Version der statischen C/C++-API-Bibliothek zu verwenden, müssen Sie in Ihrem Projekt „U_STATIC_IMPLEMENTATION“ definieren.

Um die dynamische Version zu verwenden, müssen Sie in Ihrem Projekt „G1CLIENT_DLL“ definieren.

Wir stellen darüber hinaus im Verzeichnis der Header-Dateien eine Datei namens „auto_link.h“ zur Verfügung, die entsprechend den Projekteinstellungen automatisch eine Verlinkung zu allen zugehörigen Bibliotheken erstellt.

Um 64-Bit-Bibliotheken unter Windows aufzurufen, müssen Sie „VER_64“ in Ihrem Projekt definieren.

Statische Bibliothek

Anmerkung: Die in diesem Abschnitt aufgeführten Namen beziehen sich auf 32-Bit-Bibliotheken. Ersetzen Sie für 64-Bit-Bibliotheken „32“ im Bibliotheksnamen durch „64“.

Tabelle 1: Single-Thread/Release

	ASCII	Unicode
g1	libg1client_S.lib	libg1client_SU.lib

	ASCII	Unicode
openssl	otlibeay32.lib otlibssl32.lib	otlibeay32.lib otlibssl32.lib
opentop	opentop.lib	opentopw.lib
icu		libicuuc.lib libicudt.lib libicuin.lib libicuiio.lib
Poco	PocoXML32.lib	PocoXML32w.lib

Tabelle 2: Single-Thread/Debug

	ASCII	Unicode
g1	libg1client_SD.lib	libg1client_SUD.lib
openssl	otlibeay32d.lib otlibssl32d.lib	otlibeay32d.lib otlibssl32d.lib
opentop	opentopd.lib	opentopwd.lib
icu		libicuucd.lib libicudtd.lib libicuind.lib libicuiod.lib
Poco	PocoXML32d.lib	PocoXML32wd.lib

Tabelle 3: Multi/Release (verwendet Multi-Threaded CRT)

	ASCII	Unicode
g1	libg1client.lib	libg1client_U.lib

	ASCII	Unicode
openssl	otlibeay32mt.lib otlibssl32mt.lib	otlibeay32mt.lib otlibssl32mt.lib
opentop	opentopmt.lib	opentopmtw.lib
icu		libicuucmt.lib libicudtmt.lib libicuinmt.lib libicuiomt.lib
Poco	PocoXMLmt32.lib	PocoXML32mtw.lib

Tabelle 4: Multi/Debug (verwendet Multi-Threaded CRT)

	ASCII	Unicode
g1	libg1client_D.lib	libg1client_UD.lib
openssl	otlibeay32mtd.lib otlibssl32mtd.lib	otlibeay32mtd.lib otlibssl32mtd.lib
opentop	opentopmtd.lib	opentopmtwd.lib
icu		libicuucmtd.lib libicudtmtd.lib libicuinmtd.lib libicuiomtlib
Poco	PocoXMLmt32d.lib	PocoXML32mtwd.lib

Dynamische Bibliothek

Anmerkung: Die in diesem Abschnitt aufgeführten Namen beziehen sich auf 32-Bit-Bibliotheken. Ersetzen Sie für 64-Bit-Bibliotheken „32“ im Bibliotheksnamen durch „64“.

Tabelle 5: Multi/Release (verwendet Multi-Threaded CRT)

	ASCII	Unicode
g1	g1client.dll	g1client_U.dll
openssl	otlibeay32mts.dll otplibssl32mts.dll	otlibeay32mts.dll otplibssl32mts.dll
opentop	opentopmts.dll	opentopmtws.dll
icu		icuuc32.dll icuio32.dll icuin32.dll icudt32.dll
Poco	PocoXML32mts.dll	PocoXML32mtws.dll

Tabelle 6: Multi/Debug (verwendet Multi-Threaded CRT)

	ASCII	Unicode
g1	g1client_D.dll	g1client_UD.dll
openssl	otlibeay32mtds.dll otplibssl32mtds.dll	otlibeay32mtds.dll otplibssl32mtds.dll
opentop	opentopmtds.dll	opentopmtwds.dll
icu		icuuc32d.dll icuio32d.dll icuin32d.dll icudt32d.dll
Poco	PocoXML32mtds.dll	PocoXML32mtwds.dll

Unter Unix unterstützte C-Bibliotheken

Jede ClientSDK-Konfiguration erzeugt Bibliotheksdateien mit einem gemeinsamen Basisnamen (libg1client), jedoch mit einem eindeutigen Suffix. Spectrum™ Technology Platform stellt einen Multi-Thread- und einen Release-Build für die ASCII-Version und die UNICODE-Version bereit.

Die Bibliothekssuffixe geben Folgendes an:

```
libg1client<U>.<so|sl|a>
```

„U“ zeigt dabei eine UNICODE-Version an. Wenn „U“ nicht vorhanden ist, handelt es sich um ein ASCII-Build.

Um die UNICODE-Version zu verwenden, müssen Sie in Ihrem Projekt „LIB_UNICODE“ definieren. In der C++-API der UNICODE-Version ist der Namespace für alle Klassen „g1client“.

Tabelle 7: AIX

	ASCII	Unicode
g1	libg1client.so	libg1client_U.so
openssl	libcrypto.so libssl.so	libcrypto.so libssl.so
opentop	libopentop-xlCmt.so	libopentop-xlCmtw.so libotxml-xlCmtw.so
icu		libicudata34.a libicui18n34.a libicuio34.a libicuuc34.a
Poco	libPocoXML.so	

Tabelle 8: HP-UX RISC

	ASCII	Unicode
g1	libg1client.sl	libg1client_U.sl

	ASCII	Unicode
openssl	libcrypto.sl libssl.sl libcrypto.sl.0.9.7 libssl.sl.0.9.7	libcrypto.sl libssl.sl libcrypto.sl.0.9.7 libssl.sl.0.9.7
opentop	libopentop-accmt.sl	libopentop-accmtw.sl libotxml-accmtw.sl
icu		libicudata.sl libicudata.sl.34 libicui18n.sl libicui18n.sl.34 libicuio.sl libicuio.sl.34 libicuuc.sl libicuuc.sl.34
Poco	libPocoXML.sl	

Tabelle 9: HP-UX Itanium

	ASCII	Unicode
g1	libg1client.sl	libg1client_U.sl
openssl	libcrypto.a libssl.a	libcrypto.a libssl.a
opentop	libopentop-accmt.sl	libopentop-accmtw.sl libotxml-accmtw.sl
icu		libicudata.sl libicudata.sl.34 libicudata.sl.34.0 libicui18n.sl libicui18n.sl.34 libicui18n.sl.34.0 libicuio.sl libicuio.sl.34 libicuio.sl.34.0 libicuuc.sl libicuuc.sl.34 libicuuc.sl.34.0
Poco	libPocoXML.sl	

Tabelle 10: Linux

	ASCII	Unicode
g1	libg1client.so	libg1client_U.so
openssl	libcrypto.so libcrypto.so.0.9.7 libssl.so libssl.so.0.9.7	libcrypto.so libcrypto.so.0.9.7 libssl.so libssl.so.0.9.7
opentop	libopentop-gccmt.so	libopentop-gccmtw.so libotxml-gccmtw.so
icu		libicudata.so libicudata.so.34 libicui18n.so libicui18n.so.34 libicuio.so libicuio.so.34 libicuuc.so libicuuc.so.34
Poco	libPocoXML.so	

Tabelle 11: Solaris SPARC

	ASCII	Unicode
g1	libg1client.so	libg1client_U.so
openssl	libcrypto.so libcrypto.so.0.9.7 libssl.so libssl.so.0.9.7	libcrypto.so libcrypto.so.0.9.7 libssl.so libssl.so.0.9.7
opentop	libopentop-fortemt.so	libopentop-fortemtw.so libotxml-fortemtw.so
icu		libicudata.so libicudata.so.34 libicui18n.so libicui18n.so.34 libicuio.so libicuio.so.34 libicuuc.so libicuuc.so.34
Poco	libPocoXML.so	

Konstanten

Die C-API verwendet zwei Sätze mit Konstanten. Der erste Satz ist für die `Server`-Komponente, die in der unteren Tabelle beschrieben wird.

Tabelle 12: Konstanten für die `Server`-Komponente

Name der Konstante	Beschreibung/Standard	Beispiel
<code>SERVER_HOST</code>	Zeichenfolge für den Hostnamen des Servers. Der Standardname lautet „localhost“.	65.89.200.89
<code>SERVER_PORT</code>	Zeichenfolge für den Serverport. Der Standardwert ist „8080“.	10119
<code>SERVER_ACCOUNT_ID</code>	Zeichenfolge für die Serverkonto-ID. Kein Standardwert.	user1
<code>SERVER_ACCOUNT_PASSWORD</code>	Zeichenfolge für das Kennwort des Serverkontos. Kein Standardwert.	user1
<code>SERVER_CONNECTION_TIMEOUT</code>	Zeichenfolge für Timeout der Serververbindung in Millisekunden. Der Standardwert ist „5000“.	50000
<code>SERVER_CONNECTION_TYPE</code>	Zeichenfolge für den Serververbindungstyp. Derzeit werden nur HTTP, HTTPS und SOCKET unterstützt. Der Standardwert lautet „HTTP“.	HTTP(S)
<code>SERVER_PROXY_HOST</code>	Zeichenfolge für den Hostnamen des Proxy-Servers. Kein Standardwert.	192.168.1.77
<code>SERVER_PROXY_PORT</code>	Zeichenfolge für den Proxy-Serverport. Kein Standardwert.	8080

Name der Konstante	Beschreibung/Standard	Beispiel
SERVER_PROXY_USER	Zeichenfolge für die Konto-ID des Proxy-Servers. Kein Standardwert.	user1
SERVER_PROXY_PASSWORD	Zeichenfolge für das Kontokennwort des Proxy-Servers. Kein Standardwert.	user1

Der zweite Konstantensatz ist für die `Message`-Komponente vorgesehen.

Tabelle 13: Konstanten für die `Message`-Komponente

Name der Konstante	Beschreibung	Beispiel
MESSAGE_CONTEXT_ACCOUNT_ID	Zeichenfolge für den Nachrichtentext der Konto-ID.	user1
MESSAGE_CONTEXT_ACCOUNT_PASSWORD	Zeichenfolge für das Kontokennwort des Nachrichtenkontextes.	user1
MESSAGE_CONTEXT_SERVICE_NAME	Zeichenfolge für den Dienstnamen des Nachrichtenkontextes.	echoservice

Fehlermeldungen

Einige Funktionen geben einen Wert „SUCCESSFUL_RETURN“ oder 0 (NULL) zurück, wenn sie erfolgreich ausgeführt wurden. Wenn die Ausführung nicht erfolgreich war, gibt die Funktion einen Fehlercode zurück. Um Fehlermeldungen abzurufen, rufen Sie „`getErrorMessage(int errorCode)`“ auf. Beispiel:

```
Server *server = NULL;
int nRet;
//Create Server
server = createServer();
//set the property to the server
```

```

...
//Connect to server
printf("Making connection to the server...\n");
nRet = serverConnect(server);
if(nRet != SUCCESSFUL_RETURN)
{
// ASCII Version-use the following code
printf(getErrorMessage(nRet));
//Unicode Version -use the following code
UChar * error = getErrorMessage(nRet);
// more code to print out the error message...
return ;
}

```

Die C-API verwendet die folgenden Fehlermeldungen.

- Fehlermeldungen für die Übergabe einer NULL-Struktur:
 - "Input null DataRow"
 - "Input null DataTable"
 - "Input null Message"
 - "Input null Server"
- Fehlermeldungen für Verbindungen:
 - "Connection type not supported"
 - "Client timeout"
 - "Blank connection property name"
 - "Blank property name"
- Fehlermeldungen zum Erstellen von Datentabellen (DataTables):
 - "Blank column name"
 - "Duplicated column name"
- Fehlermeldungen für die MessagePackaging-Ausnahme:
 - "Input Message is null"
 - "Failed to connect to Server"
 - "Failed to disconnect from Server"
 - "Failed to open Http Connection"
 - "Failed to get Service"
 - "Failed to package the message using Serializer and Encoding"

Beispielanwendung

Der folgende Beispielcode zeigt die Verwendung der ASCII-Version der C-API.

```
// Declarations
Server *server = NULL;
Message *request = NULL;
DataTable *dataTable = NULL;
Message *reply = NULL;
Service *service = NULL;
int nRet;
DataRow *row1 = NULL;
DataRow *row2 = NULL;
DataTable *returnDataTable= NULL;
char** columnNames;
DataRow** rows;
DataRow*dataRow;
int i;
int j;
char* value;

//Create Server
server = createServer();

//Set server connection properties
nRet = setConnectionProperty(server, SERVER_HOST, "localhost");
nRet = setConnectionProperty(server, SERVER_PORT, "10119 ");
nRet = setConnectionProperty(server, SERVER_CONNECTION_TYPE,"SOCKET");

nRet = setConnectionProperty(server, SERVER_ACCOUNT_ID,"guest");
nRet = setConnectionProperty(server, SERVER_ACCOUNT_PASSWORD,"");

//Connect to server
nRet = serverConnect(server);
if(nRet != SUCCESSFUL_RETURN)
{
printf( getErrorMessage(nRet));
// free memory
if(server)
nRet = deleteServer(server);
return ;
}

//Get Service From Server
service = getServiceFromServer(server,"ValidateAddress" );

//Create Input Message
request = createMessage();

//Fill DataTable in the input message
```

```

dataTable = getDataTable(request);
nRet= addColumn( dataTable, "AddressLine1", &nRet);
nRet= addColumn( dataTable, "City", &nRet);
nRet= addColumn( dataTable, "StateProvince", &nRet);

row1 = newRow( dataTable );
setByIndex (row1, 0 , "4200 Parliament Place");
setByIndex (row1, 1 , "Lanham");
setByIndex (row1, 2 , "Maryland");

addRow( dataTable, row1);

row2 = newRow( dataTable );
setByIndex (row2, 0 , "10535 Boyer Blvd");
setByIndex (row2, 1 , "Austin");
setByIndex (row2, 2 , "Texas");

addRow( dataTable, row2);

//Set"option" Properties to the Input Message
nRet = putOption(request, "OutputCasing", "M");
nRet = putOption(request, "OutputRecordType", "A");

//Process Input Message, return output Message
nRet = processMessage(service, request, &reply);
if(nRet != SUCCESSFUL_RETURN)
{
printf("Error Occurred, " );
printf(getErrorMessage(nRet));

// free memory
if(request)
nRet = deleteMessage(request);
if(reply)
nRet = deleteMessage(reply);
if(server)
nRet = deleteServer(server);

return ;

}

//Disconnect from server
nRet = serverDisconnect(server);

//Get the result from the response message
returnDataTable = getDataTable(reply );
columnNames = getColumnNames(returnDataTable);

rows = getDataRows( returnDataTable);

for( i=0; i < getRowCount( returnDataTable); i++)
{

```

```

dataRow = rows[i];

for(j=0; j < getColumnCount(returnDataTable); j++)
{
value = (char*)getByIndex( dataRow, j);
printf(value);
printf("\n");
}
}

//Free Memory
if(request)
nRet = deleteMessage(request);
if(reply)
nRet = deleteMessage(reply);
if(server)
nRet = deleteServer(server);
}

```

Der folgende Beispielcode zeigt die Verwendung der Unicode-Version der C-API. Hier wird die Zeichenfolge durch „UChar“ (oder unsigned short*) repräsentiert, wobei die Unicode-Zeichenfolge als 16-Bit-Typ vorliegt. ICU bietet eine Funktion namens „u_charsToUChars“, die eine 8-Bit-Zeichenfolge in eine 16-Bit-Zeichenfolge konvertiert. Das folgende Beispiel zeigt den Aufruf der Unicode-Version der C-API. Die Eingabezeichenfolgen liegen alle in ASCII vor. Daher verwenden wir „u_charsToUChars“, um sie in 16-Bit-Zeichenfolgen zu konvertieren. Sie könnten auch eine Unicode-Zeichenfolge erstellen, um sie der C-API direkt zu übergeben.

```

UChar* convertcharToUChar( char* name, UChar* value)
{
int lenName= strlen(name);
u_charsToUChars(name, value, lenName );
value[ lenName]=0;
return value;
}

// Declarations
Server *server = NULL;
Message *request = NULL;
DataTable *dataTable = NULL;
DataTable *returnDataTable= NULL;
Message *reply = NULL;
Service *service = NULL;
int nRet;
DataRow* newDataRow;
UChar name[128];
UChar value[128];
UChar** columnNames;
DataRow** rows;
DataRow* dataRow;
int i, j;
UChar* columnValue;

```

```

UChar*   errorMsg;

//Create Server
server = createServer();

//Set server connection properties
setConnectionProperty(server, convertcharToUChar( SERVER_HOST, name)
, convertcharToUChar( "localhost", value));
setConnectionProperty(server, convertcharToUChar( SERVER_PORT, name)
, convertcharToUChar( "10119", value));
setConnectionProperty(server, convertcharToUChar(
SERVER_CONNECTION_TYPE, name) , convertcharToUChar( "SOCKET", value));

setConnectionProperty(server, convertcharToUChar( SERVER_ACCOUNT_ID,
name) , convertcharToUChar( "guest", value));
setConnectionProperty(server, convertcharToUChar(
SERVER_ACCOUNT_PASSWORD, name) , convertcharToUChar( "", value));

//Connect to server
nRet = serverConnect(server);
if(nRet != SUCCESSFUL_RETURN)
{
// error handling
errorMsg = getErrorMessage(nRet);
// free memory
if(server)
nRet = deleteServer(server);
return ;
}

//Get Service From Server
service = getServiceFromServer(server,convertcharToUChar(
"ValidateAddress", name));

//Create Input Message
request = createMessage();

//Fill DataTable in the input message
dataTable = getDataTable(request);
addColumn( dataTable,convertcharToUChar( "AddressLine1", name),
&nRet);
addColumn( dataTable,convertcharToUChar( "City", name), &nRet);
addColumn( dataTable,convertcharToUChar( "PostalCode", name), &nRet);

addColumn( dataTable,convertcharToUChar( "StateProvince", name),
&nRet);

newDataRow = newRow( dataTable );

setByIndex (newDataRow, 0 , convertcharToUChar( "74, Rue Octave
Bénard", name) );
setByIndex (newDataRow, 1 , convertcharToUChar( "Etang-Salé-les-

```



```

Bains", name) );
    setByIndex (newDataRow, 2 , convertcharToUChar( "97427", name) );
    setByIndex (newDataRow, 3 , convertcharToUChar( "Reunion Island",
name) );

    addRow( dataTable, newRow);

    //Set"option" Properties to the Input Message
    nRet = putOption(request, convertcharToUChar( "OutputCasing", name),
convertcharToUChar( "M", value));
    nRet = putOption(request, convertcharToUChar( "OutputRecordType",
name), convertcharToUChar( "A", value));

    //Process Input Message, return output Message
    nRet = processMessage(service, request, &reply);
    if(nRet != SUCCESSFUL_RETURN)
    {
        // error handling
        errorMsg = getErrorMessage(nRet);
        // free memory
        if(request)
            nRet = deleteMessage(request);
        if(reply)
            nRet = deleteMessage(reply);
        if(server)
            nRet = deleteServer(server);

        return ;
    }

    //Disconnect from server
    nRet = serverDisconnect(server);

    //Get the result from the response message
    returnDataTable = getDataTable(reply );
    columnNames = getColumnNames(returnDataTable);
    rows = getDataRows( dataTable);
    for( i=0; i < getRowCount( dataTable); i++)
    {
        dataRow = rows[i];
        for(j=0; j < getColumnCount(dataTable); j++)
        {
            columnValue = (UChar*)getByIndex( dataRow, j);
        }
    }

    //Free Memory
    if(request)
        nRet = deleteMessage(request);
    if(reply)
        nRet = deleteMessage(reply);

```

```
if(server)
nRet = deleteServer(server);
```

Server

Die Struktur `Server` wird verwendet, um eine Verbindung zum Server herzustellen, die Verbindung zum Server zu trennen und den Dienst vom Server abzurufen.

CreateServer

Erstellt den Server.

Syntax

```
Server* createServer()
```

Parameter

Keine.

Ergebnis

Der Server wurde erstellt.

Beispiel

```
Server *server = NULL;
//Create Server
server = createServer();
```

DeleteServer

Löscht den Server.

Syntax

```
int deleteServer(Server* server)
```

Parameter

- Server: Die ist der zu löschende Server.

Ergebnis

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

```
int nRet;
nRet = deleteServer(server);
```

SetConnectionProperty

Legt die Konfigurationseigenschaften der Serververbindung fest, z. B. den Servernamen und die Timeout-Dauer.

Syntax

ASCII-Version

```
int setConnectionProperty(Server* server, const char* name, const char*
value)
```

Unicode-Version

```
int setConnectionProperty(Server* server, const UChar* name, const UChar*
value)
```

Parameter

- Server: Dies ist der Server, zu dem sich der Client verbindet.
- Name: Dies ist der Name der Verbindungseigenschaft wie HOST.
- Value: Dies ist der Wert für den Namen der Verbindungseigenschaft wie „www.myhost.com“

Ergebnis

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

ASCII-Version

```
int nRet;
Server *server = NULL;
nRet = createServer(&server);
```

```
nRet = setConnectionProperty(server, SERVER_HOST,
"localhost");
```

Unicode-Version

```
int nRet;
// construct 16-bit string
UChar serverHost[32];
char* SERVER_HOST= SERVER_HOST;
u_charsToUChars(SERVER_HOST, serverHost, strlen(SERVER_HOST));
serverHost [ strlen(SERVER_HOST)]=0;
// construct 16-bit string
UChar hostValue [32];
char* value= "localhost";
u_charsToUChars(value, hostValue, strlen(value));
hostValue[ strlen(value)]=0;
nRet = setConnectionProperty(server, serverHost , hostValue);
```

ServerConnect

Liest die Eigenschaften, um die Konfigurationseinstellungen zu ermitteln, und baut eine Verbindung zum Server auf.

Anmerkung: C verwendet das Serververbindungsprotokoll HTTP, HTTPS oder SOCKET. HTTP und HTTPS bauen logisch eine Client-Verbindung auf, stellen jedoch erst eine tatsächliche Verbindung zum Server her, wenn eine GetService- oder Process-Methode aufgerufen wird. Das SOCKET-Protokoll baut eine Verbindung zum Server auf, wenn Connect aufgerufen wird.

Syntax

```
int serverConnect(Server* server)
```

Parameter

- Server: Dies ist der Server, zu dem sich der Client verbindet.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

```
int nRet;
nRet = serverConnect(server);
```

ServerDisconnect

Trennt die Verbindung zum Server.

Syntax

```
int serverDisconnect(Server* server)
```

Parameter

- Server: Dies ist der Server, dessen Verbindung mit dem Clients getrennt werden soll.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

```
int nRet;  
nRet = serverDisconnect(server);
```

GetServiceFromServer

Ruft den Dienst vom Server ab.

Syntax

ASCII-Version

```
Service* getServiceFromServer(Server* server, const char* serviceName )
```

Unicode-Version

```
Service* getServiceFromServer(Server* server, const UChar* serviceName  
)
```

Parameter

- Server: Dies ist der Server, von dem sich der Client: aus verbindet.
- ServiceName: Dies ist der Name des vom Client angeforderten Diensts.

Ergebnisse

Dienst wird zurückgegeben.

Beispiel

ASCII-Version

```

Server *server= NULL;
Service *service = NULL;
//Create Server
server = createServer();
...
// get Service From Server
service = getServiceFromServer(server, "ValidateAddress" );

```

Unicode-Version

```

// construct 16-bit string
UChar serviceName[32];
char* sName="ValidateAddress";
u_charsToUChars(sName, serviceName, strlen(sName));
serviceName [ strlen(sName)]=0;
service = getServiceFromServer(server , serviceName );

```

Service

Die Struktur `Service` wird verwendet, um die Nachricht zu verarbeiten (anders ausgedrückt, sendet sie die Nachricht zum Server und empfängt eine Antwort vom Server).

ProcessMessage

Verarbeitet die Eingabennachricht und ruft die Antwortnachricht vom Server ab.

Anmerkung: Sie müssen `DeleteMessage()` aufrufen, um Speicherplatz freizugeben, wenn die zurückgegebene Nachricht nicht mehr verwendet wird.

Syntax

```

int processMessage (Service* service, Message* request, Message*
returnVal)

```

Parameter

- Service: Dies ist der vom Client: angeforderte Dienst.
- Request: Dies ist die Eingabemessage, die die Optionseinstellungen und das Dataset enthält.
- returnVal: gibt die Antwortmessage vom Server zurück.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

```

Message *request = NULL;
Message *reply = NULL;
int nRet;

...
// Assume that service is given here
// Create Input Message
request = createMessage();
... more code to fill dataTable information in request message
//Process Input Message, return output Message
nRet = processMessage(service, request, &reply);
if(nRet != SUCCESSFUL_RETURN)
{
printf("Error Occurred, " );
printf(getErrorMessage(nRet));
return ;
}
if(request)
nRet = deleteMessage(request);
if(reply)
nRet = deleteMessage(reply);

```

Message

Die Struktur `Message` versendet Ihre Eingabedaten an den Dienst und empfängt Ihre Ausgabedaten von ihm. Die Eigenschaften für `Message` beinhalten Kontexteigenschaften wie Konto-ID, Kontokennwort, Dienstname und Dienstmethode ID, Optionseigenschaften, die die dienstspezifischen Laufzeioptionen repräsentieren, und die Fehlereigenschaften Fehlerklasse, Fehlermeldung und Fehler-Stacktrace.

CreateMessage

Erstellt eine Nachricht.

Syntax

```
Message* createMessage()
```

Parameter

Keine.

Ergebnisse

Die erstellte Nachricht.

Beispiel

```
Message* request = NULL;  
request = createMessage();
```

DeleteMessage

Löscht die Nachricht.

Syntax

```
int deleteMessage(Message* message)
```

Parameter

- Message: Dies ist die zu löschende Nachricht.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

```
int nRet = deleteMessage(message);
```

GetContext

Ruft den Wert der Kontextentität ab, der durch den Namen in der Kontextsitzung der Nachricht identifiziert wurde. Kontextentitäten enthalten die folgenden Konstanten: Konto-ID, Kontokennwort, Dienstname und Dienstmethode.

Syntax

ASCII-Version

```
const char* getContext(Message* message, const char* name)
```

Unicode-Version

```
const UChar * getContext(Message* message, const UChar* name)
```

Parameter

- Message: Dies ist die Nachricht, auf die diese Funktion angewendet wird.
- Name: Der Name, dessen zugeordneter Wert zurückgegeben werden soll

Ergebnis

Gibt den Wert für den Namen in der Kontextentität zurück. Wenn der Name nicht vorhanden ist, gibt die Methode eine leere Zeichenfolge zurück.

Beispiel

ASCII-Version

```
const char* value = getContext(message, "account.id");
```

Unicode-Version

```
UChar* value;
// construct 16-bit string
UChar accountID[32];
char* account="account.id";
u_charsToUChars(account, accountID, strlen(account));
accountID[ strlen(account)]=0;
value = getContext(message, accountID);
```

GetContextMap

Ruft die Zuordnung ab, die alle Kontexteinträge enthält.

Syntax

ASCII-Version

```
MAP_STRING**getContextMap(Message* message)
Where the MAP_STRING is defined by
typedef struct map_string{
char* key;
```

```
char* value;
}MAP_STRING;
```

Unicode-Version

```
MAP_STRING**getContextMap(Message* message)
Where the MAP_STRING is defined by
typedef struct map_string{
UChar* key;
UChar* value;
}MAP_STRING;
```

Parameter

- **Message:** Dies ist die Nachricht, auf die diese Funktion angewendet wird.

Ergebnisse

Gibt ein Array vom Typ MAP_STRING zurück, das alle Kontexteinträge enthält.

Beispiel

ASCII-Version

```
int i;
char* name;
char* value;
MAP_STRING** mapping;
mapping = getContextMap( message);
i=0;
while(mapping[i] != NULL)
{
name= mapping[i]->key;
value = mapping[i]->value;
i++;
}
```

Unicode-Version

```
int i;
UChar* name;
UChar* value;
MAP_STRING** mapping;
mapping = getContextMap( message);
i=0;
while(mapping[i] != NULL)
{
name= mapping[i]->key;
value = mapping[i]->value;
i++;
}
```

PutContext

Legt den Wert für den angegebenen Namen in den „Kontexteigenschaften“ fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt.

Kontexteigenschaften enthalten die folgenden Konstanten: Konto-ID, Kontokennwort, Dienstname, Dienstschlüssel und Anforderungs-ID.

Syntax

ASCII-Version

```
int putContext(Message* message, const char* name,
              const char* value)
```

Unicode-Version

```
int putContext(Message* message, const UChar* name,
              const UChar* value)
```

Parameter

- **Message:** Dies ist die Nachricht, auf die diese Funktion angewendet wird.
- **Name:** Dies ist der Name, dem der angegebene Wert zugeordnet werden soll.
- **Value:** Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

ASCII-Version

```
int nRet;
Message* message = createMessage();
nRet = putContext( message, "account.id", "user1" );
```

Unicode-Version

```
int nRet;
Message* message;
// construct 16-bit string
UChar accountID[32];
char* account="account.id";
UChar accountIDValue[32];
char* accountValue="user1";
u_charsToUChars(account, accountID, strlen(account));
accountID [ strlen(account)]=0;
```

```

u_charsToUChars(accountValue, accountIDValue, strlen(accountValue));
accountIDValue [ strlen(accountValue)]=0;
message = createMessage();
nRet = putContext( message, accountID, accountIDValue);

```

PutContextMap

Fügt den aktuellen Kontexteigenschaften die neuen Kontexteigenschaften hinzu.

Syntax

```
int putContextMap(Message* message, MAP_STRING** context)
```

Parameter

- **Message:** Dies ist die Nachricht, auf die diese Funktion angewendet wird.
- die neue Kontextzuordnung, die der aktuellen Kontextzuordnung hinzugefügt werden soll

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

ASCII-Version

```

MAP_STRING** mapping;
Message* message;
message = createMessage();
int nRet;
mapping = (MAP_STRING **)malloc(3 * sizeof(MAP_STRING *));
mapping[0] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[0]->key = "key1" ;
mapping[0]->value = "value1" ;
mapping[1] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[1]->key = "key2" ;
mapping[1]->value = "value2" ;
mapping[2] = NULL;
nRet = putContextMap( message, mapping) ;

```

Unicode-Version

```

MAP_STRING** mapping;
Message* message;
int nRet;
UChar key1[32];
char* key1String="key1";
UChar value1[32];

```

```

char* value1String="value1";

u_charsToUChars(key1String, key1, strlen(key1String));
key1[ strlen(key1String)]=0;
u_charsToUChars(value1String, value1, strlen(value1String));
value1[ strlen(value1String)]=0;

message = createMessage();
mapping = (MAP_STRING **)malloc(2 * sizeof(MAP_STRING *));
mapping[0] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[0]->key = key1;
mapping[0]->value = value1 ;
mapping[1] = NULL;
nRet = putContextMap( message, mapping) ;

```

SetContextMap

Überschreibt die aktuellen Kontexteigenschaften mit den neuen Kontexteigenschaften.

Syntax

```
int setContextMap(Message* message, MAP_STRING** context)
```

Parameter

- Message: Dies ist die Nachricht, auf die diese Funktion angewendet wird.
- die neue Kontextzuordnung, die die aktuelle Kontextzuordnung ersetzen soll

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

ASCII-Version

```

MAP_STRING** mapping;
Message* message;
int nRet;
message = createMessage();
mapping = (MAP_STRING **)malloc(2 * sizeof(MAP_STRING *));
mapping[0] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[0]->key = "key1" ;
mapping[0]->value = "value1" ;
mapping[1] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[1]->key = "key2" ;
mapping[1]->value = "value2" ;

```

```
mapping[2] = NULL;
nRet=setContextMap( message, mapping) ;
```

Unicode-Version

```
MAP_STRING** mapping;
Message* message;
int nRet;
UChar key1[32];
char* key1String="key1";
UChar value1[32];
char* value1String="value1";
u_charsToUChars(key1String, key1, strlen(key1String));
key1[ strlen(key1String)]=0;
u_charsToUChars(value1String, value1, strlen(value1String));
value1[ strlen(value1String)]=0;
message = createMessage();
mapping = (MAP_STRING **)malloc(2 * sizeof(MAP_STRING *));
mapping[0] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[0]->key = key1 ;
mapping[0]->value = value1 ;
mapping[1] = NULL;
nRet=setContextMap( message, mapping) ;
```

GetOption

Ruft den Wert der Optionsentität ab, der durch den Namen in der Optionssitzung der Nachricht identifiziert wurde. „Optionsentitäten“ enthalten die dienstspezifischen Laufzeitoptionen wie Groß-/Kleinschreibung der Ausgabe, Ausgabedatenformat usw.

Syntax

ASCII-Version

```
const char* getOption(Message* message, const char* name)
```

Unicode-Version

```
const UChar* getOption(Message* message, const UChar* name)
```

Parameter

- **Message:** Dies ist die Nachricht, auf die diese Funktion angewendet wird.
- **Name:** Dies ist der Name, dessen zugehöriger Wert zurückgegeben werden soll

Ergebnisse

Gibt den Wert für den Namen in der Eigenschaft „Option“ der Nachricht zurück. Wenn der Name nicht vorhanden ist, wird eine leere Zeichenfolge zurückgegeben.

Beispiel

ASCII-Version

```
const char* value = getOption (message, " OutputCasing");
```

Unicode-Version

```
UChar* value;
// construct 16-bit string
UChar option[32];
char* optionValue="OutputCasing";
u_charsToUChars(optionValue, option, strlen(optionValue));
option [ strlen(optionValue)]=0;
value = getOption(message, option);
```

GetOptions

Ruft die Zuordnung ab, die alle Optionseinträge enthält.

Syntax

```
MAP_STRING**      getOptions(Message* message)
```

Parameter

- Message: Dies ist die Nachricht, auf die diese Funktion angewendet wird.

Ergebnisse

Gibt ein Array vom Typ MAP_STRING zurück, das alle Kontexteinträge enthält.

Beispiel

ASCII-Version

```
int i;
char* name;
char* value;
MAP_STRING** mapping;
mapping = getOptions( message);
i=0;
while (mapping[i] != NULL)
```

```
{
name= mapping[i]->key;
value = mapping[i]->value;
i++;
}
```

Unicode-Version

```
int i;
UChar* name;
UChar* value;
MAP_STRING** mapping;
mapping = getOptions( message);
i=0;
while(mapping[i] != NULL)
{
name= mapping[i]->key;
value = mapping[i]->value;
i++;
}
```

PutOption

Legt den Wert für den angegebenen Namen in den „Optionseigenschaften“ fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt. Bei Optionseigenschaften handelt es sich um die dienstspezifischen Laufzeitoptionen.

Syntax

ASCII-Version

```
int putOption(Message* message, const char* name,
const char* value)
```

Unicode-Version

```
int putOption(Message* message, const UChar* name,
const UChar* value)
```

Parameter

- **Message:** Dies ist die Nachricht, auf die diese Funktion angewendet wird.
- **Name:** Dies ist der Name, dem der angegebene Wert zugeordnet werden soll.
- **Value:** Dies ist der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

ASCII-Version

```
int nRet;
Message* message = createMessage();
nRet = putOption( message, "OutputCasing", "M");
```

Unicode-Version

```
int nRet;
Message* message;
// construct 16-bit string
UChar option[32];
char* optionString="OutputCasing";

UChar optionValue[32];
char* optionValueString="M";

u_charsToUChars(optionString, option, strlen(optionString));
option[ strlen(optionString) ]=0;

u_charsToUChars(optionValueString, optionValue,
strlen(optionValueString));
optionValue [ strlen(optionValueString) ]=0;

message = createMessage();
nRet = putOption( message, option, optionValue);
```

PutOptions

Fügt die neuen Optionseigenschaften zu den aktuellen Optionseigenschaften hinzu.

Syntax

```
int putOptions(Message* message, MAP_STRING** context)
```

Parameter

- **Message:** Dies ist die Nachricht, auf die diese Funktion angewendet wird.
- Die neue Optionszuordnung, die den aktuellen Optionseigenschaften hinzugefügt werden soll.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

ASCII-Version

```

MAP_STRING** mapping;
Message* message;
message = createMessage();
int nRet;
mapping = (MAP_STRING **)malloc(3 * sizeof(MAP_STRING *));
mapping[0] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[0]->key = "key1" ;
mapping[0]->value = "value1" ;
mapping[1] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[1]->key = "key2" ;
mapping[1]->value = "value2" ;
mapping[2] = NULL;
nRet = putOptions( message, mapping) ;

```

Unicode-Version

```

MAP_STRING** mapping;
Message* message;
int nRet;
UChar key1[32];
char* key1String="key1";
UChar value1[32];
char* value1String="value1";
u_charsToUChars(key1String, key1, strlen(key1String));
key1[ strlen(key1String)]=0;
u_charsToUChars(value1String, value1, strlen(value1String));
value1[ strlen(value1String)]=0;
message = createMessage();
mapping = (MAP_STRING **)malloc(2 * sizeof(MAP_STRING *));
mapping[0] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[0]->key = key1;
mapping[0]->value = value1 ;
mapping[1] = NULL;
nRet = putOptions ( message, mapping) ;

```

SetOptions

Überschreibt die aktuellen Optionseigenschaften mit den neuen Optionseigenschaften.

Syntax

```
int setOptions(Message* message, MAP_STRING** context)
```

Parameter

- Message: Dies ist die Nachricht, auf die diese Funktion angewendet wird.
- Die neue Optionszuordnung, durch die die aktuelle Optionszuordnung ersetzt werden soll.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

ASCII-Version

```
MAP_STRING** mapping;
Message* message;
int nRet;
message = createMessage();
mapping = (MAP_STRING **)malloc(3 * sizeof(MAP_STRING *));
mapping[0] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[0]->key = "key1" ;
mapping[0]->value = "value1" ;
mapping[1] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[1]->key = "key2" ;
mapping[1]->value = "value2" ;
mapping[2] = NULL;
nRet=setOptions( message, mapping) ;
```

Unicode-Version

```
MAP_STRING** mapping;
Message* message;
int nRet;
UChar key1[32];
char* key1String="key1";
UChar value1[32];
char* value1String="value1";
u_charsToUChars(key1String, key1, strlen(key1String));
key1[ strlen(key1String)]=0;
u_charsToUChars(value1String, value1, strlen(value1String));
value1[ strlen(value1String)]=0;
message = createMessage();
mapping = (MAP_STRING **)malloc(2 * sizeof(MAP_STRING *));
mapping[0] = (MAP_STRING *)malloc( sizeof(MAP_STRING));
mapping[0]->key = key1 ;
mapping[0]->value = value1 ;
mapping[1] = NULL;
nRet= setOptions ( message, mapping) ;
```

GetError

Ruft die Fehlermeldung aus der Nachricht ab.

Syntax

ASCII-Version

```
const char* getError(Message* message )
```

Unicode-Version

```
const UChar* getError(Message* message )
```

Parameter

- Message: Dies ist die Nachricht, auf die diese Funktion angewendet wird.

Ergebnis

Gibt die Fehlermeldung für die Nachricht zurück.

Beispiel

ASCII-Version

```
const char* error = getError(message );
```

Unicode-Version

```
const UChar* error = getError(message );
```

GetDataTable

Ruft die Datentabelle (DataTable) in der Nachricht ab.

Syntax

```
DataTable* getDataTable(Message* message )
```

Parameter

- Message: Dies ist die Nachricht, auf die diese Funktion angewendet wird.

Beispiel

```
// Assume that message is given here
DataTable *dataTable ;
dataTable = getDataTable( message );
```

DataTable

DataTable enthält die Datensätze für die Eingabe- und Ausgabedaten.

CreateDataTable

Erstellt die Datentabelle (DataTable).

Syntax

```
DataTable* createDataTable ()
```

Ergebnisse

Gibt die erstellte Datentabelle (DataTable) zurück.

Beispiel

```
DataTable* dataTable;
dataTable = createDataTable ();
```

DeleteDataTable

Löscht die Datentabelle (DataTable).

Syntax

```
int deleteDataTable(DataTable* dataTable)
```

Parameter

- DataTable: die zu löschende Datentabelle (DataTable)

Beispiel

```
DataTable* dataTable;
dataTable = createDataTable();
...
if(dataTable) deleteDataTable(dataTable);
```

AddColumn

Fügt eine neue Spalte hinzu.

Syntax

ASCII-Version

```
int addColumn(DataTable* dataTable, const char* columnName,
int* indexReturn)
```

Unicode-Version

```
int addColumn(DataTable* dataTable, const UChar* columnName,
int* indexReturn)
```

Parameter

- **DataTable**: Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.
- **der zur Datentabelle (DataTable) hinzuzufügende Spaltenname**
- **gibt den zugehörigen Index zurück**

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Ausnahmen

- leerer Spaltenname
- doppelter Spaltenname

Beispiel

ASCII-Version

```
int nIndex;
int nRet;
nRet= addColumn( dataTable, "AddressLine1", &nIndex);
nRet= addColumn( dataTable, "City",&nIndex);
nRet= addColumn( dataTable, "State", &nIndex);
if(nRet != SUCCESSFUL_RETURN)
```

```
{
printf(getErrorMessage(nRet));
return ;
}
```

Unicode-Version

```
int nRet;
int nIndex;
UChar* error;
UChar city[64];
char* cityString= "City"
u_charsToUChars(cityString, city, strlen(cityString));
city[ strlen(cityString)]=0;

nRet= addColumn( dataTable, city,&nIndex);
if(nRet != SUCCESSFUL_RETURN)
{
error = getErrorMessage(nRet);
//more code
}
```

GetColumnNames

Ruft alle Spaltennamen ab.

Syntax

ASCII-Version

```
char** getColumnNames(dataTable )
```

Unicode-Version

```
UChar** getColumnNames(dataTable )
```

Parameter

- DataTable: Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.

Ergebnisse

Gibt das Array mit Spaltennamen zurück.

Beispiel

ASCII-Version

```
char* value;
char** columnNames;
int i;
columnNames =getColumnNames ( dataTable) ;
for( i=0; i < getColumncount( dataTable); i++)
{
value = columnNames[i];
}
```

Unicode-Version

```
UChar* value;
UChar** columnNames;
int i;
columnNames =getColumnNames ( dataTable) ;
for( i=0; i < getColumncount( dataTable); i++)
{
value = columnNames[i];
}
```

GetColumnIndex

Ruft den entsprechenden Spaltenindex ab.

Syntax

ASCII-Version

```
int getColumnIndex(DataTable* dataTable ,const char* columnName)
```

Unicode-Version

```
int getColumnIndex(DataTable* dataTable ,const UChar* columnName)
```

Parameter

- DataTable: Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.
- Spaltenname

Ergebnisse

Gibt den entsprechenden Spaltenindex zurück.

Beispiel

ASCII-Version

```
int nIndex ;
nIndex = getColumnIndex(dataTable , "AddressLine1")
```

Unicode-Version

```
int nIndex ;
UChar columnName[64];
char* columnNameStr= "AddressLine1" u_charsToUChars(columnNameStr,
columnName, strlen(columnNameStr));
columnName [strlen(columnNameStr)]=0;
nIndex = getColumnIndex(dataTable , columnName);
```

GetColumnCount

Ruft die Anzahl der Spalten ab.

Syntax

```
int getColumnCount(DataTable* dataTable )
```

Parameter

- **DataTable:** Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.

Ergebnisse

Gibt die Anzahl der Spalten zurück.

Beispiel

```
// Assume that dataTable is given here int nColumnCount ;
nColumnCount = getColumnCount( dataTable ) ;
```

Clear

Löscht die Daten in der Datentabelle (DataTable).

Syntax

```
int clear(DataTable* dataTable)
```

Parameter

- **DataTable**: Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

```
// Assume that dataTable is given here
clear(dataTable);
```

GetDataRows

Ruft ein Array mit allen Datenzeilen (DataRow) in der Datentabelle (DataTable) ab.

Syntax

```
DataRow** getDataRows(DataTable* dataTable)
```

Parameter

- **DataTable**: Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.

Ergebnisse

Gibt eine Array mit Datenzeilen (DataRow) zurück.

Beispiel

```
// Assume that dataTable is given here
DataRows** rows;
DataRow* dataRow;
int i;
int j;
rows = getDataRows( dataTable);
for( i=0; i < getRowCount( dataTable); i++)
{
    dataRow = rows[i];

    for(j=0; j < getColumnCount( dataTable); j++)
    {
        value = (char*)getByIndex( dataRow, j);
    }
}
```

AddRow

Fügt der Datentabelle (DataTable) eine Datenzeile (DataRow) hinzu.

Syntax

```
int addRow(DataTable* dataTable, DataRow* dataRow)
```

Parameter

- **DataTable:** Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.
- **DataRow:** Dies ist die zur Datentabelle (DataTable) hinzuzufügende Datenzeile (DataRow).

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

```
// Assume that dataTable is given here DataRow* newDataRow;  
int nRet;  
newDataRow = newRow( dataTable );  
setByIndex (newDataRow, 0 , "10535 Boyer Blvd");  
setByIndex (newDataRow, 1 , "Austin");  
setByIndex (newDataRow, 2 , "Texas");  
nRet = addRow( dataTable, newDataRow);
```

NewRow

Erstellt eine neue Datenzeile (DataRow) in der Datentabelle (DataTable).

Syntax

```
DataRow* newRow(DataTable* dataTable )
```

Parameter

- **DataTable:** Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.

Ergebnisse

Gibt die neu erstellte Datenzeile (DataRow) zurück.

Beispiel

```
// Assume that dataTable is given here
DataRow* newRow;
int nRet;
newDataRow = newRow( dataTable );
setByIndex (newDataRow, 0 , "10535 Boyer Blvd");
setByIndex (newDataRow, 1 , "Austin");
setByIndex (newDataRow, 2 , "Texas");
nRet = addRow( dataTable, newRow);
```

GetRowCount

Ruft die Anzahl der Datenzeilen (DataRows) in dieser Datentabelle (DataTable) ab.

Syntax

```
int getRowCount(DataTable* dataTable)
```

Parameter

- **DataTable:** Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.

Ergebnisse

Gibt die Anzahl der Datenzeilen (DataRows) in dieser Datentabelle (DataTable) zurück.

Beispiel

```
// Assume that dataTable is given here
int nRowCount ;
nRowCount = getRowCount( dataTable);
```

MergeDataTable

Führt die vorhandene Datentabelle (DataTable) und die aktuelle Datentabelle zusammen.

Syntax

```
int mergeDataTable(DataTable* dataTable ,DataTable* other )
```

Parameter

- **DataTable:** Dies ist die Datentabelle (DataTable), auf die diese Funktion angewendet wird.

- Other: Dies ist die andere Datentabelle (DataTable), die mit der aktuellen Datentabelle (DataTable) zusammengeführt werden soll.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

```
// Assume that dataTable and otherDataTable are given here  
mergeDataTable (dataTable ,otherDataTableDataRow)
```

DataRow

DataRow enthält den Datensatz für die Eingabe- und Ausgabedaten.

CreateDataRow

Erstellt die Datenzeile (DataRow).

Syntax

```
DataRow* createDataRow ()
```

Ergebnisse

Gibt die erstellte Datenzeile (DataRow) zurück.

Beispiel

```
DataRow* dataRow;  
dataRow = createDataRow ();
```

DeleteDataRow

Löscht die Datenzeile (DataRow).

Syntax

```
int deleteDataRow(DataRow* dataRow)
```

Parameter

- die zu löschende Datenzeile (DataRow)

Beispiel

```
DataRow* dataRow;
dataRow = createDataRow();
...
if(dataRow)
    deleteDataRow (dataRow);
```

GetColumnNamesFromRow

Ruft alle Spaltennamen ab.

Syntax

ASCII-Version

```
char** getColumnNamesFromRow(DataRow* dataRow)
```

Unicode-Version

```
UChar** getColumnNamesFromRow(DataRow* dataRow)
```

Parameter

- DataRow: Dies ist die Datenzeile (DataRow), auf die diese Funktion angewendet wird.

Ergebnisse

Gibt das Array mit Spaltennamen zurück.

Beispiel

ASCII-Version

```
char* value;
char** columnNames;
int i;
columnNames = getColumnNamesFromRow (dataRow) ;
for( i=0; i < getColumnCountFromRow (dataRow); i++)
{
```

```
value = columnNames[i];
}
```

Unicode-Version

```
UChar* value;
UChar** columnNames;
int i;
columnNames = getColumnNamesFromRow (dataRow) ;
for( i=0; i < getColumnCountFromRow (dataRow); i++)
{
    value = columnNames[i];
}
```

GetColumnIndexFromRow

Ruft den entsprechenden Spaltenindex ab.

Syntax

ASCII-Version

```
int getColumnIndexFromRow(DataRow* dataRow, const char* name)
```

Unicode-Version

```
int getColumnIndexFromRow(DataRow* dataRow, const UChar* name)
```

Parameter

- DataRow: Dies ist die Datenzeile (DataRow), auf die diese Funktion angewendet wird.
- Spaltenname

Ergebnisse

Gibt den entsprechenden Spaltenindex zurück.

Beispiel

ASCII-Version

```
int nIndex
nIndex = getColumnIndexFromRow ("AddressLine1");
```

Unicode-Version

```
int nIndex
UChar columnName[64];
```

```
char* columnNameStr= "AddressLine1"
u_charsToUChars(columnNameStr, columnName, strlen(columnNameStr));
columnName [strlen(columnNameStr)]=0;
nIndex = getColumnIndexFromRow (columnName);
```

GetColumnCountFromRow

Ruft die Anzahl der Spalten ab.

Syntax

```
int getColumnCountFromRow(DataRow* dataRow )
```

Parameter

- DataRow: Dies ist die Datenzeile (DataRow), auf die diese Funktion angewendet wird.

Ergebnisse

Gibt die Anzahl der Spalten zurück.

Beispiel

```
//Assume that the dataRow is given here
int nColumnCount ;
nColumnCount = getColumnCountFromRow (dataRow );
```

GetByIndex

Ruft den Wert aus dem Feld-Array nach dem Spaltenindex in dieser Datenzeile (DataRow) ab.

Syntax

ASCII-Version

```
const char* getByIndex(DataRow* dataRow, int index)
```

Unicode-Version

```
const UChar* getByIndex(DataRow* dataRow, int index)
```

Parameter

- DataRow: Dies ist die Datenzeile (DataRow), auf die diese Funktion angewendet wird.

- Index: Dies ist der Index, dem der angegebene Wert zugeordnet werden soll.

Ergebnisse

Gibt den Wert für den Spaltenindex in der Datenzeile (DataRow) zurück, und gibt eine leere Zeichenfolge zurück, wenn der Index ungültig ist.

Beispiel

ASCII-Version

```
char* value = getByIndex( dataRow, 0 );
```

Unicode-Version

```
UChar* value = getByIndex( dataRow, 0 );
```

GetByName

Ruft den Wert aus dem Feld-Array nach dem Spaltennamen in dieser Datenzeile (DataRow) ab.

Syntax

ASCII-Version

```
const char* getByName(DataRow* dataRow, const char* name )
```

Unicode-Version

```
const UChar* getByName(DataRow* dataRow, const UChar* name )
```

Parameter

- DataRow: Dies ist die Datenzeile (DataRow), auf die diese Funktion angewendet wird.
- Name, unter dem der angegebene Wert zugeordnet werden soll.

Ergebnisse

Gibt den Wert für den Spaltennamen in der Datenzeile (DataRow) zurück, und gibt eine leere Zeichenfolge zurück, wenn der Spaltenname nicht vorhanden ist.

Beispiel

ASCII-Version

```
char* value = getByName ( dataRow, "City" )
```

Unicode-Version

```
UChar* value;
UChar columnName[64];
char* columnNameStr= "City"
u_charsToUChars(columnNameStr, columnName, strlen(columnNameStr));
columnName [strlen(columnNameStr)]=0;
value = getByName ( dataRow, columnName);
```

MergeDataRow

Führt die angegebene Datenzeile (DataRow) mit der aktuellen Datenzeile zusammen.

Syntax

```
int mergeDataRow(DataRow* dataRow, DataRow* other)
```

Parameter

- DataRow: Dies ist die Datenzeile (DataRow), auf die diese Funktion angewendet wird.
- Die andere Datenzeile (DataRow), die mit der aktuellen Datenzeile (DataRow) zusammengeführt werden soll.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

```
//Assume that the dataRow and otherDataRow are given here
int nRet;
nRet= mergeDataRow(dataRow, otherDataRow);
```

SetByName

Legt den Wert für die entsprechende Spalte der Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.

Syntax

ASCII-Version

```
int setByName(DataRow* dataRow, const char* name, const char* value)
```

Unicode-Version

```
int setByName(DataRow* dataRow, const UChar* name, const
UChar* value)
```

Parameter

- DataRow: Dies ist die Datenzeile (DataRow), auf die diese Funktion angewendet wird.
- Name, unter dem der angegebene Wert zugeordnet werden soll.
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ausnahmen

Wenn ein leerer oder ein doppelter Spaltenname angegeben wird, wird ein Fehler zurückgegeben.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

ASCII-Version

```
int nRet;
nRet= setByName (dataRow, "City", "Austin");
if(nRet != SUCCESSFUL_RETURN)
{ printf(getErrorMessage(nRet));
//more code
}
```

Unicode-Version

```
int nRet;
UChar* error;
UChar columnName[64];
char* columnNameStr= "City"
UChar columnValue[64];
char* columnValueStr= "Austin";
u_charsToUChars(columnNameStr, columnName, strlen(columnNameStr));
columnName [strlen(columnNameStr)]=0;
u_charsToUChars(columnValueStr, columnValue, strlen(columnValueStr));
columnValue [strlen(columnValueStr)]=0;
nRet= setByName (dataRow, columnName, columnValue);
if(nRet != SUCCESSFUL_RETURN)
{ error = getErrorMessage(nRet);
//more code
}
```

SetByIndex

Legt den Wert für die entsprechende Spalte der Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.

Syntax

ASCII-Version

```
int setByIndex(DataRow* dataRow, int index, const char* value)
```

Unicode-Version

```
int setByIndex(DataRow* dataRow, int index, const UChar* value)
```

Parameter

- DataRow: Dies ist die Datenzeile (DataRow), auf die diese Funktion angewendet wird.
- Spaltenindex, dem der angegebene Wert zugeordnet werden soll
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ausnahmen

- Der Spaltenindex ist ungültig.

Ergebnisse

Gibt bei Erfolg 0, ansonsten den Fehlercode zurück.

Beispiel

ASCII-Version

```
int nRet;
nRet= setByIndex (dataRow, 1, "Austin");
if(nRet != SUCCESSFUL_RETURN)
{
printf(getErrorMessage(nRet));
//more code
}
```

Unicode-Version

```
int nRet;
UChar* error;
UChar columnValue[64];
char* columnValueStr= "Austin";
u_charsToUChars(columnValueStr, columnValue, strlen(columnValueStr));
```

```
columnValue [strlen(columnValueStr)]=0;
nRet= setByIndex (dataRow, 1, columnValue);
if(nRet != SUCCESSFUL_RETURN)
{
error = getErrorMessage (nRet);
//more code
}
```

AddChild

Fügt der benannten übergeordneten/untergeordneten Beziehung eine Datenzeile (DataRow) hinzu. Wenn die benannte Beziehung vorhanden ist, wird die angegebene Datenzeile (DataRow) der vorhandenen Datenzeilensammlung angefügt. Anderenfalls wird eine neue Sammlung mit der angegebenen Datenzeile (DataRow) als einziges Element erstellt.

Syntax

ASCII-Version

```
void addChild(DataRow* dataRow, const char* childName, DataRow*
childDataRow)
```

Unicode-Version

```
void addChild(DataRow* dataRow, const UChar* childName, DataRow*
childDataRow)
```

Parameter

- Der Name der übergeordneten/untergeordneten Beziehung (z. B. „Überschwemmungsgebietsdaten“, „Referenzen“, „Verwendet von“ usw.)
- Die Datenzeile (DataRow), die der Beziehung hinzugefügt werden soll.

Beispiel

ASCII-Version

```
DataRow* dataRow = createDataRow();
DataRow* child1DataRow1 = createDataRow();

setName(child1DataRow1, "City", "Austin");
setName(child1DataRow1, "State", "Texas");

addChild( dataRow, "child1", child1DataRow1);
```

Unicode-Version

```

UChar* convertcharToUChar( char* name, UChar* value)
{
    int lenName= strlen(name);

    u_charsToUChars(name, value, lenName );

    value[ lenName]=0;
    return value;
} >
DataRow* dataRow = createDataRow();
DataRow* child1DataRow1 = createDataRow();
UChar    name[128];
UChar    columnValue[128];
setByName(child1DataRow1, convertcharToUChar("City", name),
    convertcharToUChar("Austin", columnValue));
setByName(child1DataRow1, convertcharToUChar("State", name),
    convertcharToUChar("Texas", columnValue));
addChild( dataRow, "child1", child1DataRow1);

```

GetChildren

Ruft die untergeordneten Zeilen einer benannten Beziehung ab.

Syntax

ASCII-Version

```
DataRow** getChildren(DataRow* dataRow, const char* childName)
```

Unicode-Version

```
DataRow** getChildren(DataRow* dataRow, const UChar* childName)
```

Parameter

- Der Name der übergeordneten/untergeordneten Beziehung (z. B. „Überschwemmungsgebietsdaten“, „Referenzen“, „Verwendet von“ usw.)

Ergebnisse

Gibt die untergeordneten Zeilen aus der benannten Beziehung zurück.

Beispiel

ASCII-Version

```
DataRow** child1Rows;
child1Rows = getChildren(dataRow, "child1");
```

Unicode-Version

```
DataRow** child1Rows;
UChar childName[128];
/* see convertcharToUChar in the Example section of "addChild" */
child1Rows = getChildren(dataRow, convertcharToUChar("child1",
childName));
```

ListChildNames

Ruft alle Namen der benannten übergeordneten/untergeordneten Beziehungen.

Syntax

ASCII-Version

```
char** listChildNames(DataRow* dataRow)
```

Unicode-Version

```
UChar** listChildNames(DataRow* dataRow)
```

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

ASCII-Version

```
char** childsNames;
childsNames =listChildNames( dataRow);
```

Unicode-Version

```
UChar** childsNames;
childsNames=listChildNames( dataRow);
```

SetChildren

Legt die Zeilen einer angegebenen benannten übergeordneten/untergeordneten Beziehung fest. Wenn zuvor Zeilen mit diesem Namen vorhanden waren, werden diese an den Aufrufer zurückgegeben.

Syntax

ASCII-Version

```
DataRow** setChildren(DataRow* dataRow, const char* childName, DataRow** dataRows)
```

Unicode-Version

```
DataRow** setChildren(DataRow* dataRow, const UChar* childName, DataRow** dataRows)
```

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

ASCII-Version

```
DataRow* dataRow = createDataRow();
DataRow* child1DataRow1 = createDataRow();
DataRow* child1DataRow2 = createDataRow();
DataRow* child2DataRow = createDataRow();
DataRow** child2Rows;
DataRow** returnRows;

setByName(child1DataRow1, "Address", "200 Congress");
setByName(child1DataRow1, "City", "Austin");

setByName(child1DataRow2, "Address", "100 Congress");
setByName(child1DataRow2, "City", "Dallas");

setByName(child2DataRow, "Address", "100 Congress");
setByName(child2DataRow, "City", "Austin");

addChild( dataRow, "child1", child1DataRow1);
addChild( dataRow, "child1", child1DataRow2);
addChild( dataRow, "child2", child2DataRow );

child2Rows=getChildren(dataRow, "child2");

returnRows=setChildren( dataRow, "child1", child2Rows);
```


Unicode-Version

```

DataRow* dataRow = createDataRow();
DataRow* child1DataRow1 = createDataRow();
DataRow* child1DataRow2 = createDataRow();
DataRow* child2DataRow = createDataRow();
DataRow** child2Rows;
DataRow** returnRows;
UChar name[128];
UChar columnValue[128];
UChar childName[128];

setByName(child1DataRow1, convertcharToUChar("Address", name),
convertcharToUChar("200 Congress", columnValue));
setByName(child1DataRow1, convertcharToUChar("City", name),
convertcharToUChar("Austin", columnValue));
setByName(child1DataRow2, convertcharToUChar("Address", name),
convertcharToUChar("100 Congress", columnValue));
setByName(child1DataRow2, convertcharToUChar("City", name)
convertcharToUChar("Dallas", columnValue) );
setByName(child2DataRow, convertcharToUChar("Address", name),
convertcharToUChar("100 Congress", columnValue) );
setByName(child2DataRow, convertcharToUChar("City", name),
convertcharToUChar("Austin", columnValue) );

addChild( dataRow, convertcharToUChar("child1", childName),
child1DataRow1);
addChild( dataRow, convertcharToUChar("child1",
childName), child1DataRow2);
addChild( dataRow, convertcharToUChar("child2", childName), child2DataRow
);

child2Rows=getChildren(dataRow, convertcharToUChar("child2", childName));

returnRows=setChildren( dataRow, convertcharToUChar("child1", childName),
child2Rows);

```

3 - Die C++-API

In this section

Einführung in die C++ API	75
Server	89
Service	92
Message	93
DataTable	104
DataRow	111

Einführung in die C++ API

Die C++ API setzt sich aus den folgenden Klassen zusammen:

- Server
- Service
- Message
- DataTable
- DataRow

UnicodeString in ICU ist eine String-Klasse, die Unicode-Zeichen direkt speichert und ein ähnliches Feature wie die Java String- und StringBuffer-Klassen bietet. Die Spectrum™ Technology Platform Unicode C++ API verwendet diese Klasse zum Speichern von Unicode-Strings.

Unterstützte Bibliotheken

Spectrum™ Technology Platform stellt eine C-API in einer ASCII- und Unicode-Version bereit, wobei die Unicode-Version so kompatibel wie möglich mit dem ursprünglichen Design der ASCII-Version bleibt. Spectrum™ Technology Platform wendet International Components for Unicode (ICU) in der API an, um das Unicode-Feature zu unterstützen. Bei ICU handelt es sich um eine ausgereifte und weit verbreitete Gruppe von C/C++-Bibliotheken zur Unicode-Unterstützung, die von IBM entwickelt wurde.

Der Unicode-Standard definiert eine Standardcodierung basierend auf 16-Bit-Codeeinheiten. Dies wird in ICU durch die Definition von UChar als 16-Bit-Ganzzahltyp ohne Vorzeichen (kurze ganze Zahl ohne Vorzeichen *) unterstützt. Dies ist der Basistyp für Zeichenarrays für Zeichenfolgen in ICU. Spectrum™ Technology Platform verwendet UChar als die Unicode-Zeichenfolgendarstellung in unserer C-API.

Anmerkung: Nicht alle Dienste unterstützen den vollständigen Unicode-Zeichensatz. Der ValidateAddress-Dienst unterstützt beispielsweise den Zeichensatz nach ISO 8859-1 für US-Eingaben sowie internationale Eingaben und Ausgaben und den Zeichensatz nach CP 850 für kanadische Eingaben und Ausgaben. Die Unicode-Bibliotheken sollten jedoch immer verwendet werden, wenn Ihre Eingabedaten möglicherweise Zeichen enthalten, die nicht zum ASCII-Zeichensatz gehören, auch wenn der zugrunde liegende Dienst den vollständigen Unicode-Zeichensatz nicht unterstützt.

Detaillierte Informationen zu UChar finden Sie auf den folgenden beiden Websites:

- icu.sourceforge.net/userguide/
- www-306.ibm.com/software/globalization/icu/index.jsp

Windows

Jede API-Konfiguration erzeugt Bibliotheksdateien mit einem gemeinsamen Basisnamen (g1client), jedoch mit einem eindeutigen Suffix und gegebenenfalls Präfix (im Fall von statischen Bibliotheken „lib“). Die Bibliothekssuffixe geben Folgendes an:

```
<lib>g1client<S><U><D>.<lib|dll>
```

- lib: Gibt eine statische Bibliothek an.
- dll: Gibt eine dynamische (freigegebene) Bibliothek an.
- S: Verweist auf einen Single-Thread-Build. Wenn „S“ nicht vorhanden ist, handelt es sich um eine Multi-Thread-Version.
- U: Verweist auf einen Build der UNICODE-Version. Wenn „U“ nicht vorhanden ist, handelt es sich um ein ASCII-Build.
- D: Verweist auf einen Debug-Build. Wenn dieses Suffix nicht vorhanden ist, wird ein optimierter Release-Build angegeben.

Um die UNICODE-Version zu aktivieren, muss die Makrodefinition „LIB_UNICODE“ in Ihrem Projekt vorhanden sein.

Um die UNICODE-Version der statischen C/C++-API-Bibliothek zu verwenden, müssen Sie in Ihrem Projekt „U_STATIC_IMPLEMENTATION“ definieren.

Um die dynamische Version zu verwenden, müssen Sie in Ihrem Projekt „G1CLIENT_DLL“ definieren.

Wir stellen darüber hinaus im Verzeichnis der Header-Dateien eine Datei namens „auto_link.h“ zur Verfügung, die entsprechend den Projekteinstellungen automatisch eine Verlinkung zu allen zugehörigen Bibliotheken erstellt.

Um 64-Bit-Bibliotheken unter Windows aufzurufen, müssen Sie „VER_64“ in Ihrem Projekt definieren.

Statische Bibliothek

Anmerkung: Die in diesem Abschnitt aufgeführten Namen beziehen sich auf 32-Bit-Bibliotheken. Ersetzen Sie für 64-Bit-Bibliotheken „32“ im Bibliotheksnamen durch „64“.

Single-Thread/Release

	ASCII	Unicode
g1	libg1client_S.lib	libg1client_SU.lib
openssl	otlibeay32.lib otlibssl32.lib	otlibeay32.lib otlibssl32.lib

opentop	opentop.lib	opentopw.lib
icu		libicuuc.lib libicudt.lib libicuin.lib libicuio.lib
Poco	PocoXML32.lib	PocoXML32w.lib
Single-Thread/Debug		
	ASCII	Unicode
g1	libg1client_SD.lib	libg1client_SUD.lib
openssl	otlibey32d.lib otlibssl32d.lib	otlibey32d.lib otlibssl32d.lib
opentop	opentopd.lib	opentopwd.lib
icu		libicuucd.lib libicudtd.lib libicuind.lib libicuiod.lib
Poco	PocoXML32d.lib	PocoXML32wd.lib
Multi/Release (verwendet Multi-Threaded CRT)		
	ASCII	Unicode
g1	libg1client.lib	libg1client_U.lib
openssl	otlibey32mt.lib otlibssl32mt.lib	otlibey32mt.lib otlibssl32mt.lib
opentop	opentopmt.lib	opentopmtw.lib

icu		libicuucmt.lib libicudtmt.lib libicuinmt.lib libicuiomt.lib
-----	--	--

Poco	PocoXMLmt32.lib	PocoXML32mtw.lib
------	-----------------	------------------

Multi/Debug (verwendet Multi-Threaded CRT)

	ASCII	Unicode
--	-------	---------

g1	libg1client_D.lib	libg1client_UD.lib
----	-------------------	--------------------

openssl	otlibeay32mtd.lib otlibssl32mtd.lib	otlibeay32mtd.lib otlibssl32mtd.lib
---------	-------------------------------------	-------------------------------------

opentop	opentopmtd.lib	opentopmtwd.lib
---------	----------------	-----------------

icu		libicuucmtd.lib libicudtmtd.lib libicuinmtd.lib libicuiomtlib
-----	--	--

Poco	PocoXMLmt32d.lib	PocoXML32mtwd.lib
------	------------------	-------------------

Dynamische Bibliothek

Anmerkung: Die in diesem Abschnitt aufgeführten Namen beziehen sich auf 32-Bit-Bibliotheken. Ersetzen Sie für 64-Bit-Bibliotheken „32“ im Bibliotheksnamen durch „64“.

Multi/Release (verwendet Multi-Threaded CRT)

	ASCII	Unicode
--	-------	---------

g1	g1client.dll	g1client_U.dll
----	--------------	----------------

openssl	otlibeay32mts.dll otlibssl32mts.dll	otlibeay32mts.dll otlibssl32mts.dll
---------	-------------------------------------	-------------------------------------

opentop	opentopmts.dll	opentopmtws.dll
---------	----------------	-----------------

icu		icuuc32.dll icuio32.dll icuin32.dll icudt32.dll
-----	--	--

Poco	PocoXML32mts.dll	PocoXML32mtws.dll
------	------------------	-------------------

Multi/Debug (verwendet Multi-Threaded CRT)

ASCII

Unicode

g1	g1client_D.dll	g1client_UD.dll
----	----------------	-----------------

openssl	otlibeay32mtds.dll otlibssl32mtds.dll	otlibeay32mtds.dll otlibssl32mtds.dll
---------	---------------------------------------	---------------------------------------

opentop	opentopmtds.dll	opentopmtwds.dll
---------	-----------------	------------------

icu		icuuc32d.dll icuio32d.dll icuin32d.dll icudt32d.dll
-----	--	--

Poco	PocoXML32mtds.dll	PocoXML32mtwds.dll
------	-------------------	--------------------

UNIX

Jede ClientSDK-Konfiguration erzeugt Bibliotheksdateien mit einem gemeinsamen Basisnamen (libg1client), jedoch mit einem eindeutigen Suffix. Spectrum™ Technology Platform stellt einen Multi-Thread- und einen Release-Build für die ASCII-Version und die UNICODE-Version bereit.

Die Bibliothekssuffixe geben Folgendes an:

```
libg1client<U>.<so|sl|a>
```

- U: Verweist auf einen Build der UNICODE-Version. Wenn „U“ nicht vorhanden ist, handelt es sich um ein ASCII-Build.

Um die UNICODE-Version zu verwenden, müssen Sie in Ihrem Projekt „LIB_UNICODE“ definieren. In der C++-API der UNICODE-Version ist der Namespace für alle Klassen „g1client“.

AIX

	ASCII	Unicode
g1	libg1client.so	libg1client_U.so
openssl	libcrypto.so libssl.so	libcrypto.so libssl.so
opentop	libopentop-xlCmt.so	libopentop-xlCmtw.so libotxml-xlCmtw.so
icu		libicudata34.a libicui18n34.a libicuio34.a libicuuc34.a
Poco	libPocoXML.so	

HP-UX

	ASCII	Unicode
g1	libg1client.sl	libg1client_U.sl
openssl	libcrypto.sl libssl.sl libcrypto.sl.0.9.7 libssl.sl.0.9.7	libcrypto.sl libssl.sl libcrypto.sl.0.9.7 libssl.sl.0.9.7
opentop	libopentop-accmt.sl	libopentop-accmtw.sl libotxml-accmtw.sl
icu		libicudata.sl libicudata.sl.34 libicui18n.sl libicui18n.sl.34 libicuio.sl libicuio.sl.34 libicuuc.sl libicuuc.sl.34

Poco	libPocoXML.sl	
<hr/>		
Itanium		
	ASCII	Unicode
<hr/>		
g1	libg1client.sl	libg1client_U.sl
<hr/>		
openssl	libcrypto.a libssl.a	libcrypto.a libssl.a
<hr/>		
opentop	libopentop-accmt.sl	libopentop-accmtw.sl libotxml-accmtw.sl
<hr/>		
icu		libicudata.sl libicudata.sl.34 libicudata.sl.34.0 libicui18n.sl libicui18n.sl.34 libicui18n.sl.34.0 libicuio.sl libicuio.sl.34 libicuio.sl.34.0 libicuuc.sl libicuuc.sl.34 libicuuc.sl.34.0
<hr/>		
Poco	libPocoXML.sl	
<hr/>		
Linux		
	ASCII	Unicode
<hr/>		
g1	libg1client.so	libg1client_U.so
<hr/>		
openssl	libcrypto.so libcrypto.so.0.9.7 libssl.so libssl.so.0.9.7	libcrypto.so libcrypto.so.0.9.7 libssl.so libssl.so.0.9.7
<hr/>		
opentop	libopentop-gccmt.so	libopentop-gccmtw.so libotxml-gccmtw.so
<hr/>		

icu		libicudata.so libicudata.so.34 libicui18n.so libicui18n.so.34 libicuio.so libicuio.so.34 libicuuc.so libicuuc.so.34
Poco	libPocoXML.so	
Solaris		
	ASCII	Unicode
g1	libg1client.so	libg1client_U.so
openssl	libcrypto.so libcrypto.so.0.9.7 libssl.so libssl.so.0.9.7	libcrypto.so libcrypto.so.0.9.7 libssl.so libssl.so.0.9.7
opentop	libopentop-fortemt.so	libopentop-fortemtw.so libotxml-fortemtw.so
icu		libicudata.so libicudata.so.34 libicui18n.so libicui18n.so.34 libicuio.so libicuio.so.34 libicuuc.so libicuuc.so.34
Poco	libPocoXML.so	

Konstanten

Die C++ API verwendet zwei Konstantengruppen. Die erste Gruppe dient der `Server`-Klasse, die in der Tabelle unten beschrieben wird.

Tabelle 14: Konstanten für die `Server`-Komponente

Name der Konstante	Beschreibung/Standard	Beispiel
<code>Server::HOST</code>	Zeichenfolge für den Hostnamen des Servers. Der Standardname lautet „localhost“.	65.89.200.89
<code>Server::PORT</code>	Zeichenfolge für den Serverport. Der Standardwert ist „8080“.	10119
<code>Server::ACCOUNT_ID</code>	Zeichenfolge für die Serverkonto-ID. Kein Standardwert.	user1
<code>Server::ACCOUNT_PASSWORD</code>	Zeichenfolge für das Kennwort des Serverkontos. Kein Standardwert.	user1
<code>Server::CONNECTION_TIMEOUT</code>	Zeichenfolge für Timeout der Serververbindung in Millisekunden. Der Standardwert ist „5000“.	50000
<code>Server::CONNECTION_TYPE</code>	Zeichenfolge für den Serververbindungstyp. Derzeit werden nur HTTP, HTTPS und SOCKET unterstützt. Der Standardwert lautet „HTTP“.	HTTP(S)
<code>Server::PROXY_HOST</code>	Zeichenfolge für den Hostnamen des Proxy-Servers. Kein Standardwert.	192.168.1.77
<code>Server::PROXY_PORT</code>	Zeichenfolge für den Proxy-Serverport. Kein Standardwert.	8080
<code>Server::PROXY_USER</code>	Zeichenfolge für die Konto-ID des Proxy-Servers. Kein Standardwert.	user1
<code>Server::PROXY_PASSWORD</code>	Zeichenfolge für das Kontokennwort des Proxy-Servers. Kein Standardwert.	user1

Die zweite Konstantengruppe gehört zur `Message`-Klasse:

Tabelle 15: Konstanten für die `Message`-Komponente

Name der Konstante	Beschreibung	Beispiel
<code>Message::CONTEXT_ACCOUNT_ID</code>	Zeichenfolge für den Nachrichtentext der Konto-ID.	user1
<code>Message::CONTEXT_ACCOUNT_PASSWORD</code>	Zeichenfolge für das Kontokennwort des Nachrichtenkontextes.	user1
<code>Message::CONTEXT_SERVICE_NAME</code>	Zeichenfolge für den Dienstnamen des Nachrichtenkontextes.	echoservice

Fehlermeldungen

Verwenden Sie die Exception-Klasse, um Fehlermeldungen zu erhalten. Erfassen Sie die Fehlermeldung mit den TRY/CATCH-Konstrukten. Beispiel:

```
try{
    Server *server=new Server();

    //Connect to server
    server->connect();

}catch(Exception e)
{
    // ASCII Version-use the following code
    cout << "Error Occurs," << e.getErrorMessage();
    //Unicode Version -use the following code

    UnicodeString error = e.getErrorMessage();

    wcout << error.getTerminatedBuffer();
}
```

Die C++ API verwendet die folgenden Fehlermeldungen:

- Fehlermeldungen zur Verbindung:
 - "Connection type not supported"
 - "Client timeout"

- "Blank connection property name"
- "Blank property name"
- Fehlermeldungen zum Erstellen von Datentabellen (DataTables):
 - "Blank column name"
 - "Duplicated column name"
 - "The column index is invalid"
- Fehlermeldungen für die MessagePackaging-Ausnahme:
 - "Input Message is null"
 - "Failed to connect to Server"
 - "Failed to disconnect from Server"
 - "Failed to open Http Connection"
 - "Failed to get Service"
 - "Failed to package the message using Serializer and Encoding"

SmartPointer

Spectrum™ Technology Platform stellt eine Klasse mit der Bezeichnung „SmartPointer“ bereit, die eine einfache Form der Verweiszählung verwendet, um die Zuweisung von dynamischem Arbeitsspeicher nachzuverfolgen und Aufgaben im Rahmen der Arbeitsspeicherverwaltung auszuführen.

Beispiel:

```
SmartPointer<Server> server =new Server();
server.connect();
...
server.disconnect();
```

Sie müssen den Arbeitsspeicher für den Pointer-Server nicht löschen. SmartPointer übernimmt die Arbeitsspeicherverwaltung für Sie.

Beispielanwendung

Der unten dargestellte Beispielcode zeigt anschaulich auf, wie die ASCII-Version der C++ API verwendet werden kann.

```
try{
    //Create Server
```

```

SmartPointer<Server> server =new Server();

//Set server connection properties
server->setConnectionProperty(Server::HOST, "localhost");
server->setConnectionProperty(Server::PORT, "10119");
server->setConnectionProperty(Server::CONNECTION_TYPE , "SOCKET");
server->setConnectionProperty(Server::ACCOUNT_ID, "guest");
server->setConnectionProperty(Server::ACCOUNT_PASSWORD, "");

//Connect to server
server->connect();

//Get Service From Server
SmartPointer<Service> service = server-
>getService("ValidateAddress");

//Create Input Message
SmartPointer<Message> request = new Message();

//Fill DataTable in the input message
SmartPointer<DataTable> dataTable = request->getDataTable();
SmartPointer<DataRow> row1 = dataTable->newRow();
row1->set("AddressLine1", "4200 Parliament Place");
row1->set("City", "Lanham");
row1->set("StateProvince", "Maryland");
dataTable->addRow(row1);

SmartPointer<DataRow> row2 = dataTable->newRow();
row2->set("AddressLine1", "100 Congress");
row2->set("City", "Austin");
row2->set("StateProvince", "Texas");
dataTable->addRow(row2);

//Set"option" Properties to the Input Message
request->putOption("OutputCasing", "M");
request->putOption("OutputRecordType", "A");

//Process Input Message, return output Message
SmartPointer<Message> reply = service->process(request);

//Disconnect from server
server->disconnect();

//Get the result from the reponse message
SmartPointer<DataTable> returnDataTable = reply->getDataTable();

vector<string> columnName = returnDataTable->getColumnNames();
vector< SmartPointer<DataRow> >::iterator iter =
returnDataTable->iterator();

for (int i=0; i< returnDataTable->getRowCount(); i++, iter++)
{
SmartPointer<DataRow> dataRow = *iter;

```

```

for (int col = 0; col < returnDataTable->getColumnCount(); col++)
{
    const char* value = DataRow->get(columnName[col].c_str());
    cout << value << "\n";
}
}
} catch (Exception e)
{
    cout << "Error Occurred, " << e.getErrorMessage();
}
}

```

Der unten dargestellte Beispielcode zeigt anschaulich auf, wie die Unicode-Version der C++ API verwendet werden kann.

```

try{
    //Create Server
    SmartPointer<Server> server =new Server();

    //Set server connection properties
    server->setConnectionProperty(Server::HOST,"localhost");
    server->setConnectionProperty(Server::PORT, "10119");
    server->setConnectionProperty(Server::CONNECTION_TYPE , "SOCKET");
    server->setConnectionProperty(Server::ACCOUNT_ID, "guest");
    server->setConnectionProperty(Server::ACCOUNT_PASSWORD, "");

    //Connect to server
    server->connect();

    //Get Service From Server
    //NOTE: ValidateAddress does not support unicode, but supports
    //characters in Canadian address and International address data files.

    SmartPointer<Service> service = server->getService("ValidateAddress");

    //Create Input Message
    SmartPointer<Message> request = new Message();

    //Fill DataTable in the input message
    SmartPointer<DataTable> dataTable = request->getDataTable();
    dataTable->addColumn("AddressLine1");
    dataTable->addColumn("City");
    dataTable->addColumn("PostalCode");
    dataTable->addColumn("Country");

    SmartPointer<DataRow> row1 = dataTable->newRow();

    UnicodeString address1 = "74, Rue Octave Bénard";
    row1->set( 0 , address1);
    UnicodeString city1 = "Etang-Salé-les-Bains";
}

```

```

row1->set( 1 , city1);
UnicodeString postalCode1 = "97427";
row1->set( 2 , postalCode1);
UnicodeString country1 = "Reunion Island";
row1->set( 3 , country1);

dataTable->addRow(row1);

SmartPointer<DataRow> row2 = dataTable->newRow();
UnicodeString address2 = "Final Av. Panteón Foro Libertador";
row2->set( 0 , address2);
UnicodeString city2 = "Caracas";
row2->set( 1 , city2);
UnicodeString postalCode2 = "1010";
row2->set( 2 , postalCode2);
UnicodeString country2 = "Venezuela";
row2->set( 3 , country2);

dataTable->addRow(row2);

//Set"option" Properties to the Input Message
request->putOption("OutputCasing", "M");
request->putOption("OutputRecordType", "A");

//Process Input Message, return output Message
SmartPointer<Message> reply = service->process(request);

//Disconnect from server
server->disconnect();

//Get the result from the response message
SmartPointer<DataTable> returnDataTable = reply->getDataTable();

vector<UnicodeString> columnName = returnDataTable->getColumnNames();

vector< SmartPointer<DataRow> >::iterator iter = returnDataTable->iterator();

for (int i=0; i< returnDataTable->getRowCount(); i++, iter++)
{
SmartPointer<DataRow> dataRow = *iter;

for (int col = 0; col < returnDataTable->getColumnCount(); col++)
{
UnicodeString value = dataRow->get(columnName[col]);
wcout <<value.getTerminatedBuffer() <<"\n"; }
}

}catch(Exception e)
{
UnicodeString error = e.getErrorMessage() ;

```



```
wcout << error.getTerminatedBuffer();  
}
```

Server

Mithilfe der `Server`-Klasse können Sie eine Verbindung zum Server aufbauen, die Verbindung zum Server trennen und den Dienst vom Server abrufen.

Konstruktoren

Die Konstruktoren für die `Server`-Klasse sind die folgenden:

- `Server()`

Destruktor

Der Destruktor für die `Server`-Klasse ist der folgende:

- `~Server()`

Connect

Liest die Eigenschaften, um die Konfigurationseinstellungen zu ermitteln, und baut eine Verbindung zum Server auf. Sie können über HTTP, HTTPS oder SOCKET eine Verbindung herstellen.

Anmerkung: C++ verwendet das HTTP, HTTPS oder SOCKET-Protokoll zum Verbindungsaufbau mit dem Server. HTTP und HTTPS bauen logisch eine Client-Verbindung auf, stellen jedoch erst eine tatsächliche Verbindung zum Server her, wenn eine `GetService`- oder `Process`-Methode aufgerufen wird. Das SOCKET-Protokoll baut eine Verbindung zum Server auf, wenn `Connect` aufgerufen wird.

Syntax

```
void connect();
```

Parameter

Keine.

Ergebnisse

Stellt eine Client-Verbindung zum Server her.

Beispiel

```
//Create Server
SmartPointer<Server> server =new Server();

//Set server connection properties
server->setConnectionProperty(Server::HOST,"localhost");
server->setConnectionProperty(Server::PORT, "10119");
server->setConnectionProperty(Server::CONNECTION_TYPE , "SOCKET");
server->setConnectionProperty(Server::ACCOUNT_ID, "guest");
server->setConnectionProperty(Server::ACCOUNT_PASSWORD, "");

//Connect to server
server->connect();
```

Disconnect

Trennt die Verbindung zum Server.

Syntax

```
void disconnect()
```

Parameter

Keine.

Ergebnisse

Die Client-Verbindung zum Server wird getrennt.

Beispiel

```
SmartPointer<Server> server =new Server()
server->connect();
...
server->disconnect();
```

SetConnectionProperty

Legt die Konfigurationseigenschaften der Serververbindung fest, z. B. den Servernamen und die Timeout-Dauer.

Syntax

ASCII-Version:

```
void setConnectionProperty(const char* name, const char* value)
```

Unicode-Version:

```
void setConnectionProperty(const UnicodeString name, const UnicodeString value)
```

Parameter

- Name: Dies ist der Name der Verbindungseigenschaft wie HOST.
- Value: Dies ist der Wert für den Namen der Verbindungseigenschaft wie „www.myhost.com“

Ergebnisse

Die Konfigurationseinstellungen für die Verbindung zum Server sind festgelegt.

Beispiel

ASCII-Version

```
SmartPointer<Server> server =new Server()
server->setConnectionProperty(Server::HOST, "localhost");
server->setConnectionProperty(Server::PORT, "8080");
```

Unicode-Version

Entspricht ASCII oder:

```
SmartPointer<Server> server =new Server()
UnicodeString host="localhost";// Or input unicode string
server->setConnectionProperty(Server::HOST, host);
```

GetService

Ruft den Dienst vom Server ab.

Anmerkung: Eine Liste der Dienste, die Ihnen zur Verfügung stehen, finden Sie im Abschnitt „Komponentenreferenz“ in diesem Handbuch.

Syntax

ASCII-Version:

```
SmartPointer<Service> getService(const char* serviceName)
```

Unicode-Version:

```
SmartPointer<Service> getService(const UnicodeString serviceName)
```

Parameter

- Name des Dienstes

Ergebnisse

Gibt den bestimmten Dienst zurück.

Beispiel

ASCII-Version

```
// Get Service From Server
SmartPointer<Service> service = server->getService("ValidateAddress");
```

Unicode-Version

Entspricht ASCII oder:

```
// Get Service From Server
UnicodeString serviceName="ValidateAddress";// Or input unicode string
SmartPointer<Service> service = server->getService(serviceName);
```

Service

Die `Service`-Klasse dient der Verarbeitung der Nachricht (d. h., dass sie die Nachricht an den Server sendet und eine Antwort vom Server erhält).

Process

Verarbeitet die Eingabenachricht und gibt die Antwortnachricht zurück.

Syntax

```
SmartPointer<Message> process(Message* message)
```

Parameter

- Eingabenachricht

Ergebnisse

Gibt die Antwortnachricht zurück.

Beispiel

```
SmartPointer<Message> reply = service->process(request);
```

Message

Die `Message`-Klasse sendet Ihre Eingabedaten und empfängt Ihre Ausgabedaten vom Dienst. Die Eigenschaften für `Message` umfassen Kontextentitäten wie Konto-ID, Kontokennwort, Dienstname und Dienstmethode, sowie Optionsentitäten, bei denen es sich um die dienstspezifischen Laufzeitoptionen handelt, und Fehlerentitäten, bei denen es sich um die Fehlerklasse, Fehlermeldung und Fehler-Stacktrace handelt.

Konstruktoren

Die Konstruktoren für die `Message`-Klasse sind die folgenden:

- `Message()`

Beispiel:

```
Message *request = new Message();
```

- `Message(const Message&)`

Beispiel:

```
Message* request = new Message();
Message anotherMessage = request;
Message message(anotherMessage);
```

Destruktor

Der Destruktor für die Message-Klasse ist der folgende:

- `~Message()`;

In der folgenden Tabelle werden die Funktionen zusammengefasst, die jede Methode in der Message-Klasse ausführt.

Tabelle 16: Message Methodenzusammenfassung

Methode	Funktion
getContext	Ruft den Wert der Kontextentität ab, der durch den Namen in der Kontextsitzung der Nachricht identifiziert wurde.
getContext	Ruft die Zuordnung ab, die alle Kontexteinträge enthält.
putContext	Legt den Wert der Kontextentität fest, der durch den Namen in der Kontextsitzung der Nachricht identifiziert wurde. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt.
putContext	Fügt den aktuellen Kontexteigenschaften die neuen Kontexteigenschaften hinzu.
setContext	Überschreibt die aktuellen Kontexteigenschaften mit den neuen Kontexteigenschaften.
getOption	Ruft den Wert der Optionsentität ab, der durch den Namen in der Optionssitzung der Nachricht identifiziert wurde.

Methode	Funktion
getOptions	Ruft die Zuordnung ab, die alle Optionseinträge enthält.
putOption	Legt den Wert der Optionsentität fest, der durch den Namen in der Optionssitzung der Nachricht identifiziert wurde. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt.
putOptions	Fügt die neuen Optionseigenschaften zu den aktuellen Optionseigenschaften hinzu.
setOptions	Überschreibt die aktuellen Optionseigenschaften mit den neuen Optionseigenschaften.
getError	Ruft die Fehlermeldung ab.
getDataTable	Ruft die Datentabelle aus der Nachricht ab.

GetContext

Ruft den Wert der Kontextentität ab, der durch den Namen in der Kontextsitzung der Nachricht identifiziert wurde.

Syntax

ASCII-Version

```
const char* getContext(const char* name)
```

Unicode-Version

```
const UnicodeString getContext(const UnicodeString name)
```

Parameter

- Der Name, dessen zugeordneter Wert zurückgegeben werden soll.

Ergebnisse

Gibt den Wert für den Namen in der Kontextentität zurück. Wenn der Name nicht vorhanden ist, gibt die Methode eine leere Zeichenfolge zurück.

Beispiel

ASCII-Version

```
const char* value= msg->getContext(Server::ACCOUNT_ID);
```

Unicode-Version

Entspricht ASCII oder:

```
UnicodeString name= Server::ACCOUNT_ID;// Or input unicode string
UnicodeString value= msg->getContext(name);
```

GetContext

Ruft die Zuordnung ab, die alle Kontexteinträge enthält.

Syntax

ASCII-Version

```
map<string , string> getContext()
```

Unicode-Version

```
map< UnicodeString, UnicodeString > getContext()
```

Parameter

Keine.

Ergebnisse

Gibt die Zuordnung zurück, die alle Kontexteinträge enthält.

Beispiel

ASCII-Version

```
map<string , string> context = message->getContext();
```

Unicode-Version

```
map< UnicodeString, UnicodeString > context = message->getContext();
```


PutContext

Legt den Wert für den angegebenen Namen in den Kontexteigenschaften fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt.

Kontexteigenschaften enthalten die folgenden Konstanten: Konto-ID, Kontokennwort, Dienstname, Dienstschlüssel und Anforderungs-ID.

Syntax

ASCII-Version

```
void putContext(const char* name, const char* value)
```

Unicode-Version

```
void putContext(const UnicodeString name, const UnicodeString value)
```

Parameter

- Name, unter dem der angegebene Wert zugeordnet werden soll.
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Beispiel

ASCII-Version

```
message->putContext(Message.CONTEXT_ACCOUNT_ID, "user1");
```

Unicode-Version

Entspricht ASCII oder:

```
UnicodeString account="user1" ;// Or input unicode string
message->putContext(Message.CONTEXT_ACCOUNT_ID, account);
```

PutContext

Fügt den aktuellen Kontexteigenschaften die neuen Kontexteigenschaften hinzu.

Syntax

ASCII-Version

```
void putContext(map<string , string> context)
```

Unicode-Version

```
void putContext(map< UnicodeString, UnicodeString > context)
```

Parameter

- Die neue Kontextzuordnung, die der aktuellen Kontextzuordnung hinzugefügt werden soll.

*Beispiel***ASCII-Version**

```
map<string , string> context ;
//more code
message->putContext(context);
```

Unicode-Version

```
map< UnicodeString, UnicodeString > context ;
//more code
message->putContext(context);
```

SetContext

Überschreibt die aktuellen Kontexteigenschaften mit den neuen Kontexteigenschaften.

*Syntax***ASCII-Version**

```
void setContext(map<string , string> context)
```

Unicode-Version

```
void setContext(map< UnicodeString, UnicodeString > context)
```

Parameter

- Die neue Kontextzuordnung, durch die die aktuelle Kontextzuordnung ersetzt werden soll.

*Beispiel***ASCII-Version**

```
map<string , string> context ;
//more code
message->setContext(context);
```

Unicode-Version

```
map< UnicodeString, UnicodeString > context ;
//more code
message->setContext(context);
```

GetOption

Ruft den Wert der Optionsentität ab, der durch den Namen im Optionsabschnitt der Nachricht identifiziert wird. Die Optionsentitäten umfassen dienstspezifische Laufzeitoptionen, z. B. die Groß-/Kleinschreibung der Ausgabe, das Format der Ausgabedaten usw.

*Syntax***ASCII-Version**

```
const char* getOption(const char* name)
```

Unicode-Version

```
const UnicodeString getOption(const UnicodeString name)
```

Parameter

- Der Name, dessen zugeordneter Wert zurückgegeben werden soll.

Ergebnisse

Gibt den Wert für den Namen in der Kontextentität zurück. Wenn der Name nicht vorhanden ist, gibt die Methode eine leere Zeichenfolge zurück.

*Beispiel***ASCII-Version**

```
const char* value = message->getOption("OutputCasing");
```

Unicode-Version

Entspricht ASCII oder:

```
UnicodeString option="OutputCasing"; // Or input unicode string
UnicodeString value= message->getOption(option);
```

GetOptions

Ruft die Zuordnung ab, die alle Optionseinträge enthält.

Syntax

ASCII-Version

```
map<string , string> getOptions()
```

Unicode-Version

```
map< UnicodeString, UnicodeString > getOptions()
```

Parameter

Keine.

Ergebnisse

Gibt die Zuordnung zurück, die alle Optionseinträge enthält.

Beispiel

ASCII-Version

```
const char* value = message->getOption("OutputCasing");
```

Unicode-Version

```
UnicodeString option="OutputCasing"; //or input Unicode string
UnicodeString value= message->getOption(option);
```

PutOption

Legt den Wert für den angegebenen Namen in den Optionseigenschaften fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt. Bei Optionseigenschaften handelt es sich um die dienstspezifischen Laufzeioptionen.

Syntax

ASCII-Version

```
void putOption(const char* name, const char* value)
```

Unicode-Version

```
void putOption(const UnicodeString name, const UnicodeString value)
```

Parameter

- Name, unter dem der angegebene Wert zugeordnet werden soll.
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

*Beispiel***ASCII-Version**

```
message->putOption("OutputCasing", "M");
```

Unicode-Version

Entspricht ASCII oder:

```
UnicodeString option="M"; // Or input unicode string
message->putOption("OutputCasing", option);
```

PutOptions

Fügt die neuen Optionseigenschaften zu den aktuellen Optionseigenschaften hinzu.

*Syntax***ASCII-Version**

```
void putOptions(map<string , string> options)
```

Unicode-Version

```
void putOptions(map< UnicodeString, UnicodeString > options)
```

Parameter

- Die neue Optionszuordnung, die den aktuellen Optionseigenschaften hinzugefügt werden soll.

*Beispiel***ASCII-Version**

```
map<string , string> options ;
//more code
message->putOptions(options);
```

Unicode-Version

```
map< UnicodeString, UnicodeString > options ;
//more code
message->putOptions(options);
```

SetOptions

Überschreibt die aktuellen Optionseigenschaften mit den neuen Optionseigenschaften.

*Syntax***ASCII-Version**

```
void setOptions(map<string , string> options)
```

Unicode-Version

```
void setOptions(map< UnicodeString, UnicodeString > options)
```

Parameter

- Die neue Optionszuordnung, durch die die aktuelle Optionszuordnung ersetzt werden soll.

*Beispiel***ASCII-Version**

```
map<string , string> options ;
//more code
message->setOptions(options);
```

Unicode-Version

```
map< UnicodeString, UnicodeString > options ;
//more code
message->setOptions(options);
```

GetError

Ruft die Fehlermeldung aus der Nachricht ab.

Syntax

ASCII-Version

```
string getError()
```

Unicode-Version

```
UnicodeString getError()
```

Parameter

Keine.

Ergebnisse

Gibt die Fehlermeldung in der Nachricht zurück.

Beispiel

ASCII-Version

```
String error = message->getError();
```

Unicode-Version

```
UnicodeString error = message->getError();
```

GetDataTable

Ruft die Datentabelle (DataTable) in der Nachricht ab.

Syntax

```
SmartPointer<DataTable> getDataTable()
```

Parameter

Keine.

Beispiel

```
SmartPointer<DataTable> dataTable  
= message->getDataTable();
```

DataTable

DataTable enthält die Datensätze für die Eingabe- und Ausgabedaten.

Konstruktoren

Die Konstruktoren für die DataTable-Klasse sind die folgenden:

- `DataTable()`

Beispiel:

```
DataTable* dataTable = new DataTable();
```

Destruktor

Der Destruktor für die DataTable-Klasse ist der folgende:

- `~DataTable();`

In der folgenden Tabelle werden die Funktionen zusammengefasst, die jede Methode in der DataTable-Klasse ausführt.

Tabelle 17: Zusammenfassung der DataTable-Methoden

Methode	Funktion
<code>addColumn</code>	Fügt eine neue Spalte hinzu.
<code>getColumnNames</code>	Ruft alle Spaltennamen ab.
<code>getColumnIndex</code>	Ruft den entsprechenden Spaltenindex ab.
<code>getColumnCount</code>	Ruft die Anzahl der Spalten ab.

Methode	Funktion
clear	Löscht die Daten in der Datentabelle (DataTable).
iterator	Ein Iterator, der alle Datenzeilen (DataRows) in der Datentabelle (DataTable) enthält.
addRow	Fügt der Datentabelle (DataTable) eine Datenzeile (DataRow) hinzu.
newRow	Erstellt eine neue Datenzeile (DataRow) in der Datentabelle (DataTable).
getRowCount	Ruft die Anzahl der Datenzeilen (DataRows) in dieser Datentabelle (DataTable) ab.
merge	Führt die vorhandene Datentabelle (DataTable) und die aktuelle Datentabelle zusammen.

AddColumn

Fügt eine neue Spalte hinzu.

Syntax

ASCII-Version

```
int addColumn(const char* columnName)
```

Unicode-Version

```
int addColumn(const UnicodeString columnName)
```

Parameter

- Spaltenname

Ergebnisse

- Gibt den Index der Spalte zurück.

Ausnahmen

- leerer Spaltenname
- doppelter Spaltenname

Beispiel

ASCII-Version

```
SmartPointer<DataTable> dataTable = message.getDataTable();
dataTable->addColumn("Address");
dataTable->addColumn("City");
```

Unicode-Version

Entspricht ASCII oder:

```
SmartPointer<DataTable> dataTable = message.getDataTable();
UnicodeString columnName="Address"; // Or input unicode string
dataTable->addColumn(columnName);
```

GetColumnNames

Ruft alle Spaltennamen ab.

Syntax

ASCII-Version

```
vector<string> getColumnNames();
```

Unicode-Version

```
vector<UnicodeString> getColumnNames();
```

Parameter

Keine.

Ergebnisse

Gibt den Vektor der Spaltennamen zurück.

*Beispiel***ASCII-Version**

```
vector<string> columnNames = dataTable->getColumnNames();
```

Unicode-Version

```
vector<UnicodeString> columnNames = dataTable->getColumnNames();
```

GetColumnIndex

Ruft den entsprechenden Spaltenindex ab.

*Syntax***ASCII-Version**

```
int getColumnIndex(const char* columnName)
```

Unicode-Version

```
int getColumnIndex(const UnicodeString columnName)
```

Parameter

- Spaltenname

Ergebnisse

Gibt den entsprechenden Spaltenindex zurück.

*Beispiel***ASCII-Version**

```
int columnIndex = dataTable->getColumnIndex ("City");
```

Unicode-Version

Entspricht ASCII oder:

```
UnicodeString columnName="City"; // Or input unicode string
int columnIndex = dataTable->getColumnIndex (columnName);
```

GetColumnCount

Ruft die Anzahl der Spalten ab.

Syntax

```
int getColumnCount()
```

Parameter

Keine.

Ergebnisse

Gibt die Anzahl der Spalten zurück.

Beispiel

```
int columnCount = dataTable->getColumnCount ();
```

Clear

Löscht die Daten in der Datentabelle (DataTable).

Syntax

```
void clear()
```

Parameter

Keine.

Beispiel

```
dataTable->clear ();
```

Iterator

Ein Iterator, der alle Datenzeilen (DataRows) in der Datentabelle (DataTable) enthält.

Syntax

```
vector< SmartPointer<DataRow> >::iterator iterator()
```

Parameter

Keine.

Ergebnisse

Gibt einen Iterator zurück, der alle Datenzeilen (DataRows) in der Datentabelle (DataTable) enthält.

Beispiel

```
vector<string> columnName
= returnDataTable->getColumnNames();

vector< SmartPointer<DataRow> >::iterator theIterator
= returnDataTable->iterator();

for (int i=0; i< returnDataTable->getRowCount();
i++, theIterator++)
{
SmartPointer<DataRow> dataRow = *theIterator;

for (int col = 0;
col < returnDataTable->getColumnCount(); col++)
{
const char* value = dataRow->get(columnName[col].c_str());
}
}
}
```

AddRow

Fügt der Datentabelle (DataTable) eine Datenzeile (DataRow) hinzu.

Syntax

```
void addRow( SmartPointer<DataRow> dataRow)
```

Parameter

- DataRow: Dies ist die zur Datentabelle (DataTable) hinzuzufügende Datenzeile (DataRow).

Beispiel

```
SmartPointer<DataRow> newRow = dataTable->newRow();
newRow->set( 0 , "10535 Boyer");
newRow->set( 1 , "Austin");
newRow->set( 2 , "Texas");
dataTable->addRow(newRow);
```

NewRow

Erstellt eine neue Datenzeile (DataRow) in der Datentabelle (DataTable).

Syntax

```
SmartPointer<DataRow> newRow()
```

Ergebnisse

Gibt die neu erstellte Datenzeile (DataRow) zurück.

Beispiel

```
SmartPointer<DataRow> newRow = dataTable->newRow();
newRow->set( 0 , "10535 Boyer");
newRow->set( 1 , "Austin");
newRow->set( 2 , "Texas");
dataTable->addRow(newRow);
```

GetRowCount

Ruft die Anzahl der Datenzeilen (DataRows) in dieser Datentabelle (DataTable) ab.

Syntax

```
int getRowCount()
```

Ergebnisse

Gibt die Anzahl der Datenzeilen (DataRows) in dieser Datentabelle (DataTable) zurück.

Beispiel

```
int rowCount = dataTable->getRowCount();
```

Merge

Führt die vorhandene Datentabelle (DataTable) und die aktuelle Datentabelle zusammen.

Syntax

```
void merge(DataTable* other)
```

Parameter

- **Other:** Dies ist die andere Datentabelle (DataTable), die mit der aktuellen Datentabelle (DataTable) zusammengeführt werden soll.

Beispiel

```
DataTable* otherDataTable = new DataTable();  
dataTable->merge(otherDataTable);
```

DataRow

DataRow enthält den Datensatz für die Eingabe- und Ausgabedaten.

Konstruktor

Die Konstruktoren für die DataRow-Klasse sind die folgenden:

- DataRow ()

Beispiel:

```
DataRow * dataRow = new DataRow();
```

- DataRow(const DataRow&)

Beispiel:

```
DataRow* dataRow = new DataRow();  
DataRow anotheDataRow = dataRow;  
DataRow newDataRow(anotheDataRow);
```

Destruktor

Der Destruktor für die DataRow-Klasse ist der folgende:

- `~ DataRow ();`

In der folgenden Tabelle werden die Funktionen zusammengefasst, die jede Methode in der DataRow-Klasse ausführt.

Tabelle 18: Zusammenfassung der DataRow-Methoden

Methode	Funktion
getColumnNames	Ruft alle Spaltennamen ab.
getColumnIndex	Ruft den entsprechenden Spaltenindex ab.
getColumnCount	Ruft die Anzahl der Spalten ab.
get	Ruft den Wert aus dem Feld-Array nach dem Spaltenindex in dieser Datenzeile (DataRow) ab.
get	Ruft den Wert aus dem Feld-Array nach dem Spaltennamen in dieser Datenzeile (DataRow) ab.
merge	Führt die vorhandene Datentabelle (DataTable) und die aktuelle Datentabelle zusammen.
set	Legt den Wert für den entsprechenden Spaltennamen für die Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.
set	Legt den Wert für den entsprechenden Spaltenindex für die Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.

Methode	Funktion
addChild	Fügt der benannten übergeordneten/untergeordneten Beziehung eine Datenzeile (DataRow) hinzu. Wenn die benannte Beziehung vorhanden ist, wird die bereitgestellte Datenzeile (DataRow) der vorhandenen DataRow-Auflistung angefügt. Anderenfalls wird eine neue Sammlung mit der angegebenen Datenzeile (DataRow) als einziges Element erstellt.
getChildren	Ruft die untergeordneten Zeilen einer benannten Beziehung ab.
listChildNames	Ruft alle Namen der benannten übergeordneten/untergeordneten Beziehungen.
setChildren	Legt die Zeilen einer angegebenen benannten übergeordneten/untergeordneten Beziehung fest. Wenn zuvor Zeilen mit diesem Namen vorhanden waren, werden diese an den Aufrufer zurückgegeben.

GetColumnNames

Ruft alle Spaltennamen ab.

Syntax

ASCII-Version

```
vector<string> getColumnNames()
```

Unicode-Version

```
vector<UnicodeString> getColumnNames()
```

Parameter

Keine.

Ergebnisse

Gibt den Vektor der Spaltennamen zurück.

*Beispiel***ASCII-Version**

```
vector<string> columnNames = dataRow->getColumnNames();
```

Unicode-Version

```
vector<UnicodeString> columnNames = dataRow->getColumnNames();
```

GetColumnIndex

Ruft den entsprechenden Spaltenindex ab.

*Syntax***ASCII-Version**

```
int getColumnIndex(const char* columnName)
```

Unicode-Version

```
int getColumnIndex(const UnicodeString columnName)
```

Parameter

- Spaltenname

Ergebnisse

Gibt den entsprechenden Spaltenindex zurück.

*Beispiel***ASCII-Version**

```
int columnIndex = dataRow->getColumnIndex ("City");
```

Unicode-Version

Entspricht ASCII oder:

```
UnicodeString columnName="City"; // Or input unicode string
int columnIndex = dataRow->getColumnIndex (columnName);
```

GetColumnCount

Ruft die Anzahl der Spalten ab.

Syntax

```
int getColumnCount()
```

Parameter

Keine.

Ergebnisse

Gibt die Anzahl der Spalten zurück.

Beispiel

```
int columnCount = dataRow->getColumnCount ();
```

Get

Ruft den Wert aus dem Feld-Array nach dem Spaltenindex in dieser Datenzeile (DataRow) ab.

Syntax

ASCII-Version

```
const char* get(int index)
```

Unicode-Version

```
const UnicodeString get(int index)
```

Parameter

- Der Index, dem der angegebene Wert zugeordnet werden soll.

Ergebnisse

Gibt den Wert für den Spaltenindex in der Datenzeile (DataRow) zurück, und gibt eine leere Zeichenfolge zurück, wenn der Index ungültig ist.

Beispiel

ASCII-Version

```
const char* value = dataRow->get(1);
```

Unicode-Version

```
const UnicodeString value = dataRow->get(1);
```

Get

Ruft den Wert aus dem Feld-Array nach dem Spaltennamen in dieser DataRow ab.

Syntax

ASCII-Version

```
const char* get(const char* columnName)
```

Unicode-Version

```
const UnicodeString get(const UnicodeString columnName)
```

Parameter

- Name, unter dem der angegebene Wert zugeordnet werden soll.

Ergebnisse

Gibt den Wert für den Spaltennamen in der Datenzeile (DataRow) zurück, und gibt eine leere Zeichenfolge zurück, wenn der Spaltenname nicht vorhanden ist.

Beispiel

ASCII-Version

```
const char* value = dataRow->get("City");
```

Unicode-Version

Entspricht ASCII oder:

```
UnicodeString columnName="City"; // Or input unicode string
const UnicodeString value = dataRow->get(columnName);
```

Merge

Führt die angegebene Datenzeile (DataRow) mit der aktuellen Datenzeile zusammen.

Syntax

```
void merge(DataRow* other)
```

Parameter

- Die andere Datenzeile (DataRow), die mit der aktuellen Datenzeile (DataRow) zusammengeführt werden soll.

Beispiel

```
DataRow* otherDataRow = new DataRow();  
DataRow->merge(otherDataRow);
```

Set

Legt den Wert für die entsprechende Spalte der Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.

Syntax

ASCII-Version

```
void set(const char* columnName, const char* value)
```

Unicode-Version

```
void set(const UnicodeString columnName, const UnicodeString value)
```

Parameter

- Name, unter dem der angegebene Wert zugeordnet werden soll.
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ausnahmen

- leerer Spaltenname
- doppelter Spaltenname

Beispiel

ASCII-Version

```
SmartPointer<DataRow> newRow = dataTable->newRow();
newRow->set( "AddressLine1" , "10535 Boyer");
newRow->set( "City" , "Austin");
newRow->set( "State" , "Texas");
```

Unicode-Version

Entspricht ASCII oder:

```
SmartPointer<DataRow> newRow = dataTable->newRow();
UnicodeString address="10535 Boyer"; // Or input unicode string
newRow->set( "AddressLine1" , address);
```

Set

Legt den Wert für die entsprechende Spalte der Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.

Syntax

ASCII-Version

```
void set(int index, const char* value)
```

Unicode-Version

```
void set(int index, const UnicodeString value)
```

Parameter

- Spaltenindex, dem der angegebene Wert zugeordnet werden soll
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ausnahmen

- Der Spaltenindex ist ungültig.

Beispiel

ASCII-Version

```
SmartPointer<DataRow> newRow = dataTable->newRow();
newRow->set( 0 , "10535 Boyer");
newRow->set( 1 , "Austin");
newRow->set( 2 , "Texas");
```

Unicode-Version

Entspricht ASCII oder:

```
SmartPointer<DataRow> newRow = dataTable->newRow();
UnicodeString address="10535 Boyer"; // Or input unicode string
newRow->set( 0 , address);
```

AddChild

Fügt der benannten übergeordneten/untergeordneten Beziehung eine Datenzeile (DataRow) hinzu. Wenn die benannte Beziehung vorhanden ist, wird die angegebene Datenzeile (DataRow) der vorhandenen Datenzeilensammlung angefügt. Anderenfalls wird eine neue Sammlung mit der angegebenen Datenzeile (DataRow) als einziges Element erstellt.

Syntax

ASCII-Version

```
void addChild(const char* childName, SmartPointer<DataRow> childDataRow)
```

Unicode-Version

```
void addChild(const UnicodeString childName, SmartPointer<DataRow>
childDataRow)
```

Parameter

- Der Name der übergeordneten/untergeordneten Beziehung (z. B. „Daten zum Überschwemmungsgebiet“, „Verweise“, „Verwendet von“)
- Die Datenzeile (DataRow), die der Beziehung hinzugefügt werden soll.

Beispiel

```
SmartPointer<DataRow> childDataRow =new DataRow();
childDataRow ->set("Address", "100 Congress");
childDataRow ->set("City", "Austin");
SmartPointer<DataRow> dataRow =new DataRow();
dataRow->addChild("child1", childDataRow );
```

GetChildren

Ruft die untergeordneten Zeilen einer benannten Beziehung ab.

Syntax

ASCII-Version

```
list< SmartPointer<DataRow> > getChildren(const char* childName)
```

Unicode-Version

```
list< SmartPointer<DataRow> > getChildren(const UnicodeString childName)
```

Parameter

- Der Name der übergeordneten/untergeordneten Beziehung (z. B. „Überschwemmungsgebietsdaten“, „Referenzen“, „Verwendet von“ usw.)

Ergebnisse

Gibt die untergeordneten Zeilen aus der benannten Beziehung zurück.

Beispiel

```
list< SmartPointer<DataRow> > rowsChild2= dataRow->getChildren("child2");
```

ListChildNames

Ruft alle Namen der benannten übergeordneten/untergeordneten Beziehungen.

Syntax

ASCII-Version

```
list<string> listChildNames()
```

Unicode-Version

```
list<UnicodeString> listChildNames()
```

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

```
list<G1CLIENT_STRING> names = dataRow->listChildNames();
```


SetChildren

Legt die Zeilen einer angegebenen benannten übergeordneten/untergeordneten Beziehung fest. Wenn zuvor Zeilen mit diesem Namen vorhanden waren, werden diese an den Aufrufer zurückgegeben.

Syntax

ASCII-Version

```
list< SmartPointer<DataRow> > setChildren(const char* childName, list< SmartPointer<DataRow> > dataRows)
```

Unicode-Version

```
list< SmartPointer<DataRow> > setChildren(const UnicodeString childName, list< SmartPointer<DataRow> > dataRows)
```

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

```
SmartPointer<DataRow> dataRow1=new DataRow();
dataRow1->set("Address", "100 Congress");
dataRow1->set("City", "Austin");
SmartPointer<DataRow> dataRow2=new DataRow();
dataRow2->set("Address", "200 Congress");
dataRow2->set("City", "Austin");
list< SmartPointer<DataRow> > rows ;
rows.push_back(dataRow1);
rows.push_back(dataRow2);
list< SmartPointer<DataRow> > rowsNewChildren = dataRowSpt-
>setChildren("child1", rows);
```

4 - Die COM-API

In this section

Einführung	123
Server	127
Service	130
Message	131
DataTable	138
DataRow	144
Map	152

Einführung

Der Begriff Component Object Model (COM) bezieht sich auf eine offene Architektur für plattformübergreifende Entwicklung von Client/Server-Anwendungen, die auf objektorientierter Technologie basiert. COM ist eine Möglichkeit, wiederverwendbare Softwarekomponenten zu erstellen. Clients haben über im Objekt implementierte Schnittstellen Zugriff auf das Objekt. Anders ausgedrückt, sind Objekte Kommunikationsmedien zwischen Client und Server. Das Component Object Model bietet eine flexible Möglichkeit, verteilte objektorientierte Systeme zu erstellen. COM-Objekte sind sprachunabhängig, können in binärer Form versendet werden, können ohne Änderungen am vorhandenen Integrierungscode aktualisiert werden und können in einem Netzwerk transparent verschoben werden. Aufgrund dieser Vorteile sind COM-Objekte extrem flexibel und können angepasst werden, um fast alle Windows-basierten Client-Server-Systeme um bestimmte Features zu erweitern.

Anmerkung: Die Beispiele in diesem Kapitel sind in Visual Basic geschrieben.

Die COM-API von Spectrum™ Technology Platform enthält folgende Schnittstellen:

- Server
- Dienst
- Nachricht
- DataTable
- DataRow
- Karte

Konstanten

Die COM-API verwendet zwei Sätze mit Konstanten. Der erste Satz gilt für das Objekt `Server` und wird in der folgenden Tabelle beschrieben.

Tabelle 19: Konstanten für die `Server`-Komponente

Name der Konstante	Beschreibung/Standard	Beispiel
SERVER.HOST	Zeichenfolge für den Hostnamen des Servers. Der Standardname lautet „localhost“.	65.89.200.89

Name der Konstante	Beschreibung/Standard	Beispiel
SERVER.PORT	Zeichenfolge für den Serverport. Der Standardwert ist „8080“.	10119
SERVER.ACCOUNT_ID	Zeichenfolge für die Serverkonto-ID. Kein Standardwert.	user1
SERVER.ACCOUNT_PASSWORD	Zeichenfolge für das Kennwort des Serverkontos. Kein Standardwert.	user1
SERVER.CONNECTION_TIMEOUT	Zeichenfolge für Timeout der Serververbindung in Millisekunden. Der Standardwert ist „5000“.	50000
SERVER.CONNECTION_TYPE	Zeichenfolge für den Serververbindungstyp. Derzeit werden nur HTTP, HTTPS und SOCKET unterstützt. Der Standardwert lautet „HTTP“.	HTTP(S)
SERVER.PROXY_HOST	Zeichenfolge für den Hostnamen des Proxy-Servers. Kein Standardwert.	192.168.1.77
SERVER.PROXY_PORT	Zeichenfolge für den Proxy-Serverport. Kein Standardwert.	8080
SERVER.PROXY_USER	Zeichenfolge für die Konto-ID des Proxy-Servers. Kein Standardwert.	user1
SERVER.PROXY_PASSWORD	Zeichenfolge für das Kontokennwort des Proxy-Servers. Kein Standardwert.	user1

Der zweite Konstantensatz ist für die `Message`-Komponente vorgesehen.

Tabelle 20: Konstanten für die Message-Komponente

Name der Konstante	Beschreibung	Beispiel
MESSAGE.CONTEXT_ACCOUNT_ID	Zeichenfolge für den Nachrichtentext der Konto-ID.	user1
MESSAGE.CONTEXT_ACCOUNT_PASSWORD	Zeichenfolge für das Kontokennwort des Nachrichtenkontextes.	user1
MESSAGE.CONTEXT_SERVICE_NAME	Zeichenfolge für den Dienstnamen des Nachrichtenkontextes.	echoservice

Fehlermeldungen

Die COM API verwendet die folgenden Fehlermeldungen:

- Fehlermeldungen zur Verbindung:
 - "Connection type not supported"
 - "Client timeout"
- Fehlermeldungen zum Erstellen von Datentabellen (DataTables):
 - "Blank column name"
 - "Duplicated column name"
 - "The column index is invalid"
- Fehlermeldungen für Ausnahmen beim Verpacken der Nachrichten:
 - "Input Message is null"
 - "Failed to connect to Server"
 - "Failed to disconnect to Server"
 - "Failed to open Http Connection"
 - "Failed to get Service"
 - "Failed to package the message using Serializer and Encoding"

Beispiel:

```
On Error GoTo ErrorHandler
Dim server As New G1CLIENTLib.server
```

```

server.setConnectionProperty server.HOST, "localhost"
server.setConnectionProperty server.Port, "8080"
'Making connection to the server
server.Connect
...
Exit Sub
ErrorHandler:
MsgBox Err.Description

```

Beispielanwendung

Der folgende Beispielcode zeigt die Verwendung der COM-API.

```

On Error GoTo ErrorHandler

Dim server As New G1CLIENTLib.server
Dim service As G1CLIENTLib.service
Dim requestMsg As New G1CLIENTLib.Message
Dim replyMsg As G1CLIENTLib.Message
Dim dataTable As G1CLIENTLib.dataTable
Dim newRow As G1CLIENTLib.dataRow
Dim returnDataTable As G1CLIENTLib.dataTable
Dim row As G1CLIENTLib.DataRow
Dim sColumnNames() As String
Dim sColumnName As String
Dim sFieldValue As String
Dim rows() As Variant
Dim nRow As Integer
Dim nColumn As Integer
'Set server connection properties
server.setConnectionProperty server.HOST, "localhost"
server.setConnectionProperty server.Port, "10119"
server.setConnectionProperty server.CONNECTION_TYPE, "SOCKET"
server.setConnectionProperty server.ACCOUNT_ID, "guest"
server.setConnectionProperty server.ACCOUNT_PASSWORD, ""

'Connect to server
server.Connect

'Get the service from the server
Set service = server.getService("ValidateAddress")

'Fill DataTable in the input message
Set dataTable = requestMsg.getDataTable
dataTable.addColumn ("AddressLine1")
dataTable.addColumn ("City")
dataTable.addColumn ("StateProvince")

Set newRow = dataTable.newRow

```

```

newRow.setByIndex 0, "10535 Boyer"
newRow.setByIndex 1, "Austin"
newRow.setByIndex 2, "Texas"
dataTable.addRow newRow

'Set"option" Properties to the Input Message
requestMsg.putOption "OutputCasing", "M"
requestMsg.putOption "OutputRecordType", "A"

'Process Input Message, return output Message
Set replyMsg = service.process(requestMsg)

'Disconnect from the server
server.disconnect

'Get the result from the response message
Set returnDataTable = replyMsg.getDataTable
ReDim rows(returnDataTable.getRowCount) As Variant

rows = returnDataTable.iterator

ReDim sColumnNames(returnDataTable.getColumnCount) As String
sColumnNames = returnDataTable.getColumnNames

For nRow = 0 To returnDataTable.getRowCount - 1
Set row = rows(nRow)

For nColumn = 0 To row.getColumnCount - 1
    sColumnName = sColumnNames(nColumn)
    sFieldValue = row.getByName(sColumnName)
Next
Next

Exit Sub

ErrorHandler:
MsgBox Err.Description

```

Server

Mithilfe des Objekts `Server` können Sie eine Verbindung zum Server aufbauen, die Verbindung zum Server trennen und den Dienst vom Server abrufen.

Connect

Stellt eine Verbindung zum Server her. Die Verbindung kann über HTTP oder einen SOCKET aufgebaut werden.

Anmerkung: COM verwendet das HTTP, HTTPS oder SOCKET-Protokoll zum Verbindungsaufbau mit dem Server. HTTP und HTTPS bauen logisch eine Client-Verbindung auf, stellen jedoch erst eine tatsächliche Verbindung zum Server her, wenn eine GetService- oder Process-Methode aufgerufen wird. Das SOCKET-Protokoll baut eine Verbindung zum Server auf, wenn CONNECT aufgerufen wird.

Syntax

```
Sub connect ()
```

Parameter

Keine.

Ergebnisse

Keine.

Ausnahme

Verbindungstyp wird nicht unterstützt.

Beispiel

```
Dim server As New G1CLIENTLib.server  
server.connect
```

Disconnect

Trennt die Verbindung zum Server.

Syntax

```
Sub disconnect ()
```

Parameter

Keine.

Ergebnisse

Keine.

Beispiele

```
Dim server As New G1CLIENTLib.server
server.disconnect
```

GetService

Ruft den Dienst (z. B. ValidateAddress) vom Server ab.

Syntax

```
Function getService(serviceName As String) As Service
```

Parameter

- `serviceName`: Der Name des Dienstes, der für den Client erforderlich ist.

Ergebnisse

Der angeforderte Dienst oder NULL, wenn der Dienst nicht vorhanden ist.

Ausnahmen

- `ERROR_FAIL_TO_GET_SERVICE`: bei fehlender Verbindung zum Server.

Beispiel

```
Dim server As New G1CLIENTLib.server
Dim service As G1CLIENTLib.service
...
'get the service from the server
Set service = server.getService("ValidateAddress")
```

SetConnectionProperty

Legt die Konfigurationseigenschaften der Serververbindung fest, z. B. den Servernamen und die Timeout-Dauer.

Syntax

```
Sub setConnectionProperty(name As String, value As String)
```

Parameter

- Name: Dies ist der Name der Verbindungseigenschaft wie HOST.
- Value: Dies ist der Wert für den Namen der Verbindungseigenschaft wie „www.myhost.com“

Ergebnisse

Rückgabecodes: keine.

Ausnahmen

- ERROR_INVALID_COLUMN_NAME: Ein leerer Spaltenname oder ein Spaltenname mit NULL-Wert.
- ERROR_INVALID_VALUE: Ein NULL-Wert.

Beispiel

```
set connection properties
Dim server As New G1CLIENTLib.server

server.setConnectionProperty server.HOST, "localhost"
server.setConnectionProperty server.PORT, "8080"
```

Service

`Service` ruft den Dienst auf und verarbeitet die von Ihnen gesendete Nachricht (d. h., die Eingabenachricht wird versandt und die Antwort empfangen).

Process

Verarbeitet die Eingabenachricht und ruft die Antwortnachricht des Servers ab.

Syntax

```
Function process(IRequest As Message) As Message
```

Parameter

- `iRequest`: Das Objekt der Eingabennachricht, das die Einstellung „Option“ und die Datentabelle (`DataTable`) enthält

Ergebnisse

Gibt die Antwortnachricht für die Anforderung zurück.

Ausnahmen:

- `ERROR_NULL_INPUT_MESSAGE`: Anforderungsnachricht ist NULL.

Beispiel

```
Dim service As New G1CLIENTLib.service
Dim replyMsg As G1CLIENTLib.Message
...
'Process the message and return back the response message
Set replyMsg = service.process(requestMsg)
```

Message

Mithilfe von Nachrichten werden Ihre Eingabedaten gesendet und Ihre Ausgabedaten vom Dienst empfangen. Die Eigenschaften für `Message` umfassen Kontextentitäten wie Konto-ID, Kontokennwort, Dienstname und Dienstmethode, sowie Optionsentitäten, bei denen es sich um die dienstspezifischen Laufzeioptionen handelt, und Fehlerentitäten, bei denen es sich um die Fehlerklasse, Fehlermeldung und Fehler-Stacktrace handelt.

GetContext

Ruft den Wert der Kontextentität ab, der durch den Namen im Kontextabschnitt der Nachricht identifiziert wurde. Kontextentitäten enthalten die folgenden Konstanten: Konto-ID, Kontokennwort, Dienstname und Dienstmethode.

Syntax

```
Function getContext(name As String) As String
```

Parameter

- `Name`: Der Name, dessen zugeordneter Wert zurückgegeben werden soll.

Ergebnisse

Zeichenfolge: Der Wert der benannten Entität oder der leeren Zeichenfolge, wenn die benannte Entität nicht vorhanden ist.

Beispiel

```
Dim msg As New G1CLIENTLib.Message
Dim accountID As String

accountID = msg.getContext(msg.CONTEXT_ACCOUNT_ID)
```

GetContextMap

Ruft die Zuordnung ab, die alle Kontexteinträge enthält.

Syntax

```
Function getContextMap() As Map
```

Parameter

- Keine

Ergebnisse

Gibt die Zuordnung zurück, die alle Kontexteinträge enthält.

Beispiel

```
Dim map As G1CLIENTLib.Map
Dim requestMsg As New G1CLIENTLib.Message
Dim sKey As String
Dim sValue As String

requestMsg.putContext
  requestMsg.CONTEXT_ACCOUNT_ID, "admin"
requestMsg.putContext
  requestMsg.CONTEXT_ACCOUNT_PASSWORD, "admin"

Set map = requestMsg.getContextMap

map.Reset
While (map.Next)
  sKey = map.getKey
  sValue = map.getValue
Wend
```

PutContext

Legt den Wert für den angegebenen Namen in den Kontexteigenschaften fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt.

„Context“-Eigenschaften enthalten die folgenden Konstanten: Konto-ID, Kontokennwort, Dienstname und Dienstmethode.

Syntax

```
Sub putContext(name As String, value As String)
```

Parameter

- Name: Der Name, unter dem der angegebene Wert zugeordnet werden soll.
- Value: Der Wert, der dem spezifischen Namen zugeordnet werden soll.

Ergebnisse

Keine.

Beispiel

```
Dim requestMsg As New G1CLIENTLib.Message  
  
requestMsg.putContext  
  requestMsg.CONTEXT_ACOUNT_ID, "admin"  
requestMsg.putContext  
  requestMsg.CONTEXT_ACOUNT_PASSWORD, "admin"
```

PutContextMap

Fügt den aktuellen Kontexteigenschaften die neuen Kontexteigenschaften hinzu.

Syntax

```
Sub putContextMap(context As Map)
```

Parameter

- Die neue Kontextzuordnung, die der aktuellen Kontextzuordnung hinzugefügt werden soll.

Ergebnisse

Keine.

Beispiel

```
Dim map As New G1CLIENTLib.Map
Dim requestMsg As New G1UBCAPICOMLib.Message

map.Insert requestMsg.CONTEXT_ACCOUNT_ID, "admin"
map.Insert requestMsg.CONTEXT_ACCOUNT_PASSWORD, "admin"

requestMsg.putContextMap map
```

SetContextMap

Überschreibt die aktuellen Kontexteigenschaften mit den neuen Kontexteigenschaften.

Syntax

```
Sub setContextMap(context As Map)
```

Parameter

- Die neue Kontextzuordnung, die die aktuelle Kontextzuordnung ersetzen soll.

Ergebnisse

Keine.

Beispiel

```
Dim map As New G1CLIENTLib.Map
Dim requestMsg As New G1UBCAPICOMLib.Message

map.Insert requestMsg.CONTEXT_ACCOUNT_ID, "admin"
map.Insert requestMsg.CONTEXT_ACCOUNT_PASSWORD, "admin"

requestMsg.setContextMap map
```

GetOption

Ruft den Wert der Optionsentität ab, der durch den Namen im Optionsabschnitt der Nachricht identifiziert wird. Die Optionsentitäten umfassen dienstspezifische Laufzeitoptionen, z. B. die Groß-/Kleinschreibung der Ausgabe, das Format der Ausgabedaten usw.

Syntax

```
Function getOption(name As String) As String
```

Parameter

- Name: Der Name, dessen zugeordneter Wert zurückgegeben werden soll.

Ergebnisse

- Zeichenfolge: Der Wert der benannten Entität oder der leeren Zeichenfolge, wenn die benannte Entität nicht vorhanden ist.

Beispiel

```
Dim msg As New G1CLIENTLib.Message
Dim optionValue As String

OptionValue = msg.getOption("OutputCasing")
```

GetOptions

Ruft die Zuordnung ab, die alle Optionseinträge enthält.

Syntax

```
Function getOptions() As Map
```

Parameter

- Keine

Ergebnisse

Gibt die Zuordnung zurück, die alle Optionseinträge enthält.

Beispiel

```
Dim map As New G1CLIENTLib.Map
Dim requestMsg As New G1CLIENTLib.Message
Dim sKey As String
Dim sValue As String

requestMsg.putOption "OutputCasing", "M"
requestMsg.putOption "OutputRecordType", "A"

Set map = requestMsg.getOptions
```

```
map.Reset
While (map.Next)
  sKey = map.GetKey
  sValue = map.GetValue
Wend
```

PutOption

Legt den Wert für den angegebenen Namen in den Optionseigenschaften fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt. „Option“-Eigenschaften sind die dienstspezifischen Laufzeitoptionen.

Syntax

```
Sub putOption(name As String, value As String)
```

Parameter

- Name: Der Name, unter dem der angegebene Wert zugeordnet werden soll.
- Value: Der Wert, der dem spezifischen Namen zugeordnet werden soll.

Beispiel

```
Dim requestMsg As New G1CLIENTLib.Message
requestMsg.putOption "OutputCasing", "M"
requestMsg.putOption "OutputRecordType", "A"
```

PutOptions

Fügt die neuen Optionseigenschaften zu den aktuellen Optionseigenschaften hinzu.

Syntax

```
Sub putOptions(options As Map)
```

Parameter

- Die neue Optionszuordnung, die den aktuellen Optionseigenschaften hinzugefügt werden soll.

Beispiel

```
Dim map As New G1CLIENTLib.Map
Dim requestMsg As New G1CLIENTLib.Message

map.Insert "OutputCasing", "M"
map.Insert "OutputRecordType", "A"

requestMsg.putOptions map
```

SetOptions

Überschreibt die aktuellen Optionseigenschaften mit den neuen Optionseigenschaften.

Syntax

```
Sub setOptions(options As Map)
```

Parameter

- Die neue Optionszuordnung, durch die die aktuelle Optionszuordnung ersetzt werden soll

Beispiel

```
Dim map As New G1CLIENTLib.Map
Dim requestMsg As New G1CLIENTLib.Message

map.Insert "OutputCasing", "M"
map.Insert "OutputRecordType", "A"

requestMsg.setOptions map
```

GetError

Ruft den Fehler aus der Fehlermeldung ab.

Syntax

```
Function getError() As String
```

Parameter

- Keine

Ergebnisse

Gibt die Fehlermeldung in der Nachricht zurück.

Beispiel

```
Dim sErrorMessage As String
...
sErrorMessage = replyMsg.getError()
```

GetDataTable

Ruft die Datentabelle (DataTable) in der Nachricht ab.

Syntax

```
Function getDataTable() As DataTable
```

Parameter

- Keine.

Beispiel

```
Dim DataTable AS G1CLIENTLib.dataTable
Set DataTable = message.getDataTable
```

DataTable

DataTable enthält die Datensätze für Ihre Eingabe- und Ausgabedaten. Mithilfe der diesem Objekt zugeordneten Methoden definieren Sie die Spaltennamen für Ihre Ausgabe und fügen der Datentabelle (DataTable) Zeilen hinzu.

AddColumn

Fügt eine neue Spalte zur Datentabelle (DataTable) hinzu.

Syntax

```
Function addColumn(columnName As String) As Integer
```

Parameter

- Spaltenname

Ergebnisse

Gibt den Spaltenindex zurück.

Ausnahmen

- leerer Spaltenname
- doppelter Spaltenname

Beispiel

```
Dim dataTable As G1CLIENTLib.dataTable
dataTable.addColumn "AddressLine1"
dataTable.addColumn "City"
```

GetColumnNames

Ruft alle Spaltennamen ab.

Syntax

```
Syntax Function getColumnNames() As String()
```

Parameter

- Keine

Ergebnisse

Gibt das Array mit Spaltennamen zurück.

Beispiel

```
Dim sColumnNames() As String
Dim sColumnName As String
Dim nColumn As Integer

ReDim sColumnNames(returnDataTable.getColumnCount) As String
```

```
sColumnName = returnDataTable.getColumnNames  
  
For nColumn = 0 To dataRow.getColumnCount - 1  
sColumnName = sColumnNames (nColumn)  
Next
```

GetColumnIndex

Ruft den entsprechenden Spaltenindex ab.

Syntax

```
Function getColumnIndex(columnName As String) As Integer
```

Parameter

- Spaltenname

Ergebnisse

Gibt den entsprechenden Spaltenindex zurück.

Beispiel

```
Dim nIndex As Integer  
nIndex = dataTable.getColumnIndex("AddressLine1")
```

GetColumnCount

Ruft die Anzahl der Spalten in der Datentabelle (DataTable) ab.

Syntax

```
Function getColumnCount() As Integer
```

Parameter

- Keine

Ergebnisse

Gibt die Anzahl der Spalten zurück.

Beispiel

```
Dim nColumnCount As Integer  
nColumnCount = dataTable.getColumnCount()
```

Clear

Löscht die Daten in der Datentabelle (DataTable).

Syntax

```
Sub clear()
```

Parameter

- Keine

Beispiel

```
dataTable.clear()
```

Iterator

Ein Iterator, der alle Datenzeilen (DataRows) in der Datentabelle (DataTable) enthält.

Syntax

```
Syntax Function iterator() As DataRow()
```

Parameter

- Keine

Ergebnisse

Gibt einen Iterator zurück, der alle Datenzeilen (DataRows) in der Datentabelle (DataTable) enthält.

Beispiel

```
Dim returnDataTable As G1CLIENTLib.dataTable  
Dim row As G1CLIENTLib.DataRow  
Dim sColumnName As String  
Dim sFieldValue As String
```

```

Dim rows() As Variant
Dim nRow As Integer
Dim nColumn As Integer

'Get the result from the response message
Set returnDataTable = replyMsg.getDataTable
ReDim rows(returnDataTable.getRowCount) As Variant

rows = returnDataTable.iterator

For nRow = 0 To returnDataTable.getRowCount - 1
Set row = rows(nRow)

For nColumn = 0 To row.getColumnCount - 1
sColumnName = row.getColumnNames(nColumn)
sFieldValue = row.getByIndex(nColumn)
Next
Next

```

AddRow

Fügt der Datentabelle (DataTable) eine Datenzeile (DataRow) hinzu.

Syntax

```
Sub addRow(DataRow As DataRow)
```

Parameter

- **DataRow:** Dies ist die zur Datentabelle (DataTable) hinzuzufügende Datenzeile (DataRow).

Ergebnisse

Keine.

Beispiel

```

Dim dataTable As G1CLIENTLib.dataTable
Dim newRow As G1CLIENTLib.DataRow

Set dataTable=requestMsg.getDataTable
dataTable.addColumn("AddressLine1")
dataTable.addColumn("City")
dataTable.addColumn("State")
Set newRow=dataTable.newRow
newRow.setByIndex 0, "10535 Boyer"
newRow.setByIndex 1, "Austin"

```

```
newRow.setByIndex 2, "Texas"
dataTable.addRow newRow
```

NewRow

Erstellt eine neue Datenzeile (DataRow) in der Datentabelle (DataTable).

Syntax

```
Function newRow() As DataRow
```

Parameter

- Keine

Ergebnisse

Gibt die neu erstellte Datenzeile (DataRow) zurück.

Beispiel

```
Dim dataTable As G1CLIENTLib.dataTable
Dim newRow As G1CLIENTlib.DataRow

Set dataTable=requestMsg.getDataTable

Set newRow=dataTable.newRow
newRow.setByName "AddressLine1","10535 Boyer"
newRow.setByName "City", "Austin"
newRow.setByName "State", "Texas"
dataTable.addRow newRow
```

GetRowCount

Ruft die Anzahl der Datenzeilen (DataRow) in der Datentabelle (DataTable) ab.

Syntax

```
Function getRowCount() As Integer
```

Parameter

- Keine

Ergebnisse

Gibt die Anzahl der Datenzeilen (DataRow) in der Datentabelle (DataTable) zurück.

Beispiel

```
Dim nRowCount As Integer  
nRowCount = dataTable.getRowCount
```

Merge

Führt die vorhandene Datentabelle (DataTable) und die aktuelle Datentabelle zusammen.

Syntax

```
Sub merge(other As DataTable)
```

Parameter

- Die andere Datentabelle (DataTable), die mit der aktuellen Datentabelle zusammengeführt werden soll.

Ergebnisse

Keine.

Beispiel

```
Dim otherDataTable As New G1CLIENTlib.dataTable  
...  
dataTable.merge(otherDataTable)
```

DataRow

DataRow enthält die individuellen Datensätze für Ihre Eingabe- und Ausgabedaten. Mit den Methoden, die dieser Klasse zugeordnet sind, definieren Sie die Spaltennamen für Ihre Ausgabe und fügen Datensätze zur DataTable (Datentabelle) hinzu.

GetColumnNames

Ruft alle Spaltennamen ab.

Syntax

```
Function getColumnNames() As String()
```

Parameter

- Keine

Ergebnisse

Gibt das Array mit Spaltennamen zurück.

Beispiel

```
Dim sColumnNames() As String
Dim sColumnName As String
Dim nColumn As Integer
ReDim sColumnNames(dataRow.getColumnCount) As String
sColumnName = sColumnNames(nColumn)
For nColumn = 0 To dataRow.getColumnCount - 1
    sColumnName = sColumnNames(nColumn)
Next
```

GetColumnIndex

Ruft den entsprechenden Spaltenindex ab.

Syntax

```
Function getColumnIndex(columnName As String) As Integer
```

Parameter

- Spaltenname

Ergebnisse

Gibt den entsprechenden Spaltenindex zurück.

Beispiel

```
Dim nIndex As Integer  
nIndex = dataRow.getColumnIndex("AddressLine1")
```

GetColumnCount

Ruft die Anzahl von Spalten in der Datenzeile (DataRow) ab.

Syntax

```
Function getColumnCount() As Integer
```

Parameter

- Keine

Ergebnisse

Gibt die Anzahl der Spalten zurück.

Beispiel

```
Dim nColumnCount As Integer  
nColumnCount = dataRow.getColumnCount()
```

GetByIndex

Ruft den Wert aus dem Feld-Array nach dem Spaltenindex in dieser Datenzeile (DataRow) ab.

Syntax

```
Function getByIndex(index As Integer) As String
```

Parameter

- Der Index, dem der angegebene Wert zugeordnet werden soll.

Ergebnisse

Gibt den Wert für den Spaltenindex in dieser Datenzeile (DataRow) zurück. Gibt eine leere Zeichenfolge zurück, wenn der Index ungültig ist.

Beispiel

```
Dim sValue As String
sValue = dataRow.GetByIndex(1)
```

GetByName

Ruft den Wert aus dem Feld-Array nach dem Spaltennamen in dieser Datenzeile (DataRow) ab.

Syntax

```
Function GetByName(columnName As String) As String
```

Parameter

- Name, unter dem der angegebene Wert zugeordnet werden soll.

Ergebnisse

Gibt den Wert für den Spaltennamen in dieser Datenzeile (DataRow) zurück, und gibt eine leere Zeichenfolge zurück, wenn der Spaltenname nicht vorhanden ist.

Beispiel

```
Dim sValue As String
sValue = dataRow.GetByName("City")
```

Merge

Führt die angegebene Datenzeile (DataRow) mit der aktuellen Datenzeile zusammen.

Syntax

```
Sub Merge(other As DataRow)
```

Parameter

- Dies ist die andere Datenzeile (DataRow), die mit der aktuellen Datenzeile zusammengeführt werden soll.

Ergebnisse

Keine.

Beispiel

```
Dim otherDataRow As New G1CLIENTlib.DataRow
...
dataRow.merge(otherDataRow)
```

SetByName

Legt den Wert für die entsprechende Spalte der Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.

Syntax

```
Sub setByName(columnName As String, value As String)
```

Parameter

- Name, unter dem der angegebene Wert zugeordnet werden soll.
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ergebnisse

Keine.

Ausnahmen

- leerer Spaltenname
- doppelter Spaltenname

Beispiel

```
Dim newRow As G1CLIENTlib.DataRow
Set newRow= dataTable.netRow
newRow.setByName "AddressLine1", "100 Congress"
newRow.setByName "City", "Austin"
newRow.setByName "State", "Texas"
dataTable.addRow newRow
```

SetByIndex

Legt den Wert für die entsprechende Spalte der Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.

Syntax

```
Sub setByIndex(index As Integer, value As String)
```

Parameter

- Der Spaltenindex, dem der angegebene Wert zugeordnet werden soll
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ergebnisse

Keine.

Ausnahmen

- Der Spaltenindex ist ungültig.

Beispiel

```
Dim newRow As G1CLIENTLib.DataRow  
Set newRow= dataTable.netRow  
newRow.setByIndex 0, "100 Congress"  
newRow.setByIndex 1, "Austin"  
newRow.setByIndex 2, "Texas"  
dataTable.addRow newRow
```

AddChild

Fügt der benannten übergeordneten/untergeordneten Beziehung eine Datenzeile (DataRow) hinzu. Wenn die benannte Beziehung vorhanden ist, wird die angegebene Datenzeile (DataRow) der vorhandenen Datenzeilensammlung angefügt. Anderenfalls wird eine neue Sammlung mit der angegebenen Datenzeile (DataRow) als einziges Element erstellt.

Syntax

```
Sub addChild( childName As String, childDataRow As DataRow)
```

Parameter

- Der Name der übergeordneten/untergeordneten Beziehung (z. B. „Überschwemmungsgebietsdaten“, „Referenzen“, „Verwendet von“ usw.)
- Die Datenzeile (DataRow), die der Beziehung hinzugefügt werden soll.

Ergebnisse

Keine.

Beispiel

```
Dim dataRow As New G1CLIENTLib.dataRow
Dim childDataRow As New G1CLIENTLib.dataRow

childDataRow .setByName "Address", "100 Congress"
childDataRow .setByName "City", "Austin"

dataRow.addChild "child1", dataRow
```

GetChildren

Ruft die untergeordneten Zeilen einer benannten Beziehung ab.

Syntax

```
Function getChildren(childName As String) As DataRow()
```

Parameter

- Der Name der übergeordneten/untergeordneten Beziehung (z. B. „Überschwemmungsgebietsdaten“, „Referenzen“, „Verwendet von“ usw.)

Ergebnis

Gibt die untergeordneten Zeilen aus der benannten Beziehung zurück.

Beispiel

```
Dim dataRow As New G1CLIENTLib.dataRow
' Assume that dataRow has children .....
' Or more code to be needed
Dim rowsChild1() As Variant
rowsChild1 = dataRow.getChildren("child1")
```

ListChildNames

Ruft alle Namen der benannten übergeordneten/untergeordneten Beziehungen.

Syntax

```
Function listChildNames() As String()
```

Parameter

Keine.

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

```
Dim dataRow As New G1CLIENTLib.dataRow
' Assume that dataRow has children .....
' Or more code to be needed
Dim sChildNames() As String
sChildNames = dataRow.listChildNames
```

SetChildren

Legt die Zeilen einer angegebenen benannten übergeordneten/untergeordneten Beziehung fest. Wenn zuvor Zeilen mit diesem Namen vorhanden waren, werden diese an den Aufrufer zurückgegeben.

Syntax

```
Function setChildren(childName As String, DataRows As DataRow()) As
DataRow()
```

Parameter

Keine.

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

```
Dim dataRow1 As New G1CLIENTLib.dataRow
Dim dataRow2 As New G1CLIENTLib.dataRow
dataRow1.setByName "Address", "100 Congress"
dataRow1.setByName "City", "Austin"
dataRow2.setByName "Address", "200 Congress"
dataRow2.setByName "City", "Austin"

Dim rows(1) As G1CLIENTLib.dataRow

Set rows(0) = dataRow1
Set rows(1) = dataRow2
```

```
Dim newRows() As Variant
newRows = dataRowSpt.setChildren("child1", rows())
```

Map

Map ist ein Objekt, das Werten Schlüssel zuordnet. In einer Zuordnung dürfen keine doppelten Schlüssel enthalten sein: Jeder Schlüssel kann höchstens einem Wert zugeordnet werden.

Reset

Setzt den Cursor vor die erste Zuordnung.

Syntax

```
Sub Reset()
```

Parameter

- Keine

Ergebnisse

Keine.

Beispiel

```
Dim requestMsg As New G1CLIENTLib.Message
Dim map As G1CLIENTLib.Map
Dim sKey As String
Dim sValue As String

requestMsg.putContext requestMsg.CONTEXT_ACCOUNT_ID, "admin"
requestMsg.putContext requestMsg.CONTEXT_ACCOUNT_PASSWORD, "admin"

Set map = requestMsg.getContextMap

map.Reset
While (map.Next)
    sKey = map.getKey
    sValue = map.getValue
Wend
```


Next

Bewegt den Cursor von seiner aktuellen Position eine Zuordnung nach unten.

Syntax

```
Sub Next()
```

Parameter

- Keine

Beispiel

```
Dim requestMsg As New G1CLIENTLib.Message
Dim map As G1CLIENTLib.Map
Dim sKey As String
Dim sValue As String

requestMsg.putContext requestMsg.CONTEXT_ACCOUNT_ID, "admin"
requestMsg.putContext requestMsg.CONTEXT_ACCOUNT_PASSWORD, "admin"

Set map = requestMsg.getContextMap

map.Reset
While (map.Next)
    sKey = map.getKey
    sValue = map.getValue
Wend
```

GetKey

Ruft den Schlüssel in der aktuellen Zuordnung ab.

Syntax

```
Function getKey() As String
```

Parameter

- Keine

Ergebnisse

Gibt den Schlüssel in der aktuellen Zuordnung zurück.

Beispiel

```

Dim requestMsg As New G1CLIENTLib.Message
Dim map As G1CLIENTLib.Map
Dim sKey As String
Dim sValue As String

requestMsg.putContext requestMsg.CONTEXT_ACCOUNT_ID, "admin"
requestMsg.putContext requestMsg.CONTEXT_ACCOUNT_PASSWORD, "admin"

Set map = requestMsg.getContextMap

map.Reset
While (map.Next)
    sKey = map.getKey
    sValue = map.getValue
Wend

```

GetValue

Ruft den Wert in der aktuellen Zuordnung ab.

Syntax

```
Function getValue() As String
```

Parameter

- Keine

Ergebnisse

Gibt den Wert in der aktuellen Zuordnung zurück.

Beispiel

```

Dim requestMsg As New G1CLIENTLib.Message
Dim map As G1CLIENTLib.Map
Dim sKey As String
Dim sValue As String

requestMsg.putContext requestMsg.CONTEXT_ACCOUNT_ID, "admin"
requestMsg.putContext requestMsg.CONTEXT_ACCOUNT_PASSWORD, "admin"

Set map = requestMsg.getContextMap

map.Reset
While (map.Next)

```

```
sKey = map.getKey  
sValue = map.getValue  
Wend
```

5 - Java-API

In this section

Einführung	157
Server	161
Service	168
Message	169
DataTable	175
DataRow	181

Einführung

Eine Java-Klasse ist eine Blaupause oder ein Prototyp, die bzw. der die Variablen und Methoden definiert, die von allen Objekten eines bestimmten Typs verwendet werden. Eine Java-Klasse definiert zudem die Implementierung eines bestimmten Objekttyps. Mit diesen Klassen können Sie Java-Anwendungen erstellen. Java-Objekte werden im Allgemeinen von Java-Klassen erstellt.

Ein Java-Objekt ist eine Sammlung von entsprechenden Variablen und Methoden, die unter Verwendung der Java Virtual Machine (JVM) in der Sprache Java geschrieben wurden. Die einer Klasse oder einem Objekt zugeordneten Daten werden in Variablen gespeichert. Das einer Klasse oder einem Objekt zugeordnete Verhalten wird mithilfe von Methoden implementiert. Die Methoden ähneln den Funktionen oder Verfahren in Prozedursprachen wie C.

Java-Softwareobjekte interagieren und kommunizieren über Nachrichten miteinander. Zusätzliche Informationen, die das empfangende Objekt zum Ausführen der zugehörigen Aufgaben eventuell benötigt, werden durch Parameter übergeben.

Weitere Informationen zur Java-Technologie finden Sie unter www.oracle.com/java.

Konstanten

Die Java-API verwendet zwei Konstantensätze. Der erste Satz ist für die `Server`-Komponente, die in der unteren Tabelle beschrieben wird.

Tabelle 21: Konstanten für die `Server`-Komponente

Name der Konstante	Beschreibung	Beispiel
<code>Server.HOST</code>	Zeichenfolge für den Hostnamen des Servers. Der Standardname lautet „localhost“.	65.89.200.89
<code>Server.PORT</code>	Zeichenfolge für den Serverport. Der Standardwert ist „8080“.	10119
<code>Server.ACCOUNT_ID</code>	Zeichenfolge für die Serverkonto-ID. Der Standardwert ist NULL.	user1

Name der Konstante	Beschreibung	Beispiel
Server.ACCOUNT_PASSWORD	Zeichenfolge für das Kennwort des Serverkontos. Der Standardwert ist NULL.	user1
Server.CONNECTION_TIMEOUT	Zeichenfolge für das Timeout der Serververbindung in Millisekunden. Der Standardwert ist „10000“.	50000
Server.CONNECTION_TYPE	Zeichenfolge für den Serververbindungstyp. Derzeit werden nur HTTP, HTTPS und SOCKET unterstützt. Der Standardwert lautet „HTTP“.	HTTP
Server.PROXY_HOST	Zeichenfolge für den Hostnamen des Proxy-Servers. Der Standardwert ist NULL.	192.168.1.77
Server.PROXY_PORT	Zeichenfolge für den Proxy-Serverport. Der Standardwert ist NULL.	8080
Server.PROXY_USER	Zeichenfolge für die Konto-ID des Proxy-Servers. Der Standardwert ist NULL.	user1
Server.PROXY_PASSWORD	Zeichenfolge für das Kontokennwort des Proxy-Servers. Der Standardwert ist NULL.	user1
Server.INPUT_CLEANUP	<p>Boolescher Wert, der angibt, ob die Sonderzeichen in den Eingabedaten entfernt werden müssen. Der Standardwert ist „false“.</p> <p>Anmerkung: Wenn dieses Attribut auf „Falsch“ festgelegt ist und die Eingabedaten Sonderzeichen enthalten, tritt eine Ausnahme auf.</p> <p>Wichtig: Legen Sie diesen Wert nur dann auf „Wahr“ fest, wenn Sie sicher sind, dass in den Eingabedaten Sonderzeichen enthalten sind. Eine Aktivierung dieses Attributs würde sich sonst negativ auf die Leistung auswirken.</p>	true

Der zweite Konstantensatz ist für die `Message`-Komponente vorgesehen.

Tabelle 22: Konstanten für die Message-Komponente

Name der Konstante	Beschreibung/Standard	Beispiel
Message.CONTEXT_ACCOUNT_ID	Zeichenfolge für den Nachrichtentext der Konto-ID.	user1
Message.CONTEXT_ACCOUNT_PASSWORD	Zeichenfolge für das Kontokennwort des Nachrichtenkontextes.	user1
Message.CONTEXT_SERVICE_NAME	Zeichenfolge für den Dienstnamen des Nachrichtenkontextes.	echoservice
Message.CONTEXT_SPECTRUM_DISPLAY_VERSION	Zeichenfolge für die Spectrum-Anzeigeversion des Nachrichtenkontextes.	12.1
Message.CONTEXT_SPECTRUM_SERVER_VERSION	Zeichenfolge für die Spectrum-Serverversion des Nachrichtenkontextes.	12.1

Fehlermeldungen

Die Java-API verwendet die folgenden Fehlermeldungen:

- Fehlermeldungen für Verbindungen
 - "Connection type not supported."
 - "Client timeout"
- Fehlermeldungen zum Erstellen von Datentabellen (DataTables):
 - "Blank column name"
 - "Duplicated column name"
 - "Index is out of bounds"
- Fehlermeldungen für Ausnahmen beim Verpacken der Nachrichten
 - "Cannot pack null Message"
 - "Input Message is null"
 - "Unable to connect to Server:"

- "Failed to get Service"
- "Unknown serialization type:"
- "Unknown encoding type:"
- "Gateway is not connected" (for SOCKET)

Beispielanwendung

Der unten dargestellte Beispielcode zeigt, wie die Java-API verwendet wird.

```
try
{
    // Create Server
    Server server = new Server();

    // Set server connection properties
    server.setConnectionProperty(Server.HOST, "localhost");
    server.setConnectionProperty(Server.PORT, "10119");
    server.setConnectionProperty(Server.CONNECTION_TYPE, "SOCKET");
    server.setConnectionProperty(Server.ACCOUNT_ID, "guest");
    server.setConnectionProperty(Server.ACCOUNT_PASSWORD, "");

    // Connect to server
    server.connect();

    // Get Service From Server
    Service service = server.getService("ValidateAddress");

    // Create Input Message
    Message request = new Message();

    // Fill DataTable in the input message
    DataTable dataTable = request.getDataTable();
    DataRow row1 = dataTable.newRow();
    row1.set("AddressLine1", "4200 Parliament Place");
    row1.set("City", "Lanham");
    row1.set("StateProvince", "Maryland");
    dataTable.addRow(row1);
    DataRow row2 = dataTable.newRow();
    row2.set("AddressLine1", "100 Congress");
    row2.set("City", "Austin");
    row2.set("StateProvince", "Texas");
    dataTable.addRow(row2);

    // Set "option" Properties to the Input
    Message request.putOption("OutputCasing", "M");
    request.putOption("OutputRecordType", "A");
}
```



```

// Process Input Message, return output Message
Message reply = service.process(request);

// Disconnect from server
server.disconnect();

// Get the result from the response message
DataTable returnDataTable = reply.getDataTable();
String[] columnNames = returnDataTable.getColumnNames();
Iterator iter = returnDataTable.iterator();
while (iter.hasNext())
{
    DataRow row = (DataRow) iter.next();
    for (int col = 0; col < returnDataTable.getColumnCount();
col++)
    {
        String value = row.get(columnNames[col]);
        System.out.println(value);
    }
}
catch (Exception e)
{
    System.out.println("Error Occurred, " + e.getMessage());
}

```

Server

Mithilfe der `Server`-Klasse können Sie eine Verbindung zum Server aufbauen, die Verbindung zum Server trennen und den Dienst vom Server abrufen.

Connect

Liest die Eigenschaften, um zu bestimmen, welche Gateway-Verbindung verwendet werden soll, und stellt eine Verbindung zum Server her. Sie können über HTTP, HTTPS oder SOCKET eine Verbindung herstellen. Über HTTP und HTTPS wird eigentlich erst dann eine Verbindung zum Server hergestellt, wenn die Methode „GetService“ oder „Process“ aufgerufen wird. Bei dem Verbindungstyp SOCKET ist die Methode `Connect` voll funktionsfähig.

Syntax

```
public void connect()
```

Parameter

Keine.

Ergebnisse

Auslösungen:

- **ConfigurationException:** Wenn durch eine ungültige Konfiguration keine Verbindung zum Server hergestellt werden kann. Ein unbekanntes Protokoll verursacht beispielsweise eine **ConfigurationException**. Bei dem Versuch, die Methode „connect()“ erneut auszuführen, ist bei Auftreten dieses Fehlers kein Wert vorhanden.
- **ConnectionException:** Wenn keine Verbindung zum Server hergestellt werden kann. Abhängig von der zugrunde liegenden Ursache für die Ausnahme kann die Verbindung eventuell wiederhergestellt werden.
- **MessageProcessingException:** Wenn auf dem Server ein Fehler auftritt, der nicht durch Konfigurations- oder Verbindungsprobleme verursacht wurde.

Beispiel

```

Server server = new Server();

server.setConnectionProperty(Server.HOST, "localhost");
server.setConnectionProperty(Server.PORT, "10119");
server.setConnectionProperty(Server.CONNECTION_TYPE, "SOCKET");
server.setConnectionProperty(Server.ACCOUNT_ID, "guest");
server.setConnectionProperty(Server.ACCOUNT_PASSWORD, "");

try
{
    //Connect to server
    server.connect();
}
catch (ConfigurationException e)
{
    // indicate an error with configuration
}
catch (ConnectionException e)
{
    // handle connection issue (retry, report error, etc.)
}
catch (MessageProcessingException e)
{
    // report error
}

```

Verbindungspooling

Verbindungspooling für den Verbindungstyp SOCKET ist für den Java-Client verfügbar. In diesem Abschnitt wird beschrieben, wie Verbindungspooling aktiviert und deaktiviert wird. Verbindungspooling ist standardmäßig deaktiviert.

So aktivieren Sie das Verbindungspooling:

```
Server server = new Server();
Server.setConnectionProperty(Connection.SOCKET_POOL, "true");
```

So deaktivieren Sie das Verbindungspooling:

```
Server server = new Server();
Server.setConnectionProperty(Connection.SOCKET_POOL, "false");
```

Wenn Verbindungspooling aktiviert ist, leiht sich die Methode „connect()“ eine Verbindung aus dem Pool und die Methode „disconnect()“ gibt die Verbindung wieder an den Pool zurück. Beim Pooling muss der Client die Methode „disconnect()“ jedes Mal aufrufen, um die Verbindung an den Pool zurückzugeben.

Jeder Thread sollte seinen eigenen Server enthalten, wie im folgenden Beispiel dargestellt:

```
{
    ...
    Server server = new Server();
    server.setConnectionProperty(Server.HOST, "localhost");
    server.setConnectionProperty(Server.PORT, "10119");
    server.setConnectionProperty(Server.CONNECTION_TYPE, "SOCKET");
    server.setConnectionProperty(Server.ACCOUNT_ID, "yourID");
    server.setConnectionProperty(Server.ACCOUNT_PASSWORD, "pwd");
    server.setConnectionProperty(Connection.SOCKET_POOL, "true");
    server.setConnectionProperty(Connection.SOCKET_POOL_MAX_ACTIVE, "20");

    server.setConnectionProperty(Connection.SOCKET_POOL_MIN_IDLE, "10");
    server.setConnectionProperty(Connection.SOCKET_POOL_MAX_TOTAL, "25");

    server.connect();
    ...
    service = server.getService(serviceName);
    reply = service.process(requestMessage);
    server.disconnect();
    ...
}
```

In der folgenden Tabelle werden die Konstanten aufgeführt, die Sie für Verbindungspooling verwenden können.

Tabelle 23: Konstanten für Verbindungspooling

Name der Konstante	Beschreibung
SOCKET_POOL	Gibt an, ob beim Verbindungstyp SOCKET Verbindungspooling verwendet werden soll. Die Werte sind „wahr“ oder „falsch“. Der Standardwert ist „false“.
SOCKET_POOL_MAX_ACTIVE*	Die maximale Anzahl der aktiven Socket-Verbindungen, die vom Pool ausgeliehen werden können. Der Standardwert ist -1 und gibt keinen maximalen Wert an.
SOCKET_POOL_MAX_IDLE*	Die maximale Anzahl der im Pool verbleibenden inaktive Socket-Verbindungen. Der Standardwert ist -1 und gibt keinen maximalen Wert an.
SOCKET_POOL_MAX_TOTAL*	Die maximale Gesamtzahl der Socket-Verbindungen im Pool (aktiv und inaktiv). Der Standardwert ist -1 und gibt keinen maximalen Wert an.
SOCKET_POOL_MAX_WAIT*	Die maximale Wartezeit (in Millisekunden) bevor eine Ausnahme ausgelöst werden soll, wenn der Pool erschöpft ist und der Wert der Aktion „Wenn erschöpft“ WHEN_EXHAUSTED_BLOCK lautet. Der Standardwert ist -1 und gibt keinen maximalen Wert an.
SOCKET_POOL_MIN_EVICTABLE_IDLE_TIME_MILLIS*	Die minimale Zeitdauer, die eine Verbindung in einem Pool inaktiv sein kann, bevor Sie entfernt werden kann. Der Standardwert ist 1800000 (30 Minuten).
SOCKET_POOL_MIN_IDLE*	Die minimal zulässige Anzahl der Verbindungen im Pool, bevor der Evictor-Thread (sofern aktiv) neue Verbindungen erstellt. Der Standardwert ist 0.

Name der Konstante	Beschreibung
SOCKET_POOL_NUM_TESTS_PER_EVICTION_RUN*	Legt die Anzahl der inaktiven Verbindungen fest, die während den einzelnen Ausführungen des Evictor-Threads (sofern aktiv) überprüft werden sollen. Der Standardwert ist -1 und gibt an, dass alle inaktiven Verbindungen geprüft wurden.
SOCKET_POOL_TEST_ON_BORROW*	Gibt an, ob Verbindungen überprüft werden, bevor sie vom Pool ausgeliehen werden. Der Standardwert ist „Wahr“.
SOCKET_POOL_TEST_ON_RETURN*	Gibt an, ob Verbindungen überprüft werden, bevor sie an den Pool zurückgegeben werden. Der Standardwert ist „false“.
SOCKET_POOL_TEST_WHILE_IDLE*	Gibt an, ob Verbindungen durch das Evictor-Thread für inaktive Verbindungen überprüft werden. Der Standardwert ist „false“.
SOCKET_POOL_TIME_BETWEEN_EVICTION_RUNS_MILLIS*	Legt die Anzahl der Millisekunden des Ruhezustands zwischen Ausführungen des Evictor-Threads für inaktive Verbindungen fest. Wenn der Wert auf null oder eine negative Zahl festgelegt ist, wird kein Evictor-Thread für inaktive Verbindungen ausgeführt. Der Standardwert ist 300000 (5 Minuten).
SOCKET_POOL_WHEN_EXHAUSTED_ACTION*	Legt die Aktion „Wenn erschöpft“ fest, die ausgeführt werden soll, wenn versucht wird, eine Verbindung zu leihen und keine Verbindung verfügbar ist. Der Standardwert ist SOCKET_POOL_WHEN_EXHAUSTED_BLOCK.
SOCKET_POOL_WHEN_EXHAUSTED_BLOCK*	Der Aktionstyp „Wenn erschöpft“, der angibt, dass der Aufrufer so lange blockiert werden sollte, bis ein neues Objekt verfügbar ist, wenn versucht wird, eine Verbindung auszuleihen und keine Verbindungen verfügbar sind. Alternativ kann es auch sein, dass die maximale Wartezeit verstrichen ist.

Name der Konstante	Beschreibung
SOCKET_POOL_WHEN_EXHAUSTED_FAIL*	Der Aktionstyp „Wenn erschöpft“, der angibt, dass der Aufrufer fehlschlagen sollte, wenn versucht wird, eine Verbindung auszuleihen und keine Verbindungen verfügbar sind. Dadurch wird eine <code>ConnectionException</code> ausgelöst.
SOCKET_POOL_WHEN_EXHAUSTED_GROW*	Der Aktionstyp „Wenn erschöpft“, der angibt, dass ohnehin eine neue Verbindung hergestellt wird, wenn versucht wird, eine Verbindung auszuleihen und keine Verbindungen verfügbar sind.

* Gilt nur, wenn der Verbindungstyp `SOCKET` verwendet wird und das Verbindungspooling aktiviert ist.

Disconnect

Trennt die Verbindung zum Server.

Syntax

```
public void disconnect()
```

Parameter

Keine.

Ergebnisse

Die Client-Verbindung zum Server wird getrennt.

Beispiel

```
...
//Disconnect from server
server.disconnect();
```

SetConnectionProperty

Legt die Konfigurationseigenschaften der Serververbindung fest, z. B. den Servernamen und die Timeout-Dauer.

Syntax

```
public void setConnectionProperty(String name, String value)
```

Parameter

- Name: Dies ist der Name der Verbindungseigenschaft wie HOST.
- Value: Dies ist der Wert für den Namen der Verbindungseigenschaft wie „www.myhost.com“

Ergebnisse

Keine.

Ausnahmen

- ERROR_INVALID_COLUMN_NAME: Ein leerer Spaltenname oder ein Spaltenname mit NULL-Wert.
- ERROR_INVALID_VALUE: Ein NULL-Wert.

Beispiel

```
Server server = new Server();  
  
server.setConnectionProperty(Server.HOST, "localhost");  
server.setConnectionProperty(Server.PORT, "8080");  
  
//Connect to server  
server.connect();
```

GetService

Ruft den Dienst vom Server ab.

Syntax

```
public Service getService(String serviceName)
```

Parameter

- Name – Der Name des Dienstes

Ergebnisse

Gibt den bestimmten Dienst zurück.

Ausnahmen

Löst `ServiceNotFoundException`, `ServiceCreationException` aus

Beispiel

```
Service service = server.getService("ValidateAddress");
```

Service

Die `Service`-Klasse dient der Verarbeitung der Nachricht (d. h., dass sie die Nachricht an den Server sendet und eine Antwort vom Server erhält).

Process

Verarbeitet die Eingabennachricht und gibt die Antwortnachricht zurück.

Syntax

```
public Message process (Message message)
```

Parameter

- Eingabennachricht

Ergebnisse

Gibt die Antwortnachricht zurück.

Ausnahmen

- `TimeoutException`: Wenn durch eine ungültige Konfiguration keine Verbindung zum Server hergestellt werden kann. Ein unbekanntes Protokoll verursacht beispielsweise eine `ConfigurationException`. Bei dem Versuch, die Methode „connect()“ erneut auszuführen, ist bei Auftreten dieses Fehlers kein Wert vorhanden.
- `ConnectionException`: Wenn keine Verbindung zum Server hergestellt werden kann. Abhängig von der zugrunde liegenden Ursache für die Ausnahme kann die Verbindung eventuell wiederhergestellt werden.

- `MessageProcessingException`: Wenn auf dem Server ein Fehler auftritt, der nicht durch Konfigurations- oder Verbindungsprobleme verursacht wurde.

Beispiel

```
try
{
    //Process Input Message, return output Message
    Message response = service.process(message);
}
catch (ConnectionException e)
{
    // handle connection issue (retry, report error, etc.)
}
catch (TimeoutException e)
{
    // handle timeout issue (retry, report error, etc.)
}
catch (MessageProcessingException e)
{
    // report error
}
```

Message

Die `Message`-Klasse sendet Ihre Eingabedaten und empfängt Ihre Ausgabedaten vom Dienst. Die Eigenschaften für `Message` umfassen Kontexteigenschaften wie Konto-ID, Kontokennwort, Dienstname und Dienstmethode sowie Optionseigenschaften, bei denen es sich um die dienstspezifischen Laufzeitoptionen handelt.

getContext

Ruft den Wert nach dem Namen in den „Kontext“-Eigenschaften ab. Kontexteigenschaften enthalten die folgenden Konstanten: Konto-ID, Kontokennwort, Dienstname, Dienstschlüssel und Anforderungs-ID.

Syntax

```
public String getContext(String name)
```

Parameter

- Name: Der Name, dessen zugeordneter Wert zurückgegeben werden soll

Ergebnisse

Gibt den Wert zurück, der dem Namen in den Kontexteigenschaften zugeordnet ist. Wenn der Name nicht vorhanden ist, gibt die Methode NULL zurück.

Beispiel

```
String value = message.getContext(Message.CONTEXT_ACCOUNT_ID);
```

GetContext

Ruft die Zuordnung ab, die alle Kontexteinträge enthält.

Syntax

```
public Map getContext()
```

Parameter

- Keine

Ergebnisse

Gibt die Zuordnung zurück, die alle Kontexteinträge enthält.

Beispiel

```
Map context = message.getContext();
```

PutContext

Legt den Wert für den angegebenen Namen in den Kontexteigenschaften fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt. Kontexteigenschaften enthalten die folgenden Konstanten: Konto-ID, Kontokennwort, Dienstname, Dienstschlüssel und Anforderungs-ID.

Syntax

```
public void putContext(String name, String value)
```

Parameter

- Name – Der Name, dem der angegebene Wert zugeordnet werden soll
- Wert – Der Wert, dem der angegebene Name zugeordnet werden soll

Ergebnisse

Keine.

Beispiel

```
message.putContext(Message.CONTEXT_ACCOUNT_ID, "user1");
```

PutContext

Fügt den aktuellen Kontexteigenschaften die neuen Kontexteigenschaften hinzu.

Syntax

```
public void putContext(Map map)
```

Parameter

- Die neue Kontext-Hashtabelle, die der aktuellen Kontext-Hashtabelle hinzugefügt werden soll.

Ergebnisse

Keine.

Beispiel

```
Map context = new HashMap();  
...  
message.putContext(context);
```

SetContext

Überschreibt die aktuellen Kontexteigenschaften mit den neuen Kontexteigenschaften.

Syntax

```
public void setContext(Map map)
```

Parameter

- Die neue Kontextzuordnung, durch die die aktuelle Kontextzuordnung ersetzt wird.

Ergebnisse

Keine.

Beispiel

```
Map context = new Map ();  
...  
message.setContext(context);
```

GetOption

Ruft den Wert nach dem Namen in den Optionseigenschaften ab. Bei Optionseigenschaften handelt es sich um die dienstspezifischen Laufzeitoptionen.

Syntax

```
public String getOption(String name)
```

Parameter

- Name: Der Name, dessen zugeordneter Wert zurückgegeben werden soll.

Ergebnisse

Gibt den Wert für den Namen in den Optionseigenschaften in der Nachricht OR NULL zurück, wenn der Name nicht vorhanden ist.

Beispiel

```
String value = message.getOption("OutputCasing");
```

GetOptions

Ruft die Zuordnung ab, die alle Optionseinträge enthält.

Syntax

```
public Map getOptions();
```

Parameter

- Keine

Ergebnisse

Gibt die Zuordnung zurück, die alle Optionseinträge enthält.

Beispiel

```
Map options = message.getOptions();
```

PutOption

Legt den Wert für den angegebenen Namen in den Optionseigenschaften fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt. Bei Optionseigenschaften handelt es sich um die dienstspezifischen Laufzeitoptionen.

Syntax

```
public void setOption(String name, String value)
```

Parameter

- Name: Dies ist der Name, dem der angegebene Wert zugeordnet werden soll.
- Value: Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ergebnisse

Keine.

Beispiel

```
message.setOption("OutputCasing", "M");
```

PutOptions

Fügt die neuen Optionseigenschaften zu den aktuellen Optionseigenschaften hinzu.

Syntax

```
public void putOptions(Map map)
```

Parameter

- Die neue Optionszuordnung, die den aktuellen Optionseigenschaften hinzugefügt werden soll.

Beispiel

```
Map options = new HashMap();  
...  
message.putOptions(options);
```

SetOptions

Überschreibt die aktuellen Optionseigenschaften mit den neuen Optionseigenschaften.

Syntax

```
public void setOptions(Map map)
```

Parameter

- Die neue Optionszuordnung, durch die die aktuelle Optionszuordnung ersetzt werden soll

Ergebnisse

Keine.

Beispiel

```
Map options = new HashMap();  
...  
message.setOptions(options);
```

GetError

Ruft die Fehlermeldung aus der Nachricht ab.

Syntax

```
public String getError()
```

Parameter

- Keine

Ergebnisse

Gibt die Fehlermeldung in der Nachricht zurück.

Beispiel

```
String error = message.getError();
```

GetDataTable

Ruft die Datentabelle in dieser Nachricht ab.

Syntax

```
public DataTable getDataTable()
```

Parameter

Keine.

Ergebnisse

Keine.

Beispiel

```
DataTable dataTable = message.getDataTable();
```

DataTable

`DataTable` enthält die Datensätze für Ihre Eingabe- und Ausgabedaten. Mit den Methoden, die dieser Klasse zugeordnet sind, definieren Sie die Spaltennamen für Ihre Ausgabe und fügen Datensätze zur `DataTable` hinzu.

AddColumn

Fügt eine neue Spalte zur Datentabelle (`DataTable`) hinzu.

Syntax

```
public int addColumn(String columnName)
```

Parameter

- Spaltenname

Ergebnisse

Gibt den Spaltenindex zurück.

Beispiel

```
DataTable dataTable = message.getDataTable();  
int columnIndex = dataTable.addColumn("AddressLine1");  
columnIndex = dataTable.addColumn("City");
```

GetColumnNames

Ruft alle Spaltennamen ab.

Syntax

```
public String[] getColumnNames()
```

Parameter

- Keine

Ergebnisse

Gibt das Zeichenfolgenarray von Spaltennamen zurück.

Beispiel

```
String[] columnNames = dataTable.getColumnNames();
```

GetColumnIndex

Ruft den entsprechenden Spaltenindex ab.

Syntax

```
public int getColumnIndex(String columnName)
```

Parameter

- Spaltenname

Ergebnisse

Gibt den entsprechenden Spaltenindex zurück.

Beispiel

```
int columnIndex = dataTable.getColumnIndex("City");
```

GetColumnCount

Ruft die Anzahl der Spalten in der Datentabelle (DataTable) ab.

Syntax

```
public int getColumnCount()
```

Parameter

- Keine

Ergebnisse

Gibt die Anzahl der Spalten zurück.

Beispiel

```
int columnCount = dataTable.getColumnCount();
```

Clear

Löscht die Daten in der Datentabelle (DataTable).

Syntax

```
public void clear()
```

Parameter

- Keine

Ergebnisse

Keine.

Beispiel

```
dataTable.clear();
```

Iterator

Ein Iterator, der alle Datenzeilen (DataRows) in der Datentabelle (DataTable) enthält.

Syntax

```
public Iterator iterator()
```

Parameter

- Keine

Ergebnisse

Gibt einen Iterator zurück, der alle Datenzeilen (DataRows) in der Datentabelle (DataTable) enthält.

Beispiel

```
Iterator iter = dataTable.iterator();  
while (iter.hasNext())  
{  
    DataRow row = (DataRow)iter.next();  
}
```

AddRow

Fügt eine Zeile zur DataTable hinzu.

Syntax

```
public void addRow(DataRow row)
```

Parameter

- Zeile – DataRow, die zur DataTable hinzugefügt werden soll

Ergebnisse

Keine.

Beispiel

```
DataTable dataTable = message.getDataTable();  
  
DataRow row = dataTable.newRow();  
row.set("AddressLine1", "4203 Greenridge");  
  
dataTable.addRow(row);
```

NewRow

Erstellt eine neue DataRow in der DataTable.

Syntax

```
public DataRow newRow()
```

Parameter

- Keine

Ergebnisse

Gibt die neu erstellte Datenzeile (DataRow) zurück.

Beispiel

```
DataRow row = dataTable.newRow();  
row.set("AddressLine1", "4203 Greenridge");  
  
dataTable.addRow(row);
```

GetRowCount

Ruft die Anzahl der Datenzeilen (DataRow) in der Datentabelle (DataTable) ab.

Syntax

```
public int getRowCount()
```

Parameter

- Keine

Ergebnisse

Gibt die Anzahl der Datenzeilen (DataRow) in der Datentabelle (DataTable) zurück.

Beispiel

```
int rowCount = dataTable.getRowCount();
```

Merge

Führt die vorhandene Datentabelle (DataTable) und die aktuelle Datentabelle zusammen.

Syntax

```
public void merge(DataTable other)
```

Parameter

- Die andere Datentabelle (DataTable), die mit der aktuellen Datentabelle zusammengeführt werden soll.

Ergebnisse

Keine.

Beispiel

```
DataTable otherDataTable = new DataTable();  
dataTable.merge(otherDataTable);
```

DataRow

`DataRow` enthält die einzelnen Datensätze für Ihre Eingabe- und Ausgabedaten. Mit den Methoden, die dieser Klasse zugeordnet sind, definieren Sie die Spaltennamen für Ihre Ausgabe und fügen Datensätze zur `DataTable` (Datentabelle) hinzu.

GetColumnNames

Ruft alle Spaltennamen ab.

Syntax

```
public String[] getColumnNames()
```

Parameter

- Keine

Ergebnisse

Gibt das Zeichenfolgenarray von Spaltennamen zurück.

Beispiel

```
String[] columnNames = dataRow.getColumnNames();
```

GetColumnIndex

Ruft den entsprechenden Spaltenindex ab.

Syntax

```
public int getColumnIndex(String columnName)
```

Parameter

- Name – Spaltenname

Ergebnisse

Gibt den entsprechenden Spaltenindex zurück.

Beispiel

```
int columnIndex = dataRow.getColumnIndex("City");
```

Get

Ruft den Wert aus dem Feld-Array nach dem Spaltenindex in dieser Datenzeile (DataRow) ab.

Syntax

```
public String get(int index)
```

Parameter

- Der Index, dem der angegebene Wert zugeordnet werden soll.

Ergebnisse

Gibt den Wert für den Spaltenindex in dieser Datenzeile (DataRow) zurück.

Beispiel

```
String value = dataRow.get(1);
```

Get

Ruft den Wert aus dem Feld-Array nach dem Spaltennamen in dieser Datenzeile (DataRow) ab.

Syntax

```
public String get(String columnName)
```

Parameter

- Name: Dies ist der Name, dem der angegebene Wert zugeordnet werden soll.

Ergebnisse

Gibt den Wert für den Spaltennamen in dieser Datenzeile (DataRow) zurück, und gibt eine leere Zeichenfolge zurück, wenn der Spaltenname nicht vorhanden ist.

Beispiel

```
String value = dataRow.get("City");
```

Merge

Führt die angegebene Datenzeile (DataRow) mit der aktuellen Datenzeile zusammen.

Syntax

```
public void merge(DataRow other)
```

Parameter

- Dies ist die andere Datenzeile (DataRow), die mit der aktuellen Datenzeile zusammengeführt werden soll.

Ergebnisse

Keine.

Beispiel

```
DataRow otherDataRow = new DataRow();  
dataRow.merge(otherDataRow);
```

Set

Legt den Wert für die entsprechende Spalte der Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.

Syntax

```
public void set(int Index, String value)
```

Parameter

- Der Spaltenindex, dem der angegebene Wert zugeordnet werden soll
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ergebnisse

Keine.

Ausnahmen

- `IndexOutOfBoundsException` – der Spaltenindex ist ungültig

Beispiel

```
DataRow row = dataTable.newRow();
row.set(0, "4203 Greenridge");
row.set(1, "Austin");
row.set(2, "Texas");
dataTable.addRow(row);
```

AddChild

Fügt der benannten übergeordneten/untergeordneten Beziehung eine Datenzeile (`DataRow`) hinzu. Wenn die benannte Beziehung vorhanden ist, wird die angegebene Datenzeile (`DataRow`) der vorhandenen Datenzeilensammlung angefügt. Anderenfalls wird eine neue Sammlung mit der angegebenen Datenzeile (`DataRow`) als einziges Element erstellt.

Syntax

```
public void addChild(String childName, DataRow childDataRow)
```

Parameter

- Name: Der Name der übergeordneten/untergeordneten Beziehung (z. B. „Daten zum Überschwemmungsgebiet“, „Verweise“, „Verwendet von“ usw.)
- Wert – Die `DataRow`, die zur Beziehung hinzugefügt werden soll.

Ergebnisse

Keine.

Beispiel

```
DataRow childDataRow = new DataRow();
childDataRow.set("Address", "100 Congress");
...
DataRow dataRow = new DataRow();
...
dataRow.addChild("child1", childDataRow);
```


GetChildren

Ruft die untergeordneten Zeilen einer benannten Beziehung ab.

Syntax

```
public List getChildren(String childName)
```

Parameter

- Der Name der übergeordneten/untergeordneten Beziehung (z. B. „Überschwemmungsgebietsdaten“, „Referenzen“, „Verwendet von“ usw.)

Ergebnisse

Gibt die untergeordneten Zeilen aus der benannten Beziehung zurück.

Beispiel

```
List childRows = row.getChildren("child1");
```

ListChildNames

Ruft alle Namen der benannten übergeordneten/untergeordneten Beziehungen.

Syntax

```
public Set listChildNames()
```

Parameter

Keine.

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

```
Set childNames = row.listChildNames();
```

SetChildren

Legt die Zeilen einer angegebenen benannten übergeordneten/untergeordneten Beziehung fest. Wenn zuvor Zeilen mit diesem Namen vorhanden waren, werden diese an den Aufrufer zurückgegeben.

Syntax

```
public List setChildren(String childName, List DataRows)
```

Parameter

Keine.

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

```
List rows = dataRow.getChildren("child1");  
parentRow.setChildren("child2", rows);
```

Set

Legt den Wert für die entsprechende Spalte der Datenzeile (DataRow) fest. Wenn der Wert für den Namen vorhanden ist, wird der alte Wert ersetzt.

Syntax

```
public void set(int Index, String value)
```

Parameter

- Der Spaltenindex, dem der angegebene Wert zugeordnet werden soll
- Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Ergebnisse

Keine.

Ausnahmen

- `IndexOutOfBoundsException` – der Spaltenindex ist ungültig

Beispiel

```
DataRow row = dataTable.newRow();  
row.set(0, "4203 Greenridge");  
row.set(1, "Austin");  
row.set(2, "Texas");  
dataTable.addRow(row);
```

6 - Die .NET-API

In this section

Einführung	189
Server	193
Service	196
Message	197
EnhancedDataTable	203

Einführung

.NET ist eine Betriebssystemplattform von Microsoft®, die Anwendungen und eine Suite aus Tools und Diensten enthält. Diese optimieren die Entwicklung von Webdiensten und Anwendungen.

Das .NET Framework verwendet Komponenten mit den Bezeichnungen Common Language Runtime (CLR), Framework Class Library (FCL) und ASP.NET. CLR entspricht der Java Virtual Machine insofern, dass sie Codes verwaltet und sie in der nativen Sprache der Maschine ausführt, auf der sie ausgeführt wird. Bei Framework Class Library handelt es sich um eine umfangreiche Bibliothek wiederverwendbarer Objekttypen, die unzählige Programmfunktionen abdecken. ASP.NET ist eine serverseitige Technologie, mit der Webseiten und Dienste viel schneller als herkömmliche ASP-Seiten geladen werden können. Zusammen erleichtern und optimieren diese drei Komponenten von .NET Framework die Anwendungs- und Webentwicklung und bieten eine leichtere Integration in vorhandene Umgebungen. Clients und Server auf unterschiedlichen Plattformen, die in unterschiedlichen Programmiersprachen geschriebene Dienste ausführen, können schnell und einfach miteinander kommunizieren.

Weitere Informationen zur .NET-Technologie finden Sie unter msdn.microsoft.com/netframework.

Konstanten

Die .NET API verwendet zwei Konstantengruppen. Der erste Satz ist für die `Server`-Komponente, die in der unteren Tabelle beschrieben wird.

Tabelle 24: Konstanten für die `Server`-Komponente

Name der Konstante	Beschreibung	Beispiel
<code>Server.HOST</code>	Zeichenfolge für den Hostnamen des Servers. Der Standardname lautet „localhost“.	65.89.200.89
<code>Server.PORT</code>	Zeichenfolge für den Serverport. Der Standardwert ist „8080“.	10119
<code>Server.ACCOUNT_ID</code>	Zeichenfolge für die Serverkonto-ID. Der Standardwert ist NULL.	user1

Name der Konstante	Beschreibung	Beispiel
Server.ACCOUNT_PASSWORD	Zeichenfolge für das Kennwort des Serverkontos. Der Standardwert ist NULL.	user1
Server.CONNECTION_TIMEOUT	Zeichenfolge für das Timeout der Serververbindung in Millisekunden. Der Standardwert ist „10000“.	50000
Server.CONNECTION_TYPE	Zeichenfolge für den Serververbindungstyp. Derzeit werden nur HTTP, HTTPS und SOCKET unterstützt. Der Standardwert lautet „HTTP“.	HTTP(S)
Server.PROXY_HOST	Zeichenfolge für den Hostnamen des Proxy-Servers. Der Standardwert ist NULL.	192.168.1.77
Server.PROXY_PORT	Zeichenfolge für den Proxy-Serverport. Der Standardwert ist NULL.	8080
Server.PROXY_USER	Zeichenfolge für die Konto-ID des Proxy-Servers. Der Standardwert ist NULL.	user1
Server.PROXY_PASSWORD	Zeichenfolge für das Kontokennwort des Proxy-Servers. Der Standardwert ist NULL.	user1

Der zweite Konstantensatz ist für die `Message`-Komponente vorgesehen.

Tabelle 25: Konstanten für die `Message`-Komponente

Name der Konstante	Beschreibung	Beispiel
Message.CONTEXT_ACCOUNT_ID	Zeichenfolge für den Nachrichtentext der Konto-ID.	user1

Name der Konstante	Beschreibung	Beispiel
Message.CONTEXT_ACCOUNT_PASSWORD	Zeichenfolge für das Kontokennwort des Nachrichtenkontextes.	user1
Message.CONTEXT_SERVICE_NAME	Zeichenfolge für den Dienstnamen des Nachrichtenkontextes.	echoservice

Fehlermeldungen

Die .NET API verwendet die folgenden Fehlermeldungen:

- Fehlermeldungen für Verbindungen
 - "Connection type not supported."
 - "Client timeout"
- Fehlermeldungen für Ausnahmen beim Verpacken der Nachrichten
 - "Input Message is null."

Die anderen Fehlermeldungen werden aus der .NET Framework Class Library stammen, wenn sie nicht ordnungsgemäß verwendet werden.

Beispielanwendung

Der unten dargestellte Beispielcode zeigt anschaulich auf, wie die .NET API verwendet werden kann.

```
using System;
using System.IO;
using System.Collections;
using System.Text;
using System.Data;
using g1client;

try
{
    //Create Server
    Server server = new Server();
```

```
//Set connect property to the server
server.SetConnectionProperty(Server.HOST, "localhost");
server.SetConnectionProperty(Server.PORT, "10119");
server.SetConnectionProperty(Server.CONNECTION_TYPE, "SOCKET");
server.SetConnectionProperty(Server.ACCOUNT_ID, "guest");
server.SetConnectionProperty(Server.ACCOUNT_PASSWORD, "");

//Connect to server
server.Connect();

//Get Service From Server
Service service = server.GetService("ValidateAddress");

//Create Input Message
Message request = new Message();

//Fill dataTable in the input message
//Datatable is the .net Framework class
DataTable dataTable = request.GetDataTable();

DataColumn column1 = new DataColumn();
column1.DataType = System.Type.GetType("System.String");
column1.ColumnName = "AddressLine1";
dataTable.Columns.Add(column1);

DataColumn column2 = new DataColumn();
column2.DataType = System.Type.GetType("System.String");
column2.ColumnName = "City";
dataTable.Columns.Add(column2);

DataColumn column3 = new DataColumn();
column3.DataType = System.Type.GetType("System.String");
column3.ColumnName = "StateProvince";
dataTable.Columns.Add(column3);

DataRow newRow = dataTable.NewRow();
newRow[0]="4200 Parliament Place";
newRow[1]="Lanham";
newRow[2]="Maryland";

dataTable.Rows.Add(newRow);

//Set "option" Properties to the Input Message
request.PutOption("OutputCasing", "M");
request.PutOption("OutputRecordType", "A");

//Process Input Message, return output Message
Message reply = service.Process(request);

//Disconnect from server
server.Disconnect();
```



```

//Get the result from the response message
DataTable returnDataTable = reply.GetDataTable();

foreach(DataColumn dc in returnDataTable.Columns)
{
// more code to be added
string columnName = dc.ColumnName;
}
foreach(DataRow dr in returnDataTable.Rows)
{
for (int col = 0; col < returnDataTable.Columns.Count; col++)
{
// more code to be added
string value = (String)dr[col] ;
Console.WriteLine(value);
}
}
}
catch (Exception e)
{

//Error handling
Console.WriteLine("Error Ocurrred, " + e.ToString());
}

```

Server

Mithilfe der `Server`-Klasse können Sie eine Verbindung zum Server aufbauen, die Verbindung zum Server trennen und den Dienst vom Server abrufen.

Connect

Liest die Eigenschaften, um zu bestimmen, welche Gateway-Verbindung verwendet werden soll, und stellt eine Verbindung zum Server her.

Anmerkung: .NET verwendet das HTTP, HTTPS oder SOCKET-Protokoll zum Verbindungsaufbau mit dem Server. HTTP und HTTPS bauen logisch eine Client-Verbindung auf, stellen jedoch erst eine tatsächliche Verbindung zum Server her, wenn eine `GetService`- oder `Process`-Methode aufgerufen wird. Das SOCKET-Protokoll baut eine Verbindung zum Server auf, wenn `Connect` aufgerufen wird.

Syntax

```
public void Connect()
```

Parameter

Keine.

Ergebnisse

Keine.

Ausnahmen

- „Verbindungstyp wird nicht unterstützt.“

Beispiel

```
Server server = new Server();  
  
// set connect property to the server  
server.SetConnectionProperty(Server.HOST, "localhost");  
server.SetConnectionProperty(Server.PORT, "8080");  
// more connection properties to be set  
// Connect to server  
server.Connect();
```

Disconnect

Trennt die Verbindung zum Server.

Syntax

```
public void Disconnect()
```

Parameter

Keine.

Ergebnisse

Die Client-Verbindung zum Server wird getrennt.

Beispiel

```
//Disconnect from server  
server.Disconnect();
```

SetConnectionProperty

Legt die Konfigurationseigenschaften der Serververbindung fest, z. B. den Servernamen und die Timeout-Dauer.

Syntax

```
public void SetConnectionProperty(String name, String value)
```

Parameter

- Name: Dies ist der Name der Verbindungseigenschaft wie HOST.
- Value: Dies ist der Wert für den Namen der Verbindungseigenschaft wie „www.myhost.com“

Ergebnisse

Keine.

Beispiel

```
Server server = new Server();

server.SetConnectionProperty(Server.HOST, "localhost");
server.SetConnectionProperty(Server.PORT, "8080");

//Connect to server
server.Connect();
```

GetService

Ruft den Dienst vom Server ab.

Anmerkung: Eine Liste der Dienste, die Ihnen zur Verfügung stehen, finden Sie im Abschnitt „Komponentenreferenz“ in diesem Handbuch.

Syntax

```
public Service getService(String serviceName)
```

Parameter

- Name: Name des Dienstes

Ergebnisse

Gibt den bestimmten Dienst zurück.

Beispiel

```
Service service = server.GetService("ValidateAddress");
```

Service

Die `Service`-Klasse dient der Verarbeitung der Nachricht (d. h., dass sie die Nachricht an den Server sendet und eine Antwort vom Server erhält).

Process

Verarbeitet die Eingabenachricht und gibt die Antwortnachricht zurück.

Syntax

```
public Message Process(Message, message)
```

Parameter

- Eingabenachricht

Ergebnisse

Gibt die Antwortnachricht zurück.

Ausnahmen

`MessageProcessingException`

Beispiel

```
//Process Input Message, return output Message  
Message reply = service.Process(request);
```

Message

Die `Message`-Klasse sendet Ihre Eingabedaten und empfängt Ihre Ausgabedaten vom Dienst. Die Eigenschaften für `Message` umfassen Kontexteigenschaften wie Konto-ID, Kontokennwort, Dienstname und Dienstmethode sowie Optionseigenschaften, bei denen es sich um die dienstspezifischen Laufzeitoptionen handelt.

GetContext

Ruft den Wert nach dem Namen in den Kontexteigenschaften ab. Kontexteigenschaften enthalten die folgenden Konstanten: Konto-ID, Kontokennwort, Dienstname, Dienstschlüssel und Anforderungs-ID.

Syntax

```
public String GetContext(String name)
```

Parameter

Keine.

Ergebnisse

Gibt den mit dem Namen verknüpften Wert in den „Context“-Eigenschaften zurück. Wenn der Name nicht vorhanden ist, gibt die Methode `NULL` zurück.

Beispiel

```
String value = message.GetContext(Message.CONTEXT_ACCOUNT_ID);
```

GetContext

Ruft die Hashtabelle ab, die alle Kontexteinträge enthält. Bei Hashtabelle handelt es sich um eine .NET Framework-Klasse.

Syntax

```
public Hashtable GetContext()
```

Parameter

- Keine

Ergebnisse

Gibt die Hashtabelle zurück, die alle Kontexteinträge enthält.

Beispiel

```
Hashtable context = message.GetContext();
```

PutContext

Legt den Wert für den angegebenen Namen in den Kontexteigenschaften fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt. Kontexteigenschaften enthalten die folgenden Konstanten: Konto-ID, Kontokennwort, Dienstname, Dienstschlüssel und Anforderungs-ID.

Syntax

```
public void PutContext(String name, String value)
```

Parameter

- Name: Dies ist der Name, dem der angegebene Wert zugeordnet werden soll.
- Value: Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Beispiel

```
message.PutContext(Message.CONTEXT_ACCOUNT_ID, "user1");
```

PutContext

Fügt den aktuellen Kontexteigenschaften die neuen Kontexteigenschaften hinzu.

Syntax

```
public void PutContext(Hashtable context)
```

Parameter

- Die neue Kontext-Hashtabelle, die der aktuellen Kontext-Hashtabelle hinzugefügt werden soll.

Ergebnisse

Keine.

Beispiel

```
//Hashtable is the .NET Framework class
Hashtable context = new Hashtable();
//more code
message.PutContext(context);
```

SetContexts

Überschreibt die aktuellen Kontexteigenschaften mit den neuen Kontexteigenschaften.

Syntax

```
public void SetContexts(Hashtable context)
```

Parameter

- Context: Die neue Kontext-Hashtabelle, die die aktuelle Kontext-Hashtabelle ersetzen wird.

Ergebnisse

Keine.

Beispiel

```
//Hashtable is the .NET Framework class
Hashtable context = new Hashtable();
//more code
message.SetContexts(context);
```

GetOption

Ruft den Wert nach dem Namen in den Optionseigenschaften ab. Bei Optionseigenschaften handelt es sich um die dienstspezifischen Laufzeitoptionen.

Syntax

```
public String GetOption(String name)
```

Parameter

- Name: Der Name, dessen zugeordneter Wert zurückgegeben werden soll.

Ergebnisse

Gibt den Wert für den Namen in den „Option“-Eigenschaften in der Nachricht oder NULL zurück, wenn der Name nicht vorhanden ist.

Beispiel

```
String value = message.GetOption("OutputCasing");
```

GetOptions

Ruft die Hashtabelle auf, die alle Optionseinträge enthält. Bei Hashtable handelt es sich um eine .NET Framework-Klasse.

Syntax

```
public Hashtable GetOptions();
```

Parameter

- Keine

Ergebnisse

Gibt die Hashtabelle zurück, die alle Optionseinträge enthält.

Beispiel

```
Hashtable options = message.GetOptions();
```

PutOption

Legt den Wert für den angegebenen Namen in den Optionseigenschaften fest. Wenn für die durch den Namen identifizierte Entität bereits ein Wert vorhanden ist, wird dieser ersetzt. Bei Optionseigenschaften handelt es sich um die dienstspezifischen Laufzeitoptionen.

Syntax

```
public void PutOption(String name, String value)
```


Parameter

- Name: Dies ist der Name, dem der angegebene Wert zugeordnet werden soll.
- Value: Der Wert, der dem angegebenen Namen zugeordnet werden soll.

Beispiel

```
message.PutOption("OutputCasing", "M");
```

PutOptions

Fügt die neuen Optionseigenschaften zu den aktuellen Optionseigenschaften hinzu.

Syntax

```
public void PutOptions(Hashtable options)
```

Parameter

- Option: Die neue Option-Hashtabelle, der der aktuellen Option-Hashtabelle hinzugefügt werden soll.

Ergebnisse

Keine.

Beispiel

```
//Hashtable is the .NET Framework class  
Hashtable options = new Hashtable();  
// more code  
message.PutOptions(options);
```

SetOptions

Überschreibt die aktuellen Optionseigenschaften mit den neuen Optionseigenschaften.

Syntax

```
public void SetOptions(Hashtable options)
```

Parameter

- Optionen: Die neue Options-Hashtabelle, die die aktuelle Options-Hashtabelle ersetzen soll.

Ergebnisse

Keine.

Beispiel

```
//Hashtable is the .NET Framework class  
Hashtable options = new Hashtable();  
//more code  
message.SetOptions(options);
```

GetError

Ruft die Fehlermeldung aus der Nachricht ab.

Syntax

```
public String GetError()
```

Parameter

- Keine

Ergebnisse

Gibt die Fehlermeldung in der Nachricht zurück.

Beispiel

```
String error = message.GetError();
```

GetDataTable

Ruft die Datentabelle in dieser Nachricht ab. Bei DataTable handelt es sich um eine .NET Framework-Klasse.

Syntax

```
public DataTable GetDataTable()
```

Parameter

Keine.

Ergebnisse

Keine.

Beispiel

```
//DataTable is the .net Framework class
DataTable dataTable = message.GetDataTable();

DataColumn column1 = new DataColumn();
column1.DataType = System.Type.GetType("System.String");
column1.ColumnName = "AddressLine1";
dataTable.Columns.Add(column1);

DataColumn column2 = new DataColumn();
column2.DataType = System.Type.GetType("System.String");
column2.ColumnName = "City";
dataTable.Columns.Add(column2);

DataRow newRow = dataTable.NewRow();
newRow[0]="4203 Greenridge";
newRow[1]="Austin";

dataTable.Rows.Add(newRow);
```

EnhancedDataTable

EnhancedDataTable ist eine Klasse, die die DataTable der .NET-Klasse erweitert.

AddChild

Fügt der benannten übergeordneten/untergeordneten Beziehung eine Datenzeile (DataRow) hinzu. Wenn die benannte Beziehung vorhanden ist, wird die angegebene Datenzeile (DataRow) der vorhandenen Datenzeilensammlung angefügt. Anderenfalls wird eine neue Sammlung mit der angegebenen Datenzeile (DataRow) als einziges Element erstellt.

Syntax

```
public void AddChild(DataRow parentRow, string name, DataRow newChild)
```

Parameter

- Name: Der Name der übergeordneten/untergeordneten Beziehung (z. B. „Daten zum Überschwemmungsgebiet“, „Verweise“, „Verwendet von“ usw.)
- DataRow: Die Datenzeile, die der Beziehung hinzugefügt werden soll.

Ergebnisse

Keine.

Beispiel

```
EnhancedDataTable dataTable = new EnhancedDataTable();

dataTable.Columns.Add(new DataColumn("AddressLine1",
System.Type.GetType("System.String")));
dataTable.Columns.Add(new DataColumn("City",
System.Type.GetType("System.String")));
dataTable.Columns.Add(new DataColumn("StateProvince",
System.Type.GetType("System.String")));
dataTable.Columns.Add(new DataColumn("PostalCode",
System.Type.GetType("System.String")));

DataRow row = dataTable.NewRow();

row[0] = "510 S Coit St";
row[1] = "Florence";
row[2] = "SC";
row[3] = "29501-5221";

EnhancedDataTable childDataTable = new EnhancedDataTable();

childDataTable.Columns.Add(new DataColumn("AddressLine2",
System.Type.GetType("System.String")));
childDataTable.Columns.Add(new DataColumn("City",
System.Type.GetType("System.String")));
childDataTable.Columns.Add(new DataColumn("StateProvince",
System.Type.GetType("System.String")));
childDataTable.Columns.Add(new DataColumn("PostalCode",
System.Type.GetType("System.String")));

DataRow childRow = childDataTable.NewRow();

childRow[0] = "241 Ne C St";
childRow[1] = "Willamina";
childRow[2] = "OR";
childRow[3] = "97396-2714";

dataTable.AddChild(row, "Child1", childRow);
dataTable.Rows.Add(row);
```

GetChildren

Ruft die untergeordneten Zeilen einer benannten Beziehung ab.

Syntax

```
public EnhancedDataTable GetChildren(DataRow parentRow, string name)
```

Parameter

- ParentRow: Die übergeordnete Zeile
- Name: Der Name der Beziehung der übergeordneten/untergeordneten Elemente, z. B. „Daten zum Überschwemmungsgebiet“, „Verweise“, „Verwendet von“ usw.

Ergebnisse

Gibt die untergeordneten Zeilen aus der benannten Beziehung zurück.

Beispiel

```
EnhancedDataTable childRows = dataTable.GetChildren(parentRow, "child1");
```

ListChildNames

Ruft alle Namen der benannten übergeordneten/untergeordneten Beziehungen.

Syntax

```
public string[] ListChildrenNames(DataRow parentRow)
```

Parameter

Keine.

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

```
string[] childNames = dataTable.ListChildrenNames( parentRow);
```

SetChildren

Legt die Zeilen einer angegebenen benannten übergeordneten/untergeordneten Beziehung fest. Wenn zuvor Zeilen mit diesem Namen vorhanden waren, werden diese an den Aufrufer zurückgegeben.

Syntax

```
public void SetChildren(DataRow parentRow, string name, EnhancedDataTable newTable)
```

Ergebnisse

Gibt die Namen der übergeordneten/untergeordneten Beziehungen zurück.

Beispiel

```
EnhancedDataTable childRows = dataTable1.GetChildren(parentRow, "child1");  
dataTable2.SetChildren(otherParentRow, "child1", childRows);
```

7 -

ManagementAPI-Methoden (veraltet)

In this section

Einführung	208
GetLicenseInfo	208
GetVersionInfo	209

Einführung

Wichtig: Der ManagementAPI-Webservice ist veraltet und wird in einer zukünftigen Version entfernt werden. Verwenden Sie die Administrationsumgebung, um Lizenz- und Versionsinformationen zu Ihrem System abzurufen. Weitere Informationen zur Administrationsumgebung finden Sie im *Administratorhandbuch*.

Es sind zwei API-Methoden, die zur Verwaltung dienen, öffentlich über den ManagementAPI-Webservice verfügbar: `getLicenseInfo` und `getVersionInfo`. Die WSDL-URL für den ManagementAPI-Webservice lautet:

```
http://SpectrumServer:8080/managers/ManagementAPIService?wsdl
```

Dabei ist *SpectrumServer* der Hostname oder die IP-Adresse Ihres Spectrum™ Technology Platform-Servers.

GetLicenseInfo

Wichtig: Der ManagementAPI-Webservice ist veraltet und wird in einer zukünftigen Version entfernt werden. Verwenden Sie die Administrationsumgebung, um Lizenz- und Versionsinformationen zu Ihrem System abzurufen. Weitere Informationen zur Administrationsumgebung finden Sie im *Administratorhandbuch*.

Die Methode „GetLicenseInfo“ gibt ein Lizenzobjekt zurück. Das Lizenzobjekt enthält Eigenschaften für den Rechnertyp, das Betriebssystem, den Hostnamen und das CPU-Limit. Es enthält außerdem eine Auswahl von Feature-Objekten und Einschränkungobjekten. Diese Auswahl kann verarbeitet werden, um bestimmte Informationen über Features und Einschränkungen zu bestimmen. Ein Feature besitzt eine ID, einen Namen und eine aktivierte Kennzeichnung. Eine Einschränkung besitzt eine ID, ein Limit und ein Startdatum.

Webservice

ManagementAPIService

Parameter

Keine.

Ergebnis

Gibt das Lizenzobjekt zurück.

Beispiel

```

License
  string machineType
  string osType
  string hostName
  string CPULimit
  Feature[] features
  Restriction[] restrictions

Feature
  string ID;
  string name;
  Restriction[] restrictions

Restriction
  string ID
  long limit
  datetime startDate
  Feature[] features

ExpirationRestriction extends Restriction

UsageRestriction extends Restriction
  long usages

```

GetVersionInfo

Wichtig: Der ManagementAPI-Webservice ist veraltet und wird in einer zukünftigen Version entfernt werden. Verwenden Sie die Administrationsumgebung, um Lizenz- und Versionsinformationen zu Ihrem System abzurufen. Weitere Informationen zur Administrationsumgebung finden Sie im *Administratorhandbuch*.

Die Methode „GetVersionInfo“ gibt eine Auswahl von VersionInfo-Objekten zurück. Ein VersionInfo-Objekt hat einen Namen und eine Versionsnummer und enthält eine Liste von VersionAttribut-Objekten. VersionAttribute-Objekte sind einfache Klassen mit einer Bezeichnung und einem Wert. GetVersionInfo-Attribute sind produktspezifisch, da die Informationen von dem Produkt selbst gesammelt und zurückgegeben werden. Die gleichen Informationen werden auch auf dem Knoten „Versionsinformationen“ in der Management Console angezeigt.

Anmerkung: Sie müssen die Methode „GetVersionInfo“ einmal ausführen, prüfen, welche Werte zurückgegeben werden, und die Informationen anschließend parsen, um bestimmte Informationseinheiten abzurufen.

Webservice

ManagementAPIService

Parameter

Keine.

Ergebnis

Gibt VersionInfo-Objekte zurück.

Beispiel

```
VersionInfo
  string name
  string version
  VersionAttribute[] attributes

VersionAttribute
  string label
  string value
```

8 - Moduldienste

In this section

Address Now-Modul	212
Enterprise Geocoding-Modul	267
GeoConfidence-Modul	354
Universal Addressing-Modul	357
Universal Name-Modul	564

Address Now-Modul

Address Now-Modul

Das Address Now-Modul ist ein Tool zur Standardisierung und Überprüfung von Adressen. Es bietet eine umfassende Abdeckung für Adressen außerhalb von den USA und Kanada. Address Now ist eines von zwei Modulen für die Standardisierung und Überprüfung von Adressen, das für Spectrum™ Technology Platform zur Verfügung steht. Das andere Modul ist das Universal Addressing-Modul. Das Address Now-Modul bietet gegenüber dem Universal Addressing-Modul für Adressen außerhalb von den USA und Kanada die folgenden Vorteile:

- **Bessere Daten** – Die vom Address Now-Modul verwendete Datenbank ist in vielen Ländern aktueller und vollständiger als die vom Universal Addressing-Modul verwendete Datenbank. Grund dafür ist, dass sich das Universal Addressing-Modul im Hinblick auf die internationalen Daten auf Daten der Universal Postal Union (UPU), einer Einrichtung der Vereinten Nationen, stützt. Während die Daten in umfangreichem Rahmen abgedeckt werden, werden die Updates und Details zur Adressenebene nicht proaktiv von der UPU verwaltet. Address Now hingegen stützt sich auf Daten, die direkt von den Postbehörden stammen (in den meisten Ländern) sowie auf Daten von anderen Drittanbietern. Das heißt, die Daten sind im Hinblick auf Poständerungen aktueller und detaillierter.
- **Drilldown-Feature** – Das Address Now-Modul bietet auch Drilldown-Funktionen für Daten aus einem beliebigen Land, mit denen Benutzer Adresseninformationen schnell eingeben können, ohne sich über die Struktur oder über mögliche Fehler bei der Dateneingabe Gedanken machen zu müssen.
- **Doppelbyte-Unterstützung** – Das Address Now-Modul ist Unicode-aktiviert und erkennt Kanji-Zeichen sowie andere Doppelbytezeichen.

Address Now-Komponenten

Address Now umfasst folgende Komponenten. Diese Komponenten können mit US-Adressen, kanadischen Adressen und internationalen Adressen arbeiten.

- **BuildGlobalAddresses** – Ermöglicht es Ihnen, durch die Suche nach einzelnen Adressenelementen eine Adresse interaktiv zu erstellen.
- **GetGlobalCandidateAddresses** – Gibt eine Liste mit Adressen zurück, die bei einer angegebenen Adresse als Übereinstimmungen gelten.
- **ValidateGlobalAddress** – Standardisiert Adressen durch die Verwendung internationaler postalischer Daten. „ValidateGlobalAddress“ kann auch Adressen in den USA und Kanada überprüfen. Die Stärke liegt jedoch in der Überprüfung von Adressen in anderen Ländern. Wenn

Sie über eine große Menge von Adresdaten verfügen, die nicht aus den USA und Kanada stammen, sollten Sie in Erwägung ziehen, die Komponente „ValidateGlobalAddress“ zu verwenden.

In Fällen, in denen „ValidateGlobalAddress“ mehrere Adressübereinstimmungen für eine bestimmte eingegebene Adresse zurückgibt, können Sie den Adressstapel mit „GetGlobalCandidateAddresses“ zurückgeben. „GetGlobalCandidateAddresses“ wurde für die Rückgabe zusätzlicher Informationen aus den postalischen Datenbanken entworfen, um Sie bei der Bestimmung zu unterstützen, welche der zurückgegebenen Adressen die größte Übereinstimmung aufweisen.

Address Now-Datenbank

Die Address Now-Datenbank enthält postalische Daten von allen unterstützten Ländern. Sie können die gesamte Datenbank oder nur die Daten von bestimmten Ländern installieren. Die Datenbank wird auf dem Server installiert. Diese Datenbank ist über das Abonnement von Pitney Bowes verfügbar und wird monatlich aktualisiert.

BuildGlobalAddress

Mithilfe von „BuildGlobalAddress“ können Sie eine gültige Adresse erstellen, die mit einem einzelnen Adresselement oder wenigen Adresselementen beginnt. BuildGlobalAddress ist Teil des „Address Now“-Moduls.

Verwenden von BuildGlobalAddress

Das Erstellen einer Adresse ist ein interaktiver Prozess, bei dem Sie in den einzelnen Schritten der Adressenerstellung Adresselemente auswählen müssen. Das heißt, für das Erstellen einer Adresse muss BuildGlobalAddress nicht nur einmal, sondern mehrmals aufgerufen werden. Führen Sie zu Beginn in BuildGlobalAddress einen Aufruf zum Initialisieren durch. Bei diesem Aufruf wird eine Sitzungs-ID zurückgegeben. Diese Sitzungs-ID verwenden Sie dann in nachfolgenden Aufrufen. Bei jedem Aufruf zeigt BuildGlobalAddress eine Liste mit alternativen Werten für ein Adresselement an. Sie wählen den gewünschten Wert aus, fahren mit dem nächsten Adresselement fort und machen dies so lange, bis die Adresse vollständig ist. Von einigen Ausnahmen abgesehen müssen Sie für jedes Adresselement einen separaten Aufruf durchführen.

Der gesamte Prozess läuft wie folgt ab:

- Zunächst führen Sie einen Aufruf zum Initialisieren durch, um eine Sitzung zu öffnen und eine vom System zugewiesene Sitzungs-ID zu empfangen.
- Führen Sie einen Aufruf zum Suchen durch, um mögliche Werte für ein bestimmtes Adresselement zu finden.
- Wenn Sie den gewünschten Wert ausgewählt haben, führen Sie einen Aufruf zum Übergeben durch, um den Wert für das angegebene Adresselement anzugeben.
- Fahren Sie so lange mit dem Durchführen von Aufrufen zum Suchen/Übergeben fort, bis alle Adresselemente übergeben wurden.
- Führen Sie zum Schluss einen Aufruf zum Schließen durch, um die Sitzung zu beenden.

Verwenden Sie die Registerkarte „Vorschau“ der Management Console, um die folgende Prozedur zu durchlaufen und sich mit der Funktionsweise des Prozesses vertraut zu machen.

1. Öffnen Sie die Management Console.
2. Wählen Sie auf der Registerkarte „Dienste“ die Option **Address Now** aus.
3. Wählen Sie auf der linken Seite des Bereichs die Option **Build Global Address** aus der Liste der Dienste aus.
4. Geben Sie auf der Registerkarte **Optionen** Ihre gewünschte Option an. Weitere Informationen zu den Optionen finden Sie unter **Optionen** auf Seite 219.
5. Klicken Sie auf die Registerkarte **Vorschau**.
6. Geben Sie im Feld **Aktion** `init` ein.
7. Geben Sie im Feld **Land** das Land der zu erstellenden Adresse ein.
8. Klicken Sie auf **Vorschau ausführen**.
9. Suchen Sie in der Vorschauausgabe das Feld **SessionID**, markieren Sie den Wert, klicken Sie mit der rechten Maustaste darauf und wählen Sie `Kopieren` aus dem Kontextmenü aus.
10. Markieren Sie in der Vorschau eingabe das Feld **SessionID**, klicken Sie mit der rechten Maustaste darauf und wählen Sie `Einfügen` aus.
11. Geben Sie die folgenden Werte in die Eingabefelder ein:
 - Aktion – Geben Sie die Suche ein.
 - Land – Nehmen Sie keine Änderungen an diesem Feld vor.
 - FieldIndex – Geben Sie den Indexwert des ersten Feldes ein, nach dem Sie suchen möchten. Wenn Sie beispielsweise wissen, dass Sie nach einer Adresse in Chicago suchen möchten, würden Sie „1“ eingeben, da Fieldindex 1 bei Adressen in den USA dem Feld „Ort“ entspricht.
 - SearchValue – Geben Sie den Wert ein, nach dem Sie suchen möchten. Wenn Sie beispielsweise eine Adresse in Chicago erstellen möchten, würden Sie „chicago“ eingeben.
 - SessionId – Behalten Sie den Wert bei.

Anmerkung: Die Werte in den anderen Eingabefeldern werden ignoriert.

12. Klicken Sie erneut auf **Vorschau ausführen**.
13. Die Ergebnisse der Suche werden in bis zu zwei Ausgabefeldern platziert: **Alternatives.InContext** und **AlternativesOutContext**. Eine Erläuterung des Unterschieds zwischen „In Kontext“-Ergebnissen und „Nicht in Kontext“-Ergebnissen finden Sie unter **Was ist Kontext?** auf Seite 226.
14. Geben Sie die folgenden Werte in die Eingabefelder ein, wenn Sie den gewünschten Wert gefunden haben:
 - **Aktion** – Geben Sie `commit` ein.
 - **AlternativeIndex** – Geben Sie die Indexnummer für die von Ihnen ausgewählte Alternative ein. Indexwerte beginnen bei 0, nicht bei 1. Wenn Sie beispielsweise nach Chicago suchen, würden die von BuildGlobalAddress zurückgegebenen Alternativen wie folgt indiziert. Wenn Sie den Wert „CHICAGO“ übergeben möchten, geben Sie im Feld „AlternativeIndex“ den Wert „0“ ein.

- 0 – CHICAGO
 - 1 – CHICAGO HTS
 - 2 – CHICAGO PARK
 - 3 – CHICAGO RIDGE
 - 4 – EAST CHICAGO
 - 5 – NORTH CHICAGO
 - 6 – WEST CHICAGO
- **AlternativeContext** – Geben Sie `in` oder `out` ein, um anzugeben, ob der von Ihnen im Feld **AlternativeIndex** angegebene Indexwert für die Liste der Alternativen im Feld **Alternatives.InContext** oder im Feld **Alternatives.OutContext** bestimmt ist.
 - **SessionId** – Behalten Sie diesen Wert bei.

Anmerkung: Die Werte in den anderen Eingabefeldern werden ignoriert.

15. Klicken Sie erneut auf **Vorschau ausführen**. Der von Ihnen angegebene Wert befindet sich jetzt im Feld „Feld.n.Wert“ des entsprechenden Adressenelements.
16. Wiederholen Sie die Schritte zum Suchen und Übergeben so oft wie nötig, bis die Adresse erstellt wurde.
17. Schließen Sie die Sitzung, indem Sie folgende Werte in die Eingabefelder eingeben:
 - **Aktion** – Geben Sie `close` ein.
 - **SessionId** – Behalten Sie diesen Wert bei.

Anmerkung: Die Werte in den anderen Eingabefeldern werden ignoriert.

Eingabe

Tabelle 26: Eingaben für BuildGlobalAddress

Spaltenname	Format	Beschreibung
Parameter		
Action	Zeichenfolge	Gibt die auszuführende Aktion an. Zur Auswahl stehen: <ul style="list-style-type: none"> init Initialisierung. Durch diese Aktion wird eine Sitzung geöffnet und eine Sitzungs-ID zurückgegeben, die für alle anderen Aktionen erforderlich ist. Für die INIT-Aktion ist das Eingabefeld „Land“ erforderlich. search Sucht nach Werten für ein bestimmtes Adresselement und gibt eine Liste alternativer Werte zurück, aus der Sie wählen können. Für die SEARCH-Aktion sind die folgenden Eingabefelder erforderlich. <ul style="list-style-type: none"> • FieldIndex • SearchValue • SessionId commit Weist dem Feld einen der Werte zu, die von der SEARCH-Aktion zurückgegeben werden. Für die COMMIT-Aktion sind die folgenden Eingabefelder erforderlich. <ul style="list-style-type: none"> • AlternativeIndex • AlternativeContext • SessionId clear Hebt die Übermittlung des im Feld „FieldIndex“ angegebenen Feldes auf. Für die CLEAR-Aktion sind die folgenden Eingabefelder erforderlich. <ul style="list-style-type: none"> • FieldIndex • SessionID close Beendet eine Sitzung. Für die CLOSE-Aktion ist das Eingabefeld „SessionId“ erforderlich.

Spaltenname	Format	Beschreibung
Parameter		
AlternativeContext	Zeichenfolge	<p>Gibt bei der COMMIT-Aktion an, ob Sie einen Wert aus dem Feld Alternatives.InContext oder Alternatives.OutContext auswählen. Dieses Feld wird für andere Aktionen ignoriert. Zur Auswahl stehen:</p> <p>in Sie übermitteln einen Wert aus dem Feld Alternatives.InContext. Das bedeutet, dass der im Eingabefeld AlternativeIndex angegebene Wert einem Wert im Ausgabefeld Alternatives.InContext entspricht.</p> <p>out Sie übermitteln einen Wert aus dem Feld Alternatives.OutContext. Das bedeutet, dass der im Eingabefeld AlternativeIndex angegebene Wert einem Wert im Ausgabefeld Alternatives.OutContext entspricht.</p>
AlternativeIndex	Zeichenfolge [79]	<p>Gibt bei der COMMIT-Aktion den Wert an, den Sie in der Adresse, die Sie erstellen, verwenden möchten. Wenn Sie beispielsweise nach einem Ort gesucht haben und BuildGlobalAddress eine Liste von drei Orten zurückgibt, würden Sie den gewünschten Ort angeben, indem Sie den Indexwert für Ihre Auswahl angeben. Indexwerte für die von BuildGlobalAddress vorgelegten Alternativen sind nullbasiert, was bedeutet, dass die erste Alternative einen Index von 0 aufweist, die zweite Alternative einen Wert von 1 usw.</p> <p>Mit Ausnahme der COMMIT-Aktion wird dieses Eingabefeld für Aktionen ignoriert.</p>
Country	Zeichenfolge [79]	<p>Gibt für die INIT-Aktion das Land an, in dem Sie eine Adresse erstellen möchten. Geben Sie das Land im Format an, das Sie als Eingabelandesformat ausgewählt haben (englischer Name, zweistelliger ISO 3116-1 ALPHA-2-Code oder dreistelliger ISO 3116-1 ALPHA-3-Code). Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.</p> <p>Mit Ausnahme der INIT-Aktion wird dieses Eingabefeld für Aktionen ignoriert.</p>

Spaltenname	Format	Beschreibung
Parameter		
FieldIndex	Zeichenfolge [79]	<p>Gibt bei der SEARCH-Aktion das Adresselement an, in dem Sie suchen möchten. Gibt bei der CLEAR-Aktion das Adresselement an, bei dem Sie die Übermittlung aufheben möchten. Zur Auswahl stehen:</p> <p>all Führt die CLEAR-Aktion auf allen Adresselementen aus. Diese Option gilt nur für die CLEAR-Aktion.</p> <p><IndexNumber> Führt die Aktion in einem bestimmten Adresselement aus. Um den Index eines Adresselements zu bestimmen, prüfen Sie zunächst die Felder „Field.n.Name“ und suchen Sie das gewünschte Feld. Der Wert n gibt den Index des Feldes an. Sie möchten beispielsweise ZIP-Codes für US-amerikanische Adressen. Nach dem INIT-Aufruf sehen Sie, dass Field.0.Name „Zip“ lautet, was bedeutet, dass der ZIP-Code einen Feldindex von „0“ hat.</p> <p>Mit Ausnahme der SEARCH- und CLEAR-Aktionen wird dieses Eingabefeld für Aktionen ignoriert.</p>
SearchValue	Zeichenfolge [79]	<p>Gibt bei der SEARCH-Aktion den Wert an, den Sie suchen möchten. Dieser Wert muss entsprechend dem Feld gewählt sein, das Sie in FieldIndex angegeben haben. Wenn Sie beispielsweise das Feld „ZIP-Code“ in FieldIndex angegeben haben, würden Sie einen ZIP-Code oder einen partiellen ZIP-Code in dieses Feld eingeben. Ebenso würden Sie einen Ortsnamen oder einen partiellen Ortsnamen in dieses Feld eingeben, wenn Sie in FieldIndex das Feld „Ort“ ausgewählt hätten. Wenn Sie das Feld leer lassen, gibt die Suche alle Werte innerhalb des Kontexts zurück. Weitere Informationen zu Werten innerhalb und außerhalb des Kontexts finden Sie unter Was ist Kontext? auf Seite 226.</p> <p>Mit Ausnahme der SEARCH-Aktion wird dieses Eingabefeld für Aktionen ignoriert.</p>
SessionId	Zeichenfolge [79]	<p>Gibt die Sitzungs-ID an, die Sie für diesen Aufruf verwenden möchten. Rufen Sie Sitzungs-IDs über die INIT-Aktion ab. Wenn eine Sitzung 5 Minuten lang inaktiv ist, wird sie ablaufen und Sie müssen einen neuen INIT-Aufruf ausführen, um eine neue Sitzung zu starten.</p> <p>Dieses Feld ist mit Ausnahme von INIT für alle Felder erforderlich.</p>

Optionen

Tabelle 27: „BuildGlobalAddress“-Optionen

Optionsname	Beschreibung
HomeCountry	Gibt das Standardland an. Geben Sie das Land an, in dem sich die meisten Adressen unter ihren Daten befinden. Wenn beispielsweise die meisten Adressen aus Kanada stammen, geben Sie Kanada an. „BuildGlobalAddress“ verwendet das angegebene Land, um eine Prüfung einzuleiten, wenn es das Land nicht aus den Adressenfeldern Bundesland/Kanton, Postleitzahl und Land ermitteln kann.
OutputCountryFormat	Gibt das Format an, das in der Ausgabe für den Landesnamen verwendet werden soll. Zur Auswahl stehen: <ul style="list-style-type: none"> E Das Land in der Ausgabe auf Deutsch (Standard). I Das Land in der Ausgabe per ISO-Code mit zwei Zeichen ausgedrückt. U Das Land in der Ausgabe per UPU-Code mit drei Zeichen ausgedrückt.
ShowExtraAddressLine	Gibt an, ob in eines der AddressLine-Ausgabefelder der Ort, das Bundesland/Kanton und die Postleitzahl aufgenommen werden sollen. Unabhängig von Ihren Eingaben in dieser Option werden die Ausgabefelder „City“, „State/Province“ und „PostalCode“ immer den Ort, das Bundesland/den Kanton und die Postleitzahl enthalten. <ul style="list-style-type: none"> Y Ja, Ort, Bundesland/Kanton und Postleitzahl in ein AddressLine-Ausgabefeld einschließen (Standard). N Nein, Ort, Bundesland/Kanton und Postleitzahl nicht in ein AddressLine-Ausgabefeld einschließen.
OutputPostalCodeSeparator	Gibt an, ob in ZIP Codes™ oder kanadischen Postleitzahlen Trennzeichen (Leerzeichen oder Bindestriche) verwendet werden sollen. <p>Ein ZIP + 4®-Code mit dem Trennzeichen würde beispielsweise 20706-1844 und ohne Trennzeichen 207061844 lauten. Eine kanadische Postleitzahl mit dem Trennzeichen würde P5E"1S7 und ohne das Trennzeichen P5E1S7 lauten.</p> <ul style="list-style-type: none"> Y Ja, Trennzeichen verwenden (Standard). N Nein, kein Trennzeichen verwenden. <p>Anmerkung: Leerzeichen werden in kanadischen Postleitzahlen und Bindestriche in US-amerikanischen ZIP + 4®-Codes verwendet.</p>

Optionsname	Beschreibung
MaximumResults	Hiermit können Sie den Standardwert für diese Option auf einen beliebigen Wert zwischen 1 und 10000 festlegen. Der eingestellte Standardwert beträgt 50 Datensätze Hinweis: Im Enterprise Designer festgelegte Werte überschreiben die in der Management Console festgelegten Werte.

Ausgabe

BuildGlobalAddress gibt Adressdaten und Rückgabecodes für jede Eingabeadresse zurück.

Adressdaten

Tabelle 28: Ausgabe für BuildGlobalAddress

columnName	Format	Beschreibung
Action	Zeichenfolge [79]	Zeigt den Wert an, der für diesen Aufruf im Feld „Action“ festgelegt wurde. Weitere Informationen zu diesem Eingabefeld finden Sie unter Eingabe auf Seite 216.
AddressLine1	Zeichenfolge [79]	Die formatierte erste Adresszeile.
AddressLine2	Zeichenfolge [79]	Die formatierte zweite Adresszeile.
AddressLine3	Zeichenfolge [79]	Die formatierte dritte Adresszeile.
AddressLine4	Zeichenfolge [79]	Die formatierte vierte Adresszeile.
AddressLine5	Zeichenfolge [79]	Die formatierte fünfte Adresszeile.

columnName	Format	Beschreibung
AddressLine6	Zeichenfolge [79]	Die formatierte sechste Adresszeile.
AddressLine7	Zeichenfolge [79]	Die formatierte siebte Adresszeile.
AddressLine8	Zeichenfolge [79]	Die formatierte achte Adresszeile.
AlternativeContext	Zeichenfolge [79]	Zeigt den Wert an, der für diesen Aufruf im Feld „AlternativeContext“ festgelegt wurde. Weitere Informationen finden Sie unter Eingabe auf Seite 216.
AlternativeIndex	Zeichenfolge [79]	Zeigt den Wert an, der für diesen Aufruf im Feld „AlternativeIndex“ festgelegt wurde. Weitere Informationen zu diesem Eingabefeld finden Sie unter Eingabe auf Seite 216.
Alternatives.InContext	Zeichenfolge [79]	Eine durch Trennzeichen getrennte Liste möglicher Werte für das gesuchte Feld, die dem Kontext der bereits übermittelten Felder entsprechen. Weitere Informationen zum Kontext finden Sie unter Was ist Kontext? auf Seite 226.
Alternatives.InContext.Count	Zeichenfolge [79]	Die Anzahl der Ergebnisse „innerhalb des Kontexts“, die von Ihrer Suche zurückgegeben wurden. Weitere Informationen zum Kontext finden Sie unter Was ist Kontext? auf Seite 226.
Alternatives.OutContext	Zeichenfolge [79]	Eine durch Trennzeichen getrennte Liste möglicher Werte für das gesuchte Feld, die dem Kontext der bereits übermittelten Felder nicht entsprechen. Weitere Informationen zum Kontext finden Sie unter Was ist Kontext? auf Seite 226.

columnName	Format	Beschreibung
Alternatives.OutContext.Count	Zeichenfolge [79]	Die Anzahl der Ergebnisse „außerhalb des Kontexts“, die von Ihrer Suche zurückgegeben wurden. Weitere Informationen zum Kontext finden Sie unter Was ist Kontext? auf Seite 226.
ApartmentLabel	Zeichenfolge [79]	Wohnungsbezeichner (z. B. Whng. oder APT). Beispiel: 123 E Main St. Whng. 3
ApartmentNumber	Zeichenfolge [79]	Wohnungsnummer. Beispiel: 123 E Main St. APT 3
Building	Zeichenfolge [79]	Der Name eines Gebäudes.
City	Zeichenfolge [79]	Der Ortsname.
Country	Zeichenfolge [79]	Zeigt den Wert an, der für diesen Aufruf im Feld „Country“ festgelegt wurde. Weitere Informationen zu diesem Eingabefeld finden Sie unter Eingabe auf Seite 216.
Country	Zeichenfolge [79]	Der zwei- oder dreistellige ISO-Code oder der englische Name des Landes. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
Department	Zeichenfolge [79]	Der Name eines abgegrenzten Teils eines Ganzen, das in Bereiche unterteilt ist. Beispiel: Eine Finanzabteilung in einem Konzern.
Field.n.CommitFlag	Zeichenfolge [79]	Gibt an, ob Sie einen Wert für das Feld n gewählt haben (d. h. einen Wert „übermittelt“ haben). Zur Auswahl stehen: Y Ja, der Wert aus diesem Feld wurde übermittelt. N Nein, der Wert aus diesem Feld wurde nicht übermittelt.

columnName	Format	Beschreibung
Field.n.Index	Zeichenfolge [79]	Ein Indexwert, der als Verweis auf Feld n verwendet wird, wobei n 0 bis 10 ist. Beispiel: Bei US-amerikanischen Adressen ist der Indexwert des Feldes „ZIP“ 0.
Field.n.Name	Zeichenfolge [79]	Der Name des Adresselements, das in Feld n enthalten ist, wobei n 0 bis 10 ist. Beispiel: Bei US-amerikanischen Adressen ist Field.0.Name „ZIP“.
Field.n.Value	Zeichenfolge [79]	Der Wert, der an Feld n übermittelt wurde, wobei n 0 bis 10 ist. Dieses Feld ist beim INIT-Aufruf leer.
FieldIndex	Zeichenfolge [79]	Zeigt den Wert an, der für diesen Aufruf im Feld „FieldIndex“ festgelegt wurde. Weitere Informationen zu diesem Eingabefeld finden Sie unter Eingabe auf Seite 216.
FirmName	Zeichenfolge [79]	Der Name eines Unternehmens. Beispiel: Pitney Bowes 4200 PARLIAMENT PL STE 600 LANHAM MD 20706-1844 USA
HouseNumber	Zeichenfolge [79]	Hausnummer. Beispiel: 123 E Main St. Whng. 3
POBox	Zeichenfolge [79]	Die Nummer des Postfachs. Sollte es sich bei der Adresse um eine Landstraßenadresse handeln, wird hier die Postfachnummer der Landstraße angezeigt.
PostalCode	Zeichenfolge [79]	Die Postleitzahl. In den USA ist dies der ZIP-Code™.
PostalCode.AddOn	Zeichenfolge [79]	Der vierstellige Zusatzteil des ZIP+4®-Codes. Im ZIP-Code™ 60655-1844 lautet der vierstellige Zusatzteil 1844. (Nur US-amerikanische Adressen.)

columnName	Format	Beschreibung
PostalCode.Base	Zeichenfolge [79]	Der fünfstellige ZIP-Code™. Beispiel: 20706 (nur US-amerikanische Adressen.)
Principality	Zeichenfolge [79]	Eine Region innerhalb eines Landes. England, Schottland und Wales sind beispielsweise Fürstentümer. Dieses Feld ist in der Regel leer.
SearchFieldIndex	Zeichenfolge [79]	Der Indexwert des Feldes, das in der vorherigen Suchaktion gesucht wurde.
SearchValue	Zeichenfolge [79]	Zeigt den Wert an, der für diesen Aufruf im Feld „SearchValue“ festgelegt wurde. Weitere Informationen zu diesem Eingabefeld finden Sie unter Eingabe auf Seite 216.
SessionId	Zeichenfolge [79]	Zeigt den Wert an, der für diesen Aufruf im Feld „SessionId“ festgelegt wurde. Weitere Informationen zu diesem Eingabefeld finden Sie unter Eingabe auf Seite 216.
StateProvince	Zeichenfolge [79]	Die Abkürzung für das Bundesland oder den Kanton.
StreetName	Zeichenfolge [79]	Straßenname. Beispiel: 123 E Main St. Whng. 3
StreetSuffix	Zeichenfolge [79]	Straßensuffix. Beispiel: 123 E Main St. Whng. 3
SubCity	Zeichenfolge [79]	Ein Bezirk oder Vorort. „SubCity“ wird in Ländern verwendet, in denen der Bezirk oder Stadtteil in die Adresse aufgenommen wird. Beispiel: 27 Crystal Way Bradley Stoke Bristol BS32 8GA In diesem Fall ist „Bradley Stoke“ die SubCity-Angabe.

columnName	Format	Beschreibung
SubStreet	Zeichenfolge [79]	<p>Die zweite Straßenadresse, die zur Identifizierung einer Adresse verwendet wird. Unterstraßen werden in Ländern verwendet, in denen häufig zwei Straßennamen in der Adresse angegeben werden. Beispiel:</p> <p>12 The Mews High Street</p> <p>In diesem Beispiel ist „High Street“ die Unterstraße. Mithilfe von Unterstraßen kann der Zustellungsort genau ermittelt werden. Im Beispiel könnte „The Mews“ eine kleine Straße sein, die eine weitere Straßenidentifizierung erfordert, um die Adresse genau zu finden. Also wird „High Street“ einbezogen. In diesem Fall ist „High Street“ die Hauptstraße bzw. bekannte Straße.</p>
USCountyName	Zeichenfolge [79]	Bei US-amerikanischen Adressen ist es der Name des Bezirks, in dem sich die Adresse befindet.

Rückgabecodes

Tabelle 29: Rückgabecodes für BuildGlobalAddress

columnName	Format	Beschreibung				
Status	Zeichenfolge [79]	<p>Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs.</p> <table> <tr> <td>null</td> <td>Erfolg</td> </tr> <tr> <td>F</td> <td>Fehler</td> </tr> </table>	null	Erfolg	F	Fehler
null	Erfolg					
F	Fehler					
Status.Code	Zeichenfolge [79]	<p>Ursache für den Fehler, falls vorhanden.</p> <ul style="list-style-type: none"> • SessionError • SeverError • CountryNotFound 				

columnName	Format	Beschreibung
Status.Description	Zeichenfolge [79]	<p>Beschreibung des Problems, wenn eines vorliegt.</p> <p>Please initialize new session Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=SessionError.</p> <p>Null or empty action Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=SessionError.</p> <p>Unknown action Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=SessionError.</p> <p>Invalid session Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=SessionError.</p> <p>Invalid value for Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=SessionError.</p> <p>Cannot Search Committed Field Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=SessionError.</p> <p>Module not licensed Dieser Wert wird angezeigt, wenn Status.Code=ServerError.</p> <p>Could Not Identify Country Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=CountryNotFound.</p>

Was ist Kontext?

Wenn Sie die Suche nach einem Adressenelement durchführen, prüft BuildGlobalAddress die Adressenelemente, die Sie bereits übergeben haben, und teilt die Werte, die zurückgegeben werden, basierend darauf auf, ob die zurückgegebenen Werte im Kontext der von Ihnen übergebenen Adressenelemente vorhanden sind. In den USA gibt es beispielsweise die folgenden Städte:

In Illinois:

- CHICAGO
- CHICAGO HTS
- CHICAGO RIDGE
- NORTH CHICAGO
- WEST CHICAGO

In Indiana:

- EAST CHICAGO

In Nevada:

- CHICAGO PARK

Wenn Sie bereits den Wert „IN“ (Indiana) für den Bundesstaat übergeben haben und anschließend nach der Stadt „chicago“ gesucht haben, gibt BuildGlobalAddress EAST CHICAGO als „In Kontext“-Ergebnis zurück, da dies in Indiana liegt, und alle anderen Übereinstimmungen für „chicago“ als „Nicht in Kontext“-Ergebnisse. Gleichermaßen gibt BuildGlobalAddress EAST CHICAGO und CHICAGO PARK als „Nicht in Kontext“-Ergebnisse zurück und CHICAGO, CHICAGO HTS, CHICAGO RIDGE, NORTH CHICAGO und WEST CHICAGO als „In Kontext“-Ergebnisse, wenn Sie den Wert „IL“ (Illinois) für den Bundesstaat übergeben haben.

GetGlobalCandidateAddresses

GetGlobalCandidateAddresses gibt eine Liste der Adressen zurück, die bei einer angegebenen Eingabeadresse als Übereinstimmungen gelten. Wenn die Eingabeadresse mit mehreren Adressen in der Address Now-Datenbank übereinstimmt, werden die möglichen Übereinstimmungen zurückgegeben. Wenn die Eingabeadresse nur mit einer Adresse in der Address Now-Datenbank übereinstimmt, werden keine Adressdaten zurückgegeben.

GetGlobalCandidateAddresses ist Teil des Address Now-Moduls.

Eingabe

GetGlobalCandidateAddresses nimmt eine Standardadresse als Eingabe. Alle Adressen verwenden dieses Format, unabhängig davon, aus welchem Land die Adresse stammt. „AddressLine1“ und „Country“ sind erforderliche Eingabefelder. Die anderen Felder sind optional.

Tabelle 30: „GetGlobalCandidateAddresses“-Eingabe

columnName	Format	Beschreibung
AddressLine1	Zeichenfolge [79]	Erste Adresszeile. Dies ist ein erforderliches Feld.
AddressLine2	Zeichenfolge [79]	Zweite Adresszeile
AddressLine3	Zeichenfolge [79]	Dritte Adresszeile

columnName	Format	Beschreibung
AddressLine4	Zeichenfolge [79]	Vierte Adresszeile
AddressLine5	Zeichenfolge [79]	Fünfte Adresszeile
AddressLine6	Zeichenfolge [79]	Sechste Adresszeile
AddressLine7	Zeichenfolge [79]	Siebte Adresszeile
AddressLine8	Zeichenfolge [79]	Achte Adresszeile
City	Zeichenfolge [79]	Ortsname
StateProvince	Zeichenfolge [79]	Bundesland oder Kanton.
PostalCode	Zeichenfolge [10]	Die Postleitzahl der Adresse in einem der folgenden Formate: 99999 99999-9999 A9A9A9 A9A 9A9 9999 999
Country	Zeichenfolge	Das Land. Geben Sie das Land in dem Format an, das Sie für das Eingabeländerformat ausgewählt haben (englischer Name oder ISO-Code). Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
FirmName	Zeichenfolge [79]	Der Unternehmens- bzw. Firmenname.

Optionen

Tabelle 31: „GetGlobalCandidateAddresses“-Optionen

optionName	Beschreibung/Gültige Werte
HomeCountry	Gibt das Standardland an. Geben Sie das Land an, das das Zielland Ihrer meisten Sendungen ist. Geben Sie beispielsweise Kanada an, wenn die meisten Ihrer Sendungen nach Kanada gehen. GetGlobalCandidateAddresses verwendet das von Ihnen angegebene Land, um eine Überprüfung einzuleiten, wenn es das Land nicht über die Adressfelder „Bundesland/Kanton“, „Postleitzahl“ und „Land“ bestimmen kann. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
OutputCasing	Gibt die Groß-/Kleinschreibung der Ausgabedaten an. Zur Auswahl stehen: <ul style="list-style-type: none"> M Die Ausgabe in Groß- und Kleinbuchstaben (Standard). Beispiel: Hauptstraße 123, 12345 Meine Stadt U Die Ausgabe in Großbuchstaben. Beispiel: HAUPTSTRASSE 123, 12345 MEINE STADT
OutputCountryFormat	Gibt das Format an, das in der Ausgabe für den Landesnamen verwendet werden soll. Zur Auswahl stehen: <ul style="list-style-type: none"> E Das Land in der Ausgabe auf Deutsch (Standard). I Das Land in der Ausgabe im zweistelligen ISO-Code-Format. U Das Land in der Ausgabe im dreistelligen UPU-Code-Format.
ShowExtraAddressLine	Gibt an, ob in eines der AddressLine-Ausgabefelder der Ort, das Bundesland/Kanton und die Postleitzahl aufgenommen werden sollen. Unabhängig von Ihren Eingaben in dieser Option werden die Ausgabefelder „City“, „State/Province“ und „PostalCode“ immer den Ort, das Bundesland/den Kanton und die Postleitzahl enthalten. <ul style="list-style-type: none"> Y Ja, Ort, Bundesland/Kanton und Postleitzahl in ein AddressLine-Ausgabefeld einschließen (Standard). N Nein, Ort, Bundesland/Kanton und Postleitzahl nicht in ein AddressLine-Ausgabefeld einschließen.

optionName	Beschreibung/Gültige Werte
OutputPostalCodeSeparator	<p>Geben Sie an, ob in ZIP™ Codes oder kanadischen Postleitzahlen Trennzeichen (Leerzeichen oder Bindestriche) verwendet werden sollen.</p> <p>Ein ZIP + 4®-Code mit dem Trennzeichen würde beispielsweise 20706-1844 und ohne Trennzeichen 207061844 lauten. Eine kanadische Postleitzahl mit dem Trennzeichen würde P5E"1S7 und ohne das Trennzeichen P5E1S7 lauten.</p> <p>Y Ja, Trennzeichen verwenden (Standard).</p> <p>N Nein, kein Trennzeichen verwenden.</p> <p>Anmerkung: Leerzeichen werden in kanadischen Postleitzahlen und Bindestriche in US-amerikanischen ZIP + 4®-Codes verwendet.</p>
MaximumResults	Die maximale Anzahl auszugebender Kandidatenadressen. Der Standardwert ist 50. Der maximale Wert ist 100.
ReturnUserData	<p>Gibt an, ob Ausgabedaten aus der eingegebenen Adresse eingeschlossen werden sollen, die nicht überprüft werden konnten.</p> <p>Y Ja, Eingabedaten einschließen, die nicht überprüft werden konnten.</p> <p>N Nein, keine Eingabedaten einschließen, die nicht überprüft werden konnten (Standard).</p>

Ausgabe

GetGlobalCandidateAddresses gibt für die einzelnen Adressen Adressdaten und Rückgabecodes zurück.

Adressdaten

Tabelle 32: Ausgabe der Adressdaten von GetGlobalCandidateAddresses

columnName	Format	Beschreibung
AddressLine1	Zeichenfolge [79]	Die formatierte erste Adresszeile.
AddressLine2	Zeichenfolge [79]	Die formatierte zweite Adresszeile.

columnName	Format	Beschreibung
AddressLine3	Zeichenfolge [79]	Die formatierte dritte Adresszeile.
AddressLine4	Zeichenfolge [79]	Die formatierte vierte Adresszeile.
AddressLine5	Zeichenfolge [79]	Die formatierte fünfte Adresszeile.
AddressLine6	Zeichenfolge [79]	Die formatierte sechste Adresszeile.
AddressLine7	Zeichenfolge [79]	Die formatierte siebte Adresszeile.
AddressLine8	Zeichenfolge [79]	Die formatierte achte Adresszeile.
ApartmentLabel	Zeichenfolge [79]	Wohnungsbezeichner (z. B. Whng. oder APT). Beispiel: 123 E Main St. APT 3
ApartmentNumber	Zeichenfolge [79]	Wohnungsnummer. Beispiel: 123 E Main St. APT 3
Building	Zeichenfolge [79]	Der Name eines Gebäudes.
City	Zeichenfolge [79]	Der Ortsname.
Country	Zeichenfolge [79]	Der ISO-Code oder englische Name des Landes. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.

columnName	Format	Beschreibung
Department	Zeichenfolge [79]	Der Name eines abgegrenzten Teils eines Ganzen, das in Bereiche unterteilt ist. Beispiel: Eine Finanzabteilung in einem Konzern.
FirmName	Zeichenfolge [79]	Der Name eines Unternehmens. Beispiel: Pitney Bowes 4200 PARLIAMENT PL STE 600 LANHAM MD 20706-1844 USA
HouseNumber	Zeichenfolge [79]	Hausnummer. Beispiel: 123 E Main St. Whng. 3
POBox	Zeichenfolge [79]	Postfachnummer. Sollte es sich bei der Adresse um eine Landstraßenadresse handeln, wird hier die Postfachnummer der Landstraße angezeigt.
PostalCode	Zeichenfolge [79]	Die Postleitzahl entsprechend den Vorschriften der lokalen Postbehörde. In den USA ist die Postleitzahl beispielsweise der ZIP-Code.
PostalCode.AddOn	Zeichenfolge [79]	Bei US-amerikanischen Adressen die letzten vier Zeichen des ZIP + 4 [®] Code.
PostalCode.Base	Zeichenfolge [79]	Bei US-amerikanischen Adressen der fünfstellige ZIP-Code.
Principality	Zeichenfolge [79]	Eine Region innerhalb eines Landes. England, Schottland und Wales sind beispielsweise Fürstentümer. Dieses Feld ist in der Regel leer.
StateProvince	Zeichenfolge [79]	Die Abkürzung für das Bundesland oder den Kanton.
StreetName	Zeichenfolge [79]	Straßenname. Beispiel: 123 E Main St. Whng. 3

columnName	Format	Beschreibung
StreetSuffix	Zeichenfolge [79]	Straßensuffix. Beispiel: 123 E Main St. Whng. 3
SubCity	Zeichenfolge [79]	Ein Bezirk oder Vorort. „SubCity“ wird in Ländern verwendet, in denen der Bezirk oder Stadtteil in die Adresse aufgenommen wird. Beispiel: 27 Crystal Way Bradley Stoke Bristol BS32 8GA In diesem Fall ist „Bradley Stoke“ die SubCity-Angabe.
SubStreet	Zeichenfolge [79]	Die zweite Straßenadresse, die zur Identifizierung einer Adresse verwendet wird. Unterstraßen werden in Ländern verwendet, in denen häufig zwei Straßennamen in der Adresse angegeben werden. Beispiel: 12 The Mews High Street In diesem Beispiel ist „High Street“ die Unterstraße. Mithilfe von Unterstraßen kann der Zustellungsort genau ermittelt werden. Im Beispiel könnte „The Mews“ eine kleine Straße sein, die eine weitere Straßenidentifizierung erfordert, um die Adresse genau zu finden. Also wird „High Street“ einbezogen. In diesem Fall ist „High Street“ die Hauptstraße bzw. bekannte Straße.
USCountyName	Zeichenfolge [79]	Bei US-amerikanischen Adressen ist es der Name des Bezirks, in dem sich die Adresse befindet.

Rückgabecodes

Tabelle 33: GetGlobalCandidateAddresses-Rückgabecodes

columnName	Format	Beschreibung
ACRCode	Zeichenfolge [79]	Der Code für das Adressenkorrekturergebnis beschreibt, welche Daten in jedem Datensatz geändert wurden. Informationen zur Bedeutung dieses Codes finden Sie unter Der ACR-Code auf Seite 264.

columnName	Format	Beschreibung
Confidence	Zeichenfolge [79]	Das Zuverlässigkeitsniveau, das der zurückgegebenen Adresse zugewiesen ist. Der Bereich reicht von 0 (null) bis 100. Null gibt an, dass ein Fehler vorliegt; 100 gibt an, dass das Zuverlässigkeitsniveau sehr hoch ist und die Übereinstimmungsergebnisse richtig sind.
Status	Zeichenfolge [79]	Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs. Null Erfolg F Fehler
Status.Code	Zeichenfolge [79]	Ursache für den Fehler, falls vorhanden. <ul style="list-style-type: none"> RequestFailed ServerError CountryNotFound
Status.Description	Zeichenfolge [79]	Beschreibung des Problems, wenn eines vorliegt. Maximum records cannot be set to 0. Minimum value should be 1 Dieser Wert wird angezeigt, wenn Status.Code=RequestFailed. Address Not Found Dieser Wert wird angezeigt, wenn Status.Code=RequestFailed. Module not licensed Dieser Wert wird angezeigt, wenn Status.Code=ServerError. Could Not Identify Country Dieser Wert wird angezeigt, wenn Status.Code=CountryNotFound.

ValidateGlobalAddress

„ValidateGlobalAddress“ bietet eine erweiterte Adressenstandardisierung und -überprüfung für Adressen außerhalb der USA und Kanada. „ValidateGlobalAddress“ kann auch Adressen in den USA und Kanada überprüfen. Die Stärke liegt jedoch in der Überprüfung von Adressen in anderen Ländern. Wenn Sie Adressen außerhalb der USA und Kanada überprüfen möchten, müssen Sie die Verwendung von „ValidateGlobalAddress“ in Betracht ziehen.

„ValidateGlobalAddress“ ist ein Teil des Address Now-Moduls.

Eingabe

„ValidateGlobalAddress“ nutzt eine Standardadresse als Eingabe. Alle Adressen verwenden dieses Format, unabhängig davon, aus welchem Land die Adresse stammt.

Tabelle 34: „ValidateGlobalAddress“-Eingabe

columnName	Format	Beschreibung
AddressLine1	Zeichenfolge [79]	Erste Adresszeile
AddressLine2	Zeichenfolge [79]	Zweite Adresszeile
AddressLine3	Zeichenfolge [79]	Dritte Adresszeile
AddressLine4	Zeichenfolge [79]	Vierte Adresszeile
AddressLine5	Zeichenfolge [79]	Fünfte Adresszeile
AddressLine6	Zeichenfolge [79]	Sechste Adresszeile
AddressLine7	Zeichenfolge [79]	Siebte Adresszeile
AddressLine8	Zeichenfolge [79]	Achte Adresszeile
City	Zeichenfolge [79]	Ortsname

columnName	Format	Beschreibung
StateProvince	Zeichenfolge [79]	Bundesland oder Kanton.
PostalCode	Zeichenfolge [79]: 99999 99999-9999 A9A9A9 A9A 9A9 9999 999	Die Postleitzahl zu der Adresse. In den USA ist dies der ZIP-Code™.
Country	Zeichenfolge [79]	Geben Sie das Land in dem Format an, das Sie für das Eingabeländerformat ausgewählt haben (englischer Name oder ISO-Code). Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
FirmName	Zeichenfolge [79]	Der Unternehmens- bzw. Firmenname.

Optionen

Eingabedatenoptionen

Tabelle 35: Eingabedatenoptionen für „ValidateGlobalAddress“

optionName	Beschreibung
HomeCountry	Gibt das Standardland an. Sie sollten das Land angeben, in dem sich die meisten Adressen befinden. Wenn Sie beispielsweise vor allem kanadische Adressen verarbeiten, geben Sie Kanada an. „ValidateGlobalAddress“ verwendet das Heimatland, um eine Prüfung einzuleiten, wenn es das Land nicht aus den Adressenfeldern „StateProvince“, „PostalCode“ und „Country“ ermitteln kann. Eine Liste der Werte finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.

Ausgabedatenoptionen

Tabelle 36: Ausgabedatenoptionen für ValidateGlobalAddress

optionName	Beschreibung
OutputCasing	<p>Gibt die Groß-/Kleinschreibung der Ausgabedaten an. Zur Auswahl stehen:</p> <p>M Gibt die Ausgabe in Groß- und Kleinbuchstaben zurück (Standard). Beispiel: Hauptstraße 123, 12345 Meine Stadt</p> <p>U Gibt die Ausgabe in Großbuchstaben zurück. Beispiel: HAUPTSTRASSE 123, 12345 MEINE STADT</p>
OutputCountryFormat	<p>Gibt das Format an, das in der Ausgabe für den Landesnamen verwendet werden soll. Zur Auswahl stehen:</p> <p>E Das Land in der Ausgabe ist Deutschland (Standard).</p> <p>I Das Land in der Ausgabe im zweistelligen ISO-Code-Format.</p> <p>U Das Land in der Ausgabe im dreistelligen UPU-Code-Format.</p>
StandardizeAddressOnFail	<p>Gibt an, ob eine standardisierte Adresse zurückgegeben werden soll, wenn eine Adresse nicht überprüft werden kann. Die Adresse wird entsprechend dem durch das Adressland bevorzugten Adressenformat formatiert. Wenn diese Option nicht aktiviert wird, sind die Felder für die Komponenten der Ausgabeadresse (Straßenname, Hausnummer usw.) leer, wenn die Adressenüberprüfung fehlschlägt.</p> <p>N Nein, fehlgeschlagene Adressen nicht formatieren (Standard).</p> <p>Y Ja, fehlgeschlagene Adressen standardisieren.</p>
ShowExtraAddressLine	<p>Gibt an, ob in eines der AddressLine-Ausgabefelder der Ort, das Bundesland/der Kanton und die Postleitzahl aufgenommen werden sollen. Unabhängig von Ihren Eingaben in dieser Option werden die Ausgabefelder „City“, „State/Province“ und „PostalCode“ immer den Ort, das Bundesland/den Kanton und die Postleitzahl enthalten.</p> <p>Y Ja, Ort, Bundesland/Kanton und Postleitzahl in ein AddressLine-Ausgabefeld einschließen (Standard).</p> <p>N Nein, Ort, Bundesland/Kanton und Postleitzahl nicht in ein AddressLine-Ausgabefeld einschließen.</p>

optionName	Beschreibung
OutputPostalCodeSeparator	<p>Gibt an, ob Trennzeichen (Leerzeichen oder Bindestriche) in ZIP™ Codes oder kanadischen Postleitzahlen verwendet werden sollen.</p> <p>Ein ZIP + 4®-Code mit dem Trennzeichen würde beispielsweise 20706-1844 und ohne Trennzeichen 207061844 lauten. Eine kanadische Postleitzahl mit dem Trennzeichen würde P5E"1S7 und ohne das Trennzeichen P5E1S7 lauten.</p> <p>Y Ja, Trennzeichen verwenden (Standard).</p> <p>N Nein, kein Trennzeichen verwenden.</p> <p>Anmerkung: Leerzeichen werden in kanadischen Postleitzahlen und Bindestriche in US-amerikanischen ZIP + 4®-Codes verwendet.</p>
FormatOnFail	<p>Gibt an, ob eine formatierte Adresse zurückgegeben werden soll, wenn eine Adresse nicht überprüft werden kann. Die Adresse wird entsprechend dem durch das Adressland bevorzugten Adressenformat formatiert.</p> <p>Y Ja, eine formatierte Adresse zurückgeben, wenn eine Adresse nicht überprüft werden kann.</p> <p>N Nein, formatierte nicht Adresse zurückgeben, wenn eine Adresse nicht überprüft werden kann (Standard).</p>
ValidateAddress	<p>Aktiviert die Adressenüberprüfung. Bei der Adressenüberprüfung werden die folgenden Aktionen ausgeführt:</p> <ul style="list-style-type: none"> • Vergleicht Komponenten mit den Referenzdaten des jeweiligen Landes • Korrigiert Rechtschreibfehler • Fügt fehlende Komponenten hinzu • Berichtigt Postleitzahlen oder fügt sie hinzu <p>Y Ja, Adressen überprüfen (Standard).</p> <p>N Nein, Adressen nicht überprüfen.</p>
FormatAddress	<p>Formatiert die Adresskomponenten in die gesetzlich vorgegebenen postalischen oder üblichen Formate.</p> <p>Y Ja, Adressen formatieren (Standard).</p> <p>N Nein, Adressen nicht formatieren.</p>

Standardisierungsoptionen

Tabelle 37: Standardisierungsoptionen für „ValidateGlobalAddress“

optionName	Beschreibung
FlagVulgarWords	Gibt an, ob Vulgärsprache in der Ausgabe mithilfe des Formats „>VulgarWord<“ markiert werden soll.
	Y Ja
	N Nein (Standard)
DebugOutput	Gibt an, ob in die Ausgabefelder „Email1“, „Email2“, „URL1“ und „URL2“ Fehlerbehandlungsinformationen aufgenommen werden sollen.
	Y Ja
	N Nein (Standard)
ReportVulgarWords	Gibt an, ob auf Vulgärsprache geachtet werden soll. Wenn diese Option aktiviert ist, gibt „ValidateGlobalAddress“ einen Wert im Ausgabefeld „WCRCode“ zurück, der die Ergebnisse angibt.
	Y Ja
	N Nein (Standard)
StandardizeComponent.Department	Gibt an, ob beim Standardisieren einer Adresse das Feld „Department“ geladen werden soll.
	Y Ja (Standard)
	N Nein
StandardizeComponent.FirmName	Gibt an, ob beim Standardisieren einer Adresse das Feld „FirmName“ geladen werden soll.
	Y Ja (Standard)
	N Nein
StandardizeComponent.Building	Gibt an, ob beim Standardisieren einer Adresse das Feld „Building“ geladen werden soll.
	Y Ja (Standard)
	N Nein

optionName	Beschreibung
StandardizeComponent.SubBuilding	Gibt an, ob beim Standardisieren einer Adresse das Feld „SubBuilding“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.HouseNumber	Gibt an, ob beim Standardisieren einer Adresse das Feld „HouseNumber“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.SubStreet	Gibt an, ob beim Standardisieren einer Adresse das Feld „SubStreet“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.StreetName	Gibt an, ob beim Standardisieren einer Adresse das Feld „StreetName“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.POBox	Gibt an, ob beim Standardisieren einer Adresse das Feld „POBox“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.SubCity	Gibt an, ob beim Standardisieren einer Adresse das Feld „SubCity“ geladen werden soll. Y Ja (Standard) N Nein

optionName	Beschreibung
StandardizeComponent.City	Gibt an, ob beim Standardisieren einer Adresse das Feld „City“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.USCountyName	Gibt an, ob beim Standardisieren einer Adresse das Feld „USCountyName“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.StateProvince	Gibt an, ob beim Standardisieren einer Adresse das Feld „StateProvince“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.Principality	Gibt an, ob beim Standardisieren einer Adresse das Feld „Principality“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.PostalCode	Gibt an, ob beim Standardisieren einer Adresse das Feld „PostalCode“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.Plus4	Gibt an, ob beim Standardisieren einer Adresse das Feld „+4“ geladen werden soll. Y Ja (Standard) N Nein
StandardizeComponent.Country	Gibt an, ob beim Standardisieren einer Adresse das Feld „Country“ geladen werden soll. Y Ja (Standard) N Nein

Überprüfungsoptionen

Tabelle 38: Überprüfungsoptionen unter „Validate Global Address“

Optionsname	Beschreibung
ValidateComponent.Department	Gibt an, ob bei der Überprüfung einer Adresse das Feld „Department“ mit einbezogen werden soll.
	Y Ja
	N Nein (Standard)
ValidateComponent.FirmName	Gibt an, ob bei der Überprüfung einer Adresse das Feld „FirmName“ mit einbezogen werden soll.
	Y Ja
	N Nein (Standard)
ValidateComponent.Building Option.ValidateComponent.Building	Gibt an, ob bei der Überprüfung einer Adresse das Feld „Building“ mit einbezogen werden soll.
	Y Ja (Standard)
	N Nein
ValidateComponent.SubBuilding	Gibt an, ob bei der Überprüfung einer Adresse das Feld „SubBuilding“ mit einbezogen werden soll.
	Y Ja (Standard)
	N Nein
ValidateComponent.HouseNumber	Gibt an, ob bei der Überprüfung einer Adresse das Feld „HouseNumber“ mit einbezogen werden soll.
	Y Ja (Standard)
	N Nein

Optionsname	Beschreibung
ValidateComponent.SubStreet	Gibt an, ob bei der Überprüfung einer Adresse das Feld „SubStreet“ mit einbezogen werden soll.
	Y Ja (Standard)
	N Nein
ValidateComponent.StreetName	Gibt an, ob bei der Überprüfung einer Adresse das Feld „StreetName“ mit einbezogen werden soll.
	Y Ja (Standard)
	N Nein
ValidateComponent.POBox	Gibt an, ob bei der Überprüfung einer Adresse das Feld „POBox“ mit einbezogen werden soll.
	Y Ja (Standard)
	N Nein
ValidateComponent.SubCity	Gibt an, ob bei der Überprüfung einer Adresse das Feld „SubCity“ mit einbezogen werden soll.
	Y Ja (Standard)
	N Nein
ValidateComponent.City	Gibt an, ob bei der Überprüfung einer Adresse das Feld „City“ mit einbezogen werden soll.
	Y Ja (Standard)
	N Nein
ValidateComponent.USCountyName	Gibt an, ob bei der Überprüfung einer Adresse das Feld „USCountyName“ mit einbezogen werden soll.
	Y Ja (Standard)
	N Nein

Optionsname	Beschreibung
ValidateComponent.StateProvince	Gibt an, ob bei der Überprüfung einer Adresse das Feld „StateProvince“ mit einbezogen werden soll. Y Ja (Standard) N Nein
ValidateComponent.Principality	Gibt an, ob bei der Überprüfung einer Adresse das Feld „Principality“ mit einbezogen werden soll. Y Ja (Standard) N Nein
ValidateComponent.PostalCode	Gibt an, ob bei der Überprüfung einer Adresse das Feld „PostalCode“ mit einbezogen werden soll. Y Ja (Standard) N Nein
ValidateComponent.Plus4	Gibt an, ob bei der Überprüfung einer Adresse das Feld „+4“ mit einbezogen werden soll. Y Ja (Standard) N Nein
ValidateComponent.Country	Gibt an, ob bei der Überprüfung einer Adresse das Feld „Country“ mit einbezogen werden soll. Y Ja (Standard) N Nein
ForceUpdate.Department	Gibt an, ob bei der Überprüfung einer Adresse das Feld „Country“ korrigiert werden soll. Y Ja (Standard) N Nein

Optionsname	Beschreibung
ForceUpdate.FirmName	Gibt an, ob bei der Überprüfung einer Adresse das Feld „FirmName“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.Building	Gibt an, ob bei der Überprüfung einer Adresse das Feld „Building“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.SubBuilding	Gibt an, ob bei der Überprüfung einer Adresse das Feld „SubBuilding“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.HouseNumber	Gibt an, ob bei der Überprüfung einer Adresse das Feld „HouseNumber“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.SubStreet	Gibt an, ob bei der Überprüfung einer Adresse das Feld „SubStreet“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.StreetName	Gibt an, ob bei der Überprüfung einer Adresse das Feld „StreetName“ korrigiert werden soll. Y Ja (Standard) N Nein

Optionsname	Beschreibung
ForceUpdate.POBox	Gibt an, ob bei der Überprüfung einer Adresse das Feld „POBox“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.SubCity	Gibt an, ob bei der Überprüfung einer Adresse das Feld „SubCity“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.City	Gibt an, ob bei der Überprüfung einer Adresse das Feld „City“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.USCountyName	Gibt an, ob bei der Überprüfung einer Adresse das Feld „USCountyName“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.StateProvince	Gibt an, ob bei der Überprüfung einer Adresse das Feld „StateProvince“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.Principality	Gibt an, ob bei der Überprüfung einer Adresse das Feld „Principality“ korrigiert werden soll. Y Ja (Standard) N Nein

Optionsname	Beschreibung
ForceUpdate.PostalCode	Gibt an, ob bei der Überprüfung einer Adresse das Feld „PostalCode“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.Plus4	Gibt an, ob bei der Überprüfung einer Adresse das Feld „+4“ korrigiert werden soll. Y Ja (Standard) N Nein
ForceUpdate.Country	Gibt an, ob bei der Überprüfung einer Adresse das Feld „Country“ korrigiert werden soll. Y Ja (Standard) N Nein
ReplaceAlias.Department	Gibt an, ob das Feld „Department“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird. Y Ja N Nein (Standard)
ReplaceAlias.FirmName	Gibt an, ob das Feld „FirmName“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird. Y Ja N Nein (Standard)
ReplaceAlias.Building	Gibt an, ob das Feld „Building“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird. Y Ja N Nein (Standard)

Optionsname	Beschreibung
ReplaceAlias.SubBuilding	Gibt an, ob das Feld „SubBuilding“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)
ReplaceAlias.HouseNumber	Gibt an, ob das Feld „HouseNumber“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)
ReplaceAlias.SubStreet	Gibt an, ob das Feld „SubStreet“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)
ReplaceAlias.StreetName	Gibt an, ob das Feld „StreetName“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)
ReplaceAlias.POBox	Gibt an, ob das Feld „POBox“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)
ReplaceAlias.SubCity	Gibt an, ob das Feld „Subcity“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)

Optionsname	Beschreibung
ReplaceAlias.City	Gibt an, ob das Feld „City“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)
ReplaceAlias.USCountyName	Gibt an, ob das Feld „USCountyName“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)
ReplaceAlias.StateProvince	Gibt an, ob das Feld „StateProvince“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)
ReplaceAlias.Principality	Gibt an, ob das Feld „Principality“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)
ReplaceAlias.PostalCode	Gibt an, ob das Feld „PostalCode“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja (Standard)
	N Nein
ReplaceAlias.Plus4	Gibt an, ob das Feld „+4“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.
	Y Ja
	N Nein (Standard)

Optionsname	Beschreibung
ReplaceAlias.Country	<p>Gibt an, ob das Feld „Country“ überschrieben werden soll, wenn ein Alias in der Datenbank „Address Now“ gefunden wird.</p> <p>Y Ja</p> <p>N Nein (Standard)</p>
CautiousUpdate	<p>Diese Option wird zusammen mit „Update erzwingen“ verwendet und stellt sicher, dass während der Verarbeitung keine wichtigen Änderungen an den Daten vorgenommen werden.</p> <p>Y Ja</p> <p>N Nein (Standard)</p>
CrossComponentMatch	<p>Gibt an, ob häufig vorkommende Fehler bei der Adressenstandardisierung und -überprüfung durch komponentenübergreifendes Vergleichen korrigiert werden sollen. Das komponentenübergreifende Vergleichen prüft auf Übereinstimmungen zwischen Daten in einem Feld der Eingabedaten und einem anderen Feld in der Datenbank „Address Now“.</p> <p>Y Ja</p> <p>N Nein (Standard)</p>

Optionsname	Beschreibung
UseReferenceDiacritics	<p>Gibt an, ob „Validate Global Address“ die Adresse so ändert, dass sie mit den Diakritika (Akzente, Umlaute usw.) in der postalischen Datenbank übereinstimmt, wenn nur die Diakritika geändert wurden. Zur Auswahl stehen:</p> <p>Y Ja (Standard)</p> <p>N Nein</p> <p>Wenn beispielsweise Diakritische Referenzzeichen verwenden aktiviert ist, tritt Folgendes ein:</p> <p>Eingabestadt: Chalon-Sur-Saône Stad in der postalischen Datenbank: CHALON SUR SAONE Ausgabestadt: CHALON SUR SAONE</p> <p>Eingabestadt: ARTEMIVS'K Stad in der postalischen Datenbank: ARTEMIVSK Ausgabestadt: ARTEMIVSK</p> <p>Wenn Diakritische Referenzzeichen verwenden nicht aktiviert ist, tritt Folgendes ein:</p> <p>Eingabestadt: Chalon-Sur-Saône Referenzstadt: CHALON SUR SAONE Ausgabestadt: Chalon-Sur-Saône</p> <p>Eingabestadt: ARTEMIVS'K Referenzstadt: ARTEMIVSK Ausgabestadt: ARTEMIVS'K</p> <p>Beachten Sie, dass diese Option keine Auswirkung auf die Option Transkription hat.</p>
KeepStandardizationChanges	<p>Gibt an, ob Standardisierungsänderungen, wie z. B. das Ändern von „ROAD“ in „RD“, im ACR-Code ausgewiesen werden sollen.</p> <p>Y Ja</p> <p>N Nein (Standard)</p>

Optionsname	Beschreibung
AcceptanceLevel	<p>Die Einstellung „Akzeptanzniveau“ gibt die minimale Anzahl von Adressenkomponenten an, die überprüft werden müssen, damit die gesamte Adresse als überprüft gilt. Der für das Akzeptanzniveau angegebene Wert entspricht dem zweiten Zeichen des ACR-Codes. Weitere Informationen finden Sie unter Der ACR-Code auf Seite 264.</p> <p>Das Akzeptanzniveau unterscheidet sich von der Option „InnerMatchScore“ darin, dass das Akzeptanzniveau misst, wie viele Komponenten „Validate Global Address“ überprüft hat. Dabei ist es unerheblich, wie gut die überprüften Komponenten mit den Adresskomponenten in den postalischen Datenbanken übereinstimmen. Dahingegen gibt „InnerMatchScore“ die Wahrscheinlichkeit an, dass die Ausgabeadresse korrekt ist, also die überprüfte Version der Eingabeadresse.</p> <p>Zur Auswahl stehen:</p> <ul style="list-style-type: none"> -1 Das Akzeptanzniveau wird automatisch auf ein entsprechendes Niveau basierend auf dem Land der Adresse festgelegt. US-amerikanische Adressen werden z. B. mit einem Akzeptanzniveau von 4 verarbeitet. 0 Es werden keine Komponenten überprüft (Standard). 1 Es wird nur das Land überprüft. 2 Es werden Stadt und Land überprüft. 3 Es werden Stadt, Postleitzahl und Land überprüft. 4 Es werden Straße, Stadt, Postleitzahl und Land überprüft. 5 Es werden Hausname, Gebäudename, Nebengebäude, Postfach, Unternehmen, Straße, Stadt, Postleitzahl und Land überprüft.
InnerMatchScore	<p>Gibt das minimale Konfidenzniveau für die Adressenüberprüfung an. Adressen mit einem Wert im Ausgabefeld „Zuverlässigkeit“, der größer oder gleich diesem Wert ist, wird überprüft. Die Felder mit einem niedrigeren Wert werden nicht überprüft (das Ausgabefeld „Status“ enthalten ein F).</p> <p>Geben Sie einen Wert zwischen 0 und 100 an. Je höher der Wert ist, desto höher ist der notwendige Zuverlässigkeitsgrad für eine effektive Adressenüberprüfung. Der Standardwert ist 60.</p>
CompanyWeight	<p>Eine ganze Zahl von 0 bis 10, die die relative Wichtigkeit des Feldes „FirmName“ im Vergleich zu den Daten in der Datenbank „Address Now“ angibt. Dies beeinflusst das Konfidenzniveau und kann verwendet werden, um die Zuverlässigkeit so anzupassen, dass zwischen korrekten und falschen Aktualisierungen unterschieden wird. Weitere Informationen finden Sie unter Der ACR-Code auf Seite 264.</p> <p>Der Standardwert ist 1.</p>

Optionsname	Beschreibung
StreetWeight	<p>Eine ganze Zahl von 0 bis 10, die die relative Wichtigkeit des Feldes „StreetName“ im Vergleich zu den Daten in der Datenbank „Address Now“ angibt. Eine ganze Zahl von 0 bis 10, die die relative Wichtigkeit dieses Feldes im Vergleich zu anderen angibt. Weitere Informationen finden Sie unter Der ACR-Code auf Seite 264.</p> <p>Der Standardwert ist 10.</p>
CityWeight	<p>Eine ganze Zahl von 0 bis 10, die die relative Wichtigkeit des Feldes „City“ im Vergleich zu den Daten in der Datenbank „Address Now“ angibt. Eine ganze Zahl von 0 bis 10, die die relative Wichtigkeit dieses Feldes im Vergleich zu anderen angibt. Weitere Informationen finden Sie unter Der ACR-Code auf Seite 264.</p> <p>Der Standardwert ist 8.</p>
PostcodeWeight	<p>Eine ganze Zahl von 0 bis 10, die die relative Wichtigkeit des Feldes „PostalCode“ im Vergleich zu den Daten in der Datenbank „Address Now“ angibt. Eine ganze Zahl von 0 bis 10, die die relative Wichtigkeit dieses Feldes im Vergleich zu anderen angibt. Weitere Informationen finden Sie unter Der ACR-Code auf Seite 264.</p> <p>Der Standardwert ist 8.</p>
OuterMatchScoreLines	<p>Ein Wert von 0 bis 8, der die Anzahl der Adresszeilen angibt, die beim Berechnen der Punktzahl für äußere Übereinstimmung verwendet werden sollen. Der Standardwert ist 8. Weitere Informationen über die Punktzahl für die äußere Übereinstimmung finden Sie unter Punktzahl für äußere Übereinstimmung auf Seite 263.</p>

Ausgabeformatoptionen

Tabelle 39: Ausgabeformatoptionen für ValidateGlobalAddress

optionName	Beschreibung
Transliteration	<p>Gibt an, wie diakritische Zeichen in der Ausgabeadresse formatiert werden sollen. Zur Auswahl stehen:</p> <p>0 Es wird keine Transkription durchgeführt. Diakritische Zeichen bleiben wie in der Eingabe und/oder Postdatenbank angegeben unverändert. Standardeinstellung.</p> <p>1 Diakritische Zeichen werden entfernt und durch das entsprechende unmarkierte Zeichen ersetzt.</p> <p>2 Diakritische Zeichen werden in ein entsprechendes unmarkiertes Zeichen oder eine Zeichenfolge transkribiert. Dabei werden sprachspezifische Transkriptionsregeln verwendet.</p> <p>Im folgenden Beispiel werden die Auswirkungen von jeder der drei Transkriptionsoptionen auf eine schwedische Adresse dargestellt. Achten Sie auf die Unterschiede in „Västra Frölunda“.</p> <p>0</p> <p>Gustaf Wernersgata 12 S-42132 Västra Frölunda</p> <p>1</p> <p>Gustaf Wernersgata 12 S-42132 Vastra Frolunda</p> <p>2</p> <p>Gustaf Wernersgata 12 S-42132 Vaestra Froelunda</p>
FormatComponent.Department	<p>Gibt an, ob das Feld „Abteilung“ in der Ausgabe einer formatierten Adresse enthalten sein soll.</p> <p>Y Ja (Standard)</p> <p>N Nein</p>
FormatComponent.FirmName	<p>Gibt an, ob das Feld „FirmName“ in der Ausgabe einer formatierten Adresse enthalten sein soll.</p> <p>Y Ja (Standard)</p> <p>N Nein</p>

optionName	Beschreibung
FormatComponent.Building	Gibt an, ob das Feld „Building“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.SubBuilding	Gibt an, ob das Feld „SubBuilding“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.HouseNumber	Gibt an, ob das Feld „HouseNumber“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.SubStreet	Gibt an, ob das Feld „SubStreet“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.StreetName	Gibt an, ob das Feld „StreetName“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.POBox	Gibt an, ob das Feld „POBox“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein

optionName	Beschreibung
FormatComponent.SubCity	Gibt an, ob das Feld „SubCity“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.City	Gibt an, ob das Feld „City“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.USCountyName	Gibt an, ob das Feld „USCountyName“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.StateProvince	Gibt an, ob das Feld „StateProvince“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.Principality	Gibt an, ob das Feld „Principality“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.PostalCode	Gibt an, ob das Feld „PostalCode“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein

optionName	Beschreibung
FormatComponent.Plus4	Gibt an, ob das Feld „+4“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja (Standard) N Nein
FormatComponent.Country	Gibt an, ob das Feld „Country“ in der Ausgabe einer formatierten Adresse enthalten sein soll. Y Ja N Nein (Standard)

Ausgabe

Adressdatenausgabe

Tabelle 40: „ValidateGlobalAddress“-Adressdatenausgabe

Spaltenname	Format	Beschreibung
AddressLine1	Zeichenfolge [79]	Die formatierte erste Adresszeile.
AddressLine2	Zeichenfolge [79]	Die formatierte zweite Adresszeile.
AddressLine3	Zeichenfolge [79]	Die formatierte dritte Adresszeile.
AddressLine4	Zeichenfolge [79]	Die formatierte vierte Adresszeile.
AddressLine5	Zeichenfolge [79]	Die formatierte fünfte Adresszeile.

Spaltenname	Format	Beschreibung
AddressLine6	Zeichenfolge [79]	Die formatierte sechste Adresszeile.
AddressLine7	Zeichenfolge [79]	Die formatierte siebte Adresszeile.
AddressLine8	Zeichenfolge [79]	Die formatierte achte Adresszeile.
ApartmentLabel	Zeichenfolge [79]	Wohnungsbezeichner (z. B. Whng. oder APT). Beispiel: 123 E Main St. Whng. 3
ApartmentNumber	Zeichenfolge [79]	Wohnungsnummer. Beispiel: 123 E Main St. APT 3
Building	Zeichenfolge [79]	Der Name eines Gebäudes.
City	Zeichenfolge [79]	Der Ortsname.
Country	Zeichenfolge [79]	Der ISO-Code oder englische Name des Landes. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
Department	Zeichenfolge [79]	Eine Unterteilung eines französisch- oder spanischsprachigen Landes. Frankreich ist z. B. in 100 Departments unterteilt.
FirmName	Zeichenfolge [79]	Der Name eines Unternehmens. Beispiel: Pitney Bowes 4200 PARLIAMENT PL STE 600 LANHAM MD 20706-1844 USA

Spaltenname	Format	Beschreibung
HouseNumber	Zeichenfolge [79]	Hausnummer. Beispiel: 123 E Main St. Whng. 3
Latitude	Zeichenfolge [79]	Der genauste Breitengrad, der für die Adresse bestimmt werden konnte. Dies könnte ein punktueller Standort oder ein Zentroid sein. Der Genauigkeitsgrad kann mithilfe des Ausgabefeldes des Codes für das erweiterte Korrekturergebnis bestimmt werden. Weitere Informationen finden Sie unter ECR-Code auf Seite 262.
Longitude	Zeichenfolge [79]	Der genauste Längengrad, der für die Adresse bestimmt werden konnte. Dies könnte ein punktueller Standort oder ein Zentroid sein. Der Genauigkeitsgrad kann mithilfe des Ausgabefeldes des Codes für das erweiterte Korrekturergebnis bestimmt werden. Weitere Informationen finden Sie unter ECR-Code auf Seite 262.
POBox	Zeichenfolge [79]	Die Nummer des Postfachs. Sollte es sich bei der Adresse um eine Landstraßenadresse handeln, wird hier die Postfachnummer der Landstraße angezeigt.
PostalCode	Zeichenfolge [79]	Die Postleitzahl. In den USA ist dies der ZIP-Code™.
PostalCode.AddOn	Zeichenfolge [79]	Der vierstellige Zusatzteil des ZIP+4®-Codes. Im ZIP-Code™ 60655-1844 lautet der vierstellige Zusatzteil 1844. (Nur US-amerikanische Adressen.)
PostalCode.Base	Zeichenfolge [79]	Der fünfstelligen ZIP-Code™. Beispiel: 20706 (nur US-amerikanische Adressen.)
Principality	Zeichenfolge [79]	Eine Region innerhalb eines Landes. England, Schottland und Wales sind beispielsweise Fürstentümer. Dieses Feld ist in der Regel leer.
StateProvince	Zeichenfolge [79]	Die Abkürzung für das Bundesland oder den Kanton.

Spaltenname	Format	Beschreibung
StreetName	Zeichenfolge [79]	Straßenname. Beispiel: 123 E Main St. Whng. 3
StreetSuffix	Zeichenfolge [79]	Straßensuffix. Beispiel: 123 E Main St. Whng. 3
SubCity	Zeichenfolge [79]	Ein Bezirk oder Vorort. „SubCity“ wird in Ländern verwendet, in denen der Bezirk oder Stadtteil in die Adresse aufgenommen wird. Beispiel: 27 Crystal Way Bradley Stoke Bristol BS32 8GA In diesem Fall ist „Bradley Stoke“ die SubCity-Angabe.
SubStreet	Zeichenfolge [79]	Die zweite Straßenadresse, die zur Identifizierung einer Adresse verwendet wird. Unterstraßen werden in Ländern verwendet, in denen häufig zwei Straßennamen in der Adresse angegeben werden. Beispiel: 12 The Mews High Street In diesem Beispiel ist „High Street“ die Unterstraße. Mithilfe von Unterstraßen kann der Zustellungsort genau ermittelt werden. Im Beispiel könnte „The Mews“ eine kleine Straße sein, die eine weitere Straßenidentifizierung erfordert, um die Adresse genau zu finden. Also wird „High Street“ einbezogen. In diesem Fall ist „High Street“ die Hauptstraße bzw. bekannte Straße.
USCountyName	Zeichenfolge [79]	Bei US-amerikanischen Adressen ist es der Name des Bezirks, in dem sich die Adresse befindet.

Rückgabecodes

Tabelle 41: „ValidateGlobalAddress“-Rückgabecodes

Spaltenname	Format	Beschreibung
ACRCode	Zeichenfolge [79]	Der Code für das Adressenkorrekturergebnis beschreibt, welche Daten in jedem Datensatz geändert wurden. Informationen zur Bedeutung dieses Codes finden Sie unter Der ACR-Code auf Seite 264.
Confidence	Zeichenfolge [79]	Das Zuverlässigkeitsniveau, das der zurückgegebenen Adresse zugewiesen ist. Der Bereich reicht von 0 (null) bis 100. Null gibt an, dass ein Fehler vorliegt; 100 gibt an, dass das Zuverlässigkeitsniveau sehr hoch ist und die Übereinstimmungsergebnisse richtig sind. Dieser Wert ist der gleiche wie die letzten drei Stellen des Codes für das Adressenkorrekturergebnis, der als Überprüfungspunktzahl bezeichnet wird. Weitere Informationen finden Sie unter Der ACR-Code auf Seite 264.
ECRCode	Zeichenfolge [79]	Der Code des erweiterten Korrekturergebnisses beschreibt den Genauigkeitsgrad des Breiten- und Längengrads, der für eine Adresse zurückgegeben wird. Weitere Informationen finden Sie unter ECR-Code auf Seite 262.
Email1	Zeichenfolge [79]	Zusätzliche Standardisierungsinformation.
Email2	Zeichenfolge [79]	Zusätzliche Standardisierungsinformation.
OuterMatchScore	Zeichenfolge [79]	Eine Punktzahl, die die Änderungen in jeder Adresszeile angibt. Weitere Informationen finden Sie unter Punktzahl für äußere Übereinstimmung auf Seite 263.
Status	Zeichenfolge [79]	Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs. <ul style="list-style-type: none"> • null—Success • F—Failure

Spaltenname	Format	Beschreibung
Status.Code	Zeichenfolge [79]	Ursache für den Fehler, falls vorhanden. <ul style="list-style-type: none"> • UnableToValidate • ServerError • CountryNotFound
Status.Description	Zeichenfolge [79]	Beschreibung des Problems, wenn eines vorliegt. <ul style="list-style-type: none"> • Address Not Found: Dieser Wert erscheint, wenn Status.Code=UnableToValidate. • Module not licensed: Dieser Wert erscheint, wenn Status.Code=ServerError. • Could Not Identify Country: Dieser Wert erscheint, wenn Status.Code=CountryNotFound.
URL1	Zeichenfolge [79]	Zusätzliche Standardisierungsinformation.
URL2	Zeichenfolge [79]	Zusätzliche Standardisierungsinformation.
WCRCCode	Zeichenfolge [79]	Der Code des Wortkorrekturergebnisses beschreibt die in der Eingabeadresse gefundenen vulgären Wörter. Der Code besteht aus zwei Komponenten: <ul style="list-style-type: none"> • Speicherortcode: Einer der Folgenden: • AB: Gibt an, dass eine Vulgarität in der Adresse gefunden wurde. • NB: Gibt an, dass eine Vulgarität im Namen gefunden wurde. • Anzahl: Die Anzahl der vulgären Wörter, die an dem im Speicherortcode angegeben Speicherort gefunden wurden. <p>AB2 gibt z. B. an, dass zwei vulgäre Wörter in der Eingabeadresse gefunden wurden.</p>

ECR-Code

Der Code des erweiterten Korrekturergebnisses beschreibt den Genauigkeitsgrad der Breiten- und Längengradkoordinaten, der für eine Adresse zurückgegeben werden. Der Code besteht aus einem Präfix gefolgt von einem Gedankenstrich und dem Codetext.

Das Präfix beginnt immer mit „EL“ gefolgt von einer Zahl zwischen 1 und 5, die den gesamten Genauigkeitsgrad angibt:

- 5 – Punktgeocode
- 4 – Straßenzentroid
- 3 – Postleitzahlzentroid
- 2 – Ortszentroid
- 1 – Regionszentroid

Der Codetext identifiziert die Komponenten, die für den Vergleich der Adresse mit einem Geocode verwendet wurden. Beachten Sie, dass der Text aus Buchstaben und Zahlen besteht. Diese haben folgende Bedeutungen:

- P – Einrichtung/Hausnummer, Gebäude oder Postfach
- S – Straße
- T – Ort
- R – Region/Staat
- Z – Postcode
- C – Land

Es gibt nur zwei numerische Optionen im Text: 4 oder 0

- 4 – Die Komponentendaten waren verfügbar, um den Vergleich von Adresse und Geocode auszuführen.
- 0 – Die Komponentendaten waren nicht verfügbar.

Zum Beispiel: EL4-P0S4T4R4Z4C4

In diesem Beispiel gibt 0 gefolgt von einem P an, dass die Einrichtungs-/Hausnummerdaten nicht verfügbar waren, um diesen Adressenvergleich auszuführen. Alles von Straße bis Land wurde jedoch in der Zuweisung verwendet.

Punktzahl für äußere Übereinstimmung

Die Punktzahl für die äußere Übereinstimmung gibt an, in welchem Umfang „ValidateGlobalAddress“ jede Adresszeile zur Überprüfung der Adresse geändert hat. Die Punktzahl vergleicht die Adresszeilen vor der Standardisierung und nach der Überprüfung und Formatierung. Diese Punktzahl wird nur generiert, wenn Sie die Option `OuterMatchScoreLines` auf einen Wert größer als 0 einstellen.

Die Punktzahl für die äußere Übereinstimmung ähnelt der Überprüfungspunktzahl, die Teil des Adressenkorrekturergebniscodes ist (siehe [Der ACR-Code](#) auf Seite 264). Der Unterschied ist, dass die Punktzahl für die äußere Übereinstimmung jede Veränderung in einer Adresszeile einschließlich der Formatierung bewertet, wobei die Überprüfungspunktzahl nur bewertet, ob die Daten überprüft werden konnten oder nicht.

Nehmen Sie beispielsweise die folgenden Eingabeadresszeilen vor der Verarbeitung:

Adresszeile 1: Hauptstr. 5
 Adresszeile 2: berlin
 Adresszeile 3: D

Nach der Verarbeitung lauten die Adresszeilen:

Adresszeile 1: Hauptstraße 5
 Adresszeile 2: Berlin
 Adresszeile 3: 01234
 Adresszeile 4: Deutschland

Dies hat eine Überprüfungspunktzahl von 84 % und eine Punktzahl für die äußere Übereinstimmung von 23 % ergeben.

Die Überprüfungspunktzahl ist hoch, da die Adressenkomponenten vor der Überprüfung ziemlich genau waren. Der Straßenname ist gültig, mit Ausnahme der Groß- und Kleinschreibung und der Verwendung einer Abkürzung. Der Ort und das Land sind beide gültig. Die einzige falsche Sache war die Postleitzahl (in diesem Fall hat sie gefehlt). Folglich betrug die Überprüfungspunktzahl relativ hohe 84 %.

Die Punktzahl für die äußere Übereinstimmung war niedrig, da sich die Adresszeilen nach der Formatierung wesentlich von der Eingabe unterschieden haben. In diesem Fall enthielt Adresszeile 3 den Wert „D“ in der Eingabe und den Wert „01235“ in der Ausgabe. Zeile 4 war in der Eingabe leer und enthielt in der Ausgabe einen Wert. Adresszeile 1 hat sich ebenfalls geändert. Die äußere Punktzahl ist deshalb ziemlich gering.

Der ACR-Code

Der Code für das Adressenkorrekturergebnis beschreibt, welche Daten in jedem Datensatz geändert wurden. Beispiel für einen ACR:

L5-P0S0A5T1R0Z0C4-098

ACR-Codes sind aus drei Teilen zusammengesetzt:

- Überprüfungsstufe
- Komponentenstatus
- Überprüfungspunktzahl

Überprüfungsstufe

Die ersten beiden Zeichen des ACR-Codes geben den Überprüfungstypen und die Überprüfungsstufe an.

Das erste Zeichen, das immer alphabetisch ist, gibt den Überprüfungstyp an.

- **U**: Adresse kann nicht standardisiert werden
- **C**: Adresse wird in Komponentenform dargestellt
- **L**: Adresse wurde in Adresszeilen formatiert
- **R**: Adresse wurde wiederhergestellt und hat keine akzeptable Stufe erreicht

Das zweite Zeichen, das immer numerisch ist, gibt die Überprüfungsstufe an. Je höher die Stufe, desto besser fällt die Überprüfung aus. Die folgenden Stufen können erreicht werden:

- **0**: Es wurden keine Komponenten überprüft.

- **1:** Es wird nur das Land überprüft.
- **2:** Es werden Stadt und Land überprüft.
- **3:** Es werden Stadt, Postleitzahl und Land überprüft.
- **4:** Es werden Straße, Stadt, Postleitzahl und Land überprüft.
- **5:** Es werden Hausname, Gebäudename, Nebengebäude, Postfach, Unternehmen, Straße, Stadt, Postleitzahl und Land überprüft.

Komponentenstatus

Der zweite Teil des ACR-Codes gibt den Status der wesentlichen Adressenkomponenten an. Die Adressenkomponenten werden folgendermaßen identifiziert:

- Zeichen 3–4: P—Hausnummer
- Zeichen 5–6: S—Straße
- Zeichen 7–8: A—Stadtteil (Stadtgebiet)
- Zeichen 9–10: T—Ort
- Zeichen 11–12: R—Region/Bundesland/Kanton
- Zeichen 13–14: Z—Postleitzahl/ZIP-Code®
- Zeichen 15–16: C—Land

Auf jede Komponente folgt eine Nummer, die einem der folgenden Werte entspricht:

- **0:** Nicht gefunden/leer
- **1:** Mithilfe der Position in den Eingabedaten abgeleitet
- **2:** Mithilfe der Datenbank des „Address Now“-Moduls erkannt
- **3:** Mithilfe der Datenbank des „Address Now“-Moduls erkannt und auf die Standardform aktualisiert
- **4:** Mithilfe der Datenbank des „Address Now“-Moduls geprüft
- **5:** Mithilfe der Datenbank des „Address Now“-Moduls aktualisiert/korrigiert
- **6:** Mithilfe der Datenbank des „Address Now“-Moduls hinzugefügt
- **7:** Richtigerweise leer
- **8:** Mithilfe der Datenbank des „Address Now“-Moduls teilweise erkannt
- **9:** Muss korrigiert werden, um der Datenbank des „Address Now“-Moduls zu entsprechen

Überprüfungspunktzahl

Die Überprüfungspunktzahl besteht aus den Zeichen 17–19, den drei letzten Ziffern des ACR-Codes. Hierbei handelt es sich um einen Vergleich zwischen den standardisierten Daten (im Komponentenformat) und der vorgeschlagenen Übereinstimmung, die von der Datenbank des „Address Now“-Moduls zurückgegeben wurde.

Diese Punktzahl wird berechnet, indem alle Felder geprüft werden, die von der Datenbank des „Address Now“-Moduls zurückgegeben wurden, und indem sie einzeln mit den vorhandenen Komponentendaten abgeglichen werden. Die Gesamtpunktzahl wird anschließend berechnet, indem diese Einzelwerte in einer durchschnittlichen Punktzahl kombiniert werden. Dabei werden die Punktzahlgewichtungen berücksichtigt, die über das Dialogfeld der Optionen für die Adressenüberprüfung konfiguriert werden können. Beispiel:

Eingabedaten:

AddressLine1: 11 High Street
City: Anytown
Country: Großbritannien

Standardisierte Daten:

Premise: 11
Street: High Street
City: Anytown

Nach der Überprüfung können die Daten, die von der Datenbank des „Address Now“-Moduls zurückgegeben wurden, folgendermaßen aussehen:

Premise: 11
Street: High Street
City: Anytown
PostalCode: ZZ9 9ZZ

Beim Vergleich der Datenbank des „Address Now“-Moduls mit den standardisierten Daten erhalten wir Folgendes:

- Premise: 100 % Übereinstimmung
- Street: 100 % Übereinstimmung
- City: 100 % Übereinstimmung
- PostalCode: nicht verwendet, da leer bei der Eingabe

Wenn wir diese Prozentsätze kombinieren, erhalten wir eine Punktzahl von 100 %.

Ein weiteres Beispiel könnte folgendermaßen aussehen:

Eingabedaten:

AddressLine1: bergerstrasse 12
AddressLine2: munich
AddressLine3: 80124
Country: Deutschland

Standardisierte Daten:

Premise: 12
Street: Bergerstr.
City: München
PostalCode: 80124

Ausgabe der Datenbank des „Address Now“-Moduls:

Premise: 12
Street: Burgerstr.
City: München
PostalCode: 80142

Beim Vergleich der Ausgabe der Datenbank des „Address Now“-Moduls mit den standardisierten Daten erhalten wir Folgendes:

- Premise: 100 % Übereinstimmung
- Street: 90 % Übereinstimmung (der Istwert wird durch einen Textvergleich der beiden Werte bestimmt)
- City: 100 % Übereinstimmung
- PostalCode: 80 % Übereinstimmung (da die Nummern transponiert werden)

Dadurch ergibt sich eine Gesamtpunktzahl von 92 %, wenn die Punktzahlgewichtungen alle auf 1 eingestellt sind. Wenn die Punktzahlgewichtung der Postleitzahl erhöht wird, wird die Punktzahl reduziert, da der Punktzahl der Postleitzahlkomponente (80 %) eine höhere Bedeutung in der Berechnung zugewiesen wird. Wenn die Punktzahlgewichtung des Orts erhöht wird, wird die Punktzahl erhöht, da der Punktzahl der Ortskomponente (100 %) eine höhere Bedeutung zugewiesen wird.

Beispiel:

L5-P4S4A5T5R4Z4C4-098

- L weist darauf hin, dass eine Formatierung durchgeführt wurde, um die Adresszeilen zu erstellen.
- Die Überprüfungsstufe ist 5, was bedeutet, dass die höchste Stufe für die Überprüfung der Datenbank des „Address Now“-Moduls erreicht wurde.
- Alle Komponentencodes mit Ausnahme von Stadtteil (A) und Ort (T) werden auf 4 eingestellt, was bedeutet, dass sie mithilfe der Datenbank des „Address Now“-Moduls überprüft wurden.
- Der Stadtteilcode und der Ortscode werden auf 5 eingestellt, was bedeutet, dass diese Komponenten mithilfe der Datenbank des „Address Now“-Moduls korrigiert wurden.

Die gesamte Adresse hat zu 98 % mit der Datenbank des „Address Now“-Moduls übereingestimmt.

Anmerkung: Gegebenenfalls erhalten Sie den Wert „SDS“ für die Übereinstimmungspunktzahl. Der Rückgabewert SDS weist darauf hin, dass die Adresse eventuell infolge der Wiederherstellung der Adresse nicht standardisiert wurde.

Enterprise Geocoding-Modul

Enterprise Geocoding-Modul

Das Enterprise Geocoding-Modul führt Adresstandardisierung und -Geocoding sowie PLZ-Mittelpunkt-Geocoding aus. Sie können eine Adresse eingeben und Ausgaben wie geografische Koordinaten erhalten, die zu einer detaillierten Geoanalyse und Zuordnung demografischer Daten verwendet werden können. Sie können auch einen Geocode eingeben – einem durch eine Breiten-

und Längengradkoordinate dargestellten Punkt – und Adressinformationen zum bereitgestellten Geocode erhalten.

Komponenten

Das Enterprise Geocoding-Modul setzt sich aus den folgenden Schritten zusammen: Je nach Lizenz stehen Ihnen unterschiedliche Schritte zur Verfügung.

- **GeocodeAddressAUS:** Verwendet eine Adresse in Australien und gibt Koordinaten zum Breitengrad/Längengrad sowie weitere Informationen zurück.

Anmerkung: Geocode Address AUS ist veraltet. GNAF PID Location Search ist der einzige von Geocode Address AUS verwendete Schritt. Verwenden Sie für alle anderen Geocoding-Funktionen für Australien die Komponente „Geocode Address Global“.

- **GeocodeAddressGBR:** Verwendet eine Adresse in Großbritannien und gibt Koordinaten zum Breitengrad/Längengrad sowie weitere Informationen zurück.

Anmerkung: Geocode Address GBR unterstützt die Datenquelle „GBR AddressBase Plus“. Verwenden Sie Geocode Address Global für die Datenquelle „GBR Streets“ (TomTom).

- **GeocodeAddressGlobal:** Verwendet eine Adresse aus einem beliebigen unterstützten Land und gibt Koordinaten zum Breitengrad/Längengrad sowie weitere Informationen zurück. Geocode Address Global geocodiert nur Adressen aus Ländern, für die Sie eine Lizenz besitzen. Australien und Großbritannien werden nicht unterstützt.
- **Geocode Address:** Verwendet eine Adresse aus einem beliebigen der unterstützten Länder und gibt den Ortsmittelpunkt oder (für einige Länder) den PLZ-Mittelpunkt zurück. Geocode Address World kann keine Geocodierung auf Adressebene durchführen.
- **Geocode Africa:** Bietet die Geocodierung auf Straßenebene für viele afrikanische Länder. Die Komponente kann auch Orts- oder Lokaliättsmittelpunkte sowie PLZ-Mittelpunkte für ausgewählte Länder ermitteln.
- **Geocode Middle East:** Bietet die Geocodierung auf Straßenebene für viele Länder im Nahen Osten. Die Komponente kann auch Orts- oder Lokaliättsmittelpunkte ermitteln. Die Komponente „Middle East“ unterstützt sowohl englische als auch arabische Zeichensätze.
- **Geocode Latin America:** Bietet die Geocodierung auf Straßenebene für viele lateinamerikanische Länder. Die Komponente kann auch Orts- oder Lokaliättsmittelpunkte ermitteln. Für bestimmte Länder werden auch Postleitzahlen abgedeckt.
- **GeocodeUSAddress:** Verwendet eine Eingabeadresse und gibt Koordinaten zum Breitengrad/Längengrad sowie weitere Informationen zurück.
- **GNAFPIDLocationSearch:** Identifiziert die Adresse und die Koordinaten zum Breitengrad/Längengrad für einen Geocoded National Address File Persistent Identifier (G-NAF PID).
- **ReverseAPNLookup:** Dabei werden die Parzellennummer des Prüfers (APN – Assessor's Parcel Number), die Ländercodes der bundesstaatlichen Informationsverarbeitungsstandards (FIPS – Federal Information Processing Standards) sowie die FIPS-Bundesländercodes genutzt und es wird die Adresse der Parzelle zurückgegeben.

- **ReverseGeocodeUSLocation:** Verwendet einen Geocode (Koordinaten zum Breitengrad und Längengrad) als Eingabe und gibt die Adresse des Standorts zurück.
- **ReversePBKeyLookup:** Akzeptiert eine eindeutige Kennung „pbKey™“ als Eingabe und gibt alle Standardrückgaben zurück, die als Teil des Adressabgleichs verfügbar sind.

Enterprise Geocoding-Datenbanken

Die folgenden Enterprise Geocoding-Moduldatenbanken sind auf dem Spectrum™ Technology Platform-Server installiert. Manche der Datenbanken sind per Abonnement bei Pitney Bowes verfügbar und werden monatlich oder vierteljährlich aktualisiert. Andere werden vom USPS® lizenziert.

US-amerikanische Geocoding-Datenbanken (nur USA)

Diese Datenbanken enthalten die Geodaten, die notwendig sind, um Standardisierung und Geocoding für Adressen auszuführen. Sie müssen mindestens eine dieser Datenbanken installieren, um Geocoding für die USA auszuführen. Sie legen die Datenbank, die für Vergleiche verwendet werden soll, mit den Verarbeitungsoptionen fest. Enterprise Geocoding versucht, einen Vergleich mit der angegebenen Datenbank durchzuführen. Um zu verifizieren, ob der Vergleich mit der gewünschten Datenbank erfolgt, können Sie den im Ausgabefeld „StreetDataType“ zurückgegebenen Wert überprüfen.

Diese Datenbanken nutzen als GSD-Dateien bezeichnete proprietäre Dateien. Für den Vergleich des PLZ-Gebiet-Mittelpunktes enthält die Datei us.Z9 alle Mittelpunktdaten für alle Bundesstaaten. Diese Datei hat normalerweise die Erweiterung z9.

- **Centrus Enhanced Geocoding:** Diese Datenbank enthält TIGER-Daten, die von der United States Geological Survey bereitgestellt werden, und Adressdaten, die vom United States Postal Service bereitgestellt werden.
- **TomTom Geocoding:** Diese Datenbank stellt aktuellere Daten als die Centrus Enhanced Geocoding-Datenbank bereit. Sie erfordert eine zusätzliche Lizenz. Diese Daten werden von TomTom, einem Drittanbieter von Geodaten, und die Postdaten vom United States Postal Service bereitgestellt.
- **NAVTEQ Geocoding:** Diese Datenbank stellt aktuellere Daten bereit als die Centrus Enhanced Geocoding-Datenbank. Sie erfordert eine zusätzliche Lizenz. NAVTEQ-Daten werden von NAVTEQ, einem Drittanbieter von Geodaten, bereitgestellt. Weitere Informationen zu diesen Datenbanken erhalten Sie von Ihrem Vertriebsbeauftragten.
- **PLZ + 4-Mittelpunkt:** Diese Datenbank liefert nur Adressenstandardisierung und PLZ + 4-Mittelpunktvergleiche. Sie ermöglicht keine Vergleiche auf Straßenebene.

Jede Geocoding-Datenbank hat einen optionalen Statewide Intersection Index. Der Statewide Intersection Index ermöglicht eine schnelle Identifizierung von landesweiten Kreuzungen. Beispiel: Der Statewide Intersection Index ermöglicht die Datenbanksuche nach „1st and Main St, CO“ und gibt eine Liste möglicher Übereinstimmungen in Colorado schneller zurück, als wenn die gesamte Geocoding-Datenbank nach jeder Instanz der Kreuzung durchsucht würde.

US-amerikanische Punktdatenbanken (nur USA)

Punktgedatenbanken enthalten Daten, um den Mittelpunkt einer Parzelle zu suchen. Diese Datenbanken stellen eine verbesserte Geocoding-Genauigkeit für Internet-Mapping, Gebäude- und Unfallversicherung, Telekommunikationswesen, Versorger und andere Zwecke bereit.

Diese Datenbanken sind optional, aber entweder Centrus Enhanced Points oder Centrus Premium Points ist für die Umkehrsuche der Parzellennummer des Prüfers erforderlich. Diese Datenbanken werden auch separat lizenziert.

- **Centrus Points:** Diese Datenbank enthält die Daten, die notwendig sind, um den Mittelpunkt einer Parzelle oder eines Gebäudes zu suchen. Es enthält keine Parzellennummer des Prüfers oder Höhendaten.
- **Centrus Elevation:** Diese Datenbank enthält dieselben Daten wie Centrus Points plus Höhendaten.
- **Centrus Enhanced Points:** Diese Datenbank enthält dieselben Daten wie Centrus Points plus Daten für die Parzellennummer des Prüfers.
- **Centrus Premium Points:** Diese Datenbank enthält dieselben Daten wie Centrus Points plus Daten für die Parzellennummer des Prüfers und Höhendaten.
- **Centrus TomTom Points:** Die Daten in dieser Datenbank werden von TomTom, einem Drittanbieter von Geodaten, bereitgestellt.
- **Master Location Data:** Diese Datenbank stellt die beste verfügbare Adresspunktposition für jede postversandfähige und lieferfähige Adresse in den USA bereit.

Reverse Geocoding Database (nur USA)

Diese Datenbank enthält die Daten, die Sie benötigen, um eine Breitengrad/Längengrad-Position in eine Adresse zu konvertieren.

Diese Datenbank ist optional, aber für „ReverseGeocodeUS“ erforderlich. Diese Datenbank wird auch separat lizenziert.

Hilfsdateien (nur USA)

Hilfsdateien enthalten benutzerdefinierte Datensätze. Sie können Hilfsdateien verwenden, um benutzerdefinierte Daten bereitzustellen, die für den Adressabgleich und Geocode-Abgleich verwendet werden.

DPV[®]-Datenbank (nur USA)

Die Delivery Point Validation-Datenbank ermöglicht Ihnen, die Gültigkeit jeder einzelnen Postanschrift in den USA zu überprüfen. Die DPV-Datenbank wird als optionales Feature verteilt und kann installiert werden, um die Fähigkeit der Geocoding-Datenbank zum Überprüfen von Postanschriften zu verbessern. Jedes Mal, wenn eine Edition der Geocoding-Datenbank freigegeben wird, wird eine entsprechende Edition der optionalen DPV-Datenbank freigegeben. Das Datum der DPV-Datenbank muss dem Datum der Geocoding-Datenbank entsprechen, damit die DPV-Verarbeitung funktioniert. Es dürfen keine DPV-Suchen nach dem Ablaufdatum der DPV-Datenbank ausgeführt werden.

Diese Datenbank ist optional, aber für die CASS™-Verarbeitung erforderlich. Die DPV-Datenbank ist auch erforderlich, um eine PLZ + 4- und PLZ + 4-bezogene Ausgabe (DPBC, USPS-Datensatztyp usw.) zu bestimmen. Diese Datenbank wird auch separat lizenziert.

Anmerkung:

Die Postal Service-Lizenzierung verbietet die Verwendung von DPV zum Generieren von Adressen oder Adresslisten und darüber hinaus, dass die DPV-Datenbank außerhalb der USA exportiert wird.

EWS-Datenbank (nur USA)

Die EWS-Datenbank (Early Warning System) enthält Daten, die verhindern, dass Adressdatensätze wegen einer Verzögerung beim Eingang der Postdaten in der U.S. Postal-Datenbank fehlerhaft codiert werden.

Das USPS® aktualisiert wöchentlich die EWS-Datei. Anders als die DPV- und LACS^{Link}-Datenbanken muss die EWS-Datenbank nicht dasselbe Datum wie die Geocoding-Datenbank haben. Sie können die Datei „EWS.zip“ kostenlos aus dem CASS-Abschnitt der USPS® RIBBS-Website herunterladen:

<https://ribbs.usps.gov/index.cfm?page=doclist>

Beim Herunterladen der EWS-Datenbank erhalten Sie eine Datei namens OUT. Sie müssen die OUT-Datei vor der Verwendung in „EWS.txt“ umbenennen.

LACS^{Link}-Datenbank (nur USA)

Die LACS^{Link}-Datenbank ermöglicht Ihnen, Adressen zu korrigieren, die sich infolge einer ländlichen Routenadresse, die in die Straßenadresse konvertiert wird, einer Postfach-Umnummerierung oder einer Straßenadressenänderung geändert haben.

Diese Datenbank ist optional, aber für die CASS™-Verarbeitung erforderlich. Die LACS^{Link}-Datenbank ist auch im CASS-Modus erforderlich, um eine PLZ + 4- und PLZ + 4-bezogene Ausgabe (DPBC, USPS-Datensatztyp usw.) zu erhalten.

Das Datum der LACS^{Link}-Datenbank muss dem Datum der Geocoding-Datenbank entsprechen, damit die LACS^{Link}-Verarbeitung funktioniert.

Anmerkung:

Die USPS-Lizenzierung verbietet die Verwendung von LACS^{Link} zum Generieren von Adressen oder Adresslisten und darüber hinaus, dass die LACS^{Link}-Datenbank außerhalb der USA exportiert wird.

Internationale Geocoding-Datenbanken

Internationale Geocoding-Datenbanken enthalten die Geodaten, die notwendig sind, um Adressstandardisierung und Geocoding für Orte außerhalb der USA auszuführen. Jedes Land hat seine eigene Datenbank, und einige Länder haben optionale Datenbanken, die ein erweitertes Geocoding ermöglichen.

AddressBase Premium-Datenbank United Kingdom

AddressBase Premium ist eine Punktdatenbank, deren Quelle Ordnance Survey[®], Royal Mail und örtliche Behörden sind.

Die AddressBase Premium-Datenbank liefert höchste Genauigkeit, was sich in den S8-Ergebniscodes widerspiegelt. Die Datenbank enthält Objekte ohne Postadressen, wie unterteilte Liegenschaften, Kultstätten und Gemeindezentren.

Die AddressBase Premium-Datenbank basiert auf der UPRN (Unique Property Reference Number). Die UPRN ist die eindeutige Kennung, die einen festen Bezug zu einer eindeutigen Liegenschaft bietet, unabhängig von Änderungen beim Namen, Status, der Unterteilung oder Nutzung der Liegenschaft (etwa von Einzel- in Mehrfachbelegung) oder dem Abbruch der Liegenschaft. Alle historischen, alternativen und vorläufigen Adressen werden für dieselbe UPRN aufgezeichnet. Die UPRN wird mit jedem AddressBase Premium-Kandidaten zurückgegeben, außer bei Adressen in Nordirland.

Da die Datenquelle „Ordnance Survey“ keine Adressen für Nordirland enthält, wird AddressBase Premium um PLZ-Adressdaten für Nordirland von Royal Mail[®] ergänzt. Diese nordirischen Daten verfügen nur über eine Genauigkeit in Bezug auf den PLZ-Mittelpunkt (Ergebniscode S3).

Weitere Informationen zu AddressBase Premium finden Sie unter

<https://www.ordnancesurvey.co.uk/business-and-government/help-and-support/products/addressbase-premium.html> von Ordnance Survey.

CodePoint-Datenbank United Kingdom

Die CodePoint-PAF-Datenbank (Postal Address File) ermöglicht PLZ-Mittelpunkt-Geocoding. Die CodePoint-Datenbank ist für die meisten Anwendungen geeignet, darunter Adressabgleich, Gültigkeitsprüfung usw.

Die CodePoint-Datenbank hat als Quelle Royal Mail und deckt Straßenadressen für United Kingdom (Großbritannien und Nordirland) ab. Die CodePoint-Datenbank wird für das komplette Dataset statt nach Region lizenziert. Die Genauigkeit des PLZ-Mittelpunktes, die von der CodePoint-Datenbank ermöglicht wird, spiegelt sich in S3-Ergebniscodes wider.

Weitere Informationen zur Datenquelle Royal Mail finden Sie unter:

<http://www.royalmail.com>

Australia Geocoded National Address File (G-NAF)

Diese Datenbank bietet verbessertes Geocoding für australische Adressen. Hierbei handelt es sich um den einzigen verbindlichen nationalen Index von Lokalitäten, Straßen und Nummern für Australien, der mit geografischen Koordinaten validiert ist. Er enthält sowohl offiziell anerkannte ländliche und städtische Adressen als auch inoffizielle Adressen (Aliase). Postadressen und Postfächer sind darin nicht enthalten. Weil es aber für manche ländlichen Gebiete keine geeigneten ländlichen Adressdaten gibt, wurden RMB-Nummern (Roadside Mail Box), Parzellennummern sowie Block- und Abschnittsnummern in das G-NAF-Dataset eingeschlossen.

Wenn Sie diese Datenbank installieren, finden Sie zwei Unterordner:

- **GNAF123:** Enthält das Verzeichnis auf Punktebene. Dieses hat die höchste Geocoding-Genauigkeit (charakterisiert durch Zuverlässigkeitsstufe 1, 2 oder 3).
- **GNAF456:** Enthält den Rest der Adressdaten in G-NAF, der keine Geocoding-Kriterien hoher Genauigkeit erfüllt (charakterisiert durch Zuverlässigkeitsstufe 4, 5, oder 6).

Sie müssen jeden dieser beiden als separate Datenbankressourcen in der Management Console angeben.

Wir empfehlen die Verwendung beider Datenbanken, um zu überprüfen, ob Adressen vorhanden sind. GANF123 sollte hingegen nur für das Geocoding auf der Ebene von Parzellen verwendet werden. Wenn Sie keine Geocodes auf der Ebene von Parzellen benötigen, können Sie die Datenbank GANF456 zum Geocoding verwenden.

New Zealand Point Database

Die New Zealand Point Database basiert auf Postpunktdaten, die einen Punkt mit Dachgenauigkeit für jede eindeutige Straßenadresse haben. Position X und Y, die für Kandidaten aus dieser Datenbank zurückgegeben werden, haben Dachgenauigkeit.

Diese Daten werden von der Regierungsbehörde „Land Information New Zealand“ verwaltet. Diese Datenbank stellt eine monatliche Aktualisierung aus den Daten dar, die von den lokalen Bezirksämtern bereitgestellt werden.

Weitere Punktdatenbanken für internationale Länder

Es sind mehrere weitere Punktdatenbanken für internationale Länder verfügbar. Zusätzlich zu den USA, UK, Australien und Neuseeland sind die Punktdatenbanken des Enterprise Geocoding-Moduls für die folgenden Länder verfügbar:

- Andorra
- Österreich
- Belgien
- Kanada
- Tschechien
- Dänemark
- Frankreich
- Französisch-Guayana
- Deutschland
- Gibraltar
- Hongkong
- Indien
- Irland
- Japan
- Luxemburg
- Malaysia
- Martinique
- Mayotte

- Mexiko
- Monaco
- Marokko
- Niederlande
- Portugal
- Réunion
- Singapur
- Slowakei
- Spanien
- Schweden

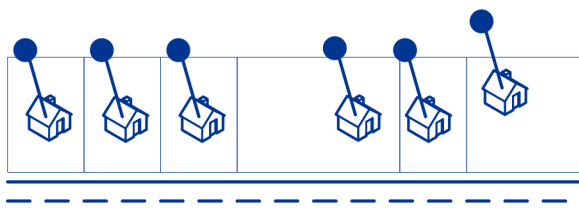
Zusätzliche Punktdatenbanken können in zukünftigen Releases verfügbar sein. Informationen zur Lizenzierung der Punktdatenbanken des Enterprise Geocoding-Moduls erhalten Sie von Ihrem Vertriebsbeauftragten.

Geocoding-Konzepte

Geocoding ist der Prozess, bei dem die Breitengrad/Längengrad-Koordinaten einer Adresse bestimmt werden. Es gibt unterschiedliche Methoden, um Geocoding für eine Adresse durchzuführen. In der Reihenfolge von der genauesten zur am wenigsten genauen sind diese Methoden:

Punktebenenvergleich

Der Punktebenenvergleich ermittelt das Zentrum eines Gebäude- oder Parzellengrundrisses. Dies ist der genaueste Typ eines Geocodes und wird in Branchen wie Internet-Mapping, Versicherung, Telekommunikation und Versorgung verwendet.



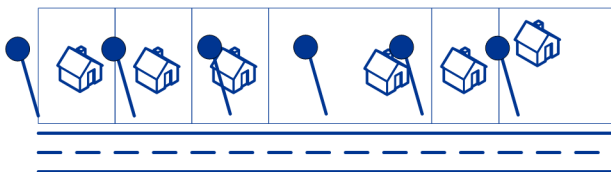
Der Mittellinienvergleich wird mit dem Punktebenenvergleich verwendet, um einen Punktebenen-Geocode an sein übergeordnetes Straßensegment zu binden. So erhalten Sie zusätzliche Daten über das übergeordnete Straßensegment, die nicht bei einem Punktebenenvergleich abgerufen werden können. Die Ausgabeinformationen beinhalten außerdem die Kompassrichtung vom Punktdaten-Geocode zur Mittellinienübereinstimmung.

Straßenvergleich

Straßenvergleich identifiziert eine ungefähre Position einer Adresse oder eines Straßensegments. Bei dem Straßenvergleich wird die Position durch Berechnung der ungefähren Position einer Hausnummer basierend auf dem Hausnummernbereich der entsprechenden Straße ermittelt. Wenn sich die Adresse beispielsweise in einem Straßensegment mit einem Hausnummernbereich von 50 bis 99 befindet, wird angenommen, dass die Hausnummer 75 in der Mitte des Straßensegments

liegen würde. Bei dieser Methode wird angenommen, dass die Adressen gleichmäßig entlang des Straßensegments verteilt liegen. Das Ergebnis ist nicht so genau wie ein Punktebenenvergleich, da es vorkommen kann, dass Adressen nicht gleichmäßig entlang eines Straßensegments verteilt sind.

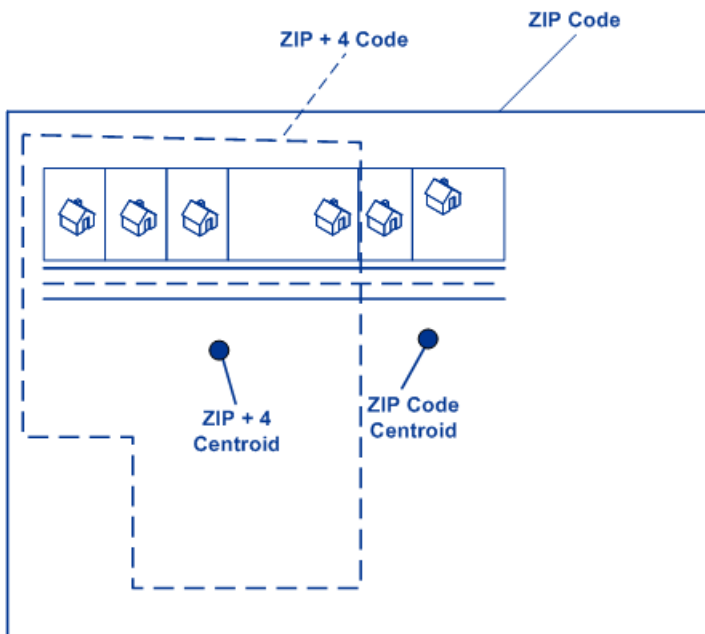
Beispielsweise zeigt das folgende Diagramm die Ergebnisse eines Vergleichs auf Straßenebene mit einem Segment mit ungleichmäßig verteilten Gebäuden. Die ersten drei Gebäude wurden einigermaßen genau geocodiert, da sie gleichmäßig verteilt liegen. Das vierte Gebäude liegt jedoch auf einer etwas größeren Parzelle als die anderen Gebäude entlang der Straße. Da bei dem Straßenebenenvergleich angenommen wird, dass die Gebäude gleichmäßig verteilt liegen, sind die Ergebnisse für das vierte, fünfte und sechste Haus nicht so präzise wie die für die ersten drei Häuser. Bei der Verwendung von Geocoding auf Punktebene wären die Ergebnisse genauer.



Zentroidvergleich

Ein PLZ-Code-Zentroid ist der Mittelpunkt eines Bereichs, der entweder durch einen PLZ-Code oder eine PLZ + 4 definiert wird, und ist der ungenaueste Typ Geocode. Ein PLZ-Zentroid ist der Mittelpunkt eines PLZ-Code-Bereichs, ein PLZ + 4-Zentroid der Mittelpunkt eines PLZ + 4-Bereichs. Da PLZ + 4 einen kleineren Bereich als ein PLZ-Code abdeckt, ist ein PLZ + 4-Zentroid genauer als ein PLZ-Code-Zentroid.

Im folgenden Diagramm wird der Zentroidvergleich veranschaulicht. Alle sechs Häuser in diesem Beispiel würden denselben Geocode aufweisen, da sie sich alle innerhalb desselben PLZ + 4-Codes befinden.



Geocoding-Vergleichsstrategien für Standorte außerhalb der USA

Das Enterprise Geocoding-Modul bietet zahlreiche Optionen zur Steuerung der Geocoding-Genauigkeit und Übereinstimmungsrate. Im Folgenden werden unterschiedliche Ansätze für Vergleiche erläutert, die Sie auf jeden Länder-Geocoder mit Ausnahme des Geocoders für die USA (GeocodeUSAddress) anwenden können. Dieser verfügt über eine andere Gruppe von Optionen.

Maximierung der Übereinstimmungsrate

Um die höchstmögliche Übereinstimmungsrate zu generieren, geben Sie Hausnummer, Straße und Ort/Lokalität nicht mit der Option „ExactMatch“ an.

Eine andere Methode, die Übereinstimmungsrate zu maximieren, ist das Festlegen von „FallbackToPostal=Y“. Das bedeutet, dass der Geocoder zum vierstelligen PLZ-Mittelpunkt zurückkehrt, wenn keine hohe Übereinstimmung auf Straßenebene möglich ist. Während dieses Szenario zu falsch-positiven Resultaten führen könnte, kann es die beste Vergleichslösung sein, wenn Sie große Datenbanken geocodieren müssen.

Sie sollten auswerten, ob der Prozentsatz falsch-positiver Resultate Auswirkungen auf Ihre Analyse hat. Um die Anzahl falsch-positiver Resultate zu verringern, ohne die Trefferrate zu beeinträchtigen, analysieren Sie die Ergebniscodes nach einer Geocoding-Sitzung und passen Sie Ihre Einstellungen entsprechend an.

Maximieren der Genauigkeit

Wenn Ihre Analyse äußerst genaue geocodierte Adressen erfordert, wählen Sie eine Strategie, bei welcher der Geocoder den maximalen Prozentsatz von Geocodes mit hoher Genauigkeit und die geringste Anzahl ungenauer Übereinstimmungen (falsch-positive Übereinstimmungen) zurückgibt. Verwenden Sie dazu die Option „ExactMatch“, damit hohe Übereinstimmungen bei allen Adresselementen übereinstimmen müssen. Definieren Sie außerdem FallbackToPostal=N.

Diese Methode kann zu einer geringeren prozentualen Übereinstimmungsrate führen, bietet aber die beste Genauigkeit.

Ausgleichen der Übereinstimmungsrate und Genauigkeit

Eventuell benötigen Sie ein Gleichgewicht zwischen der Übereinstimmungsrate und der geografischen Genauigkeit. Dementsprechend möchten Sie gegebenenfalls so viele Datensätze wie möglich geocodieren. Gleichzeitig soll die Anzahl der niedrigeren Übereinstimmungen (falsch-positive Übereinstimmungen) minimiert werden. Falsch-positive Übereinstimmungen können beispielsweise auftreten, wenn der Geocoder:

- eine Straße findet, die der eingegebenen Straße ähnelt.
- dieselbe Straße in einem anderen Ort findet (wenn kein Postleitzahlvergleich erforderlich ist).
- die Straße zwar findet, diese jedoch eine andere Hausnummer hat (wenn die Hausnummer nicht erforderlich ist).

Mithilfe der folgenden Einstellungen lässt sich ein Gleichgewicht zwischen Übereinstimmungsrate und Genauigkeit erreichen:

- **CloseMatchesOnly:** „Y“ angeben.
- **MustMatchHouseNumber:** „Y“ angeben.
- **MustMatchStreet:** „Y“ angeben.
- **FallbackToPostal—:** „N“ angeben.

Postkonzepte

Die folgenden Abschnitte enthalten Informationen zu den durch das Enterprise Geocoding-Modul verwendeten Postkonzepten.

Anmerkung: Dieser Abschnitt und die Themen „Locatable Address Conversion System“, „Delivery Point Validation“ und „Frühwarnsystem“ sind nur für das US-Geocoding relevant.

Duale Adressen

GeocodeUSAddress kann Eingaben verarbeiten, die zwei Adressen für denselben Datensatz auf derselben Adresszeile enthalten. GeocodeUSAddress kann beispielsweise die folgende Eingabeadresse verarbeiten:

```
3138 HWY 371
PO BOX 120
PRESCOTT AR 71857
```

GeocodeUSAddress erkennt keine dualen Adressen, bei denen die beiden Adressen Straßenanschriften sind. GeocodeUSAddress erkennt 135 Main St 4750 Walnut St Ste 200 beispielsweise NICHT. GeocodeUSAddress erkennt duale Adressen, bei denen die beiden Adressen denselben Straßentyp aufweisen, aber keine Straßenanschriften sind. GeocodeUSAddress erkennt beispielsweise PO BOX 12 PO BOX 2000.

Nachdem GeocodeUSAddress die duale Adresse geparkt hat, wird nach einer Übereinstimmung gesucht. GeocodeUSAddress ermittelt, welche Adresse basierend auf dem Bearbeitungsmodus für einen Vergleich priorisiert wird. Im CASS-Modus ignoriert GeocodeUSAddress die Optionen „Postfach bevorzugen“ und „Straße bevorzugen“ und versucht, basierend auf der folgenden Reihenfolge eine Übereinstimmung zu finden: Postfach, Straße, ländliche Route und allgemeine Zustellung. Im Modus mit unscharfer Übereinstimmung erkennt GeocodeUSAddress die Eingabeoption „Adresseinstellung“ (AddressPreference).

Anmerkung: GeocodeUSAddress verarbeitet keine dualen Adressen in den Modi „Genau“ und „Hoch“. GeocodeUSAddress verarbeitet keine dualen Adressen, wenn diese mehrere Zeilen aufweisen.

Locatable Address Conversion System (LACS)

Das USPS® Locatable Address Conversion System (LACS) korrigiert Adressen, die infolge der Umwandlung einer Landstraßenadresse in eine straßenähnliche Adresse, der Neunummerierung eines Postfachs oder der Änderung einer straßenähnlichen Adresse geändert wurden. Im Folgenden finden Sie einige Beispiele für LACS^{Link}-Umwandlungen:

- Umwandlung einer Landstraßenadresse in eine straßenähnliche Adresse: Alte Adresse: RR 3 Box 45 Neue Adresse: 1292 North Ridgeland Drive
- Umbenennung und Neunummerierung einer Straße: Alte Adresse: 23 Main Street Neue Adresse: 45 West First Avenue
- Neunummerierung eines Postfachs: Alte Adresse: PO Box 453 Neue Adresse: PO Box 10435

LACS^{Link} ist für die CASS-Verarbeitung erforderlich.

Delivery Point Validation

Delivery Point Validation (DPV[®]) ist eine Technologie des United States Postal Service[®] (USPS[®]), die die Genauigkeit von Adressdaten bis hin zur einzelnen Postanschrift überprüft. Durch die Verwendung von DPV[®] zur Adressenüberprüfung können Sie die Menge der unzustellbaren Post verringern und dadurch die Portokosten und andere Geschäftskosten aufgrund von ungenauen Adressinformationen reduzieren.

Anmerkung: Die Verfügbarkeit von DPV[®] gilt nur für US-amerikanische Adressen.

Ohne DPV[®] überprüft die Adressenüberprüfung nur, ob eine einzelne Adresse innerhalb eines Bereichs von gültigen Adressen für die gegebene Straße liegt. Laut USPS-Daten geht der Adressbereich der „Maple Lane“ beispielsweise von 500 bis 1000. Sie versuchen, die Adresse „610 Maple Ln“ zu validieren. Ohne DPV[®] würde diese Adresse gültig erscheinen, da sie im Bereich von 500 bis 1000 liegt. In der Realität gibt es die Adresse „610 Maple Ln“ nicht: Die Hausnummern in diesem Straßenbereich sind 608, 609, 613 und 616. Mit DPV[®]-Verarbeitung würden Sie gewarnt werden, dass „610 Maple Ln“ nicht existiert, und Sie könnten Korrekturmaßnahmen ergreifen.

DPV[®] bietet zudem eindeutige Adressattribute, um die Erstellung mehr zielgerichteter Adresslisten zu unterstützen. Beispielsweise kann DPV[®] anzeigen, ob eine Adresse leerstehend ist, und eine Commercial Mail Receiving Agency (CMRA) oder eine Private Mail Box identifizieren.

Obwohl DPV[®] die Genauigkeit einer vorhandenen Adresse überprüfen kann, können Sie DPV[®] nicht verwenden, um Adresslisten zu erstellen. Sie können beispielsweise überprüfen, ob „123 Elm Street Apartment 6“ existiert, Sie können allerdings nicht fragen, ob es an derselben Straßenanschrift ein „Apartment 7“ gibt. Zur Verhinderung der Erstellung von Adresslisten enthält die DPV[®]-Datenbank falsch-positive Datensätze. Diese Datensätze sind künstlich erzeugte Adressen in einer falsch-positiven Tabelle. Für jede negative Antwort, die auf eine DPV[®]-Abfrage folgt, erfolgt eine Abfrage in der falsch-positiven Tabelle. Eine Übereinstimmung mit dieser Tabelle stoppt die DPV[®]-Verarbeitung.

Early Warning System (EWS)

Das Early Warning System (EWS) bietet aktuelle Adressinformationen für neue und kürzlich geänderte Adressen, die noch nicht in der monatlichen USPS-Datenbank aktualisiert wurden. Das Frühwarnsystem verhindert, dass Adressdatensätze falsch codiert werden, wenn Postdaten zu spät die USPS[®]-Datenbanken erreichen.

Je älter die US- Postdatenbank ist, desto größer ist die Wahrscheinlichkeit, dass falsch codierte Adressen auftreten. Wenn eine gültige Adresse falsch codiert wird, da die übereinstimmende Adresse in der US- Postdatenbank nicht korrekt ist, resultiert daraus eine defekte Adresse.

EWS-Daten bestehen aus unvollständigen Adressinformationen und sind auf ZIP-Code™, Straßennamen, Präfixrichtung, Postfixrichtung und einen Suffix beschränkt. Damit sich ein Adressdatensatz für EWS eignet, darf die Adresse nicht in der aktuellsten monatlichen Version der US-Postdatenbank enthalten sein.

Das USPS® aktualisiert wöchentlich die EWS-Datei. Sie können die EWS-Datei von der USPS®-Website unter https://ribbs.usps.gov/cassmass/documents/tech_guides/ herunterladen.

Geocode Address Global

Informationen zur Verwendung der API für den Zugriff auf Geocode Address Global finden Sie in den Geocoding-Handbüchern.

GeocodeAddressWorld

Geocode Address verwendet eine Adresse aus einem beliebigen der unterstützten Länder und gibt den Ortsmittelpunkt oder (für einige Länder) den PLZ-Mittelpunkt zurück. GeocodeAddressWorld kann kein Geocoding auf Straßenanschriftsebene durchführen. Verwenden Sie GeocodeAddressGlobal, wenn Sie Geocoding auf Adressebene durchführen müssen.

GeocodeAddressWorld wird in der Regel als Ersatzgeocoder verwendet, um Länder abzudecken, für die ein Geocode Address Global-Land nicht verfügbar ist. Beispiel: Sie haben den Geocoder für Australien lizenziert, da Sie primär daran interessiert sind, australische Adressen zu geocodieren. Ihre Daten enthalten jedoch möglicherweise Datensätze mit Standorten außerhalb von Australien. In diesem Fall könnten Sie mithilfe von GeocodeAddressWorld Mittelpunktgeocodes für Standorte außerhalb von Australien bereitstellen und gleichzeitig mit den Geocodern für Australien genauere Geocodes für australische Adressen bereitstellen. In anderen Datenflüssen können Sie auswählen, ob GeocodeAddressWorld als Geocoder für den ersten Durchlauf verwendet werden soll, und die Ergebnisse anschließend an landesspezifische Geocoder weiterleiten. Welche Strategie am besten ist, hängt von Ihrem Geschäftsfall und der Beschaffenheit Ihrer Adressdaten ab.




GeocodeAddressWorld ist ein optionaler Teil des Enterprise Geocoding-Moduls. Weitere Informationen zum Enterprise Geocoding-Modul finden Sie unter [Enterprise Geocoding-Modul](#) auf Seite 267.

Hinzufügen einer Datenbankressource für das Enterprise Geocoding-Modul für den Welt-Geocoder

Bei jeder Installation einer neuen Datenbankressource oder Änderung einer vorhandenen Datenbankressource müssen Sie sie in der Management Console definieren, damit sie auf Ihrem

System verfügbar wird. In dieser Prozedur wird beschrieben, wie Sie eine Datenbankressource für das Enterprise Geocoding-Modul für den Welt-Geocoder hinzufügen oder ändern.

So erstellen Sie eine Datenbankressource „Geocode Address World“:

1. Installieren Sie die Datenbankdateien auf Ihrem System, wenn dies nicht bereits erfolgt ist. Anleitungen zur Installation von Datenbanken finden Sie im *Spectrum™ Technology Platform Installationshandbuch*.
2. Wählen Sie in der Management Console unter **Ressourcen** die Option **Spectrum-Datenbanken** aus.
3. Klicken Sie auf die Schaltfläche „Hinzufügen“ , um eine neue Datenbank zu erstellen, oder wählen Sie eine vorhandene Datenbankressource aus, und klicken Sie anschließend auf die Schaltfläche „Bearbeiten“ , um sie zu ändern. Sie können auch eine neue Datenbankressource erstellen, indem Sie eine vorhandene Ressource durch einen Klick auf die Schaltfläche „Kopieren“  kopieren.
4. Wenn Sie eine neue Datenbank erstellen, müssen Sie einen Namen für die Datenbankressource in das Feld **Name** eingeben. Sie können den Namen frei wählen. Wenn Sie eine neue Datenbank erstellen, indem Sie eine vorhandene kopieren, müssen Sie den Standardnamen nach Bedarf ändern. Sie können den Namen einer vorhandenen Datenbankressource nicht ändern, da alle Dienste oder Aufträge fehlschlagen würden, die mit ihrem ursprünglichen Namen auf die Datenbankressource verweisen.
5. Geben Sie im Feld **Poolgröße** die maximale Anzahl der gleichzeitigen Anforderungen ein, die von dieser Datenbank verarbeitet werden sollen.

Die optimale Poolgröße ist je nach Modul unterschiedlich. Im Allgemeinen erzielen Sie die besten Ergebnisse, wenn Sie die Poolgröße auf einen Wert zwischen der Hälfte und der doppelten Anzahl der Server-CPU's festlegen. Für die meisten Module entspricht die optimale Poolgröße der Anzahl der CPU's. Wenn Ihr Server beispielsweise über vier CPU's verfügt, können Sie mit einer Poolgröße zwischen 2 (die Hälfte der vorhandenen CPU's) und 8 (das Doppelte der vorhandenen CPU's) experimentieren, wobei die optimale Größe möglicherweise bei 4 liegt (Anzahl der vorhandenen CPU's).

6. Wählen Sie im Feld **Modul** den Eintrag „InternationalGeocoder World“ aus.
7. Wählen Sie im Feld **Typ** den Eintrag „Geocode Address Global“ aus.
Wenn Sie Ihre .SPD-Dateien entpackt und die Inhalte im Ordner `\server\app\dataimport` platziert haben, fügt Spectrum sie automatisch zum Ordner `\repository\datastorage` hinzu. Auf dem Bildschirm „Datenbank hinzufügen“ wird eine Liste von Datasets angezeigt.
8. Wählen Sie die Datasets aus, die Sie der Datenbank als Ressource hinzufügen möchten. Verwenden Sie das Textfeld „Filter“, um in einer langen Liste nach einem Dataset zu suchen.
9. Speichern Sie die Datenbank.
10. Wenn noch Enterprise Designer-Sitzungen geöffnet sind, klicken Sie auf die Schaltfläche „Aktualisieren“, um den neuen Schritt anzuzeigen.

Geocode-Genauigkeit

GeocodeAddressWorld liefert automatisch den bestmöglichen Geocode basierend auf den von Ihnen bei der Eingabe angegebenen Daten. Wenn Sie einen Ort und eine gültige Postleitzahl angeben, erhalten Sie einen PLZ-Mittelpunkt. Wenn Sie einen Ort und eine ungültige Postleitzahl oder einen Ort ohne Postleitzahl angeben, gibt GeocodeAddressWorld den geografischen Mittelpunkt des Orts zurück.

Weitere Informationen finden Sie unter [Geografisches Geocoding](#) auf Seite 282 und [Postalisches Geocoding](#) auf Seite 281.

Über Management Console können Sie zwischen dem geografischen und postalischen Geocoding wählen. Sie können außerdem die Option „Größte Übereinstimmung“ auswählen. Wenn sowohl das geografische als auch das postalische Geocoding möglich sind, gibt die Auswahl „Größte Übereinstimmung“ einen geografischen Kandidaten mit hoher Übereinstimmung zurück, wenn das geografische Ergebnis auf Ortsebene oder höher zurückgegeben wird (d. h. ein G3- oder G4-Ergebniscode). Wenn das geografische Ergebnis eine geringere Genauigkeit als die Ortsebene aufweist (d. h. ein G1- oder G2-Ergebniscode), gibt „Größte Übereinstimmung“ eventuell ein postalisches Ergebnis (Z1-Ergebnis) zurück. Wenn kein postalisches Ergebnis verfügbar ist, wird der beste verfügbare geografische Kandidat zurückgegeben.

Weitere Informationen finden Sie unter [Ergebniscodes für geografisches Geocoding](#) und [Ergebniscodes beim postalischen Geocoding](#).

Postalisches Geocoding

Geocode Address Geocode Address World kann das Geocoding zu einem PLZ-Mittelpunkt durchführen, wenn für das Land Postleitzahldaten verfügbar sind. Postleitzahldaten können von jeder der Datenquellen stammen (TomTom, GeoNames oder Pitney Bowes). Einen Überblick über die Abdeckung postalischer Daten von Geocode Address Geocode Address World finden Sie unter [Abdeckung postalischer Daten von Ländern](#) auf Seite 304. Je nach Land kann das postalische Geocoding genauere Ergebnisse liefern als das geografische Geocoding.

Das Geocoding auf postalischer Ebene ist möglich, wenn diese Bedingungen erfüllt werden:

- Ihre Eingabeadresse besteht aus einer gültigen Postleitzahl.
- Die Datenquelle enthält Postleitzahldaten für das Land. Nicht jedes Land verfügt über Postleitzahldaten.

Geocode Address Geocode Address World kann mehrere hohe Übereinstimmungen beim postalischen Geocoding zurückgeben. Die Postleitzahl 12180 beispielsweise stimmt mit Troy NY überein, aber es gibt sie auch in einigen anderen Ländern. Wenn die Eingabe nur aus der Postleitzahl besteht, werden alle diese Kandidaten als hohe Übereinstimmungen zurückgegeben.

Wenn die Eingabe geografische Adresselemente enthält (z. B. Land, Staat, Region oder Ortsname), ist Geocode Address Geocode Address World in der Lage, diese Informationen für eine genauere einzelne hohe Übereinstimmung zu nutzen. Wenn Sie geografische Adressdaten verwenden möchten, um die Ergebnisse des postalischen Geocoding zu verfeinern, beachten Sie Folgendes:

Anmerkung: Verschiedene Länder leiten ihre postalischen Daten entweder von TomTom, GeoNames oder Pitney Bowes ab. Daher variieren die verfügbaren geografischen Daten in den postalischen Datenquellen je nach Land. Der Ortsname (City) beispielsweise ist ein Gewichtungsfaktor für hohe Übereinstimmungen bei Ländern, die GeoNames als postalische Datenquelle verwenden. Der Ortsname wird jedoch bei Ländern ignoriert, die TomTom als postalische Datenquelle nutzen. Weitere Informationen über die geografischen Daten der Datenquellen TomTom, GeoNames und Pitney Bowes finden Sie unter [Datenquellen und Abdeckung](#) auf Seite 285.

Postalisches Geocoding mit geografischen Daten

In diesem Beispiel für postalisches Geocoding enthält die Eingabeadresse die gültige Postleitzahl 41012 und die Provinz (StateProvince) Emilia Romagna. Es ist eine Straßenanschrift angegeben, die aber beim postalischen Geocoding ignoriert wird.

Fornaci 40
Emilia Romagna
41012

Da die postalische Datenquelle TomTom StateProvince für Italien enthält, wird die Provinz Emilia Romagna bei der Auswertung hoher Übereinstimmungen berücksichtigt. Daher wird Emilia Romagna, Italien, mit der entsprechenden Postleitzahl 41012 als einzelne hohe Übereinstimmung mit einem Ergebniscode Z1 zurückgegeben. Kandidaten mit der Postleitzahl 41012 aus anderen Ländern werden eventuell als Kandidaten mit geringer Übereinstimmung zurückgegeben. Würden keine Landesdaten oder StateProvince eingegeben, würde Geocode Address Geocode Address World mehrere hohe Übereinstimmungen zurückgeben, da die fünfstellige Postleitzahl 41012 auch in einigen anderen Ländern vorkommt.

Anmerkung: Die geografischen Daten müssen in der postalischen Datenquelle vorhanden sein, damit Ergebnisse des postalischen Geocoding verfeinert werden können. Die postalische Datenquelle TomTom enthält für Italien beispielsweise keinen Ort bzw. keine Stadt (City). Wenn Sie also den Ort Carpi mit der Postleitzahl 41012 eingeben, ignoriert Geocode Address Geocode Address World den Ortsnamen und gibt mehrere hohe Übereinstimmungen für die Postleitzahl 41012 zurück (es sei denn, Sie haben außerdem ITA als Ländernamen angegeben). Weitere Informationen über die geografischen Daten der Datenquellen TomTom, GeoNames und Pitney Bowes finden Sie unter [Datenquellen und Abdeckung](#) auf Seite 285.

Geografisches Geocoding

Geocode Address World kann ein Geocoding auf den Mittelpunkt einer Verwaltungseinheit (z. B. Stadt oder Dorf) durchführen.

Geocode Address World kann ein Geocoding auf geografischer Ebene durchführen, wenn die folgenden Bedingungen erfüllt sind:

- Ihre Eingabeadressen enthalten genaue geografische Informationen ohne gültigen Inhalt in Bezug auf die Postleitzahl der Adresse in der Eingabe. Wenn die betreffende Adresse eine gültige Postleitzahleneingabe enthält, versucht der Geocode Address World, ein postalisches Geocoding durchzuführen.
- Die Datenquelle enthält Informationen auf geografischer Ebene für das Land. Geografische Informationen können aus einer beliebigen Datenquelle stammen (TomTom, GeoNames oder Pitney Bowes).
- Ländernamen oder ISO-Ländercodes sind nicht erforderlich. Wenn sie enthalten sind, müssen sie jedoch verglichen werden. Das Einschließen des Ländernamens kann zu besseren hohen Übereinstimmungen führen.

Geografisches Geocoding von City

In dem folgenden Beispiel enthält die Eingabeadresse den Ort (City) Vaihingen an der Enz. Das Land ist in diesem Beispiel nicht angegeben. Zum Zwecke des geografischen Geocodings werden die Informationen zur Straßenanschrift (Straßennahme und -nummer) ignoriert.

Muldenweg 2
Vaihingen an der Enz

Geocode Address World gibt einen Kandidaten mit einer hohen Übereinstimmung (G3) zurück. Obwohl das Land nicht angegeben wurde, identifiziert Geocode Address World eine hohe Übereinstimmung in Deutschland (DEU).

StateProvince: Baden-Württemberg
County: Ludwigsburg
City: Vaihingen an der Enz
Land: DEU
Ergebniscode: G3
X: 8,95948
Y: 48,930059

Geografisches Geocoding mit gängigem Ortsnamen

In dem folgenden Beispiel enthält die Eingabeadresse den Ort (City) Venice. Dieser Ortsname tritt in einigen Ländern auf, das Land wurde bei der Eingabe aber nicht angegeben.

St Marks Plaza
Venice

Geocode Address World wählt Venice, Italien als Kandidaten mit hoher Übereinstimmung aus, da es eine große Bevölkerungszahl (ca. 270.000) hat und die Verwaltungshauptstadt der Region Veneto in Italien ist. Möglicherweise werden für Orte mit dem Namen Venice in anderen Ländern auch einige geringe Übereinstimmungen zurückgegeben. Der Kandidat mit der hohen Übereinstimmung für Venice, ITA lautet:

StateProvince: Veneto
 County: Venezia
 City: Venice
 Land: ITA
 Ergebniscode: G3
 X: 12,33878
 Y: 45,43434

Geografisches Geocoding mit Abkürzung für Bundesland/Kanton

In dem folgenden Beispiel enthält die Eingabeadresse den Ortsnamen Rome und GA, welches die Abkürzung für den Bundesstaat Georgia in den USA ist. Unter [Abkürzungen für Bundesstaaten/-länder oder Provinzen/Kantone](#) auf Seite 321 finden Sie die Länder, bei denen Abkürzungen für Bundesländer/Kantone erkannt werden. Da die Abkürzung für das Bundesland/Kanton verwendet wird, muss der Name des Landes nicht angegeben werden.

Rome, GA

Geocode Address World prüft das StateProvince und gibt eine hohe Übereinstimmung für Rome, Georgia USA, zurück. Obwohl Rome (Italien) eine viel größere Stadt und die Hauptstadt Italiens ist, wird es als Kandidat mit geringer Übereinstimmung zurückgegeben, da StateProvince (GA) bei der Eingabe angegeben wurde

StateProvince: Georgia
 County: Floyd
 City: Rome
 Land: USA
 Ergebniscode: G3
 X: -85,16467
 Y: 34,25704

Geografisches Geocoding von Locality

In dem folgenden Beispiel enthält die Eingabeadresse die Lokalität Altamira und die Abkürzung GRO für die Provinz. Da Geocode Address World die Abkürzung GRO für das Bundesland/Kanton erkennt, wird der Ländername nicht benötigt.

City: Altamira
 StateProvince: GRO

In dem folgenden Beispiel gibt Geocode Address World eine hohe Übereinstimmung mit der (Locality) von Altamira zurück, selbst dann, wenn Altamira als City eingegeben wurde. Das (StateProvince) GRO wird ebenfalls zurückgegeben. Wenn Guerrero als StateProvince eingegeben wurde, wird Guerrero zurückgegeben.

StateProvince: GRO
 City: ACAPULCO DE JUÁREZ
 Locality: ALTAMIRA

Land: MEX
Ergebniscode: G4
X: 99,87984
Y: 16,87637

Die Adresseingabe kann in separate Eingabefelder formatiert werden oder die Eingabe kann unformatiert sein (einzeilige Eingabe). Informationen zum Geocoding einer unformatierten Eingabe finden Sie unter [Einzeilige Eingabe](#) auf Seite 323.

Geografische Gebiete

In jedem Land gibt es Verwaltungseinheiten und viele dieser Verwaltungsbereiche werden in Adressen verwendet. Geocode Address World Gibt vier AreaNames an, von denen jeder einer Verwaltungseinheit entspricht. Die Benennung und Hierarchie von Verwaltungseinheiten variiert je nach Land.

- Lokalität
- Ort
- Bezirk
- Bundesland/Kanton

Datenquellen und Abdeckung

Geocode Address Geocode Address World verwendet mehrere Datenquellen, um seine umfassenden, weltweiten Adressdatenbanken aufzubauen. Wenn eine Eingabeadresse nicht mithilfe einer dieser Datenquellen gefunden werden kann, verwendet Geocode Address World Geocode Address World eine der anderen Datenquellen. Der beste verfügbare Kandidat wird zurückgegeben.

Diese Datenquellen (für geografische und postalische Daten) werden in der aufgeführten Reihenfolge verwendet:

- TomTom-Daten
- GeoNames-Daten
- Pitney Bowes World-Daten

Geocode Address Geocode Address World ist je nach Kontinent in sechs Datenbanken partitioniert. Die geografischen und postalischen Daten sind in jedes Adresswörterbuch integriert, um sowohl das geografische Geocoding als auch das postalische Geocoding zu unterstützen.

- Afrika
- Asien
- Europa
- Nordamerika
- Ozeanien
- Südamerika

Eine Beschreibung und Beispiele zum geografischen Geocoding finden Sie unter **Geografisches Geocoding** auf Seite 282. Eine Beschreibung und Beispiele zum postalischen Geocoding finden Sie unter **Postalisches Geocoding** auf Seite 281.

Die postalischen Quelldaten können auf den geografischen Inhalt zugreifen, mithilfe dessen postalische Ergebnisse verfeinert werden können. Das bedeutet, dass hohe Übereinstimmungen mithilfe geografischer Informationen (Ländername und Verwaltungseinheiten) ausgewertet werden können, wenn dieselbe Postleitzahl in unterschiedlichen Ländern ermittelt wird.

Je nach Quelle der postalischen Daten sind die folgenden geografischen Informationen verfügbar, um postalische Ergebnisse zu verfeinern:

- TomTom-Quelle: Land, StateProvince
- GeoNames-Quelle: Land, StateProvince und City
- Pitney Bowes World-Quelle: Land, StateProvince, County, City und Locality

Anmerkung: Das Geocode Address Geocode Address World Dataset enthält lizenzierte Daten des GeoNames-Projekts (<http://www.geonames.org>), die unter den Bedingungen der Creative Commons Attribution License („Attribution License“) bereitgestellt werden. Die Attribution License können Sie unter <http://creativecommons.org/licenses/by/3.0/legalcode> einsehen. Ihre Nutzung der GeoNames-Daten (wie im Spectrum-Benutzerhandbuch beschrieben) unterliegt den Bedingungen der Attribution License. Bei Konflikten zwischen Ihrer Vereinbarung mit PBSI und der Attribution License hat die Attribution License lediglich bezüglich der Nutzung von GeoNames-Daten Vorrang.

Länderabdeckung

Geocode Address Geocode Address World deckt die meisten Länder auf der Welt ab. Die Genauigkeit und der Umfang der Abdeckung variiert je nach Qualität der verfügbaren Datenquelle. Bei einigen Ländern werden auch Postleitzahlendaten eingeschlossen, während bei anderen Ländern nur eine geografische Abdeckung gegeben ist.

Unter **Datenquellen und Abdeckung** auf Seite 285 erhalten Sie Informationen zu TomTom, GeoNames und geografischen und postalischen Datenquellen von Pitney Bowes.

Eine vollständige Liste der geografischen Abdeckung nach Land finden Sie unter **Abdeckung geografischer Länderdaten** auf Seite 286. Unter **Abdeckung postalischer Daten von Ländern** auf Seite 304 finden Sie Informationen zur postalischen Abdeckung nach Land.

Abdeckung geografischer Länderdaten

Tabelle 42: Ländernamen und Abdeckung geografischer Daten

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
AFGHANISTAN	AFG	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
ÅLANDINSELN	ALA	GeoNames	2011,07
ALBANIEN	ALB	TomTom	2011,06
ALGERIEN	DZA	GeoNames	2011,07
AMERIKANISCH-SAMOA	ASM	GeoNames	2011,07
ANDORRA	AND	TomTom	2011,06
ANGOLA	AGO	TomTom	2011,06
ANGUILLA	AIA	GeoNames	2011,07
ANTARKTIKA	ATA	GeoNames	2011,07
ANTIGUA UND BARBUDA	ATG	GeoNames	2011,07
ARGENTINIEN	ARG	TomTom	2011,06
ARMENIEN	ARM	GeoNames	2011,07
ARUBA	ABW	GeoNames	2011,07
AUSTRALIEN	AUS	GeoNames	2011,07
ÖSTERREICH	AUT	TomTom	2011,06

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
ASERBAIDSCHAN	AZE	GeoNames	2011,07
BAHAMAS	BHS	GeoNames	2011,07
BAHRAIN	BHR	TomTom	2011,06
BANGLADESCH	BGD	GeoNames	2011,07
BARBADOS	BRB	GeoNames	2011,07
WEISSRUSSLAND	BLR	TomTom	2011,06
BELGIEN	BEL	TomTom	2011,06
BELIZE	BLZ	GeoNames	2011,07
BENIN	BEN	TomTom	2011,06
BERMUDA	BMU	GeoNames	2011,07
BHUTAN	BTN	GeoNames	2011,07
BOLIVIEN	BOL	GeoNames	2011,07
BONAIRE, SINT EUSTATIUS UND SABA	BES	GeoNames	2011,07
BOSNIEN UND HERZEGOWINA	BIH	TomTom	2011,06

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
BOTSWANA	BWA	TomTom	2011,06
BOUVET-INSEL	BVT	GeoNames	2011,07
BRASILIEN	BRA	TomTom	2011,06
BRITISCHES TERRITORIUM IM INDISCHEN OZEAN	IOT	GeoNames	2011,07
BRUNEI DARUSSALAM	BRN	TomTom	2011,06
BULGARIEN	BGR	TomTom	2011,06
BURKINA FASO	BFA	TomTom	2011,06
BURUNDI	BDI	GeoNames	2011,07
KAMBODSCHA	KHM	GeoNames	2011,07
KAMERUN	CMR	TomTom	2011,06
KANADA	CAN	TomTom	2011,06
KAP VERDE	CPV	GeoNames	2011,07
CAYMANINSELN	CYM	GeoNames	2011,07
ZENTRALAFRIKANISCHE REPUBLIK	CAF	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
TSCHAD	TCD	GeoNames	2011,07
CHILE	CHL	TomTom	2011,06
CHINA	CHN	GeoNames	2011,07
WEIHNACHTSINSEL	CXR	GeoNames	2011,07
COCOSINSELN	CCK	GeoNames	2011,07
KOLUMBIEN	COL	GeoNames	2011,07
KOMOREN	COM	GeoNames	2011,07
KONGO	COG	TomTom	2011,06
DEMOKRATISCHE REPUBLIK KONGO	COD	TomTom	2011,06
COOK-INSELN	COK	GeoNames	2011,07
COSTA RICA	CRI	GeoNames	2011,07
ELFENBEINKÜSTE	CIV	GeoNames	2011,07
KROATIEN (LOKALE BEZEICHNUNG: HRVATSKA)	HRV	TomTom	2011,06
KUBA	CUB	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
CURAÇAO	CUW	GeoNames	2011,07
ZYPERN	CYP	GeoNames	2011,07
TSCHECHIEN	CZE	TomTom	2011,06
DÄNEMARK	DNK	GeoNames	2011,07
DSCHIBUTI	DJI	GeoNames	2011,07
DOMINICA	DMA	GeoNames	2011,07
DOMINIKANISCHE REPUBLIK	DOM	GeoNames	2011,07
ECUADOR	ECU	GeoNames	2011,07
ÄGYPTEN	EGY	TomTom	2011,06
EL SALVADOR	SLV	GeoNames	2011,07
ÄQUATORIALGUINEA	GNQ	GeoNames	2011,07
ERITREA	ERI	GeoNames	2011,07
ESTLAND	EST	TomTom	2011,06
ÄTHIOPIEN	ETH	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
FALKLANDINSELN (MALWINEN)	FLK	GeoNames	2011,07
FÄRÖER-INSELN	FRO	GeoNames	2011,07
FIDSCHI	FJI	GeoNames	2011,07
FINNLAND	FIN	TomTom	2011,06
FRANKREICH	FRA	TomTom	2011,06
FRANZÖSISCH-GUAYANA	GUF	TomTom	2011,06
FRANZÖSISCH-POLYNESIEN	PYF	GeoNames	2011,07
FRANZÖSISCHE GEBIETE IM SÜDLICHEN INDISCHEN OZEAN	ATF	GeoNames	2011,07
GABUN	GAB	TomTom	2011,06
GAMBIA	GMB	GeoNames	2011,07
GEORGIEN	GEO	GeoNames	2011,07
DEUTSCHLAND	DEU	TomTom	2011,06
GHANA	GHA	TomTom	2011,06
GIBRALTAR	GIB	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
GRIECHENLAND	GRC	TomTom	2011,06
GRÖNLAND	GRL	GeoNames	2011,07
GRENADA	GRD	GeoNames	2011,07
GUADELOUPE	GLP	TomTom	2011,06
GUAM	GUM	GeoNames	2011,07
GUATEMALA	GTM	GeoNames	2011,07
GUERNSEY	GGY	GeoNames	2011,07
GUINEA	GIN	GeoNames	2011,07
GUINEA-BISSAU	GNB	GeoNames	2011,07
GUYANA	GUY	GeoNames	2011,07
HAITI	HTI	GeoNames	2011,07
HEARD UND MCDONALDINSELN	HMD	GeoNames	2011,07
HONDURAS	HND	GeoNames	2011,07
HONGKONG	HKG	TomTom	2011,06

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
UNGARN	HUN	TomTom	2011,06
ISLAND	ISL	GeoNames	2011,07
INDIEN	IND	GeoNames	2011,07
INDONESIEN	IDN	TomTom	2011,06
IRAN (ISLAMISCHE REPUBLIK)	IRN	GeoNames	2011,07
IRAK	IRQ	GeoNames	2011,07
IRLAND	IRL	TomTom	2011,06
ISLE OF MAN	IMN	GeoNames	2011,07
ISRAEL	ISR	GeoNames	2011,07
ITALIEN	ITA	TomTom	2011,06
JAMAICA	JAM	GeoNames	2011,07
JAPAN	JPN	GeoNames	2011,07
JERSEY	JEY	GeoNames	2011,07
JORDANIEN	JOR	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
KASACHSTAN	KAZ	GeoNames	2011,07
KENIA	KEN	TomTom	2011,06
KIRIBATI	KIR	GeoNames	2011,07
DEMOKRATISCHE VOLKSREPUBLIK KOREA	PRK	GeoNames	2011,07
REPUBLIK KOREA	KOR	GeoNames	2011,07
KUWAIT	KWT	TomTom	2011,06
KIRGISISTAN	KGZ	GeoNames	2011,07
VOLKSDEMOKRATISCHE REPUBLIK LAOS	LAO	GeoNames	2011,07
LETTLAND	LVA	TomTom	2011,06
LIBANON	LBN	GeoNames	2011,07
LESOTHO	LSO	TomTom	2011,06
LIBERIA	LBR	GeoNames	2011,07
LIBYSCH-ARABISCHE DSCHAMAHIRIJA	LBY	GeoNames	2011,07
LIECHTENSTEIN	LIE	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
LITAUEN	LTU	TomTom	2011,06
LUXEMBURG	LUX	TomTom	2011,06
MACAO	MAC	TomTom	2011,06
EHEMALIGE JUGOSLAWISCHE REPUBLIK MAZEDONIEN	MKD	TomTom	2011,06
MADAGASKAR	MDG	GeoNames	2011,07
MALAWI	MWI	TomTom	2011,06
MALAYSIA	MYS	TomTom	2011,06
MALEDIVEN	MDV	GeoNames	2011,07
MALI	MLI	TomTom	2011,06
MALTA	MLT	TomTom	2011,06
MARSHALLINSELN	MHL	GeoNames	2011,07
MARTINIQUE	MTQ	GeoNames	2011,07
MAURETANIEN	MRT	TomTom	2011,06
MAURITIUS	MUS	TomTom	2011,06

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
MAYOTTE	MYT	GeoNames	2011,07
MEXIKO	MEX	TomTom	2011,06
FÖDERIERTE STAATEN VON MIKRONESIEN	FSM	GeoNames	2011,07
REPUBLIK MOLDAU	MDA	TomTom	2011,06
MONACO	MCO	GeoNames	2011,07
MONGOLEI	MNG	GeoNames	2011,07
MONTENEGRO	MNE	TomTom	2011,06
MONTSERRAT	MSR	GeoNames	2011,07
MAROKKO	MAR	TomTom	2011,06
MOSAMBIK	MOZ	TomTom	2011,06
MYANMAR	MMR	GeoNames	2011,07
NAMIBIA	NAM	GeoNames	2011,07
NAURU	NRU	GeoNames	2011,07
NEPAL	NPL	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
NIEDERLANDE	NLD	TomTom	2011,06
NIEDERLÄNDISCHE ANTILLEN	ANT	Pitney Bowes	C. 2006
NEUKALEDONIEN	NCL	GeoNames	2011,07
NEUSEELAND	NZL	GeoNames	2011,07
NICARAGUA	NIC	GeoNames	2011,07
NIGER	NER	TomTom	2011,06
NIGERIA	NGA	TomTom	2011,06
NIUE	NIU	GeoNames	2011,07
NORFOLKINSEL	NFK	GeoNames	2011,07
NÖRDLICHE MARIANEN	MNP	GeoNames	2011,07
NORWEGEN	NOR	TomTom	2011,06
OMAN	OMN	TomTom	2011,06
PAKISTAN	PAK	GeoNames	2011,07
PALAU	PLW	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
BESETZTES PALÄSTINENSISCHES GEBIET	PSE	GeoNames	2011,07
PANAMA	PAN	GeoNames	2011,07
PAPUA-NEUGUINEA	PNG	GeoNames	2011,07
PARAGUAY	PRY	GeoNames	2011,07
PERU	PER	GeoNames	2011,07
PHILIPPINEN	PHL	TomTom	2011,06
PITCAIRN	PCN	GeoNames	2011,07
POLEN	POL	TomTom	2011,06
PORTUGAL	PRT	TomTom	2011,06
PUERTO RICO	PRI	GeoNames	2011,07
KATAR	QAT	TomTom	2011,06
RÉUNION	REU	TomTom	2011,06
RUMÄNIEN	ROU	TomTom	2011,06
RUSSISCHE FÖDERATION	RUS	TomTom	2011,06

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
RUANDA	RWA	GeoNames	2011,07
SAINT BARTHÉLEMY	BLM	GeoNames	2011,07
ST. HELENA, ASCENSION UND TRISTAN DA CUNHA	SHN	GeoNames	2011,07
SAINT KITTS UND NEVIS	KNA	GeoNames	2011,07
ST. LUCIA	LCA	GeoNames	2011,07
SAINT-MARTIN (FRANZÖSISCHER TEIL)	MAF	GeoNames	2011,07
SAINT PIERRE UND MIQUELON	SPM	GeoNames	2011,07
ST. VINCENT / GRENADINEN	VCT	GeoNames	2011,07
SAMOA	WSM	GeoNames	2011,07
SAN MARINO	SMR	TomTom	2011,06
SAO TOME UND PRINCIPE	STP	GeoNames	2011,07
SAUDI-ARABIEN	SAU	TomTom	2011,06
SENEGAL	SEN	TomTom	2011,06
SERBIEN	SRB	TomTom	2011,06

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
SEYCHELLEN	SYC	GeoNames	2011,07
SIERRA LEONE	SLE	GeoNames	2011,07
SINGAPUR	SGP	TomTom	2011,06
SINT MAARTEN (NIEDERLÄNDISCHER TEIL)	SXM	GeoNames	2011,07
SLOWAKEI (SLOWAKISCHE REPUBLIK)	SVK	TomTom	2011,06
SLOWENIEN	SVN	TomTom	2011,06
SALOMONINSELN	SLB	GeoNames	2011,07
SOMALIA	SOM	GeoNames	2011,07
SÜDAFRIKA	ZAF	GeoNames	2011,07
SÜDGEORGIEN UND SÜDLICHE SANDWICHINSELN	SGS	GeoNames	2011,07
SPANIEN	ESP	TomTom	2011,06
SRI LANKA	LKA	GeoNames	2011,07
SUDAN	SDN	GeoNames	2011,07
SURINAM	SUR	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
SVALBARD UND JAN MAYEN INSELN	SJM	GeoNames	2011,07
SWASILAND	SWZ	TomTom	2011,06
SCHWEDEN	SWE	TomTom	2011,06
SCHWEIZ	CHE	TomTom	2011,06
SYRISCHE ARABISCHE REPUBLIK	SYR	GeoNames	2011,07
TAIWAN	TWN	TomTom	2011,06
TADSCHIKISTAN	TJK	GeoNames	2011,07
VEREINIGTE REPUBLIK TANSANIA	TZA	TomTom	2011,06
THAILAND	THA	TomTom	2011,06
TIMOR-LESTE	TLS	GeoNames	2011,07
TOGO	TGO	TomTom	2011,06
TOKELAU	TKL	GeoNames	2011,07
TONGA	TON	GeoNames	2011,07
TRINIDAD UND TOBAGO	TTO	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
TUNESIEN	TUN	GeoNames	2011,07
TÜRKEI	TUR	TomTom	2011,06
TURKMENISTAN	TKM	GeoNames	2011,07
TURKS- UND CAICOSINSELN	TCA	GeoNames	2011,07
TUVALU	TUV	GeoNames	2011,07
UGANDA	UGA	TomTom	2011,06
UKRAINE	UKR	TomTom	2011,06
VEREINIGTE ARABISCHE EMIRATE	ARE	TomTom	2011,06
GROSSBRITANNIEN	GBR	TomTom	2011,06
VEREINIGTE STAATEN	USA	GeoNames	2011,07
UNITED STATES MINOR OUTLYING ISLANDS	UMI	GeoNames	2011,07
URUGUAY	URY	TomTom	2011,06
USBEKISTAN	UZB	GeoNames	2011,07
VANUATU	VUT	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
VATIKANSTADT (APOSTOLISCHER STUHL)	VAT	GeoNames	2011,07
VENEZUELA	VEN	GeoNames	2011,07
VIETNAM	VNM	GeoNames	2011,07
BRITISCHE JUNGFERINSELN	VGB	GeoNames	2011,07
AMERIKANISCHE JUNGFERINSELN	VIR	GeoNames	2011,07
WALLIS UND FUTUNA INSELN	WLF	GeoNames	2011,07
WESTSAHARA	ESH	GeoNames	2011,07
JEMEN	YEM	GeoNames	2011,07
SAMBIA	ZMB	TomTom	2011,06
ZIMBABWE	ZWE	GeoNames	2011,07

Abdeckung postalischer Daten von Ländern

Tabelle 43: Ländernamen und Abdeckung postalischer Daten

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
ALGERIEN	DZA	Pitney Bowes	C. 2006

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
AMERIKANISCH-SAMOA	ASM	GeoNames	2011,07
ANDORRA	AND	TomTom	2011,06
ARGENTINIEN	ARG	GeoNames	2011,07
ARMENIEN	ARM	Pitney Bowes	C. 2006
AUSTRALIEN	AUS	GeoNames	2011,07
ÖSTERREICH	AUT	TomTom	2011,06
ASERBAIDSCHAN	AZE	Pitney Bowes	C. 2006
BAHRAIN	BHR	Pitney Bowes	C. 2006
BANGLADESCH	BGD	GeoNames	2011,07
WEISSRUSSLAND	BLR	Pitney Bowes	C. 2006
BELGIEN	BEL	TomTom	2011,06
BERMUDA	BMU	Pitney Bowes	C. 2006
BOSNIEN UND HERZEGOWINA	BIH	Pitney Bowes	C. 2006
BRASILIEN	BRA	TomTom	2011,09

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
BRITISCHES TERRITORIUM IM INDISCHEN OZEAN	IOT	Pitney Bowes	C. 2006
BRUNEI DARUSSALAM	BRN	Pitney Bowes	C. 2006
BULGARIEN	BGR	GeoNames	2011,07
KAMBODSCHA	KHM	Pitney Bowes	C. 2006
KANADA	CAN	TomTom	2011,09
KAP VERDE	CPV	Pitney Bowes	C. 2006
CHILE	CHL	Pitney Bowes	C. 2006
CHINA	CHN	Pitney Bowes	C. 2006
WEIHNACHTSINSEL	CXR	Pitney Bowes	C. 2006
COCOSINSELN	CCK	Pitney Bowes	C. 2006
COSTA RICA	CRI	Pitney Bowes	C. 2006
KROATIEN (LOKALE BEZEICHNUNG: HRVATSKA)	HRV	GeoNames	2011,07
KUBA	CUB	Pitney Bowes	C. 2006
ZYPERN	CYP	Pitney Bowes	C. 2006

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
TSCHECHIEN	CZE	TomTom	2011,06
DÄNEMARK	DNK	GeoNames	2011,07
DOMINIKANISCHE REPUBLIK	DOM	GeoNames	2011,07
ECUADOR	ECU	Pitney Bowes	C. 2006
ÄGYPTEN	EGY	Pitney Bowes	C. 2006
EL SALVADOR	SLV	Pitney Bowes	C. 2006
ESTLAND	EST	TomTom	2011,06
ÄTHIOPIEN	ETH	Pitney Bowes	C. 2006
FALKLANDINSELN (MALWINEN)	FLK	Pitney Bowes	C. 2006
FÄRÖER-INSELN	FRO	GeoNames	2011,07
FINNLAND	FIN	TomTom	2011,06
FRANKREICH	FRA	TomTom	2011,06
FRANZÖSISCH-GUAYANA	GUF	GeoNames	2011,07
FRANZÖSISCH-POLYNESIEN	PYF	Pitney Bowes	C. 2006

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
GEORGIEN	GEO	Pitney Bowes	C. 2006
DEUTSCHLAND	DEU	TomTom	2011,06
GRIECHENLAND	GRC	TomTom	2011,06
GRÖNLAND	GRL	GeoNames	2011,07
GADELOUPE	GLP	GeoNames	2011,07
GUAM	GUM	GeoNames	2011,07
GUATEMALA	GTM	GeoNames	2011,07
GUERNSEY	GGY	GeoNames	2011,07
GUINEA	GIN	Pitney Bowes	C. 2006
GUINEA-BISSAU	GNB	Pitney Bowes	C. 2006
HAITI	HTI	Pitney Bowes	C. 2006
HONDURAS	HND	Pitney Bowes	C. 2006
UNGARN	HUN	GeoNames	2011,07
ISLAND	ISL	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
INDIEN	IND	GeoNames	2011,07
INDONESIEN	IDN	TomTom	2011,06
IRAN (ISLAMISCHE REPUBLIK)	IRN	Pitney Bowes	C. 2006
IRAK	IRQ	Pitney Bowes	C. 2006
IRLAND	IRL	Pitney Bowes	C. 2006
ISLE OF MAN	IMN	GeoNames	2011,07
ISRAEL	ISR	Pitney Bowes	C. 2006
ITALIEN	ITA	TomTom	2011,06
JAMAICA	JAM	Pitney Bowes	C. 2006
JAPAN	JPN	GeoNames	2011,07
JERSEY	JEY	GeoNames	2011,07
JORDANIEN	JOR	Pitney Bowes	C. 2006
KASACHSTAN	KAZ	Pitney Bowes	C. 2006
KENIA	KEN	Pitney Bowes	C. 2006

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
REPUBLIK KOREA	KOR	Pitney Bowes	C. 2006
KUWAIT	KWT	Pitney Bowes	C. 2006
KIRGISISTAN	KGZ	Pitney Bowes	C. 2006
VOLKSDEMOKRATISCHE REPUBLIK LAOS	LAO	Pitney Bowes	C. 2006
LETTLAND	LVA	TomTom	2011,06
LIBANON	LBN	Pitney Bowes	C. 2006
LESOTHO	LSO	Pitney Bowes	C. 2006
LIBERIA	LBR	Pitney Bowes	C. 2006
LIECHTENSTEIN	LIE	GeoNames	2011,07
LITAUEN	LTU	TomTom	2011,06
LUXEMBURG	LUX	GeoNames	2011,07
EHEMALIGE JUGOSLAWISCHE REPUBLIK MAZEDONIEN	MKD	GeoNames	2011,07
MADAGASKAR	MDG	Pitney Bowes	C. 2006
MALAYSIA	MYS	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
MALEDIVEN	MDV	Pitney Bowes	C. 2006
MALTA	MLT	Pitney Bowes	C. 2006
MARSHALLINSELN	MHL	GeoNames	2011,07
MARTINIQUE	MTQ	GeoNames	2011,07
MAYOTTE	MYT	GeoNames	2011,07
MEXIKO	MEX	TomTom	2011,06
FÖDERIERTE STAATEN VON MIKRONESIEN	FSM	Pitney Bowes	C. 2006
REPUBLIK MOLDAU	MDA	GeoNames	2011,07
MONACO	MCO	GeoNames	2011,07
MONGOLEI	MNG	Pitney Bowes	C. 2006
MAROKKO	MAR	TomTom	2011,06
MOSAMBIK	MOZ	Pitney Bowes	C. 2006
MYANMAR	MMR	Pitney Bowes	C. 2006
NEPAL	NPL	Pitney Bowes	C. 2006

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
NIEDERLANDE	NLD	TomTom	2011,06
NEUKALEDONIEN	NCL	Pitney Bowes	C. 2006
NEUSEELAND	NZL	GeoNames	2011,07
NICARAGUA	NIC	Pitney Bowes	C. 2006
NIGER	NER	Pitney Bowes	C. 2006
NIGERIA	NGA	Pitney Bowes	C. 2006
NORFOLKINSEL	NFK	Pitney Bowes	C. 2006
NÖRDLICHE MARIANEN	MNP	GeoNames	2011,07
NORWEGEN	NOR	TomTom	2011,06
OMAN	OMN	Pitney Bowes	C. 2006
PAKISTAN	PAK	GeoNames	2011,07
PALAU	PLW	Pitney Bowes	C. 2006
PAPUA-NEUGUINEA	PNG	Pitney Bowes	C. 2006
PARAGUAY	PRY	Pitney Bowes	C. 2006

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
PHILIPPINEN	PHL	GeoNames	2011,07
PITCAIRN	PCN	Pitney Bowes	C. 2006
POLEN	POL	TomTom	2011,06
PORTUGAL	PRT	TomTom	2011,06
PUERTO RICO	PRI	GeoNames	2011,07
RÉUNION	REU	GeoNames	2011,07
RUMÄNIEN	ROU	Pitney Bowes	C. 2006
RUSSISCHE FÖDERATION	RUS	TomTom	2011,06
ST. HELENA, ASCENSION UND TRISTAN DA CUNHA	SHN	Pitney Bowes	C. 2006
SAINT PIERRE UND MIQUELON	SPM	GeoNames	2011,07
SAN MARINO	SMR	TomTom	2011,06
SAUDI-ARABIEN	SAU	Pitney Bowes	C. 2006
SENEGAL	SEN	Pitney Bowes	C. 2006
SINGAPUR	SGP	TomTom	2011,06

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
SLOWAKEI (SLOWAKISCHE REPUBLIK)	SVK	TomTom	2011,06
SLOWENIEN	SVN	TomTom	2011,06
SÜDAFRIKA	ZAF	GeoNames	2011,07
SÜDGEORGIEN UND SÜDLICHE SANDWICHINSELN	SGS	Pitney Bowes	C. 2006
SPANIEN	ESP	TomTom	2011,06
SRI LANKA	LKA	GeoNames	2011,07
SUDAN	SDN	Pitney Bowes	C. 2006
SWASILAND	SWZ	Pitney Bowes	C. 2006
SCHWEDEN	SWE	GeoNames	2011,07
SCHWEIZ	CHE	TomTom	2011,06
TAIWAN	TWN	TomTom	2011,06
TADSCHIKISTAN	TJK	Pitney Bowes	C. 2006
THAILAND	THA	TomTom	2011,06
TIMOR-LESTE	TLS	Pitney Bowes	C. 2006

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
TUNESIEN	TUN	Pitney Bowes	C. 2006
TÜRKEI	TUR	TomTom	2011,06
TURKMENISTAN	TKM	Pitney Bowes	C. 2006
TURKS- UND CAICOSINSELN	TCA	Pitney Bowes	C. 2006
UKRAINE	UKR	Pitney Bowes	C. 2006
VEREINIGTE ARABISCHE EMIRATE	ARE	Pitney Bowes	C. 2006
GROSSBRITANNIEN	GBR	TomTom	2011,06
VEREINIGTE STAATEN	USA	TomTom	2011,06
URUGUAY	URY	Pitney Bowes	C. 2006
USBEKISTAN	UZB	Pitney Bowes	C. 2006
VATIKANSTADT (APOSTOLISCHER STUHL)	VAT	TomTom	2011,06
VENEZUELA	VEN	Pitney Bowes	C. 2006
VIETNAM	VNM	Pitney Bowes	C. 2006
AMERIKANISCHE JUNGFERNINSELN	VIR	GeoNames	2011,07

Ländername	ISO 3166-Ländercode	Datenquelle	Alter
WALLIS UND FUTUNA INSELN	WLF	Pitney Bowes	C. 2006
WESTSAHARA	ESH	Pitney Bowes	C. 2006
SAMBIA	ZMB	Pitney Bowes	C. 2006

Geocoding-Szenarien

Sie können den Enterprise Manager verwenden, um Datenflüsse zu erstellen, die für Ihre Geschäftsanforderungen und für die Beschaffenheit und Qualität Ihrer Daten geeignet sind.

„Multiple Country“-Schritt mit „Geocode Address World“ als letztem Geocoding-Durchlauf

Sie können Ihre Ergebnisse optimieren, indem Sie Ihre Eingabe in mehreren Durchläufen geocodieren. Im Allgemeinen können Sie im ersten Durchlauf strengere Vergleichskriterien verwenden. In nachfolgenden Geocoding-Durchläufen können Sie dann weniger einschränkende Vergleichskriterien auf Adressen anwenden, für die zuvor kein Kandidat mit hoher Übereinstimmung zurückgegeben wurde. Mit dieser Strategie können Sie für hochwertige Adressen genaue Übereinstimmungen und für weniger genaue Adressen oder für Adressen in Ländern ohne umfassende Abdeckung weiterhin die bestmöglichen Übereinstimmungen erhalten.

Gehen wir vom folgenden Szenario aus:

- Ihre Eingabedatei enthält Adressen für sechs Länder: Argentinien (ARG), Brasilien (BRA), Mexiko (MEX), Chile (CHL), Venezuela (VEN) und Panama (PAN).
 - Sie haben Geocoder für drei dieser Länder (ARG, BRA und MEX) in einem „Multiple Country“-Schritt bereitgestellt.
 - „Geocode Address World“ wird in einem separaten Schritt bereitgestellt, um Adressen zu geocodieren, die durch die landesspezifischen Geocoder nicht identifiziert werden konnten.
 - In Ihrem Schritt wird Conditional Router (und optional Stream Combiner) verwendet, um den Geocoding-Fluss zu verwalten.
1. Lesen Sie die Eingabe in den „Multiple Country“-Schritt. Geocodierte Adressen können in eine Datei geschrieben oder optional an den Stream Combiner gesendet werden.
 2. Es gibt einige Adressen, die in Schritt 1 nicht geocodiert werden konnten. Das kann daran liegen, dass es sich um Adressen aus CHL, VEN oder PAN handelt

und Sie für diese Länder keine Geocoder im ersten Schritt haben. Oder für diese Adressen wurde wegen Eingabefehlern oder Mehrdeutigkeiten kein Kandidat mit hoher Übereinstimmung im ersten Schritt zurückgegeben. Diese nicht geocodierten Adressen werden an den „Geocode Address World“-Schritt gesendet.

3. Adressen können von „Geocode Address World“ mit geografischer oder postalischer Genauigkeit geocodiert werden. Erfolgreich geocodierte Adressen können in eine Datei geschrieben oder optional an den Stream Combiner gesendet werden.

Postalische geocodierte Kandidaten haben einen Z1-Ergebniscode. Postalische geocodierte Ergebnisse können in Ländern mit zuverlässigen Postleitzahlensystemen sehr genau sein. Siehe [Postalisches Geocoding](#) auf Seite 281. Geografische Kandidaten haben einen G-Ergebniscode (z. B. G3 für eine Übereinstimmung bei Stadt/Ort). Siehe [Geografisches Geocoding](#) auf Seite 282.

4. Der Stream Combiner (sofern er in Ihrem Datenfluss verwendet wird) kann alle geocodierten Adressen kombinieren und in eine Datei schreiben oder zur weiteren Verarbeitung weiterleiten.

Dies ist ein Szenario. Sie könnten den Enterprise Manager verwenden, um komplexere Datenflüsse zu entwerfen, die für Ihre Anforderungen geeignet sind.

Verwenden von „Geocode Address World“ als erstem Geocoding-Durchlauf

Sie könnten auch eine Strategie mit „Geocode Address World“ als erstem Geocoding-Durchlauf verwenden.

Nehmen wir Folgendes an:

- In Ihren Adressen ist normalerweise kein Land angegeben (dies kann aber bei einigen anders sein).
- Manche Adressen enthalten nur Straßen- und Ortsadressdaten.
- Sie haben landesspezifische Geocoder für einige Länder, aber nicht für alle.
- Sie verwenden einen Hauptdatenfluss mit Unterflüssen für den Geocoding-Prozess.

Verwenden Sie einen Datenfluss (vielleicht mit Unterflüssen), um die folgenden Aktionen auszuführen. Beachten Sie, dass in diesen Schritten ein Beispieldatenfluss vereinfacht dargestellt wird.

1. Lesen Sie die Eingabe in den „Multiple Country“-Schritt, der auch „Geocode Address World“ beinhaltet. Auf Basis des Ortsnamens (und eventuell des Namens des Bundesstaats für Adressen in den USA) können sich für jede Adresse ein oder mehr potenzielle Kandidaten mit hoher Übereinstimmung für mehrere unterschiedliche Länder ergeben. Jeder Kandidat wird jetzt einem Land zugeordnet, auch wenn die Eingabeadresse kein Land enthalten hat.

2. Falls ein landesspezifischer Geocoder verfügbar ist, wird der Kandidat an diesen Geocoder gesendet. Diese Verarbeitung beinhaltet bedingte Weiterleitung, Stream Combiner und weitere Spectrum™ Technology Platform-Steuerungsschritte. Je nach Vollständigkeit der Eingabeadresse und der Funktionalität des landesspezifischen Geocoders können Kandidaten auf Straßen- (S-Ergebniscode), geografischer (G-Ergebniscode) oder postalischer (Z-Ergebniscode) Ebene geocodiert werden.
3. Falls kein landesspezifischer Geocoder verfügbar ist, wird der Kandidat an „Geocode Address World“ weitergeleitet, wo die Kandidaten auf geografischer oder postalischer Ebene geocodiert werden können.
4. Die Kandidaten aus allen Unterflüssen werden kombiniert und anhand einer Reihe von Kriterien priorisiert. Die Priorität könnte auf der Bevölkerung des Orts (Ortsrang), der Genauigkeit der Übereinstimmung (Straße, geografisch, postalisch), der Nähe zur Lokalität eines Benutzers oder anderen Kriterien basieren.

Eingabe

„GeocodeAddressWorld“ verwendet eine Adresse als Eingabe. Für die beste Leistung und die höchstmöglichen Ergebnisse müssen Ihre Eingabeadressenlisten so vollständig wie möglich, frei von Schreibfehlern und unvollständigen Adressen und so nahe wie möglich an Postbehördenstandards sein. Die meisten Postbehörden besitzen Websites, die Informationen über Adressenstandards für ihr jeweiliges Land enthalten.

Anmerkung: Der Ländername oder der zwei- bzw. dreistellige ISO-Code ist optional. Bei Weglassen des Landes gibt „GeocodeAddressWorld“ die besten verfügbaren Kandidaten auf Basis der restlichen Eingabedaten zurück.

Eingabefelder

Die folgende Tabelle bietet Informationen über das Format und Layout von „GeocodeAddressWorld“-Eingaben.

Anmerkung: Geben Sie die Eingabe mithilfe der Klasse `DataTable` an. Weitere Informationen finden Sie im Spectrum™ Technology Platform-API-Handbuch.

Tabelle 44: „GeocodeAddressWorld“-Eingabedaten

Spaltenname	Format	Beschreibung
AddressLine1	Zeichenfolge	Die erste Adresszeile. Beispiel: 4360 DUKES RD: 4360 DUKES RD KALGOORLIE WA 6430

Spaltenname	Format	Beschreibung
AddressLine2	Zeichenfolge	<p>Die zweite Adresszeile einer zweizeiligen Adresse. Beispiel: Level 6 51 Jacobson St:</p> <p>26 WELLINGTON ST E SUITE 500 TORONTO ON M5E 1S2</p> <p>Dieses Feld wird in Australien, Österreich, Belgien, Brasilien, Dänemark, Finnland, Frankreich, Deutschland, Irland, Italien, Liechtenstein, Luxemburg, Malaysia, den Niederlanden, Polen, Portugal, Spanien, Schweden, der Schweiz und Thailand nicht verwendet.</p>
City	Zeichenfolge	<p>Der Orts- oder Stadtname. Ihre Eingabeadresse muss den offiziellen Ortsnamen enthalten. Dadurch werden die besten Geocoding-Ergebnisse erzielt.</p> <p>Für Thailand enthält dieses Feld den Unterbezirk (tambon).</p>
County	Zeichenfolge	<p>Der Name einer der folgenden Optionen in Abhängigkeit vom Land:</p> <ul style="list-style-type: none"> • Nicht verwendet: AUT, BRA, CAN, FIN, GBR, MYS, PRT, SGP. • Department: FRA • Bezirk (amphoe): THA • Bezirk (fylke/Grafschaften): NOR • Bezirk (powiat): POL • Kommun: SWE • Kreis: DEU • Local Government Authority (LGA): AUS • Provinz: BEL, CHE, DNK, ESP, IRL, ITA, LIE, LUX, NLD • Region: NZL
FirmName	Zeichenfolge	<p>Unternehmen, Name oder Ortsname. Zum Beispiel: PITNEY BOWES.</p> <p>PITNEY BOWES 4360 DUKES RD KALGOORLIE WA 6430</p>
LastLine	Zeichenfolge	<p>Die letzte Zeile der Adresse. Beispiel: KALGOORLIE WA 6430:</p> <p>4360 DUKES RD KALGOORLIE WA 6430</p>
Locality	Zeichenfolge	<p>Der Name einer der folgenden Optionen in Abhängigkeit vom Land:</p> <ul style="list-style-type: none"> • Nicht verwendet: AUS, AUT, BEL, CHE, DEU, DNK, FIN, FRA, IRL, LIE, LUX, MYS, NLD, NOR, POL, SGP, SWE, THA • Dissemination Area und Enumeration Area (DA und EA): CAN • Lokalität: BRA, GBR, ITA, PRT • Vorort: NZL

Spaltenname	Format	Beschreibung
PostalCode	Zeichenfolge	Die Postleitzahl im entsprechenden Format des Landes.
StateProvince	Zeichenfolge	Der Name einer der folgenden Optionen in Abhängigkeit vom Land: <ul style="list-style-type: none"> • Nicht verwendet: BEL, CHE, DNK, IRL, LIE, LUX, NLD, NOR, SGP • Bundesland: DEU • Provinz: CAN • Provinz (changwat): THA • Provinz (voivodship): POL • Region: AUT, ESP, FRA, GBR, NZL, PRT • Region (län): FIN • Region (lan): SWE • Bundesstaat: AUS, BRA • Bundesstaat (negeri): MYS
Country	Zeichenfolge	Der zwei- oder dreistellige ISO-Ländercode. Dieses Feld ist optional. Bei Weglassen des Landes gibt „GeocodeAddressWorld“ die besten verfügbaren Kandidaten auf Basis der restlichen Eingabedaten zurück. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.

Adressaliasse

Einige Länder verfügen über alternative administrative Namen. Beispielsweise kann es für eine Stadt oder einen Ort einen offiziellen Namen geben, es kann allerdings für dieselbe Stadt bzw. denselben Ort auch einen gängigen, aber inoffiziellen alternativen Namen geben. Wenn in den Quelldaten Aliasinformationen verfügbar sind, schließt Geocode Address World diesen Alias in die Datenbank ein. Dadurch kann Geocode Address World das Geocoding erfolgreich durchführen, wenn in Eingabeadressen alternative Namen verwendet werden.

[Sprachaliasse](#) auf Seite 320 werden auch unterstützt.

Sprachaliasse

Einige Länder verfügen über mehrere offizielle Sprachen oder Hauptsprachen. Eine Stadt kann beispielsweise unter ihrem deutschen und italienischen Namen allgemein bekannt sein. Wenn in den Quelldaten Sprachaliasinformationen verfügbar sind, verwendet Geocode Address World diese in der Datenbank. Dadurch kann Geocode Address World das Geocoding erfolgreich durchführen, wenn in Eingabeadressen alternative Sprachbezeichnungen verwendet werden.

Aliasse können auf allen Verwaltungsebenen vorhanden sein, von StateProvince Bundesland/Kanton bis hin zu Locality Lokalität. Unter [Verwaltungseinheiten und Postleitzahlen](#) auf Seite 321 finden Sie eine Beschreibung der Verwaltungsebenen in Verbindung mit den geografischen Daten.

Adressaliasse auf Seite 320 werden auch für häufig verwendete, alternative Verwaltungsbereiche unterstützt.

Abkürzungen für Bundesstaaten/-länder oder Provinzen/Kantone

In einigen Ländern sind Bundesstaaten/-länder oder Provinzen/Kantone ein wichtiger Teil der Adresse, und oft wird dieses Adresselement abgekürzt. Bei bestimmten Ländern werden diese Abkürzungen für Bundesstaaten/-länder oder Provinzen/Kantone von Geocode Address World erkannt. In den Vereinigten Staaten beispielsweise besitzt jeder Bundesstaat eine aus zwei Buchstaben bestehende Abkürzung (z. B. CA für California). Ebenso werden Abkürzungen für die Bundesländer/Provinzen in den Niederlanden erkannt (z. B. GLD für Gelderland).

Geocode Address World akzeptiert Abkürzungen von Bundesstaaten/-ländern oder Provinzen/Kantonen für die folgenden Länder:

Tabelle 45: Unterstützung von Abkürzungen für Bundesstaaten/-länder bzw. Provinzen/Kantone von Ländern

Ländername	Abkürzungen für Bundesstaat/-land oder Provinz/Kanton	Beispiel
Australien (AUS)	StateProvince (Bundesstaat)	NSW (Abkürzung für New South Wales)
Kanada (CAN)	StateProvince (Provinz)	AB (Abkürzung für Alberta)
Italien (ITA)	County (Provinz)	MO (Abkürzung für Modena)
Mexiko (MEX)	StateProvince (Bundesstaat)	JA (Abkürzung für Jalisco)
Niederlande (NLD)	County (Bundesstaat)	FR (Abkürzung für Friesland)
Vereinigte Staaten (USA)	StateProvince (Bundesstaat)	CA (Abkürzung für California)

Geocode Address World wertet diese Abkürzungen von Bundesstaaten/-ländern bzw. Provinzen/Kantonen aus, um hohe Übereinstimmungen besser zu ermitteln. Ein Beispiel für diese Funktion finden Sie unter **Geografisches Geocoding mit Abkürzung für Bundesland/Kanton** auf Seite 284.

Verwaltungseinheiten und Postleitzahlen

Typische Eingabeadressen bestehen aus der Straßenadresse, der Verwaltungseinheit und den Postleitzahleninformationen. Geocode Address World verwendet die Verwaltungseinheiten und Postleitzahlen für das geografische oder postalische Geocoding.

- StateProvince (Bundesstaat oder Provinz)
- County (Grafschaft, Region oder Bezirk)

- City (Ort oder Stadt)
- Locality (Lokalität, Vorstadt oder Dorf)
- Postleitzahl

Bestimmte Verwaltungseinheiten variieren je nach Land. Zum Beispiel kann Locality Lokalität, Vorstadt oder Barrio in Abhängigkeit vom Land enthalten. StateProvince kann Bundesstaat, Provinz, Region oder einen anderen Namen in Abhängigkeit vom Land enthalten. Unter [Abkürzungen für Bundesstaaten/-länder oder Provinzen/Kantone](#) auf Seite 321 finden Sie weitere Informationen dazu, wie Abkürzungen für Bundesstaat/Provinz durch Geocode Address World interpretiert werden.

Nicht alle Verwaltungseinheiten werden in den Adresskonventionen für alle Länder verwendet. Zum Beispiel wird in den USA County (Grafschaft) normalerweise nicht in Adressen verwendet. Aber bei manchen Ländern ist County ein wichtiger Bestandteil der Adresse.

Wenn Ihre Eingabedaten Postleitzahlen enthalten, kann Geocode Address World diese für das postalische Geocoding verwenden, vorausgesetzt, die Quelldaten enthalten postalische Daten für das jeweilige Land.

Empfehlungen für die Eingabe

Sie können Geocode Address World-Ergebnisse optimieren, wenn Sie Ihre Eingabedatensätze vorbereiten und mit ihnen vertraut sind. Befolgen Sie die folgenden Richtlinien:

- Stellen Sie sicher, dass Ihre Eingabeadressen so vollständig und so genau wie möglich sind. Wenn Ihre Eingabeadressen Fehler enthalten, kann Geocode Address World diese Adressen eventuell trotzdem geocodieren. Es ist jedoch möglich, dass es mehrere mögliche Übereinstimmungen gibt oder dass keine hohen Übereinstimmungen gefunden werden. Wenn Sie eine unvollständige oder ungenaue Eingabeadresse überprüfen und korrigieren können, kann dies zu besseren Ergebnissen führen.
- Schließen Sie Postleitzahlen in Ihre Eingabeadressen mit ein, sofern Ihnen diese vorliegen. Dies ist nicht erforderlich, es ermöglicht Geocode Address World allerdings, ein postalisches Geocoding auszuführen. Dies kann bei einigen Adressen zu genaueren Ergebnissen führen, abhängig von dem Land und von der Vollständigkeit und Genauigkeit anderer Adresselemente.
- Schließen Sie den Ländernamen oder den offiziellen dreistelligen oder zweistelligen ISO-Ländercode in Ihre Eingabeadressen mit ein. Dies ist nicht erforderlich, kann Geocode Address World jedoch dabei helfen, zwischen ähnlichen Adressen und Ortsnamen zu unterscheiden, die in verschiedenen Ländern auftreten können.
- Formatieren Sie Ihre Eingabeadressen einheitlich. Geocode Address World kann Eingabeadressen in einer Vielzahl von Eingabeformaten oder unformatierten Eingabe (einzelne Zeile) verarbeiten. Sie können jedoch genauere und schnellere Ergebnisse erzielen, wenn Ihre Eingabeadressen einheitlich formatiert sind und mit den landesspezifischen Adresskonventionen übereinstimmen. Selbst wenn Ihre Eingabeadresse eine einzelne Zeile umfasst (unformatiert), erzielen Sie bessere Ergebnisse und eine bessere Leistung, wenn die Adresselemente einheitlich sortiert sind. Verwenden Sie den Eingabebereich „AddressLine1“ für unformatierte Adressen. Siehe [Einzeilige Eingabe](#) auf Seite 323

Einzeilige Eingabe

Die Adresseingabe kann in separate Eingabefelder oder als einzeilige Eingabe formatiert werden. Verwenden Sie „AddressLine1“ für eine einzeilige Eingabe.

Einzeiliges geografisches Geocoding

In diesem Beispiel wird eine unformatierte Eingabe (einzelne Zeile) verwendet. Geocode Address World analysiert die einzeilige Eingabe, um die geografischen Adresselemente (in diesem Beispiel Graz) zu identifizieren, und führt anschließend ein Geocoding auf einen geografischen Mittelpunkt durch. Die MainAddress (Straßeninformationen) wird nicht verwendet.

Sackstraße 10 Graz

Geocode Address World gibt basierend auf einer Übereinstimmung mit dem City einen Kandidaten mit einer hohen geografischen Übereinstimmung zurück. Obwohl das Land nicht angegeben wurde, identifiziert Geocode Address World die hohe Übereinstimmung in Österreich (AUT).

StateProvince: Steiermark

County: Graz (Stadt)

City: Graz

Land: AUT

Ergebniscode: G3

X: 15,44172

Y: 47,06792

Wenn Ihre Eingabeadressen korrekt sind, kann eine unformatierte Eingabe zu einer Übereinstimmungsrate führen, die mit der einer formatierten Eingabe vergleichbar ist. Die Leistung beim Geocoding unformatierter Adressen ist jedoch normalerweise schlechter als die beim Geocoding formatierter Adressen.

Einzeiliges postalisches Geocoding mit angegebenem Land

In dem folgenden Beispiel wird eine Eingabe in eine einzelne Zeile verwendet und eine Postleitzahl bereitgestellt. Auch das Land Österreich (AUT) ist angegeben. Die Straßenanschrift wird ebenfalls eingegeben, dies wird jedoch von Österreich ignoriert.

Alpenstraße 117 5020 AUT

Österreich gibt einen Kandidaten mit einer hohen Übereinstimmung beim PLZ-Mittelpunkt zurück (Z1-Ergebniscode). Da das Land (AUT) in der Eingabe angegeben ist, muss dieses verglichen werden und es wird eine einzelne hohe Übereinstimmung für diese Postleitzahl in Österreich zurückgegeben. Auch geringe Übereinstimmungen mit der Postleitzahl 5020 von anderen Ländern werden zurückgegeben.

StateProvince: Salzburg

Land: AUT

Postleitzahl: 5020 Ergebniscode: Z1 X: 13,04685 Y: 47,80262
--

Optionen

Geocoding-Optionen

In der folgenden Tabelle werden die Optionen aufgeführt, die das Ermitteln der Koordinaten eines Standorts steuern.

Tabelle 46: Geocoding-Optionen

Optionsname	Beschreibung
CoordinateSystem	<p>Bei einem Koordinatensystem handelt es sich um ein Referenzsystem für den eindeutigen Standort eines Punktes im Raum. Kartesische (planare) und geodätische (geografische) Koordinaten sind Beispiele für Referenzsysteme basierend auf der euklidischen Geometrie. Spectrum™ Technology Platform unterstützt die von der European Petroleum Survey Group (EPSG) anerkannten Systeme.</p> <p>Zur Auswahl stehen:</p> <p>EPSG:4283 Auch als GDA94-Koordinatensystem bezeichnet.</p> <p>EPSG:4326 Auch als WGS84-Koordinatensystem bezeichnet. Standardeinstellung.</p>

Vergleichsoptionen

Tabelle 47: Vergleichsoptionen

Optionsname	Beschreibung
KeepMultimatch	<p>Gibt an, ob Ergebnisse zurückgegeben werden sollen, wenn die Adresse mit mehreren Kandidaten in der Datenbank übereinstimmt. Wenn diese Option deaktiviert ist, kann eine Adresse, für die sich mehrere Kandidaten ergeben, nicht geocodiert werden.</p> <p>Wenn Sie diese Option auswählen, müssen Sie die maximale Anzahl zurückzugebender Kandidaten über die Option „MaxCandidates“ (siehe unten) neben dem Kontrollkästchen angeben.</p> <p>Y Ja, Kandidaten zurückgeben, wenn mehrere Kandidaten ermittelt werden. Standardeinstellung.</p> <p>N Nein, Kandidaten nicht zurückgeben. Bei Adressen, die mehrere Kandidaten ergeben, ist kein Geocoding möglich.</p>
MaxCandidates	<p>Wenn Sie „KeepMultimatch = Y“ angeben, gibt diese Option die maximale Anzahl zurückzugebender Ergebnisse an.</p> <p>Der Standardwert ist 1.</p>
CloseMatchesOnly	<p>Gibt an, ob nur geocodierte Ergebnisse zurückgegeben werden sollen, die Kandidaten mit hoher Übereinstimmung sind. Wenn beispielsweise 10 Kandidaten vorhanden sind, von denen zwei Kandidaten mit einer hohen Übereinstimmung bilden, würden nur die beiden Kandidaten mit hoher Übereinstimmung statt alle 10 zurückgegeben werden.</p> <p>Y Ja, nur hohe Übereinstimmungen zurückgeben.</p> <p>N Nein, nicht nur hohe Übereinstimmungen zurückgeben. Standardeinstellung.</p>

Datenoptionen

Auf der Registerkarte „Daten“ können Sie angeben, welche Datenbanken beim Geocoding verwendet werden sollen. Datenbanken enthalten die Adressen- und Geocode-Daten, die zum Bestimmen des Geocodes für eine bestimmte Adresse notwendig sind. Die Daten basieren auf Adress- und Geocoding-Daten von Postbehörden und Anbietern geografischer Daten.

Anmerkung: Da das EGM-Modul administrative Aufgaben an eine webbasierte Management Console abgibt, haben die Optionen möglicherweise eine andere Bezeichnung als im Enterprise Designer. Es gibt allerdings keine Unterschiede beim Verhalten.

Tabelle 48: Datenoptionen

Optionsname	Beschreibung
DatabaseSearchOrder	<p>Der beim Suchvorgang zu verwendende Name von mindestens einer Datenbankressource. Verwenden Sie den Datenbanknamen, der auf der Seite „Spectrum-Datenbanken“ in der Management Console angegeben ist. tool. Weitere Informationen finden Sie im <i>Spectrum™ Technology Platform-Administratorhandbuch</i>.</p> <p>Sie können mehrere Datenbankressourcen angeben. Bei mehreren Datenbanken geben Sie diese in der gewünschten Reihenfolge an. Die Reihenfolge der Datenbanken hat Auswirkungen, wenn Kandidaten mit hohen Übereinstimmungen aus anderen Datenbanken vorhanden sind. Die zurückgegebenen hohen Übereinstimmungen stammen aus der Datenbank, die in der Suchliste an erster Stelle steht. Hohe Übereinstimmungen aus Datenbanken mit einem niedrigeren Rang werden in geringe Übereinstimmungen zurückgestuft.</p>

Ausgabe

GeocodeAddressWorld gibt den Breitengrad/Längengrad, den Ort, den Bezirk sowie die Ergebnisindikatoren zurück. Ergebnisindikatoren geben an, wie gut der Geocoder die Eingabe mit einem bekannten Ort abgeglichen und einen Breitengrad/Längengrad zugewiesen hat. Sie geben außerdem den allgemeinen Status eines Vergleichsversuchs an. Die Informationen werden in Großbuchstaben zurückgegeben.

Beim Verwenden der API gilt: Die zurückgegebene Ausgabe ist in der Klasse `DataTable`. Weitere Informationen finden Sie im *Spectrum™ Technology Platform-API-Handbuch*.

Adressenausgabe

Tabelle 49: Adressenausgabe

Spaltenname	Beschreibung
City	Gemeindenname.
CityRank	„CityRank“ ist ein numerischer Wert von 1 (hoch) bis 10 (niedrig), basierend auf der Gesamtbevölkerung und relativen Bevölkerung, Bedeutung und anderen Kriterien.

Spaltenname	Beschreibung
Country	<p>Der aus drei Buchstaben bestehende ISO 3166-1 Alpha 3-Ländercode. Der aus zwei Buchstaben bestehende Code kann ebenfalls verwendet werden. Eine Liste von Ländern und Datenquellen für das geografische Geocoding finden Sie unter Abdeckung geografischer Länderdaten auf Seite 286. Eine Liste von Ländern und Datenquellen für das postalische Geocoding finden Sie unter Abdeckung postalischer Daten von Ländern auf Seite 304.</p>
County	<p>Dieses Feld enthält eine Region, die kleiner als ein Bundesstaat/-land bzw. eine Provinz ist, aber größer als eine Stadt. Die Region variiert je nach Land.</p> <ul style="list-style-type: none"> • AUS – Lokale Verwaltungsbehörde (LGA) • AUT – Provinz • BEL – Provinz • BHS – Nicht verwendet • BRA – Nicht verwendet • CAN – Nicht verwendet • CHE – Provinz • DEU – Kreis • DNK – Provinz • FIN – Provinz (Kommune) • FRA – Departement • GBR – County • ITA – Provinz • LIE – Provinz • LUX – Provinz • MYS – Bezirk (Daerah) • NLD – Provinz • NZL – Nicht verwendet • POL – Bezirk (Powiat) • PRT – Nicht verwendet • SGP – Bezirk • SWE – Region (Kommun) • THA – Bezirk (Amphoe)
PostalCode	<p>Die Postleitzahl zu der Adresse. Das Format der Postleitzahl variiert je nach Land.</p>

Spaltenname	Beschreibung
StateProvince	<p>Die Bedeutung von StateProvince variiert je nach Land.</p> <ul style="list-style-type: none"> • AUS – Bundesstaat • AUT – Region • BEL – Nicht verwendet • BRA – Bundesstaat • CAN – Provinz • CHE – Bundesstaat • DEU – Bundesland • DNK – Nicht verwendet • ESP – Region • FIN – Region (Län) • FRA – Region • GBR – Region • IRL – Nicht verwendet • ITA – Region • LIE – Bundesstaat • LUX – Nicht verwendet • MYS – Bundesstaat (Negeri) • NLD – Nicht verwendet • NOR – Nicht verwendet • NZL – Region • POL – Provinz (Woiwodschaft) • PRT – Region • SGP – Nicht verwendet • SWE – Region (Län) • THA – Provinz (Changwat)

Geocode-Ausgabe

Tabelle 50: Geocode-Ausgabe

Spaltenname	Beschreibung
CoordinateSystem	<p>Das verwendete Koordinatensystem zur Bestimmung der Längengrad- und Breitengradkoordinaten. Ein Koordinatensystem gibt unter anderem eine Kartenprojektion und Koordinateneinheiten an. Ein Beispiel ist EPSG:4326. EPSG steht für die European Petroleum Survey Group (europäische Öl- und Gaserkundungsgruppe).</p>

Spaltenname	Beschreibung
Latitude	Siebenstellige Zahl in Grad mit vier Dezimalstellen (im angegebenen Format).
Longitude	Siebenstellige Zahl in Grad mit vier Dezimalstellen (im angegebenen Format).

Ergebniscodes

Ergebniscodes enthalten Informationen über den erfolgreichen oder fehlgeschlagenen Geocoding-Versuch sowie über die Genauigkeit des Geocodes.

Tabelle 51: Ergebniscodeausgabe

columnName	Beschreibung
Geocoder.MatchCode	Gibt an, wie genau die Eingabeadresse mit der Kandidatenadresse übereinstimmt.
IsCloseMatch	<p>Gibt an, ob die Adresse als hohe Übereinstimmung gilt. Eine Adresse wird basierend auf den Optionen „Kriterien für hohe Übereinstimmung“ auf der Registerkarte „Vergleiche“ als hohe Übereinstimmung gezählt.</p> <p>Y Ja, die Adresse weist eine hohe Übereinstimmung auf.</p> <p>N Nein, die Adresse weist keine hohe Übereinstimmung auf.</p>
MultiMatchCount	<p>Beim Geocoding von Straßenanschriften ist dies die Anzahl an übereinstimmenden Adresspositionen, die für die angegebene Adresse gefunden wurden.</p> <p>Beim Kreuzungs-Geocoding ist dies die Anzahl an übereinstimmenden Adresspositionen, die für die angegebenen Adressen gefunden wurden.</p>
Status	<p>Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs.</p> <p>null Erfolg</p> <p>F Fehler</p>

columnName	Beschreibung										
Status.Code	<p>Wenn der Geocoder die Adresse nicht verarbeiten konnte, wird in diesem Feld der Grund dafür angezeigt.</p> <ul style="list-style-type: none"> • Interner Systemfehler • Kein Geocode gefunden • Nicht genügend Eingabedaten • Mehrere Übereinstimmungen gefunden • Eine Ausnahme ist aufgetreten • Geocoder kann nicht initialisiert werden • Keine Übereinstimmung gefunden 										
Status.Description	<p>Wenn der Geocoder die Adresse nicht verarbeiten konnte, wird in diesem Feld eine Beschreibung des Fehlers angezeigt.</p> <table border="0"> <tr> <td>Problem + Erklärung</td> <td>Wird zurückgegeben, wenn Status.Code = Interner Systemfehler.</td> </tr> <tr> <td>Geocoding Failed</td> <td>Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.</td> </tr> <tr> <td>No location returned</td> <td>Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.</td> </tr> <tr> <td>No Candidates Returned</td> <td>Der Geocoder konnte keine Kandidatentreffer für die Adresse ermitteln.</td> </tr> <tr> <td>Multiple Candidates Returned and Keep Multiple Matches not selected</td> <td>Die Adresse ergab mehrere Kandidaten. Damit die Kandidatenadresse zurückgegeben wird, müssen Sie <code>KeepMultimatch=Y</code> angeben.</td> </tr> </table>	Problem + Erklärung	Wird zurückgegeben, wenn Status.Code = Interner Systemfehler.	Geocoding Failed	Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.	No location returned	Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.	No Candidates Returned	Der Geocoder konnte keine Kandidatentreffer für die Adresse ermitteln.	Multiple Candidates Returned and Keep Multiple Matches not selected	Die Adresse ergab mehrere Kandidaten. Damit die Kandidatenadresse zurückgegeben wird, müssen Sie <code>KeepMultimatch=Y</code> angeben.
Problem + Erklärung	Wird zurückgegeben, wenn Status.Code = Interner Systemfehler.										
Geocoding Failed	Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.										
No location returned	Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.										
No Candidates Returned	Der Geocoder konnte keine Kandidatentreffer für die Adresse ermitteln.										
Multiple Candidates Returned and Keep Multiple Matches not selected	Die Adresse ergab mehrere Kandidaten. Damit die Kandidatenadresse zurückgegeben wird, müssen Sie <code>KeepMultimatch=Y</code> angeben.										

columnName	Beschreibung
LocationPrecision	Ein Code, der die Genauigkeit des Geocodes angibt. Zur Auswahl stehen:
	0 Es sind keine Koordinateninformationen zu dieser Adresse verfügbar.
	1 Interpolierte Straßenanschrift.
	2 Mittelpunkt des Straßensegments.
	3 Mittelpunkt von Postleitzahl 1.
	4 Mittelpunkt der Teil-Postleitzahl 2.
	5 Mittelpunkt von Postleitzahl 2.
	6 Kreuzung.
	7 Point of Interest. Hierbei handelt es sich um einen Platzhalterwert. Spectrum-Datenbanken haben keine POI-Daten, sodass dies nicht zurückgegeben werden kann.
	8 Bundesland-/Kantonmittelpunkt.
	9 Bezirksmittelpunkt.
	10 Ortsmittelpunkt.
	11 Lokalitätsmittelpunkt.
	12–15 Für die meisten Länder sind die LocationPrecision-Codes 12–15 für unbestimmte benutzerdefinierte Elemente reserviert.
	13 Zusätzliche Punktgenauigkeit für ein unbestimmtes benutzerdefiniertes Element.
	14 Zusätzliche Punktgenauigkeit für ein unbestimmtes benutzerdefiniertes Element.
	15 Zusätzliche Punktgenauigkeit für ein unbestimmtes benutzerdefiniertes Element.
	16 Das Ergebnis ist ein Adresspunkt.
	17 Das Ergebnis wurde generiert, indem die Segmentdaten des Kandidaten mithilfe von Adresspunktdaten geändert wurden.
	18 Das Ergebnis ist ein Adresspunkt, der mithilfe des Features „Versatz von Mittellinie“ projiziert wurde. Zur Verwendung des Features „Versatz von Mittellinie“ müssen Sie eine Punkt- und Straßenbereichsdatenbank haben, und darüber LocationPrecision 18 zurückgeben.

columnName	Beschreibung
StreetDataType	<p>Der Rang der zum Geocodieren der Adresse verwendeten Datenbank in der Standardsuchreihenfolge. Der Wert „1“ bedeutet, dass die Datenbank an erster Stelle der Standardsuchreihenfolge steht. Der Wert „2“ bedeutet, dass die Datenbank an zweiter Stelle der Standardsuchreihenfolge steht, usw.</p> <p>Die Standardsuchreihenfolge für Datenbanken wird in Management Console angegeben.</p>

Priorität geografischer Kandidaten

Identische geografische Bereichsnamen finden sich in vielen Ländern. In solchen Fällen verwendet Geocode Address World ein Prioritätssystem, um zu bestimmen, welcher der potenziellen Kandidaten die wahrscheinlichste hohe Übereinstimmung darstellt.

Die konkreten Details dieser gewichteten Priorität hängen ein wenig von der Datenquelle ab (TomTom, GeoNames oder Pitney Bowes Quelle). Grundsätzlich werden die folgenden Kriterien gewichtet, um den wahrscheinlichsten Kandidaten mit hoher Übereinstimmung zu bestimmen.

- Landeshauptstadt
- Hauptstadt des Verwaltungsbereichs (Bundesstaat/Provinz, Region, Grafschaft)
- Bevölkerungszahl

Der Status als Landeshauptstadt wiegt schwerer als alle anderen geografischen Prioritätskriterien. Beispiel: Bei Eingabe von San Juan als „Ort“ wird San Juan, Puerto Rico (PRI) als hohe Übereinstimmung zurückgegeben, weil es sich dabei um die Hauptstadt von PRI handelt. Andere Orte mit Namen San Juan auf der Welt (einschließlich Spanien, Costa Rica, der Dominikanischen Republik und der Philippinen) können unabhängig von ihrer Bevölkerung als nicht hohe Übereinstimmungen zurückgegeben werden. Damit Übereinstimmungen zurückgegeben werden, müssen Sie das Kontrollkästchen „Mehrere Übereinstimmungen beibehalten“ in der Management Console aktivieren und die Anzahl der zurückzugebenden Übereinstimmungen angeben.

Ebenso gibt Geocode Address World Roma, ITA als hohe Übereinstimmung zurück, da es sich um die Hauptstadt von Italien handelt, während Roma in Rumänien, Honduras und Panama als nicht hohe Übereinstimmungen zurückgegeben werden.

Hauptstädte des Verwaltungsbereichs Bundesstaat/Provinz werden hoch gewichtet, auch wenn ihre Bevölkerungen nicht sehr groß sind. Zum Beispiel wird für Springfield eine hohe geografische Übereinstimmung mit Springfield, Illinois USA zurückgegeben, weil es sich dabei um die Hauptstadt des Bundesstaates Illinois handelt. Springfield, Massachusetts hat eine etwas größere Bevölkerung, aber der Status von Springfield als Hauptstadt des Bundesstaates Illinois wird höher gewichtet. Weitere dünner besiedelte Springfield-Gemeinden in den USA und anderen Ländern werden ebenfalls als hohe Übereinstimmungen zurückgegeben, aber unter dem Kandidaten Springfield Illinois aufgelistet. Es ist möglich, dass ein großer Ort als gleich hohe Übereinstimmung wie eine gleichnamige Hauptstadt des Bundesstaates bzw. der Provinz gewertet wird. Allerdings wird die

Hauptstadt des Bundesstaates bzw. der Provinz nicht herabgestuft, auch wenn sie eine relativ kleine Bevölkerung hat.

Ebenso gilt: Wenn Ihre Eingabe in „Ort“ Albany ohne Angabe eines Landes lautet, gibt Geocode Address World Albany, NY, USA als Kandidat mit hoher Übereinstimmung zurück. Dies deshalb, weil Albany die Hauptstadt des Bundesstaates New York ist und aus diesem Grund eine höhere Priorität als die Hauptstadt des Verwaltungsbereichs erhält. Die Bevölkerung trägt als Faktor ebenfalls zur Priorität bei. Wenn Sie den Ort Albany mit einem anderen Land wie Neuseeland angeben, dann wird das Land verwendet und Albany, NZL als Kandidat mit hoher Übereinstimmung zurückgegeben.

Wenn ein Kandidat einen Ort einschließt, wird auch ein CityRank-Wert zurückgegeben, sofern verfügbar. CityRank ist ein numerischer Wert im Bereich von 1 (hoch) bis 10 (niedrig), der die relative Priorität des Ortes angibt. Diese Priorität basiert auf der relativen Bevölkerung, dem Verwaltungsstatus und anderen Kriterien. Wenn mehrere geografische Kandidaten zurückgegeben werden, werden sie sortiert nach Ortsrang aufgelistet.

Vergleichscodes

Übereinstimmungen in der Kategorie G geben an, dass sich der Kandidat am geografischen Mittelpunkt befindet, mit den folgenden möglichen Genauigkeitsstufen. Nicht alle Genauigkeitsstufen sind für alle Länder möglich.

- **G0:** Landesmittelpunkt. Wird für „GeocodeAddressWorld“ nicht zurückgegeben.
- **G1:** Bundelands- oder Provinzmittelpunkt. Für Japan wird damit eine Übereinstimmung bei der Präfektur (ken) angegeben.
- **G2:** Landesmittelpunkt. Für Japan wird damit eine Übereinstimmung bei der Stadt (shi) angegeben.
- **G3:** Stadtmittelpunkt. Für Japan wird damit eine Übereinstimmung bei der Unterteilung Gemeinde oder Teilstadt (oaza) angegeben. Für Australien können LGA-Informationen (Local Government Authority) nur aus der Adressdatenbank für Straßenbereiche (nicht der G-NAF-Datenbank) zurückgegeben werden.
- **G4:** Lokaliätätsmittelpunkt. Für Japan wird damit eine Übereinstimmung beim Stadtbezirk (chome) angegeben.

Übereinstimmungen in der Kategorie Z geben an, dass aus einem der folgenden Gründe kein Straßenvergleich durchgeführt wurde:

- Sie haben angegeben, dass mit PLZ-Mittelpunkten verglichen werden soll. Der resultierende Punkt befindet sich im PLZ-Mittelpunkt mit vier möglichen Genauigkeitsstufen.
- Es gibt keine hohe Übereinstimmung, und Sie haben als Fallback-Option PLZ-Mittelpunkt angegeben.

Die Kategorie Z umfasst die folgenden Genauigkeitsstufen:

- **Z0:** PLZ-Übereinstimmung, keine Koordinaten verfügbar (kommt selten vor).
- **Z1:** PLZ-Mittelpunkt-Übereinstimmung.
- **Z3:** Vollständige PLZ-Mittelpunkt-Übereinstimmung. Für Kanada ist das ein FSALDU-Mittelpunkt.
- **Z6:** PLZ-Mittelpunkt-Übereinstimmung für Punkt-PLZ.

GNAFPIDLocationSearch

GNAFPIDLocationSearch identifiziert die Adresse und die Koordinaten für den Breitengrad/Längengrad für einen Geocoded National Address File Persistent Identifier (G-NAF PID). Der G-NAF PID ist eine vierzehnstellige alphanumerische Zeichenfolge, die die einzelnen G-NAF-Adressen in der G-NAF-Datenbank (Datenbank mit australischen Standorten) eindeutig identifiziert. Der PID wird aus einer Kombination der Hauptadressfelder der G-NAF-Datenbank erstellt. Beispiel für einen G-NAF PID:

GAVIC411711441

Anmerkung: Die G-NAF-Datenbank muss installiert sein, damit GNAFPIDLocationSearch verwendet werden kann.

GNAFPIDLocationSearch ist Teil der Geocoding Address AUS-Komponente. GNAF PID Location Search ist der einzige von Geocode Address AUS verwendete Schritt. Diese Komponente ist für alle anderen Nutzungsarten veraltet. Verwenden Sie die Geocode Address Global-Komponente für alle anderen australischen Geocoding-Funktionen.

Weitere Informationen zum Enterprise Geocoding-Modul finden Sie unter [Enterprise Geocoding-Modul](#) auf Seite 267.

Eingabe für G-NAF PID

GNAFPIDLocationSearch verwendet eine G-NAF PID als Eingabe und gibt die Adresse und die Koordinaten für den Breitengrad/Längengrad für einen Geocoded National Address File Persistent Identifier (G-NAF PID) zurück.

Anmerkung: Die Funktionalität von GNAF PID Location Search wird von der Komponente „Geocode Address Global“ nicht unterstützt. Für diese Funktionalität müssen Sie die Komponente „Geocode Address AUS“ verwenden. GNAF PID Location Search ist der einzige von Geocode Address AUS verwendete Schritt. Diese Komponente ist für alle anderen Nutzungsarten veraltet.

Tabelle 52: Eingabe für GNAFPIDLocationSearch

Spaltenname	Format	Beschreibung
GNAFPID	Zeichenfolge	Der vierzehnstellige beständige G-NAF-Bezeichner, nach dem Sie suchen möchten. Beispiel: GAVIC411711441

Anmerkung: Eingabe mithilfe der Klasse `DataTable` angeben. Weitere Informationen finden Sie im Spectrum™ Technology Platform-API-Handbuch.

Optionen für G-NAF PID Location Search

GNAFPIDLocationSearch enthält eine Option, um die G-NAF-Datenbank für die PID-Suche auszuwählen.

Geocoding-Optionen für G-NAF

Tabelle 53: Geocoding-Optionen für GNAFPIDLocationSearch

optionName	Beschreibung
GNAFPointType	<p>Gibt an, ob der Breitengrad/Längengrad der Parzelle oder der Straßenfront zurückgegeben werden soll. Diese Option ist nur verfügbar, wenn die G-NAF-Datenbank installiert ist. Diese Option hat nur Auswirkungen auf Adressen, die mit der G-NAF-Datenbank abgeglichen wurden.</p> <p>Zur Auswahl stehen:</p> <ul style="list-style-type: none"> P Bei einer Übereinstimmung der Straßenanschrift den genauen Standort der Parzelle zurückgeben. Hierbei handelt es sich um den standardmäßigen G-NAF-Punkt, der den genauen von der G-NAF-Datenbank zurückgegebenen maßgebenden Punkt bildet. Standardeinstellung. S Bei einer Übereinstimmung der Straßenanschrift den Straßenfrontpunkt der Parzelle zurückgeben. Der Straßenfrontpunkt liegt 12,5 m von der Vorgrenze der Parzelle entfernt. Straßenfrontpunkte eignen sich besser für Routenführungsanwendungen.

optionName	Beschreibung
Return8DecimalPlaceLatLong	<p>Gibt an, ob der ursprüngliche Breiten- und Längengrad mit einer Genauigkeit von bis zu acht Stellen nach dem Dezimalzeichen zurückgegeben werden sollen. Dies ist der Breitengrad/Längengrad, mit dem der Kandidat in der G-NAF-Datenbank übereinstimmt. Dies sind die direkt aus den G-NAF-Daten stammenden Originalkoordinaten vor dem Abschneiden oder Runden. Diese Option ist nur verfügbar, wenn die G-NAF-Datenbank installiert ist. Diese Option hat nur Auswirkungen auf Adressen, die mit der G-NAF-Datenbank abgeglichen wurden.</p> <p>Y Ja, ursprünglichen Breitengrad/Längengrad mit einer Genauigkeit von bis zu acht Stellen nach dem Dezimalzeichen zurückgeben.</p> <p>N Nein, ursprünglichen Breitengrad/Längengrad nicht zurückgeben.</p>

Optionen für G-NAF PID-Daten

Tabelle 54: Optionen für GNAFPIDLocationSearch-Daten

optionName	Beschreibung
Database	<p>Gibt die Datenbank an, über die Sie nach der Parzelle suchen können. Verwenden Sie den Datenbanknamen, der in Management Console angegeben ist. Weitere Informationen finden Sie im <i>Spectrum™ Technology Platform-Administratorhandbuch</i>.</p> <p>Anmerkung: In dieser Liste sind nur Datenbankressourcen verfügbar, die G-NAF-Datenbanken enthalten.</p>

Ausgabe

Adressenausgabe

Tabelle 55: Adressenausgabe

Spaltenname	Beschreibung
AddressLine1	Erste Zeile der Adresse.
AddressLine2	Zweite Zeile der Adresse.
ApartmentLabel	Der Typ der Wohneinheit, z. B. Apartment, Suite oder Parzelle.
ApartmentNumber	Nummer der Wohneinheit.
City	Gemeindenname.
Country	Der aus drei Buchstaben bestehende ISO 3166-1 Alpha 3-Ländercode.
County	Die Local Government Authority (LGA).
FirmName	Name des Unternehmens oder ein Ortsname.
HouseNumber	Gebäudennummer des abgeglichenen Orts.
HouseNumberHigh	Die höchste Hausnummer in dem Bereich, in dem die Adresse zu finden ist.
HouseNumberLow	Die niedrigste Hausnummer in dem Bereich, in dem die Adresse zu finden ist.

Spaltenname	Beschreibung
HouseNumberParity	Gibt an, ob der Hausnummernbereich gerade Zahlen, ungerade Zahlen oder beides umfasst. E Gerade O Ungerade B Beide
LastLine	Vollständige letzte Adresszeile (Ort, Bundesstaat-/land bzw. Provinz/Kanton und Postleitzahl).
LeadingDirectional	Die Straßenrichtung, die vor dem Straßennamen steht. Zum Beispiel das „N“ in 138 N Main Street.
Locality	In der Regel eine Lokalität in ländlichen Gebieten oder ein Vorort in Stadtgebieten.
NumberOfCandidateRanges	Gibt an, ob bei der Adresse eine Hausnummer angegeben ist. Zur Auswahl stehen: 0 Die Adresse hat keine Hausnummer. Adressen, die keine Hausnummer haben, sind beispielsweise Postfachadressen und allgemeine Lieferadressen. 1 Die Adresse hat eine Hausnummer. Informationen über den Bereich, in den die Hausnummer fällt, finden Sie bei den Feldern „HouseNumberHigh“, „HouseNumberLow“ und „HouseNumberParity“.
NumberOfRangeUnits	Gibt an, ob die Adresse eine Wohneinheitsnummer, z. B. eine Suite- oder Apartmentnummer, hat. Zur Auswahl stehen: 0 Die Adresse hat keine Wohneinheitsnummer. 1 Die Adresse hat eine Wohneinheitsnummer. Informationen über den Bereich, in den die Wohneinheit fällt, finden Sie unter „UnitNumberHigh“ und „UnitNumberLow“.
PostalCode	Die Postleitzahl zu der Adresse. Das Format der Postleitzahl variiert je nach Land.
PostalCode.Addon	Der zweite Teil der Postleitzahl. Bei kanadischen Adressen ist dieser beispielsweise die LDU. Dieses Feld wird von den meisten Ländern nicht verwendet.
PreAddress	Sonstige Informationen, die vor dem Straßennamen stehen.

Spaltenname	Beschreibung
PrivateMailbox	Dieses Feld wird derzeit nicht verwendet.
SegmentParity	Gibt an, welche Straßenseite gerade Nummern hat. L Linke Seite der Straße R Rechte Seite der Straße B Beide Seiten der Straße U Unbestimmt
StateProvince	Der Name des Bundesstaats.
StreetDataType	Der Rang der zum Geocodieren der Adresse verwendeten Datenbank in der Standardsuchreihenfolge. Der Wert „1“ bedeutet, dass die Datenbank an erster Stelle der Standardsuchreihenfolge steht. Der Wert „2“ bedeutet, dass die Datenbank an zweiter Stelle der Standardsuchreihenfolge steht, usw. Die Standardsuchreihenfolge für Datenbanken wird in Management Console über die Seite „Spectrum-Datenbanken“ angegeben.
StreetName	Der Straßename.
StreetPrefix	Der Straßentyp, wenn dieser vor dem Basisstraßennamen steht. Beispiel: AVENUE 12 AVENUE B KALGOORLIE WA 6430
StreetSuffix	Der Straßentyp des abgeglichenen Orts. Beispiel: AVE für „Avenue“.
TrailingDirectional	Die Straßenrichtung, die hinter dem Straßennamen steht. Beispiel: das „N“ in 456 Washington N.
UnitNumberHigh	Die höchste Nummer einer Wohneinheit in dem Bereich, in dem sich die Wohneinheit befindet.
UnitNumberLow	Die niedrigste Nummer einer Wohneinheit in dem Bereich, in dem sich die Wohneinheit befindet.

Geocode-Ausgabe

Tabelle 56: Geocode-Ausgabe

columnName	Beschreibung
CoordinateSystem	Das verwendete Koordinatensystem zur Bestimmung der Längengrad- und Breitengradkoordinaten. Ein Koordinatensystem gibt eine Kartenprojektion, Koordinateneinheiten usw. an. Ein Beispiel ist EPSG:4326. EPSG steht für die European Petroleum Survey Group (europäische Öl- und Gaserkundungsgruppe).
Latitude	Siebenstellige Zahl in Grad mit vier Dezimalstellen (im angegebenen Format).
Longitude	Siebenstellige Zahl in Grad mit vier Dezimalstellen (im angegebenen Format).

Ergebniscodes

Ergebniscodes enthalten Informationen über den erfolgreichen oder fehlgeschlagenen Geocoding-Versuch sowie über die Genauigkeit des Geocodes.

Anmerkung: Da das EGM-Modul administrative Aufgaben an eine webbasierte Management Console abgibt, haben die Optionen möglicherweise eine andere Bezeichnung als im Enterprise Designer. Es gibt allerdings keine Unterschiede beim Verhalten.

Tabelle 57: Ergebniscodeausgabe

columnName	Beschreibung
Geocoder.MatchCode	Gibt an, wie genau die Eingabeadresse mit der Kandidatenadresse übereinstimmt.
IsCloseMatch	Gibt an, ob die Adresse als hohe Übereinstimmung gilt. Eine Adresse wird basierend auf den Optionen „Kriterien für hohe Übereinstimmung“ auf der Registerkarte „Vergleiche“ als hohe Übereinstimmung gezählt. Y Ja, die Adresse weist eine hohe Übereinstimmung auf. N Nein, die Adresse weist keine hohe Übereinstimmung auf.

columnName	Beschreibung										
MultiMatchCount	<p>Beim Geocoding von Straßenanschriften ist dies die Anzahl an übereinstimmenden Adresspositionen, die für die angegebene Adresse gefunden wurden.</p> <p>Beim Kreuzungs-Geocoding ist dies die Anzahl an übereinstimmenden Adresspositionen, die für die angegebenen Adressen gefunden wurden.</p>										
Status	<p>Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs.</p> <table border="0"> <tr> <td>null</td> <td>Erfolg</td> </tr> <tr> <td>F</td> <td>Fehler</td> </tr> </table>	null	Erfolg	F	Fehler						
null	Erfolg										
F	Fehler										
Status.Code	<p>Wenn der Geocoder die Adresse nicht verarbeiten konnte, wird in diesem Feld der Grund dafür angezeigt.</p> <ul style="list-style-type: none"> • Interner Systemfehler • Kein Geocode gefunden • Nicht genügend Eingabedaten • Mehrere Übereinstimmungen gefunden • Eine Ausnahme ist aufgetreten • Geocoder kann nicht initialisiert werden • Keine Übereinstimmung gefunden 										
Status.Description	<p>Wenn der Geocoder die Adresse nicht verarbeiten konnte, wird in diesem Feld eine Beschreibung des Fehlers angezeigt.</p> <table border="0"> <tr> <td>Problem + Erklärung</td> <td>Wird zurückgegeben, wenn Status.Code = Interner Systemfehler.</td> </tr> <tr> <td>Geocoding Failed</td> <td>Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.</td> </tr> <tr> <td>No location returned</td> <td>Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.</td> </tr> <tr> <td>No Candidates Returned</td> <td>Der Geocoder konnte keine Kandidatentreffer für die Adresse ermitteln.</td> </tr> <tr> <td>Multiple Candidates Returned and Keep Multiple Matches not selected</td> <td>Die Adresse ergab mehrere Kandidaten. Damit die Kandidatenadresse zurückgegeben wird, müssen Sie <code>KeepMultimatch=Y</code> angeben.</td> </tr> </table>	Problem + Erklärung	Wird zurückgegeben, wenn Status.Code = Interner Systemfehler.	Geocoding Failed	Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.	No location returned	Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.	No Candidates Returned	Der Geocoder konnte keine Kandidatentreffer für die Adresse ermitteln.	Multiple Candidates Returned and Keep Multiple Matches not selected	Die Adresse ergab mehrere Kandidaten. Damit die Kandidatenadresse zurückgegeben wird, müssen Sie <code>KeepMultimatch=Y</code> angeben.
Problem + Erklärung	Wird zurückgegeben, wenn Status.Code = Interner Systemfehler.										
Geocoding Failed	Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.										
No location returned	Wird zurückgegeben, wenn Status.Code = Kein Geocode gefunden.										
No Candidates Returned	Der Geocoder konnte keine Kandidatentreffer für die Adresse ermitteln.										
Multiple Candidates Returned and Keep Multiple Matches not selected	Die Adresse ergab mehrere Kandidaten. Damit die Kandidatenadresse zurückgegeben wird, müssen Sie <code>KeepMultimatch=Y</code> angeben.										

columnName	Beschreibung
LocationPrecision	Ein Code, der die Genauigkeit des Geocodes angibt. Zur Auswahl stehen:
	0 Es sind keine Koordinateninformationen zu dieser Adresse verfügbar.
	1 Interpolierte Straßenanschrift.
	2 Mittelpunkt des Straßensegments.
	3 Mittelpunkt von Postleitzahl 1.
	4 Mittelpunkt der Teil-Postleitzahl 2.
	5 Mittelpunkt von Postleitzahl 2.
	6 Kreuzung.
	7 Point of Interest. Hierbei handelt es sich um einen Platzhalterwert. Spectrum-Datenbanken haben keine POI-Daten, sodass dies nicht zurückgegeben werden kann.
	8 Bundesland-/Kantonmittelpunkt.
	9 Bezirksmittelpunkt.
	10 Ortsmittelpunkt.
	11 Lokalitätsmittelpunkt.
	12–15 (LocationPrecision-Codes) Für die meisten Länder sind die LocationPrecision-Codes 12–15 für unbestimmte benutzerdefinierte Elemente reserviert.
	13 Zusätzliche Punktgenauigkeit für ein unbestimmtes benutzerdefiniertes Element.
	14 Zusätzliche Punktgenauigkeit für ein unbestimmtes benutzerdefiniertes Element.
	15 Zusätzliche Punktgenauigkeit für ein unbestimmtes benutzerdefiniertes Element.
	16 Das Ergebnis ist ein Adresspunkt.
	17 Das Ergebnis wurde generiert, indem die Segmentdaten des Kandidaten mithilfe von Adresspunktdaten geändert wurden.
	18 Das Ergebnis ist ein Adresspunkt, der mithilfe des Features „Versatz von Mittellinie“ projiziert wurde. Zur Verwendung des Features „Versatz von Mittellinie“ müssen Sie eine Punkt- und Straßenbereichsdatenbank haben, und darüber LocationPrecision 18 zurückgeben.

columnName	Beschreibung
StreetDataType	<p>Der Rang der zum Geocodieren der Adresse verwendeten Datenbank in der Standardsuchreihenfolge. Der Wert „1“ bedeutet, dass die Datenbank an erster Stelle der Standardsuchreihenfolge steht. Der Wert „2“ bedeutet, dass die Datenbank an zweiter Stelle der Standardsuchreihenfolge steht, usw.</p> <p>Die Standardsuchreihenfolge für Datenbanken wird in Management Console angegeben.</p>

G-NAF-Ausgabe

In der folgenden Tabelle werden Ausgabefelder aufgeführt, die ausschließlich für die australische Geocoded National Address File (G-NAF[®])-Datenbank gelten. G-NAF ist eine optionale Datenbank, die für alle sechs Bundesstaaten und zwei Territorien verfügbar ist. G-NAF ist der einzige verbindliche nationale Index von Lokalitäten, Straßen und Nummern für Australien, der mit geografischen Koordinaten überprüft wird.

Tabelle 58: Ausgabe aus australischer G-NAF

Spaltenname	Beschreibung												
AUS.GNAF_ADDRESS_CLASS	<p>Address_Class wird mithilfe einer Kombination von Elementen aus den G-NAF Data Dictionary-Quelltabellen erstellt. Die Komponenten des Feldes „Address_Class“ lauten wie folgt:</p> <table border="0"> <tr> <td>A</td> <td>Alias-Adressdatensatz</td> </tr> <tr> <td>P</td> <td>Hauptadressdatensatz</td> </tr> <tr> <td>PP</td> <td>Primärer Hauptadressdatensatz</td> </tr> <tr> <td>PS</td> <td>Sekundärer Hauptadressdatensatz</td> </tr> <tr> <td>AP</td> <td>Primärer Alias-Adressdatensatz</td> </tr> <tr> <td>AS</td> <td>Sekundärer Alias-Adressdatensatz</td> </tr> </table>	A	Alias-Adressdatensatz	P	Hauptadressdatensatz	PP	Primärer Hauptadressdatensatz	PS	Sekundärer Hauptadressdatensatz	AP	Primärer Alias-Adressdatensatz	AS	Sekundärer Alias-Adressdatensatz
A	Alias-Adressdatensatz												
P	Hauptadressdatensatz												
PP	Primärer Hauptadressdatensatz												
PS	Sekundärer Hauptadressdatensatz												
AP	Primärer Alias-Adressdatensatz												
AS	Sekundärer Alias-Adressdatensatz												

Spaltenname	Beschreibung
AUS.GNAF_CONFIDENCE	<p>Eine Zahl, die angibt, in wie vielen G-NAF-Datasets die Adresse gefunden wurde. Ein höheres Zuverlässigkeitsniveau bedeutet, dass dieselbe Adresse in mehreren Datenquellen von Mitwirkenden gefunden wurde. Zur Auswahl stehen:</p> <p><Anzahl> Die Anzahl der Datasets, in denen die Adresse gefunden wurde, abzüglich 1. Beispiel: Der Wert 0 gibt an, dass die Adresse im Dataset eines Mitwirkenden gefunden wurde; der Wert 1 gibt an, dass die Adresse in Datasets von zwei Mitwirkenden gefunden wurde; der Wert 2 gibt an, dass die Adresse in Datasets von drei Mitwirkenden gefunden wurde, usw.</p> <p>-1 Die Adresse konnte in keinem G-NAF-Dataset gefunden werden.</p>
AUS.GNAF_EIGHT_DECIMAL_PLACE_LATITUDE	<p>Der Breitengrad der Parzelle, mit einer Genauigkeit von acht Stellen nach dem Dezimalzeichen. Dies ist der Breitengrad, mit dem der Kandidat in der G-NAF-Datenbank abgeglichen wurde. Dies sind die direkt aus den G-NAF-Daten stammenden Originalkoordinaten vor dem Abschneiden oder Runden.</p> <p>Dieses Feld wird nur zurückgegeben, wenn Sie Return8DecimalPlaceLatLong=Y angeben.</p>
AUS.GNAF_EIGHT_DECIMAL_PLACE_LONGITUDE	<p>Der Längengrad der Parzelle, mit einer Genauigkeit von acht Stellen nach dem Dezimalzeichen. Dies ist der Längengrad, mit dem der Kandidat in der G-NAF-Datenbank abgeglichen wurde. Dies sind die direkt aus den G-NAF-Daten stammenden Originalkoordinaten vor dem Abschneiden oder Runden.</p> <p>Dieses Feld wird nur zurückgegeben, wenn Sie Return8DecimalPlaceLatLong=Y angeben.</p>

Spaltenname	Beschreibung
AUS.GNAF_GEOCODE_LEVEL	<p>Eine Zahl, die die Ebene des Geocodes für die Adresse angibt. Die einzelnen Hauptadressen innerhalb der G-NAF-Datenbank verfügen mindestens über einen Geocode auf Lokalitätsebene. Sie können auch über einen Geocode auf Straßenebene oder über einen Geocode auf Punktebene verfügen.</p> <p>Zur Auswahl stehen:</p> <ul style="list-style-type: none"> 0 Kein Geocode. 1 Nur Geocode auf Parzellenebene (kein Geocode auf Lokalitäts- oder Straßenebene). 2 Nur Geocode auf Straßenebene (kein Geocode auf Lokalitäts- oder Parzellenebene). 3 Geocodes auf Straßen- und Parzellenebene (kein Geocode auf Lokalitätsebene). 4 Nur Geocode auf Lokalitätsebene (kein Geocode auf Straßen- oder Parzellenebene). 5 Geocodes auf Lokalitäts- und Parzellenebene (kein Geocode auf Straßenebene). 6 Geocodes auf Lokalitäts- und Straßenebene (kein Geocode auf Parzellenebene). 7 Geocodes auf Lokalitäts-, Straßen- und Parzellenebene.
AUS_GNAF_PARCEL_ID	<p>Das Feld „Parzellen-ID“ ist das generische Parzellenkennungsfeld, das durch depotführende Daten bereitgestellt wird. Es stellt eine Plangrundstücksbeschreibung dar, die für Regierungsbehörden hilfreich ist. Das genaue Format variiert. G-NAF-Quelldaten enthalten über 7 Millionen Datensätze mit einer Parzellen-ID. Der australische Geocoder ergänzt dies, um mehr als 12.730.000 G-NAF-Datensätze mit einem Parzellen-ID-Feld auszufüllen.</p>
AUS.GNAF_PID	<p>Der G-NAF Persistent Identifier (G-NAF PID) ist eine 14-stellige alphanumerische Zeichenfolge, die jede G-NAF-Adresse eindeutig identifiziert. Der PID wird aus einer Kombination der Hauptadressfelder der G-NAF-Datenbank erstellt. Beispiel für einen G-NAF PID:</p> <p>GAVIC411711441</p>

Spaltenname

Beschreibung

AUS.GNAF_RELIABILITY

Eine Zahl, die die Genauigkeit des Geocodes angibt. Zuverlässigkeit bezieht sich auf das Verzeichnis, das für die Bestimmung des Geocodes verwendet wird. Daten mit den geocodierten Zuverlässigkeitsniveaus 1, 2 und 3 sind im Verzeichnis GNAF123 enthalten. Dies sind auf Punktebene (Parzelle) geocodierte Daten. Daten mit den geocodierten Zuverlässigkeitsniveaus 4, 5 und 6 sind im Verzeichnis GNAF456 enthalten. Es enthält geocodierte Daten ohne Parzellenmittelpunkt.

- 1** Die Genauigkeit des Geocodes wurde für einen angemessenen Überprüfungsstandard erfasst. Dies könnte beispielsweise für einen Geocode auf Adressebene gelten, der manuell geocodiert wurde. Die Geocode-Auflösung reicht aus, um den Mittelpunkt mit einem GPS innerhalb einer Adressgrenze zu platzieren.
- 2** Die Genauigkeit des Geocodes reicht aus, um den Mittelpunkt innerhalb einer Adressgrenze zu platzieren. Dies könnte beispielsweise für einen Geocode auf Adressebene gelten, der automatisch als Mittelpunkt der entsprechenden Katasterparzelle berechnet wurde.
- 3** Die Genauigkeit des Geocodes reicht aus, um den Mittelpunkt nahe (oder eventuell innerhalb von) einer Adressgrenze zu platzieren. Dies könnte beispielsweise für einen Geocode auf Adressebene gelten, der automatisch berechnet wurde, indem ermittelt wurde, wo in der Straße die Adresse basierend auf anderen geocodierten Adressgrenzen zu erwarten war.
- 4** Die Genauigkeit des Geocodes reicht aus, um der Adresse ein eindeutiges Straßenmerkmal zuzuordnen. Dies könnte beispielsweise für einen Geocode auf Straßenebene gelten, der automatisch anhand der Verweisdaten der Straßenmittellinie berechnet wurde.
- 5** Die Auflösung des Geocodes reicht aus, um der Adresse eine eindeutige Lokalität oder Umgebung zuzuordnen. Dies könnte beispielsweise für einen Geocode auf Lokalitätsebene gelten, der automatisch als Mittelpunkt der Lokalität berechnet wurde.
- 6** Die Auflösung des Geocodes reicht aus, um der Adresse eine eindeutige Region zuzuordnen. Dies könnte beispielsweise für einen Geocode auf Lokalitätsebene gelten, der über ein topographisches Merkmal abgeleitet wurde.

Spaltenname	Beschreibung
AUS.GNAF_SA1	<p>Das Feld „Statistical Area Level 1“ (SA1) ist der zweit kleinste geografische Bereich, der im Australian Statistical Geography Standard (ASGS) definiert ist. Meshblock ist die kleinste Einheit. SA1 dient der Nutzung bei der Bevölkerungs- und Unterkunftszählung als kleinste Einheit für die Verarbeitung und Herausgabe von Zensusdaten. SA1 wird durch einen eindeutigen 7-stelligen Code dargestellt.</p>
AUS.LEVEL_NUMBER	<p>Die Nummer einer Etage oder eines Stockwerks in einem mehrstöckigen Gebäude. Beispiel:</p> <p>Floor 2, 17 Jones Street</p> <p>Die G-NAF-Datenbank beinhaltet Etageninformationen für einige australische Bundesstaaten. Etageninformationen können optional mit Einheitsinformationen verknüpft werden. Falls die G-NAF-Datenbank mehrere Datensätze mit der gleichen Etage enthält, wird nur die Etageninformation zurückgegeben, wenn die Eingabeadresse einen eindeutigen Inhalt (wie eine Einheitsnummer) enthält. Falls die G-NAF-Datenbank Etageninformationen für eine Adresse enthält, werden diese Informationen mit dem verglichenen Kandidaten zurückgegeben.</p> <p>Die korrekten Etageninformationen werden zurückgegeben (falls verfügbar), selbst wenn die Eingabeadresse keine Etageninformationen enthielt oder falls die Eingabe die falschen Etageninformationen enthielt. Wenn die Eingabeadresse Etageninformationen enthält, aber die G-NAF-Datenbank für die übereinstimmende Adresse keine Etageninformationen enthält, werden die Eingabeetageninformationen verworfen, da sie nicht mit den G-NAF-Daten überprüft werden können.</p>

Spaltenname	Beschreibung
AUS.LEVEL_TYPE	<p>Die für eine Etage in einem mehrstöckigen Gebäude verwendete Bezeichnung. Beispiel: „Level“ (Stockwerk) oder „Floor“ (Etage). In dem folgenden Beispiel lautet der Stockwerkstyp „Level“:</p> <p>Suite 3 Level 7, 17 Jones Street</p> <p>In diesem Beispiel stellt „Suite 3“ eine Einheit dar.</p> <p>Die G-NAF-Datenbank beinhaltet Etageninformationen für einige australische Bundesstaaten. Etageninformationen können optional mit Einheitsinformationen verknüpft werden. Falls die G-NAF-Datenbank mehrere Datensätze mit der gleichen Etage enthält, wird nur die Etageninformation zurückgegeben, wenn die Eingabeadresse einen eindeutigen Inhalt (wie eine Einheitsnummer) enthält. Falls die G-NAF-Datenbank Etageninformationen für eine Adresse enthält, werden diese Informationen mit dem verglichenen Kandidaten zurückgegeben.</p> <p>Die korrekten Etageninformationen werden zurückgegeben (falls verfügbar), selbst wenn die Eingabeadresse keine Etageninformationen enthielt oder falls die Eingabe die falschen Etageninformationen enthielt. Wenn die Eingabeadresse Etageninformationen enthält, aber die G-NAF-Datenbank für die übereinstimmende Adresse keine Etageninformationen enthält, werden die Eingabeetageninformationen verworfen, da sie nicht mit den G-NAF-Daten überprüft werden können.</p>
AUS.MESH_BLOCK_ID	<p>Bei einem Meshblock handelt es sich um die kleinste geografische Einheit, für die das Australian Bureau of Statistics (ABS) statistische Daten erhebt. Meshblocks umfassen normalerweise mindestens 20 bis 50 Haushalte. Dies ist etwa ein Fünftel von der Größe eines „Collection District“ (CD). Sie können mithilfe der Meshblock-ID zusätzliche Zuordnungen für Ihre eigenen Daten durchführen.</p>
AUS.LOT_NUMBER	<p>Parzellennummern werden für die G-NAF-Kandidaten zurückgegeben, weil einige ländliche Adressen keine geeigneten physischen Daten oder Hausnummerndaten besitzen.</p>
AUS.STREET_TYPE_ABB	<p>Dies ist eine Abkürzung für den Straßentyp. EX ist beispielsweise eine Abkürzung für „Extension“ (Erweiterung) und FTRL eine Abkürzung für „Firetrail“ (Feuerschneise).</p>

Reverse Geocode Address Global

Informationen zur Verwendung der API für den Zugriff auf Reverse Geocode Address Global finden Sie in den Geocoding-Handbüchern.

Ergebniscodes für internationales Geocoding

Kandidaten, die durch die Spectrum Geocoder zurückgegeben werden, geben eine weitere Klasse von Rückgabecodes zurück, die als internationale Geocoding-Ergebniscodes bezeichnet werden. Bei jedem Vergleichsversuch wird im Ausgabefeld „Geocoder.MatchCode“ ein Ergebniscode zurückgegeben.

Internationale Straßen-Geocoding-Ergebniscodes (S-Codes)

Auf Straßenebene geocodierte Kandidaten geben einen Ergebniscode zurück, der mit dem Buchstaben „S“ beginnt. Das zweite Zeichen des Codes gibt die Positionsgenauigkeit des resultierenden Punktes für den geocodierten Datensatz an.

Tabelle 59: Ergebniscode auf Straßenebene (S)

S-Ergebniscode	Beschreibung
S1	Einzelne hohe Übereinstimmung mit dem Punkt im PLZ-Mittelpunkt.
S3	Einzelne hohe Übereinstimmung mit dem Punkt im PLZ-Mittelpunkt.
S4	Einzelne hohe Übereinstimmung mit dem Punkt im Straßenmittelpunkt. Für alte Datenbanken 2014 Q4 oder neuer wird mit dem Kandidaten die Eingabehausnummer zurückgegeben, auch wenn keine solche Hausnummer gefunden wurde. Auf den Code S4 folgen Buchstaben und Bindestriche. Diese geben die Übereinstimmungsgenauigkeit an. Siehe dazu Interpretieren von S-Ergebniscodes auf Seite 350.
S5	Einzelne hohe Übereinstimmung mit dem Punkt an einer Straßenadressenposition. Auf den Code S5 folgen Buchstaben und Striche. Diese geben die Übereinstimmungsgenauigkeit an. Weitere Informationen zu diesen Buchstaben finden Sie unter Interpretieren von S-Ergebniscodes auf Seite 350.

S-Ergebniscode	Beschreibung
S7	Einzelne Übereinstimmung mit dem Punkt an einem interpolierten Punkt entlang des Straßensegments des Kandidaten. Wenn der potenzielle Kandidat kein Adresspunktkandidat ist und es keine genauen Hausnummernübereinstimmungen unter anderen Adresspunktkandidaten gibt, wird das S7-Ergebnis mithilfe einer Adresspunktinterpolation zurückgegeben. Der Punkt wird gemäß dem nächsthöchsten oder -niedrigsten Adresspunktkandidaten interpoliert, der das Segment schneidet und dessen Hausnummer innerhalb des Häuserbereichs des ursprünglichen Kandidaten enthalten ist. Durch Verwendung bekannter Adressreferenzpunkte im Straßensegment kann der S7-Punkt auf eine genauere Position angepasst werden.
S8	Einzelne hohe Übereinstimmung mit dem Punkt entweder an dem einzelnen Punkt, der einem Adresspunktkandidaten zugeordnet ist, oder an einem Adresspunktkandidaten, der dieselbe Hausnummer hat. Es ist keine Interpolation erforderlich. S8-Rückgaben sind nur mit Punktdatenbanken möglich.
SX	Einzelne hohe Übereinstimmung mit dem Punkt an der Straßenkreuzung.

Interpretieren von S-Ergebniscodes

Bei internationalen S-Ergebniscodes (geocodierte Straßen) beschreiben acht zusätzliche Zeichen, wie hoch die Übereinstimmung der Adresse mit einer Adresse aus der Datenbank ist. Die Zeichen werden in der Reihenfolge angezeigt, in der sie in der folgenden Tabelle aufgelistet sind. Alle nicht übereinstimmenden Adresselemente werden durch einen Strich dargestellt.

Der Ergebniscode „S5--N-SCZA“ stellt beispielsweise eine einzelne hohe Übereinstimmung dar, bei der der Straßename, das Straßensuffix (Richtung), der Ort und die Postleitzahl übereinstimmen. Die Striche geben an, dass es keine Übereinstimmung bei der Hausnummer, der vorangestellten Straßenrichtungsangabe oder dem Landstraßentyp gab. Die Übereinstimmung wurde in der Adressdatenbank für Straßenbereiche gefunden. Dieser Datensatz wird an der Straßenanschriftsposition des Kandidaten mit der Übereinstimmung geocodiert.

Kategorie	Beschreibung	Beispiel
H	Hausnummer	18

Kategorie	Beschreibung	Beispiel
P	<p>Vorangestellte Straßenrichtungsangabe</p> <p>„P“ ist vorhanden, wenn eine der folgenden Bedingungen erfüllt wurde:</p> <ul style="list-style-type: none"> • Der Kandidat stimmt bei der vorangestellten Straßenrichtungsangabe mit der Eingabe überein. • Die nachgestellte Richtungsangabe des Kandidaten stimmt mit der eingegebenen vorgestellten Richtungsangabe überein, nachdem voran- und nachgestellte Richtungsangaben getauscht wurden. • Die Eingabe weist keine vorangestellte Richtungsangabe auf. 	Norden
N	Straßenname	Merivale
T	Straßentyp	St
S	<p>Straßensuffix (Richtung)</p> <p>„S“ ist im Ergebniscode vorhanden, wenn eine der folgenden Bedingungen erfüllt wurde:</p> <ul style="list-style-type: none"> • Die nachgestellte Richtungsangabe des Kandidaten stimmt mit der eingegebenen nachgestellten Richtungsangabe überein. • Die vorgestellte Richtungsangabe des Kandidaten stimmt mit der eingegebenen nachgestellten Richtungsangabe überein, nachdem voran- und nachgestellte Richtungsangaben getauscht wurden. • Die Eingabe weist keine nachgestellte Richtungsangabe auf. 	W
C	Ortsname	South Brisbane
Z	Postleitzahl	4101
A, G oder U	<p>Zum Erzielen der Übereinstimmung verwendeter Datenbanktyp.</p> <ul style="list-style-type: none"> • A: Adressdatenbank für Straßenbereiche. • U: Kundendatenbank (benutzerdefiniert). 	A

Internationale postalische Geocoding-Ergebniscodes (Z-Codes)

Übereinstimmungen in der Kategorie Z geben an, dass auf PLZ-Code-Ebene eine Übereinstimmung gefunden wurde. Eine Übereinstimmung auf PLZ-Code-Ebene wird in einem der folgenden Fälle zurückgegeben:

- Sie haben angegeben, dass mit PLZ-Mittelpunkten verglichen werden soll. Der resultierende Punkt befindet sich im PLZ-Mittelpunkt mit den folgenden möglichen Genauigkeitsstufen.
- Es gibt keine hohe Übereinstimmung auf Straßenebene, und Sie haben als Fallback-Option PLZ-Mittelpunkt angegeben.

Tabelle 60: Postalische Ergebniscodes (Z)

Ergebniscode Z	Beschreibung
Z1	PLZ-Mittelpunkt-Übereinstimmung.
Z3	Vollständige PLZ-Mittelpunkt-Übereinstimmung. Für Kanada ist das ein FSALDU-Mittelpunkt.

Auf postalischer Ebene geocodierte Kandidaten geben einen Ergebniscode zurück, der mit dem Buchstaben Z beginnt. Geocode Address World kann einen Ergebniscode Z1 generieren. Landesspezifische Geocoder können oftmals genauere Postleitzahlergebnisse generieren (mit Ergebniscodes Z2 oder Z3).

Wenn der postalische Kandidat aus einem Benutzerverzeichnis stammt, wird der Buchstabe U an das Ergebnis angefügt. Zum Beispiel gibt Z1U eine Übereinstimmung beim postalischen Mittelpunkt aus einem benutzerdefinierten Benutzerverzeichnis an.

Internationale geografische Geocoding-Ergebniscodes (G-Codes)

Auf geografischer Ebene geocodierte Kandidaten geben einen Ergebniscode zurück, der mit dem Buchstaben „G“ beginnt. Die Zahlen nach dem G im Ergebniscode liefern detailliertere Informationen über die Genauigkeit des Kandidaten.

Tabelle 61: Geografische Ergebniscodes (G)

G-Ergebniscode	Beschreibung
G1	Mittelpunkt des Bundeslandes oder Kantons. stimmt überein.
G2	Übereinstimmung mit dem Mittelpunkt des Bezirks (Bezirk oder Region).
G3	Übereinstimmung mit dem Mittelpunkt des Orts oder der Stadt (Gemeinde).

G-Ergebniscode	Beschreibung
G4	Übereinstimmung mit dem Mittelpunkt einer Lokalität (Dorf, Vorort oder Umgebung).

Wenn der geografische Kandidat aus einem Benutzerverzeichnis stammt, wird der Buchstabe U an den Ergebniscode angefügt. Zum Beispiel gibt G4U eine Übereinstimmung beim Lokaliätsmittelpunkt aus einem benutzerdefinierten Benutzerverzeichnis an.

Reverse Geocoding-Codes (R-Codes)

Übereinstimmungen in der Kategorie R geben an, dass der Datensatz mit Reverse Geocoding abgeglichen wurde. Die zweiten zwei Zeichen des R-Ergebniscodes geben den Typ der gefundenen Übereinstimmung an. R-Geocode-Ergebnisse enthalten einen zusätzlichen Buchstaben, der das Verzeichnis angibt, aus dem der Vergleich durchgeführt wurde.

Beispiele für Reverse Geocoding-Codes:

Tabelle 62: Reverse Geocoding-Ergebniscodes (R)

Reverse Geocoding-Code	Beschreibung
RS8A	Genauigkeit auf Punkt-/Parzellenebene für Reverse Geocoding. Aus dem Adresswörterbuch zurückgegebener Kandidat.
RS5A	Interpolierter Straßenkandidat für Reverse Geocoding. Aus dem Adresswörterbuch zurückgegebener Kandidat.
RS4A	Straßenmittelpunkt-Kandidat für Reverse Geocoding. Aus dem Adresswörterbuch zurückgegebener Kandidat.

Wenn der Reverse Geocoding-Kandidat aus einem Benutzerverzeichnis stammt, wird der Buchstabe U an das Ergebnis angefügt. Zum Beispiel gibt RS8U eine Reverse Geocoding-Übereinstimmung auf Punkt-/Parzellenebene aus einem benutzerdefinierten Benutzerverzeichnis an.

Codes ohne Übereinstimmung

Die folgenden Ergebniscode geben an, dass keine Übereinstimmung gefunden wurde:

- **N**: Keine hohe Übereinstimmung.
- **NX**: Keine hohe Übereinstimmung bei Straßenkreuzungen.
- **ND**: Spectrum™ Technology Platform konnte die Geocoding-Datenbank für die angegebene Postleitzahl oder Gemeinde/Bundesland/Kanton nicht finden.

GeoConfidence-Modul

GeoConfidence-Modul

Das GeoConfidence-Modul wird verwendet, um die Wahrscheinlichkeit zu bestimmen, dass eine Adresse oder Straßenkreuzung innerhalb eines gegebenen Bereichs liegt. Das Modul nimmt eine Adresse oder den Standort einer Kreuzung (durch Geocode US Address bestimmt), wandelt diesen Standort in einen Punkt, eine Linie oder ein Polygon um (je nach Genauigkeit der Übereinstimmung), und vergleicht anschließend diese Form mit einer Datenbank mit bekannten Formen, um auf eine Überlappung zu prüfen und die prozentuale Überlappung anzugeben. Sie könnten beispielsweise das GeoConfidence-Modul verwenden, um Entscheidungen zur Flutgebietenbewertung basierend darauf treffen, wie groß die Überlappung zwischen einem Adressstandort und den Flutgebieten ist. Alles über 95 % Überlappung in einem 100-jährigen Flutgebiet kann angeben, dass die Adresse im Flutgebiet liegt. Im Gegensatz könnten Werte kleiner als 95 % dazu führen, dass ein Geschäftsprozess die Adresse zur Ausnahmenverarbeitung schickt, die eine manuelle Überprüfung beinhalten könnte.

Eine Adresse oder Kreuzung kann zu einem Punkt, einer Adresse entlang eines Straßensegments (eine Anordnung von Straßensegmentpunkten), einem ZIP + 4-Mittelpunkt, einem ZIP + 2-Mittelpunkt oder einem PLZ-Gebiet-Mittelpunkt (Polygone) geocodiert werden. Sie können diese Formen (Punkte, Linien oder Polygone) verwenden, um sie mit anderen Formen zur Bestimmung einer Überlappung zu verwenden, die wiederum zur Bestimmung eines Risikos oder einer Wahrscheinlichkeit verwendet werden kann.

Unterschiedliche GeoConfidence-Polygone werden in Abhängigkeit zum GeoConfidence-Ergebnis generiert, das durch das Enterprise Geocoding-Modul zurückgegeben wird. Weitere Informationen über die vom Enterprise Geocoding-Modul zurückgegebenen GeoConfidence-Daten finden Sie in der Dokumentation des Enterprise Geocoding-Moduls.

Das GeoConfidence-Modul unterstützt nur US-amerikanische Standorte.

Anmerkung: GeoConfidence verwendet Dienste, die vom Enterprise Geocoding- und Location Intelligence-Modul bereitgestellt werden.

Komponenten

GeoConfidence stellt drei Datenflüsse bereit, die Sie im Enterprise Designer ändern können. Jeder Datenfluss besteht aus verschiedenen Komponenten, die mit den Enterprise Geocoding- und Location Intelligence-Modulen installiert wurden.

Informationen über jede Komponente in installierten Datenflüssen finden Sie im relevanten Komponentenkapitel im *Spectrum™ Technology Platform-Benutzerhandbuch*

Die Namen der Datenflüsse lauten:

- **GeoConfidenceSurface**: Dies ist der Datenfluss, der die GeoConfidenceSurface erstellt, die für weitere Analysen verwendet werden können. Die Eingabe ist die GeoConfidence-Information, die vom Enterprise Geocoding-Modul zurückgegeben wird. Momentan kann nur der „Geocode US Address“-Schritt diese Information zurückgeben.
- **CreatePointsConvexHull**: Dies ist ein Datenfluss, der von der GeoConfidenceSurface-Vorlage verwendet wird. Sie müssen höchstwahrscheinlich keine Änderungen an diesem Unterfluss durchführen.
- **FloodRiskAnalysis**: Dies ist ein Beispieldatenfluss.

GeoConfidence-Datenbanken

GeoConfidence verwendet die gleichen Datenbanken wie die Enterprise Geocoding- und Location Intelligence-Module.

Informationen über das Hinzufügen dieser Datenbanken finden Sie im *Spectrum™ Technology PlatformAdministratorhandbuch*.

Neben diesen Datenbanken beinhaltet das GeoConfidence-Modul eine Datenbank mit PLZ-Codepolygonen. Diese Datenbank wird von GeoConfidenceSurface verwendet.

GeoConfidenceSurface

GeoConfidenceSurface gibt GeoConfidence-Polygone (auch Flächen genannt) basierend auf der Qualität der Geocodeinformationen zurück, die vom Enterprise Geocoding-Modul generiert wurden. Mit den generierten GeoConfidence-Polygonen können Sie anschließend dieses Polygon mit anderen räumlichen Daten überlappen, um ein Risiko oder eine Wahrscheinlichkeit zu bestimmen.

Dieser Dienst wird von der FloodZoneAnalysis-Datenflussvorlage des GeoConfidence-Moduls verwendet.

Anmerkung: GeoConfidence verwendet Dienste, die vom Enterprise Geocoding- und Location Intelligence-Modul bereitgestellt werden.

Eingabe

Die Eingabefelder für GeoConfidenceSurface sind die Ausgabefelder, die durch die GeoConfidence-Ausgabekategorie des Enterprise Geocoding-Moduls zurückgegeben wurden. Diese Felder werden unten beschrieben.

columnName	Max. Feldlänge mit NULL-Wert	Beschreibung
GeoConfidenceCode	13	<p>Der in diesem Feld zurückgegebene Wert gibt an, welcher GeoConfidence Surface-Typ zurückgegeben wurde.</p> <p>Die folgenden Werte sind möglich:</p> <p>INTERSECTION Ein Geocode-Punkt für die Kreuzung zweier Straßen.</p> <p>ADDRESS Ein Array von Straßensegmentpunkten, das das Straßensegment darstellt, in dem sich die Adresse befindet.</p> <p>POINT Wenn der Geocoder die Adresse mithilfe von Punktdaten vergleichen konnte, dann ist dies die Punktgeometrie, an der sich die Adresse befindet.</p> <p>POSTAL1 Ein Geocode-Punkt für den PLZ-Mittelpunkt.</p> <p>POSTAL2 Ein Array von Punkten für alle Straßensegmente in ZIP + 2, wo sich die Adresse befindet.</p> <p>POSTAL3 Ein Array von Punkten für alle Straßensegmente in ZIP + 4, wo sich die Adresse befindet.</p> <p>ERROR Ein Fehler ist aufgetreten.</p>
StreetSegmentPoints	1024	<p>Ein Array von Breitengrad-/Längengradwerten, die die Straßensegmentpunkte darstellen.</p> <p>Anmerkung: Dieses Feld enthält nur dann Werte, wenn das Feld „GeoConfidenceCode“ folgenden Wert zurückgibt: ADDRESS, POSTAL2 oder POSTAL3.</p>
GeoConfidenceCentroidLatitude	11	Der Breitenrad des Mittelpunktes des GeoConfidence-Polygons.
GeoConfidenceCentroidLongitude	12	Der Längengrad des Mittelpunktes des GeoConfidence-Polygons.

Ausgabe

Das GeoConfidenceSurface-Ausgabefeld enthält das GeoConfidence-Polygon.

Spaltenname	Beschreibung
Geometry	Ein GeoConfidence-Polygon, das die zurückgegebene Geometrie darstellt.

Anpassen des GeoConfidence-Moduls

Das GeoConfidence-Modul stellt drei Datenflussvorlagen bereit, die Sie im Enterprise Designer ändern können. Jeder Datenfluss besteht aus verschiedenen Komponenten, die mit den Enterprise Geocoding- und Location Intelligence-Modulen installiert wurden.

Die Namen der Datenflussvorlagen lauten:

- **GeoConfidenceSurface**: Dies ist die Vorlage zur Erstellung der GeoConfidenceSurface, die für weitere Analysen verwendet werden kann. Die Eingabe ist die GeoConfidence-Information, die vom Enterprise Geocoding-Modul zurückgegeben wird. Momentan kann nur der „Geocode US Address“-Schritt diese Information zurückgeben. Zur Anpassung dieser Vorlage müssen Sie mindestens die räumliche 5-stellige PLZ-Codequelle im PLZ-Schritt (Query Spatial) angeben.
- **CreatePointsConvexHull**: Dies ist ein Datenfluss, der von der GeoConfidenceSurface-Vorlage verwendet wird. Sie müssen höchstwahrscheinlich keine Änderungen an diesem Unterfluss durchführen.
- **FloodRiskAnalysis**: Dies ist eine Beispielvorlage. Zur Anpassung dieser Vorlage müssen Sie mindestens die räumliche Flood-Quelle im Schritt „Find Nearest“ angeben.

Universal Addressing-Modul

Universal Addressing-Modul

Bei dem Universal Addressing-Modul handelt es sich um ein Adressqualitätsmodul, mit dem Adressen standardisiert und geprüft werden können, um somit die Zustellbarkeit von Postsendungen zu verbessern. Über das Universal Addressing-Modul kann sichergestellt werden, dass Ihre Adressdaten den Qualitätsstandards entsprechen, die von der Postbehörde aufgestellt wurden. Bei einer Adresse, die diesen Standards entspricht, ist die Wahrscheinlichkeit für eine pünktliche Lieferung höher. Außerdem können Mailer, die diese Standards befolgen, für erhebliche Portoermäßigungen berechtigt sein. Informationen zu Ermäßigungen für Briefe in den USA finden Sie im *USPS Domestic Mail Manual (DMM)*, das unter www.usps.com verfügbar ist. Informationen zu Ermäßigungen für Briefe in Kanada finden Sie auf der Website der kanadischen Post unter www.canadapost.ca.

Informationen zu Ermäßigungen für Briefe in Australien finden Sie auf der Website der australischen Post unter www.auspost.com.au.

Das Universal Addressing-Modul kann, abhängig von den von Ihnen lizenzierten Optionen, im Batch-Modus, im Echtzeitmodus oder als gehosteter Dienst verwendet werden. Die Batch-Version des Universal Addressing-Moduls ist durch den USPS® CASS-zertifiziert™. Sie ist außerdem durch die australische Post AMAS-zertifiziert.

Das Universal Addressing-Modul ist eines von zwei Adressqualitätsmodulen, die für Spectrum™ Technology Platform verfügbar sind. Das andere Adressqualitätsmodul, das Address Now-Modul, bietet erweiterte Unterstützung für Adressen außerhalb der USA und Kanada, einschließlich der Überprüfung für mehr Länder und Doppelbyte-Unterstützung. Wenn Sie über eine große Menge internationaler Adressdaten verfügen, möchten Sie vielleicht das Address Now-Modul zur Standardisierung und Überprüfung von Adressen in Betracht ziehen.

Komponenten

Das Universal Addressing-Modul setzt sich aus den folgenden Komponenten zusammen. Diese Komponenten funktionieren mit US-amerikanischen, kanadischen und internationalen Adressen, so lange Sie für die entsprechende Datenbank lizenziert sind (falls Sie Universal Addressing in Ihrer eigenen Umgebung ausführen) oder einen gehosteten Dienst nutzen (falls Sie Universal Addressing über die gehosteten Pitney Bowes-Dienste nutzen).

- **AutoCompleteLoqate**: Bietet eine Echtzeit-Eingabe von Adressdaten und gibt sofort Ergebnisse basierend auf jedem Zeichen zurück, das in das Formular eingegeben wurde. Dadurch wird sichergestellt, dass nur genaue Daten in die Datenbank eingegeben werden.
- **GetCandidateAddresses**: Gibt eine Liste aller möglichen Übereinstimmungen für eine angegebene Adresse zurück.
- **GetCandidateAddressesLoqate**: Gibt mithilfe einer Loqate-Engine und -Datenbank eine Liste aller möglichen Übereinstimmungen für eine angegebene Adresse zurück.
- **GetCityStateProvince**: Gibt den Ort und das Bundesland oder den Kanton für eine angegebene Postleitzahl zurück.
- **GetCityStateProvinceLoqate**: Gibt mithilfe einer Loqate-Engine und -Datenbank den Ort und das Bundesland oder den Kanton für eine angegebene Postleitzahl zurück.
- **GetPostalCodes**: Gibt die Postleitzahlen für einen angegebenen Ort zurück.
- **GetPostalCodesLoqate**: Gibt mithilfe einer Loqate-Engine und -Datenbank die Postleitzahlen für einen gegebenen Ort zurück.
- **ValidateAddress**: Standardisiert und überprüft Adressen mithilfe von US-amerikanischen, kanadischen und internationalen Postdaten.
- **ValidateAddressAUS**: Standardisiert und überprüft Adressen mithilfe von australischen Postdaten.
- **ValidateAddressGlobal**: „ValidateAddressGlobal“ bietet eine erweiterte Adressenstandardisierung und -überprüfung für Adressen außerhalb der USA und Kanada. „ValidateAddressGlobal“ kann auch Adressen in den USA und Kanada überprüfen. Die Stärke liegt jedoch in der Überprüfung von Adressen in anderen Ländern. Wenn Sie eine große Anzahl an Adressen außerhalb der USA und Kanada verarbeiten, sollten sie in Erwägung ziehen, „ValidateAddressGlobal“ zu verwenden.

- **ValidateAddressLoqate:** „ValidateAddressLoqate“ standardisiert und überprüft Adressen mithilfe der Adressdaten von Postbehörden. „ValidateAddress Loqate“ kann Daten korrigieren und die Adresse entsprechend dem von der jeweiligen Postbehörde bevorzugten Format formatieren. Diese Komponente fügt zudem fehlende postalische Daten wie Postleitzahlen, Ortsnamen, Namen von Bundesländern oder Kantonen usw. hinzu.

Universal Addressing-Datenbanken

Das Universal Addressing-Modul nutzt eine Vielzahl von erforderlichen und optionalen Datenbanken. Die Datenbanken werden auf dem Spectrum™ Technology Platform-Server installiert. Manche der Datenbanken sind per Abonnement bei Pitney Bowes verfügbar und werden monatlich oder vierteljährlich aktualisiert. Andere werden vom USPS® lizenziert. Die folgende Tabelle enthält die Universal Addressing-Datenbanken.

Tabelle 63: Datenbanken für das Universal Addressing-Modul

Datenbankname und -beschreibung	Erforderlich oder optional	Anbieter
<p>US-amerikanische Postdatenbank</p> <p>Der US-amerikanische Postdatenbank ist im eigenen Format von Pitney Bowes. Sie enthält jeden Hausnummernbereich in den Vereinigten Staaten und wird monatlich aktualisiert. Die Datenbankdateien enthalten die folgenden Informationen:</p> <ul style="list-style-type: none"> • ZIP + 4®-Code • Standardisierte Adresselemente • Orts- und Bundesstaatsinformationen <p>Der US-amerikanische Postdatenbank enthält außerdem die notwendigen Daten für die Durchführung von erweiterten Straßenvergleichen (ESM) und Vergleichen aller Straßen (ASM). ESM und ASM wenden eine zusätzliche Vergleichslogik auf eine beliebige Eingabeadresse an, die nicht durch den regulären Adressenüberprüfungsvorgang verglichen wird.</p>	Erforderlich für die Verarbeitung von US-amerikanischen Adressen	Monatliches Pitney Bowes-Abonnement
<p>Postdatenbank für Kanada</p> <p>Die kanadische Postdatenbank ist im eigenen Format von Pitney Bowes. Die Datenbankdateien enthalten die folgenden Informationen:</p> <ul style="list-style-type: none"> • Postleitzahl • Standardisierte Adresselemente • Gemeinde- und Provinzinformationen 	Zur Verarbeitung von kanadischen Adressen erforderlich	Monatliches Pitney Bowes-Abonnement

Datenbankname und -beschreibung	Erforderlich oder optional	Anbieter
<p>Postal Address File-Datenbank von Australia Post</p> <p>Die Postal Address File ist Teil des „Address Matching Approval System“ (AMAS)-Programms der Australia Post. Die Datenbankdatei enthält die folgenden Informationen:</p> <ul style="list-style-type: none"> • Postleitzahl • Standardisierte Adresselemente 	Zur Verarbeitung von australischen Adressen erforderlich	Monatliches Pitney Bowes-Abonnement
<p>Internationale Postdatenbank</p> <p>Die internationale Postdatenbank ist eine Sammlung von Postadressendaten aus der ganzen Welt. Daten aus jedem Land sind gemäß der verfügbaren Datenebene kategorisiert. Die Kategorien sind:</p> <ul style="list-style-type: none"> • Kategorie A: Ermöglicht die Überprüfung und Korrektur von Postleitzahl, Ortsname, Bundesland/Bezirk/Kanton, Straßenanschriftselementen und Ländername einer Adresse. • Kategorie A: Ermöglicht die Überprüfung und Korrektur von Postleitzahl, Ortsname, Bundesland/Bezirk/Kanton und Ländername einer Adresse. Sie unterstützt nicht die Überprüfung oder Korrektur von Straßenanschriftselementen. • Kategorie C: Ermöglicht die Überprüfung und Korrektur des Ländernamens sowie die Überprüfung des Formats der Postleitzahl. 	Zur Verarbeitung von internationalen Adressen erforderlich	Vierteljährliches Pitney Bowes-Abonnement

Datenbankname und -beschreibung	Erforderlich oder optional	Anbieter
<p>DPV®-Datenbank</p> <p>Die Delivery Point Validation-Datenbank ermöglicht Ihnen, die Gültigkeit einer einzelnen Postanschrift in den USA zu überprüfen. Die DPV-Datenbank verbessert die Fähigkeit der US-Postdatenbank, Postadressen zu überprüfen.</p> <p>Anmerkung: Die DPV-Datenbank enthält außerdem die für die „Commercial Mail Receiving Agency“ (CMRA)-Verarbeitung erforderlichen Daten.</p>	<p>Optional, aber für die CASS-zertifizierte™ Verarbeitung erforderlich; nur US-amerikanische Adressen</p>	<p>Monatliches Pitney Bowes-Abonnement</p>
<p>Bei jeder Veröffentlichung einer US-Postdatenbankversion wird eine entsprechende Ausgabe der DPV-Datenbank herausgegeben. Obwohl die USPS-Lizenzierung den Gebrauch der US-Postdatenbank über das Ablaufdatum hinaus erlaubt (mit bestimmten Einschränkungen), können DPV-Suchen nach dem Ablauf der DPV-Datenbank nicht mehr ausgeführt werden.</p>		
<p>Die USPS-Lizenzierung verbietet die Verwendung von DPV-Daten für die Erstellung von Adressen und Adresslisten. Zur Verhinderung der Erstellung von Adresslisten enthält die DPV-Datenbank „falsch-positive Datensätze“. Diese Datensätze sind künstlich erzeugte Adressen. Für jede negative Antwort, die auf eine DPV-Abfrage folgt, erfolgt eine Abfrage in der Falsch/Positiv-Tabelle in der DPV-Datenbank. Eine Übereinstimmung mit dieser Tabelle stoppt die DPV-Verarbeitung.</p>		
<p>Die USPS-Lizenzierung verbietet außerdem den Export von DPV-Daten außerhalb der Vereinigten Staaten.</p>		
<p>eLOT®-Datenbank</p> <p>Die Enhanced Line of Travel (eLOT)-Datenbank ist eine US-amerikanische Adressdatenbank, die sicherstellt, dass die „Enhanced Carrier Route“-Zustellungen so nahe wie möglich an der tatsächlichen Zustellungssequenz sortiert werden. Die eLOT-Datenbank ist für bestimmte Typen von Postrabatten erforderlich.</p> <p>Sie erhalten monatliche Aktualisierungen für Ihre eLOT-Datenbank auf dem gleichen Medium wie die US-Postdatenbank.</p> <p>Sie müssen die US-Postdatenbank und eLOT-Datenbank des gleichen Monats installieren (d. h. eLOT-Daten für September müssen mit einer US-Postdatenbank für September verarbeitet werden). Wenn die US-Postdatenbank und die eLOT-Datenbank nicht aus dem gleichen Monat stammen, könnte es ZIP + 4®-Codes geben, denen keine eLOT-Nummern zugewiesen werden können. Der ZIP Code™, ZIP + 4-Code, Beförderungsroutencode und Zustellort einer Adresse müssen bereitgestellt werden, um einen eLOT-Code zuzuweisen.</p>	<p>Optional; nur US-amerikanische Adressen</p>	<p>Monatliches Pitney Bowes-Abonnement</p>

Datenbankname und -beschreibung	Erforderlich oder optional	Anbieter
<p>EWS-Datenbank</p> <p>Die „Early Warning System“ (EWS)-Datenbank verhindert Adressenüberprüfungsfehler, die dann auftreten können, wenn Postdaten zu spät in der US-Postdatenbank ankommen.</p> <p>Die EWS-Datenbank besteht aus unvollständigen Adressinformationen, die auf ZIP Code™, Straßename, voran- und nachgestellte Richtungsangaben und ein Suffix begrenzt sind. Damit sich ein Adressdatensatz für EWS eignet, darf die Adresse nicht in der aktuellsten monatlichen Version der US-Postdatenbank enthalten sein.</p> <p>Das USPS® aktualisiert die EWS-Datei wöchentlich (donnerstags). Sie können die EWS-Datei von der USPS®-Website unter ribbs.usps.gov herunterladen.</p>	Optional; nur US-amerikanische Adressen	Kostenloser Download von der USPS®-Website
<p>LACS^{Link}®-Datenbank</p> <p>Die LACS^{Link}-Datenbank ermöglicht Ihnen, Adressen zu korrigieren, die sich infolge einer ländlichen Routenadresse, die in die Straßenadresse konvertiert wird, einer Postfach-Umnummerierung oder einer Straßenadressenänderung geändert haben.</p> <p>Die USPS-Lizenzierung verbietet die Verwendung von LACS^{Link} für die Erstellung von Adressen und Adresslisten. Zur Verhinderung der Erstellung von Adresslisten enthält die LACS^{Link}-Datenbank „falsch-positive Datensätze“. Diese Datensätze sind künstlich erzeugte Adressen. Für jede negative Antwort, die auf eine LACS^{Link}-Abfrage folgt, erfolgt eine Abfrage in der Falsch/Positiv-Tabelle in der LACS^{Link}-Datenbank. Eine Übereinstimmung mit dieser Tabelle stoppt die LACS^{Link}-Verarbeitung.</p> <p>Die USPS-Lizenzierung verbietet außerdem den Export von LACS^{Link}-Daten außerhalb der Vereinigten Staaten.</p>	Optional, aber für die CASS-zertifizierte™ Verarbeitung erforderlich; nur US-amerikanische Adressen	Monatliches Pitney Bowes-Abonnement
<p>RDI™-Datenbank</p> <p>Die „Residential Delivery Indicator“ (RDI™)-Datenbank enthält Daten, die bei der Bestimmung der besten Kosten für den Versand Ihrer Pakete helfen.</p> <p>RDI ähnelt DPV darin, dass RDI-Daten als Hash-Tabellen bereitgestellt werden. RDI ist jedoch ein viel einfacherer Vorgang als DPV, da der standardmäßige Hash-Algorithmus nur für den 9-stelligen und 11-stelligen ZIP Code™ anstatt für die gesamte Adresse bestimmt wird.</p>	Optional; nur US-amerikanische Adressen	Lizenz direkt von USPS®

Datenbankname und -beschreibung	Erforderlich oder optional	Anbieter
Suite^{Link™}-Datenbank Suite ^{Link™} korrigiert sekundäre Adressinformationen für US-amerikanische Firmenadressen, deren sekundäre Adressinformationen nicht überprüft werden konnten. Wenn die Suite ^{Link™} -Verarbeitung aktiviert ist, versucht „ValidateAddress“ den Wert des Feldes „FirmName“ mit einer Datenbank für bekannte Firmennamen zu vergleichen. „ValidateAddress“ stellt anschließend die korrekten sekundären Adressinformationen bereit.	Optional; nur US-amerikanische Adressen	Monatliches Pitney Bowes-Abonnement

„AutoCompleteLoqate“

„AutoCompleteLoqate“ bietet einen Echtzeit-Eintrag von Adressdaten zur Erzielung schneller, genauer Ergebnisse. Benutzern werden sofort Ergebnisse basierend auf jedem Zeichen zurückgegeben, das in das Formular eingegeben wurde. Dadurch wird sichergestellt, dass nur genaue Daten in die Datenbank eingegeben werden. „AutoCompleteLoqate“ umfasst zudem die Powersuchoption, durch die die Eingabezeit bei 238 Ländern um bis zu 80 % reduziert wird, indem Daten in Form einer Indexdatei verwendet werden.

Eingabe

In der folgenden Tabelle wird die Eingabe für „AutoCompleteLoqate“ aufgeführt.

Tabelle 64: Eingabeformat

columnName	Beschreibung
AddressLine1	Die erste Adresszeile.
AddressLine2	Die zweite Adresszeile.
AddressLine3	Die dritte Adresszeile.
AddressLine4	Die vierte Adresszeile.

columnName	Beschreibung
City	Der Ortsname.
Country	<p>Der Ländercode oder Ländername, in einem der folgenden Formate:</p> <ul style="list-style-type: none"> • zweistelliger ISO-Ländercode • dreistelliger UPU-Ländercode • Englischer Ländername <p>Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.</p>
FirmName	Der Unternehmens- bzw. Firmenname.
PostalCode	Die Postleitzahl zu der Adresse.
StateProvince	Das Bundesland oder der Kanton.

Optionen

Tabelle 65: „AutoCompleteLoqate“-Optionen

optionName	Beschreibung
Database.Loqate	Gibt die für die Verarbeitung von Adressen zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich Datenbankressourcen definiert wurden.

optionName	Beschreibung
OutputCasing	<p>Gibt die Groß-/Kleinschreibung der Ausgabedaten an. Zur Auswahl stehen:</p> <p>M Gibt die Ausgabe in Groß- und Kleinbuchstaben zurück (Standard). Beispiel:</p> <p style="padding-left: 40px;">123 Main St Mytown FL 12345</p> <p>U Gibt die Ausgabe in Großbuchstaben zurück. Beispiel:</p> <p style="padding-left: 40px;">123 MAIN ST MYTOWN FL 12345</p>
HomeCountry	<p>Gibt das Standardland an. Sie sollten das Land angeben, in dem sich die meisten Adressen befinden. Wenn Sie beispielsweise vor allem deutsche Adressen verarbeiten, geben Sie „Germany“ an. Gültige Ländernamen sind:</p> <p>Afghanistan, Albania, Algeria, American Samoa, Andorra, Angola, Anguilla, Antigua And Barbuda, Argentina, Armenia, Aruba, Australia, Austria, Azerbaijan, Bahamas, Bahrain, Bangladesh, Barbados, Belarus, Belgium, Belize, Benin, Bermuda, Bhutan, Bolivia, Bosnia And Herzegovina, Botswana, Brazil, British Virgin Islands, Brunei Darussalam, Bulgaria, Burkina Faso, Burundi, Cambodia, Cameroon, Canada, Cape Verde, Cayman Islands, Central African Republic, Chad, Chile, China, Colombia, Comoros Islands, Congo, Cook Islands, Costa Rica, Cote D'Ivoire, Croatia, Cuba, Cyprus, Czech Republic, Democratic Republic Of Congo, Denmark, Djibouti, Dominica, Dominican Republic, East Timor, Ecuador, Egypt, El Salvador, Equatorial Guinea, Eritrea, Estonia, Ethiopia, Falkland Islands, Faroe Islands, Federated States Of Micronesia, Fiji, Finland, France, French Guiana, Gabon, Gambia, Germany, Ghana, Gibraltar, Greece, Greenland, Grenada, Guadeloupe, Guam, Guatemala, Guinea, Guinea Bissau, Guyana, Haiti, Holy See, Honduras, Hong Kong, Hungary, Iceland, India, Indonesia, Iran, Iraq, Ireland, Israel, Italy, Jamaica, Japan, Jordan, Kazakhstan, Kenya, Kiribati, Korea, Kuwait, Kyrgyzstan, Laos, Latvia, Lebanon, Lesotho, Liberia, Libya, Liechtenstein, Lithuania, Luxembourg, Macau, Macedonia, Madagascar, Malawi, Malaysia, Maldives, Mali, Malta, Marshall Islands, Martinique, Mauritania, Mauritius, Mayotte, Mexico, Moldova, Monaco, Mongolia, Monserrat, Morocco, Mozambique, Myanmar, Namibia, Nauru, Nepal, Netherlands Antilles, New Caledonia, New Zealand, Nicaragua, Niger, Nigeria, Niue, Norway, Oman, Pakistan, Palau, Panama, Papua New Guinea, Paraguay, Peru, Philippines, Pitcairn Islands, Poland, Portugal, Puerto Rico, Qatar, Republic Of Georgia, Republic Of Korea, Republic Of Singapore, Reunion, Romania, Russia, Rwanda, Saint Helena, Saint Kitts And Nevis, Saint Lucia, Saint Pierre And Miquelon, Saint Vincent and the Grenadines, Samoa, San Marino, Sao Tome and Principe, Saudi Arabia, Senegal, Seychelles, Sierra Leone, Slovakia, Slovenia, Solomon Islands, Somalia, South Africa, Spain, Sri Lanka, Sudan, Suriname, Swaziland, Sweden, Switzerland, Syria, Tahiti, Taiwan, Tajikistan, Tanzania, Thailand, The Netherlands, Togo, Tonga, Trinidad And Tobago, Tristan Da Cunha, Tunisia, Turkey, Turkmenistan, Turks And Caicos Islands, Tuvalu, Uganda, Ukraine, United Arab Emirates, United Kingdom, United States, Uruguay, Uzbekistan, Vanuatu, Venezuela, Vietnam, Virgin Islands (US), Wallis And Futuna, Yemen, Yugoslavia, Zambia, Zimbabwe</p>

optionName	Beschreibung
OutputCountryFormat	<p>Gibt das Format an, das für den im Ausgabefeld Country zurückgegebenen Ländernamen verwendet werden soll. Wenn Sie beispielsweise „Englisch“ auswählen, wird der Ländername „Deutschland“ als „Germany“ zurückgegeben.</p> <p>E Englische Ländernamen verwenden (Standard).</p> <p>I ISO-Abkürzung mit zwei Buchstaben anstelle von Ländernamen für die Länder verwenden.</p> <p>U Abkürzung der Universal Postal Union anstelle von Ländernamen für die Länder verwenden.</p>
OutputScript	<p>Gibt das Alphabet oder die Schrift an, in dem oder in der die Ausgabe zurückgegeben werden soll. Diese Option ist bidirektional und ist im Allgemeinen als von Muttersprache zu Latein oder als von Latein zu Muttersprache festgelegt.</p> <p>Input Es wird keine Transkription durchgeführt. Die Ausgabe erfolgt in derselben Schrift wie die Eingabe (Standard).</p> <p>Native Die Ausgabe erfolgt in der muttersprachlichen Schrift des ausgewählten Landes, falls möglich.</p> <p>Latn Englische Werte werden verwendet.</p>
MaximumResults	<p>Die maximale Anzahl der Adresse, die von AutoCompleteLoqate zurückgegeben werden sollten. Der Standardwert ist 10.</p>

optionName	Beschreibung
isPowersearchEnable	<p>Reduziert die Eingabezeit bei 240 Ländern um bis zu 80 %, indem Daten in Form einer Indexdatei verwendet werden. Beim Durchführen einer Suche sucht die Loqate-Engine zunächst nach dem entsprechenden Index. Falls vorhanden, versucht die Methode, direkt eine Liste der Kandidatenadressen zurückzugeben. Wenn der Index nicht vorhanden ist oder keine Ergebnisse zurückgibt, wird der ursprüngliche Suchprozess ausgelöst.</p> <p>Anmerkung: Die Powersuche kann durchgeführt werden, wenn in der Eingabedatei nur zwei Felder enthalten sind: das Feld „Country“ und eines der „AddressLine“-Felder. Wenn Sie diese Option auswählen und Ihre Eingabedatei zusätzliche Felder enthält, wird der ursprüngliche Suchprozess automatisch ausgelöst.</p> <p>Zum Durchführen der Suche werden in Auto Complete-Indizes für Suchvorgänge innerhalb der Vereinigten Staaten bis zu 10 Zeichen verwendet und für Suchvorgänge in allen anderen auswählbaren Ländern bis zu 15 Zeichen. Leerzeichen und Interpunktion werden bei dieser Zahl nicht berücksichtigt.</p> <p>Für folgende Länder kann die Powersuche nicht verwendet werden: Botswana, Äthiopien, Indien, Kasachstan, Malaysia, Mongolei, Saint Kitts und Nevis und San Marino.</p> <p>Anmerkung: Sie müssen über eine gültige Lizenz für die Verarbeitung der Powersuche verfügen. Wenn Sie diese Option auswählen, die Powersuche jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben.</p>
IsDuplicateHandlingMaskEnable	<p>Aktiviert die Dublettenbehandlungsmaske und gibt an, wie Datensatzdubletten verarbeitet und entfernt werden. Wählen Sie mindestens eine der folgenden Optionen aus:</p> <ul style="list-style-type: none">S Ist standardmäßig ausgewählt. Eingabe vorab bearbeiten und Dubletten entfernen, die in einem Feld auftreten.C Ist standardmäßig ausgewählt. Eingabe vorab bearbeiten und Dubletten über alle Felder entfernen.T Eingabe vorab bearbeiten und Dubletten in Feldern entfernen, bei denen es sich nicht um Standardadressfelder handelt.F Ist standardmäßig ausgewählt. Ausgabe der Überprüfung nachbearbeiten und Dubletten aus nicht überprüften Feldern entfernen.

optionName	Beschreibung
FailJobOnDataLicenseError	Gibt an, wie Spectrum Technology Platform reagieren soll, wenn ein Datenlizenzfehler auftritt.
	<p>Auftrag als ungültig erklären Der gesamte Auftrag wird als ungültig erklärt, wenn ein Datenlizenzfehler auftritt.</p> <p>Datensatz als ungültig erklären Ein oder mehrere Datensätze werden als ungültig erklärt, wenn ein Datenlizenzfehler auftritt. Die Verarbeitung wird fortgesetzt.</p>

Ausgabe

Die Ausgabe von „AutoCompleteLoqate“ ist optional und entspricht direkt den von Ihnen im Abschnitt „Ausgabefelder“ im Dialogfeld „Optionen für ‘AutoCompleteLoqate‘“ ausgewählten Feldern.

Tabelle 66: Ausgabe für „AutoCompleteLoqate“

columnName	Beschreibung
AddressLine1	Die erste Adresszeile.
AddressLine2	Die zweite Adresszeile.
AddressLine3	Die dritte Adresszeile.
AddressLine4	Die vierte Adresszeile.
City	Der Ortsname.
Country	Der dreistellige ISO 3116-1 Alpha-3-Code für das Land. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
FirmName	Der Firmenname.

columnName	Beschreibung
HouseNumber	Die letzte Hausnummer des Bereichs, in den die Hausnummer der Kandidatenadresse fällt.
PostalCode	Die Postleitzahl.
PostalCode.AddOn	Die letzten vier Zeichen des ZIP + 4 [®] -Codes.
ProcessedBy	Zeigt an, welcher Adressen-Codierer die Adresse verarbeitet. LOQATE Der Loqate-Codierer verarbeitet die Adressen.
StateProvince	Die Abkürzung für das Bundesland oder den Kanton.
Status	Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs. null Erfolg F Fehler
Status.Code	Die Ursache für den Fehler, falls vorhanden. • DisabledCoder • RequestFailed • NoLookupAddressFound
Status.Description	Eine Beschreibung des Problems, wenn eines vorliegt. Did not return multiples Die Eingabeadresse stimmte nur mit einer Adresse in der Datenbank überein. „AutoCompleteLoqate“ gibt nur dann Daten zurück, wenn mehrere mögliche Übereinstimmungen gefunden wurden. Not able to look up the address pattern „AutoCompleteLoqate“ kann die unvollständige Adresse nicht verarbeiten.

„GetCandidateAddresses“

„GetCandidateAddresses“ gibt eine Liste der Adressen zurück, die bei einer angegebenen Eingabeadresse als Übereinstimmungen gelten. „GetCandidateAddresses“ gibt Kandidatenadressen nur dann zurück, wenn die Eingabeadresse mit mehreren Adressen in der Postdatenbank übereinstimmt. Wenn die Eingabeadresse nur mit einer Adresse in der Postdatenbank übereinstimmt, werden keine Adressdaten zurückgegeben.

Bei Adressen außerhalb der USA und Kanada treten bei den von „ValidateAddress“ zurückgegebenen Mehrfachübereinstimmungen und den für dieselbe Adresse von „GetCandidateAddresses“ zurückgegebenen Ergebnissen möglicherweise inkonsistente Ergebnisse auf. Die Ursache für die inkonsistenten Ergebnisse ist wahrscheinlich, dass Sie die Einstellung für die Leistungsoptimierung in „ValidateAddress“ auf einen anderen Wert als 100 festgelegt haben. Legen Sie die Option für die Leistungsoptimierung auf 100 fest, um konsistente Ergebnisse bei „GetCandidateAddresses“ und „ValidateAddress“ zu erzielen.

Anmerkung: Standardmäßig gleicht „GetCandidateAddresses“ keine einzelnen Hausnummern ab. Stattdessen werden für jede Straße Hausnummernbereiche verwendet. Nachdem „GetCandidateAddresses“ den Straßennamen, den Ortsnamen, Bundesland/Kanton und Postleitzahl bestimmt hat, überprüft es, ob die eingegebene Hausnummer in einen der Hausnummernbereiche fällt, die für die übereinstimmende Straße angegeben sind. Dieselbe Logik wird auf Einheitsnummern angewendet. Wenn Sie bestimmen möchten, ob eine einzelne Hausnummer gültig ist, müssen Sie die Delivery Point Validation(DPV)-Verarbeitungsoption von „ValidateAddress“ verwenden. Die Verfügbarkeit der DPV-Verarbeitung gilt nur für US-amerikanische Adressen.

Der kanadische Coder umfasst eine „Reverse Lookup“-Routine, die eine bestimmte Postleitzahl als Eingabe verwendet und die für diese Postleitzahl in der Datenbank gespeicherten Straßeninformationen zurückgibt. Geben Sie für die Verwendung dieser Funktion lediglich eine kanadische Postleitzahl in das Feld „PostalCode“ ein. Im zweiten Beispiel wird dargestellt, wie die Rückgabe über eine beispielhafte Postleitzahl erfolgt.

„GetCandidateAddresses“ ist Teil des Universal Addressing-Moduls.

Eingabe

Die folgende Tabelle enthält die Eingabe für „GetCandidateAddresses“.

Tabelle 67: Eingabeformat

columnName	Beschreibung
AddressLine1	Die erste Adresszeile.

columnName	Beschreibung
AddressLine2	Die zweite Adresszeile.
AddressLine3	Die dritte Adresszeile. Gilt nicht für US-amerikanische und kanadische Adressen standardisiert.
AddressLine4	Die vierte Adresszeile. Gilt nicht für US-amerikanische und kanadische Adressen standardisiert.
AddressLine5	Die fünfte Adresszeile. Gilt nur für britische Adressen. Kann Straßenname, Einheitsnummer, Gebäudenummer usw. enthalten.
City	Der Ortsname.
StateProvince	Das Bundesland oder der Kanton. Bei US-amerikanischen Adressen können Sie den Bundesstaat in das Feld „City“ anstelle des Feldes „StateProvince“ eintragen.
PostalCode	Die Postleitzahl zu der Adresse. Bei US-amerikanischen Adressen ist dies der ZIP Code™ in einem der folgenden Formate: 99999 99999-9999 A9A9A9 A9A 9A9 9999 999 Anmerkung: Bei kanadischen Adressen können Sie einfach dieses Feld ausfüllen, und anschließend werden Kandidatenadressdaten zurückgegeben. Bei anderen Ländern müssen auch die Felder „AddressLine1“ und „AddressLine2“ ausgefüllt werden.

columnName	Beschreibung
Country	<p>Der Ländercode oder Ländername, in einem der folgenden Formate:</p> <ul style="list-style-type: none">• zweistelliger ISO-Ländercode• dreistelliger UPU-Ländercode• Englischer Ländername• Französischer Ländername• Deutscher Ländername• Spanischer Ländername <p>Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.</p>
FirmName	Der Unternehmens- bzw. Firmenname.
USUrbanName	US-amerikanische Name der Wohnanlage. Wird vor allem für puerto-ricanische Adressen verwendet.

Optionen

Tabelle 68: Optionen für „GetCandidateAddresses“

optionName	Beschreibung
PerformUSProcessing	<p>Gibt an, ob folgende Adressen verarbeitet werden sollen: US-amerikanische Adressen. Wenn Sie die Verarbeitung US-amerikanischer Adressen aktivieren, versucht „GetCandidateAddresses“, Kandidaten für folgende Adressen abzurufen: US-amerikanische Adressen. Wenn Sie die Verarbeitung US-amerikanischer Adressen deaktivieren, schlagen US-amerikanische Adressen fehl. Das bedeutet, dass diese mit einem „F“ im Ausgabefeld „Status“ zurückgegeben werden. Das Ausgabefeld „Status.Code“ enthält dann „DisabledCoder“. Wenn Sie die Verarbeitung US-amerikanischer Adressen nicht lizenziert haben, müssen Sie die Verarbeitung US-amerikanischer Adressen deaktivieren, damit Ihre Aufträge erfolgreich abgeschlossen werden, auch mit US-amerikanischen Adressen.</p> <p>Anmerkung: Sie müssen über eine gültige Lizenz für die Verarbeitung US-amerikanischer Adressen verfügen für die erfolgreiche Verarbeitung US-amerikanischer Adressen. Wenn Sie die Verarbeitung US-amerikanischer Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben.</p> <p>Y Ja, US-amerikanische Adressen verarbeiten (Standard).</p> <p>N Nein, keine Verarbeitung von US-amerikanischen Adressen.</p>
Database.US	<p>Gibt die für die Verarbeitung US-amerikanischer Adressen zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich US-Datenbankressourcen definiert wurden.</p>

optionName	Beschreibung
PerformCanadianProcessing	<p>Gibt an, ob kanadische Adressen verarbeitet werden sollen. Wenn Sie die Verarbeitung kanadischer Adressen aktivieren, versucht GetCandidateAddresses, die Kandidatenadressen für kanadische Adressen abzurufen. Wenn Sie die Verarbeitung kanadischer Adressen deaktivieren, schlagen kanadische Adressen fehl. Das bedeutet, dass diese mit einem „F“ im Ausgabefeld „Status“ zurückgegeben werden. Das Ausgabefeld „Status.Code“ enthält dann „DisabledCoder“. Wenn Sie die Verarbeitung kanadischer Adressen nicht lizenziert haben, müssen Sie die Verarbeitung kanadischer Adressen deaktivieren, damit Ihre Aufträge erfolgreich abgeschlossen werden, auch wenn sie kanadische Adressen enthalten.</p> <p>Anmerkung: Für eine erfolgreiche Verarbeitung kanadischer Adressen müssen Sie über eine gültige Lizenz für die Verarbeitung kanadischer Adressen verfügen. Wenn Sie die Verarbeitung kanadischer Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben.</p> <p>Y Ja, kanadische Adressen verarbeiten (Standard).</p> <p>N Nein, kanadische Adressen nicht verarbeiten.</p>
Database.Canada	<p>Gibt die für die Verarbeitung von kanadischen Adressen zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich Kanadische Datenbankressourcen definiert wurden.</p>

optionName	Beschreibung
PerformInternationalProcessing	<p>Gibt an, ob internationale Adressen (Adressen außerhalb der USA und Kanada) verarbeitet werden sollen. Wenn Sie die Verarbeitung internationaler Adressen aktivieren, versucht GetCandidateAddresses, die Kandidatenadressen für internationale Adressen abzurufen. Wenn Sie die Verarbeitung internationaler Adressen deaktivieren, schlagen internationale Adressen fehl. Das bedeutet, dass diese mit einem „F“ im Ausgabefeld „Status“ zurückgegeben werden. Das Ausgabefeld „Status.Code“ enthält dann „DisabledCoder“. Wenn Sie die Verarbeitung internationaler Adressen nicht lizenziert haben, müssen Sie die Verarbeitung internationaler Adressen deaktivieren, damit Ihre Aufträge erfolgreich abgeschlossen werden, auch wenn sie internationale Adressen enthalten.</p> <p>Anmerkung: Sie müssen eine gültige Lizenz für die Verarbeitung internationaler Adressen haben, um internationale Adressen erfolgreich verarbeiten zu können. Wenn Sie die Verarbeitung internationaler Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben.</p> <p>Y Ja, internationale Adressen verarbeiten (Standard).</p> <p>N Nein, internationale Adressen nicht verarbeiten.</p>
Database.International	<p>Gibt die für die Verarbeitung von internationalen Adressen zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich Internationale Datenbankressourcen definiert wurden.</p>
OutputCasing	<p>Gibt die Groß-/Kleinschreibung der Ausgabedaten an. Zur Auswahl stehen:</p> <p>M Gibt die Ausgabe in Groß- und Kleinbuchstaben zurück (Standard). Beispiel:</p> <p>123 Main St Mytown FL 12345</p> <p>U Gibt die Ausgabe in Großbuchstaben zurück. Beispiel:</p> <p>123 MAIN ST MYTOWN FL 12345</p>
MaximumResults	<p>Die maximale Anzahl der Kandidatenadressen, die „GetCandidateAddresses“ zurückgeben soll. Der Standardwert ist 10. Der Maximalwert ist 10.</p>

optionName	Beschreibung
OutputShortCityName	<p>Bei US-amerikanischen wird angegeben, ob die vom USPS® genehmigte Abkürzung für den Ort zurückgegeben werden soll, sofern vorhanden. Der USPS® bietet Abkürzungen für Ortsnamen, die mindestens 14 Zeichen umfassen. Abkürzungen für Orte sind 13 Zeichen lang oder kürzer und können bei begrenztem Platz auf dem Adressetikett verwendet werden. Wenn es keinen kurzen Ortsnamen für den Ort gibt, wird der komplette Ortsname zurückgegeben.</p> <p>Y Ja, den kurzen Ortsnamen zurückgeben.</p> <p>N Nein, den kurzen Ortsnamen nicht zurückgeben.</p>
DualAddressLogic	<p>(Nur US-amerikanische Adressen.) Steuert, ob „GetCandidateAddresses“ eine Straßenübereinstimmung oder eine Übereinstimmung mit Postfach/Landstraße/Autobahn zurückgeben soll, wenn die Adresse Straßeninformationen und Informationen zu Postfach/Landstraße/Autobahn enthält. Weitere Informationen finden Sie unter DualAddressLogic auf Seite 419.</p> <p>N (Standard) USPS®CASS™-Bestimmungen geben anhand der folgenden Rangfolge die Formatierung der zurückgegebenen Adresse vor:</p> <ol style="list-style-type: none"> 1. Postfach 2. Firma 3. Hochhaus 4. Straße 5. Landstraße 6. Allgemeine Zustellung <p>S Gibt eine Straßenübereinstimmung unabhängig von der Adresszeile zurück.</p> <p>P Gibt eine Postfachübereinstimmung unabhängig von der Adresszeile zurück.</p>
StreetMatchingStrictness	<p>Die Genauigkeit für den Vergleich von Straßennamen (nur bei US-amerikanischen Adressen.)</p> <p>E Der Straßename in der Eingabe muss exakt mit der Datenbank übereinstimmen.</p> <p>T Der Vergleichsalgorithmus ist „eng“.</p> <p>M Der Vergleichsalgorithmus ist „mittel“ (Standard).</p> <p>L Der Vergleichsalgorithmus ist „lose“.</p>

optionName	Beschreibung
FirmMatchingStrictness	<p>Die Genauigkeit für den Vergleich von Firmennamen (nur bei US-amerikanischen Adressen.)</p> <p>E Der Firmenname in der Eingabe muss exakt mit der Datenbank übereinstimmen.</p> <p>T Der Vergleichsalgorithmus ist „eng“.</p> <p>M Der Vergleichsalgorithmus ist „mittel“ (Standard).</p> <p>L Der Vergleichsalgorithmus ist „lose“.</p>
DirectionalMatchingStrictness	<p>Die Genauigkeit des Richtungsvergleichs.</p> <p>E Die Richtungsangabe in der Eingabe muss exakt mit der Datenbank übereinstimmen.</p> <p>T Der Vergleichsalgorithmus ist „eng“.</p> <p>M Der Vergleichsalgorithmus ist „mittel“ (Standard).</p> <p>L Der Vergleichsalgorithmus ist „lose“.</p>
PerformESM	<p>Gibt an, ob ein erweiterter Straßenvergleich (Enhanced Street Matching, ESM) ausgeführt werden soll. Der EMS wendet eine zusätzliche Vergleichslogik mit zusätzlichen Daten auf eine Eingabeadresse an, die nicht durch den regulären Adressenüberprüfungsprozess verglichen wird. Der ESM gilt für US-amerikanische Adressen.</p> <p>Y Ja, EMS-Verarbeitung durchführen.</p> <p>N Nein, ESM-Verarbeitung nicht durchführen (Standard).</p>

optionName	Beschreibung
AddressLineSearchOnFail	<p>Gibt an, ob ValidateAdress für Ort, Bundesland/Kanton und Postleitzahl nach Adresszeilen sucht.</p> <p>Bei Aktivierung dieser Option kann „ValidateAddress“ die „AddressLine“-Eingabefelder nach dem Ort, dem Bundesland/Kanton, der Postleitzahl und dem Land durchsuchen, wenn die Adresse mithilfe der Werte in den Eingabefeldern „City“, „StateProvince“ und „PostalCode“ nicht verglichen werden kann.</p> <p>Überlegen Sie, diese Option zu aktivieren, wenn die „AddressLine“-Felder Ihrer Eingabeadressen Angaben zu Ort, Bundesland/-staat/Kanton und Postleitzahl enthalten.</p> <p>Ziehen Sie die Deaktivierung dieser Option in Betracht, wenn bei Ihren Eingabeadressen die Felder „City“, „State/Province“ und „PostalCode“ verwendet werden. Wenn Sie diese Option aktivieren und diese Felder verwendet werden, besteht eine größere Möglichkeit, dass „ValidateAddress“ die Werte in diesen Feldern (z. B. einen falsch geschriebenen Ortsnamen) nicht korrigieren kann.</p> <p>Y Ja, Adresszeilenfelder durchsuchen (Standard).</p> <p>N Nein, „AddressLine“-Felder nicht durchsuchen.</p>

Ausgabe

GetCandidateAddresses gibt die folgende Ausgabe zurück.

Tabelle 69: Ausgabe von GetCandidateAddresses

columnName	Beschreibung
AddressLine1	Die erste Adresszeile.
AddressLine2	Die zweite Adresszeile.
AddressLine3	Die dritte Adresszeile.
AddressLine4	Die vierte Adresszeile.

columnName	Beschreibung
AddressLine5	Nur für britische Adressen. Wenn die Adresse überprüft wurde, die fünfte Zeile der überprüften und standardisierten Adresse. Wenn die Adresse nicht überprüft werden konnte, die fünfte Zeile der Eingabeadresse ohne Änderungen.
City	Der Ortsname.
Country	Der dreistellige ISO 3116-1 Alpha-3-Code für das Land. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
FirmName	Der Firmenname.
HouseNumberHigh	Die letzte Hausnummer des Bereichs, in den die Hausnummer der Kandidatenadresse fällt.
HouseNumberLow	Die erste Hausnummer des Bereichs, in den die Hausnummer der Kandidatenadresse fällt.
HouseNumberParity	<p>Zeigt das Nummerierungsschema für die Hausnummern zwischen „HouseNumberLow“ und „HouseNumberHigh“ an, wie folgt:</p> <p>E Nur gerade Werte.</p> <p>O Nur ungerade Werte.</p> <p>B Beide</p>
MatchLevel	<p>Identifiziert bei Adressen außerhalb der USA und Kanada die Vergleichsebene für die Kandidatenadresse. US-amerikanische und kanadische Adressen sind immer „A“. Folgende Werte sind möglich:</p> <p>A Der Kandidat stimmt auf Straßenebene mit der Eingabeadresse überein.</p> <p>B Der Kandidat stimmt auf Bundesland/Kanton-Ebene mit der Eingabeadresse überein.</p>
PostalCode	Die Postleitzahl. In den USA ist dies der ZIP-Code™.
PostalCode.AddOn	Die letzten vier Zeichen des ZIP + 4®-Codes. US-amerikanische Adressen.

columnName	Beschreibung
RecordType	<p>Der Typ des Adressdatensatzes, wie er von den Postbehörden der USA und Kanada definiert ist (nur für Adressen in den USA und Kanada):</p> <ul style="list-style-type: none"> • FirmRecord • GeneralDelivery • HighRise • PostOfficeBox • RRHighwayContract • Normal
RecordType.Default	<p>Code, der die Übereinstimmung mit dem „Standard“ anzeigt:</p> <p>Y Die Adresse stimmt mit einem Standarddatensatz überein.</p> <p>null Die Adresse stimmt nicht mit einem Standarddatensatz überein.</p>
StateProvince	Die Abkürzung für das Bundesland oder den Kanton.
Status	<p>Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs.</p> <p>null Erfolg</p> <p>F Fehler</p>
Status.Code	<p>Die Ursache für den Fehler, falls vorhanden. Es ist nur ein Wert möglich:</p> <ul style="list-style-type: none"> • DisabledCoder • RequestFailed

columnName	Beschreibung
Status.Description	<p>Eine Beschreibung des Problems, wenn eines vorliegt.</p> <p>Did not return multiples Die Eingabeadresse stimmte nur mit einer Adresse in der Datenbank überein. GetCandidateAddresses gibt nur Daten zurück, wenn mehrere mögliche Übereinstimmungen gefunden wurden.</p> <p>Number of candidates is not greater than 1 Die Eingabeadresse stimmte mit mehr als einer Adresse in der Datenbank überein, es wurden jedoch keine Adressen zurückgegeben.</p> <p>PerformUSProcessing disabled Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=DisabledCoder.</p> <p>PerformCanadianProcessing disabled Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=DisabledCoder.</p> <p>PerformInternationalProcessing disabled Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=DisabledCoder.</p>
UnitNumberHigh	Die letzte Einheitsnummer des Bereichs, in den die Einheitsnummer der Kandidatenadresse fällt.
UnitNumberLow	Die erste Einheitsnummer des Bereichs, in den die Einheitsnummer der Kandidatenadresse fällt.
UnitNumberParity	<p>Zeigt das Nummerierungsschema für die Einheitsnummern zwischen „UnitNumberLow“ und „UnitNumberHigh“ an, wie folgt:</p> <p>E Nur gerade Werte.</p> <p>O Nur ungerade Werte.</p> <p>B Beide</p>
USUrbanName	Der überprüfte Name der Ortswohnanlage. Wohnanlagennamen werden hauptsächlich bei Adressen in Puerto Rico verwendet.

GetCandidateAddressesLoqate

GetCandidateAddressesLoqate gibt eine Liste der Adressen zurück, die bei einer angegebenen Eingabeadresse als Übereinstimmungen gelten. GetCandidateAddressesLoqate gibt nur Kandidatenadressen zurück, wenn die Eingabeadresse mit mehreren Adressen in der Postdatenbank übereinstimmt. Wenn die Eingabeadresse nur mit einer Adresse in der Postdatenbank übereinstimmt, werden keine Adressdaten zurückgegeben. Das Eingabefeld „Land“ ist erforderlich. Wenn dieses Feld leer ist, wird keine Ausgabe zurückgegeben.

Anmerkung: Standardmäßig gleicht GetCandidateAddressesLoqate keine einzelnen Hausnummern ab. Stattdessen werden für jede Straße Hausnummernbereiche verwendet. Nachdem GetCandidateAddressesLoqate den Straßennamen, den Ortsnamen, Bundesland/Kanton und Postleitzahl bestimmt hat, überprüft es, ob die eingegebene Hausnummer in einen der Hausnummernbereiche fällt, die für die übereinstimmende Straße angegeben sind. Dieselbe Logik wird auf Einheitsnummern angewendet.

GetCandidateAddressesLoqate ist Teil des Universal Addressing-Moduls.

Eingabe

In der folgenden Tabelle ist die Eingabe für GetCandidateAddressesLoqate aufgeschlüsselt.

Tabelle 70: Eingabeformat

columnName	Beschreibung
AddressLine1	Die erste Adresszeile.
AddressLine2	Die zweite Adresszeile.
AddressLine3	Die dritte Adresszeile.
AddressLine4	Die vierte Adresszeile.
City	Der Ortsname.

columnName	Beschreibung
Country	<p>Der Ländercode oder Ländername, in einem der folgenden Formate:</p> <ul style="list-style-type: none"> • zweistelliger ISO-Ländercode • dreistelliger UPU-Ländercode • Englischer Ländername <p>Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.</p> <p>Anmerkung: Dieses Feld ist ein Pflichtfeld. Wenn dieses Feld leer ist, wird keine Ausgabe zurückgegeben.</p>
FirmName	Der Unternehmens- bzw. Firmenname.
PostalCode	Die Postleitzahl zu der Adresse. Bei US-amerikanischen Adressen ist dies der ZIP Code™ in einem der folgenden Formate:
StateProvince	<p>Das Bundesland oder der Kanton.</p> <p>Bei US-amerikanischen Adressen können Sie den Bundesstaat in das Feld „City“ anstelle des Feldes „StateProvince“ eintragen.</p>

Optionen

Tabelle 71: Optionen für GetCandidateAddressesLoqate

optionName	Beschreibung
Database.Loqate	Gibt die für die Verarbeitung von Adressen zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console definiert wurden.

optionName	Beschreibung
OutputCasing	<p>Gibt die Groß-/Kleinschreibung der Ausgabedaten an. Zur Auswahl stehen:</p> <p>M Gibt die Ausgabe in Groß- und Kleinbuchstaben zurück (Standard). Beispiel: 123 Main St Mytown FL 12345</p> <p>U Gibt die Ausgabe in Großbuchstaben zurück. Beispiel: 123 MAIN ST MYTOWN FL 12345</p>
CandidateProcessOption	<p>Gibt die Methode zur Kandidatensuche an. Zur Auswahl stehen:</p> <p>S Geben Sie eine vollständige oder unvollständige Adresse als Eingabe ein. Sie erhalten dann als Ausgabe eine Liste von Ergebnissen mit hoher Übereinstimmung (Standard).</p> <p>V Geben Sie Adressinformationen in Adresszeilen, Adresskomponenten oder eine Kombination aus beiden als Eingabe ein. Sie erhalten dann als Ausgabe Ergebnisse, die besser mit der Eingabe übereinstimmen.</p>

optionName	Beschreibung
HomeCountry	<p>Gibt das Standardland an. Sie sollten das Land angeben, in dem sich die meisten Adressen befinden. Wenn Sie beispielsweise vor allem deutsche Adressen verarbeiten, geben Sie „Germany“ an. GetCandidateAddressesLoqate verwendet das von Ihnen angegebene Land, um eine Überprüfung einzuleiten, wenn es das Land nicht über die Adressfelder „StateProvince“, „PostalCode“ und „Country“ bestimmen kann. Gültige Ländernamen sind:</p> <p>Afghanistan, Albania, Algeria, American Samoa, Andorra, Angola, Anguilla, Antigua And Barbuda, Argentina, Armenia, Aruba, Australia, Austria, Azerbaijan, Bahamas, Bahrain, Bangladesh, Barbados, Belarus, Belgium, Belize, Benin, Bermuda, Bhutan, Bolivia, Bosnia And Herzegovina, Botswana, Brazil, British Virgin Islands, Brunei Darussalam, Bulgaria, Burkina Faso, Burundi, Cambodia, Cameroon, Canada, Cape Verde, Cayman Islands, Central African Republic, Chad, Chile, China, Colombia, Comoros Islands, Congo, Cook Islands, Costa Rica, Cote D'Ivoire, Croatia, Cuba, Cyprus, Czech Republic, Democratic Republic Of Congo, Denmark, Djibouti, Dominica, Dominican Republic, East Timor, Ecuador, Egypt, El Salvador, Equatorial Guinea, Eritrea, Estonia, Ethiopia, Falkland Islands, Faroe Islands, Federated States Of Micronesia, Fiji, Finland, France, French Guiana, Gabon, Gambia, Germany, Ghana, Gibraltar, Greece, Greenland, Grenada, Guadeloupe, Guam, Guatemala, Guinea, Guinea Bissau, Guyana, Haiti, Holy See, Honduras, Hong Kong, Hungary, Iceland, India, Indonesia, Iran, Iraq, Ireland, Israel, Italy, Jamaica, Japan, Jordan, Kazakhstan, Kenya, Kiribati, Korea, Kuwait, Kyrgyzstan, Laos, Latvia, Lebanon, Lesotho, Liberia, Libya, Liechtenstein, Lithuania, Luxembourg, Macau, Macedonia, Madagascar, Malawi, Malaysia, Maldives, Mali, Malta, Marshall Islands, Martinique, Mauritania, Mauritius, Mayotte, Mexico, Moldova, Monaco, Mongolia, Monserrat, Morocco, Mozambique, Myanmar, Namibia, Nauru, Nepal, Netherlands Antilles, New Caledonia, New Zealand, Nicaragua, Niger, Nigeria, Niue, Norway, Oman, Pakistan, Palau, Panama, Papua New Guinea, Paraguay, Peru, Philippines, Pitcairn Islands, Poland, Portugal, Puerto Rico, Qatar, Republic Of Georgia, Republic Of Korea, Republic Of Singapore, Reunion, Romania, Russia, Rwanda, Saint Helena, Saint Kitts And Nevis, Saint Lucia, Saint Pierre And Miquelon, Saint Vincent And The Grenadines, Samoa, San Marino, Sao Tome and Principe, Saudi Arabia, Senegal, Seychelles, Sierra Leone, Slovakia, Slovenia, Solomon Islands, Somalia, South Africa, Spain, Sri Lanka, Sudan, Suriname, Swaziland, Sweden, Switzerland, Syria, Tahiti, Taiwan, Tajikistan, Tanzania, Thailand, The Netherlands, Togo, Tonga, Trinidad And Tobago, Tristan Da Cunha, Tunisia, Turkey, Turkmenistan, Turks And Caicos Islands, Tuvalu, Uganda, Ukraine, United Arab Emirates, United Kingdom, United States, Uruguay, Uzbekistan, Vanuatu, Venezuela, Vietnam, Virgin Islands (US), Wallis And Futuna, Yemen, Yugoslavia, Zambia, Zimbabwe</p>

optionName	Beschreibung
OutputCountryFormat	<p>Gibt das Format an, das für den im Ausgabefeld Country zurückgegebenen Ländernamen verwendet werden soll. Wenn Sie beispielsweise „Englisch“ auswählen, wird der Ländername „Deutschland“ als „Germany“ zurückgegeben.</p> <p>E Englische Ländernamen verwenden (Standard).</p> <p>I ISO-Abkürzung mit zwei Buchstaben anstelle von Ländernamen für die Länder verwenden.</p> <p>U Abkürzung der Universal Postal Union anstelle von Ländernamen für die Länder verwenden.</p>
OutputScript	<p>Gibt das Alphabet oder die Schrift an, in dem oder in der die Ausgabe zurückgegeben werden soll. Diese Option ist bidirektional und ist im Allgemeinen als von Muttersprache zu Latein oder als von Latein zu Muttersprache festgelegt.</p> <p>Input Es wird keine Transkription durchgeführt. Die Ausgabe erfolgt in derselben Schrift wie die Eingabe (Standard).</p> <p>Native Die Ausgabe erfolgt in der muttersprachlichen Schrift des ausgewählten Landes, falls möglich.</p> <p>Latn Englische Werte werden verwendet.</p>
MaximumResults	Die maximale Anzahl an Kandidatenadressen, die GetCandidateAddressesLoqate zurückgeben soll. Der Standardwert ist 10. Der Maximalwert ist 99.

Ausgabe

GetCandidateAddressesLoqate gibt die folgende Ausgabe zurück.

Tabelle 72: Ausgabe von GetCandidateAddressesLoqate

columnName	Beschreibung
AddressLine1	Die erste Adresszeile.
AddressLine2	Die zweite Adresszeile.
AddressLine3	Die dritte Adresszeile.

columnName	Beschreibung
AddressLine4	Die vierte Adresszeile.
City	Der Ortsname.
Country	Der dreistellige ISO 3116-1 Alpha-3-Code für das Land. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
FirmName	Der Firmenname.
PostalCode	Die Postleitzahl. In den USA ist dies der ZIP-Code™.
PostalCode.AddOn	Die letzten vier Zeichen des ZIP + 4®-Codes. US-amerikanische Adressen.
ProcessedBy	Zeigt an, welcher Adressen-Codierer die Adresse verarbeitet. LOQATE Der Loqate-Codierer verarbeitet die Adressen.
StateProvince	Die Abkürzung für das Bundesland oder den Kanton.
Status	Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs. null Erfolg F Fehler
Status.Code	Die Ursache für den Fehler, falls vorhanden. Es ist nur ein Wert möglich: • RequestFailed
Status.Description	Eine Beschreibung des Problems, wenn eines vorliegt. Es ist nur ein Wert möglich: Did not return multiples Die Eingabeadresse stimmte nur mit einer Adresse in der Datenbank überein. GetCandidateAddressesLoqate gibt nur Daten zurück, wenn mehrere mögliche Übereinstimmungen gefunden wurden.

GetCityStateProvince

GetCityStateProvince gibt Ort und Bundesland/Kanton für eine angegebene Eingabepostleitzahl zurück.

Anmerkung: GetCityStateProvince funktioniert nur mit US-amerikanischen und kanadische Adressen.

GetCityStateProvince ist Teil des Universal Addressing-Moduls.

Eingabe

In der folgenden Tabelle sind die Eingabefelder aufgeführt.

Tabelle 73: Eingabe von GetCityStateProvince

columnName	Beschreibung
PostalCode	Eine US-amerikanische (ZIP Code™) oder kanadische Postleitzahl in einem der folgenden Formate: 99999 99999-9999 A9A9A9 A9A 9A9

Optionen

Tabelle 74: Optionen für GetCityStateProvince

optionName	Beschreibung
PerformUSProcessing	<p>Gibt an, ob folgende Adressen verarbeitet werden sollen: US-amerikanische Adressen. Wenn Sie die Verarbeitung US-amerikanischer Adressen aktivieren, versucht GetCityStateProvince, den Bundesstaat zurückzugeben für US-amerikanische Adressen. Wenn Sie die Verarbeitung US-amerikanischer Adressen deaktivieren, schlagen US-amerikanische Adressen fehl. Das bedeutet, dass diese mit einem „F“ im Ausgabefeld „Status“ zurückgegeben werden. Das Ausgabefeld „Status.Code“ enthält dann „DisabledCoder“. Wenn Sie die Verarbeitung US-amerikanischer Adressen nicht lizenziert haben, müssen Sie die Verarbeitung US-amerikanischer Adressen deaktivieren, damit Ihre Aufträge erfolgreich abgeschlossen werden, auch mit US-amerikanischen Adressen.</p> <p>Anmerkung: Sie müssen über eine gültige Lizenz für die Verarbeitung US-amerikanischer Adressen verfügen für die erfolgreiche Verarbeitung US-amerikanischer Adressen. Wenn Sie die Verarbeitung US-amerikanischer Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben. Wenn Sie die Verarbeitung US-amerikanischer Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben.</p> <p>Y Ja, US-amerikanische Adressen verarbeiten (Standard).</p> <p>N Nein, keine Verarbeitung von US-amerikanischen Adressen.</p>
Database.US	<p>Gibt die für die Verarbeitung US-amerikanischer Adressen zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich US-Datenbankressourcen definiert wurden.</p>

optionName	Beschreibung
PerformCanadianProcessing	<p>Gibt an, ob kanadische Adressen verarbeitet werden sollen. Wenn Sie die Verarbeitung kanadischer Adressen aktivieren, versucht GetCityStateProvince, die Provinz für kanadische Adressen zurückzugeben. Wenn Sie die Verarbeitung kanadischer Adressen deaktivieren, schlagen kanadische Adressen fehl. Das bedeutet, dass diese mit einem „F“ im Ausgabefeld „Status“ zurückgegeben werden. Das Ausgabefeld „Status.Code“ enthält dann „DisabledCoder“. Wenn Sie die Verarbeitung kanadischer Adressen nicht lizenziert haben, müssen Sie die Verarbeitung kanadischer Adressen deaktivieren, damit Ihre Aufträge erfolgreich abgeschlossen werden, auch wenn sie kanadische Adressen enthalten.</p> <p>Anmerkung: Sie müssen über eine gültige Lizenz für die Verarbeitung kanadischer Adressen verfügen, um kanadische Adressen erfolgreich zu verarbeiten. Wenn Sie die Verarbeitung kanadischer Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben. Wenn Sie die Verarbeitung kanadischer Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben.</p> <p>Y Ja, kanadische Adressen verarbeiten (Standard).</p> <p>N Nein, kanadische Adressen nicht verarbeiten.</p>
Database.Canada	<p>Gibt die für die Verarbeitung von kanadischen Adressen zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich Kanadische Datenbankressourcen definiert wurden.</p>
OutputVanityCity	<p>Gibt an, ob inoffizielle Ortsnamen in die Ausgabe eingeschlossen werden sollen. Ein inoffizieller Ortsname ist eine alternativer Name für den primären Ortsnamen. Beispielsweise ist „Hollywood“ ein inoffizieller Ortsname für Los Angeles.</p> <p>Y Ja, inoffizielle Ortsnamen einschließen.</p> <p>N Nein, inoffizielle Ortsnamen nicht einschließen (Standard).</p>
MaximumResults	<p>Gibt die maximale Anzahl von Ort-Bundesland/Kanton-Paaren an, die zurückgegeben werden sollen. Der Standardwert ist 10.</p>

Ausgabe

GetCityStateProvince gibt den übereinstimmenden Ort und Bundesland/Kanton für die Eingabepostleitzahl zurück sowie einen Code, der den Erfolg oder Fehler des Übereinstimmungsversuchs anzeigt. Wenn mehr als ein Ort/Bundesland oder Ort/Kanton mit der Eingabepostleitzahl übereinstimmt, werden mehrere Ausgabedatensätze zurückgegeben.

Tabelle 75: Ausgabe von GetCityStateProvince

columnName	Beschreibung
City	Der übereinstimmende Ortsname.
City.Type	Der nach USPS® standardisierte Ortsname (nur US-amerikanische Adressen). V Alternativer (inoffizieller) Ortsname. P Primär. Der Ortsname ist der primäre, offizielle Ortsname. S Sekundär. Der Ortsname ist ein alternativer, aber zulässiger Ortsname. Ein Ort kann mehrere sekundäre Ortsnamen haben.
PostalCode	Die eingegebene Postleitzahl.
ProcessedBy	Zeigt an, welcher Adressen-Codierer die Adresse verarbeitet. Zur Auswahl stehen: USA Die US-amerikanischen Adressen-Codierer hat die Adresse verarbeitet. CAN Der Codierer kanadischer Adressen hat die Adresse verarbeitet.
StateProvince	Die Abkürzung für das Bundesland oder den Kanton.
Status	Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs. null Erfolg F Fehler
Status.Code	Die Ursache für den Fehler, falls vorhanden. Der einzige gültige Wert ist: <ul style="list-style-type: none"> • DisabledCoder • UnrecognizedPostalCode

columnName	Beschreibung
Status.Description	<p>Die Beschreibung des Fehlers. Gültige Werte sind:</p> <p>Postal code not found Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=UnrecognizedPostalCode.</p> <p>PerformUSProcessing disabled Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=DisabledCoder.</p> <p>PerformCanadianProcessing disabled Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=DisabledCoder.</p>

GetCityStateProvinceLoqate

GetCityStateProvinceLoqate gibt Ort und Bundesland/Kanton für eine angegebene Eingabepostleitzahl zurück.

Dieser Schritt ist Teil des Universal Addressing-Moduls.

Eingabe

In der folgenden Tabelle sind die Eingabefelder aufgeführt.

Tabelle 76: Eingabe für GetCityStateProvinceLoqate

columnName	Beschreibung
Country	<p>Der Ländercode oder Ländername, in einem der folgenden Formate:</p> <ul style="list-style-type: none"> • zweistelliger ISO-Ländercode • dreistelliger UPU-Ländercode • Englischer Ländername <p>Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.</p>
PostalCode	Die Postleitzahl zu der Adresse.

Optionen

Tabelle 77: Optionen für GetCityStateProvinceLoqate

columnName	Beschreibung/gültige Werte						
Database.Loqate	Gibt die für die Verarbeitung von Adressen zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich Datenbankressourcen definiert wurden.						
MaximumResults	Die maximale Anzahl an Adressen, die GetCityStateProvinceLoqate zurückgeben soll. Der Standardwert ist 10.						
OutputScript	Gibt das Alphabet oder die Schrift an, in dem oder in der die Ausgabe zurückgegeben werden soll. Diese Option ist bidirektional und ist im Allgemeinen als von Muttersprache zu Latein oder als von Latein zu Muttersprache festgelegt. <table border="0" style="margin-left: 20px;"> <tr> <td style="vertical-align: top;">Input</td> <td>Es wird keine Transkription durchgeführt. Die Ausgabe erfolgt in derselben Schrift wie die Eingabe (Standard).</td> </tr> <tr> <td style="vertical-align: top;">Native</td> <td>Die Ausgabe erfolgt in der muttersprachlichen Schrift des ausgewählten Landes, falls möglich.</td> </tr> <tr> <td style="vertical-align: top;">Latn</td> <td>Englische Werte werden verwendet.</td> </tr> </table>	Input	Es wird keine Transkription durchgeführt. Die Ausgabe erfolgt in derselben Schrift wie die Eingabe (Standard).	Native	Die Ausgabe erfolgt in der muttersprachlichen Schrift des ausgewählten Landes, falls möglich.	Latn	Englische Werte werden verwendet.
Input	Es wird keine Transkription durchgeführt. Die Ausgabe erfolgt in derselben Schrift wie die Eingabe (Standard).						
Native	Die Ausgabe erfolgt in der muttersprachlichen Schrift des ausgewählten Landes, falls möglich.						
Latn	Englische Werte werden verwendet.						
FailJobOnDataLicenseError	Gibt an, wie Spectrum Technology Platform reagieren soll, wenn ein Datenlizenzfehler auftritt. <table border="0" style="margin-left: 20px;"> <tr> <td style="vertical-align: top;">Auftrag als ungültig erklären</td> <td>Der gesamte Auftrag wird als ungültig erklärt, wenn ein Datenlizenzfehler auftritt.</td> </tr> <tr> <td style="vertical-align: top;">Datensatz als ungültig erklären</td> <td>Ein oder mehrere Datensätze werden als ungültig erklärt, wenn ein Datenlizenzfehler auftritt. Die Verarbeitung wird fortgesetzt.</td> </tr> </table>	Auftrag als ungültig erklären	Der gesamte Auftrag wird als ungültig erklärt, wenn ein Datenlizenzfehler auftritt.	Datensatz als ungültig erklären	Ein oder mehrere Datensätze werden als ungültig erklärt, wenn ein Datenlizenzfehler auftritt. Die Verarbeitung wird fortgesetzt.		
Auftrag als ungültig erklären	Der gesamte Auftrag wird als ungültig erklärt, wenn ein Datenlizenzfehler auftritt.						
Datensatz als ungültig erklären	Ein oder mehrere Datensätze werden als ungültig erklärt, wenn ein Datenlizenzfehler auftritt. Die Verarbeitung wird fortgesetzt.						

Ausgabe

GetCityStateProvinceLoqate gibt den übereinstimmenden Ort und Bundesland/Kanton für die Eingabepostleitzahl zurück sowie einen Code, der den Erfolg oder Fehler des Übereinstimmungsversuchs anzeigt. Wenn mehr als ein Ort/Bundesland oder Ort/Kanton mit der Eingabepostleitzahl übereinstimmt, werden mehrere Ausgabedatensätze zurückgegeben.

Tabelle 78: Ausgabe für GetCityStateProvinceLoqate

columnName	Beschreibung
City	Der übereinstimmende Ortsname.
Country	Das Land im Format, das durch Ihre Auswahl für OutputCountryFormat bestimmt wird: <ul style="list-style-type: none"> • ISO-Code • UPU-Code • Englisch
PostalCode	Die eingegebene Postleitzahl.
ProcessedBy	Zeigt an, welcher Adressen-Codierer die Adresse verarbeitet. LOQATE Der Loqate-Codierer verarbeitet die Adressen.
StateProvince	Die Abkürzung für das Bundesland oder den Kanton.
Status	Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs. null Erfolg F Fehler
Status.Code	Die Ursache für den Fehler, falls vorhanden. Der einzige gültige Wert ist: <ul style="list-style-type: none"> • UnrecognizedPostalCode
Status.Description	Die Beschreibung des Fehlers. Der einzige gültige Wert ist: Postal code not found Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=UnrecognizedPostalCode.

GetPostalCodes

Mit GetPostalCodes können Sie die Postleitzahlen für einen bestimmten Ort nachschlagen. Der Dienst akzeptiert Ort, Bundesland/Kanton und Land als Eingabe und gibt die Postleitzahlen für diesen Ort zurück. Die Eingabe muss korrekt sein, damit Postleitzahlen zurückgegeben werden.

Anmerkung: GetPostalCodes funktioniert nur mit US-amerikanischen Adressen.

GetPostalCodes ist Teil des Universal Addressing-Moduls.

Eingabe

GetPostalCodes akzeptiert Ort, Bundesland/Kanton und Land als Eingabe.

Tabelle 79: Eingabe für GetPostalCodes

columnName	Beschreibung
City	<p>Der Ort, nach dessen Postleitzahlen Sie suchen möchten.</p> <p>Sie können Ort und Bundesland/Kanton in das Feld „City“ eingeben. In dem Fall müssen Sie das Feld „StateProvince“ leer lassen.</p> <p>Die Gesamtlänge der Felder „City“ und „StateProvince“ darf 100 Zeichen nicht überschreiten.</p>
StateProvince	<p>Das Bundesland oder der Kanton des Orts, nach dessen Postleitzahlen Sie suchen möchten.</p> <p>Sie können Bundesland/Kanton in das Feld „City“ anstatt in das Feld „StateProvince“ eingeben.</p> <p>Die Gesamtlänge der Felder „City“ und „StateProvince“ darf 100 Zeichen nicht überschreiten.</p>
Country	<p>Der Ländercode oder -name des Orts, nach dessen Postleitzahlen Sie suchen möchten. Der einzige gültige Wert ist „US“.</p>

Optionen

Tabelle 80: Optionen für GetPostalCodes

optionName	Beschreibung
Database.US	Gibt die für die Postleitzahlensuche zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich „US-Datenbankressourcen“ definiert wurden.
IncludeVanityCity	<p>Gibt an, ob Postleitzahlen für die inoffiziellen Ortsnamen eingeschlossen werden. Ein inoffizieller Ortsname ist eine alternativer Name für den primären Ortsnamen. Beispielsweise ist „Hollywood“ ein inoffizieller Ortsname für Los Angeles.</p> <p>Y Ja, Postleitzahlen für inoffizielle Ortsnamen einschließen.</p> <p>N Nein, Postleitzahlen für inoffizielle Ortsnamen nicht einschließen (Standard).</p>
OutputCityType	<p>Gibt an, ob der Ortstyp in der Ausgabe zurückgegeben werden soll. Bei Aktivierung wird der Ortstyp im Feld „City.Type“ zurückgegeben.</p> <p>Y Ja, den Ortstyp in die Ausgabe einschließen.</p> <p>N Nein, den Ortstyp nicht in die Ausgabe einschließen (Standard).</p>

Ausgabe

GetPostalCodes gibt die Postleitzahlen für einen angegebenen Ort zurück. Jede Postleitzahl wird in einem separaten Datensatz zusammen mit den in der folgenden Tabelle aufgelisteten Daten zurückgegeben.

Tabelle 81: Ausgabe von `GetPostalCodes`

columnName	Beschreibung
City.Type	<p>Der USPS®-Ortstyp (US-amerikanische Adressen). Der Ortstyp wird durch Nachschlagen des ZIP-Codes und des Ortsnamens bestimmt. Der Ort Lanham MD verfügt beispielsweise über die Postleitzahlen 20703, 20706 und 20784. Lanham ist der primäre Ort in 20703 und 20706, aber ein alternativer Ortsname in 20784.</p> <p>Diese Feldspalte wird nur ausgefüllt, wenn <code>OutputCityType=Y</code>. Die folgenden Werte sind möglich:</p> <p>V Alternativer (inoffizieller) Ortsname.</p> <p>P Primär. Der Ortsname ist der primäre, offizielle Ortsname.</p> <p>S Sekundär. Der Ortsname ist ein alternativer, aber zulässiger Ortsname. Ein Ort kann mehrere sekundäre Ortsnamen haben.</p>
PostalCode	Eine Postleitzahl im angegebenen Ort.
ProcessedBy	Da dieser Dienst nur mit US-amerikanischen Adressen funktioniert, enthält „ProcessedBy“ immer nur einen Wert: USA.
Status	<p>Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs.</p> <p>null Erfolg</p> <p>F Fehler</p>
Status.Code	<p>Ursache für den Fehler, falls vorhanden. Zur Auswahl stehen:</p> <ul style="list-style-type: none"> • CountryNotSupported • UnableToLookup
Status.Description	<p>Beschreibung des Fehlers.</p> <ul style="list-style-type: none"> • Eingabeland wird nicht unterstützt. • Eingabeort war leer. • Ort und Bundesland/Kanton der Eingabe waren leer, oder es wurde keine Übereinstimmung gefunden. • Keine Übereinstimmung bei Ort-Bundesland/Kanton (unterschiedliche Schreibweisen gefunden, Ort-Bundesland/Kanton war ein alternativer Name und Vergleichen alternativer Namen war nicht zulässig oder Ort-Bundesland/Kanton stimmte nicht mit dem ZIP-Code überein).

GetPostalCodesLoqate

Mit GetPostalCodesLoqate können Sie die Postleitzahlen für einen bestimmten Ort nachschlagen. Der Dienst akzeptiert Ort, Bundesland/Kanton und Land als Eingabe und gibt die Postleitzahlen für diesen Ort zurück. Die Eingabe muss korrekt sein, damit Postleitzahlen zurückgegeben werden.

GetPostalCodesLoqate ist Teil des Universal Addressing-Moduls.

Eingabe

GetPostalCodesLoqate akzeptiert Ort, Bundesland/Kanton und Land als Eingabe.

Tabelle 82: Eingabe für GetPostalCodesLoqate

columnName	Beschreibung/gültige Werte
City	<p>Der Ort, nach dessen Postleitzahlen Sie suchen möchten.</p> <p>Sie können Ort und Bundesland/Kanton in die Spalte „City“ eingeben. In dem Fall müssen Sie die Spalte „StateProvince“ leer lassen.</p>
Country	<p>Der Ländercode oder Ländername, in einem der folgenden Formate:</p> <ul style="list-style-type: none"> • zweistelliger ISO-Ländercode • dreistelliger UPU-Ländercode • Englischer Ländername <p>Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.</p>
StateProvince	<p>Das Bundesland oder der Kanton des Orts, nach dessen Postleitzahlen Sie suchen möchten.</p> <p>Sie können Bundesland/Kanton in der Spalte „City“ anstatt in der Spalte „StateProvince“ eingeben.</p>

Optionen

Tabelle 83: Optionen für GetPostalCodesLoqate

optionName	Beschreibung/Gültige Werte				
Database.Loqate	Gibt die für die Postleitzahlensuche zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console definiert wurden.				
FailJobOnDataLicenseError	Gibt an, wie Spectrum Technology Platform reagieren soll, wenn ein Datenlizenzfehler auftritt. <table border="0" style="margin-left: 20px;"> <tr> <td style="vertical-align: top;">Auftrag als ungültig erklären</td> <td>Der gesamte Auftrag wird als ungültig erklärt, wenn ein Datenlizenzfehler auftritt.</td> </tr> <tr> <td style="vertical-align: top;">Datensatz als ungültig erklären</td> <td>Ein oder mehrere Datensätze werden als ungültig erklärt, wenn ein Datenlizenzfehler auftritt. Die Verarbeitung wird fortgesetzt.</td> </tr> </table>	Auftrag als ungültig erklären	Der gesamte Auftrag wird als ungültig erklärt, wenn ein Datenlizenzfehler auftritt.	Datensatz als ungültig erklären	Ein oder mehrere Datensätze werden als ungültig erklärt, wenn ein Datenlizenzfehler auftritt. Die Verarbeitung wird fortgesetzt.
Auftrag als ungültig erklären	Der gesamte Auftrag wird als ungültig erklärt, wenn ein Datenlizenzfehler auftritt.				
Datensatz als ungültig erklären	Ein oder mehrere Datensätze werden als ungültig erklärt, wenn ein Datenlizenzfehler auftritt. Die Verarbeitung wird fortgesetzt.				

Ausgabe

GetPostalCodesLoqate gibt die Postleitzahlen für einen angegebenen Ort zurück. Jede Postleitzahl wird in einem separaten Datensatz zusammen mit den in der folgenden Tabelle aufgelisteten Daten zurückgegeben.

Tabelle 84: Ausgabe für GetPostalCodesLoqate

columnName	Beschreibung/gültige Werte				
PostalCode	Eine Postleitzahl im angegebenen Ort.				
ProcessedBy	Zeigt an, welcher Adressen-Codierer die Adresse verarbeitet. <table border="0" style="margin-left: 20px;"> <tr> <td style="vertical-align: top;">LOQATE</td> <td>Der Loqate-Codierer verarbeitet die Adressen.</td> </tr> </table>	LOQATE	Der Loqate-Codierer verarbeitet die Adressen.		
LOQATE	Der Loqate-Codierer verarbeitet die Adressen.				
Status	Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs. <table border="0" style="margin-left: 20px;"> <tr> <td style="vertical-align: top;">null</td> <td>Erfolg</td> </tr> <tr> <td style="vertical-align: top;">F</td> <td>Fehler</td> </tr> </table>	null	Erfolg	F	Fehler
null	Erfolg				
F	Fehler				

columnName	Beschreibung/gültige Werte
Status.Code	Ursache für den Fehler, falls vorhanden. Zur Auswahl stehen: <ul style="list-style-type: none"> InvalidCountry UnableToLookup
Status.Description	Beschreibung des Fehlers. <ul style="list-style-type: none"> Eingabeland wird nicht unterstützt. Eingabeort war leer. Ort und Bundesland/Kanton der Eingabe waren leer, oder es wurde keine Übereinstimmung gefunden.

Validate Address

„Validate Address“ standardisiert und prüft Adressen mithilfe der Adressdaten von Postbehörden. „Validate Address“ kann Daten korrigieren und die Adresse entsprechend dem durch die jeweilige Postbehörde bevorzugten Format formatieren. Durch „Validate Address“ werden zudem fehlende postalische Daten wie Postleitzahlen, Ortsnamen, Namen von Bundesländern/Kantonen usw. hinzugefügt.

Des Weiteren gibt „Validate Address“ Indikatoren zu Vergleichsversuchen zurück, z. B. ob „Validate Address“ die Adresse geprüft hat, das Zuverlässigkeitsniveau der zurückgegebenen Adresse, den Fehlergrund, wenn die Adresse nicht validiert werden konnte, usw.

Während des Adressenvergleichs und der Standardisierung trennt „Validate Address“ Adresszeilen in Komponenten und vergleicht sie mit den Inhalten der Datenbanken des Universal Addressing-Moduls. Wenn eine Übereinstimmung ermittelt wird, wird die eingegebene Adresse entsprechend den Informationen in der Datenbank *standardisiert*. Wenn keine Übereinstimmung in der Datenbank gefunden wird, *formatiert* „Validate Address“ die eingegebenen Adressen (optional). Bei dem Formatierungsvorgang wird versucht, die Adresszeilen gemäß den Standards der jeweiligen Postbehörde zu strukturieren.

„Validate Address“ ist Teil des Universal Addressing-Moduls.

Eingabe

„ValidateAddress“ nutzt eine Adresse als Eingabe. Alle Adressen verwenden dieses Format ungeachtet des Landes der Adresse. Unter [Adresszeilenverarbeitung für US-amerikanische Adressen](#) auf Seite 403 finden Sie wichtige Informationen über die Verarbeitung von Adresszeilendaten für US-amerikanische Adressen.

Tabelle 85: Eingabeformat

columnName	Format	Beschreibung
AddressLine1	Zeichenfolge [50]	Die erste Adresszeile.
AddressLine2	Zeichenfolge [50]	Die zweite Adresszeile.
AddressLine3	Zeichenfolge [50]	Die dritte Adresszeile. Gilt nicht für kanadische Adressen.
AddressLine4	Zeichenfolge [50]	Die vierte Adresszeile. Gilt nicht für kanadische Adressen.
AddressLine5	Zeichenfolge [50]	Die fünfte Adresszeile. Gilt nur für britische Adressen. Kann Straßennamen, Einheitsnummer, Gebäudenummer usw. enthalten.
City	Zeichenfolge [50]	Der Ortsname. Bei US-amerikanischen Adressen können Sie den Ort, den Staat und den ZIP-Code™ im Feld „City“ angeben. Dabei müssen Sie die Felder „StateProvince“ und „PostalCode“ leer lassen.
StateProvince	Zeichenfolge [50]	Das Bundesland oder der Kanton. Bei US-amerikanischen Adressen können Sie den Bundesstaat in das Feld „City“ anstelle des Feldes „StateProvince“ eintragen.

columnName	Format	Beschreibung
PostalCode	Zeichenfolge [10]	<p>Die Postleitzahl der Adresse in einem der folgenden Formate:</p> <p>99999 99999-9999 A9A9A9 A9A 9A9 9999 999</p> <p>Bei US-amerikanischen Adressen können Sie den ZIP-Code™ in das Feld „City“ eintragen.</p> <p>Bei US-amerikanischen Adressen kann „Validate Address“ die Daten parsen und die Adresse erfolgreich verarbeiten, wenn Ort/Bundesstaat/ZIP-Code™ im Feld „PostalCode“ stehen. Beste Ergebnisse erhalten Sie jedoch, wenn Sie diese Daten in die entsprechenden Felder (City, StateProvince und PostalCode) eintragen.</p>
Country	Zeichenfolge [50]	<p>Der Ländercode oder Ländername, in einem der folgenden Formate:</p> <ul style="list-style-type: none"> • Zweistelliger Ländercode gemäß ISO 3116-1 Alpha-2. • Dreistelliger Ländercode gemäß ISO 3116-1 Alpha-3. • Englischer Ländername • Französischer Ländername • Deutscher Ländername • Spanischer Ländername <p>Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.</p>
FirmName	Zeichenfolge [50]	Der Unternehmens- bzw. Firmenname.
USUrbanName	Zeichenfolge [50]	Der US-amerikanische Name der Wohnanlage. Dies wird vor allem für puerto-ricanische Adressen verwendet.
CustomerID	Zeichenfolge [9]	Wenn diese Postsendung einen allgemeinen Strichcode verwendet, geben Sie Ihre von USPS® zugewiesene Kunden-ID in das Feld ein. Der allgemeine Strichcode von „Validate Address“ wird für Postsendungen verwendet, die den OneCode ACS®-Dienst nutzen.

columnName	Format	Beschreibung
CanLanguage	Zeichenfolge	<p>Gibt nur bei kanadischen Adressen an, ob die Adresse in Englisch oder Französisch ist, wenn die Option <code>CanFrenchFormat=T</code> verwendet wird.</p> <p>Wenn dieses Feld leer ist, wird die Adresse in Englisch formatiert. Wenn das Feld einen Wert enthält, wird die Adresse in Französisch formatiert. Beachten Sie, dass Adressen in Quebec ungeachtet des Wertes in diesem Feld immer in Französisch formatiert wird.</p>

Adresszeilenverarbeitung für US-amerikanische Adressen

Die Eingabefelder „AddressLine1“ bis „AddressLine4“ werden für US-amerikanische Adressen unterschiedlich behandelt, je nachdem ob die Optionen für die Firmennamen-Extraktion oder Wohnanlagen-Codeextraktion aktiviert sind. Wenn eine dieser Optionen aktiviert ist, prüft „ValidateAddress“ die Daten in allen vier Feldern, um die Adresse zu überprüfen und die angefragten Daten (Firmenname und/oder Wohnanlagencode) zu extrahieren. Wenn keine dieser Optionen aktiviert ist, verwendet „ValidateAddress“ nur die ersten beiden nicht leeren Adresszeilenfelder beim Überprüfungsversuch. Die Daten der anderen Adresszeilenfelder werden im Ausgabefeld „AdditionalInputData“ zurückgegeben. Beispiel:

AddressLine1: A1 Calle A
AddressLine2:
AddressLine3: URB Alamar
AddressLine4: Pitney Bowes

Wenn in dieser Adresse die Firmennamen-Extraktion oder die Wohnanlagen-Codeextraktion aktiviert wäre, würde „ValidateAddress“ alle vier Adresszeilen untersuchen. Wenn weder die Firmennamen-Extraktion noch die Wohnanlagen-Codeextraktion aktiviert ist, würde „ValidateAddress“ die Felder „AddressLine1“ und „AddressLine3“ (die ersten beiden nicht leeren Adresszeilen) untersuchen und versuchen, die Adresse mithilfe dieser Daten zu überprüfen. Die Daten in „AddressLine4“ würden im Ausgabefeld „AdditionalInputData“ zurückgegeben werden.

Optionen

Ausgabedatenoptionen

Die folgende Tabelle führt die Optionen auf, die den Informationstyp steuern, der von „ValidateAddress“ zurückgegeben wird. Einige dieser Optionen können für kanadische Adressen überschrieben werden. Weitere Informationen finden Sie unter [Optionen für kanadische Adressen](#) auf Seite 435.

Tabelle 86: Ausgabedatenoptionen

optionName	Beschreibung
OutputRecordType	<p data-bbox="683 436 1427 495">Typ des Ausgabedatensatzes. Geben Sie bei mehr als einem Typ eine Liste an.</p> <ul style="list-style-type: none"> <li data-bbox="683 512 1427 953"> <p>A Gibt 1–4 Zeilen der Adressdaten plus Informationen zu Ort, Bundesland/-staat, Postleitzahl, Firmenname und Wohnanlage zurück. Jede Adresszeile steht für eine Zeile der Adresse, wie sie tatsächlich auf einem Briefumschlag erscheinen würde. Weitere Informationen finden Sie unter Ausgabe auf Seite 446. Wenn die Adresse überprüft wird, enthalten die Adresszeilen die standardisierte Adresse. Bei der Adressenstandardisierung wird die Interpunktion entfernt, Richtungsangaben und Straßensuffixe werden abgekürzt und Adresselemente korrigiert. Wenn die Adresse nicht überprüft wird, enthalten die Adresszeilen die Adresse, wie sie in der Eingabe vorgekommen ist („Durchlaufdaten“). Nicht überprüfte Adressen werden immer als Durchlaufdaten in die Adresszeilenfelder eingeschlossen, selbst wenn Sie <code>OutputRecordType=A</code> nicht angeben.</p> <li data-bbox="683 974 1427 1255"> <p>E Geparste Adresselemente. Jede Komponente der Adresse, z. B. die Hausnummer, der Straßename, das Straßensuffix, Richtungsangaben usw., wird in einem separaten Feld zurückgegeben. Weitere Informationen finden Sie unter Ausgabe von geparsten Adresselementen auf Seite 448. Beachten Sie, dass bei Angabe von „E“ und <code>OutputFormattedOnFail=Y</code> die geparsten Adresselemente die Eingabeadresse für Adressen enthalten, die nicht überprüft werden konnten.</p> <li data-bbox="683 1276 1427 1625"> <p>I Geparste Eingabe. Diese Option gibt die Eingabeadresse in geparster Form zurück, unabhängig davon, ob die Adresse überprüft wird. Jede Komponente der Eingabeadresse, z. B. die Hausnummer, der Straßename, das Straßensuffix, Richtungsangaben usw., wird in einem separaten Feld zurückgegeben. Die geparste Eingabe (Wert „I“) unterscheidet sich von der Kombination aus <code>OutputRecordType=E</code> und <code>OutputFormattedOnFail=Y</code> insofern, als dass „I“ alle Eingabeadressen in geparster Form zurückgibt, und nicht nur Eingaben, die nicht überprüft werden konnten. Weitere Informationen finden Sie unter Geparste Eingabe auf Seite 451.</p> <li data-bbox="683 1646 1427 1738"> <p>P Postalische Daten. Ausgabeadressen enthalten zusätzliche Daten für jede überprüfte Adresse. Weitere Informationen finden Sie unter Ausgabe von postalischen Daten auf Seite 453.</p> <li data-bbox="683 1759 1427 1780"> <p>Leer Kein Adressdaten oder postalischen Daten zurückgeben.</p>

optionName	Beschreibung
OutputFieldLevelReturnCodes	<p>Gibt an, ob Ergebnisindikatoren auf Feldebene eingeschlossen werden sollen. Ergebnisindikatoren auf Feldebene erläutern, wie jedes einzelne Adresselement verarbeitet wurde. Ergebnisindikatoren auf Feldebene werden im Begrenzungszeichen „Ergebnis“ zurückgegeben. Der Ergebnisindikator auf Feldebene für „HouseNumber“ ist beispielsweise in HouseNumber.Result enthalten. Eine vollständige Liste der Ausgabefelder für Ergebnisindikatoren finden Sie unter Ergebnisindikatoren auf Feldebene auf Seite 460.</p> <p>N Nein, keine Rückgabecodes auf Feldebene zurückgeben (Standard).</p> <p>Y Ja, Rückgabecodes auf Feldebene zurückgeben.</p>

optionName	Beschreibung
------------	--------------

OutputFormattedOnFail	
-----------------------	--

optionName

Beschreibung

Gibt an, ob eine formatierte Adresse zurückgegeben werden soll, wenn eine Adresse nicht überprüft werden kann. Die Adresse wird entsprechend dem durch das Adressland bevorzugten Adressenformat formatiert. Wird diese Option nicht ausgewählt, bleiben die Ausgabeadressfelder leer, wenn die Adresse nicht überprüft werden kann.

Anmerkung: Diese Option gilt nur für US-amerikanische und kanadische Adressen. Formatierte Daten werden für keine andere Adresse zurückgegeben.

- N** Nein, fehlgeschlagene Adressen nicht formatieren (Standard).
- Y** Ja, fehlgeschlagene Adressen formatieren.

Formatierte Adressen werden in dem über die Option `OutputRecordType` angegebenen Format zurückgegeben. Beachten Sie, dass bei Angabe von `OutputRecordType=E` die geparsten Adresselemente die geparste, überprüfte Adresse für Adressen enthalten, die überprüft werden konnten. Wenn die Adresse nicht überprüft werden konnte, enthalten die geparsten Adresselemente die Eingabeadresse in geparster Form. Wenn Sie immer möchten, dass die Ausgabe die Eingabeadresse in geparster Form enthält, unabhängig davon, ob „ValidateAddress“ die Adresse überprüfen konnte, geben Sie `OutputRecordType=I` an.

Formatierte Adressen werden in dem über die Option `Option.OutputRecordType` angegebenen Format zurückgegeben. Beachten Sie, dass bei Angabe von `Option.OutputRecordType=E` die geparsten Adresselemente die geparste, überprüfte Adresse für Adressen enthalten, die überprüft werden konnten. Wenn die Adresse nicht überprüft werden konnte, enthalten die geparsten Adresselemente die Eingabeadresse in geparster Form. Wenn Sie immer möchten, dass die Ausgabe die Eingabeadresse in geparster Form enthält, unabhängig davon, ob „ValidateAddress“ die Adresse überprüfen konnte, geben Sie `Option.OutputRecordType=I` an.

Formatierte Adressen werden im Format zurückgegeben, das durch die Kontrollkästchen **Standardadresse einschließen**, **Adresszeilenelemente einschließen** und **Postalische Informationen einschließen** aktiviert wurde. Beachten Sie, dass bei Auswahl von **Adresszeilenelemente einschließen** die geparsten Adresselemente die geparste, überprüfte Adresse für Adressen enthalten, die überprüft werden konnten. Wenn die Adresse nicht überprüft werden konnte, enthalten die geparsten Adresselemente die Eingabeadresse in geparster Form. Wenn Sie immer möchten, dass die Ausgabe die Eingabeadresse in geparster Form enthält, unabhängig davon, ob „ValidateAddress“ die Adresse überprüfen konnte, wählen Sie **Standardisierte Eingabeadresselemente einschließen** aus.

optionName	Beschreibung
	<p>Wenn Sie „Y“ angeben, müssen Sie „A“ und/oder „E“ für „OutputRecordType“ angeben.</p> <p>Wenn Sie „Y“ angeben, müssen Sie „A“ und/oder „E“ für <code>Option.OutputRecordType</code> angeben.</p> <p>Wenn Sie diese Option aktivieren, müssen Sie Standardadresse einschließen und/oder Adresszeilenelemente einschließen auswählen.</p>
OutputStreetNameAlias	<p>Gibt nur bei US-amerikanischen Adressen an, ob in der Ausgabe ein Straßenalias verwendet werden soll. Bei einem Straßen-Alias handelt es sich um einen alternativen Namen für eine Straße, der in der Regel nur für einen bestimmten Bereich von Adressen in der Straße gilt. Wenn Sie keine Straßen-Aliasse in der Ausgabe zulassen, wird der „Basis“-Name der Straße in der Ausgabe angezeigt, unabhängig davon, ob ein Alias für die Straße vorhanden ist. Bei dem Basisnamen handelt es sich um den Namen, der für die gesamte Straße gilt.</p> <p>N Nein, keine Straßennamen-Aliasse in der Ausgabe zurückgeben.</p> <p>Y Ja, Straßennamen-Aliasse in der Ausgabe zurückgeben, wenn es einen Alias für die Straße gibt (Standard).</p>

optionName	Beschreibung
OutputStreetNameAlias	<p data-bbox="678 373 1427 527">Gibt nur bei US-amerikanischen Adressen an, wie in der Eingabe verwendete Straßennamen-Aliasse verarbeitet werden sollen. Bei einem Straßen-Alias handelt es sich um einen alternativen Namen für eine Straße, der in der Regel nur für einen bestimmten Bereich von Adressen in der Straße gilt.</p> <p data-bbox="678 537 1427 690">Wenn Sie diese Option aktivieren, erscheinen in der Eingabe verwendete Straßennamen-Aliasse in der Ausgabe . Wenn Sie diese Option nicht aktivieren, werden Straßennamen-Aliasse in der Eingabe bei der Ausgabe in den Basis-Straßennamen umgewandelt, mit folgenden Ausnahmen:</p> <ul data-bbox="678 701 1427 831" style="list-style-type: none"><li data-bbox="678 701 1427 768">• Wenn ein bevorzugter Alias bei der Eingabe verwendet wird, wird dieser immer in der Ausgabe verwendet.<li data-bbox="678 768 1427 831">• Bei der Eingabe verwendete geänderte Aliasse werden bei der Ausgabe immer in den Basis-Straßennamen umgewandelt. <p data-bbox="678 842 1427 982">Diese ist eine von drei Optionen, die steuern, wie „ValidateAddress“ Straßennamen-Aliasse verarbeitet. Die anderen beiden Optionen sind <code>OutputPreferredAlias</code> und <code>OutputAbbreviatedAlias</code>.</p> <p data-bbox="678 993 1427 1129">Anmerkung: Wenn <code>OutputAbbreviatedAlias</code> aktiviert wird, erscheint der abgekürzte Alias immer in der Ausgabe, selbst wenn Sie <code>OutputStreetNameAlias</code> deaktiviert haben.</p> <p data-bbox="678 1140 1427 1220">N Nein, keine Straßennamen-Aliasse in der Ausgabe zurückgeben.</p> <p data-bbox="678 1230 1427 1297">Y Ja, Straßennamen-Aliasse in der Ausgabe zurückgeben, wenn der Straßenname bei der Eingabe ein Alias ist (Standard).</p>

optionName	Beschreibung
OutputAddressBlocks	<p data-bbox="678 373 1429 520">Gibt an, ob die Adresse in einem Format zurückgegeben werden soll, das dem Aufdruck auf einer tatsächlichen Postsendung entspricht. Jede Adresszeile wird in einem separaten Adressblockfeld zurückgegeben. Es kann bis zu neun Adressblock-Ausgabefelder geben: AddressBlock1 bis AddressBlock9.</p> <p data-bbox="678 537 1073 571">Zum Beispiel diese Eingabeadresse:</p> <p data-bbox="678 588 1084 735">AddressLine1: 4200 Parliament Place AddressLine2: Suite 600 City: Lanham StateProvince: MD PostalCode: 20706</p> <p data-bbox="678 751 1159 785">Dies führt zu folgender Adressblockausgabe:</p> <p data-bbox="678 802 1198 894">AddressBlock1: 4200 Parliament PL STE 600 AddressBlock2: Lanham MD 20706-1882 AddressBlock3: UNITED STATES OF AMERICA</p> <p data-bbox="678 911 1429 1100">„ValidateAddress“ formatiert die Adresse gemäß Postbehördenstandards in Adressblöcke. Der Ländername wird gemäß Universal Postal Union-Ländername zurückgegeben. Beachten Sie, dass die Option <code>OutputCountryFormat</code> keine Auswirkung auf den Ländernamen im Adressblock hat, sondern lediglich auf den Namen, der im Ausgabefeld Land zurückgegeben wird.</p> <p data-bbox="678 1117 1429 1230">Wenn bei Adressen außerhalb der USA und Kanada „ValidateAddress“ die Adresse nicht überprüfen kann, werden keine Adressblöcke zurückgegeben. Bei Adressen in den USA und Kanada werden Adressblöcke zurückgegeben, selbst wenn die Überprüfung fehlschlägt.</p> <p data-bbox="678 1247 1227 1310">N Nein, keine Adressblöcke zurückgeben. Standardeinstellung.</p> <p data-bbox="678 1327 1138 1360">Y Ja, Adressblöcke zurückgeben.</p>

optionName	Beschreibung
OutputAMAS	<p>Gibt an, ob die Adresse in einem Format zurückgegeben werden soll, das dem Ausdruck auf einer tatsächlichen Postsendung entspricht. Jede Adresszeile wird in einem separaten Adressblockfeld zurückgegeben. Es kann bis zu neun Adressblock-Ausgabefelder geben: AddressBlock1 bis AddressBlock9.</p> <p>Zum Beispiel diese Eingabeadresse:</p> <p>AddressLine1: 4200 Parliament Place AddressLine2: Suite 600 City: Lanham StateProvince: MD PostalCode: 20706</p> <p>Dies führt zu folgender Adressblockausgabe:</p> <p>AddressBlock1: 4200 Parliament PL STE 600 AddressBlock2: Lanham MD 20706-1882 AddressBlock3: UNITED STATES OF AMERICA</p> <p>„ValidateAddress“ formatiert die Adresse gemäß Postbehördenstandards in Adressblöcke. Der Ländername wird gemäß Universal Postal Union-Ländername zurückgegeben. Beachten Sie, dass die Option <code>OutputCountryFormat</code> keine Auswirkung auf den Ländernamen im Adressblock hat, sondern lediglich auf den Namen, der im Ausgabefeld Land zurückgegeben wird.</p> <p>Wenn bei Adressen außerhalb der USA und Kanada „ValidateAddress“ die Adresse nicht überprüfen kann, werden keine Adressblöcke zurückgegeben. Bei Adressen in den USA und Kanada werden Adressblöcke zurückgegeben, selbst wenn die Überprüfung fehlschlägt.</p> <p>N Nein, keine Adressblöcke zurückgeben. StandardEinstellung.</p> <p>Y Ja, Adressblöcke zurückgeben.</p>

Ermitteln von „Congressional Districts“

„Validate Address“ kann den US-amerikanischen „Congressional District“ für eine Adresse bestimmen. Zur Ermittlung von „Congressional Districts“ muss der Parameter `OutputRecordType` den Wert **P** enthalten. Weitere Informationen zu `OutputRecordType` finden Sie unter [Ausgabedatenoptionen](#) auf Seite 403.

Tabelle 87: Ausgabe eines „Congressional District“

Spaltenname	Beschreibung
USCongressionalDistrict	Nummer des „Congressional District“. Wenn die Adresse eine nicht staatliche Adresse ist (z. B. Puerto Rico oder Washington D.C.), ist dieses Feld leer.

Ermitteln von Bezirksnamen

„Validate Address“ kann den Bezirk bestimmen, in dem sich eine bestimmte Adresse befindet, und den Bezirksnamen zurückgeben.

Anmerkung: Bezirksnamen sind nur für US-amerikanische Adressen.

Zur Ermittlung von Bezirksnamen muss der Parameter `OutputRecordType` den Wert `P` enthalten. Weitere Informationen zu `OutputRecordType` finden Sie unter [Ausgabedatenoptionen](#) auf Seite 403.

Tabelle 88: Ausgabe des Bezirksnamens

Spaltenname	Beschreibung
USCountyName	Bezirksname

Ermitteln von FIPS-Bezirksnummern

„Federal Information Processing Standards“ (FIPS)-Bezirksnummern sind Nummern, die jeden Bezirk in einem Bundesstaat identifizieren. Beachten Sie, dass diese Nummern nur auf Bundesstaatsebene eindeutig sind. Dies gilt nicht auf nationaler Ebene. Weitere Informationen finden Sie unter <http://www.census.gov>.

Anmerkung: FIPS-Bezirksnummern sind nur für US-amerikanische Adressen.

Zur Ermittlung von FIPS-Bezirksnummern muss der Parameter `OutputRecordType` den Wert `P` enthalten. Weitere Informationen zu `OutputRecordType` finden Sie unter [Ausgabedatenoptionen](#) auf Seite 403.

Tabelle 89: Ausgabe einer FIPS-Bezirksnummer

columnName	Beschreibung
USFIPSCountyNumber	FIPS (Federal Information Processing Standards)-Bezirksnummer

Ermitteln von Beförderungsroutencodes

Beförderungsroutencodes sind eindeutige Identifikatoren, die jedem Postboten zugewiesen werden, der Post zustellt. Dadurch kann jede US-Zustellroute eindeutig identifiziert werden. „ValidateAddress“ kann den Code zurückgeben, der die Beförderungsrouten eines Adressaten darstellt.

Anmerkung: Beförderungsroutencodes sind nur für US-amerikanische Adressen.

Zur Ermittlung von Beförderungsroutencodes muss der Parameter `OutputRecordType` den Wert P enthalten. Weitere Informationen zu `OutputRecordType` finden Sie unter [Ausgabedatenoptionen](#) auf Seite 403.

Tabelle 90: Ausgabe des Beförderungsroutencodes

columnName	Beschreibung
USCarrierRouteCode	Beförderungsroutencode

Erstellen von „Delivery Point Barcodes“

Ein „Delivery Point Barcode“ (DPBC) ist eine POSTNET™-Strichcodedarstellung der Adresse. Er besteht aus 62 Strichen mit beginnenden und endenden Rahmenstrichen sowie fünf Strichen für den ZIP + 4®-Code, einem Wert basierend auf der Straßenanschriftennummer und einer Korrekturstelle. Der DPBC ermöglicht die automatisierte Sortierung von Briefsendungen entsprechend der Laufsequenz des Postboten. „Validate Address“ erstellt die Daten, die Sie zur Generierung eines DPBC benötigen.

Anmerkung: „Delivery Point Barcodes“ sind nur verfügbar für US-amerikanische Adressen. Weitere Informationen über „Delivery Point Barcodes“ finden Sie unter <http://www.usps.com>.

Zur Erstellung von notwendigen Daten für die Generierung eines DPBC muss `OutputRecordType` den Wert P enthalten. Weitere Informationen zu `OutputRecordType` finden Sie unter [Ausgabedatenoptionen](#) auf Seite 403.

Tabelle 91: Ausgabe eines „Delivery Point Barcode“

columnName	Beschreibung
PostalBarCode	Der Zustellortteil des „Delivery Point Barcode“.
USBCCheckDigit	Prüfstellenteil des 11-stelligen „Delivery Point Barcode“.

Verketten Sie zur Generierung eines DPBC die Werte der Ausgabespalten von „Validate Address“ wie folgt:

`PostalCode.Base + PostalCode.Addon + PostalBarcode + USBCCheckDigit`

Wenn Sie über folgende Werte verfügen:

- **PostalCode.Base** = 49423
- **PostalCode.Addon** = 4506
- **PostalBarcode** = 29
- **USBCCheckDigit** = 2

Würde der generierte Barcode wie folgt aussehen:

494234506292

Standardoptionen

Die folgende Tabelle führt die Optionen auf, die das Format und die Verarbeitung von Adressen steuern. Sie werden „Standardoptionen“ genannt, weil sie standardmäßig für alle Adressen gelten. Einige dieser Optionen können für kanadische Adressen überschrieben werden. Weitere Informationen finden Sie unter [Optionen für kanadische Adressen](#) auf Seite 435.

Tabelle 92: Standardoptionen

optionName	Beschreibung
OutputCasing	<p>Gibt die Groß-/Kleinschreibung der Ausgabeadresse an. Zur Auswahl stehen:</p> <p>M Gibt die Ausgabe in Groß- und Kleinbuchstaben zurück (Standard). Beispiel:</p> <p style="padding-left: 40px;">123 Main St Mytown FL 12345</p> <p>U Gibt die Ausgabe in Großbuchstaben zurück. Beispiel:</p> <p style="padding-left: 40px;">123 MAIN ST MYTOWN FL 12345</p>
OutputPostalCodeSeparator	<p>Gibt an, ob Trennzeichen (Leerzeichen oder Bindestriche) in ZIP™-Codes oder kanadischen Postleitzahlen verwendet werden sollen.</p> <p>Ein ZIP + 4®-Code mit dem Trennzeichen würde beispielsweise 20706-1844 und ohne Trennzeichen 207061844 lauten. Eine kanadische Postleitzahl mit dem Trennzeichen würde P5E"1S7 und ohne das Trennzeichen P5E1S7 lauten.</p> <p>Y Ja, Trennzeichen verwenden (Standard).</p> <p>N Nein, kein Trennzeichen verwenden.</p> <p>Anmerkung: Leerzeichen werden in kanadischen Postleitzahlen und Bindestriche in US-amerikanischen ZIP + 4®-Codes verwendet.</p>
OutputMultinationalCharacters	<p>Gibt an, ob internationale Zeichen, einschließlich diakritische Zeichen wie Umlaute oder Akzente, zurückgegeben werden sollen. (Wird nicht für US-amerikanische Adressen unterstützt).</p> <p>N Nein, keine internationalen Zeichen in der Ausgabe verwenden (Standard). Es werden nur Standard-ASCII-Zeichen zurückgegeben.</p> <p>Y Ja, internationalen Zeichen in der Ausgabe verwenden.</p>

optionName	Beschreibung
KeepMultimatch	<p>Gibt an, ob für Eingabeadressen mit mehreren möglichen Übereinstimmungen mehrere Adressen zurückgegeben werden sollen.</p> <p>Y Ja, Mehrfachübereinstimmungen zurückgeben (Standardeinstellung).</p> <p>N Nein, keine Mehrfachübereinstimmungen zurückgeben.</p> <p>Weitere Informationen finden Sie unter Rückgabe von Mehrfachübereinstimmungen auf Seite 420.</p>
StandardAddressFormat	<p>Gibt an, wo sekundäre Adressinformationen von US-amerikanischen Adressen platziert werden sollen. Sekundäre Adressinformationen beziehen sich auf Apartmentnummern, Suitenummern und ähnliche Bezeichnungen. In dieser Adresse beispielsweise lautet die sekundäre Adressinformation „Apt 10E“ und die primären Adressinformation „424 Washington Blvd“.</p> <p>Apt 10E 424 Washington Blvd Springfield MI 49423</p> <p>C Sowohl primäre als auch sekundäre Adressinformationen in „AddressLine1“ einfügen (Standard).</p> <p>S Primäre Adressinformationen in „AddressLine1“ und sekundäre Adressinformationen in „AddressLine2“ einfügen.</p> <p>D Sowohl primäre als auch sekundäre Adressinformationen in „AddressLine1“ einfügen und verworfene Adressinformationen von dualen Adressen in „AddressLine2“ einfügen. Eine duale Adresse ist eine Adresse, die sowohl Straßeninformationen als auch Postfach-, Landstraßen- oder Autobahninformationen enthält. Weitere Informationen finden Sie unter DualAddressLogic auf Seite 419.</p>

optionName	Beschreibung
OutputShortCityName	<p>Gibt an, wie kurze Ortsnamen oder inoffizielle alternative Ortsnamen formatiert werden sollen. Gilt für US-amerikanische und kanadische Adressen.</p> <ul style="list-style-type: none"> Y Gibt die vom USPS® genehmigte Abkürzung für den Ort zurück, sofern vorhanden. Der USPS® bietet Abkürzungen für Ortsnamen, die mindestens 14 Zeichen umfassen. Abkürzungen für Orte sind 13 Zeichen lang oder kürzer und können bei begrenztem Platz auf dem Adressticket verwendet werden. Wenn es keinen kurzen Ortsnamen für den Ort gibt, wird der komplette Ortsname zurückgegeben. N Gibt den langen Ortsnamen zurück (Standard). S Gibt den abgekürzten Ortsnamen nur zurück, wenn ein abgekürzter Ortsname in der Eingabeadresse verwendet wird. Wenn bei der Eingabeadresse kein kurzer Ortsname verwendet wird, kann entweder der kurze oder lange Ortsname zurückgegeben werden, je nach den USPS®-Bestimmungen für diesen bestimmten Ort. Wählen Sie diese Option aus, wenn Sie einen CASS™-Test durchführen. V Gib den inoffiziellen Ortsnamen (den alternativen Namen) aus, wenn der Ortsname bei der Eingabe ein inoffizieller Ortsname ist. Beispielsweise ist „Hollywood“ ein inoffizieller Ortsname für „Los Angeles“. Wenn Sie diese Option nicht auswählen und der eingegebene Ortsname ein inoffizieller Ortsname ist, wird die lange Version des offiziellen Ortsnamens zurückgegeben.
OutputCountryFormat	<p>Gibt das Format an, das für den im Ausgabefeld Country zurückgegebenen Ländernamen verwendet werden soll. Wenn Sie beispielsweise „Englisch“ auswählen, wird der Ländername „Deutschland“ als „Germany“ zurückgegeben.</p> <ul style="list-style-type: none"> E Englische Ländernamen verwenden (Standard). S Spanische Ländernamen verwenden. F Französische Ländernamen verwenden. G Deutsche Ländernamen verwenden. I ISO-Abkürzung mit zwei Buchstaben anstelle von Ländernamen für die Länder verwenden. U Abkürzung der Universal Postal Union anstelle von Ländernamen für die Länder verwenden.

optionName	Beschreibung
HomeCountry	<p>Gibt das Standardland an. Sie sollten das Land angeben, in dem sich die meisten Adressen befinden. Wenn Sie beispielsweise vor allem kanadische Adressen verarbeiten, geben Sie Kanada an.</p> <p>„ValidateAddress“ verwendet das von Ihnen angegebene Land, um eine Überprüfung einzuleiten, wenn es das Land nicht über die Adressfelder „StateProvince“, „PostalCode“ und „Country“ bestimmen kann. Gültige Ländernamen sind:</p> <p>Afghanistan, Albania, Algeria, American Somoa, Andorra, Angola, Anguilla, Antigua And Barbuda, Argentina, Armenia, Aruba, Australia, Austria, Azerbaijan, Bahamas, Bahrain, Bangladesh, Barbados, Belarus, Belgium, Belize, Benin, Bermuda, Bhutan, Bolivia, Bosnia And Herzegovina, Botswana, Brazil, British Virgin Islands, Brunei Darussalam, Bulgaria, Burkina Faso, Burundi, Cambodia, Cameroon, Canada, Cape Verde, Cayman Islands, Central African Republic, Chad, Chile, China, Colombia, Comoros Islands, Congo, Cook Islands, Costa Rica, Cote D'Ivoire, Croatia, Cuba, Cyprus, Czech Republic, Democratic Republic Of Congo, Denmark, Djibouti, Dominica, Dominican Republic, East Timor, Ecuador, Egypt, El Salvador, Equitorial Guinea, Eritrea, Estonia, Ethiopia, Falkland Islands, Faroe Islands, Federated States Of Micronesia, Fiji, Finland, France, French Guiana, Gabon, Gambia, Germany, Ghana, Gibraltar, Greece, Greenland, Grenada, Guadeloupe, Guam, Guatemala, Guinea, Guinea Bissau, Guyana, Haiti, Holy See, Honduras, Hong Kong, Hungary, Iceland, India, Indonesia, Iran, Iraq, Ireland, Israel, Italy, Jamaica, Japan, Jordan, Kazakhstan, Kenya, Kiribati, Korea, Kuwait, Kyrgyzstan, Laos, Latvia, Lebanon, Lesotho, Liberia, Libya, Liechtenstein, Lithuania, Luxembourg, Macau, Macedonia, Madagascar, Malawi, Malaysia, Maldives, Mali, Malta, Marshall Islands, Martinique, Mauritania, Mauritius, Mayotte, Mexico, Moldova, Monaco, Mongolia, Monserrat, Morocco, Mozambique, Myanmar, Namibia, Nauru, Nepal, Netherlands Antilles, New Caledonia, New Zealand, Nicaragua, Niger, Nigeria, Niue, Norway, Oman, Pakistan, Palau, Panama, Papua New Guinea, Paraguay, Peru, Philippines, Pitcairn Islands, Poland, Portugal, Puerto Rico, Qatar, Republic Of Georgia, Republic Of Korea, Republic Of Singapore, Reunion, Romania, Russia, Rwanda, Saint Helena, Saint Kitts And Nevis, Saint Lucia, Saint Pierre And Miquelon, Saint Vincent and the Grenadines, Samoa, San Marino, Sao Tome and Principe, Saudi Arabia, Senegal, Seychelles, Sierra Leone, Slovakia, Slovenia, Solomon Islands, Somalia, South Africa, Spain, Sri Lanka, Sudan, Surivalue, Swaziland, Sweden, Switzerland, Syria, Tahiti, Taiwan, Tajikistan, Tanzania, Thailand, The Netherlands, Togo, Tonga, Trinidad And Tobago, Tristan Da Cunha, Tunisia, Turkey, Turkmenistan, Turks And Caicos Islands, Tuvalu, Uganda, Ukraine, United Arab Emirates, United Kingdom, United States, Uruguay, Uzbekistan, Vanuatu, Venezuela, Vietnam, Virgin Islands (US), Wallis And Futuna, Yemen, Yugoslavia, Zambia, Zimbabwe</p>

optionName	Beschreibung
DualAddressLogic	<p>Gibt an, wie eine Übereinstimmung zurückgegeben werden soll, wenn mehrere nicht leere Adresszeilen vorhanden sind oder wenn eine Adresszeile mehrere Adresstypen enthält. (Nur US-amerikanische Adressen.)</p> <p>N (Standard) USPS® CASS™ -Bestimmungen geben anhand der folgenden Rangfolge die Formatierung der zurückgegebenen Adresse vor:</p> <ol style="list-style-type: none"> 1. Postfach 2. Firma 3. Hochhaus 4. Straße 5. Landstraße 6. Allgemeine Zustellung <p>S Gibt eine Straßenübereinstimmung unabhängig von der Adresszeile zurück.</p> <p>P Gibt eine Postfachübereinstimmung unabhängig von der Adresszeile zurück.</p> <p>Weitere Informationen finden Sie unter DualAddressLogic auf Seite 419.</p>

DualAddressLogic

Bei US-amerikanischen Adressen steuert die Option `DualAddressLogic`, ob „Validate Address“ eine Straßenübereinstimmung oder eine Postfach-/Landstraßen-/Autobahnübereinstimmung zurückgeben soll, wenn die Adresse sowohl Straßen- als auch Postfach-/Landstraßen-/Autobahninformationen in der gleichen Adresszeile enthält.

Anmerkung: Die Option `DualAddressLogic` hat keine Auswirkung, wenn die Straßeninformationen in einem anderen Adresszeileneingabefeld als die Postfach-/Landstraßen-/Autobahninformationen enthalten sind.

Hier ein Beispiel mit der folgenden Eingabeadresse:

AddressLine1: 401 N Main St Apt 1 POB 1
City: Kemp
StateProvince: TX
PostalCode: 75143

„ValidateAddress“ würde eines der folgenden Elemente zurückgeben:

- Wenn `DualAddressLogic` entweder auf N oder P eingestellt ist:

AddressLine1: PO Box 1
 City: Kemp
 StateProvince: TX
 PostalCode: 75143-0001

- Wenn `DualAddressLogic` auf S eingestellt ist:

AddressLine1: 401 N Main St Apt 1
 City: Kemp
 StateProvince: TX
 PostalCode: 75143-4806

Die Adressdaten, die nicht zum Standardisieren der Adresse verwendet werden, können an einem von zwei Orten zurückgegeben werden:

- **AddressLine2:** Die Adressinformationen, die nicht zum Standardisieren der Adresse verwendet werden, werden im Feld **AddressLine2** zurückgegeben, wenn Sie `StandardAddressFormat=D` angeben. Weitere Informationen finden Sie unter [Standardoptionen](#) auf Seite 414. Wenn Sie beispielsweise auswählen, eine Straßenübereinstimmung für duale Adressen zurückzugeben:

AddressLine1: 401 N Main St Apt 1
 AddressLine2: PO Box 1
 City: Kemp
 StateProvince: TX
 PostalCode: 75143-0001

- **AdditionalInputData:** Wenn Sie `StandardAddressFormat=D` nicht angeben, werden die Adressinformationen, die nicht zum Standardisieren der Adresse verwendet werden, im Feld **AdditionalInputData** zurückgegeben. Weitere Informationen zu dieser Option Sie unter [Standardoptionen](#) auf Seite 414. Wenn Sie beispielsweise auswählen, eine Straßenübereinstimmung für duale Adressen zurückzugeben:

AddressLine1: 401 N Main St Apt 1
 City: Kemp
 StateProvince: TX
 PostalCode: 75143-0001
 AdditionalInputData: PO Box 1

Verworfen Adressinformationen können abgerufen werden, indem Sie die Option `StandardAddressFormat` auf D einstellen. Weitere Informationen finden Sie unter [Standardoptionen](#) auf Seite 414.

Rückgabe von Mehrfachübereinstimmungen

Wenn „Validate Address“ mehrere Adressen in der Postdatenbank findet, die mögliche Übereinstimmungen für die Eingabeadresse sind, können Sie mithilfe von „Validate Address“ die möglichen Übereinstimmungen zurückgeben. Die folgende Adresse stimmt mit mehreren Adressen in der US-amerikanischen Postdatenbank überein:

PO BOX 1
New York, NY

Optionen

Verwenden Sie zum Zurückgeben von Mehrfachübereinstimmungen die in der folgenden Tabelle beschriebenen Optionen.

Tabelle 93: Option für Mehrfachübereinstimmung

optionName	Beschreibung
KeepMultimatch	<p>Gibt an, ob für Eingabeadressen mit mehreren möglichen Übereinstimmungen mehrere Adressen zurückgegeben werden sollen.</p> <p>Y Ja, Mehrfachübereinstimmungen zurückgeben (Standardeinstellung).</p> <p>N Nein, keine Mehrfachübereinstimmungen zurückgeben.</p>
MaximumResults	<p>Eine Zahl zwischen 1 und 10, die die maximale Anzahl zurückzugebender Adressen angibt.</p> <p>Der Standardwert ist 1.</p> <p>Anmerkung: Der Unterschied zwischen „Keepmultimatch=N“ und „KeepMultimatch=Y/MaximumResults=1“ ist, dass eine Mehrfachübereinstimmung einen Fehler zurückgibt, wenn „KeepMultimatch=N“, wobei eine Mehrfachübereinstimmung einen Datensatz zurückgibt, wenn „KeepMultimatch=Y“ und „MaximumResults=1“.</p>
OutputFieldLevelReturnCodes	<p>Zur Identifizierung, welche Ausgabeadressen Kandidatenadressen sind, müssen Sie den Wert Y für „OutputFieldLevelReturnCodes“ angeben. Dadurch enthalten Datensätze, die Kandidatenadressen sind, einen oder mehrere M-Werte in den Ergebnisindikatoren auf Feldebene.</p>

Ausgabe

Wenn Sie Mehrfachübereinstimmungen zurückgeben lassen, werden die Adressen in dem von Ihnen festgelegten Adressenformat zurückgegeben. Weitere Informationen zur Festlegung des Adressenformats finden Sie unter [Ausgabedatenoptionen](#) auf Seite 403. Suchen Sie zur Identifizierung, welche Datensätze Kandidatenadressen sind, nach mehreren M-Werten in den Ergebnisindikatoren auf Feldebene. Weitere Informationen finden Sie unter [Ergebnisindikatoren auf Feldebene](#) auf Seite 460.

US-amerikanische Adressoptionen

optionName	Beschreibung
PerformUSProcessing	<p>Gibt an, ob US-amerikanische Adressen verarbeitet werden sollen. Wenn Sie die Verarbeitung US-amerikanischer Adressen aktivieren, versucht „ValidateAddress“, US- Adressen zu überprüfen. Wenn Sie die Verarbeitung US-amerikanischer Adressen deaktivieren, schlagen US-amerikanische Adressen fehl. Das bedeutet, dass diese mit einem „F“ im Ausgabefeld „Status“ zurückgegeben werden. Das Ausgabefeld „Status.Code“ enthält dann „DisabledCoder“. Wenn Sie die Verarbeitung US-amerikanischer Adressen nicht lizenziert haben, müssen Sie die Verarbeitung US-amerikanischer Adressen deaktivieren, damit Ihre Aufträge erfolgreich abgeschlossen werden, auch mit US-amerikanischen Adressen.</p> <p>Anmerkung: Sie müssen über eine gültige Lizenz für die Verarbeitung US-amerikanischer Adressen verfügen für die erfolgreiche Verarbeitung US-amerikanischer Adressen. Wenn Sie die Verarbeitung US-amerikanischer Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben.</p> <p>N Nein, US-amerikanische Adressen nicht verarbeiten.</p> <p>Y Ja, US-amerikanische Adressen verarbeiten. Standardeinstellung.</p>
Database.US	<p>Gibt an, welche Datenbank zum Überprüfen von US-amerikanischen Adressen verwendet werden soll. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich „US-Datenbankressourcen“ definiert wurden.</p>
PerformLOT	<p>Bei der Enhanced Line of Travel (eLOT)-Verarbeitung wird Ihren Adressen ein Line of Travel-Sequenzcode zugewiesen. Beachten Sie, dass Adressen nicht in eLOT-Reihenfolge sortiert werden, aber ihnen wird ein Line of Travel-Sequenzcode zugewiesen, anhand dessen Sie Adressen in eLOT-Reihenfolge sortieren können.</p> <p>Zur Ausführung der eLOT-Verarbeitung muss die eLOT-Datenbank installiert sein.</p> <p>N Nein, keine Line of Travel-Verarbeitung ausführen. Standardeinstellung.</p> <p>Y Ja, Line of Travel-Verarbeitung ausführen.</p> <p>Eine Liste der durch diese Option zurückgegebenen Ausgabefelder finden Sie unter Ausgabe aus Enhanced Line of Travel (eLot) auf Seite 477.</p>

optionName	Beschreibung
PerformRDI	<p>Die Residential Delivery Indicator (RDI™)-Verarbeitung überprüft, ob eine Adresse eine Wohnanschrift (keine Geschäftsanschrift) ist. Zur Ausführung der RDI™-Verarbeitung muss die RDI™-Datenbank installiert sein.</p> <p>Wenn Sie sowohl DPV® als auch RDI™ als Verarbeitungsmethode aktivieren, werden RDI™-Informationen nur zurückgegeben, wenn die Adresse ein gültiger Zustellort ist. Wenn DPV® die Adresse nicht überprüft, werden keine RDI™-Daten zurückgegeben.</p> <p>N Nein, Residential Delivery Indicator-Verarbeitung nicht ausführen. Standardeinstellung.</p> <p>Y Ja, Residential Delivery Indicator-Verarbeitung ausführen.</p>
PerformESM	<p>Beim erweiterten Straßenvergleich (Enhanced Street Matching, ESM) wird eine zusätzliche Vergleichslogik angewendet, um falsch geschriebene oder komplexe Straßennamen zu korrigieren und eine Übereinstimmung zu erzielen. Mit ESM ist die Überprüfung einer größeren Anzahl von Adressen, möglich, aber die Leistung ist geringer. Sie können ESM nicht ausführen, wenn ASM aktiviert ist.</p> <p>N Nein, erweiterten Straßenvergleich nicht ausführen. Standardeinstellung.</p> <p>Y Ja, erweiterten Straßenvergleich ausführen.</p>
PerformASM	<p>Beim Vergleich aller Straßen (All Street Matching, ASM) wird sowohl die ESM-Verarbeitung als auch eine zusätzliche Vergleichslogik angewendet, um fehlerhafte Straßennamen zu korrigieren und eine Übereinstimmung zu erzielen. Beim Vergleich aller Straßen ist es effektiv, wenn der erste Buchstabe der Straße falsch ist. ASM bietet die beste Adressenüberprüfung, aber verringert die Leistung.</p> <p>N Nein, Vergleich aller Straßen nicht ausführen.</p> <p>Y Ja, Vergleich aller Straßen ausführen. Standardeinstellung.</p>

optionName	Beschreibung
PerformDPV	<p>Bei der Delivery Point Validation (DPV[®]) wird das Vorhandensein einer bestimmten Adresse überprüft, im Gegensatz zur Überprüfung, ob sich eine bestimmte Adresse in einem Bereich gültiger Adressen befindet. Bei der CMRA-Verarbeitung wird überprüft, ob eine Adresse für ein von einem privaten Unternehmen gemietetes Postfach steht, das als Commercial Mail Receiving Agent (CMRA) bezeichnet wird.</p> <p>Zur Ausführung der DPV- und CMRA-Verarbeitung muss die DPV-Datenbank installiert sein. Die DPV-Datenbank enthält sowohl DPV- als auch CMRA-Daten.</p> <p>N Nein, keine Delivery Point Validation oder CMRA-Verarbeitung ausführen. Standardeinstellung.</p> <p>Y Ja, Delivery Point Validation oder CMRA-Verarbeitung ausführen.</p> <p>Eine Liste der durch diese Option zurückgegebenen Ausgabefelder finden Sie unter DPV- und CMRA-Ausgabe auf Seite 480.</p>
PerformLACSLink	<p>Mit dem USPS[®] Locatable Address Conversion System (LACS) können Sie Adressen korrigieren, die infolge der Umwandlung einer Landstraßenadresse in eine straßenähnliche Adresse, der Neummerierung eines Postfachs oder der Änderung einer straßenähnlichen Adresse geändert wurden. Die LACS^{Link}-Verarbeitung ist bei Aktivierung für Adressen gedacht, die nicht überprüft werden konnten oder für solche, die überprüft und für die LACS^{Link}-Umwandlung markiert wurden.</p> <p>Zur Ausführung der LACS^{Link}-Verarbeitung muss die LACS^{Link}-Datenbank installiert sein.</p> <p>N Nein, LACS^{Link}-Umwandlung nicht ausführen. Standardeinstellung.</p> <p>Y Ja, LACS^{Link}-Umwandlung ausführen.</p> <p>Eine Liste der durch diese Option zurückgegebenen Ausgabefelder finden Sie unter LACSLink-Ausgabe auf Seite 478.</p>
PerformEWS	<p>Das Frühwarnsystem (EWS) nutzt die USPS[®]-EWS-Datei, um Adressen zu überprüfen, die sich nicht in der ZIP + 4[®]-Datenbank befinden.</p> <p>Zur Ausführung der EWS-Verarbeitung muss die EWS-Datenbank installiert sein.</p> <p>Wenn eine Eingabeadresse mit einer Adresse in der EWS-Datei übereinstimmt, werden die folgenden Ergebnisindikatoren auf Datensatzebene zurückgegeben:</p> <ul style="list-style-type: none"> • Status="F" • Status.Code="EWSFailure" • Status.Description="Adresse in CMRA-Tabelle gefunden" <p>N Nein, keine EWS-Verarbeitung ausführen. Standardeinstellung.</p> <p>Y Ja, EWS-Verarbeitung ausführen.</p>

optionName	Beschreibung
------------	--------------

ExtractFirm	
-------------	--

optionName

Beschreibung

Gibt an, ob der Firmenname von „AddressLine1“ bis „AddressLine4“ extrahiert und in das Ausgabefeld „FirmName“ eingefügt werden soll. Diese Option funktioniert in solchen Fällen, wo das Feld „FirmName“ des Eingabedatensatzes leer und mehr als eine Adresszeile vorhanden ist.

Y Ja, Firmenname extrahieren.

N Nein, Firmenname nicht extrahieren. Standardeinstellung.

Um Firmennamen in Adresszeilen zu identifizieren, werden die Adresszeilen nach Schlüsselwörtern und Mustern durchsucht, die angeben, welche Felder Adresszeilen und welche „FirmName“-Zeilen sind. Da dieser Vorgang auf Basis von Mustern erfolgt, werden Felder möglicherweise falsch identifiziert. Mit den folgenden Tipps wird eine optimale Firmenextraktion sichergestellt:

- Fügen Sie die primären Adresselemente möglichst in „AddressLine1“, die sekundären Adresselemente in „AddressLine2“, die Wohnanlage in „AddressLine3“ und den Firmennamen in „AddressLine4“ ein. Wenn die Adresse keinen Wohnanlagencode hat, fügen Sie den Firmennamen in „AddressLine3“ ein und lassen Sie „AddressLine4“ leer. Beispiel:

AddressLine1: 4200 Parliament Place

AddressLine2: Suite 600

AddressLine3: Pitney Bowes

AddressLine4: <leer>

- Wenn Sie nur zwei Adresszeilen definieren, wird „AddressLine2“ meistens der sekundären Adresse zugewiesen. Wenn Sie die Chance erhöhen möchten, dass „AddressLine2“ als Firmenname behandelt wird, fügen Sie den Firmennamen in „AddressLine3“ ein und lassen Sie „AddressLine2“ leer.
- Nummern in einem Firmennamen (z. B. „1“ in „1 Stop Software“) erhöhen die Wahrscheinlichkeit, dass das Feld als eine Adresszeile behandelt wird.

Hier einige Beispiele von Firmennamen-Extraktionen:

- In diesem Beispiel würde „AddressLine2“ in das Ausgabefeld „FirmName“ extrahiert werden.

FirmName: <leer>

AddressLine1: 4200 Parliament Place Suite 600

AddressLine2: International Goose Feathers inc.

- In diesem Beispiel würde „AddressLine3“ in das Ausgabefeld „FirmName“ extrahiert werden.

FirmName: <leer>

AddressLine1: 4200 Parliament Place

AddressLine2: Suite 600

AddressLine3: Pitney Bowes

- In diesem Beispiel würde „AddressLine3“ in das Ausgabefeld „AdditionalInputData“ eingefügt werden. Der Firmenname würde nicht extrahiert werden, da das Eingabefeld „FirmName“ nicht leer ist.

FirmName: International Goose Feathers Inc.

optionName

Beschreibung

AddressLine1: 4200 Parliament Place

AddressLine2: Suite 600

AddressLine3: Pitney Bowes

- In diesem Beispiel würde kein Firmenname extrahiert, weil es nur eine nicht leere Adresszeile gibt, die immer als primäres Adresselement behandelt wird.

FirmName: <leer>

AddressLine1: 4200 Parliament Place Suite 600

- In diesem Beispiel würde „AddressLine2“ als sekundäres Adresselement behandelt werden, weil die Nummernangabe „1“ dazu führt, dass dieses Feld als sekundäres Adresselement behandelt wird.

FirmName: <leer>

AddressLine1: 4200 Parliament Place Suite 600

AddressLine2: 1 Stop Software

ExtractUrb

Gibt an, ob der Wohnanlagenname von „AddressLine1“ bis „AddressLine4“ extrahiert und in das Ausgabefeld „USUrbanName“ eingefügt werden soll. Diese Option funktioniert in solchen Fällen, wo das Feld „USUrbanName“ des Eingabedatensatzes leer und mehr als eine Adresszeile vorhanden ist.

Y Ja, Wohnanlagenname extrahieren.

N Nein, Wohnanlagenname nicht extrahieren. Standardeinstellung.

Um Wohnanlagennamen zu identifizieren, werden die Adresszeilen nach Schlüsselwörtern und Mustern durchsucht, die angeben, welche Felder Adresszeilen und welche Wohnanlagenname-Zeilen sind. Da dieser Vorgang auf Basis von Mustern erfolgt, werden Felder möglicherweise falsch identifiziert. Um eine optimale Wohnanlagen-Extraktion sicherzustellen, fügen Sie die primären Adresselemente möglichst in „AddressLine1“, die sekundären Adresselemente in „AddressLine2“, die Wohnanlage in „AddressLine3“ und den Firmennamen in „AddressLine4“ ein. Beispiel:

AddressLine1: A1 Calle A

AddressLine2:

AddressLine3: URB Alamar

AddressLine4: Pitney Bowes

optionName	Beschreibung
PerformSuiteLink	<p>Gibt an, ob die Suite^{Link™}-Verarbeitung ausgeführt werden soll.</p> <p>Suite^{Link} korrigiert sekundäre Adressinformationen für US-amerikanische Firmenadressen, deren sekundäre Adressinformationen nicht überprüft werden konnten. Wenn die Suite^{Link}-Verarbeitung aktiviert ist, wird der Firmenname mit einer Datenbank mit bekannten Firmennamen und deren sekundären Adressinformationen verglichen.</p> <p>Beispiel:</p> <p>Firmenname: Pitney Bowes Adresszeile 1: 4200 Parliament Place Adresszeile 2: STE 1 Postleitzahl: 20706</p> <p>In diesem Fall würde die Suite^{Link}-Verarbeitung die Suitenummer in die korrekte Suitenummer ändern:</p> <p>Firmenname: Pitney Bowes Adresszeile 1: 4200 Parliament Place Adresszeile 2: STE 600 Postleitzahl: 20706-1844</p> <p>Zur Ausführung der Suite^{Link™}-Verarbeitung muss die Suite^{Link™}-Datenbank installiert sein.</p> <p>Diese Option akzeptiert einen der folgenden Werte:</p> <p>N Nein, Suite^{Link™}-Verarbeitung nicht verwenden. Standardeinstellung. Y Ja, Suite^{Link™}-Verarbeitung verwenden.</p> <p>Eine Liste der durch diese Option zurückgegebenen Felder finden Sie unter SuiteLink-Ausgabe auf Seite 482.</p>

optionName	Beschreibung
OutputPreferredAlias	<p>Gibt an, ob in der Ausgabe ein bevorzugter Alias der Straße verwendet werden soll.</p> <p>Straßennamen-Aliasse in den USA sind alternative Namen für bestimmte Straßenabschnitte. Es gibt vier Typen von Straßennamen-Aliassen:</p> <ul style="list-style-type: none"> • Bevorzugt: Ein bevorzugter Alias ist der Straßename, der lokal bevorzugt wird. In der Regel gilt er nur für einen bestimmten Bereich von Adressen in der Straße. • Abgekürzt: Ein Abkürzungsalias ist eine Variation des Straßennamens, der verwendet werden kann, wenn die Länge von AddressLine1 31 Zeichen überschreitet. Beispielsweise könnte der Straßename 1234 BERKSHIRE VALLEY RD APT 312A abgekürzt werden zu 1234 BERKSHIRE VLLY RD APT 312A. • Geändert: Es fand eine offizielle Änderung des Straßennamens statt und der Alias zeigt den neuen Namen. Wenn SHINGLE BROOK RD z. B. in CANNING DR geändert wurde, dann wäre CANNING DR ein geänderter Aliastyp. • Anderer: Der Straßenalias besteht aus anderen Namen für die Straße oder aus häufigen Abkürzungen für die Straße. <p>Die Version des Straßennamens, die kein Alias ist, wird als Basisstraßenname bezeichnet.</p> <p>Wenn der bevorzugte Alias in der Eingabe verwendet wird, ist dieser in der Ausgabe der Straßename, unabhängig davon, ob Sie diese Option aktivieren.</p> <p>Diese ist eine von drei Optionen, die steuern, wie „ValidateAddress“ Straßennamen-Aliasse verarbeitet. Die anderen beiden Optionen sind „OutputStreetNameAlias“ und „OutputAbbreviatedAlias“.</p> <p>In den meisten Fällen ist es so, wenn Sie sowohl „OutputPreferredAlias“ als auch „OutputAbbreviatedAlias“ auswählen und „ValidateAddress“ sowohl einen bevorzugten als auch einen abgekürzten Alias in der Postdatenbank findet, wird in der Ausgabe der abgekürzte Alias verwendet. Die Ausnahme zu dieser Regel ist, wenn der eingegebene Straßename ein bevorzugter Alias ist. In diesem Fall wird der bevorzugte Alias in der Ausgabe verwendet.</p> <p>Y Ja, Verarbeitung von bevorzugtem Alias ausführen.</p> <p>N Nein, Verarbeitung von bevorzugtem Alias nicht ausführen. Standardeinstellung.</p> <p>Anmerkung: Wenn die Eingabeadresse einen Straßennamen-Alias vom Typ „geändert“ enthält, zeigt die Ausgabeadresse immer den Basisstraßennamen an, unabhängig von den von Ihnen angegebenen Optionen.</p>

optionName	Beschreibung
OutputAbbreviatedAlias	<p>Gibt an, ob in der Ausgabe ein abgekürzter Alias der Straße verwendet werden soll, wenn die Ausgabezeile länger als 31 Zeichen ist.</p> <p>Diese ist eine von drei Optionen, die steuern, wie „ValidateAddress“ Straßennamen-Aliasse verarbeitet. Die anderen beiden Optionen sind „OutputStreetNameAlias“ und „OutputPreferredAlias“.</p> <p>Anmerkung: Wenn in der Eingabe ein bevorzugter Alias angegeben wird, ist der Straßennamen-Alias in der Ausgabe immer der bevorzugte Alias, selbst wenn Sie die Option „Verarbeiten von abgekürztem Alias-Straßennamen“ aktivieren.</p> <p>Y Ja, Verarbeitung von abgekürztem Alias ausführen.</p> <p>N Nein, Verarbeitung von abgekürztem Alias nicht ausführen. Standardeinstellung.</p> <p>Anmerkung: Wenn die Eingabe eine Basisstraßenname-Alias vom Typ „geändert“ enthält, zeigt die Ausgabe immer den Basisstraßennamen an, unabhängig von den von Ihnen angegebenen Optionen.</p>
DPVDetermineNoStat	<p>Bestimmt den „nicht statistisch“-Status einer Adresse. Ein Adresse wird als „nicht statistisch“ bezeichnet, wenn sie vorhanden ist, aber keine Post empfangen kann, und deshalb bei einer Beförderungsrouten nicht zur Zustellungsstatistik gezählt wird (daher auch der Begriff „nicht statistisch“). Dazu gehören beispielsweise im Bau befindliche Gebäude oder solche, die der Postzusteller als unwahrscheinliche Zustellziele identifiziert hat.</p> <p>N Nein, „nicht statistisch“-Status nicht bestimmen. Standardeinstellung.</p> <p>Y Ja, „nicht statistisch“-Status bestimmen.</p> <p>Anmerkung: Sie müssen die DPV-Verarbeitung aktivieren, um diese Option verwenden zu können.</p> <p>Das Ergebnis wird im Feld „DPVNoStat“ zurückgegeben. Weitere Informationen finden Sie unter LACSLink-Ausgabe auf Seite 478.</p>
DPVDetermineVacancy	<p>Bestimmt, ob der Standort mindestens 90 Tage lang nicht bewohnt war.</p> <p>N Nein, Leerstand nicht bestimmen. Standardeinstellung.</p> <p>Y Ja, Leerstand bestimmen.</p> <p>Anmerkung: Sie müssen die DPV-Verarbeitung aktivieren, um diese Option verwenden zu können.</p> <p>Das Ergebnis wird im Feld „DPVVacant“ zurückgegeben. Weitere Informationen finden Sie unter LACSLink-Ausgabe auf Seite 478.</p>

optionName	Beschreibung
ReturnVerimove	<p>Gibt VeriMove-Detaildaten in der Ausgabe zurück.</p> <p>N Nein, VeriMove-Detaildaten nicht zurückgeben. Standardeinstellung.</p> <p>Y Ja, VeriMove-Detaildaten zurückgeben.</p>
SuppressZplusPhantomCarrierR777	<p>Gibt an, ob Adressen mit der Beförderungsrouten R777 unterdrückt werden sollen. Diese Adressen sind Phantomrouten und nicht für die Straßenzustellung geeignet. Da diesen Adressen ein ZIP + 4®-Code von USPS® zugewiesen ist, markiert sie „Validate Address“ als zustellbare Adressen. Wählen Sie diese Option aus, wenn Adressen mit der Beförderungsrouten R777 nicht als zustellbar markiert werden sollen. Dies führt zu folgenden Aktionen:</p> <ul style="list-style-type: none"> • Es wird kein ZIP + 4-Code zugewiesen. • Die Adresse wird nicht auf der USPS Form 3553 (CASS-Zusammenfassungsbericht) berücksichtigt. • Die DPV-Fußnote „R7“ wird zurückgegeben. <p>N Nein, Adressen mit der Beförderungsrouten R777 nicht unterdrücken.</p> <p>Y Ja, Adressen mit der Beförderungsrouten R777 unterdrücken.</p>
StreetMatchingStrictness	<p>Gibt den Algorithmus an, mit dem bestimmt werden soll, ob eine Eingabeadresse mit einer Adresse in der postalischen Datenbank übereinstimmt. Zur Auswahl stehen:</p> <p>E Der Straßensname in der Eingabe muss exakt mit der Datenbank übereinstimmen.</p> <p>T Der Vergleichsalgorithmus ist „eng“.</p> <p>M Der Vergleichsalgorithmus ist „mittel“ (Standard).</p> <p>L Der Vergleichsalgorithmus ist „lose“.</p>
FirmMatchingStrictness	<p>Gibt den Algorithmus an, mit dem bestimmt werden soll, ob eine Eingabeadresse mit einer Adresse in der postalischen Datenbank übereinstimmt. Zur Auswahl stehen:</p> <p>E Der Firmenname in der Eingabe muss exakt mit der Datenbank übereinstimmen.</p> <p>T Der Vergleichsalgorithmus ist „eng“.</p> <p>M Der Vergleichsalgorithmus ist „mittel“ (Standard).</p> <p>L Der Vergleichsalgorithmus ist „lose“.</p>

optionName	Beschreibung
DirectionalMatchingStrictness	<p>Gibt den Algorithmus an, mit dem bestimmt werden soll, ob eine Eingabeadresse mit einer Adresse in der postalischen Datenbank übereinstimmt. Zur Auswahl stehen:</p> <p>E Die Richtungsangaben in der Eingabe, z. B. „N“ in 123 N Main St. müssen exakt mit der Datenbank übereinstimmen.</p> <p>T Der Vergleichsalgorithmus ist „eng“.</p> <p>M Der Vergleichsalgorithmus ist „mittel“. Standardeinstellung.</p> <p>L Der Vergleichsalgorithmus ist „lose“.</p>
DPVSuccessfulStatusCondition	<p>Wählen Sie die Vergleichsbedingung aus, unter der ein DPV-Ergebnis NICHT dazu führt, dass ein Datensatz nicht übereinstimmt.</p> <p>F Vollständige Übereinstimmung</p> <p>P Partielle Übereinstimmung</p> <p>A Immer. Standardeinstellung.</p> <p>Anmerkung: Sie müssen die DPV-Verarbeitung aktivieren, um diese Option verwenden zu können.</p>
FailOnCMRAMatch	<p>Commercial Mail Receiving Agency (CMRA)-Übereinstimmungen als Fehler behandeln?</p> <p>N Nein, CMRA-Übereinstimmungen nicht als Fehler behandeln. Standardeinstellung.</p> <p>Y Ja, CMRA-Übereinstimmungen als Fehler behandeln.</p> <p>Anmerkung: Sie müssen die DPV-Verarbeitung aktivieren, um diese Option verwenden zu können.</p>
StandardAddressPMBLine	<p>Gibt an, wo private Postfachdaten (PMB) eingefügt werden.</p> <p>N Keine. PMB-Daten nicht in standardmäßige Adressausgabe einschließen (Standard).</p> <p>1 PMB-Daten in „AddressLine1“ einfügen. Wenn Sie „1“ angeben, müssen Sie als „StandardAddressFormat“ entweder „C“ oder „D“ festlegen.</p> <p>2 PMB-Daten in „AddressLine2“ einfügen.</p>

optionName	Beschreibung
PreferredCity	<p>Gibt an, ob der bevorzugte Ortsname der letzten Zeile gespeichert werden soll.</p> <p>Z Speichert den bevorzugten Ortsnamen der letzten Zeile aus der USPS ZIP+4-Datei (Ortsname überschreiben).</p> <p>Anmerkung: Wenn Sie diese Option auswählen, generiert „Validate Address“ eine CASS-zertifizierte Konfiguration und den USPS 3553-Bericht.</p> <p>C Speichert den von USPS bevorzugten Ortsnamen aus der „USPS Ort/Bundesstaat“-Datei.</p> <p>Anmerkung: Wenn Sie diese Option auswählen, generiert „Validate Address“ keine CASS-zertifizierte Konfiguration und keinen USPS 3553-Bericht.</p> <p>P Speichert den primären Ortsnamen aus der „USPS Ort/Bundesstaat“-Datei.</p> <p>Anmerkung: Wenn Sie diese Option auswählen, generiert „Validate Address“ keine CASS-zertifizierte Konfiguration und keinen USPS 3553-Bericht.</p>

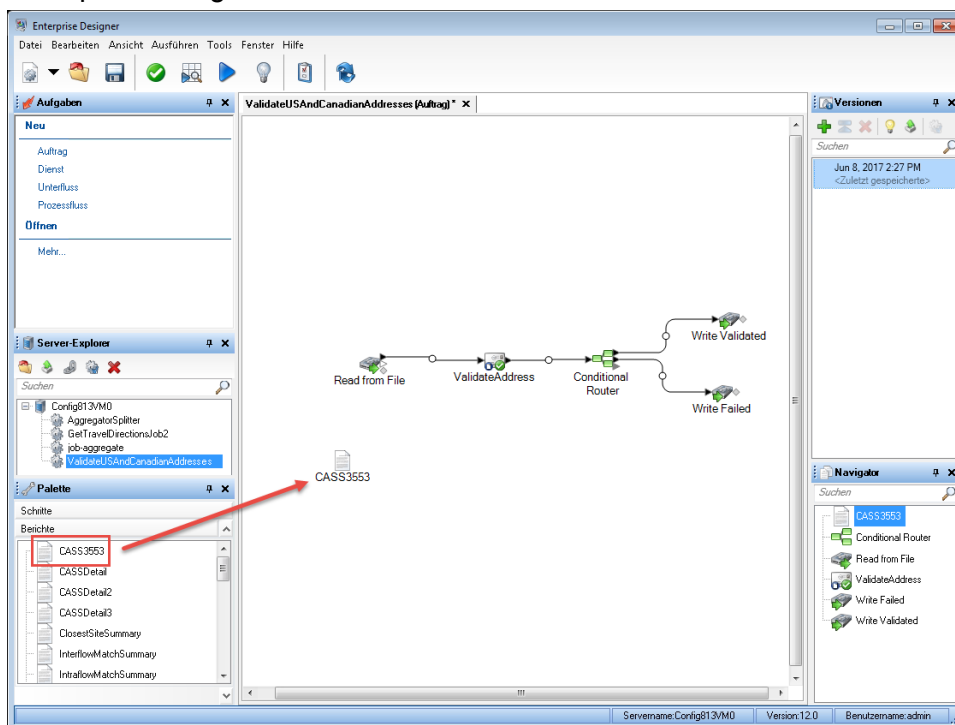
CASS-zertifizierte Verarbeitung

Die CASS-zertifizierte™ Verarbeitung generiert ebenfalls den USPS CASS Detailed Report, der einige der gleichen Informationen wie das Formular 3553 enthält. Er bietet jedoch einen besseren Einblick in DPV-, LACS- und SuiteLink-Statistiken. Der USPS CASS Detailed Report ist nicht für Postrabatte erforderlich und muss nicht mit Ihrer Postsendung übermittelt werden.

1. „Validate Address“ muss im CASS-zertifizierten™ Modus sein. Wenn **(Nicht CASS-zertifiziert)** oben im Fenster angezeigt wird, klicken Sie auf die Schaltfläche **CASS aktivieren**. Das Kontrollkästchen **CASS-Regeln erzwingen** wird angezeigt.
2. Klicken Sie auf **CASS 3553 konfigurieren**. Das Dialogfeld **CASS-Berichtsfelder** wird angezeigt.
3. Geben Sie den **Listenprozessor**-Unternehmensnamen, den **Listenname oder die ID-Nr.** und die **Anzahl der Listen** ein, die für diesen Auftrag verwendet werden.
4. Geben Sie **Mailer-Name, Adresse und Ort, Bundesland/Kanton, PLZ** ein.
5. Klicken Sie auf **OK**.

Die Listeninformationen sind in Abschnitt B und die Mailer-Informationen in Abschnitt D des erstellten USPS® CASS Form 3553 aufgeführt.

6. Ziehen Sie im Enterprise Designer den **CASS3553**-Bericht aus dem Berichtsbereich auf die



Arbeitsfläche.

7. Doppelklicken Sie auf der Arbeitsfläche auf das Symbol **CASS3553**.
8. Aktivieren Sie auf der Registerkarte **Schritte** das Kontrollkästchen **Validate Address**. Beachten Sie, dass wenn Sie den „Validate Address“-Schritt umbenannt haben, Sie das Kästchen mit dem entsprechenden Namen aktivieren müssen.
9. Wählen Sie auf der Registerkarte **Parameter** das Format für den Bericht aus. Sie können den Bericht im PDF-, HTML- oder Reintextformat erstellen.
10. Klicken Sie auf **OK**.
11. Wiederholen Sie die Schritte 6–10 für **CASSDetail**, wenn Sie den CASS Detailed Report erstellen möchten.

Optionen für kanadische Adressen

optionName	Beschreibung
PerformCanadianProcessing	<p>Gibt an, ob kanadische Adressen verarbeitet werden sollen. Wenn Sie die Verarbeitung kanadischer Adressen aktivieren, versucht „ValidateAddress“, kanadische Adressen zu überprüfen. Wenn Sie die Verarbeitung kanadischer Adressen deaktivieren, schlagen kanadische Adressen fehl. Das bedeutet, dass diese mit einem „F“ im Ausgabefeld „Status“ zurückgegeben werden. Das Ausgabefeld „Status.Code“ enthält dann „DisabledCoder“. Wenn Sie die Verarbeitung kanadischer Adressen nicht lizenziert haben, müssen Sie die Verarbeitung kanadischer Adressen deaktivieren, damit Ihre Aufträge erfolgreich abgeschlossen werden, auch wenn sie kanadische Adressen enthalten.</p> <p>Anmerkung: Für eine erfolgreiche Verarbeitung kanadischer Adressen müssen Sie über eine gültige Lizenz für die Verarbeitung kanadischer Adressen verfügen. Wenn Sie die Verarbeitung kanadischer Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben.</p> <p>N Nein, kanadische Adressen nicht verarbeiten.</p> <p>Y Ja, kanadische Adressen verarbeiten (Standard).</p>
Database.Canada	<p>Gibt an, welche Datenbank Sie zum Überprüfen kanadischer Adressen verwenden möchten. Wählen Sie für die Angabe einer Datenbank zur Überprüfung kanadischer Adressen eine Datenbank aus der Dropdown-Liste Datenbank aus. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich CAN-Datenbankressourcen definiert wurden.</p>

optionName	Beschreibung
CanFrenchFormat	<p>Gibt an, wie die Sprache (Englisch oder Französisch) zum Formatieren der Adresse und Richtungsangabe bestimmt werden soll. Das folgende Beispiel zeigt eine Adresse, die in Englisch und Französisch formatiert ist:</p> <p>Englisch: 123 Main St W Französisch: 123 Rue Main O</p> <p>Der Parameter steuert die Formatierung der Adresse. Er hat auch Einfluss auf die Schreibweise der Richtungsangabe, jedoch nicht auf die Schreibweise des Suffix.</p> <p>C Bestimmen Sie die Sprache mithilfe des beim Vergleichsprozess zurückgegebenen Straßensuffix. Das beim Vergleichsprozess zurückgegebene Straßensuffix, das intern von „ValidateAddress“ während der Verarbeitung verwendet wird, kann sich von dem in der Eingabeadresse unterscheiden. Mehrdeutige Datensätze werden wie die Eingabe formatiert. Standardeinstellung. Alle Adressen in Quebec werden in Französisch formatiert.</p> <p>S Sie können mithilfe der Datenbank für Kanada die Sprache bestimmen. Die Datenbank für Kanada enthält Daten von der Canada Post Corporation (CPC). Alle Adressen in Quebec werden in Französisch formatiert.</p> <p>T Im Eingabefeld „CanLanguage“ können Sie die Sprache bestimmen. Wenn dieses Feld einen nicht leeren Wert enthält, wird die Adresse in Französisch formatiert.</p>
CanEnglishApartmentLabel	<p>Gibt bei englischen Adressen die Standard-Apartmentbeschriftung an, die in der Ausgabe verwendet werden soll, wenn die Eingabeadresse keine Apartmentbeschriftung enthält. Diese Einstellung wird ignoriert, wenn Sie <code>CanStandardAddressFormat=F</code> angeben.</p> <p>Apt Verwenden Sie „Apt“ als Beschriftung. Standardeinstellung.</p> <p>Apartment Verwenden Sie „Apartment“ als Beschriftung.</p> <p>Suite Verwenden Sie „Suite“ als Beschriftung.</p> <p>Unit Verwenden Sie „Einheit“ als Beschriftung.</p>

optionName	Beschreibung
CanFrenchApartmentLabel	<p>Gibt bei französischen Adressen die Standard-Apartmentbeschriftung an, die in der Ausgabe verwendet werden soll, wenn die Eingabeadresse keine Apartmentbeschriftung enthält. Diese Einstellung wird ignoriert, wenn Sie <code>CanStandardAddressFormat=F</code> angeben.</p> <p>App Verwenden Sie „App“ als Beschriftung. Standardeinstellung.</p> <p>Appartement Use "Appartement" as the label.</p> <p>Bureau Use "Bureau" as the label.</p> <p>Suite Use "Suite" as the label.</p> <p>Unite Use "Unite" as the label.</p>
ForceCorrectionLVR	<p>Ändert die Angaben zur Anschrift/Suite so, dass sie mit dem Large Volume Receiver(LVR)- oder dem Single-Single-Datensatz (findet Verwendung, wenn für Postleitzahl/Straßenname/Straßentyp nur ein Datensatz vorhanden ist) übereinstimmen.</p> <p>N Angaben zur Anschrift/Suite nicht so ändern, dass sie mit dem LVR- oder Single-Single-Datensatz übereinstimmen. Der LVR-Datensatz wird als gültiger, aber nicht korrigierbarer Datensatz (VN) gekennzeichnet. Der Single-Single-Datensatz wird, sofern möglich, korrigiert oder als nicht korrigierbarer Datensatz verarbeitet.</p> <p>Y Angaben zur Anschrift und/oder Suite so ändern, dass sie mit dem LVR- oder Single-Single-Datensatz übereinstimmen.</p> <p>Anmerkung: Wenn Sie dieses Kästchen aktivieren, wird die Adressgenauigkeitsanweisung nicht gedruckt, da dies keine von SERP anerkannte Einstellung ist.</p>

optionName	Beschreibung
CanPreferHouseNum	<p>Wenn die Hausnummer und die Postleitzahl gültig sind, aber in Konflikt zueinander stehen, können Sie die Korrektur der Postleitzahl anhand der Hausnummer erzwingen, indem Sie <code>CanPreferHouseNum=Y</code> angeben. Wenn Sie diese Option nicht auswählen, wird die Hausnummer so geändert, dass sie mit der Postleitzahl übereinstimmt.</p> <p>N Hausnummer so ändern, dass sie mit der Postleitzahl übereinstimmt. Standardeinstellung.</p> <p>Y Postleitzahl so ändern, dass sie mit der Hausnummer übereinstimmt.</p>
CanOutputCityAlias	<p>Gibt an, ob der Ortsalias zurückgegeben werden soll, wenn der Alias in der Eingabeadresse enthalten ist. Diese Option wird deaktiviert, wenn Sie <code>CanOutputCityFormat=D</code> angeben.</p> <p>Y Ortsalias ausgeben, wenn der Ortsalias in der Eingabe enthalten ist. Standardeinstellung.</p> <p>N Ortsalias niemals ausgeben, selbst dann nicht, wenn er in der Eingabe enthalten ist.</p>
CanNonCivicFormat	<p>Gibt an, ob Schlüsselwörter für Nicht-Anschriften in der Ausgabe abgekürzt werden sollen. Beispiel: „Post Office Box“ vs. „PO BOX“.</p> <p>A Schlüsselwörter für Nicht-Anschriften abkürzen. Standardeinstellung.</p> <p>F Schlüsselwörter für Nicht-Anschriften nicht abkürzen. Das vollständige Schlüsselwort wird verwendet.</p>
EnableSERP	<p>Gibt an, ob SERP-Optionen verwendet werden sollen.</p> <p>Y SERP-Optionen aktivieren.</p> <p>N SERP-Optionen nicht aktivieren. Standardeinstellung.</p>

optionName	Beschreibung
CanStandardAddressFormat	<p>Gibt an, wo in der Ausgabeadresse sekundäre Adressinformationen platziert werden sollen. Sekundäre Adressinformationen beziehen sich auf Apartmentnummern, Suitenummern und ähnliche Bezeichnungen.</p> <ul style="list-style-type: none"> D Platziert Angaben zum Apartment an der im Feld unter der Option „StandardAddressFormat“ angegebenen Position. Standardeinstellung. B Angaben zum Apartment am Ende (zurück) des Feldes „AddressLine1“ platzieren. F Nur die Apartmentnummer (ohne Beschriftung) am Anfang des Feldes „AddressLine1“ platzieren. Beispiel: 400–123 Rue Main E Apartmentnummer und Beschriftung am Anfang des Feldes „AddressLine1“ platzieren. Beispiel: Apt 400 123 Rue Main S Angaben zum Apartment in separater Zeile platzieren. S Angaben zum Apartment an derselben Position wie die Eingabeadresse platzieren.
CanOutputCityFormat	<p>Gibt an, ob die lange, mittlere oder kurze Version des Ortsnamens verwendet werden soll, wenn der Ort einen langen Namen aufweist. Beispiel:</p> <p>Lang: BUFFALO HEAD PRAIRIE Mittel: BUFFALO-HEAD-PR Kurz: BUFFALO-HD-PR</p> <ul style="list-style-type: none"> D Von der Option <code>OutputShortCityName</code> angegebene Standardoption verwenden. Standardeinstellung. Wenn Sie <code>OutputShortCityName=V</code> angeben, wird der Ort so formatiert, als hätten Sie <code>L</code> für diese Option (siehe unten) und <code>Y</code> bei CanOutputCityAlias ausgewählt. S Kurzen Ortsnamen ausgeben. L Langen Ortsnamen ausgeben. M Mittellangen Ortsnamen ausgeben. I Das gleiche Ortsformat wie bei der Eingabeadresse verwenden. Die Ausgabe ist „L“, „M“ oder „S“.

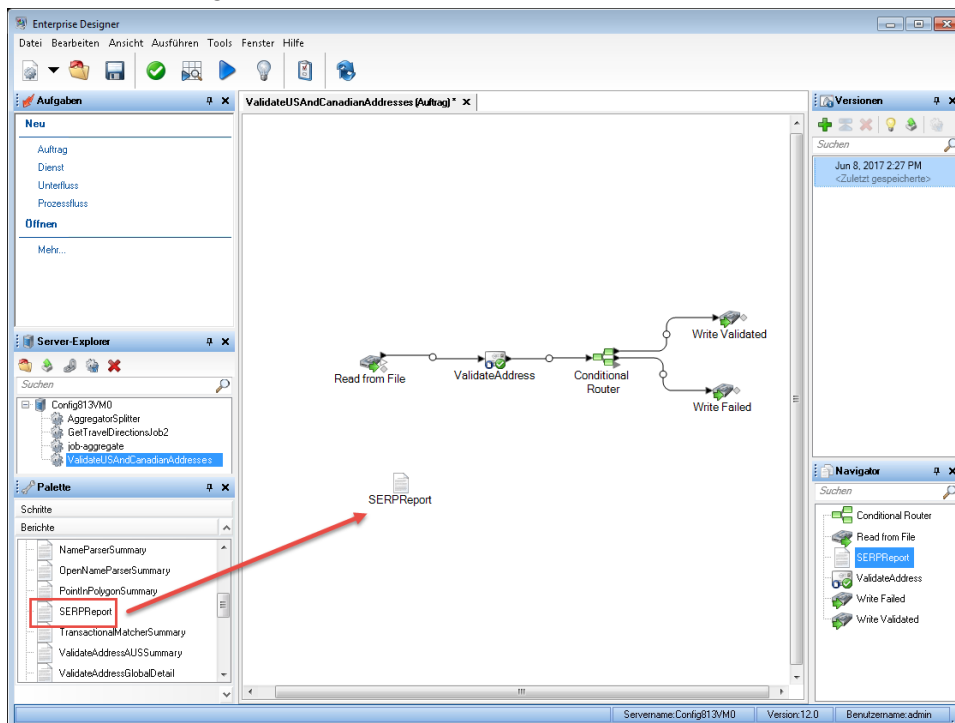
optionName	Beschreibung
CanRuralRouteFormat	<p>Gibt an, wo Informationen zur Zustellung auf ländlicher Route platziert werden sollen. Ein Beispiel für eine Adresse mit Informationen zur Zustellung auf ländlicher Route lautet wie folgt:</p> <p>36 GRANT RD RR 3 ANTIGONISH NS</p> <p>Bei dieser Adresse stellt „RR 3“ die Informationen zur Zustellung auf ländlicher Route dar.</p> <p>A Informationen zur Zustellung auf ländlicher Route in derselben Zeile wie die Adresse platzieren, hinter den Adressinformationen. Standardeinstellung. Beispiel:</p> <p>36 GRANT RD RR 3</p> <p>S Informationen zur Zustellung auf ländlicher Route in separater Zeile platzieren. Beispiel:</p> <p>36 GRANT RD RR 3</p>
CanDeliveryOfficeFormat	<p>Gibt an, wo Stationsinformationen platziert werden sollen. Ein Beispiel für eine Adresse mit Stationsinformationen lautet wie folgt:</p> <p>PO BOX 8625 STN A ST. JOHN'S NL</p> <p>I Stationsinformationen an derselben Position wie die Eingabeadresse platzieren. Standardeinstellung.</p> <p>A Stationsinformationen in derselben Zeile wie die Adresse platzieren, hinter den Adressinformationen. Beispiel:</p> <p>PO BOX 8625 STN A</p> <p>S Stationsinformationen in separater Zeile platzieren. Beispiel:</p> <p>PO BOX 8625 STN A</p>

optionName	Beschreibung
CanDualAddressLogic	<p>Gibt an, ob „ValidateAddress“ eine Straßenübereinstimmung oder eine Postfach-/Nicht-Anschrifs-Übereinstimmung zurückgeben soll, wenn die Adresse Anschrifs- und Nicht-Anschrifs-Informationen enthält. Zur Auswahl stehen:</p> <p>D Globale Option „DualAddressLogic“ verwenden. Standardeinstellung.</p> <p>P Mit Postfach oder anderen Nicht-Straßen-Daten abgleichen.</p> <p>S Mit Straße abgleichen.</p> <p>Hier ein Beispiel mit der folgenden Eingabeadresse:</p> <p>AddressLine1: 36 GRANT RD AddressLine2: RR 4 City: ANTIGONISH StateProvince: NS</p> <p>„ValidateAddress“ würde eines der folgenden Elemente zurückgeben:</p> <ul style="list-style-type: none"> • Wenn <code>CanDualAddressLogic</code> auf „S“ festgelegt ist, gibt „ValidateAddress“ folgende Elemente zurück: <p>AddressLine1: 36 GRANT RD AddressLine2: RR 3 City: ANTIGONISH StateProvince: NS PostalCode: B2G 2L1</p> • Wenn <code>CanDualAddressLogic</code> auf „P“ festgelegt ist, gibt „ValidateAddress“ folgende Elemente zurück: <p>AddressLine1: RR 4 City: ANTIGONISH StateProvince: NS PostalCode: B2G 2L2</p> <p>Die Adresdaten, die nicht zum Standardisieren der Adresse verwendet werden, werden im Feld AdditionalInputData zurückgegeben. Weitere Informationen finden Sie unter Ausgabedatenoptionen auf Seite 403.</p>

SERP-Verarbeitung

1. „Validate Address“ muss den Modus SERP-zertifiziert™ aufweisen. Klicken Sie auf die Schaltfläche **SERP-Einstellungen aktivieren**, wenn oben im Fenster **(Nicht SERP-zertifiziert)** angezeigt wird. Das Kästchen **SERP konfigurieren** wird angezeigt.
2. Klicken Sie auf **SERP konfigurieren**. Das Dialogfeld **SERP-Berichtsfelder** wird angezeigt.

3. Geben Sie Ihre **CPC-Händlernummer** ein.
4. Geben Sie **Name, Address** des Mailers und **Ort, Bundesland/Kanton, PLZ** ein.
5. Klicken Sie auf **OK**.
6. Ziehen Sie im Enterprise Designer den SERPReport aus dem Bereich „Berichte“ auf die



Arbeitsfläche.

7. Doppelklicken Sie auf das Symbol **SERPReport** auf der Arbeitsfläche.
8. Stellen Sie sicher, dass auf der Registerkarte **Schritte** das Kontrollkästchen **Validate Address** aktiviert ist. Beachten Sie, dass wenn Sie den „Validate Address“-Schritt umbenannt haben, Sie das Kästchen mit dem entsprechenden Namen aktivieren müssen.
9. Wählen Sie auf der Registerkarte **Parameter** das Format für den Bericht aus. Sie können den Bericht im PDF-, HTML- oder Reintextformat erstellen. Das PDF-Format ist das Standardformat.
10. Klicken Sie auf **OK**.

Ermitteln von SERP-Rückgabecodes

SERP-Rückgabecodes geben die Qualität der Eingabeadresse gemäß der Bestimmung durch die Richtlinien des Software Evaluation and Recognition Program der Canada Post an.

Geben Sie zur Ermittlung von SERP-Rückgabecodes den Parameter „OutputRecordType=P“ an. Weitere Informationen zu „OutputRecordType“ finden Sie unter [Ausgabedatenoptionen](#) auf Seite 403.

SERP-Rückgabecodes werden im folgenden Ausgabefeld bereitgestellt.

Tabelle 94: Ausgabe von SERP-Rückgabecodes

columnName	Beschreibung
CanadianSERPCode	<p>Überprüfungs-/Korrektur-Rückgabecode (nur kanadische Adressen):</p> <p>V Die Eingabe ist gültig. Canada Post definiert eine „gültige“ Adresse als eine Adresse, die alle der folgenden Anforderungen erfüllt:</p> <p style="padding-left: 40px;">Anmerkung: Es gibt Ausnahmen. Weitere Informationen erhalten Sie von CPC.</p> <ul style="list-style-type: none"> • Die Adresse muss alle erforderlichen Komponenten gemäß den Postal Code Data Files von CPC enthalten. • Die Adresse muss für nur eine Adresse eine genaue Übereinstimmung für alle Komponenten in den Postal Code Data Files von CPC aufweisen. • Adresskomponenten müssen in einer Form vorliegen, die eine Erkennung ohne Mehrdeutigkeit ermöglicht. Bestimmte Komponenten können „Qualifikatoren“ erfordern, um sie zu identifizieren. Eine „Route Service“-Adresse erfordert die Schlüsselwörter „Rural Route“ oder „RR“ für eine Unterscheidung von einer „Suburban Service“- oder „SS“-Adresse mit der gleichen Nummer. <p>I Die Eingabe ist ungültig. Eine „ungültige“ Adresse ist eine Adresse, die keine CPC-Anforderungen an eine gültige Adresse (siehe oben) erfüllt. Beispiele dafür beinhalten Adresskomponenten, die fehlen bzw. ungültig oder inkonsistent sind.</p> <p>C Die Eingabe ist korrigierbar. Eine „korrigierbare“ Adresse ist eine Adresse, die für eine Übereinstimmung mit einer einzigen Adresse korrigiert werden kann.</p> <p>N Die Eingabe ist nicht korrigierbar. Eine „nicht korrigierbare“ Adresse ist eine Adresse, die auf unterschiedliche Arten korrigiert werden könnte, wodurch „ValidateAddress“ keine einzelne korrekte Version finden kann.</p> <p>F Die Eingabeadresse ist fremd (außerhalb von Kanada).</p>

Optionen für internationale Adressen

Adressen außerhalb der USA und Kanada werden als „internationale“ Adressen bezeichnet. Die folgenden Optionen steuern die Verarbeitung internationaler Adressen:

optionName	Beschreibung
PerformInternationalProcessing	<p>Gibt an, ob internationale Adressen (Adressen außerhalb der USA und Kanada) verarbeitet werden sollen. Wenn Sie die Verarbeitung internationaler Adressen aktivieren, versucht „ValidateAddress“ internationale Adressen zu überprüfen. Wenn Sie die Verarbeitung internationaler Adressen deaktivieren, schlagen internationale Adressen fehl. Das bedeutet, dass diese mit einem „F“ im Ausgabefeld „Status“ zurückgegeben werden. Das Ausgabefeld „Status.Code“ enthält dann „DisabledCoder“. Wenn Sie die Verarbeitung internationaler Adressen nicht lizenziert haben, müssen Sie die Verarbeitung internationaler Adressen deaktivieren, damit Ihre Aufträge erfolgreich abgeschlossen werden, auch wenn sie internationale Adressen enthalten.</p> <p>Anmerkung: Sie müssen eine gültige Lizenz für die Verarbeitung internationaler Adressen haben, um internationale Adressen erfolgreich verarbeiten zu können. Wenn Sie die Verarbeitung internationaler Adressen aktivieren, dieses Feature jedoch nicht lizenziert haben oder Ihre Lizenz abgelaufen ist, wird ein Fehler ausgegeben.</p> <p>N Nein, internationale Adressen nicht verarbeiten.</p> <p>Y Ja, internationale Adressen verarbeiten (Standard).</p>
Database.International	<p>Gibt an, welche Datenbank Sie zum Überprüfen internationaler Adressen verwenden möchten. Wählen Sie für die Angabe einer Datenbank zur Überprüfung internationaler Adressen eine Datenbank aus der Dropdown-Liste Datenbank aus. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich INTL-Datenbankressourcen definiert wurden.</p>

optionName	Beschreibung
InternationalCityStreetSearching	<p>„ValidateAddress“ schafft ein ausgewogenes Verhältnis zwischen hoher Adressvergleichsgenauigkeit und guter Leistung. Wenn Sie anstelle der Vergleichsgenauigkeit eine schnellere Leistung bevorzugen, verwenden Sie die Option „InternationalCityStreetSearching“, um die Verarbeitungsgeschwindigkeit zu erhöhen. Wenn Sie diese Auswahl treffen, geht etwas der Genauigkeit verloren. Mit dieser Option wird nur die Leistung für Adressen außerhalb der USA und Kanada gesteuert. Diese Einstellung betrifft einen kleinen Prozentsatz von Datensätzen, größtenteils Adressen in Großbritannien. Es gibt keine Leistungssteuerung bei der Verarbeitung von US-amerikanischen und kanadischen Adressen.</p> <p>Wenn Sie „GetCandidateAddresses“ verwenden, unterscheiden sich die von „GetCandidateAddresses“ zurückgegebenen Kandidatenadressen möglicherweise von den Mehrfachübereinstimmungen, die von „ValidateAddress“ zurückgegeben werden, wenn Sie die Option zur Leistungsoptimierung für internationale Adressen auf einen anderen Wert als „100“ setzen.</p> <p>Geben Sie einen Wert zwischen 0 und 100 an, um die Leistung zu steuern. Eine Einstellung von „100“ maximiert die Genauigkeit, während eine Einstellung von „0“ die Geschwindigkeit maximiert. Der Standardwert ist 100.</p>
AddressLineSearchOnFail	<p>Bei Aktivierung dieser Option kann „ValidateAddress“ die „AddressLine“-Eingabefelder nach dem Ort, dem Bundesland/-staat/Kanton, der Postleitzahl und dem Land durchsuchen, wenn die Adresse mithilfe der Werte in den Eingabefeldern „City“, „StateProvince“ und „PostalCode“ nicht verglichen werden kann.</p> <p>Überlegen Sie, diese Option zu aktivieren, wenn die „AddressLine“-Felder Ihrer Eingabeadressen Angaben zu Ort, Bundesland/-staat/Kanton und Postleitzahl enthalten.</p> <p>Ziehen Sie die Deaktivierung dieser Option in Betracht, wenn bei Ihren Eingabeadressen die Felder „City“, „State/Province“ und „PostalCode“ verwendet werden. Wenn Sie diese Option aktivieren und diese Felder verwendet werden, besteht eine größere Möglichkeit, dass „ValidateAddress“ die Werte in diesen Feldern (z. B. einen falsch geschriebenen Ortsnamen) nicht korrigieren kann.</p> <p>N Nein, „AddressLine“-Felder nicht durchsuchen.</p> <p>Y Ja, „AddressLine“-Felder durchsuchen. StandardEinstellung.</p>

Ausgabe

Die Ausgabe von „Validate Address“ enthält unterschiedliche Informationen je nach ausgewählten Ausgabekategorien.

Standardmäßige Adressausgabe

Die standardmäßige Adressausgabe besteht aus vier Adresszeilen, die dem entsprechen, wie die Adresse auf einem Adressticket erscheint. Ort, Bundesland/Kanton, Postleitzahl und weitere Daten sind ebenfalls in der standardmäßigen Adressausgabe enthalten. Die standardmäßige Adressausgabe wird für überprüfte Adressen zurückgegeben, wenn Sie `OutputRecordType=A` einstellen. Standardmäßige Adressfelder werden immer für Adressen zurückgegeben, die nicht überprüft werden konnten. Für nicht überprüfte Adressen enthalten die standardmäßigen Adressenausgabefelder die Adresse so, wie sie in der Eingabe vorgekommen ist („Durchlauf“-Daten). Wenn Sie Adressen gemäß Postbehördenstandards bei fehlgeschlagener Überprüfung standardisieren möchten, geben Sie `OutputFormattedOnFail=Y` in Ihrer Abfrage an.

Tabelle 95: Standardmäßige Adressausgabe

columnName	Beschreibung
AdditionalInputData	Nicht bei der Adressenüberprüfung verwendete Eingabedaten. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.
AddressLine1	Wenn die Adresse überprüft wurde, die erste Zeile der überprüften und standardisierten Adresse. Wenn die Adresse nicht überprüft werden konnte, die erste Zeile der Eingabeadresse ohne Änderungen.
AddressLine2	Wenn die Adresse überprüft wurde, die zweite Zeile der überprüften und standardisierten Adresse. Wenn die Adresse nicht überprüft werden konnte, die zweite Zeile der Eingabeadresse ohne Änderungen.
AddressLine3	Wenn die Adresse überprüft wurde, die dritte Zeile der überprüften und standardisierten Adresse. Wenn die Adresse nicht überprüft werden konnte, die dritte Zeile der Eingabeadresse ohne Änderungen.
AddressLine4	Wenn die Adresse überprüft wurde, die vierte Zeile der überprüften und standardisierten Adresse. Wenn die Adresse nicht überprüft werden konnte, die vierte Zeile der Eingabeadresse ohne Änderungen.

columnName	Beschreibung
AddressLine5	Nur für britische Adressen. Wenn die Adresse überprüft wurde, die fünfte Zeile der überprüften und standardisierten Adresse. Wenn die Adresse nicht überprüft werden konnte, die fünfte Zeile der Eingabeadresse ohne Änderungen.
City	Der überprüfte Ortsname.
Country	Das Land in dem Format, das Sie unter OutputCountryFormat ausgewählt haben: <ul style="list-style-type: none"> • ISO-Code • UPU-Code • Englisch • Französisch • Deutsch • Spanisch
DepartmentName	Nur für britische Adressen; eine Unterabteilung einer Firma. Beispiel: Technikabteilung.
FirmName	Der überprüfte Firmen- oder Unternehmensname.
PostalCode	Der überprüfte ZIP-Code™ oder die überprüfte Postleitzahl.
PostalCode.AddOn	Der vierstellige Zusatzteil des ZIP-Codes™. Im ZIP-Code™ 60655-1844 lautet der vierstellige Zusatzteil 1844. (Nur US-amerikanische Adressen.)
PostalCode.Base	Der fünfstelligen ZIP-Code™; zum Beispiel: 20706 (nur US-amerikanische Adressen).
StateProvince	Die überprüfte Abkürzung für den Bundesstaat.
USUrbanName	Der überprüfte Wohnanlagenname. (Nur US-amerikanische Adressen.) Dies wird vor allem für puerto-ricanische Adressen verwendet.

Ausgabe von gearpsten Adresselementen

Ausgabeadressen werden im Format für gearpste Adressen formatiert, wenn Sie `OutputRecordType=E` einstellen. Wenn Sie Daten, die im Format für gearpste Adressen formatiert sind, bei einer fehlgeschlagenen Überprüfung zurückgeben möchten (also eine normalisierte Adresse), geben Sie `OutputFormattedOnFail=Y` an.

Anmerkung: Wenn immer gearpste Eingabedaten ungeachtet einer erfolgreichen Überprüfung zurückgegeben werden sollen, geben Sie `OutputRecordType=I` an. Weitere Informationen finden Sie unter [Gearpste Eingabe](#) auf Seite 451.

Tabelle 96: Ausgabe von gearpsten Adressen

columnName	Beschreibung
AdditionalInputData	Nicht von „Validate Address“ verwendete Eingabedaten. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.
AdditionalInputData.Base	Eingabedaten wurden von „ValidateAddress“ nicht in die standardisierte Adresse ausgegeben. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.
AdditionalInputData.Unmatched	Eingabedaten wurden an den Vergleich gesendet, jedoch nicht von „ValidateAddress“ zur Überprüfung verwendet. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.
ApartmentLabel	Apartmentbezeichnung (wie STE oder APT); z. B.: 123 E Main St APT 3
ApartmentLabel2	Sekundäre Apartmentbezeichnung, z. B.: 123 E Main St APT 3, 4th Floor Anmerkung: In diesem Release ist dieses Feld immer leer.
ApartmentNumber	Wohnungsnummer. Beispiel: 123 E Main St APT 3

columnName	Beschreibung
ApartmentNumber2	<p>Sekundäre Apartmentnummer. Beispiel: 123 E Main St APT 3, 4th Floor</p> <p>Anmerkung: In diesem Release ist dieses Feld immer leer.</p>
CanadianDeliveryInstallationAreaName	Zustellungsinstallationsname (nur kanadische Adressen)
CanadianDeliveryInstallationQualifierName	Zustellungsinstallationsqualifikator (nur kanadische Adressen)
CanadianDeliveryInstallationType	Zustellungsinstallationsstyp (nur kanadische Adressen)
City	Überprüfter Ortsname
Country	<p>Land. Das Format wurde durch Ihre Auswahl unter <code>OutputCountryFormat</code> bestimmt:</p> <ul style="list-style-type: none"> • ISO-Code • UPU-Code • Englisch • Französisch • Deutsch • Spanisch
DepartmentName	Nur für britische Adressen; eine Unterabteilung einer Firma. Beispiel: Technikabteilung.
FirmName	Der überprüfte Firmen- oder Unternehmensname
HouseNumber	Hausnummer, z. B.: 123 E Main St Apt 3
LeadingDirectional	Voranstehende Richtungsangabe, z. B.: 123 E Main St Apt 3

columnName	Beschreibung
POBox	Postfachnummer. Sollte es sich bei der Adresse um eine Landstraßenadresse handeln, wird hier die Postfachnummer der Landstraße angezeigt.
PostalCode	Überprüfte Postleitzahl. Bei US-amerikanischen Adressen ist dies der PLZ-Code.
PrivateMailbox	Indikator für privates Postfach.
PrivateMailbox.Type	Der Typ des privaten Postfachs. Mögliche Werte sind: <ul style="list-style-type: none"> • Standard • Non-Standard <p>Anmerkung: Dies ersetzt „PrivateMailboxType“ (kein Punkt im Feldname). Ändern Sie Ihre API-Aufrufe entsprechend.</p>
RRHC	Indikator für Landstraße/Autobahn
StateProvince	Überprüfter Name des Bundeslandes oder Kantons
StreetName	Straßenname, z. B.: 123 E Main St Apt 3
StreetSuffix	Straßensuffix, z. B.: 123 E Main St Apt 3
TrailingDirectional	Nachstehende Richtungsangabe, z. B.: 123 Pennsylvania Ave NW
USUrbanName	USPS®-Wohnanlagenname. Nur puerto-ricanische Adressen.

Geparste Eingabe

Die Ausgabe kann die Eingabeadresse im geparsten Format enthalten. Dieser Ausgabetyyp wird „geparste Eingabe“ genannt. Geparste Eingabefelder enthalten die Adressdaten, die als Eingabe genutzt wurden, egal ob „Validate Address“ die Adresse überprüft hat. Eine geparste Eingabe unterscheidet sich von geparsten Adresselementen der Ausgabe dahingehend, dass geparste Adresselemente die überprüfte Adresse enthalten, wenn die Adresse überprüft werden konnte, und optional die Eingabeadresse, wenn die Adresse nicht überprüft werden konnte. Die geparste Eingabe enthält immer die Eingabeadresse ungeachtet davon, ob „Validate Address“ die Adresse überprüft hat.

Stellen Sie zum Einschließen von geparsten Eingabefeldern in der Ausgabe Folgendes ein:

```
OutputRecordType=I.
```

Tabelle 97: Geparste Eingabe

columnName	Beschreibung
ApartmentLabel.Input	Apartmentbezeichnung (wie STE oder APT); z. B.: 123 E Main St APT 3
ApartmentNumber.Input	Apartmentnummer, z. B.: 123 E Main St APT 3
CanadianDeliveryInstallationAreaName.Input	Zustellungsinstallationsname (nur kanadische Adressen)
CanadianDeliveryInstallationQualifierName.Input	Zustellungsinstallationsqualifikator (nur kanadische Adressen)
CanadianDeliveryInstallationType.Input	Zustellungsinstallationsstyp (nur kanadische Adressen)
City.Input	Überprüfter Ortsname

columnName	Beschreibung
Country.Input	Land. Das Format wurde durch Ihre Auswahl unter OutputCountryFormat bestimmt: <ul style="list-style-type: none"> • ISO-Code • UPU-Code • Englisch • Französisch • Deutsch • Spanisch
FirmName.Input	Der überprüfte Firmen- oder Unternehmensname
HouseNumber.Input	Hausnummer, z. B.: 123 E Main St Apt 3
LeadingDirectional.Input	Voranstehende Richtungsangabe, z. B.: 123 E Main St Apt 3
POBox.Input	Postfachnummer. Sollte es sich bei der Adresse um eine Landstraßenadresse handeln, wird hier die Postfachnummer der Landstraße angezeigt.
PostalCode.Input	Überprüfte Postleitzahl. Bei US-amerikanischen Adressen ist dies der ZIP-Code.
PrivateMailbox.Input	Indikator für privates Postfach
PrivateMailbox.Type.Input	Der Typ des privaten Postfachs. Mögliche Werte sind: <ul style="list-style-type: none"> • Standard • Nicht-Standard
RRHC.Input	Indikator für Landstraße/Autobahn
StateProvince.Input	Überprüfter Name des Bundeslandes oder Kantons

columnName	Beschreibung
StreetName.Input	Straßenname, z. B.: 123 E Main St Apt 3
StreetSuffix.Input	Straßensuffix, z. B.: 123 E Main St Apt 3
TrailingDirectional.Input	Nachstehende Richtungsangabe, z. B.: 123 Pennsylvania Ave NW
USUrbanName.Input	USPS®-Wohnanlagenname

Ausgabe von postalischen Daten

Wenn `OutputRecordType` den Wert P enthält, werden die folgenden Felder in der Ausgabe zurückgegeben.

Tabelle 98: Ausgabe von postalischen Daten

columnName	Beschreibung
CanadianSERPCode	Überprüfungs-/Korrektur-Rückgabecode (nur kanadische Adressen). Weitere Informationen finden Sie unter Ermitteln von SERP-Rückgabecodes auf Seite 442.
IntHexaviaCode	Nur für Adressen in Frankreich; ein numerischer Code, der die Straße repräsentiert. Informationen über Hexavia-Codes finden Sie unter www.laposte.fr .
IntINSEECode	Nur für Adressen in Frankreich; ein numerischer Code, der den Ort repräsentiert. Eine Liste der INSEE-Codes finden Sie unter www.insee.fr .
PostalBarCode	Der zweistellige Zustellortsteil des „Delivery Point Barcode“ (nur US-amerikanische Adressen). Weitere Informationen finden Sie unter Erstellen von „Delivery Point Barcodes“ auf Seite 413.

columnName	Beschreibung
USAltAddr	<p>Gibt an, ob eine alternative Adressenvergleichslogik verwendet wurde, und wenn ja, welche Logik verwendet wurde (US-amerikanische Adressen). Zur Auswahl stehen:</p> <p>null Kein alternatives Adressschema verwendet.</p> <p>D Alternative Zustellortslogik wurde verwendet.</p> <p>E Erweiterte alternative Vergleichslogik für Hochhäuser wurde verwendet.</p> <p>S Standardmäßige Kleinstadtlogik wurde verwendet.</p> <p>U Eindeutig ZIP-Code-Logik wurde verwendet.</p>
USBCCheckDigit	<p>Prüfstellenteil des 11-stelligen „Delivery Point Barcodes“ (US-amerikanische Adressen). Weitere Informationen finden Sie unter Erstellen von „Delivery Point Barcodes“ auf Seite 413.</p>
USCarrierRouteCode	<p>Beförderungsroutencode (nur US-amerikanische Adressen). Weitere Informationen finden Sie unter Ermitteln von Beförderungsroutencodes auf Seite 413.</p>
USCongressionalDistrict	<p>Congressional District (nur US-amerikanische Adressen). Weitere Informationen finden Sie unter Ermitteln von „Congressional Districts“ auf Seite 411.</p>
USCountyName	<p>Bezirksname (nur US-amerikanische Adressen). Weitere Informationen finden Sie unter Ermitteln von Bezirksnamen auf Seite 412.</p>
USFinanceNumber	<p>Die Finanznummer, in der die Adresse liegt (nur US-amerikanische Adressen). Die Finanznummer ist eine von USPS zugewiesene Nummer für einen Bereich, der mehrere ZIP-Codes abdeckt. Eine Adresse wird nur überprüft, wenn ihre Finanznummer mit der Finanznummer der Kandidatenadresse in der US-amerikanischen Datenbank übereinstimmt.</p>
USFIPSCountyNumber	<p>FIPS (Federal Information Processing Standards)-Bezirksnummer (nur US-amerikanische Adressen). Weitere Informationen finden Sie unter Ermitteln von FIPS-Bezirksnummern auf Seite 412.</p>

columnName	Beschreibung
USLACS	<p>Gibt an, ob die Adresse ein Kandidat für eine LACS^{Link}-Umwandlung ist (nur US-amerikanische Adressen). Zur Auswahl stehen:</p> <p>Y Ja, die Adresse ist ein Kandidat für die LACS^{Link}-Verarbeitung. Wenn LACS^{Link} aktiviert ist, wird versucht, die Adresse mithilfe der LACS^{Link}-Datenbank umzuwandeln. Bei erfolgreicher Umwandlung ist die Ausgabeadresse die neue Adresse, die aus der LACS^{Link}-Datenbank abgerufen wurde. Wenn die Umwandlung nicht erfolgreich ist, wird die Adresse nicht umgewandelt.</p> <p>N Nein, die Adresse ist kein Kandidat für die LACS^{Link}-Verarbeitung. Die LACS^{Link}-Verarbeitung könnte weiterhin versucht werden, wenn die LACS^{Link}-Verarbeitung angefordert wird, die LACS^{Link}-Datenbank installiert wird und eine der folgenden Bedingungen erfüllt ist:</p> <ul style="list-style-type: none"> • Die Adresse stimmt mit einer Landstraßenadresse überein, und das Feld „RecordType.Default“ gibt den Wert Y zurück. • Die Eingabeadresse konnte mit keiner Adresse in der US-amerikanischen Postdatenbank verglichen werden (Fehler aufgrund von Mehrfachübereinstimmungen sind keine LACS^{Link}-Kandidaten.)
USLastLineNumber	<p>Ein sechsstelliger alphanumerischer Wert, der ZIP-Codes gruppiert, die den gleichen primären Ort teilen. Adressen mit den folgenden zwei letzten Zeilen hätten beispielsweise die gleiche Nummer für die letzte Zeile:</p> <p>Chantilly VA 20151 Chantilly VA 20152</p>

Ergebnisindikatoren

Ergebnisindikatoren bieten Informationen über die Arten der Verarbeitung, die für eine Adresse ausgeführt werden. Es gibt zwei Typen von Ergebnisindikatoren:

Ergebnisindikatoren auf Datensatzebene

Ergebnisindikatoren auf Datensatzebene bieten Daten zu den Ergebnissen der „Validate Address“-Verarbeitung für jeden Datensatz, wie z. B. den Erfolg oder Misserfolg eines Vergleichsversuchs, welcher Coder die Adresse verarbeitet hat und andere Details. Die folgende Tabelle enthält Ergebnisindikatoren auf Datensatzebene, die von „Validate Address“ zurückgegeben wurden.

Tabelle 99: Indikatoren auf Datensatzebene

columnName	Beschreibung
AddressFormat	<p>Der Typ der zurückgegebenen Adressdaten:</p> <p>F Französisches Format (z. B.: 123 Rue Main)</p> <p>E Englisch Format (z. B.: 123 Main St)</p>
Confidence	<p>Das Zuverlässigkeitsniveau, das der zurückgegebenen Adresse zugewiesen ist. Der Bereich reicht von 0 (null) bis 100. Null gibt an, dass ein Fehler vorliegt; 100 gibt an, dass das Zuverlässigkeitsniveau sehr hoch ist und die Übereinstimmungsergebnisse richtig sind. Für Mehrfachübereinstimmungen beträgt die Zuverlässigkeitsebene 0. Informationen zur Berechnung dieser Zahl finden Sie unter Einführung in den „Validate Address“-Zuverlässigkeitsalgorithmus auf Seite 557.</p>
CouldNotValidate	<p>Wenn keine Übereinstimmung gefunden wird, die Adresskomponente, die nicht überprüft werden konnte:</p> <ul style="list-style-type: none"> • ApartmentNumber • HouseNumber • StreetName • PostalCode • City • Directional • StreetSuffix • Firma • POBoxNumber • RuralRoute <p>Anmerkung: In einer durch Kommata getrennten Liste können mehrere Komponenten zurückgegeben werden.</p>

columnName	Beschreibung
CountryLevel	<p>Die Kategorie der verfügbaren Adressenübereinstimmung. Dies ist für US-amerikanische und kanadische Adressen standardisiert. Zur Auswahl stehen:</p> <p>A Die Adresse liegt in einem Land, für das hoch detaillierte postalische Daten verfügbar sind. Für Adressen auf dieser Übereinstimmungsebene können die folgenden Adresselemente überprüft, korrigiert und hinzugefügt werden, falls sie in der Eingabe fehlen:</p> <ul style="list-style-type: none">• Postleitzahl• Ortsname• Bundesland/Kanton und Bezirk• Straßenanschriftselemente• Ländername <p>B Die Adresse liegt in einem Land, für das mäßig detaillierte postalische Daten verfügbar sind. Für Adressen auf dieser Übereinstimmungsebene können die folgenden Adresselemente überprüft, korrigiert und hinzugefügt werden, falls sie in der Eingabe fehlen:</p> <ul style="list-style-type: none">• Postleitzahl• Ortsname• Bundesland/Kanton und Bezirk• Ländername <p>C Die Adresse liegt in einem Land, für das die postalischen Daten am detailärmsten sind. Für Adressen auf dieser Übereinstimmungsebene können die folgenden Aktionen durchgeführt werden:</p> <ul style="list-style-type: none">• Überprüfen und Korrigieren des Ländernamens (ein fehlender Ländername kann nicht bereitgestellt werden)• Überprüfen des Formats der Postleitzahl (eine fehlende Postleitzahl kann nicht bereitgestellt oder überprüft werden)

columnName	Beschreibung
MatchScore	<p>MatchScore gibt den Grad der Korrektheit der Ausgabeadresse an. Es unterscheidet sich wesentlich von der Zuverlässigkeit. Die Zuverlässigkeit gibt an, inwieweit die Eingabeadresse zum Erhalt einer Übereinstimmung geändert wurde. Die Bedeutung des MatchScore unterscheidet sich zwischen US-amerikanischen und nicht US-amerikanischen Adressen.</p> <p>Bei US-amerikanischen Adressen ist der MatchScore eine einstellige Punktzahl auf einer Skala von 0 bis 9. Sie gibt die Nähe der Straßennamenübereinstimmung an (nach eventuellen Transformationen durch „Validate Address“). Null gibt eine genaue Übereinstimmung und 9 die am wenigsten wahrscheinliche Übereinstimmung an. Wenn keine Übereinstimmung gefunden wurde, ist das Feld leer.</p> <p>Für nicht US-amerikanische und nicht kanadische Adressen ist der MatchScore eine fünfstellige Punktzahl mit einem maximalen Wert von 00999. Eine hohe Zahl gibt eine nähere Übereinstimmung an.</p> <p>Dieses Feld gilt nicht für kanadische Adressen.</p> <p>Beachten Sie, dass Sie MatchScores von US-amerikanischen Adressen nicht mit nicht US-amerikanischen Adressen Adressen. Ein MatchScore von 4 für eine US-amerikanische Adresse gibt z. B. nicht die gleiche Übereinstimmungsebene wie der Wert 00004 für eine nicht US-amerikanische Adresse an.</p> <p>Anmerkung: Die Komponenten „Validate Address“ und „Advanced Matching-Modul“ verwenden beide das Feld „MatchScore“. Der Wert des Feldes „MatchScore“ in der Ausgabe eines Datenflusses wird von dem letzten Schritt bestimmt, der den Wert verändert, bevor er an einen Ausgabeschritt gesendet wird. Wenn Sie einen Datenfluss mit den Komponenten „Validate Address“ und „Advanced Matching-Modul“ haben und das Ausgabefeld „MatchScore“ für jeden Schritt anzeigen möchten, kopieren Sie den MatchScore-Wert mithilfe eines Transformer-Schrittes in ein anderes Feld. „Validate Address“ erzeugt beispielsweise ein Ausgabefeld mit dem Namen „MatchScore“. Ein Transformer-Schritt kopiert anschließend das Feld „MatchScore“ aus „Validate Address“ in ein Feld mit dem Namen „AddressMatchScore“. Wenn ein Vergleichsschritt ausgeführt wird, füllt er das Feld „MatchScore“ mit dem Wert aus dem Vergleich aus und gibt den Wert für „AddressMatchScore“ aus „Validate Address“ weiter.</p>
MultimatchCount	<p>Wenn mehrere Übereinstimmungen gefunden wurden, gibt der Wert die Anzahl der Datensätze an, die mögliche Übereinstimmungen sind.</p>

columnName	Beschreibung
MultipleMatches	<p>Gibt an, welche Adresskomponente Mehrfachübereinstimmungen hatte, falls Mehrfachübereinstimmungen gefunden wurden:</p> <ul style="list-style-type: none"> • Firma • LeadingDirectional • PostalCode • StreetName • StreetSuffix • TrailingDirectional • Urbanization (Wohnsiedlung) <p>Anmerkung: In einer durch Kommata getrennten Liste können mehrere Komponenten zurückgegeben werden.</p>
ProbableCorrectness	<p>Relative Wahrscheinlichkeit der Richtigkeit der gefundenen allgemeinen Übereinstimmung:</p> <p>Leer Keine Übereinstimmung gefunden.</p> <p>0 Übereinstimmung ist höchstwahrscheinlich richtig.</p> <p>1-8 Mittlere Übereinstimmungsebene auf einer Gleitskala.</p> <p>9 Übereinstimmung ist höchstwahrscheinlich nicht richtig.</p> <p>Anmerkung: Dieser Werte stellen nur die Einschätzung des Programms hinsichtlich der „relativen Wahrscheinlichkeit der Richtigkeit“ dar. Es ist möglich, dass einige Übereinstimmungen mit einer Punktzahl von 0 nicht korrekt sind und dass Übereinstimmungen mit einer Punktzahl von 9 trotzdem richtig sind.</p>
ProcessedBy	<p>Der Adressen-Coder, der die Adresse verarbeitet hat:</p> <p>USA US-amerikanischer Adressen-Coder</p> <p>CAN Kanadischer Adressen-Coder</p> <p>INT Internationaler Adressen-Coder</p>
RecordType	<p>Typ des Adressendatensatzes gemäß der Definition von US-amerikanischen und kanadischen Postbehörden (nur für US-amerikanische und kanadische Adressen):</p> <ul style="list-style-type: none"> • FirmRecord • GeneralDelivery • HighRise • PostOfficeBox • RRHighwayContract • Normal

columnName	Beschreibung
RecordType.Default	Code, der die Übereinstimmung mit dem „Standard“ anzeigt: Y Die Adresse stimmt mit einem Standarddatensatz überein. null Die Adresse stimmt nicht mit einem Standarddatensatz überein.
Status	Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs. Bei Mehrfachübereinstimmungen enthält dieses Feld den Wert „F“ für alle möglichen Übereinstimmungen. null Erfolg F Fehler
Status.Code	Ursache für den Fehler, falls vorhanden. Bei Mehrfachübereinstimmungen ist der Wert für alle möglichen Übereinstimmungen „MultipleMatchesFound“. <ul style="list-style-type: none"> • DisabledCoder • InsufficientInputData • MultipleMatchesFound • UnableToValidate
Status.Description	Beschreibung des Problems, wenn eines vorliegt. Possible Multiple Addresses Found Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=MultipleMatchesFound. Address Not Found Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=UnableToValidate. PerformUSProcessing disabled Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=DisabledCoder. PerformCanadianProcessing disabled Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=DisabledCoder. PerformInternationalProcessing disabled Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=DisabledCoder.

Ergebnisindikatoren auf Feldebene

Ergebnisindikatoren auf Feldebene beschreiben, wie „Validate Address“ jedes Adresselement verarbeitet. Ergebnisindikatoren auf Feldebene werden im Begrenzungszeichen „Ergebnis“ zurückgegeben. Der Ergebnisindikator auf Feldebene für „HouseNumber“ ist beispielsweise in **HouseNumber.Result** enthalten.

Geben Sie zur Aktivierung von Ergebnisindikatoren auf Feldebene den Parameter `OutputFieldLevelReturnCodes=Y` an. Weitere Informationen finden Sie unter [Ausgabedatenoptionen](#) auf Seite 403.

Die folgende Tabelle enthält Ergebnisindikatoren auf Feldebene. Wenn ein bestimmtes Feld nicht für eine Adresse gilt, könnte der Ergebnisindikator leer sein.

Tabelle 100: Ergebnisindikatoren auf Feldebene

columnName	Beschreibung
AddressRecord.Result	<p>Diese Ergebniscodes gelten nur für internationale Adressen.</p> <ul style="list-style-type: none">M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld.S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.U Nicht verglichen.V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.

columnName	Beschreibung
ApartmentLabel.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische und kanadische Adressen.</p> <p>D Verworfen. Das bei der Eingabe bereitgestellte Feld wurde entfernt. Nur US-amerikanische und kanadische Adressen. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur US-amerikanische und kanadische Adressen.</p> <p>R Die Apartmentbezeichnung ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.</p> <p>U Nicht verglichen. Gilt nicht für kanadische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

columnName	Beschreibung
ApartmentNumber.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur kanadische Adressen.</p> <p>D Verworfen. Das bei der Eingabe bereitgestellte Feld wurde entfernt. Nur US-amerikanische Adressen. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. US-amerikanische Adressen, die eine EWS-Übereinstimmung sind, haben den Wert P. Nur US-amerikanische und kanadische Adressen.</p> <p>R Die Apartmentnummer ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische Adressen.</p> <p>U Nicht verglichen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

columnName	Beschreibung
City.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische und kanadische Adressen.</p> <p>F Bindestriche fehlen oder Interpunktionsfehler. Nur kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten.</p> <p>R Der Ort ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische Adressen.</p> <p>U Nicht verglichen. Gilt nicht für kanadische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>
Country.Result	<p>Diese Ergebniscodes gelten nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.</p> <p>U Nicht verglichen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

columnName	Beschreibung
FirmName.Result	<p>C Korrigiert. Nur US-amerikanische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur US-amerikanische und kanadische Adressen.</p> <p>U Nicht verglichen. Nur US-amerikanische und kanadische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert. Nur US-amerikanische Adressen.</p>
HouseNumber.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur kanadische Adressen.</p> <p>C Korrigiert. Nur kanadische Adressen.</p> <p>D Verworfen. Das bei der Eingabe bereitgestellte Feld wurde entfernt. Nur US-amerikanische Adressen. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>O Außerhalb des Bereichs. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.</p> <p>R Die Hausnummer ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur kanadische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>U Nicht verglichen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

columnName	Beschreibung
LeadingDirectional.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nicht leere Eingabe wurde in einen nicht leeren Wert korrigiert. Nur US-amerikanische Adressen.</p> <p>D Verworfen. Das bei der Eingabe bereitgestellte Feld wurde entfernt. Nur US-amerikanische Adressen. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.</p> <p>U Nicht verglichen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert. Gilt nicht für kanadische Adressen.</p>

columnName	Beschreibung
POBox.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur kanadische Adressen.</p> <p>C Korrigiert. Nur kanadische Adressen.</p> <p>D Verworfen. Das bei der Eingabe bereitgestellte Feld wurde entfernt. Nur US-amerikanische Adressen. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrfachübereinstimmungen. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.</p> <p>R Die Postfachnummer ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.</p> <p>U Nicht verglichen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

columnName	Beschreibung
PostalCode.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische und kanadische Adressen.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Gilt nicht für kanadische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Gilt nicht für US-amerikanische Adressen.</p> <p>R Die Postleitzahl ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>U Nicht verglichen. Wenn der Straßenname z. B. nicht mit der Postleitzahl übereinstimmt, enthalten „StreetName.Result“ und „PostalCode.Result“ den Wert U.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

columnName	Beschreibung
PostalCodeCity.Result	<p>Diese Ergebniscodes gelten nur für internationale Adressen.</p> <ul style="list-style-type: none"> M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. U Nicht verglichen. V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.
PostalCode.Source	<p>Diese Ergebniscodes gelten nur für US-amerikanische Adressen.</p> <ul style="list-style-type: none"> FinanceNumber Der ZIP-Code™ in der Eingabe wurde mithilfe von USPS®-Finanznummerngruppierungen überprüft. ZIPMOVE Der ZIP-Code™ in der Eingabeadresse wurde korrigiert, da USPS® die ZIP-Code™-Grenzen nachgezeichnet hat und die Adresse nun in einem anderen ZIP-Code™ liegt.
PostalCode.Type	<ul style="list-style-type: none"> P Der ZIP-Code™ enthält nur Postfachadressen. Nur US-amerikanische Adressen. U Der ZIP-Code™ ist ein eindeutiger ZIP-Code™, der einem bestimmten Unternehmen oder Standort zugewiesen ist. Nur US-amerikanische Adressen. M Der ZIP-Code™ gilt für Militäradressen. Nur US-amerikanische Adressen. null Der ZIP-Code™ ist ein standardmäßiger ZIP-Code™.

columnName	Beschreibung
RRHC.Result	<p>C Korrigiert. Nur kanadische Adressen.</p> <p>D Verworfen. Das bei der Eingabe bereitgestellte Feld wurde entfernt. Nur US-amerikanische Adressen. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.</p> <p>M Mehrfachübereinstimmungen. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.</p> <p>R Die Landstraße/Autobahn ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Nur US-amerikanische und kanadische Adressen.</p> <p>U Nicht verglichen. Nur US-amerikanische und kanadische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert. Nur US-amerikanische und kanadische Adressen.</p>
RRHC.Type	<p>Diese Ergebniscode gelten nur für US-amerikanische Adressen.</p> <p>HC Die Adresse ist eine Highway Contract-Adresse (Autobahn).</p> <p>RR Die Adresse ist eine Rural Route-Adresse (Landstraße).</p>

columnName	Beschreibung
StateProvince.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten.</p> <p>R Der Bundesstaat ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische Adressen.</p> <p>U Nicht verglichen. Gilt nicht für kanadische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>
Street.Result	<p>Diese Ergebniscodes gelten nur für internationale Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten.</p> <p>R Straße korrigiert. Hausnummer außerhalb des Bereichs. Gilt nur für französische, britische und japanische Datensätze.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.</p> <p>U Nicht verglichen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

columnName	Beschreibung
StreetName.AbbreviatedAlias.Result	<p>Gibt das Ergebnis der abgekürzten Aliasverarbeitung an. Zur Auswahl stehen:</p> <ul style="list-style-type: none">null Keine abgekürzte Aliasverarbeitung versucht.B Das Feld „StreetName“ enthält den Basisstraßennamen.L Die standardisierte Adressenlänge ist kleiner als 31 Zeichen, also enthält das Feld „StreetName“ den Basisnamen.N Kein abgekürzter Alias gefunden.Y Es wurde ein abgekürzter Alias für die Eingabeadresse gefunden. Das Feld „StreetName“ enthält den abgekürzten Alias.

columnName

Beschreibung

StreetName.Alias.Type

Dieser Ergebniscode gilt nur für US-amerikanische Adressen.

Anmerkung: In Vorgängerversionen war dieses Feld als „StreetName.AliasType“ ohne „.“ zwischen „Alias“ und „Type“ benannt. Dieser Name ist veraltet. Aktualisieren Sie Ihre Vorgänge, um den neuen Namen „StreetName.Alias.Type“ zu verwenden.

Abbreviated Der Alias ist eine Abkürzung des Straßennamens. HARTS-NM RD ist z. B. ein abgekürzter Alias für HARTSVILLE NEW MARLBORO RD.

Changed Es fand eine offizielle Änderung des Straßennamens statt und der Alias zeigt den neuen Namen. Wenn SHINGLE BROOK RD z. B. in CANNING DR geändert wurde, dann wäre CANNING DR ein geänderter Aliastyp.

Other Der Straßenalias besteht aus anderen Namen für die Straße oder aus häufigen Abkürzungen der Straße.

Preferred Der Straßenalias ist der lokal bevorzugte Alias. Zum Beispiel eine Straße mit dem Namen „South Shore Dr.“, da sie entlang des südlichen Ufers eines Sees und nicht südlich einer Gemeindeabgrenzungslinie verläuft. „South“ ist in diesem Fall also keine Präfixrichtung und darf nicht als „S“ abgekürzt werden. „South Shore Dr.“ wäre also der bevorzugte Alias.

StreetName.PreferredAlias.Result

Gibt das Ergebnis der bevorzugten Aliasverarbeitung an. Zur Auswahl stehen:

- null** Keine bevorzugte Aliasverarbeitung versucht.
- A** Die bevorzugte Aliasverarbeitung wurde nicht versucht, da die Eingabeadresse mit einem Alias übereinstimmt. Die bevorzugte Aliasverarbeitung wurde nur für Basisadressen versucht.
- N** Kein bevorzugter Alias gefunden.
- Y** Es wurde ein bevorzugter Alias für die Eingabeadresse gefunden. Das Feld „StreetName“ enthält den bevorzugten Alias.

columnName	Beschreibung
StreetName.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische und kanadische Adressen.</p> <p>D Verworfen. Das bei der Eingabe bereitgestellte Feld wurde entfernt. Nur US-amerikanische Adressen. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Gilt nicht für US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Nur US-amerikanische und kanadische Adressen.</p> <p>U Nicht verglichen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

columnName	Beschreibung
StreetSuffix.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische und kanadische Adressen.</p> <p>D Verworfen. Das bei der Eingabe bereitgestellte Feld wurde entfernt. Nur US-amerikanische und kanadische Adressen. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.</p> <p>U Nicht verglichen. Gilt nicht für US-amerikanische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

columnName	Beschreibung
TrailingDirectional.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische und kanadische Adressen.</p> <p>D Verworfen. Das bei der Eingabe bereitgestellte Feld wurde entfernt. Nur US-amerikanische und kanadische Adressen. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten auf Seite 483.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.</p> <p>U Nicht verglichen. Gilt nicht für kanadische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>
USUrbanName.Result	<p>Diese Ergebniscode gelten nur für US-amerikanische Adressen.</p> <p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt.</p> <p>C Korrigiert.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld.</p> <p>U Nicht verglichen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

Ausgabe aus Optionen

„Validate Address“ gibt zusätzliche Daten in Abhängigkeit von den ausgewählten Optionen zurück. Informationen zur Ausgabe, die durch jede Option generiert wird, finden Sie in den Optionen, die in den folgenden Abschnitten aufgeführt sind:

Ausgabe aus Enhanced Line of Travel (eLot)

Die eLot-Verarbeitung generiert die folgende Ausgabe.

columnName	Beschreibung
USLOTCode	<p>Lot-Sequenzcode und ein Indikator, der die USPS®-LOT-Sequenz kennzeichnet. Dieses Feld hat das Format „nnnnY“, wobei Folgendes gilt:</p> <p>nnnn Der vierstellige LOT-Code.</p> <p>Y Zur Auswahl stehen:</p> <ul style="list-style-type: none"> • A: Aufsteigende LOT-Sequenz • D: Absteigende LOT-Sequenz
USLOTHex	<p>Ein Hexadezimalwert, mit dem Sie Ihre Datei nur in aufsteigender Reihenfolge sortieren können. Die Hexadezimalwerte reichen aufsteigend von 0 bis FF und dann absteigend von FF bis 0.</p>
USLOTSequence	<p>Ein Zwei-Byte-Wert zur finalen Sortierung anstelle des DPC-Add-Ons. Er besteht aus einem Großbuchstaben gefolgt von einer Ziffer von 0 bis 9. Werte reichen von A0 (99 absteigend) bis J9 (00 absteigend) sowie K0 (00 aufsteigend) bis T9 (99 aufsteigend).</p>

LACS^{Link}-Ausgabe

columnName	Beschreibung
USLACS	<p>Gibt an, ob die Adresse ein Kandidat für eine LACS^{Link}-Umwandlung ist (nur US-amerikanische Adressen). Zur Auswahl stehen:</p> <p>Y Ja, die Adresse ist ein Kandidat für die LACS^{Link}-Verarbeitung. Wenn LACS^{Link} aktiviert ist, versucht „ValidateAddress“ die Adresse mithilfe der LACS^{Link}-Datenbank umzuwandeln. Bei erfolgreicher Umwandlung ist die Ausgabeadresse die neue Adresse, die aus der LACS^{Link}-Datenbank abgerufen wurde. Wenn die Umwandlung nicht erfolgreich ist, wird die Adresse nicht umgewandelt.</p> <p>N Nein, die Adresse ist kein Kandidat für die LACS^{Link}-Verarbeitung. Die LACS^{Link}-Verarbeitung könnte weiterhin versucht werden, wenn die LACS^{Link}-Verarbeitung angefordert wird, die LACS^{Link}-Datenbank installiert wird und eine der folgenden Bedingungen erfüllt ist:</p> <ul style="list-style-type: none"> • Die Adresse stimmt mit einer Landstraßenadresse überein, und das Feld „RecordType.Default“ gibt den Wert Y zurück. • Die Eingabeadresse konnte mit keiner Adresse in der US-amerikanischen Postdatenbank verglichen werden (Fehler aufgrund von Mehrfachübereinstimmungen sind keine LACS^{Link}-Kandidaten.)
USLACS.ReturnCode	<p>Gibt den Erfolg oder Misserfolg der LACS^{Link}-Verarbeitung an. (Nur US-amerikanische Adressen.)</p> <p>A LACS^{Link}-Verarbeitung erfolgreich. Datensatz durch LACS^{Link}-Verarbeitung verglichen.</p> <p>00 LACS^{Link}-Verarbeitung fehlgeschlagen. Kein übereinstimmender Datensatz während LACS^{Link}-Verarbeitung gefunden.</p> <p>09 Die LACS^{Link}-Verarbeitung hat die Eingabeadresse mit einer älteren Hochhausstandardadresse verglichen. Die Adresse wurde umgewandelt. Anstatt eine ungenaue Adresse bereitzustellen, stellt die LACS^{Link}-Verarbeitung keine neue Adresse bereit.</p> <p>14 LACS^{Link}-Verarbeitung fehlgeschlagen. Während der LACS^{Link}-Verarbeitung wurde eine Übereinstimmung gefunden, aber die Umwandlung fand aufgrund anderer USPS[®]-Bestimmungen nicht statt.</p> <p>92 LACS^{Link}-Verarbeitung erfolgreich. Datensatz durch LACS^{Link}-Verarbeitung verglichen. Bei der Eingabe verworfene Einheitsnummer.</p> <p>null LACS^{Link} hat den Datensatz nicht verarbeitet oder die LACS^{Link}-Verarbeitung wurde nicht versucht.</p>

RDI-Ausgabe

columnName	Beschreibung
RDI	Rückgabewerte, die den Adresstyp angeben. B Die Adresse ist eine Geschäftsanschrift. R Die Adresse ist eine Wohnanschrift. M Die Adresse ist eine Wohn- und Geschäftsanschrift. null Nicht aktiviert, da die Adresse nicht auf ZIP + 4 [®] -Ebene codiert wurde, oder RDI [™] wurde nicht durchgeführt.

DPV- und CMRA-Ausgabe

columnName	Beschreibung
DPV	<p>Gibt die Ergebnisse der „Delivery Point Validation“ (DPV)-Verarbeitung an.</p> <p>Y DPV-bestätigt. Der Adresse kann Post zugestellt werden.</p> <p>N Der Adresse kann keine Post zugestellt werden.</p> <p>S Die Gebäudenummer wurde überprüft, aber die Einheitsnummer konnte nicht bestätigt werden. Eine Gebäudenummer ist die primäre Adressnummer für ein Gebäude. Eine Einheitsnummer ist eine Nummer für eine bestimmte Postadresse innerhalb eines Gebäudes wie ein Apartment, eine Suite, eine Etage usw. Bei dieser Adresse ist 424 z. B. die Gebäudenummer und 12 die Einheitsnummer:</p> <p>424 Washington Blvd. Apt. 12 Oak Park IL 60302 USA</p> <p>D Die Gebäudenummer wurde überprüft, aber die Einheitsnummer fehlte in der Eingabe. Eine Gebäudenummer ist die primäre Adressnummer für ein Gebäude. Eine Einheitsnummer ist eine Nummer für eine bestimmte Postadresse innerhalb eines Gebäudes wie ein Apartment, eine Suite, eine Etage usw. Bei dieser Adresse ist 424 z. B. die Gebäudenummer und 12 die Einheitsnummer:</p> <p>424 Washington Blvd. Apt. 12 Oak Park IL 60302 USA</p> <p>M Die Adresse stimmt mit mehreren gültigen Zustellorten überein.</p> <p>U Die Adresse konnte nicht bestätigt werden, da die Adresse nicht auf ZIP + 4[®]-Ebene codiert wurde.</p> <p>V Die Adresse verursachte einen falsch-positiven Verstoß.</p>
CMRA	<p>Gibt an, ob die Adresse eine „Commercial Mail Receiving Agency“ (CMRA) ist.</p> <p>Y Ja, die Adresse ist eine CMRA.</p> <p>N Nein, die Adresse ist keine CMRA.</p> <p>U Nicht bestätigt.</p>

columnName	Beschreibung
DPVFootnote	<p>DPV-Fußnotencodes.</p> <p>AA Eingabeadresse wurde mit der ZIP + 4[®]-Datei verglichen.</p> <p>A1 Eingabeadresse wurde nicht mit der ZIP + 4[®]-Datei verglichen.</p> <p>BB Eingabeadresse wurde mit DPV (alle Komponenten) verglichen.</p> <p>CC Primäre Nummer der Eingabeadresse mit DPV verglichen, jedoch sekundäre Nummer nicht (vorhanden aber ungültig).</p> <p>F1 Eingabeadresse ist militärisch; DPV umgangen.</p> <p>G1 Eingabeadresse ist eine allgemeine Zustellung; DPV umgangen.</p> <p>M1 Primäre Nummer der Eingabeadresse fehlt.</p> <p>M3 Primäre Nummer der Eingabeadresse ist ungültig.</p> <p>N1 Die primäre Nummer der Eingabeadresse wurde mit DPV verglichen, die Hochhausanschrift enthält jedoch keine sekundäre Nummer.</p> <p>P1 In der Eingabeadresse fehlt die RR oder HC-Fachnummer.</p> <p>P3 In der Eingabeadresse fehlt die PO-, RR- oder HC-Fachnummer.</p> <p>RR Eingabeadresse mit CMRA verglichen.</p> <p>R1 Eingabeadresse mit CMRA verglichen, aber sekundäre Nummer ist nicht vorhanden.</p> <p>U1 Eingabeadresse enthält eine eindeutige PLZ; DPV umgangen.</p>
DPVVacant	<p>Gibt an, ob das Gebäude leer steht (seit 90 Tagen unbewohnt). Zur Auswahl stehen:</p> <p>Y Ja, das Gebäude steht leer.</p> <p>N Nein, das Gebäude steht nicht leer.</p> <p>null Die Option <code>DPVDetermineVacancy</code> wurde nicht aktiviert.</p>
DPVNoStat	<p>Gibt an, ob das Gebäude ein „nicht statistisches“ Gebäude ist und deshalb keine Post empfängt. Zur Auswahl stehen:</p> <p>Y Ja, das Gebäude ist ein „nicht statistisches“ Gebäude. Das heißt, dass das Gebäude keine Post empfängt.</p> <p>N Nein, das Gebäude ist kein „nicht statistisches“ Gebäude. Das heißt, dass das Gebäude Post empfängt.</p> <p>null Die Option <code>DPVDetermineNoStat</code> wurde nicht aktiviert.</p>

Suite^{Link}-Ausgabe

columnName	Beschreibung
SuiteLinkReturnCode	<p>Gibt an, ob „Validate Address“ die sekundären Adressinformationen korrigiert hat (nur US-amerikanische Adressen). Zur Auswahl stehen:</p> <p>A „Validate Address“ hat die sekundären Adressinformationen korrigiert.</p> <p>00 „Validate Address“ hat nicht die sekundären Adressinformationen korrigiert.</p> <p>null Suite^{Link} wurde nicht ausgeführt.</p> <p>XX Die Suite^{Link}-Verarbeitung hat einen Fehler erkannt. Es würde z. B. ein Fehler auftreten, wenn die Suite^{Link}-Datenbank abgelaufen ist.</p>
SuiteLinkMatchCode	<p>Stellt zusätzliche Informationen über den Suite^{Link}-Vergleichsversuch bereit. (Nur US-amerikanische Adressen)</p> <p>A „Validate Address“ hat die sekundären Adressinformationen korrigiert.</p> <p>B „Validate Address“ hat nicht die sekundären Adressinformationen korrigiert. Es ist kein zusätzliches Detail zum Vergleichsversuch verfügbar.</p> <p>C Die Wörter im Feld „FirmName“ sind alles Füllwörter. Füllwörter werden von USPS[®] definiert und ignoriert, wenn der Firmenname verglichen werden soll. Beispiele für Füllwörter sind „company“ und „corporation“. „Validate Address“ kann sekundäre Adressinformationen für Firmennamen, die vollständig aus Füllwörtern bestehen, nicht korrigieren. „Company and Corporation“ ist z. B. ein Firmenname vollständig aus Füllwörtern.</p> <p>D Die Adresse ist keine standardmäßige Hochhausanschrift. Suite^{Link}-Vergleiche werden nur für standardmäßige Hochhausanschriften durchgeführt. Ein Hochhausstandard wird standardmäßig verwendet, wenn die Adresse keine gültigen sekundären Informationen enthält (die Apartmentnummer oder der Apartmenttyp fehlt).</p> <p>E Die Suite^{Link}-Verarbeitung ist fehlgeschlagen, da die Suite^{Link}-Datenbank abgelaufen ist.</p> <p>null Suite^{Link} wurde nicht ausgeführt oder es gab einen Fehler.</p>

columnName	Beschreibung
SuiteLinkFidelity	<p>Gibt an, wie gut „Validate Address“ den Firmennamen mit den Firmennamen in der Suite^{Link}-Datenbank verglichen hat.</p> <p>1 Der Firmenname stimmt genau mit der Suite^{Link}-Datenbank überein.</p> <p>2 Gute Übereinstimmung. Alle Wörter im Firmennamen außer ein Wort stimmen mit dem Firmennamen in der Suite^{Link}-Datenbank überein.</p> <p>3 Schlechte Übereinstimmung. Mehr als ein Wort im Firmennamen stimmt nicht mit dem Firmennamen in der Suite^{Link}-Datenbank überein.</p> <p>null Suite^{Link} konnte den Firmennamen nicht vergleichen oder es wurde nicht ausgeführt, oder es gab einen Fehler.</p>

VeriMove-Ausgabe

columnName	Beschreibung
VeriMoveDataBlock	<p>Gibt an, ob „Validate Address“ ein 250-Byte-Feld mit Eingabedaten zurückgeben soll, die an VeriMove Express weitergegeben werden sollen. Das Feld enthält die Daten des Detailergebnisindikators, die von VeriMove benötigt werden. Weitere Informationen über den Inhalt dieses Feldes finden Sie im VeriMove-Benutzerhandbuch. Zur Auswahl stehen:</p> <p>Y Ja, das Feld „VeriMoveDataBlock“ zurückgeben.</p> <p>N Nein, das Feld „VeriMoveDataBlock“ nicht zurückgeben.</p>

Zusätzliche Eingabedaten

Einige Eingabedaten werden während der Adressstandardisierung ignoriert. Diese irrelevanten Daten (manchmal als „verworfenen Daten“ bezeichnet) werden im Feld „AdditionalInputData“ zurückgegeben. Beispiele für verworfene Daten:

- Zustellungsanweisungen (z. B. „an der Hintertür abgeben“)
- Telefonnummern (z. B. „01234 56789“)
- „Zu Händen von“ (z. B. „z. Hd. v. John Smith“)

Daten wie diese werden im Allgemeinen nicht in eine Adresse eingebettet. Wenn Sie dennoch enthalten sind, können die irrelevanten Daten meistens identifiziert und im Feld „AdditionalInputData“ zurückgegeben werden.

Anmerkung: Verworfenen Daten aus geteilten Indizienadressen werden nicht zurückgegeben. Eine geteilte Indizienadresse enthält eine primäre Adresse, die auf mehrere Adresszeilen

aufgeteilt ist. Wenn die primäre Adresse beispielsweise „1 Green River Valley Rd“ lautet, wäre Folgende eine geteilte Indizienversion dieser Adresse:

1 Green River
Valley Rd
01230

Wenn es mehr als einen verworfenen Datenteil in einer Adresse gibt, wird jeder Datenteil in US-Adressen durch ein Semikolon und ein Leerzeichen („;“) und in Adressen außerhalb der USA durch ein Leerzeichen getrennt. Die Reihenfolge der verworfenen Daten in „AdditionalInputData“ lautet wie folgt:

1. „Per Adresse“, Poststopp (nur US-amerikanische Adressen)
2. Weitere in Adresszeilen gefundene irrelevante Daten
3. Gesamte nicht verwendete Datenzeilen

Wenn dies beispielsweise die Eingabeadresse ist:

123 Main St C/O John Smith
Apt 5 an Hintertür abgeben
jsmith@beispiel.com
555-123-4567
05674

Würde „AdditionalInputData“ Folgendes enthalten:

C/O John Smith; Apt 5 An Hintertür abgeben; 555-123-4567; Jsmith@beispiel.com; 555-123-4567

„Per Adresse“-Daten

Bei US-amerikanischen Adresse wird „per Adresse“ in „AdditionalInputData“ zurückgegeben. Die folgenden Adressen enthalten Beispiele für „per Adresse“:

123 Main St C/O John Smith
Apt 5
05674

123 Main St
Apt 5 ATTN John Smith
05674

123 Main St Apt 5
MailStop 2
05674

Irrelevante Daten in einer eigenen Adresszeile

„Validate Address“ gibt irrelevante Daten für US-amerikanische und kanadische Adressen in einer eigenen Zeile zurück.

Bei US-amerikanischen Adressen verwendet „Validate Address“ die ersten beiden nicht leeren Adresszeilen, um eine Adressstandardisierung durchzuführen, außer wenn die Optionen für Firmennamen-Extraktion oder Wohnanlagen-Codeextraktion aktiviert sind (siehe

Adresszeilenverarbeitung für US-amerikanische Adressen auf Seite 403 für weitere Informationen). Daten in anderen Adresszeilen werden in „AdditionalInputData“ zurückgegeben. In der folgenden Adresse würde „John Smith“ in „AdditionalInputData“ zurückgegeben werden, da es in der dritten nicht leeren Adresszeile steht und „Validate Address“ nur die ersten beiden nicht leeren Adresszeilen für US-amerikanische Adressen verwendet.

123 Main St
Apt 5
John Smith
05674

Wenn eine der ersten beiden nicht leeren Adresszeilen irrelevante Daten enthält, werden diese Daten in „AdditionalInputData“ zurückgegeben. In den folgenden Adressen würde „John Smith“ in „AdditionalAddressData“ zurückgegeben werden.

123 Main St
John Smith
05674

John Smith
123 Main St
05674

In der folgenden Adresse würde sowohl „John Smith“ als auch „Apt 5“ in „AdditionalInputData“ zurückgegeben werden. „John Smith“ würde zurückgegeben werden, da es sich um irrelevante Daten in einer der ersten beiden Adresszeilen handelt, und „Apt 5“ würde zurückgegeben werden, da US-amerikanische Adressdaten in den ersten beiden leeren Adresszeilen stehen müssen.

John Smith
123 Main St
Apt 5
05674

Irrelevante Daten innerhalb einer Adresszeile

Irrelevante Daten innerhalb einer Adresszeile werden in „AdditionalInputData“ zurückgegeben. In den folgenden Adressen würde „John Smith“ in „AdditionalInputData“ zurückgegeben werden.

123 Main St John Smith
05674

123 Main St Apt 5 John Smith
05674

123 Main St John Smith
Apt 5
05674

123 Main St
Apt 5 John Smith
05674

Bei US-amerikanischen Adressen werden irrelevante Daten am Ende der Adresszeile in „AdditionalInputData“ zurückgegeben. Irrelevante Daten, die nicht am Ende der Adresszeile stehen, werden nicht für US-amerikanische Adressen zurückgegeben. In den folgenden Adressen wird beispielsweise „John Smith“ nicht zurückgegeben.

John Smith 123 Main St
05674

123 Main John Smith St
05674

Die Spalte „AdditionalInputData“ enthält in manchen Fällen den ursprünglichen Straßennamen oder das Suffix, wenn der Straßename oder das Suffix am Ende einer Zeile stand und geändert wurde, um eine Übereinstimmung zu erzielen. Zum Beispiel diese Adresse:

Pitney Bowes
4200 Parliament
Lanham MD

„Validate Address“ würde die Orthografie des Straßennamens korrigieren, das Suffix hinzufügen und dadurch „4200 Parliament Pl“ als korrigierte Straßenanschrift und „Parlament“ in „AdditionalInputData“ zurückgeben.

Duale Adressen

Eine duale Adresse ist eine Adresse, die sowohl Straßeninformationen als auch Postfach-, Landstraßen- oder Autobahninformationen enthält. Je nach ausgewählten Verarbeitungsoptionen kann der nicht für die Adressstandardisierung verwendete Teil der dualen Adresse in „AdditionalInputData“ zurückgegeben werden. Weitere Informationen finden Sie unter [DualAddressLogic](#) auf Seite 419.

ValidateAddressAUS

„ValidateAddressAUS“ standardisiert und prüft australische Adressen mithilfe der Adressdaten der Australia Post. Durch „Validate Address AUS“ werden zudem fehlende postalische Daten wie Postleitzahlen, Ortsnamen, Namen von Bundesstaaten/Territorien usw. hinzugefügt.

Des Weiteren gibt „Validate AddressAUS“ Indikatoren zu Vergleichsversuchen zurück, z. B. ob „Validate AddressAUS“ die Adresse geprüft hat sowie den Fehlergrund, wenn die Adresse nicht validiert werden konnte, usw.

Während des Adressenvergleichs und der Standardisierung trennt „Validate AddressAUS“ Adresszeilen in Komponenten und vergleicht sie mit den Inhalten einer Datenbank des Universal Addressing-Moduls. Wenn eine Übereinstimmung ermittelt wird, wird die eingegebene Adresse entsprechend den Informationen in der Datenbank *standardisiert*.

„ValidateAddressAUS“ ist Teil des Universal Addressing-Moduls.

Eingabe

„ValidateAddressAUS“ nutzt eine Standardadresse als Eingabe. Alle Adressen verwenden dieses Format.

Tabelle 101: Eingabeformat

columnName	Format	Beschreibung
AddressLine1	Zeichenfolge [288]	Die erste Adresszeile.
AddressLine2	Zeichenfolge [288]	Die zweite Adresszeile.
AddressLine3	Zeichenfolge [288]	Die dritte Adresszeile.
AddressLine4	Zeichenfolge [288]	Die vierte Adresszeile.
City	Zeichenfolge [48]	Der Orts-/Lokalitäts-/Vorortsname. Dieser kann optional in eines der „AddressLine“-Felder zusammen mit dem Bundesland/Kanton und der Postleitzahl eingegeben werden.
StateProvince	Zeichenfolge [4]	Das Bundesland/der Kanton Dies kann optional in eines der „AddressLine“-Felder zusammen mit dem Ort und der Postleitzahl eingegeben werden.
PostalCode	Zeichenfolge [8]	Die Postleitzahl. Diese kann optional in eines der „AddressLine“-Felder zusammen mit dem Bundesland/Kanton und dem Ort eingegeben werden.

Optionen

„ValidateAddressAUS“ bietet mehrere Optionen zur Steuerung der Verarbeitungsweise von Adressen und des zurückgegebenen Datentyps.

Tabelle 102: Optionen

optionName	Beschreibung/Gültige Werte
Database	Gibt die für die Überprüfung von australischen Adressen zu verwendende Datenbank an. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich Australische Datenbankressourcen definiert wurden.
OutputFieldLevelReturnCodes	Gibt Ergebnisfelder aus, die bestimmten Ausgabe-Elementen zugeordnet sind. Siehe Ergebniscodes auf Seite 491. Gültige Werte sind: N Nein, Ergebniscode für einzelne Felder in der Ausgabe nicht einschließen (Standard). Y Ja, Ergebniscode für einzelne Felder in der Ausgabe einschließen.
OutputOriginalInputFields	Gibt die ursprünglichen Eingabedaten zurück. Siehe Ursprüngliche Eingabedaten auf Seite 493. Gültige Werte sind: N Nein, ursprüngliche Eingabedaten nicht in die Ausgabe einschließen (Standard). Y Ja, ursprüngliche Eingabedaten in die Ausgabe einschließen.
OutputMatchedAddressFields	Gibt geparte Adresselemente zurück. Siehe Elemente geparter Adressen auf Seite 492. Gültige Werte sind: N Nein, geparte Adresselemente nicht in die Ausgabe einschließen (Standard). Y Ja, geparte Adresselemente in die Ausgabe einschließen.

optionName	Beschreibung/Gültige Werte
------------	----------------------------

AmasFormatting

Gibt an, dass Ausgabeadressdaten gemäß den AMAS-Konventionen (AMAS = Address Matching Approval System) formatiert werden müssen.

Durch diese Option verwendet „Validate Address AUS“ AMAS-Regeln bei der Standardisierung einer Adresse. AMAS ist ein Programm der Australia Post zur Durchsetzung von Adressenstandards. Weitere Informationen zu den AMAS-Formatierungskonventionen finden Sie im „Address Matching Approval System (AMAS) Handbook“.

Diese Option ändert die Ausgabedaten wie folgt.

- Numerische Felder werden mit Nullen aufgefüllt. Dies hat Auswirkungen auf folgende Ausgabefelder: „HouseNumber“, „HouseNumber2“, „PostalDeliveryNumber“ und „DPID“. Wenn die Eingabeadresse beispielsweise „298 New South Head Rd Double Bay NSW 2028“ lautet, dann wird das Format des Feldes „HouseNumber“ von „298“ in „00298“ geändert.
- Wenn keine Übereinstimmung gefunden wird, sind alle Ziffern im Feld „DPID“ Nullen. Beispiel: 00000000.
- Wenn keine Übereinstimmung gefunden wird, sind alle zurückgegebenen Felder (geparste Adresselemente) leer. Nur numerische Felder enthalten alle Nullen.
- Das Feld „CCD“ wird nicht ausgegeben.

Gültige Werte sind:

- | | |
|----------|--|
| N | Nein, Ausgabedaten nicht gemäß AMAS-Konventionen formatieren (Standard). |
| Y | Ja, Ausgabedaten gemäß AMAS-Konventionen formatieren. |

Ausgabe

Die Ausgabe von „ValidateAddressAUS“ umfasst mindestens die unter **Standard-Ausgabefelder** auf Seite 489 aufgeführten Standard-Ausgabefelder. Zusätzlich zu diesen Standardfeldern kann die Ausgabe auch andere Daten enthalten, je nachdem, welche Ausgabeoptionen Sie auswählen. Weitere Informationen zu den optionalen Ausgabefeldern finden Sie unter **Ergebniscodes** auf Seite 491, **Elemente geparster Adressen** auf Seite 492 und **Ursprüngliche Eingabedaten** auf Seite 493.

Standard-Ausgabefelder

Die folgende Tabelle enthält die von „ValidateAddressAUS“ ausgegebenen Standardfelder.

Tabelle 103: Ausgabefelder

columnName	Beschreibung
AddressLine1	Eine formatierte Adresszeile.
BuildingName	Der Gebäudename.
City	Der Orts-/Lokalitäts-/Vorortsname 1.
City2	Der Orts-/Lokalitäts-/Vorortsname 2 – aufgeteilte Namen z. B. VIA.
StateProvince	Das Bundesland/der Kanton
PostalCode	Die Postleitzahl.
CCD	Der Census Collection District. Die grundlegende geografische Einheit für die Erhebung, Verarbeitung und Ausgabe von Zensusdaten. Im Allgemeinen gibt es etwa 200 bis 250 Haushalte pro CCD und ca. 37.000 CCDs in ganz Australien.
DPID	Die Zustellorts-ID. Eine achtstellige Zahl aus der „Postal Address File“ der Australia Post, mit der ein Zustellort für Briefe, z. B. eine Straßenanschrift, eindeutig identifiziert werden kann.
Status	Der Erfolg oder Fehler des Übereinstimmungsversuchs. F Fehler (keine DPID oder kein CCD gefunden) null Erfolg
Status.Code	Ursache für den Fehler, falls vorhanden. <ul style="list-style-type: none"> • UnableToValidate • InsufficientInputData
Status.Description	Eine Beschreibung des Problems, wenn eines vorliegt.

columnName	Beschreibung
AMAS.ResultCode	Der von der zugrunde liegenden Engine zurückgegebene Ergebniscode.
AMAS.ResultMessage	Die von der zugrunde liegenden Engine zurückgegebenen Ergebnismeldungen.

Ergebniscodes

Mit dieser Option werden Ergebnisfelder ausgegeben, die bestimmten Ausgabe-Elementen zugeordnet sind, sowie einen Ergebniscode für jedes Ergebnisfeld, falls verfügbar. Wenn ein Ergebnisfeld keinen zugehörigen Ergebniscode besitzt, wird möglicherweise eine der folgenden Meldungen angezeigt:

- Es wurde keine Änderung am geparsten Element vorgenommen.
- Das geparste Element wurde standardisiert („Street“ wurde z. B. in „ST“ geändert).
- Es wurden keine Daten in ein entsprechendes geparstes Adresselement geparst.

Tabelle 104: Ergebniscodes

columnName	Ergebniscode	
City.Result	C	Korrigiert
HouseNumber.Result	U	Nicht verglichen, fehlend oder mehrdeutig
PostalCode.Result	C	Korrigiert
PostalDelivery.Result	C	Korrigiert
	D	Verworfen
	U	Nicht verglichen
StateProvince.Result	C	Korrigiert

columnName	Ergebniscode	
StreetName.Result	C	Korrigiert
	U	Nicht verglichen, fehlend oder mehrdeutig
StreetSuffix.Result	C	Korrigiert

Elemente gearparster Adressen

Mit dieser Option werden gearparste Adresselemente ausgegeben.

Tabelle 105: Elemente gearparster Adressen

columnName	Beschreibung
ApartmentLabel	Der Wohnungs- oder Wohneinheitstyp (wie STE oder APT), z. B.: 123 E Main St Apt 3
ApartmentNumber	Der Wohnungs- oder Wohneinheitstyp, z. B.: 123 E Main St Apt 3
FloorLabel	Der Etagen-/Stockwerkstyp, z. B.: 123 E Main St Apt 3, 4th Floor
FloorNumber	Die Etagen-/Stockwerksnummer, z. B.: 123 E Main St Apt 3, 4th Floor
LotNumber	Die Parzellennummer, z. B.: Lot 7 Caldwell Hwy
PostalDeliveryLabel	Der postalische Zustelltyp, z. B.: PO Box 42
PostalDeliveryNumber	Die Nummer für die postalische Zustellung, z. B.: PO Box 42
PostalDeliveryPrefix	Das postalische Zustellnummer-Präfix, z. B.: PO Box A42

columnName	Beschreibung
PostalDeliverySuffix	Das postalische Zustellnummer-Suffix, z. B.: PO Box 42B
HouseNumber	Die Hausnummer 1, z. B.: 298A-1B New South Head Rd
HouseSuffix	Das Hausnummer 1-Suffix, z. B.: 298A-1B New South Head Rd
HouseNumber2	Die Hausnummer 2, z. B.: 298A-1B New South Head Rd
HouseSuffix2	Das Hausnummer 2-Suffix, z. B.: 298A-1B New South Head Rd
StreetName	Der Name der Straße, in der sich das Gebäude befindet, z. B.: 123 E Main St Apt 3
StreetSuffix	Das Straßensuffix, z. B.: 123 E Main St Apt 3
TrailingDirectional	Die nachstehende Richtungsangabe, z. B.: 123 Pennsylvania Ave NW

Ursprüngliche Eingabedaten

Diese Option gibt die ursprünglichen Eingabedaten in <Feldname>.Input-Felder aus.

Tabelle 106: Eingabedaten

columnName	Beschreibung
AddressLine1.Input	Die bei Eingabe übergebene erste Adresszeile.
AddressLine2.Input	Die bei Eingabe übergebene zweite Adresszeile.
AddressLine3.Input	Die bei Eingabe übergebene dritte Adresszeile.

columnName	Beschreibung
AddressLine4.Input	Die bei Eingabe übergebene vierte Adresszeile.
City.Input	Der bei Eingabe übergebene Orts-/Lokalitäts-/Vorortsname.
StateProvince.Input	Das bzw. der bei Eingabe übergebene Bundesland/-staat.
PostalCode.Input	Die bei Eingabe übergebene Postleitzahl.

ValidateAddressGlobal

„ValidateAddressGlobal“ bietet eine erweiterte Adressenstandardisierung und -prüfung für Adressen außerhalb der USA und Kanada. „ValidateAddressGlobal“ kann auch Adressen in den USA und Kanada überprüfen. Die Stärke liegt jedoch in der Überprüfung von Adressen in anderen Ländern. Wenn Sie eine große Anzahl an Adressen außerhalb der USA und Kanada verarbeiten, sollten sie in Erwägung ziehen, „ValidateAddressGlobal“ zu verwenden.

„ValidateAddressGlobal“ ist Teil des Universal Addressing-Moduls.

„ValidateAddressGlobal“ führt mehrere Schritte aus, um eine Qualitätsadresse zu erhalten, einschließlich der Transkription, des Parsing, der Prüfung und Formatierung.

Zeichensatzzuordnung und -transkription

„ValidateAddressGlobal“ verarbeitet internationale Zeichenfolgen und ihre Komplexität. Es verwendet eine vollständig Unicode-aktivierte Zeichenfolgenverarbeitung, mit der die Transkription nicht romanischer Zeichen in den lateinischen Datensatz und die Zuordnung zwischen verschiedenen Zeichensätzen ermöglicht wird.

Zu den Funktionen der Zeichensatzzuordnung und -transkription gehören:

- Unterstützung von über 30 verschiedenen Zeichensätzen einschließlich UTF-8, ISO 8859-1, GBK, BIG5, JIS, EBCDIC
- Ordnungsgemäße „Eliminierung“ von diakritischen Zeichen gemäß Sprachregeln
- Transkription verschiedener Alphabete in die lateinische Schrift
- Griechisch (BGN/PCGN 1962, ISO 843 - 1997)
- Kyrillisch (BGN/PCGN 1947, ISO 9 - 1995)
- Hebräisch
- Japanisch Katakana, Hiragana und Kanji

- Chinesisch Pinyin (Mandarin, Kantonesisch)
- Koreanisch (Hangul)

Parsing, Formatierung und Standardisierung von Adressen

Das Umstrukturieren von Adressdaten, die in die falschen Felder aufgeteilt wurden, ist eine komplexe und schwierige Aufgabe, insbesondere, wenn es um internationale Adressen geht. Menschen machen oft mehrdeutige Angaben, wenn sie Adressdaten in Computersysteme eingeben. Die Probleme umfassen falsch platzierte Elemente (z. B. die Firma oder Personennamen in Feldern für die Straßenanschrift) oder unterschiedliche Abkürzungen, die nicht nur spezifisch für die Sprache, sondern auch für das Land sind. „ValidateAddressGlobal“ erkennt Adresselemente in Adresszeilen und weist sie den richtigen Feldern zu. Dies ist eine wichtige Vorstufe zur eigentlichen Prüfung. Ohne die Umstrukturierung kann es zu Situationen kommen, in denen es keine Übereinstimmungen gibt.

Richtig identifizierte Adresselemente sind zudem von Bedeutung, wenn Adressen abgeschnitten oder gekürzt werden müssen, um bestimmte Längenbegrenzungen zu erfüllen. Mit den richtigen Angaben in den richtigen Feldern können bestimmte Trunkierungsregeln angewandt werden.

- Parsiert und analysiert Adresszeilen und identifiziert einzelne Adresselemente
- Verarbeitet über 30 verschiedene Zeichensätze
- Formatiert Adressen entsprechend den postalischen Regeln des Ziellandes
- Standardisiert Adresselemente (z. B. die Änderung von STRASSE in STR)

Global Address Validation

Bei der Adressenüberprüfung handelt es sich um den Korrekturprozess, bei dem korrekt geparsete Adressdaten mit Referenzdatenbanken abgeglichen werden, die von postalischen Organisationen oder anderen Datenanbietern bereitgestellt werden. „ValidateAddressGlobal“ überprüft einzelne Adresselemente, um deren Richtigkeit zu prüfen. Dabei wird sich einer Technologie für unscharfe Vergleiche bedient, um schließlich standardisierte und formatierte Ausgaben basierend auf Poststandards und Benutzereinstellungen zu erzeugen. Der Überprüfungstyp „FastCompletion“ kann bei Anwendungen mit schneller Adresseingabe verwendet werden. Dadurch können abgeschnittene Daten in mehrere Adressfelder eingegeben werden. Zudem werden Vorschläge basierend auf diesen Eingaben erzeugt.

In einigen Fällen ist es nicht möglich, eine Adresse vollständig zu prüfen. Für diese Fälle verfügt „ValidateAddressGlobal“ über ein einzigartiges Zustellbarkeitsprüfungs-Feature, das Adressen nach ihrer wahrscheinlichen Zustellbarkeit kategorisiert.

Eingabe

„ValidateAddressGlobal“ nutzt eine Standardadresse als Eingabe. Alle Adressen verwenden dieses Format, unabhängig davon, aus welchem Land die Adresse stammt.

Tabelle 107: „ValidateAddressGlobal“-Eingabe

Spaltenname	Format	Beschreibung
AddressLine1 bis AddressLine6	Zeichenfolge [79]	<p>Diese Felder enthalten Adresszeilendaten. AddressLine1 enthält die erste Adresszeile, AddressLine2 die zweite Adresszeile usw. Beachten Sie, dass Daten zu Ort, Bundesland/Kanton und Postleitzahl nicht in die Adresszeilenfelder, sondern in ihre entsprechenden Felder eingegeben werden sollten. Beispiel:</p> <p>AddressLine1: 17413 Blodgett Road AddressLine2: PO Box 123 City: Mount Vernon StateProvince: WA PostalCode: 97273 Country: USA</p> <p>Wenn die Eingabeadresse nicht bereits in die entsprechende Adresszeile und in die Felder „City“, „StateProvince“ sowie „PostalCode“ geparkt wurde, verwenden Sie anstatt der Adresszeilenfelder die Felder „UnformattedLine“.</p>
City	Zeichenfolge [79]	Ortsname
StateProvince	Zeichenfolge [79]	Bundesland oder Kanton.
PostalCode	Zeichenfolge [79]: 99999 99999-9999 A9A9A9 A9A 9A9 9999 999	Die Postleitzahl zu der Adresse. In den USA ist dies der ZIP-Code®.
Contact	Zeichenfolge [79]	Der Name des Adressaten. Beispiel: „Mr. Jones“.
Country	Zeichenfolge [79]	Der Name des Landes. Wenn kein Wert in den Optionen <code>Input.ForceCountryISO3</code> oder <code>Input.DefaultCountryISO3</code> angegeben ist, müssen Sie ein Land angeben.

Spaltenname	Format	Beschreibung
FirmName	Zeichenfolge [79]	Der Unternehmens- bzw. Firmenname.
Street	Zeichenfolge [79]	Straße
Number	Gebäude [79]	Number
Building	Zeichenfolge [79]	Building
SubBuilding	Zeichenfolge [79]	SubBuilding
DeliveryService	Zeichenfolge [79]	DeliveryService
UnformattedLine1 bis UnformattedLine10	Zeichenfolge [79]	<p>Verwenden Sie diese Felder, wenn die Eingabeadresse komplett ungeparst ist und Sie möchten, dass „ValidateAddressGlobal“ versucht, die Adresse in die entsprechenden Felder zu parsen. Beispiel:</p> <p>UnformattedLine1: 17413 Blodgett Road UnformattedLine2: PO Box 123 UnformattedLine3: Mount Vernon WA 97273 UnformattedLine4: USA</p> <p>Diese Adresse würde in die folgenden Ausgabefelder geparkt:</p> <p>AddressLine1: 17413 Blodgett Road AddressLine2: PO Box 123 City: Mount Vernon StateProvince: WA PostalCode: 97273 Country: USA</p> <p>Anmerkung: Wenn Sie für die Eingabe die unformatierten Zeilenfelder nutzen, müssen Sie die gesamte Adresse ausschließlich mithilfe unformatierter Zeilenfelder angeben. Verwenden Sie keine anderen Felder wie „City“ oder „StateProvince“ in Kombination mit unformatierten Zeilenfeldern.</p>

Optionen

Eingabeoptionen

Tabelle 108: „ValidateAddressGlobal“-Eingabeoptionen

optionName	Beschreibung/Gültige Werte
Database.AddressGlobal	Gibt die Datenbankressource an, welche die für die Adressenüberprüfung zu verwendenden Postdaten enthält. Es stehen nur die Datenbanken zur Verfügung, die in der Management Console im Bereich Globale Datenbankressourcen definiert wurden. Weitere Informationen finden Sie im <i>Spectrum™ Technology Platform-Administratorhandbuch</i> .
Input.DefaultCountryISO3	Gibt ein Standardland an, das verwendet werden soll, wenn der Eingabedatensatz keine expliziten Landesdaten enthält. Geben Sie das Land gemäß ISO3-Ländercode an. Wenn Sie kein Standardland angeben, muss bei jedem Eingabedatensatz das Land im Eingabefeld „Land“ angegeben werden. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
Input.ForceCountryISO3	Führt dazu, dass Adressdatensätze immer als von dem hier angegebenen Land stammend behandelt werden; das Land im Adressdatensatz und das Standardland werden dadurch überschrieben. Geben Sie das Land gemäß ISO3-Ländercode an. Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
Input.FormatDelimiter	Damit können Sie nicht standardmäßige Formatierungen für mehrzeilige Adressen in Eingabedateien verwenden. Folgende Werte sind unter anderem für dieses Feld zulässig: <ul style="list-style-type: none"> • CRLF (Standard) • LF • CR • SEMICOLON (2101 MASSACHUSETTS AVE NW ; WASHINGTON DC 20008) • COMMA (2101 MASSACHUSETTS AVE NW , WASHINGTON DC 20008) • TAB (2101 MASSACHUSETTS AVE NW WASHINGTON DC 20008) • PIPE (2101 MASSACHUSETTS AVE NW WASHINGTON DC 20008) • SPACE (2101 MASSACHUSETTS AVE NW WASHINGTON DC 20008) <p>Anmerkung: Derselbe Wert muss sowohl für die Eingabeoption als auch die Ausgabeoption ausgewählt werden.</p>

Ausgabeoptionen

Tabelle 109: „ValidateAddressGlobal“-Ausgabeoptionen

optionName	Beschreibung
Result.MaximumResults	<p>Diese Option gibt die maximale Anzahl zurückzugebender Kandidatenadressen an. Dieses Feld ist für die Batchverarbeitung deaktiviert. Für alle anderen Verarbeitungsmodi beträgt der Standardwert 1 und der Maximalwert lautet 99. Wenn Sie den Modus „FastCompletion“ verwenden, möchten Sie vielleicht eine Nummer größer als „1“ eingeben, um sicherzustellen, dass Ihnen mehrere Optionen zum Ausfüllen eines Feldes zur Verfügung stehen.</p>
Result.IncludeInputs	<p>Gibt an, ob die Eingabedaten in die Ausgabe eingeschlossen werden sollen. Ist diese Option aktiviert, enthält die Ausgabe Felder, die auf „.Input“ enden und die entsprechenden Eingabefelddaten enthalten. Das Ausgabefeld „AddressLine1.Input“ beispielsweise würde die im Eingabefeld „AddressLine1“ angegebenen Daten enthalten.</p> <p>TRUE Eingabedaten in die Ausgabe einschließen.</p> <p>FALSE Eingabedaten nicht in die Ausgabe einschließen (Standard).</p>
Result.StateProvinceType	<p>Gibt das Format für das Feld „StateProvince“ an. Die folgenden Optionen stehen zur Auswahl:</p> <p>ABBREVIATION Gibt die Abkürzung für das Bundesland bzw. den Bundesstaat oder den Kanton bzw. die Provinz an. North Carolina würde beispielsweise als „NC“ zurückgegeben werden.</p> <p>COUNTRY_STANDARD Gibt entweder die Abkürzung oder den vollständigen Namen zurück, je nachdem, welches Format von der Postbehörde des Landes verwendet wird. (Standard)</p> <p>EXTENDED Gibt den vollständigen Namen des Bundeslands/-staats oder des Kantons bzw. der Provinz zurück und nicht die Abkürzung. Beispiel: „North Carolina“.</p>

optionName	Beschreibung
Result.CountryType	Gibt die Sprache oder den zu verwendenden Code für den Ländernamen an, der von „ValidateAddressGlobal“ zurückgegeben wird.
ISO2	Der zweistellige ISO-Code für das Land
ISO3	Der dreistellige ISO-Code für das Land
ISO_NUMBER	Die ISO-Ländernummer
NAME_CN	Chinesisch
NAME_DA	Dänisch
NAME_DE	Deutsch
NAME_EN	Englisch (Standard)
NAME_ES	Spanisch
NAME_FI	Finnisch
NAME_FR	Französisch
NAME_GR	Griechisch
NAME_HU	Ungarisch
NAME_IT	Italienisch
NAME_JP	Japanisch
NAME_KR	Koreanisch
NAME_NL	Niederländisch
NAME_PL	Polnisch
NAME_PT	Portugiesisch
NAME_RU	Russisch
NAME_SA	Sanskrit
NAME_SE	Schwedisch

optionName

Beschreibung

Result.PreferredScript

Gibt das Alphabet an, in dem die Ausgabe zurückgegeben werden soll. Das Alphabet, in dem die Daten zurückgegeben werden, ist von Land zu Land unterschiedlich. Bei den meisten Ländern ist das Alphabet der Ausgabe „Latin I“, unabhängig von der ausgewählten bevorzugten Sprache.

ASCII_Extended	ASCII-Zeichen mit Erweiterung von Sonderzeichen (z. B. Ã– = OE)
ASCII_Simplified	ASCII-Zeichen
Database	(Standard) Latin I- oder ASCII-Zeichen (gemäß Referenzdatenbank-Standard)
Latin	Latin I-Zeichen
Latin_Alt	Latin I-Zeichen (alternative Transkription)
Postal_Admin_Alt	Latin I- oder ASCII-Zeichen (Alternative der lokalen Postbehörde)
Postal_Admin_Pref	Latin I- oder ASCII-Zeichen (je nach Wahl der lokalen Postbehörde)

Bei Ländern, die ein anderes Alphabet als Latin I verwenden, ist das Alphabet von Land zu Land unterschiedlich. Weitere Informationen finden Sie unter [Alphabete für Länder, die kein Latin I verwenden](#) auf Seite 502.

Result.PreferredLanguage

Gibt die Sprache an, in der die Ausgabe zurückgegeben werden soll. Das Alphabet, in dem die Daten zurückgegeben werden, ist von Land zu Land unterschiedlich, doch bei den meisten Ländern ist das Alphabet der Ausgabe „Latin I“, unabhängig von der ausgewählten bevorzugten Sprache.

DATABASE	Die für jede Adresse aus den Referenzdaten abgeleitete Sprache. Standardeinstellung.
ENGLISH	Ausgabe von englischen Namen von Lokalitäten und Bundesländern/-staaten bzw. Kantonen/Provinzen, sofern verfügbar.

optionName	Beschreibung
Result.Casing	<p>Gibt die Groß-/Kleinschreibung der Ausgabe an.</p> <p>NATIVE Die Ausgabe basiert auf dem Referenzdatenbank-Standard.</p> <p>UPPER Alle Länder werden in Großbuchstaben ausgegeben.</p> <p>LOWER Alle Länder werden in Kleinbuchstaben ausgegeben.</p> <p>MIXED Groß-/Kleinschreibung wird von landesspezifischen Regeln bestimmt.</p> <p>NOCHANGE Im „Parse“-Modus werden die Daten zurückgegeben, wie sie eingegeben wurden. Im Überprüfungsmodus wird die Groß-/Kleinschreibung verwendet, die in den Referenzdaten gefunden wird und den postalischen Bestimmungen des Landes entspricht. Werte, die nicht mit den Referenzdaten verglichen werden konnten, behalten die eingegebene Groß-/Kleinschreibung bei.</p>
Result.FormatDelimiter	<p>Damit können Sie nicht standardmäßige Formatierungen für mehrzeilige Adressen in der Ausgabe verwenden. Folgende Werte sind unter anderem für dieses Feld zulässig:</p> <ul style="list-style-type: none"> • CRLF (Standard) • LF • CR • SEMICOLON (2101 MASSACHUSETTS AVE NW ; WASHINGTON DC 20008) • COMMA (2101 MASSACHUSETTS AVE NW , WASHINGTON DC 20008) • TAB (2101 MASSACHUSETTS AVE NW WASHINGTON DC 20008) • PIPE (2101 MASSACHUSETTS AVE NW WASHINGTON DC 20008) • SPACE (2101 MASSACHUSETTS AVE NW WASHINGTON DC 20008) <p>Anmerkung: Derselbe Wert muss sowohl für die Eingabeoption als auch die Ausgabeoption ausgewählt werden.</p>

Alphabete für Länder, die kein Latin I verwenden

Bei Ländern, die ein anderes Alphabet als Latin I verwenden, ist das Alphabet von Land zu Land unterschiedlich. Die folgende Tabelle zeigt, wie die Ausgabe für bestimmte Länder zurückgegeben wird. Alle nicht aufgeführten Länder verwenden den in der Feldoption `Result.PreferredScript` angegebenen Wert.

Country	Datenbank	Post_Admi_Pef	Post_Admi_At	Latin	Latin_Alt	ASCII_Simplified	ASCII_Extended
RUS	Kyrillisch	Kyrillisch	Kyrillisch	CYRILLIC_ISO	CYRILLIC_BGN	CYRILLIC_ISO + LATIN_SIMPLE	CYRILLIC_ISO + LATIN
JPN	Kanji	Kanji	Kana	JAPANESE	JAPANESE	JAPANESE + LATIN_SIMPLE	JAPANESE + LATIN
CHN	Hanzi	Hanzi	Hanzi	CHINESE_ MANDARIN	CHINESE_ CANTONESE	CHINESE_ MANDARIN + LATIN_SIMPLE	CHINESE_ MANDARIN + LATIN
HKG	Hanzi	Hanzi	Hanzi	CHINESE_ CANTONESE	CHINESE_ MANDARIN	CHINESE_ CANTONESE + LATIN_SIMPLE	CHINESE_ CANTONESE + LATIN
TWN	Hanzi	Hanzi	Hanzi	CHINESE_ CANTONESE	CHINESE_ MANDARIN	CHINESE_ CANTONESE + LATIN_SIMPLE	CHINESE_ CANTONESE + LATIN
GRC	Griechisch	Griechisch	Griechisch	GREEK_ISO	GREEK_BGN	GREEK_ISO + LATIN_SIMPLE	GREEK_ISO + LATIN
KOR	Latin	Hangul	Hanja	KOREAN	KOREAN	KOREAN + LATIN_SIMPLE	KOREAN + LATIN
ISR	Latin	Hebräisch	Hebräisch	HEBREW	HEBREW	HEBREW + LATIN_SIMPLE	HEBREW + LATIN
ROM	Latin-3	Latin-3	Latin-3	Latin-3	Latin-3	LATIN_SIMPLE	LATIN
POL	Latin-2	Latin-2	Latin-2	Latin-2	Latin-2	LATIN_SIMPLE	LATIN
CZE	Latin-2	Latin-2	Latin-2	Latin-2	Latin-2	LATIN_SIMPLE	LATIN
CRI	Latin-2	Latin-2	Latin-2	Latin-2	Latin-2	LATIN_SIMPLE	LATIN
HUN	Latin-2	Latin-2	Latin-2	Latin-2	Latin-2	LATIN_SIMPLE	LATIN
MDA	Latin-2	Latin-2	Latin-2	Latin-2	Latin-2	LATIN_SIMPLE	LATIN

Country	Datenbank	Post_Admi_Pref	Post_Admi_At	Latin	Latin_Alt	ASCII_Simplified	ASCII_Extended
SVK	Latin-2	Latin-2	Latin-2	Latin-2	Latin-2	LATIN_SIMPLE	LATIN
LAT	Latin-7	Latin-7	Latin-7	Latin-7	Latin-7	LATIN_SIMPLE	LATIN

Verarbeitungsoptionen

Tabelle 110: „ValidateAddressGlobal“-Verarbeitungsoptionen

optionName	Beschreibung
Process.OptimizationLevel	<p>Verwenden Sie diese Option, um ein geeignetes Gleichgewicht zwischen Verarbeitungsgeschwindigkeit und Qualität festzulegen. Zur Auswahl stehen:</p> <p>NARROW Der Parser berücksichtigt die Eingabezuweisung strengstens, mit Ausnahme der Trennung der „House Number“-Daten von „Street“-Daten.</p> <p>STANDARD Der Parser trennt Adresselemente aktiver, und zwar wie folgt:</p> <ul style="list-style-type: none"> • „Province“-Daten werden von „Locality“-Daten getrennt. • „PostalCode“-Daten werden von „Locality“-Daten getrennt. • „House Number“-Daten werden von „Street“-Daten getrennt. • „SubBuilding“-Daten werden von „Street“-Daten getrennt. • „DeliveryService“-Daten werden von „Street“-Daten getrennt. • „SubBuilding“-Daten werden von „Building“-Daten getrennt. • „Locality“-Daten werden von „PostalCode“-Daten getrennt. <p>WIDE Die Parser-Trennung geschieht ähnlich wie bei Standard, aber es werden zusätzlich bis zu 10 Parsing-Kandidaten zur Überprüfung weitergegeben. Die Überprüfung erweitert ihren Suchbaum und berücksichtigt beim Vergleich zusätzliche Referenzdateneinträge.</p> <p>Beachten Sie, dass die Anpassung des Optimierungsgrads möglicherweise keine Auswirkung auf Länder hat, denen Postreferenzdaten fehlen, die für die oben beschriebene Trennung erforderlich sind.</p> <p>Durch das Ändern der Trennungsgenauigkeit von Narrow in Standard wird eine höhere Verarbeitungsleistung nötig, doch der wesentlich Einflussfaktor auf die Verarbeitungsgeschwindigkeit stammt von der Überprüfung eines größeren Suchbaums. Dadurch wird die Anzahl der Datenzugriffe und -vergleiche für den Optimierungsgrad Wide erhöht, um das Potenzial der gegebenen Eingabedaten voll auszuschöpfen.</p>

optionName	Beschreibung
Process.Mode	<p>Gibt den für Adressen auszuführenden Verarbeitungstyp an. Zur Auswahl stehen:</p> <p>BATCH Verwenden Sie diesen Modus in Batch-Verarbeitungsumgebungen, wenn keine Benutzereingabe oder -auswahl möglich ist. Er ist für Geschwindigkeit optimiert und beendet Versuche, eine Adresse zu korrigieren, wenn mehrdeutige Daten auftreten, die nicht automatisch korrigiert werden können. Der Verarbeitungsmodus „Batch“ fällt auf den Modus „Parsen“ zurück, wenn die Datenbank für ein bestimmtes Land fehlt.</p> <p style="padding-left: 40px;">Anmerkung: Wenn der Verarbeitungsstatus den Wert I3 zurückgibt, wird der Versuch als fehlgeschlagen erachtet und der Status gibt den Wert F zurück.</p> <p>CERTIFIED Verwenden Sie diesen Modus in Batch-Verarbeitungsumgebungen für australische Post. Validate Address Global ist durch das Address Matching Approval System (AMAS) der australischen Post zertifiziert. Mit Validate Address Global wird Ihre Post entsprechend der Postal Address File standardisiert und dagegen geprüft. Zudem werden Portoermäßigungen geboten und die geringste Menge an unzustellbaren Versandstücken zugelassen.</p> <p>FASTCOMPLETION Verwenden Sie den Modus „FastCompletion“, um abgeschnittene Daten in Adressfelder einzugeben und „Validate Address Global“ Vorschläge generieren zu lassen. Wenn Sie beispielsweise in einem Callcenter oder in einer Verkaufsstelle arbeiten, können Sie nur einen Teil eines Adresselements eingeben und das „FastCompletion“-Feature wird Ihnen gültige Optionen für das vollständige Element vorschlagen.</p> <p>INTERACTIVE Verwenden Sie diesen Modus, wenn Sie in interaktiven Umgebungen arbeiten, um Vorschläge bei einer mehrdeutigen Adresseingabe zu generieren. Dieser Überprüfungstyp eignet sich am besten bei der Erfassung von Daten von Kunden oder Interessenten. Er erfordert die Eingabe einer nahezu vollständigen Adresse und versucht die angegebenen Daten zu überprüfen oder zu korrigieren. Wenn Mehrdeutigkeiten erkannt werden, generiert dieser Überprüfungstyp bis zu 20 Vorschläge, die für Auswahllisten verwendet werden können. Der Verarbeitungsmodus „Interaktiv“ fällt auf den Modus „Parsen“ zurück, wenn die entsprechende Datenbank für ein bestimmtes Land fehlt.</p> <p>PARSE Verwenden Sie diesen Modus zur Aufteilung von Adresseingaben in Token für die anschließende Verarbeitung in anderen Systemen, wodurch Überprüfung umgangen wird. Sie können diesen Modus beispielsweise nutzen, wenn Adressdaten von bereits sehr hoher Qualität einfach nur schnell tokenisiert werden sollen, um sie in ein externes System zur Verwendung durch einen Downstream-Schritt zu exportieren.</p>

optionName	Beschreibung
Process.MatchingScope	<p>Gibt an, wie genau eine Adresse mit den Referenzdaten übereinstimmen muss, damit die Adresse überprüft wird. Zur Auswahl stehen:</p> <p>Anmerkung: Diese Einstellungen haben möglicherweise keine Auswirkung auf Länder, denen die nötige Detailebene in den Postreferenzdaten fehlt.</p> <p>ALL Alle Adresselemente müssen übereinstimmen.</p> <p>DELIVERYPOINT_LEVEL „Validate Global Address“ muss eine Übereinstimmung mit den Feldern „StateProvince“, „PostalCode“, „City/Locality/Suburb“, „Street“, „House Number“ und „Sub Building“ haben.</p> <p>STREET_LEVEL „Validate Global Address“ muss eine Übereinstimmung mit den Feldern „StateProvince“, „PostalCode“, „City/Locality/Suburb“ und „Street“ haben.</p> <p>LOCALITY_LEVEL „Validate Global Address“ muss eine Übereinstimmung mit den Feldern „StateProvince“, „PostalCode“ und „City/Locality/Suburb“ haben.</p>

Ausgabe

Adressdaten

Tabelle 111: Elemente gearpster Adressen

columnName	Beschreibung
AddressBlock1-9	<p>Die „AddressBlock“-Ausgabefelder enthalten eine formatierte Version der standardisierten oder normalisierten Adresse, wie sie auf eine physische Postsendung gedruckt würde. „Validate Address Global“ formatiert die Adresse gemäß Postbehördenstandards in Adressblöcke. Jede Adresszeile wird in einem separaten Adressblockfeld zurückgegeben. Es kann bis zu neun Adressblock-Ausgabefelder geben: AddressBlock1 bis AddressBlock9. Zum Beispiel diese Eingabeadresse:</p> <p>AddressLine1: 4200 Parliament Place AddressLine2: Suite 600 City: Lanham StateProvince: MD PostalCode: 20706</p> <p>Dies führt zu folgender Adressblockausgabe:</p> <p>AddressBlock1: 4200 Parliament PL STE 600 AddressBlock2: Lanham MD 20706-1882</p>
AddressLine1-6	<p>Wenn die Adresse überprüft wurde, enthalten die Adresszeilenfelder die überprüften und standardisierten Adresszeilen. Wenn die Adresse nicht überprüft werden konnte, enthalten die Adresszeilenfelder die Eingabeadresse ohne Änderungen. Beachten Sie, dass die letzte Zeile der Adresse im Feld „LastLine“ enthalten ist. Beispiel:</p> <p>AddressLine1: 4200 Parliament PL STE 600 LastLine: LANHAM MD 20706-1882</p>
AdministrativeDistrict	Eine Fläche, die kleiner als ein Bundesland/Kanton, aber größer als ein Ort ist.
ApartmentLabel	Der Wohnungs- oder Wohneinheitstyp (wie STE oder APT), z. B.: 123 E Main St Apt 3
ApartmentNumber	Der Wohnungs- oder Wohneinheitstyp, z. B.: 123 E Main St Apt 3
BlockName	Ein Grundstücks- oder Blockname.

columnName	Beschreibung
BuildingName	Der Name eines Gebäudes, z. B. Sears Tower.
City	Der Name einer Stadt oder eines Orts. Beispiel: Vancouver , BC.
City.AddInfo	Weitere Informationen über den Ort.
City.SortingCode	Ein von Postbehörden verwendeter Code, um die Zustellung in bestimmten Ländern für große Ortschaften, z. B. Prag oder Dublin, zu beschleunigen.
Contact	Der Name des Adressaten. Beispiel: Mr. Jones .
Country	Das Land in der Sprache oder dem Code, der unter der Option <code>Result.CountryType</code> angegeben ist.
County	Abhängige Bundesland-/Kanton-/Bundesstaat- oder Provinzinformationen, die ein Bundesland/Kanton bzw. einen Bundesstaat oder eine Provinz weiter unterteilen. Ein Beispiel ist ein US-amerikanischer Bezirk (County).
FirmName	Der Name eines Unternehmens.
Floor	Informationen, die ein Gebäude weiter unterteilen, z. B. Suite- oder Apartmentnummer. Beispiel: 123 E Main St Apt 3, 4th Floor
HouseNumber	Die Hausnummer 1, z. B.: 298A-1B New South Head Rd
LastLine	Vollständige letzte Adresszeile (Ort, Bundesstaat-/land bzw. Provinz/Kanton und Postleitzahl).
LeadingDirectional	Die Straßenrichtung, die vor dem Straßennamen steht. Zum Beispiel das „N“ in 138 N Main Street.
Locality	Abhängiger Ortsname, der eine Lokalität weiter unterteilt. Beispiele sind „Colonias“ in Mexiko oder „Urbanisaciones“ in Spanien.
POBox	Postfach-Deskriptor (POBox, Postfach, Case Postale usw.) und Nummer.
PostalCode	Die Postleitzahl zu der Adresse. Das Format der Postleitzahl variiert je nach Land.

columnName	Beschreibung
PostalCode.AddOn	Der zweite Teil einer Postleitzahl. Bei kanadischen Adressen ist dieser beispielsweise die LDU. Bei US-amerikanischen Adressen ist dies der ZIP + 4-Zusatz. Dieses Feld wird von den meisten Ländern nicht verwendet.
PostalCode.Base	Der Basisteil der Postleitzahl.
Room	Eine Zimmernummer in einem Gebäude.
SecondaryStreet	Der Name einer sekundären Straße oder Landstraße.
StateProvince	Der Name des Bundesland/-staats bzw. Kantons oder der Provinz.
StreetName	Der Name der Straße, in der sich das Gebäude befindet, z. B.: 123 E Main St Apt 3
StreetSuffix	Das Straßensuffix, z. B.: 123 E Main St Apt 3
SubBuilding	Ein Teil eines Gebäudes, z. B. eine Suite. Beispiel: Suite 102.
Suburb	Abhängiger Ortsname, der eine Lokalität weiter unterteilt. Ein Beispiel ist „Mahalle“ in der Türkei.
Territory	Der Name eines Territoriums. Territorien sind größer als ein Bundesland/-staat/Kanton bzw. eine Provinz.
TrailingDirectional	Die nachstehende Richtungsangabe, z. B.: 123 Pennsylvania Ave NW

Ursprüngliche Eingabedaten

Diese Option gibt die ursprünglichen Eingabedaten in <Feldname>.Input-Felder aus.

Tabelle 112: Ursprüngliche Eingabedaten

columnName	Format	Beschreibung
AddressLine1.Input	Zeichenfolge [79]	Erste Adresszeile
AddressLine2.Input	Zeichenfolge [79]	Zweite Adresszeile
AddressLine3.Input	Zeichenfolge [79]	Dritte Adresszeile
AddressLine4.Input	Zeichenfolge [79]	Vierte Adresszeile
AddressLine5.Input	Zeichenfolge [79]	Fünfte Adresszeile
AddressLine6.Input	Zeichenfolge [79]	Sechste Adresszeile
City.Input	Zeichenfolge [79]	Ortsname
StateProvince.Input	Zeichenfolge [79]	Bundesland/-staat oder Kanton/Provinz
PostalCode.Input	Zeichenfolge [79]:	Die Postleitzahl zu der Adresse. In den USA ist dies der Postleitzahlencode. Eines dieser Formate: 99999 99999-9999 A9A9A9 A9A 9A9 9999 999

columnName	Format	Beschreibung
Contact.Input	Zeichenfolge [79]	Der Name des Adressaten. Beispiel: „Mr. Jones“.
Country.Input	Zeichenfolge [79]	Geben Sie das Land in dem Format an, das Sie für das Eingabeländerformat ausgewählt haben (englischer Name, ISO-Code oder UPU-Code). Eine Liste der Werte finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.
FirmName.Input	Zeichenfolge [79]	Der Unternehmens- bzw. Firmenname.
Street.Input	Zeichenfolge [79]	Straße
Number.Input	Gebäude [79]	Number
Building.Input	Zeichenfolge [79]	Building
SubBuilding.Input	Zeichenfolge [79]	SubBuilding
DeliveryService.Input	Zeichenfolge [79]	DeliveryService

Ergebniscodes

Diese Ausgabefelder enthalten Informationen über das Ergebnis des Überprüfungsvorgangs.

Tabelle 113: Ergebniscode

columnName	Ergebniscode
AddressType	<p>Nur bei Adressen der USA und Kanada gibt das Feld „AddressType“ den Typ der Adresse an. Zur Auswahl stehen:</p> <p>F Die Adresse wurde mit dem Firmennamen überprüft/korrigiert.</p> <p>B Die Adresse wurde mit dem Gebäudenamen überprüft/korrigiert.</p> <p>G Die Adresse ist eine allgemeine Zustelladresse.</p> <p>H Die Adresse wurde mit dem Hochhausstandard überprüft/korrigiert.</p> <p>L Die Adresse ist ein Großempfänger.</p> <p>M Die Adresse ist eine Militäradresse.</p> <p>P Die Adresse wurde mit einem Postfach überprüft/korrigiert.</p> <p>R Die Adresse wurde mit einer Landstraße überprüft/korrigiert.</p> <p>S Die Adresse wurde mit einer Straßenanschrift überprüft/korrigiert.</p> <p>U Die Adresse konnte nicht überprüft/korrigiert werden; der Typ ist daher unbekannt.</p>
Confidence	<p>Das Zuverlässigkeitsniveau, das der zurückgegebenen Adresse zugewiesen ist. Der Bereich reicht von 0 (null) bis 100. Null gibt an, dass ein Fehler vorliegt; 100 gibt an, dass das Zuverlässigkeitsniveau sehr hoch ist und die Übereinstimmungsergebnisse richtig sind.</p>
CountOverflow	<p>Gibt an, ob die Anzahl der Kandidatenadressen die zurückgegebene Anzahl überschreitet. Zur Auswahl stehen:</p> <p>Ja Ja, es gibt zusätzliche Kandidatenadressen. Erhöhen Sie zum Abrufen der zusätzlichen Kandidaten den Wert für <code>MaximumResults</code>.</p> <p>Nein Nein, es gibt keine zusätzlichen Kandidaten.</p>
ElementInputStatus	<p>„ElementInputStatus“ bietet für jedes Element Informationen über den Vergleich von Eingabe-Elementen mit Referenzdaten. Die Werte in diesem Feld variieren je nachdem, ob Sie den Batch- oder den Parse-Modus verwenden. Weitere Informationen über den Wert in diesem Feld finden Sie unter Interpretieren der Felder „ElementInputStatus“, „ElementResultStatus“, und „ElementRelevance“ auf Seite 517.</p>

columnName	Ergebniscode												
ElementRelevance	Gibt an, welche Adresselemente aus der Sicht der lokalen Postbehörde tatsächlich relevant sind. Weitere Informationen über den Wert in diesem Feld finden Sie unter Interpretieren der Felder „ElementInputStatus“, „ElementResultStatus“, und „ElementRelevance“ auf Seite 517.												
ElementResultStatus	„ElementResultStatus“ kategorisiert das Ergebnis detaillierter als das Feld „ProcessStatus“, indem es angibt, ob und wie sich die Ausgabefelder von den Eingabefeldern unterscheiden. Weitere Informationen über den Wert in diesem Feld finden Sie unter Interpretieren der Felder „ElementInputStatus“, „ElementResultStatus“, und „ElementRelevance“ auf Seite 517.												
MailabilityScore	<p>Eine Schätzung, wie wahrscheinlich es ist, dass an diese Adresse gesendete Post erfolgreich zugestellt wird. Zur Auswahl stehen:</p> <table border="0"> <tr> <td>5</td> <td>Zustellbarkeit ist vollkommen sichergestellt</td> </tr> <tr> <td>4</td> <td>Zustellbarkeit ist nahezu sichergestellt</td> </tr> <tr> <td>3</td> <td>Zustellbarkeit sollte erfolgen</td> </tr> <tr> <td>2</td> <td>Gute Chance</td> </tr> <tr> <td>1</td> <td>Riskant</td> </tr> <tr> <td>0</td> <td>Keine Chance</td> </tr> </table>	5	Zustellbarkeit ist vollkommen sichergestellt	4	Zustellbarkeit ist nahezu sichergestellt	3	Zustellbarkeit sollte erfolgen	2	Gute Chance	1	Riskant	0	Keine Chance
5	Zustellbarkeit ist vollkommen sichergestellt												
4	Zustellbarkeit ist nahezu sichergestellt												
3	Zustellbarkeit sollte erfolgen												
2	Gute Chance												
1	Riskant												
0	Keine Chance												
ModeUsed	Gibt den verwendeten Verarbeitungsmodus an. Der Verarbeitungsmodus wird unter der Option <code>Process.Mode</code> angegeben. Eine Beschreibung der Modi finden Sie unter Verarbeitungsoptionen auf Seite 504.												
MultimatchCount	Wenn die Adresse mit mehreren Kandidatenadressen in den Referenzdaten übereingestimmt hat, enthält dieses Feld die Anzahl der gefundenen Kandidatenübereinstimmungen.												

columnName Ergebniscode

ProcessStatus

columnName

Ergebniscode

Enthält eine allgemeine Beschreibung der Ausgabequalität. Eine detailliertere Beschreibung der Ausgabequalität finden Sie im Feld „ElementResultStatus“.

Zur Auswahl stehen:

- V4** Überprüft. Die Eingabedaten sind richtig. Alle Elemente wurden überprüft und die Eingabe ergab eine genaue Übereinstimmung.
- V3** Überprüft. Die Eingabedaten sind richtig bei der Eingabe, aber einige oder alle Elemente wurden standardisiert oder die Eingabe enthält veraltete Namen oder Synonyme.
- V2** Überprüft. Die Eingabedaten sind richtig, aber einige Elemente konnten aufgrund unvollständiger Referenzdaten nicht überprüft werden.
- V1** Überprüft. Die Eingabedaten sind richtig, aber die Benutzerstandardisierung hat die Zustellbarkeit verschlechtert (falsche Element-Benutzerstandardisierung – z. B. ist die ausgewählte Länge der Postleitzahl zu kurz). Nicht bei Überprüfung festgelegt.
- C4** Korrigiert. Alle Elemente wurden überprüft.
- C3** Korrigiert, aber einige Elemente konnten nicht überprüft werden.
- C2** Korrigiert, aber der Zustellstatus ist nicht eindeutig (fehlende Referenzdaten).
- C1** Korrigiert, aber der Zustellstatus ist nicht eindeutig, weil die Benutzerstandardisierung falsch war. Nicht bei Überprüfung festgelegt.
- I4** Die Daten konnte nicht vollkommen korrigiert werden, aber die Zustellbarkeit ist sehr wahrscheinlich. Einzelne Übereinstimmung (z. B. „HNO“ ist falsch, und es wurde nur „1 HNO“ in den Referenzdaten gefunden).
- I3** Die Daten konnte nicht vollkommen korrigiert werden, aber die Zustellbarkeit ist sehr wahrscheinlich. Mehrfachübereinstimmungen (z. B. „HNO“ ist falsch, aber es wurde mehr als ein „1 HNO“ in den Referenzdaten gefunden).
- I2** Die Daten konnten nicht korrigiert werden, aber es gibt eine geringe Chance, dass die Adresse zustellbar ist.
- I1** Die Daten konnte nicht korrigiert werden und die Zustellbarkeit ist unwahrscheinlich.
- RA** Das Land wurde mithilfe der Einstellung „Land erzwingen“ erkannt.
- R9** Das Land wurde mithilfe der Einstellung „DefaultCountryISO3“ erkannt.
- R8** Das Land wurde anhand des Namens fehlerfrei erkannt.
- R7** Das Land wurde anhand des Namens mit Fehlern erkannt.
- R6** Das Land wurde anhand des Territoriums erkannt.
- R5** Das Land wurde anhand der Provinz erkannt.
- R4** Das Land wurde anhand der Großstadt erkannt.
- R3** Das Land wurde anhand des Formats erkannt.

columnName	Ergebniscode
	R2 Das Land wurde anhand des Skripts erkannt.
	R1 Das Land wurde nicht erkannt – Mehrfachübereinstimmungen.
	R0 Das Land wurde nicht erkannt.
	S4 Genau geparkt.
	S3 Mit mehreren Ergebnissen geparkt.
	S2 Mit Fehlern geparkt. Elemente ändern die Position.
	S1 Parsing-Fehler. Keine Übereinstimmung beim Eingabeformat.
	N1 Überprüfungsfehler: Es wurde keine Überprüfung ausgeführt, weil das Land nicht erkannt wurde.
	N2 Überprüfungsfehler: Es wurde keine Überprüfung ausgeführt, weil die erforderliche Referenzdatenbank nicht verfügbar ist.
	N3 Überprüfungsfehler: Es wurde keine Überprüfung ausgeführt, weil das Land nicht entsperrt werden konnte.
	N4 Überprüfungsfehler: Es wurde keine Überprüfung ausgeführt, weil die Referenzdatenbank fehlerhaft ist oder das falsche Format hat.
	N5 Überprüfungsfehler: Es wurde keine Überprüfung ausgeführt, weil die Referenzdatenbank zu alt ist.
	N6 Überprüfungsfehler: Es wurde keine Überprüfung ausgeführt, weil die Eingabedaten unzureichend sind.
	Q3 Status „FastCompletion“: Vorschläge sind verfügbar – vollständige Adresse.
	Q2 Status „FastCompletion“: Vorgeschlagene Adresse ist vollständig, aber mit Elementen aus der Eingabe kombiniert (hinzugefügt oder gelöscht).
	Q1 Status „FastCompletion“: Vorgeschlagene Adresse ist unvollständig (weitere Informationen eingeben).
	Q0 Status „FastCompletion“: Unzureichende Informationen zum Generieren von Vorschlägen angegeben.
Status	Meldet den Erfolg oder Fehler des Verarbeitungsversuchs. null Erfolg F Fehler
Status.Code	Die Ursache für den Fehler, falls es einen gab.
Status.Description	Eine Beschreibung des Fehlers, falls es einen gab.

Interpretieren der Felder „ElementInputStatus“, „ElementResultStatus“, und „ElementRelevance“

Die Ausgabefelder „ElementInputStatus“, „ElementResultStatus“ und „ElementRelevance“ enthalten eine Reihe von Ziffern, die das Ergebnis des Überprüfungsvorgangs im Detail erläutern.

„ElementInputStatus“ enthält Informationen für Parsing-Vorgänge.

So sieht ein Wert des Feldes „ElementInputStatus“ aus:

44606040600000000060

So sieht ein Wert des Feldes „ElementResultStatus“ aus:

88F0F870F00000000040

So sieht ein Wert des Feldes „ElementRelevance“ aus:

11101010100000000000

Um die Werte in diesen Feldern zu verstehen, müssen Sie wissen, für welches Element jede einzelne Position steht und welche Bedeutung die Werte in jeder einzelnen Position haben. Die erste Ziffer beispielsweise gibt das Ergebnis des Ausgabefeldes „PostalCode.Base“ an. Die Bedeutungen der Positionen sind nachfolgend aufgeführt.

- Position 1: PostalCode.Base
- Position 2: PostalCode.AddOn
- Position 3: City
- Position 4: Locality and Suburb
- Position 5: StateProvince
- Position 6: County
- Position 7: StreetName
- Position 8: SecondaryStreet
- Position 9: HouseNumber
- Position 10: Number level 1
- Position 11: POBox
- Position 12: Delivery service level 1
- Position 13: Building level 0
- Position 14: BuildingName
- Position 15: Sub building level 0
- Position 16: Floor and Room
- Position 17: FirmName
- Position 18: Organization level 1
- Position 19: Country
- Position 20: Territory

Für das Feld „ElementInputStatus“ lauten die möglichen Überprüfungswerte:

- 0: Leer
- 1: Nicht gefunden
- 2: Nicht geprüft (keine Referenzdaten)

- 3: Falsch – Wird nur bei Überprüfung festgelegt: Die Referenzdatenbank gibt an, dass entweder „Number“ oder „DeliveryService“ außerhalb des gültigen Nummernbereichs liegen. Eingabe wird kopiert, im Batch-Modus nicht korrigiert, im interaktiven Modus und bei „FastCompletion“ werden Vorschläge angeboten.
- 4: Übereinstimmung mit Fehlern in diesem Element
- 5: Übereinstimmung mit Änderungen (eingefügte und gelöschte Elemente), zum Beispiel:
 - Parsing: Aufteilung der Hausnummer für „MainSt 1“
 - Überprüfung: Ersetzen synonyme Eingaben oder Weglassen überflüssiger Feldeingaben, die laut Länderreferenzdatenbank ungültig sind
- 6: Übereinstimmung ohne Fehler

Für das Feld „ElementInputStatus“ lauten die möglichen Parsing-Werte:

- 0: Leer
- 1: Element musste verschoben werden
- 2: Übereinstimmung, aber musste normalisiert werden
- 3: Übereinstimmung

Für das Feld „ElementRelevance“ lauten die möglichen Parsing-Werte:

- 0: Leer
- 1: Element musste verschoben werden
- 2: Übereinstimmung, aber musste normalisiert werden
- 3: Übereinstimmung

Für das Feld „ElementResultStatus“ lauten die möglichen Werte (für alle Adresselemente außer dem Land):

- 0: Leer
- 1: Nicht überprüft und nicht geändert. Ursprüngliche Daten werden kopiert.
- 2: Nicht überprüft, aber standardisiert.
- 3: Überprüft, aber nicht geändert aufgrund einer ungültigen Eingabe; die Datenbank gibt an, dass die Nummer außerhalb des gültigen Bereichs liegt. Eingabe wird kopiert, nicht korrigiert – dieser Statuswert wird nur im Batch-Modus festgelegt.
- 4: Überprüft, aber nicht geändert aufgrund fehlender Referenzdaten.
- 5: Überprüft, aber nicht geändert aufgrund von Mehrfachübereinstimmungen. Wird nur im Batch-Modus festgelegt, anderenfalls werden Mehrfachvorschläge, welche die Eingabe ersetzen, als korrigiert markiert (Statuswert 7).
- 6: Überprüft und geändert durch Entfernen des Eingabewertes
- 7: Überprüft und geändert durch Korrektur anhand von Referenzdaten
- 8: Überprüft und geändert durch Hinzufügen eines Wertes anhand von Referenzdaten
- 9: Überprüft, nicht geändert, aber Zustellstatus ist nicht eindeutig (z. B. falscher DPV-Wert; gegebene Nummernbereiche, die nur partiell mit Referenzdaten übereinstimmen).
- C: Überprüft, verifiziert, aber aufgrund eines veralteten Namens geändert
- D: Überprüft, verifiziert, aber von Synonym in offiziellen Namen geändert

- E: Überprüft, verifiziert, aber aufgrund von Standardisierung anhand von Groß-/Kleinschreibung oder Sprache geändert. Bei der Überprüfung wird dieser Status nur festgelegt, wenn die Eingabe vollständig mit einer sprachlichen Alternative übereinstimmt.
- F: Überprüft, verifiziert, aber aufgrund einer genauen Übereinstimmung nicht geändert

Für das Feld „Country“ (Position 19 und 20) sind die folgenden Werte möglich:

- 0: Leer
- 1: Das Land wurde nicht erkannt.
- 4: Das Land wurde mithilfe der Einstellung „DefaultCountryISO3“ erkannt.
- 5: Das Land wurde nicht erkannt – Mehrfachübereinstimmungen.
- 6: Das Land wurde anhand des Skripts erkannt.
- 7: Das Land wurde anhand des Formats erkannt.
- 8: Das Land wurde anhand der Großstadt erkannt.
- 9: Das Land wurde anhand der Provinz erkannt.
- C: Das Land wurde anhand des Territoriums erkannt.
- D: Das Land wurde anhand des Namens mit Fehlern erkannt.
- E: Das Land wurde anhand des Namens fehlerfrei erkannt.
- F: Das Land wurde mithilfe der Einstellung „ForceCountryISO3“ erkannt.

„ValidateAddressLoqate“

„ValidateAddressLoqate“ standardisiert und überprüft Adressen mithilfe der Adressdaten von Postbehörden. „ValidateAddress Loqate“ kann Daten korrigieren und die Adresse entsprechend dem von der jeweiligen Postbehörde bevorzugten Format formatieren. Durch „Validate Address Loqate“ werden zudem fehlende postalische Daten wie Postleitzahlen, Ortsnamen, Namen von Bundesländern/Kantonen usw. hinzugefügt.

Des Weiteren gibt „ValidateAddressLoqate“ Ergebnisindikatoren zu Vergleichsversuchen zurück, z. B. ob „ValidateAddressLoqate“ die Adresse überprüft hat, das Zuverlässigkeitsniveau der zurückgegebenen Adresse, den Fehlergrund, wenn die Adresse nicht überprüft werden konnte, usw.

Während des Adressenvergleichs und der Standardisierung trennt „ValidateAddressLoqate“ Adresszeilen in Komponenten und vergleicht sie mit den Inhalten der Datenbanken des Universal Addressing-Moduls. Wenn eine Übereinstimmung ermittelt wird, wird die eingegebene Adresse entsprechend den Informationen in der Datenbank *standardisiert*. Wenn keine Übereinstimmung in der Datenbank gefunden wird, *formatiert* „ValidateAddressLoqate“ die Eingabeadresse (optional). Bei dem Formatierungsvorgang wird versucht, die Adresszeilen gemäß den Standards der jeweiligen Postbehörde zu strukturieren.

„ValidateAddressLoqate“ ist Teil des Universal Addressing-Moduls.

Eingabe

Tabelle 114: Eingabeformat

columnName	Format	Beschreibung
AddressLine1	Zeichenfolge	Die erste Adresszeile.
AddressLine2	Zeichenfolge	Die zweite Adresszeile.
AddressLine3	Zeichenfolge	Die dritte Adresszeile.
AddressLine4	Zeichenfolge	Die vierte Adresszeile.
City	Zeichenfolge	Der Ortsname.
Country	Zeichenfolge	<p>Der Ländercode oder Ländername, in einem der folgenden Formate:</p> <ul style="list-style-type: none"> • Zweistelliger Ländercode gemäß ISO 3116-1 Alpha-2. • Dreistelliger Ländercode gemäß ISO 3116-1 Alpha-3. • Englischer Ländername <p>Eine Liste der ISO-Codes finden Sie unter ISO-Ländercodes und Modulunterstützung auf Seite 594.</p>
FirmName	Zeichenfolge	Der Unternehmens- bzw. Firmenname.
PostalCode	Zeichenfolge	<p>Die Postleitzahl der Adresse in einem der folgenden Formate:</p> <p>99999 99999-9999 A9A9A9 A9A 9A9 9999 999</p>
StateProvince	Zeichenfolge	Das Bundesland oder der Kanton.

Optionen

Die folgende Tabelle enthält die Optionen, die den von „ValidateAddressLoqate“ zurückgegebenen Informationstyp steuern.

Tabelle 115: Ausgabedatenoptionen

optionName	Beschreibung
Database.Loqate	Gibt an, welche Datenbank Sie zum Überprüfen internationaler Adressen verwenden möchten. Wählen Sie für die Angabe einer Datenbank zur Überprüfung internationaler Adressen eine Datenbank aus der Dropdown-Liste Datenbank aus.
OutputFieldLevelReturnCodes	<p>Gibt an, ob Ergebnisindikatoren auf Feldebene eingeschlossen werden sollen. Ergebnisindikatoren auf Feldebene beschreiben, wie „ValidateAddressLoqate“ jedes Adresselement verarbeitet. Ergebnisindikatoren auf Feldebene werden im Begrenzungszeichen „Ergebnis“ zurückgegeben. Der Ergebnisindikator auf Feldebene für „HouseNumber“ ist beispielsweise in HouseNumber.Result enthalten. Eine vollständige Liste der Ausgabefelder für Ergebnisindikatoren finden Sie unter Ergebnisindikatoren auf Seite 539.</p> <p>N Nein, keine Rückgabecodes auf Feldebene zurückgeben (Standard).</p> <p>Y Ja, Rückgabecodes auf Feldebene zurückgeben.</p>

optionName	Beschreibung
OutputFormattedOnFail	<p>Gibt an, ob eine formatierte Adresse zurückgegeben werden soll, wenn eine Adresse nicht überprüft werden kann. Die Adresse wird entsprechend dem durch das Adressland bevorzugten Adressenformat formatiert. Ist diese Option deaktiviert, sind die Ausgabeadressfelder leer, wenn „ValidateAddressLoqate“ die Adresse nicht überprüfen kann.</p> <p>N Nein, fehlgeschlagene Adressen nicht formatieren (Standard).</p> <p>Y Ja, fehlgeschlagene Adressen formatieren.</p> <p>Formatierte Adressen werden im Format zurückgegeben, das durch die Kontrollkästchen Standardadresse einschließen, Adresszeilenelemente einschließen und Postalische Informationen einschließen aktiviert wurde. Beachten Sie, dass bei Auswahl von Adresszeilenelemente einschließen die geparsten Adresselemente die geparste, überprüfte Adresse für Adressen enthalten, die überprüft werden konnten. Wenn die Adresse nicht überprüft werden konnte, enthalten die geparsten Adresselemente die Eingabeadresse in geparster Form. Wenn die Ausgabe immer eine geparste Eingabeadresse enthalten soll, unabhängig davon, ob „ValidateAddressLoqate“ die Adresse überprüfen konnte, müssen Sie die Option Standardisierte Eingabeadresselemente einschließen auswählen.</p> <p>Wenn Sie diese Option aktivieren, müssen Sie Standardadresse einschließen und/oder Adresszeilenelemente einschließen auswählen.</p> <p>Formatierte Adressen werden in dem über die Option OutputRecordType angegebenen Format zurückgegeben. Beachten Sie, dass bei Angabe von OutputRecordType=E die geparsten Adresselemente die geparste, überprüfte Adresse für Adressen enthalten, die überprüft werden konnten. Wenn die Adresse nicht überprüft werden konnte, enthalten die geparsten Adresselemente die Eingabeadresse in geparster Form. Wenn die Ausgabe immer eine geparste Eingabeadresse enthalten soll, unabhängig davon, ob „ValidateAddressLoqate“ die Adresse überprüfen konnte, müssen Sie OutputRecordType=I angeben.</p> <p>Wenn Sie „Y“ angeben, müssen Sie „A“ und/oder „E“ für „OutputRecordType“ angeben.</p> <p>Formatierte Adressen werden in dem über die Option Option.OutputRecordType angegebenen Format zurückgegeben. Beachten Sie, dass bei Angabe von Option.OutputRecordType=E die geparsten Adresselemente die geparste, überprüfte Adresse für Adressen enthalten, die überprüft werden konnten. Wenn die Adresse nicht überprüft werden konnte, enthalten die geparsten Adresselemente die Eingabeadresse in geparster Form. Wenn die Ausgabe immer eine geparste Eingabeadresse enthalten soll, unabhängig davon, ob „ValidateAddressLoqate“ die Adresse überprüfen konnte, geben Sie Option.OutputRecordType=I an.</p>

optionName	Beschreibung
OutputAddressBlocks	<p>Gibt an, ob die Adresse in einem Format zurückgegeben werden soll, das dem Aufdruck auf einer tatsächlichen Postsendung entspricht. Jede Adresszeile wird in einem separaten Adressblockfeld zurückgegeben. Es kann bis zu neun Adressblock-Ausgabefelder geben: AddressBlock1 bis AddressBlock9.</p> <p>Zum Beispiel diese Eingabeadresse:</p> <p>AddressLine1: 4200 Parliament Place AddressLine2: Suite 600 City: Lanham StateProvince: MD PostalCode: 20706</p> <p>Dies führt zu folgender Adressblockausgabe:</p> <p>AddressBlock1: 4200 Parliament PL STE 600 AddressBlock2: Lanham MD 20706-1882 AddressBlock3: UNITED STATES OF AMERICA</p> <p>„ValidateAddressLoqate“ formatiert die Adresse mithilfe von Standards der Postbehörden in Adressblöcke. Der Ländername wird gemäß Universal Postal Union-Ländername zurückgegeben. Beachten Sie, dass die Option OutputCountryFormat keine Auswirkungen auf den Ländernamen im Adressblock hat. Sie wirkt sich nur auf den im Ausgabefeld Land zurückgegebenen Namen aus.</p> <p>Zur Auswahl stehen:</p> <p>N Nein, keine Adressblöcke zurückgeben. Standardeinstellung.</p> <p>Y Ja, Adressblöcke zurückgeben.</p>

optionName	Beschreibung
AmasFormatting	<p>Gibt an, dass Ausgabeadressdaten gemäß den AMAS-Konventionen (AMAS = Address Matching Approval System) formatiert werden müssen.</p> <p>Diese Option bewirkt, dass „Validate Address Loqate“ für die Standardisierung einer Adresse AMAS-Regeln verwendet. AMAS ist ein Programm der Australia Post zur Durchsetzung von Adressenstandards. Weitere Informationen zu den AMAS-Formatierungskonventionen finden Sie im „Address Matching Approval System (AMAS) Handbook“.</p> <p>Diese Option ändert die Ausgabedaten wie folgt.</p> <ul style="list-style-type: none"> • Numerische Felder werden mit Nullen aufgefüllt. Dies hat Auswirkungen auf folgende Ausgabefelder: „HouseNumber“, „HouseNumber2“, „PostalDeliveryNumber“ und „DPID“. Wenn die Eingabeadresse beispielsweise „298 New South Head Rd Double Bay NSW 2028“ lautet, dann wird das Format des Feldes „HouseNumber“ von „298“ in „00298“ geändert. • Wenn keine Übereinstimmung gefunden wird, sind alle Ziffern im Feld „DPID“ Nullen. Beispiel: 00000000. • Wenn keine Übereinstimmung gefunden wird, sind alle zurückgegebenen Felder (geparste Adresselemente) leer. Nur numerische Felder enthalten alle Nullen. • Das Feld „CCD“ wird nicht ausgegeben. <p>Gültige Werte sind:</p> <p>N Nein, Ausgabedaten nicht gemäß AMAS-Konventionen formatieren (Standard).</p> <p>Y Ja, Ausgabedaten gemäß AMAS-Konventionen formatieren.</p> <p>Anmerkung: Wenn diese Option ausgewählt wird, werden Ergebnisse unabhängig von der in den Feldern Akzeptanzniveau und Minimaler Übereinstimmungswert getroffenen Auswahl mit AMAS-Formatierung zurückgegeben.</p>
OutputCasing	<p>Gibt die Groß-/Kleinschreibung der Ausgabedaten an. Zur Auswahl stehen:</p> <p>M Gibt die Ausgabe in Groß- und Kleinbuchstaben zurück (Standard). Beispiel:</p> <p>123 Main St Mytown FL 12345</p> <p>U Gibt die Ausgabe in Großbuchstaben zurück. Beispiel:</p> <p>123 MAIN ST MYTOWN FL 12345</p>

optionName	Beschreibung
HomeCountry	<p>Gibt das Standardland an. Sie sollten das Land angeben, in dem sich die meisten Adressen befinden. Wenn Sie beispielsweise vor allem deutsche Adressen verarbeiten, geben Sie „Germany“ an.</p> <p>„ValidateAddressLoqate“ verwendet das von Ihnen angegebene Land, um eine Überprüfung einzuleiten, wenn es das Land nicht über die Adressfelder „StateProvince“, „PostalCode“ und „Country“ bestimmen kann. Gültige Ländernamen sind:</p> <p>Afghanistan, Albania, Algeria, American Somoa, Andorra, Angola, Anguilla, Antigua And Barbuda, Argentina, Armenia, Aruba, Australia, Austria, Azerbaijan, Bahamas, Bahrain, Bangladesh, Barbados, Belarus, Belgium, Belize, Benin, Bermuda, Bhutan, Bolivia, Bosnia And Herzegovina, Botswana, Brazil, British Virgin Islands, Brunei Darussalam, Bulgaria, Burkina Faso, Burundi, Cambodia, Cameroon, Canada, Cape Verde, Cayman Islands, Central African Republic, Chad, Chile, China, Colombia, Comoros Islands, Congo, Cook Islands, Costa Rica, Cote D'Ivoire, Croatia, Cuba, Cyprus, Czech Republic, Democratic Republic Of Congo, Denmark, Djibouti, Dominica, Dominican Republic, East Timor, Ecuador, Egypt, El Salvador, Equitorial Guinea, Eritrea, Estonia, Ethiopia, Falkland Islands, Faroe Islands, Federated States Of Micronesia, Fiji, Finland, France, French Guiana, Gabon, Gambia, Germany, Ghana, Gibraltar, Greece, Greenland, Grenada, Guadeloupe, Guam, Guatemala, Guinea, Guinea Bissau, Guyana, Haiti, Holy See, Honduras, Hong Kong, Hungary, Iceland, India, Indonesia, Iran, Iraq, Ireland, Israel, Italy, Jamaica, Japan, Jordan, Kazakhstan, Kenya, Kiribati, Korea, Kuwait, Kyrgyzstan, Laos, Latvia, Lebanon, Lesotho, Liberia, Libya, Liechtenstein, Lithuania, Luxembourg, Macau, Macedonia, Madagascar, Malawi, Malaysia, Maldives, Mali, Malta, Marshall Islands, Martinique, Mauritania, Mauritius, Mayotte, Mexico, Moldova, Monaco, Mongolia, Monserrat, Morocco, Mozambique, Myanmar, Namibia, Nauru, Nepal, Netherlands Antilles, New Caledonia, New Zealand, Nicaragua, Niger, Nigeria, Niue, Norway, Oman, Pakistan, Palau, Panama, Papua New Guinea, Paraguay, Peru, Philippines, Pitcairn Islands, Poland, Portugal, Puerto Rico, Qatar, Republic Of Georgia, Republic Of Korea, Republic Of Singapore, Reunion, Romania, Russia, Rwanda, Saint Helena, Saint Kitts And Nevis, Saint Lucia, Saint Pierre And Miquelon, Saint Vincent And The Grenadines, Samoa, San Marino, Sao Tome and Principe, Saudi Arabia, Senegal, Seychelles, Sierra Leone, Slovakia, Slovenia, Solomon Islands, Somalia, South Africa, Spain, Sri Lanka, Sudan, Surivalue, Swaziland, Sweden, Switzerland, Syria, Tahiti, Taiwan, Tajikistan, Tanzania, Thailand, The Netherlands, Togo, Tonga, Trinidad And Tobago, Tristan Da Cunha, Tunisia, Turkey, Turkmenistan, Turks And Caicos Islands, Tuvalu, Uganda, Ukraine, United Arab Emirates, United Kingdom, United States, Uruguay, Uzbekistan, Vanuatu, Venezuela, Vietnam, Virgin Islands (US), Wallis And Futuna, Yemen, Yugoslavia, Zambia, Zimbabwe</p>

optionName	Beschreibung
OutputCountryFormat	<p>Gibt das Format an, das für den im Ausgabefeld Country zurückgegebenen Ländernamen verwendet werden soll. Wenn Sie beispielsweise „Englisch“ auswählen, wird der Ländername „Deutschland“ als „Germany“ zurückgegeben.</p> <p>E Englische Ländernamen verwenden (Standard).</p> <p>I ISO-Abkürzung mit zwei Buchstaben anstelle von Ländernamen für die Länder verwenden.</p> <p>U Abkürzung der Universal Postal Union anstelle von Ländernamen für die Länder verwenden.</p>
OutputScript	<p>Gibt das Alphabet oder die Schrift an, in dem oder in der die Ausgabe zurückgegeben werden soll. Diese Option ist bidirektional und ist im Allgemeinen als von Muttersprache zu Latein oder als von Latein zu Muttersprache festgelegt.</p> <p>Input Es wird keine Transkription durchgeführt. Die Ausgabe erfolgt in derselben Schrift wie die Eingabe (Standard).</p> <p>Native Die Ausgabe erfolgt in der muttersprachlichen Schrift des ausgewählten Landes, falls möglich.</p> <p>Latn Englische Werte werden verwendet.</p>

optionName	Beschreibung
------------	--------------

Akzeptanzniveau	
-----------------	--

AcceptanceLevel	
-----------------	--

optionName

Beschreibung

Gibt die minimale Überprüfungsebene an, die ein Datensatz erreichen muss, um als erfolgreich verarbeitet zu gelten. Der Wert in diesem Feld entspricht dem zweiten Zeichen des „Address Verification Code“, das als „Vergleichsebene von Überprüfungen nach der Verarbeitung“ bezeichnet wird:

- **5:** Zustellort (Gebäude oder Postfach). Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf, wenn die im Eingabedatensatz angegebenen Werte für „ApartmentNumber“, „HouseNumber“, „Street“, „City“ und „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmen. Die Zuverlässigkeit ist mäßig, wenn eine Übereinstimmung mit „ApartmentNumber“ gegeben ist, mit den anderen Feldern jedoch nicht. In diesem Fall müsste die Loqate-Engine „ApartmentNumber“ jedoch differenzierter als „ApartmentNumber“ identifizieren können. Es ist keine Zuverlässigkeit gegeben, wenn „ApartmentNumber“ und andere Felder nicht von der Loqate-Engine geparkt werden können.
- **4:** Einrichtung oder Gebäude. Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf, wenn die im Eingabedatensatz angegebenen Werte für „HouseNumber“, „Street“, „City“ und „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmen. Die Zuverlässigkeit ist mäßig, wenn eine Übereinstimmung mit „HouseNumber“ gegeben ist, mit den anderen Feldern jedoch nicht. In diesem Fall müsste die Loqate-Engine „HouseNumber“ jedoch identifizieren können, da sich „HouseNumber“ auf einer differenzierteren Ebene befindet. Es ist keine Zuverlässigkeit gegeben, wenn „HouseNumber“ und andere Felder nicht von der Loqate-Engine geparkt werden können.
- **3:** Landstraße oder Straße. Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf, wenn die im Eingabedatensatz angegebenen Werte für „Street“, „City“ und „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmen. Die Zuverlässigkeit ist mäßig, wenn eine Übereinstimmung mit „City“ gegeben ist, mit „StateProvince“ jedoch nicht. In diesem Fall müsste die Loqate-Engine „StateProvince“ identifizieren können, da „City“ selbst Teil von „StateProvince“ ist. Es ist keine Zuverlässigkeit gegeben, wenn „City“ oder beide Felder („City“ und „StateProvince“) nicht von der Loqate-Engine geparkt werden können.
- **2:** Lokalität (Stadt oder Ort). Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf, wenn die im Eingabedatensatz angegebenen Werte für „City“ und „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmen. Die Zuverlässigkeit ist mäßig, wenn eine Übereinstimmung mit „City“ gegeben ist, mit „StateProvince“ jedoch nicht. In diesem Fall müsste die Loqate-Engine „StateProvince“ identifizieren können, da „City“ selbst Teil von „StateProvince“ ist. Es ist keine Zuverlässigkeit gegeben, wenn „City“ oder beide Felder („City“ und „StateProvince“) nicht von der Loqate-Engine geparkt werden können.
- **1:** Verwaltungsbereich (Bundesland/Kanton oder Region). Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf,

optionName	Beschreibung
	<p>wenn der im Eingabedatensatz angegebene Wert für „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmt.</p> <ul style="list-style-type: none"> • 0: Keine. Dies entspricht der lockersten Option für die Übereinstimmung.
IsDuplicateHandlingMaskEnable	<p>Aktiviert die Dublettenbehandlungsmaske und gibt an, wie Datensatzdubletten verarbeitet und entfernt werden. Wählen Sie mindestens eine der folgenden Optionen aus:</p> <p>S Ist standardmäßig ausgewählt. Eingabe vorab bearbeiten und Dubletten entfernen, die in einem Feld auftreten.</p> <p>C Ist standardmäßig ausgewählt. Eingabe vorab bearbeiten und Dubletten über alle Felder entfernen.</p> <p>T Eingabe vorab bearbeiten und Dubletten in Feldern entfernen, bei denen es sich nicht um Standardadressfelder handelt.</p> <p>F Ist standardmäßig ausgewählt. Ausgabe der Überprüfung nachbearbeiten und Dubletten aus nicht überprüften Feldern entfernen.</p>
MinimumMatchScore	<p>Gibt einen numerischen Wert zwischen 0 und 100 an, der den Grad angibt, um den „Validate Address Loqate“ eine Adresse ändert, um eine Übereinstimmung in der Loqate-Referenzdatenbank zu erhalten. Je niedriger die Zahl ist, desto mehr Änderungen sind zulässig. Der Wert 100 bedeutet, dass die Eingabeadresse nach dem Parsing fast mit der überprüften Adresse übereinstimmt. Der Wert 0 gibt an, dass die geparste Eingabeadresse vollständig geändert werden kann, um eine überprüfte Adresse zu erhalten.</p>
KeepMultimatch	<p>Gibt an, ob für Eingabeadressen mit mehreren möglichen Übereinstimmungen mehrere Adressen zurückgegeben werden sollen.</p> <p>Y Ja, Mehrfachübereinstimmungen zurückgeben (Standardeinstellung).</p> <p>N Nein, keine Mehrfachübereinstimmungen zurückgeben.</p> <p>Weitere Informationen finden Sie unter Rückgabe von Mehrfachübereinstimmungen auf Seite 530.</p>
FailMultipleMatches	<p>Erklärt bei Eingabeadressen mit mehreren möglichen Übereinstimmungen mehrere Adressen für ungültig.</p>

Rückgabe von Mehrfachübereinstimmungen

Wenn „ValidateAddressLoqate“ mehrere Adressen in der Postdatenbank findet, die mögliche Übereinstimmungen für die Eingabeadresse sind, können Sie die möglichen Übereinstimmungen mithilfe von „ValidateAddressLoqate“ zurückgeben. Die folgende Adresse stimmt mit mehreren Adressen in der US-amerikanischen Postdatenbank überein:

PO BOX 1 New York, NY

Optionen

Verwenden Sie zum Zurückgeben von Mehrfachübereinstimmungen die in der folgenden Tabelle beschriebenen Optionen.

Tabelle 116: Option für Mehrfachübereinstimmung

optionName	Beschreibung/Gültige Werte
KeepMultimatch	<p>Gibt an, ob für Eingabeadressen mit mehreren möglichen Übereinstimmungen mehrere Adressen zurückgegeben werden sollen.</p> <p>Y Ja, Mehrfachübereinstimmungen zurückgeben (Standardeinstellung).</p> <p>N Nein, keine Mehrfachübereinstimmungen zurückgeben.</p>
MaximumResults	<p>Eine Zahl zwischen 1 und 10 ein, die die maximale Anzahl zurückzugebender Adressen angibt. Der Standardwert ist 1.</p> <p>Anmerkung: Der Unterschied zwischen Keepmultimatch = N und KeepMultimatch=Y/MaximumResults=1 ist, dass eine Mehrfachübereinstimmung einen Fehler zurückgibt, wenn KeepMultimatch = N. Dahingegen gibt eine Mehrfachübereinstimmung einen Datensatz zurück, wenn KeepMultimatch = Y und MaximumResults = 1 ist.</p>
OutputFieldLevelReturnCodes	<p>Zur Identifizierung, welche Ausgabeadressen Kandidatenadressen sind, müssen Sie den Wert Y für „OutputFieldLevelReturnCodes“ angeben. Dadurch enthalten Datensätze, die Kandidatenadressen sind, einen oder mehrere M-Werte in den Ergebnisindikatoren auf Feldebene.</p>

Ausgabe

Wenn Sie Mehrfachübereinstimmungen zurückgeben lassen, werden die Adressen in dem von Ihnen festgelegten Adressenformat zurückgegeben. Weitere Informationen zur Festlegung des Adressenformats finden Sie unter [Optionen](#) auf Seite 521. Suchen Sie zur Identifizierung, welche

Datensätze Kandidatenadressen sind, nach mehreren M-Werten in den Ergebnisindikatoren auf Feldebene. Weitere Informationen finden Sie unter [Ergebnisindikatoren](#) auf Seite 539.

Optionen für Grenzwerte von Vergleichspunktzahlen

Für das Festlegen von Grenzwerten von Vergleichspunktzahlen sind zwei Optionen verfügbar.

Anmerkung: Diese Optionen sind nicht in der „Validate Address Loqate“-Benutzerschnittstelle verfügbar, sondern sie befinden sich in folgender Datei:

```
SpectrumDirectory/server/modules/loqate/env.properties
```

Über die Option **MatchScoreAbsoluteThreshold** wird die minimale Übereinstimmungspunktzahl angegeben, die ein Datensatz erreichen muss, um als Kandidat mit einer hohen Übereinstimmung zu gelten. Der minimale Wert beträgt 60 und der maximale Wert 100.

MatchScoreThresholdFactor ist ein Wert, der einen Faktor für das höchste Vergleichsergebnis darstellt. Dieser Wert wird als Grenzwert für die Berücksichtigung von Ergebniskandidaten verwendet. Je höher der Wert des Faktors ist, desto größer ist die Wahrscheinlichkeit, dass ein gutes Überprüfungsergebnis erzielt wird. Der minimale Wert beträgt 95 und der maximale Wert 100.

Ausgabe

Die Ausgabe von „ValidateAddressLoqate“ enthält abhängig von den von Ihnen ausgewählten Ausgabekategorien verschiedene Informationen.

Standardmäßige Adressausgabe

Die standardmäßige Adressausgabe besteht aus vier Adresszeilen, die dem entsprechen, wie die Adresse auf einem Adressticket erscheint. Ort, Bundesland/Kanton, Postleitzahl und weitere Daten sind ebenfalls in der standardmäßigen Adressausgabe enthalten. „ValidateAddressLoqate“ gibt eine standardmäßige Adressausgabe für überprüfte Adressen zurück, wenn Sie **OutputRecordType = A** aktivieren. Standardmäßige Adressfelder werden immer für Adressen zurückgegeben, die unabhängig davon, ob Sie mit **OutputRecordType = A** aktiviert haben, nicht überprüft werden konnten. Für nicht überprüfte Adressen enthalten die standardmäßigen Adressenausgabefelder die Adresse so, wie sie in der Eingabe vorgekommen ist („Durchlauf“-Daten). Wenn Sie möchten, dass „ValidateAddressLoqate“ Adressen gemäß Postbehördenstandards bei fehlgeschlagener Überprüfung standardisiert, und geben Sie in Ihrer Anforderung **OutputFormattedOnFail = Y** an.

Tabelle 117: Standardmäßige Adressausgabe

columnName	Beschreibung
AdditionalInputData	Eingabeadressen, für die keine Übereinstimmung mit einer bestimmten Adresskomponente gefunden werden konnte. Weitere Informationen finden Sie unter Zusätzliche Eingabedaten .

columnName	Beschreibung
AddressLine1-4	Wenn die Adresse überprüft wurde, die erste Zeile der überprüften und standardisierten Adresse. Wenn die Adresse nicht überprüft werden konnte, die erste Zeile der Eingabeadresse ohne Änderungen. Es kann bis zu vier Ausgabefelder mit Adressblöcken geben: „AddressLine1“ bis „AddressLine4“.
City	Der überprüfte Ortsname.
Country	Das Land in dem von Ihnen unter OutputCountryFormat bestimmten Format: <ul style="list-style-type: none"> • ISO-Code • UPU-Code • Englisch
FirmName	Der überprüfte Firmen- oder Unternehmensname.
PostalCode	Der überprüfte ZIP-Code™ oder die überprüfte Postleitzahl.
PostalCode.AddOn	Der vierstellige Zusatzteil des ZIP-Codes™. Im ZIP-Code™ 60655-1844 lautet der vierstellige Zusatzteil 1844.
PostalCode.Base	Der fünfstelligen ZIP-Code™; Beispiel: 20706.
StateProvince	Die überprüfte Abkürzung für den Bundesstaat.

Ausgabe von geparsten Adresselementen

Ausgabeadressen werden im Format für geparste Adressen formatiert, wenn Sie **OutputRecordType = E** festlegen. Wenn Sie möchten, dass „ValidateAddressLoqate“ formatierte Daten im Format für geparste Adressen zurückgibt, wenn die Überprüfung fehlschlägt (also eine normalisierte Adresse), geben Sie **OutputFormattedOnFail = Y** an.

Anmerkung: Wenn Sie möchten, dass „ValidateAddressLoqate“ immer geparste Eingabedaten ungeachtet einer erfolgreichen Überprüfung zurückgibt, geben Sie **OutputRecordType = I** an. Weitere Informationen finden Sie unter [Geparste Eingabe](#) auf Seite 535.

Tabelle 118: Ausgabe von geparsten Adressen

columnName	Beschreibung
AddressBlock1-9	<p>Die „AddressBlock“-Ausgabefelder enthalten eine formatierte Version der standardisierten oder normalisierten Adresse, wie sie auf eine physische Postsendung gedruckt würde. „Validate Address Global“ formatiert die Adresse gemäß Postbehördenstandards in Adressblöcke. Jede Adresszeile wird in einem separaten Adressblockfeld zurückgegeben. Es kann bis zu neun Adressblock-Ausgabefelder geben: AddressBlock1 bis AddressBlock9. Zum Beispiel diese Eingabeadresse:</p> <p>AddressLine1: 4200 Parliament Place AddressLine2: Suite 600 City: Lanham StateProvince: MD PostalCode: 20706</p> <p>Dies führt zu folgender Adressblockausgabe:</p> <p>AddressBlock1: 4200 Parliament PL STE 600 AddressBlock2: Lanham MD 20706-1882</p>
ApartmentLabel	Apartmentbezeichnung (wie STE oder APT); z. B.: 123 E Main St APT 3
ApartmentNumber	Apartmentnummer, z. B.: 123 E Main St APT 3
ApartmentNumber2	<p>Sekundäre Apartmentnummer, z. B.: 123 E Main St APT 3, 4th Floor</p> <p>Anmerkung: In diesem Release ist dieses Feld immer leer.</p>
Building	Beschreibender Name, der eine einzelne Position angibt.
City	Überprüfter Ortsname

columnName	Beschreibung
Country	Land. Das Format wird durch Ihre Auswahl unter OutputCountryFormat bestimmt: <ul style="list-style-type: none"> • ISO-Code • UPU-Code • Englisch
County*	Das kleinste geografische Datenelement innerhalb eines Landes, z. B. USA County
FirmName	Der überprüfte Firmen- oder Unternehmensname
HouseNumber	Hausnummer, z. B.: 123 E Main St Apt 3
LeadingDirectional	Voranstehende Richtungsangabe, z. B.: 123 E Main St Apt 3
POBox	Postfachnummer. Sollte es sich bei der Adresse um eine Landstraßenadresse handeln, wird hier die Postfachnummer der Landstraße angezeigt.
PostalCode	Überprüfte Postleitzahl. Bei US-amerikanischen Adressen ist dies der ZIP-Code.
Principality *	Das größte geografische Datenelement innerhalb eines Landes
StateProvince	Überprüfter Name des Bundeslandes oder Kantons
StreetAlias	Ein alternativer Straßename, der in der Regel nur für einen bestimmten Bereich von Adressen in der Straße gilt. Wenn Sie keine Straßen-Aliasse in der Ausgabe zulassen, wird der „Basis“-Name der Straße in der Ausgabe angezeigt, unabhängig davon, ob ein Alias für die Straße vorhanden ist. Beispiel: 123 E Main St Apt 3

columnName	Beschreibung
StreetName	Straßenname, z. B.: 123 E Main St Apt 3
StreetSuffix	Straßensuffix, z. B.: 123 E Main St Apt 3
Subcity*	Das Datenelement eines kleineren Ballungsraums, abhängig von den Inhalten des Feldes „Lokalität“. Beispiel: Turkish Neighbourhood .
Substreet*	Das Datenelement einer abhängigen Straße oder eines Blocks innerhalb eines Landes. Beispiel: UK Dependent Street .
TrailingDirectional	Nachstehende Richtungsangabe, z. B.: 123 Pennsylvania Ave NW

*Dies ist ein Unterfeld und könnte keine Daten enthalten.

Geparste Eingabe

Die Ausgabe kann die Eingabeadresse im geparsten Format enthalten. Dieser Ausgabetyt wird „geparste Eingabe“ genannt. Geparste Eingabefelder enthalten die Adressdaten, die als Eingabe genutzt wurden, egal ob „Validate Address“ die Adresse überprüft hat. Eine geparste Eingabe unterscheidet sich von geparsten Adresselementen der Ausgabe dahingehend, dass geparste Adresselemente die überprüfte Adresse enthalten, wenn die Adresse überprüft werden konnte, und optional die Eingabeadresse, wenn die Adresse nicht überprüft werden konnte. Die geparste Eingabe enthält immer die Eingabeadresse ungeachtet davon, ob „Validate Address“ die Adresse überprüft hat.

Legen Sie **OutputRecordType = I** fest, um geparste Eingabefelder in die Ausgabe einzuschließen.

Tabelle 119: Geparste Eingabe

columnName	Beschreibung
ApartmentLabel.Input	Apartmentbezeichnung (wie STE oder APT); z. B.: 123 E Main St APT 3

columnName	Beschreibung
ApartmentNumber.Input	Apartmentnummer, z. B.: 123 E Main St APT 3
City.Input	Überprüfter Ortsname
Country.Input	Land. Das Format wird durch Ihre Auswahl unter OutputCountryFormat bestimmt: <ul style="list-style-type: none"> • ISO-Code • UPU-Code • Englisch
County.Input*	Das kleinste geografische Datenelement innerhalb eines Landes, z. B. USA County
FirmName.Input	Der überprüfte Firmen- oder Unternehmensname
HouseNumber.Input	Hausnummer, z. B.: 123 E Main St Apt 3
LeadingDirectional.Input	Voranstehende Richtungsangabe, z. B.: 123 E Main St Apt 3
POBox.Input	Postfachnummer. Sollte es sich bei der Adresse um eine Landstraßenadresse handeln, wird hier die Postfachnummer der Landstraße angezeigt.
PostalCode.Input	Überprüfte Postleitzahl. Bei US-amerikanischen Adressen ist dies der ZIP-Code.
Principality.Input *	Das größte geografische Datenelement innerhalb eines Landes
StateProvince.Input	Überprüfter Name des Bundeslandes oder Kantons

columnName	Beschreibung
StreetAlias.Input	Ein alternativer Straßename, der in der Regel nur für einen bestimmten Bereich von Adressen in der Straße gilt. Wenn Sie keine Straßen-Aliasse in der Ausgabe zulassen, wird der „Basis“-Name der Straße in der Ausgabe angezeigt, unabhängig davon, ob ein Alias für die Straße vorhanden ist. Bei dem Basisnamen handelt es sich um den Namen, der für die gesamte Straße gilt. Beispiel: Wenn „N MAIN ST“ der „StreetName“ ist, enthält das Feld „StreetAlias“ den Eintrag „MAIN“ und der Landstraßentyp, „ST“. würde im Feld „StreetSuffix“ zurückgegeben werden.
StreetName.Input	Straßenname, z. B.: 123 E Main St Apt 3
StreetSuffix.Input	Straßensuffix, z. B.: 123 E Main St Apt 3
Subcity.Input*	Das Datenelement eines kleineren Ballungsraums, abhängig von den Inhalten des Feldes „Lokalität“. Beispiel: Turkish Neighbourhood .
Substreet.Input*	Das Datenelement einer abhängigen Straße oder eines Blocks innerhalb eines Landes. Beispiel: UK Dependent Street .
TrailingDirectional.Input	Nachstehende Richtungsangabe, z. B.: 123 Pennsylvania Ave NW

*Dies ist ein Unterfeld und könnte keine Daten enthalten.

Geocode-Ausgabe

ValidateAddressLoqate gibt den Breitengrad/Längengrad, den Vergleichscode für das Geocoding, abhängige und doppelt abhängige Lokalitäten, abhängige Landstraßen, der Verwaltung unter- und übergeordnete Bereiche sowie die Suchentfernung als Ausgabe zurück. Vergleichscodes geben an, wie gut der Geocoder die Eingabeadresse mit einer bekannten Adresse abgeglichen hat. Zudem beschreiben sie den Gesamtstatus eines Vergleichsversuchs. Codes der Suchentfernung zeigen an, wie nah der Geocode der tatsächlichen physischen Position einer Adresse ist.

Tabelle 120: Ausgabe für „Geocode Address“

columnName	Beschreibung
Geocode.MatchCode	<p>Dieser Zwei-Byte-Wert spiegelt den Status und die Ebene des Geocodevergleichs für eine Adresse wider.</p> <p>Das erste Byte stellt den Geocoding-Status dar und weist einen der folgenden Werte auf:</p> <ul style="list-style-type: none"> A Es wurden mehrere Kandidatengeocodes gefunden, die mit der Eingabeadresse übereinstimmen. Von diesen Geocodes wurde ein Durchschnitt zurückgegeben I Ein Geocode konnte aus der Position der Eingabeadressen in einen Bereich interpoliert werden P Es wurde ein einzelner Geocode gefunden, der mit der Eingabeadresse übereinstimmt U Ein Geocode konnte nicht für die Eingabeadresse generiert werden <p>Das zweite Byte stellt die Ebene des Geocoding-Vergleichs dar und weist einen der folgenden Werte auf:</p> <ul style="list-style-type: none"> 5 Zustellort (Postfach oder Nebengebäude) 4 Einrichtung oder Gebäude 3 Thoroughfare 2 Lokalität 1 Verwaltungsbereich 0 Keine
Latitude	Achtstellige Zahl in Grad und mit einer Genauigkeit von vier Dezimalstellen (im angegebenen Format) berechnet.
Longitude	Achtstellige Zahl in Grad und mit einer Genauigkeit von vier Dezimalstellen (im angegebenen Format) berechnet.
SearchDistance	Der Radius der Genauigkeit in Metern, der die voraussichtliche maximale Entfernung zwischen dem angegebenen Geocode und der tatsächlichen physischen Position angibt. Die Ableitung und Abhängigkeit dieses Feldes basieren auf der Genauigkeit und Abdeckung der zugrunde liegenden Referenzdaten.

Tabelle 121: City/Street/Postal: Zentroid-Vergleichscodes

Element	Vergleichscode
Adresspunkt	P4
Adresspunkt interpoliert	I4
Straßenmittelpunkt	A4/P3
Postleitzahl/Ortszentroid	A3/P2/A2

Anmerkung: Geocode.Match.Code gibt keine zwei Koordinaten für ein Straßensegment zurück (z. B. den Anfang und das Ende eines Straßenabschnitts). Stattdessen wird bei der Eingabe, die zu I3-Rückgabecodes (Interpolation zur Landstraßen- oder Straßenebene, auf der keine Gebäudenummer eingegeben wurde) führt, die vollständige Straße für die Berechnung verwendet.

Ergebnisindikatoren

Ergebnisindikatoren bieten Informationen über die Arten der Verarbeitung, die für eine Adresse ausgeführt werden. Es gibt zwei Typen von Ergebnisindikatoren:

Ergebnisindikatoren auf Datensatzebene

Ergebnisindikatoren auf Datensatzebene stellen Daten zu den Ergebnissen von „ValidateAddressLoqate“ bereit, die für jeden Datensatz verarbeitet werden, z. B. der Erfolg oder Misserfolg eines Übereinstimmungsversuchs, welcher Coder die Adresse verarbeitet hat und andere Details. Die folgende Tabelle enthält die von „ValidateAddressLoqate“ zurückgegebenen Ergebnisindikatoren auf Datensatzebene.

Tabelle 122: Indikatoren auf Datensatzebene

columnName	Beschreibung
Confidence	Das Zuverlässigkeitsniveau, das der zurückgegebenen Adresse zugewiesen ist. Der Bereich reicht von 0 (null) bis 100. Null gibt an, dass ein Fehler vorliegt; 100 gibt an, dass das Zuverlässigkeitsniveau sehr hoch ist und die Übereinstimmungsergebnisse richtig sind. Für Mehrfachübereinstimmungen beträgt die Zuverlässigkeitsebene 0. Informationen zur Berechnung dieser Zahl finden Sie unter Einführung in den „Validate Address Loqate“-Zuverlässigkeitsalgorithmus .

columnName	Beschreibung
CouldNotValidate	<p>Wenn keine Übereinstimmung gefunden wird, die Adresskomponente, die nicht überprüft werden konnte:</p> <ul style="list-style-type: none">• ApartmentNumber• HouseNumber• StreetName• PostalCode• City• Directional• StreetSuffix• Firm• POBoxNumber <p>Anmerkung: In einer durch Kommata getrennten Liste können mehrere Komponenten zurückgegeben werden.</p>
MatchScore	<p>„MatchScore“ ermöglicht die Angabe der Ähnlichkeit zwischen den Eingabedaten und den Referenzdaten mit der höchsten Übereinstimmung. Es unterscheidet sich wesentlich von der Zuverlässigkeit. Die Zuverlässigkeit gibt an, inwieweit die Eingabeadresse zum Erhalt einer Übereinstimmung geändert wurde. Die Bedeutung des Übereinstimmungswertes unterscheidet sich zwischen US-amerikanischen und nicht US-amerikanischen Adressen.</p> <p>Das Feld „int getFieldMatchscore (unit record, const char*)“ stellt einen Dezimalwert zwischen 0 und 100 dar, der die Ähnlichkeit zwischen den identifizierten Eingabedaten und den Referenzdaten mit der höchsten Übereinstimmung widerspiegelt. Das Ergebnis 100 gibt an, dass bei den Eingabedaten nur Änderungen an dem Alias, der Groß-/Kleinschreibung oder diakritische Änderungen vorgenommen wurden. Das Ergebnis 0 gibt an, dass zwischen den Eingabedaten und den Referenzdaten mit der höchsten Übereinstimmung keine Ähnlichkeit besteht.</p> <p>Anmerkung: Die Komponenten „Validate Address Loqate“ und „Advanced Matching-Modul“ verwenden beide das Feld „MatchScore“. Der Wert des Feldes „MatchScore“ in der Ausgabe eines Datenflusses wird von dem letzten Schritt bestimmt, der den Wert verändert, bevor er an einen Ausgabeschritt gesendet wird. Wenn Sie einen Datenfluss mit den Schritten „Validate Address Loqate“ und „Advanced Matching-Modul“ haben und das Ausgabefeld „MatchScore“ für jeden Schritt anzeigen möchten, kopieren Sie den „MatchScore“-Wert in einem Transformer-Schritt in ein anderes Feld. „Validate Address Loqate“ erzeugt beispielsweise ein Ausgabefeld mit dem Namen „MatchScore“. In einem Transformer-Schritt wird anschließend das Feld „MatchScore“ aus „Validate Address Loqate“ in ein Feld mit dem Namen „AddressMatchScore“ kopiert. Wenn ein Vergleichsschritt ausgeführt wird, füllt er das Feld „MatchScore“ mit dem Wert aus dem Vergleich aus und gibt den Wert für „AddressMatchScore“ aus „Validate Address Loqate“ weiter.</p>

columnName	Beschreibung
ProcessedBy	Der Adressen-Coder, der die Adresse verarbeitet hat: LOQATE Der Loqate-Codierer verarbeitet die Adressen.
Status	Meldet den Erfolg oder Fehler des Übereinstimmungsversuchs. Bei Mehrfachübereinstimmungen enthält dieses Feld den Wert „F“ für alle möglichen Übereinstimmungen. null Erfolg F Fehler
Status.Code	Ursache für den Fehler, falls vorhanden. • UnableToValidate
Status.Description	Beschreibung des Problems, wenn eines vorliegt. Address Not Found Dieser Wert wird bei Erfüllung der folgenden Bedingung angezeigt: Status.Code=UnableToValidate.

Ergebnisindikatoren auf Feldebene

Ergebnisindikatoren auf Feldebene beschreiben, wie „ValidateAddressLoqate“ jedes Adresselement verarbeitet. Ergebnisindikatoren auf Feldebene werden im Begrenzungszeichen „Ergebnis“ zurückgegeben. Der Ergebnisindikator auf Feldebene für „HouseNumber“ ist beispielsweise in **HouseNumber.Result** enthalten.

Um Ergebnisindikatoren auf Feldebene zu aktivieren, geben Sie **OutputFieldLevelReturnCodes=Y** an.

Die folgende Tabelle enthält Ergebnisindikatoren auf Feldebene. Wenn ein bestimmtes Feld nicht für eine Adresse gilt, könnte der Ergebnisindikator leer sein.

Tabelle 123: Ergebnisindikatoren auf Feldebene

columnName	Beschreibung
ApartmentLabel.Result	A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.
	C Korrigiert. Nur US-amerikanische und kanadische Adressen.
	F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen.
	P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur US-amerikanische und kanadische Adressen.
	R Die Apartmentbezeichnung ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.
	S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.
	U Nicht verglichen. Gilt nicht für kanadische Adressen.
ApartmentNumber.Result	V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.
	A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.
	C Korrigiert. Nur kanadische Adressen.
	F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.
	P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. US-amerikanische Adressen, die eine EWS-Übereinstimmung sind, haben den Wert P. Nur US-amerikanische und kanadische Adressen.
	R Die Apartmentnummer ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.
	S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische Adressen.
U Nicht verglichen.	
V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.	

columnName	Beschreibung
City.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische und kanadische Adressen.</p> <p>F Bindestriche fehlen oder Interpunktionsfehler. Nur kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten.</p> <p>R Der Ort ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische Adressen.</p> <p>U Nicht verglichen. Gilt nicht für kanadische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>
Country.Result	<p>Diese Ergebniscodes gelten nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.</p> <p>U Nicht verglichen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>
County.Result*	Das kleinste geografische Datenelement innerhalb eines Landes, z. B. USA County

columnName	Beschreibung
FirmName.Result	C Korrigiert. Nur US-amerikanische Adressen.
	P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur US-amerikanische und kanadische Adressen.
	U Nicht verglichen. Nur US-amerikanische und kanadische Adressen.
	V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert. Nur US-amerikanische Adressen.
HouseNumber.Result	A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur kanadische Adressen.
	C Korrigiert. Nur kanadische Adressen.
	F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.
	O Außerhalb des Bereichs. Gilt nicht für US-amerikanische oder kanadische Adressen.
	P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.
	R Die Hausnummer ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur kanadische Adressen.
	S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische oder kanadische Adressen.
	U Nicht verglichen.
V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.	

columnName	Beschreibung
LeadingDirectional.Result	A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.
	C Korrigiert. Nicht leere Eingabe wurde in einen nicht leeren Wert korrigiert. Nur US-amerikanische Adressen.
	F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.
	M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.
	P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.
	S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.
	U Nicht verglichen.
	V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert. Gilt nicht für kanadische Adressen.
POBox.Result	A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur kanadische Adressen.
	C Korrigiert. Nur kanadische Adressen.
	F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.
	M Mehrfachübereinstimmungen. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.
	P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.
	R Die Postfachnummer ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.
	S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.
	V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.

columnName	Beschreibung
PostalCode.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische und kanadische Adressen.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Gilt nicht für kanadische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Gilt nicht für US-amerikanische Adressen.</p> <p>R Die Postleitzahl ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>U Nicht verglichen. Wenn der Straßenname z. B. nicht mit der Postleitzahl übereinstimmt, enthalten „StreetName.Result“ und „PostalCode.Result“ den Wert U.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>
PostalCode.Type	<p>P Der ZIP-Code™ enthält nur Postfachadressen. Nur US-amerikanische Adressen.</p> <p>U Der ZIP-Code™ ist ein eindeutiger ZIP-Code™, der einem bestimmten Unternehmen oder Standort zugewiesen ist. Nur US-amerikanische Adressen.</p> <p>M Der ZIP-Code™ gilt für Militäradressen. Nur US-amerikanische Adressen.</p> <p>null Der ZIP-Code™ ist ein standardmäßiger ZIP-Code™.</p>
Principality.Result *	Das größte geografische Datenelement innerhalb eines Landes

columnName	Beschreibung
StateProvince.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur US-amerikanische und kanadische Adressen.</p> <p>R Der Bundesstaat ist erforderlich, fehlt jedoch in der Eingabeadresse. Nur US-amerikanische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Gilt nicht für US-amerikanische Adressen.</p> <p>U Nicht verglichen. Gilt nicht für kanadische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>
StreetAlias.Result	<p>Ein alternativer Name für eine Straße, der in der Regel nur für einen bestimmten Bereich von Adressen in der Straße gilt. Wenn Sie keine Straßen-Aliasse in der Ausgabe zulassen, wird der „Basis“-Name der Straße in der Ausgabe angezeigt, unabhängig davon, ob ein Alias für die Straße vorhanden ist. Bei dem Basisnamen handelt es sich um den Namen, der für die gesamte Straße gilt. Beispiel: Wenn „N MAIN ST“ der „StreetName“ ist, enthält das Feld „StreetAlias“ den Eintrag „MAIN“ und der Landstraßentyp, „ST“. würde im Feld „StreetSuffix“ zurückgegeben werden.</p>

columnName	Beschreibung
StreetName.Result	A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur kanadische Adressen.
	C Korrigiert. Nur US-amerikanische und kanadische Adressen.
	F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.
	M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.
	P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Gilt nicht für US-amerikanische Adressen.
	S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen. Nur US-amerikanische und kanadische Adressen.
	U Nicht verglichen.
	V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.
StreetSuffix.Result	A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.
	C Korrigiert. Nur US-amerikanische und kanadische Adressen.
	F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.
	M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.
	P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.
	S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.
	U Nicht verglichen. Gilt nicht für US-amerikanische Adressen.
	V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.
Subcity.Result*	Das Datenelement eines kleineren Ballungsraums, abhängig von den Inhalten des Feldes „Lokalität“. Beispiel: Turkish Neighbourhood .

columnName	Beschreibung
Substreet.Result*	Das Datenelement einer abhängigen Straße oder eines Blocks innerhalb eines Landes. Beispiel: UK Dependent Street .
TrailingDirectional.Result	<p>A Angefügt. Das Feld wurde einem leeren Eingabefeld hinzugefügt. Nur US-amerikanische und kanadische Adressen.</p> <p>C Korrigiert. Nur US-amerikanische und kanadische Adressen.</p> <p>F Formatiert. Die Leerzeichen und/oder Interpunktion wurde geändert, um Poststandards zu entsprechen. Gilt nicht für US-amerikanische oder kanadische Adressen.</p> <p>M Mehrere. Die Eingabeadresse stimmt mit mehreren Datensätzen in der postalischen Datenbank überein, und jeder übereinstimmende Datensatz hat einen anderen Wert in diesem Feld. Nur US-amerikanische Adressen.</p> <p>P Durchlauf. Die Daten wurden nicht im Überprüfungsvorgang verwendet, sie wurden jedoch in der Ausgabe erhalten. Nur kanadische Adressen.</p> <p>S Standardisiert. Diese Option beinhaltet standardmäßige Abkürzungen.</p> <p>U Nicht verglichen. Gilt nicht für kanadische Adressen.</p> <p>V Überprüft. Die Daten wurden als korrekt bestätigt und bleiben nach der Eingabe unverändert.</p>

*Dies ist ein Unterfeld und könnte keine Daten enthalten.

Der AVC-Code

Der AVC-Code (AVC = Address Verification Code) ist ein 11-Byte-Code, der aus Genauigkeitsindikatoren für Adressen besteht. Anhand der Codes können Sie die Qualität der Verarbeitungsergebnisse erkennen. Zudem stellen sie Richtlinien darüber bereit, wie die eingegebenen Daten bei Bedarf korrigiert werden können. Jede einzelne Adresse erhält einen eigenen Code. Dieser Code wird automatisch in der Ausgabe Ihres Datenflusses zurückgegeben. Beispiel für einen AVC:

V44-I44-P6-100

AVCs bestehen aus acht Teilen:

- Überprüfungsstatus
- Vergleichsebene von Überprüfungen nach der Verarbeitung
- Vergleichsebene von Überprüfungen vor der Verarbeitung
- Parsing-Status
- Vergleichsebene der Lexikonidentifizierung
- Vergleichsebene der Kontextidentifizierung

- Postcode-Status
- Matchscore

Überprüfungsstatus

Die Ebene, auf der eine Adresse überprüft wurde.

- **V**: Verified (Überprüft). Zwischen den Eingabedaten und einem Datensatz aus den verfügbaren Referenzdaten wurde vollständiger Vergleich durchgeführt. Bei einer einfachen Adressenüberprüfung gilt dieser Code als bestmögliches Ergebnis unter den zurückgegebenen Codes.
- **P**: Partially verified (Teilweise überprüft). Zwischen den Eingabedaten und einem Datensatz aus den verfügbaren Referenzdaten wurde ein partieller Vergleich durchgeführt. Dies könnte bedeuten, dass zu den angegebenen Adressinformationen granulare Daten vorliegen, für die Rückgabe einer vollständigen Überprüfung jedoch zusätzliche Informationen erforderlich sind.
- **A**: Ambiguous (Mehrdeutig). Es liegen mehrere Adressen vor, die mit der Eingabe übereinstimmen könnten.
- **U**: Unable to verify (Überprüfung nicht möglich). Dieser Code wird zurückgegeben, wenn für die Überprüfung einer Adresse keine ausreichenden Informationen vorliegen oder wenn die eingegebene Abfrage nicht lesbar ist. Die Ausgabefelder enthalten die Eingabedaten.
- **R**: Reverted (Zurückgesetzt). Der Datensatz konnte nicht für die angegebene zulässige Mindeststufe überprüft werden. Dies kommt vor, wenn in einem Prozess erweiterte Optionen wie Mindestumkehrungsebenen festgelegt sind. Die Ausgabefelder enthalten die Eingabedaten.
- **C**: Conflict (Konflikt). Es liegen mehrere hohe Übereinstimmungen mit Referenzdaten mit widersprüchlichen Werten vor.

Vergleichsebene von Überprüfungen nach der Verarbeitung

Die Ebene, auf der Eingabedaten nach der Verarbeitung mit den verfügbaren Referenzdaten übereinstimmen.

- **5**: Zustellort (Gebäude oder Postfach). Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf, wenn die im Eingabedatensatz angegebenen Werte für „ApartmentNumber“, „HouseNumber“, „Street“, „City“ und „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmen. Die Zuverlässigkeit ist mäßig, wenn eine Übereinstimmung mit „ApartmentNumber“ gegeben ist, mit den anderen Feldern jedoch nicht. In diesem Fall müsste die Loqate-Engine „ApartmentNumber“ jedoch differenzierter als „ApartmentNumber“ identifizieren können. Es ist keine Zuverlässigkeit gegeben, wenn „ApartmentNumber“ und andere Felder nicht von der Loqate-Engine geparkt werden können.
- **4**: Einrichtung oder Gebäude. Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf, wenn die im Eingabedatensatz angegebenen Werte für „HouseNumber“, „Street“, „City“ und „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmen. Die Zuverlässigkeit ist mäßig, wenn eine Übereinstimmung mit „HouseNumber“ gegeben ist, mit den anderen Feldern jedoch nicht. In diesem Fall müsste die Loqate-Engine „HouseNumber“ jedoch identifizieren können, da sich „HouseNumber“ auf einer differenzierteren Ebene befindet. Es ist keine Zuverlässigkeit gegeben, wenn „HouseNumber“ und andere Felder nicht von der Loqate-Engine geparkt werden können.

- **3:** Landstraße oder Straße. Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf, wenn die im Eingabedatensatz angegebenen Werte für „Street“, „City“ und „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmen. Die Zuverlässigkeit ist mäßig, wenn eine Übereinstimmung mit „City“ gegeben ist, mit „StateProvince“ jedoch nicht. In diesem Fall müsste die Loqate-Engine „StateProvince“ identifizieren können, da „City“ selbst Teil von „StateProvince“ ist. Es ist keine Zuverlässigkeit gegeben, wenn „City“ oder beide Felder („City“ und „StateProvince“) nicht von der Loqate-Engine geparkt werden können.
- **2:** Lokalität (Stadt oder Ort). Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf, wenn die im Eingabedatensatz angegebenen Werte für „City“ und „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmen. Die Zuverlässigkeit ist mäßig, wenn eine Übereinstimmung mit „City“ gegeben ist, mit „StateProvince“ jedoch nicht. In diesem Fall müsste die Loqate-Engine „StateProvince“ identifizieren können, da „City“ selbst Teil von „StateProvince“ ist. Es ist keine Zuverlässigkeit gegeben, wenn „City“ oder beide Felder („City“ und „StateProvince“) nicht von der Loqate-Engine geparkt werden können.
- **1:** Verwaltungsbereich (Bundesland/Kanton oder Region). Der Datensatz wird übergeben oder weist eine hohe Zuverlässigkeit auf, wenn der im Eingabedatensatz angegebene Wert für „StateProvince“ mit dem Loqate-Referenzdataset übereinstimmt.
- **0:** Keine. Dies entspricht der lockersten Option für die Übereinstimmung.

Vergleichsebene von Überprüfungen vor der Verarbeitung

Die Ebene, auf der Eingabedaten vor der Verarbeitung mit den verfügbaren Referenzdaten übereinstimmen.

- **5:** Zustellort (Gebäude oder Postfach)
- **4:** Einrichtung oder Gebäude.
- **3:** Landstraße oder Straße.
- **2:** Lokalität (Stadt oder Ort).
- **1:** Verwaltungsbereich (Bundesland/Kanton oder Region).
- **0:** Keine.

Parsing-Status

Die Ebene, auf der eine Adresse geparkt wurde.

- **I:** Identifiziert und geparkt. Die Eingabedaten wurden identifiziert und in Komponenten platziert. Bei „123 Kingston Av“ könnte „Validate Address Loqate“ beispielsweise bestimmen, dass „123“ für die Nummer einer Einrichtung steht, „Kingston“ der Landstraßenname und „Av“ bzw. „Avenue“ der Landstraßentyp ist.
- **U:** Unable to parse (Parsen nicht möglich). „Validate Address Loqate“ konnte die Eingabedaten nicht identifizieren und parsen. Wie bei dem Überprüfungsstatus „Nicht überprüft“ waren die Eingabedaten unvollständig oder ungenau.

Vergleichsebene der Lexikonidentifizierung

Die Ebene, auf der die Eingabedaten durch Verwendung eines Mustervergleichs (Beispiel: ein numerischer Wert könnte die Nummer einer Einrichtung sein) und eines Lexikonvergleichs (Beispiel:

„rd“ konnte für den Landstraßentyp „road“ stehen, „London“ könnte eine Lokalität sein usw.) eine anerkannte Form aufweisen.

- **5:** Zustellort (Gebäude oder Postfach)
- **4:** Einrichtung oder Gebäude.
- **3:** Landstraße oder Straße.
- **2:** Lokalität (Stadt oder Ort).
- **1:** Verwaltungsbereich (Bundesland/Kanton oder Region).
- **0:** Keine.

Vergleichsebene der Kontextidentifizierung

Die Ebene, auf der die Eingabedaten basierend auf dem Kontext, in dem sie erscheinen, erkannt werden können. Dies ist die ungenaueste Vergleichsform, die auf der Identifizierung eines Wortes als bestimmtes Adresselement basiert. Beispiel: Die Eingabe konnte als Landstraße bestimmt werden, da ihr vorangestellt ein Element war, das eine Einrichtung sein könnte, und ihr nachgestellt ein Element, das eine Lokalität sein könnte. Die Elemente wurden durch einen Abgleich mit den Referenzdaten oder dem Lexikon identifiziert.

- **5:** Zustellort (Gebäude oder Postfach)
- **4:** Einrichtung oder Gebäude.
- **3:** Landstraße oder Straße.
- **2:** Lokalität (Stadt oder Ort).
- **1:** Verwaltungsbereich (Bundesland/Kanton oder Region).
- **0:** Keine.

Postcode-Status

Die Ebene, auf der eine Postleitzahl verifiziert wurde.

- **P8:** „PostalCodePrimary“ und „PostalCodeSecondary“ wurden verifiziert.
- **P7:** „PostalCodePrimary“ wurde verifiziert, „PostalCodeSecondary“ wurde hinzugefügt oder geändert.
- **P6:** „PostalCodePrimary“ wurde verifiziert.
- **P5:** „PostalCodePrimary“ wurde mit einer kleinen Änderung verifiziert.
- **P4:** „PostalCodePrimary“ wurde mit einer umfangreichen Änderung verifiziert.
- **P3:** „PostalCodePrimary“ wurde hinzugefügt.
- **P2:** „PostalCodePrimary“ wurde nach Lexikon identifiziert.
- **P1:** „PostalCodePrimary“ wurde nach Kontext identifiziert.
- **P0:** „PostalCodePrimary“ ist leer.

Punktzahl

Ein numerischer Wert zwischen 0 und 100, der die Ähnlichkeit zwischen den identifizierten Eingabedaten und den Ausgabedaten für den Datensatz darstellt. Das Ergebnis 100 bedeutet, dass nur Elemente hinzugefügt wurden oder Änderungen an dem Alias, der Groß-/Kleinschreibung oder

diakritische Änderungen vorgenommen wurden. Das Ergebnis 0 bedeutet, dass zwischen dem Eingabedatenelement und den bereitgestellten Ausgabedaten keine Ähnlichkeit besteht.

AMAS-Ausgabe

Die folgende Tabelle enthält die von „ValidateAddressAUS“ ausgegebenen Standardfelder.

Tabelle 124: Ausgabefelder

columnName	Beschreibung
Barcode	Standardmäßiger Barcode basierend auf dem DPID. F Fehler (kein Barcode gefunden) 20-stellige Zahl Erfolg
DPID	Die Zustellorts-ID. Eine achtstellige Zahl aus der „Postal Address File“ der Australia Post, mit der ein Zustellort für Briefe, z. B. eine Straßenanschrift, eindeutig identifiziert werden kann. Anmerkung: Bei australischen Adressen ohne AMAS-Verifizierung enthält dieses Feld „00000000“ und bei Adressen außerhalb Australiens ist das Feld leer.
FloorNumber	Die Etagen-/Stockwerksnummer, z. B.: 123 E Main St Apt 3, 4th Floor
FloorType	Der Etagen-/Stockwerkstyp, z. B.: 123 E Main St Apt 3, 4th Floor
PostalBoxNum	Die Nummer für die postalische Zustellung, z. B.: PO Box 42

Auftauchen von falsch-positiven Elementen

Was ist ein falsch-positives Element?

Zur Verhinderung der Erstellung von Adresslisten enthalten die DPV- und die LACS^{Link}-Datenbanken falsch-positive Datensätze. Diese Datensätze sind künstlich erzeugte Adressen in einer falsch-positiven Tabelle. Für jede negative Antwort, die auf eine DPV- oder LACS^{Link}-Abfrage folgt, erfolgt eine Abfrage in der falsch-positiven Tabelle. Eine Übereinstimmung in dieser Tabelle

(falsch-positive Übereinstimmung genannt) deaktiviert Ihren DPV- oder LACS^{Link}-Schlüssel. Bei der Batch-Verarbeitung wird der Auftrag, der den Verstoß enthält, erfolgreich abgeschlossen, doch können Sie keinen weiteren Aufträge, die DPV oder LACS^{Link} verwenden, ausführen, bis Sie den Verstoß melden und einen Schlüssel erhalten, um DPV oder LACS^{Link} zu reaktivieren.

Anmerkung: Auch der Begriff „Ausgangsdatensatz-Verstoß“ wird verwendet, wenn falsch-positive Datensätze verarbeitet werden. Diese beiden Begriffe sind synonym.

Melden eines falsch-positiven DPV-Verstoßes

Spectrum™ Technology Platform zeigt eine falsch-positive Übereinstimmung über Nachrichten im Serverprotokoll an.

Wenn eine falsch-positive Übereinstimmung auftritt, rufen Client/Server-Aufrufe einen Ausnahmefehler hervor. Wenn ein falsch-positiver DPV-Datensatz-Verstoß auftritt, wird Folgendes im Serverprotokoll verzeichnet:

```
WARN [Log] Seed record violation for S<ZIP, ZIP+4, Address, Unit> ERROR
[Log] Feature Disabled: DPU: DPV Seed Record Violation. Seed Code:
S<Address, ZIP, ZIP+4, Unit>
```

Anmerkung: Wenn ein falsch-positiver DPV-Datensatz gefunden wird, ruft die Methode „process()“ (COM, C++, Java und .NET) einen Ausnahmefehler mit der Angabe hervor, dass das Feature DPV deaktiviert wurde. In C gibt die Funktion „processMessage()“ einen Nicht-Null-Wert zurück.

Sie können den Verstoß melden und einen Neustartschlüssel erhalten, indem Sie die folgenden Schritte ausführen.

1. Navigieren Sie in Ihrem Browser zu „<http://<yourserver>:<port>/<product code>/dpv.jsp>“. Verwenden Sie beispielsweise „<http://localhost:8080/unc/dpv.jsp>“ für das Universal Addressing-Modul und „<http://localhost:8080/geostan/dpv.jsp>“ für das Enterprise Geocoding-Modul.
2. Füllen Sie die Felder mit den Informationen zum Versender aus. Die Nummer in Klammern nach jedem Feldnamen gibt die maximale Länge des Feldes an.
3. Klicken Sie auf **Absenden**, wenn Sie fertig sind. Ein Dialogfeld **Datei herunterladen** wird angezeigt.
4. Klicken Sie auf **Speichern**, um die Datei auf Ihrem Computer zu speichern. Ein Dialogfeld **Speichern unter** wird angezeigt.
5. Geben Sie einen Dateinamen und einen Speicherort auf Ihrer lokalen Festplatte an (beispielsweise `c:\DPVSeedFile.txt`) und klicken Sie auf **Speichern**.
6. Navigieren Sie zu www.g1.com/support und melden Sie sich an.
7. Klicken Sie auf **DPV & LACS^{Link} False Positive**.
8. Folgen Sie den Anweisungen auf dem Bildschirm, um Ihre Ausgangsdatei anzuhängen und einen Neustartschlüssel zu erhalten.

Layout der falsch-positiven DPV-Header-Datei

Der USPS® hat das erforderliche Layout der falsch-positiven DPV-Header-Datei festgelegt. Derzeit ist sie als Datei mit fester Länge definiert, die zwei oder mehr Datensätze mit 180 Byte enthält. Der erste Datensatz muss immer ein Header-Datensatz sein. Das Layout sieht wie folgt aus.

Tabelle 125: Layout des falsch-positiven DPV-Header-Datensatzes

Position	Länge	Beschreibung	Format
1-40	40	Unternehmensname des Versenders	Alphanumerisch
41-98	58	Adresszeile des Versenders	Alphanumerisch
99-126	28	Ortsname des Versenders	Alphanumerisch
127-128	2	Abkürzung des Bundesstaates des Versenders	Alphabetisch
129-137	9	9-stelliger PLZ-Code des Versenders	Numerisch
138-146	9	Insgesamt verarbeitete Datensätze	Numerisch
147-155	9	Summe der Übereinstimmungen mit DPV-Datensätzen	Numerisch
156-164	9	Übereinstimmungsrate mit DSF in Prozent	Numerisch
165-173	9	Übereinstimmungsrate mit ZIP + 4® in Prozent	Numerisch
174-178	5	Anzahl der PLZ-Codes in der Datei	Numerisch
179-180	2	Anzahl der falsch-positiven Elemente	Numerisch

Der nachfolgende Datensatz enthält Informationen zur falsch-positiven DPV-Übereinstimmung. Für jede falsch-positive DPV-Übereinstimmung muss ein nachfolgender Datensatz zur falsch-positiven Datei hinzugefügt werden. Das Layout ist unten dargestellt.

Tabelle 126: Layout des nachfolgenden falsch-positiven DPV-Datensatzes

Position	Länge	Beschreibung	Format
1-2	2	Präfixrichtung der Straße	Alphanumerisch
3-30	28	Straßenname	Alphanumerisch
31-34	4	Abkürzung für Straßensuffix	Alphanumerisch
35-36	2	Postfixrichtung der Straße	Alphanumerisch
37-46	10	Primäre Nummer der Adresse	Alphanumerisch
47-50	4	Sekundäre Abkürzung der Adresse	Alphanumerisch
51-58	8	Sekundäre Nummer der Adresse	Numerisch
59-63	5	Übereinstimmender PLZ-Code	Numerisch
64-67	4	Übereinstimmende ZIP + 4®	Numerisch
68-180	113	Auffüllung	Leerzeichen

Melden eines falsch-positiven LACS/Link-Verstoßes

Spectrum™ Technology Platform zeigt eine falsch-positive Übereinstimmung über Nachrichten im Serverprotokoll an. Wenn eine falsch-positive Übereinstimmung auftritt, schlagen Batchaufträge fehl und Client/Server-Aufrufe rufen einen Ausnahmefehler hervor.

Anmerkung: Auch der Begriff „Ausgangsdatensatz-Verstoß“ wird verwendet, wenn falsch-positive Datensätze verarbeitet werden. Diese beiden Begriffe sind synonym.

Wenn ein falsch-positiver Datensatz gefunden wird, wird Folgendes im Serverprotokoll verzeichnet:

```
2005-05-06 17:05:38,978 WARN [com.g1.component.ValidateAddress] Seed
record violation for RR 2 28562 31373
2005-05-06 17:05:38,978 ERROR [com.g1.component.ValidateAddress] Feature
Disabled: LLU: LACS Seed Record Violation. Seed Code: 28562 31373
2005-05-06 17:05:38,978 ERROR [com.g1.dcg.gateway.Gateway] Gateway
exception: com.g1.dcg.stage.StageException:
com.g1.dcg.component.ComponentException: Feature Disabled: LLU
2005-05-06 17:06:30,291 ERROR
[com.pb.spectrum.platform.server.runtime.core.license.impl.policy.Policy]
Feature LACSLink Real-time is disabled.
```

Anmerkung: Wenn ein falsch-positiver LACS^{Link}-Datensatz gefunden wird, ruft die Methode „process()“ (COM, C++, Java und .NET) einen Ausnahmefehler mit der Angabe hervor, dass das Feature LLU deaktiviert wurde. In C gibt die Funktion „processMessage()“ einen Nicht-Null-Wert zurück.

1. Navigieren Sie in Ihrem Browser zu „<http://<ServerName>:<port>/<product code>/lacslink.jsp>“. Verwenden Sie beispielsweise „<http://localhost:8080/unc/lacslink.jsp>“ für das Universal Addressing-Modul und „<http://localhost:8080/geostan/lacslink.jsp>“ für das Enterprise Geocoding-Modul.
2. Füllen Sie die Felder mit den Informationen zum Versender aus. Die Nummer in Klammern nach dem Feldnamen gibt die maximale Länge des Feldes an. Klicken Sie auf **Absenden**, wenn Sie fertig sind. Ein Dialogfeld **Datei herunterladen** wird angezeigt.
3. Klicken Sie auf **Speichern**, um die Datei auf Ihrem Computer zu speichern. Ein Dialogfeld **Speichern unter** wird angezeigt.
4. Geben Sie einen Dateinamen und einen Speicherort auf Ihrer lokalen Festplatte an (beispielsweise `c:\lacslink.txt`) und klicken Sie auf **Speichern**.
5. Navigieren Sie zu www.g1.com/support und melden Sie sich an.
6. Klicken Sie auf **DPV & LACS^{Link} False Positive**.
7. Folgen Sie den Anweisungen auf dem Bildschirm, um Ihre Ausgangsdatei anzuhängen und einen Neustartschlüssel zu erhalten.

„ValidateAddress“-Zuverlässigkeitsalgorithmus

Einführung in den „Validate Address“-Zuverlässigkeitsalgorithmus

„Validate Address“ berechnet für jede überprüfte Adresse eine Zuverlässigkeitspunktzahl. Diese Punktzahl beschreibt, wie wahrscheinlich es ist, dass die überprüfte Adresse korrekt ist.

Zuverlässigkeitscodewerte reichen von 0 bis 100, wobei eine Zuverlässigkeit von 0 für keine Zuverlässigkeit und eine Zuverlässigkeit von 100 für eine sehr hohe Zuverlässigkeit steht, dass die Übereinstimmungsergebnisse korrekt sind. Zuverlässigkeitscodes werden basierend auf einem Algorithmus berechnet, der die Übereinstimmungsergebnisse für einzelne Ausgabefelder berücksichtigt. Die in diese Berechnung einbezogenen Ausgabefelder lauten wie folgt:

- Country
- City
- State
- PostalCode
- StreetName
- HouseNumber
- LeadingDirectional
- TrailingDirectional
- StreetSuffix
- ApartmentNumber

Jedes Feld besitzt seine eigene Gewichtung im Algorithmus. Zusätzlich kann für jedes Feld das Übereinstimmungsergebnis mit „Erfolg“, „Fehler“ oder „Geändert“ gekennzeichnet werden. („Geändert“ bezieht sich auf Fälle, bei denen die Inhalte des Feldes zum Erzielen einer Übereinstimmung geändert wurden.) Das Übereinstimmungsergebnis (Erfolg, Fehler oder Geändert) bestimmt, welcher der Faktoren für dieses Feld gilt. Folglich ist die Berechnung des Zuverlässigkeitscodes ein Produkt von Gewichtung und Faktor wie folgt:

```
Confidence = (Weight * Factor) for City
+ (Weight * Factor) for Country
+ (Weight * Factor) for State
+ (Weight * Factor) for PostalCode
+ (Weight * Factor) for StreetName
+ (Weight * Factor) for HouseNumber
+ (Weight * Factor) for Directionals
+ (Weight * Factor) for Street Suffix
+ (Weight * Factor) for ApartmentNumber
```

Zuverlässigkeitsalgorithmus für US-amerikanische und kanadische Adressen

Die folgende Tabelle enthält die Bewertung und Logik hinter dem „Validate Address“-Zuverlässigkeitsalgorithmus für US-amerikanische und kanadische Adressen.

Tabelle 127: Zuverlässigkeitsalgorithmus für US-amerikanische und kanadische Adressen

Feld	Gewichtung/MatchScore	Faktor bei Änderung ¹	Faktor bei Auffüllung ²
Country	10	100 %	0 %
City	10	50 %	75 %
State	15	50 %	75 %
PostalCode	15	25 %	25 %
StreetName	15	50 %	75 %
HouseNumber	15	50 %	75 %
Directionals	10	50 %	75 %
StreetSuffix	5	50 %	75 %
ApartmentNumber	5	50 %	75 %

Zuverlässigkeitsalgorithmus für internationale Adressen

Es gibt zwei Zuverlässigkeitsalgorithmen für Adressen außerhalb der USA und Kanada: einer für Adressen in Ländern, die Postleitzahlen verwenden, und ein anderer für Adressen in Ländern, die keine Postleitzahlen verwenden.

Die folgende Tabelle enthält den Zuverlässigkeitsalgorithmus für Adressen außerhalb der USA und Kanada, für die Postleitzahlen verwendet werden.

² Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld nicht vorhanden waren und aufgefüllt wurden, um eine Übereinstimmung zu erzielen.

¹ Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld geändert wurden, um eine Übereinstimmung zu erzielen.

Tabelle 128: Zuverlässigkeitsalgorithmus für Länder mit Postleitzahlen

Feld	Gewichtung/MatchScore	Faktor bei Änderung ³	Faktor bei Auffüllung ⁴	Faktor bei nicht verfügbaren Postdaten
Land	11,11111111111111	100 %	0 %	0 %
City	11,11111111111111	50 %	75 % ⁵	0 %
State	16,66666666666667	100 %	100	80 %
PostalCode	16,66666666666667	100 %	100 %	80 %
StreetName	16,66666666666667	50 %	75 %	50 %
HouseNumber	16,66666666666667	50 %	75 %	50 %
Richtungsangaben	0	50 %	75 %	0 %
StreetSuffix	5,55555555555556	50 %	75 %	50 %

⁴ Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld nicht vorhanden waren und aufgefüllt wurden, um eine Übereinstimmung zu erzielen.

³ Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld geändert wurden, um eine Übereinstimmung zu erzielen.

⁵ Wenn das Land ein Land der Kategorie C ist, beträgt dieser Wert 50 %. Länder fallen in eine dieser Kategorien:

- **Kategorie A:** Ermöglicht die Überprüfung und Korrektur von Postleitzahl, Ortsname, Bundesland/Bezirk/Kanton, Straßenanschriftselementen und Ländername einer Adresse.
- **Kategorie A:** Ermöglicht die Überprüfung und Korrektur von Postleitzahl, Ortsname, Bundesland/Bezirk/Kanton und Ländername einer Adresse. Sie unterstützt nicht die Überprüfung oder Korrektur von Straßenanschriftselementen.
- **Kategorie C:** Ermöglicht die Überprüfung und Korrektur des Ländernamens sowie die Überprüfung des Formats der Postleitzahl.

Feld	Gewichtung/MatchScore	Faktor bei Änderung ³	Faktor bei Auffüllung ⁴	Faktor bei nicht verfügbaren Postdaten
ApartmentNumber	5,55555555555556	50 %	75 %	50 %

Die folgende Tabelle enthält den Zuverlässigkeitsalgorithmus für Länder, die keine Postleitzahlen verwenden.

Tabelle 129: Zuverlässigkeitsalgorithmus für Länder ohne Postleitzahlen

Feld	Gewichtung/MatchScore	Faktor bei Änderung ⁶	Faktor bei Auffüllung ⁷	Faktor bei nicht verfügbaren Postdaten
Land	13,3333333333333	100 %	0 %	0 %
City	13,3333333333333	50 %	75 % ⁸	0 %
State	20	100 %	100	80 %

⁴ Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld nicht vorhanden waren und aufgefüllt wurden, um eine Übereinstimmung zu erzielen.

³ Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld geändert wurden, um eine Übereinstimmung zu erzielen.

⁷ Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld nicht vorhanden waren und aufgefüllt wurden, um eine Übereinstimmung zu erzielen.

⁶ Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld geändert wurden, um eine Übereinstimmung zu erzielen.

⁸ Wenn das Land ein Land der Kategorie C ist, beträgt dieser Wert 50 %. Länder fallen in eine dieser Kategorien:

- **Kategorie A:** Ermöglicht die Überprüfung und Korrektur von Postleitzahl, Ortsname, Bundesland/Bezirk/Kanton, Straßenanschriftselementen und Ländername einer Adresse.
- **Kategorie A:** Ermöglicht die Überprüfung und Korrektur von Postleitzahl, Ortsname, Bundesland/Bezirk/Kanton und Ländername einer Adresse. Sie unterstützt nicht die Überprüfung oder Korrektur von Straßenanschriftselementen.
- **Kategorie C:** Ermöglicht die Überprüfung und Korrektur des Ländernamens sowie die Überprüfung des Formats der Postleitzahl.

Feld	Gewichtung/MatchScore	Faktor bei Änderung ⁶	Faktor bei Auffüllung ⁷	Faktor bei nicht verfügbaren Postdaten
StreetName	20	50 %	75 %	50 %
HouseNumber	20	50 %	75 %	50 %
Richtungsangaben	0	50 %	75 %	0 %
StreetSuffix	6,66666666666667	50 %	75 %	50 %
ApartmentNumber	6,66666666666667	50 %	75 %	50 %

Die folgende Tabelle enthält Länder ohne Postleitzahlen.

Tabelle 130: Länder ohne Postleitzahlen

Afghanistan	Albanien	Angola
Anguilla	Bahamas	Barbados
Belize	Benin	Bhutan
Botswana	Burkina Faso	Burundi
Kamerun	Caymaninseln	Zentralafrikanische Rep.

⁷ Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld nicht vorhanden waren und aufgefüllt wurden, um eine Übereinstimmung zu erzielen.

⁶ Bezieht sich auf Instanzen, wenn die Eingabedaten in diesem Feld geändert wurden, um eine Übereinstimmung zu erzielen.

Tschad	Cocosinseln	Kolumbien
Komoren	Kongo (Dem. Rep.)	Kongo (Rep.)
Elfenbeinküste	Korea (Nord)	Dschibuti
Dominica	Äquatorialguinea	Eritrea
Fidschi	Gabun	Gambia
Ghana	Grenada	Guyana
Irland	Jamaika	Kiribati
Libyen	Malawi	Mali
Mauretanien	Namibia	Nauru
Palau	Panama	Peru
Katar	Ruanda	St. Lucia
St. Vincent / Grenadinen	Samoa	Sao Tome und Principe
Seychellen	Sierra Leone	Surinam
Tansania	Timor	Togo
Tonga	Trinidad und Tobago	Tuvalu

Uganda

Vereinigte Arabische Emirate

Vanuatu

Jemen

Zimbabwe

Universal Name-Modul

OpenNameParser

OpenNameParser schlüsselt Personen- und Unternehmensnamen sowie andere Begriffe im Namensdatenfeld in ihre Komponententeile auf. Diese geparsten Namenselemente sind anschließend für andere automatisierte Vorgänge verfügbar, z. B. Namensvergleiche, die Namensstandardisierung oder die Namenskonsolidierung über mehrere Datensätze.

OpenNameParser führt die folgenden Aktionen aus:

- Es bestimmt den Typ eines Namens, um die Funktion dieses Namens zu beschreiben. Namensentitätstypen lassen sich in zwei größere Gruppen aufteilen: Personennamen und Unternehmensnamen. Innerhalb dieser größeren Gruppen gibt es Untergruppen.
- Bestimmt die Form eines Namens, um herauszufinden, welche Syntax der Parser beim Parsen verwendet sollte. Personennamen liegen üblicherweise in normaler Reihenfolge (Unterschrift) oder in umgekehrter Reihenfolge vor. Unternehmensnamen sind normalerweise hierarchisch sortiert.
- Bestimmt die Komponenten eines Namens und beschriftet sie, um die syntaktische Beziehung jedes Namensteils zum gesamten Namen so zu identifizieren. Die Syntax von Personennamen umfasst neben anderen Teilen von Personennamen Präfixe, Vorname, zweiter Vorname, Nachname, Suffixe und Begriffe zur Kontobeschreibung. Die Syntax für Unternehmensnamen umfasst den Firmennamen und Suffixe.
- Parst verbundene Personen- und Unternehmensnamen, und behält sie entweder als einen Datensatz bei oder teilt sie in mehrere Datensätze auf. Beispiele für verbundene Namen sind „Mr. und Mrs. John Smith“ und „Baltimore Gas & Electric dba Constellation Energy“.
- Parst die Ausgabe als Datensätze oder als Liste.
- Weist eine Parsing-Punktzahl zu, die den Zuverlässigkeitsgrad des Parsing widerspiegelt.

Eingabe

Tabelle 131: Eingabe für Open Name Parser

columnName	Beschreibung								
CultureCode	<p>Die Kultur der Eingabenamensdaten. Die Optionen sind unten aufgelistet.</p> <table border="0"> <tr> <td>Null (empty)</td> <td>Globale Kultur (Standard)</td> </tr> <tr> <td>de</td> <td>Deutsch</td> </tr> <tr> <td>es</td> <td>Spanisch</td> </tr> <tr> <td>ja</td> <td>Japanisch</td> </tr> </table> <p>Anmerkung: Wenn Sie über den Open Parser Domain Editor Ihre eigene Domäne hinzugefügt haben, sind auch die Kulturen und Kulturcodes für diese Domäne gültig.</p>	Null (empty)	Globale Kultur (Standard)	de	Deutsch	es	Spanisch	ja	Japanisch
Null (empty)	Globale Kultur (Standard)								
de	Deutsch								
es	Spanisch								
ja	Japanisch								
Name	Der zu parsende Name. Dieses Feld ist ein Pflichtfeld.								

Optionen

OpenNameParser-Optionen können auf Schritzebene über einen der Spectrum™ Technology Platform-Clients oder zur Laufzeit mit Datenflussoptionen konfiguriert werden.

Parsing-Optionen

Die folgende Tabelle führt die Optionen auf, die das Parsen von Namen steuern.

Tabelle 132: Parsing-Optionen für Open Name Parser

optionName	Beschreibung				
ParseNaturalOrderPersonalNames	<p>Gibt an, ob Namen geparkt werden sollen, die sich in der Reihenfolge Titel, Vorname, zweiter Vorname, Name und Suffix befinden.</p> <table border="0"> <tr> <td>true</td> <td>Personennamen, die sich in normaler Reihenfolge befinden, parsen.</td> </tr> <tr> <td>false</td> <td>Personennamen, die sich in normaler Reihenfolge befinden, nicht parsen.</td> </tr> </table>	true	Personennamen, die sich in normaler Reihenfolge befinden, parsen.	false	Personennamen, die sich in normaler Reihenfolge befinden, nicht parsen.
true	Personennamen, die sich in normaler Reihenfolge befinden, parsen.				
false	Personennamen, die sich in normaler Reihenfolge befinden, nicht parsen.				

optionName	Beschreibung
ParseReverseOrderPersonalNames	<p>Gibt an, ob Namen, bei denen der Nachname zuerst angegeben ist, geparkt werden sollen.</p> <p>true Personennamen, die sich in umgekehrter Reihenfolge befinden, parsen.</p> <p>false Personennamen, die sich in umgekehrter Reihenfolge befinden, nicht parsen.</p>
ParseConjoinedNames	<p>Gibt an, ob verbundene Namen geparkt werden sollen.</p> <p>true Verbundene Namen parsen.</p> <p>false Verbundene Namen nicht parsen.</p>
SplitConjoinedNames	<p>Gibt an, ob Namen, die mehr als eine Einzelperson enthalten, wie <i>Bill & Sally Smith</i>, in mehrere Datensätze aufgeteilt werden sollen.</p> <p>true Verbundene Namen aufteilen.</p> <p>false Verbundene Namen nicht aufteilen.</p>
ParseBusinessNames	<p>Gibt an, ob Unternehmensnamen geparkt werden sollen.</p> <p>true Unternehmensnamen parsen.</p> <p>false Unternehmensnamen nicht parsen.</p>
OutputAsList	<p>Gibt an, ob geparkte Namenselemente in Listenform zurückgegeben werden sollen.</p> <p>true Geparkte Elemente in Listenform zurückgeben.</p> <p>false Geparkte Elemente nicht in Listenform zurückgeben.</p>

optionName	Beschreibung
ShortcutThreshold	<p>Gibt an, wie Leistung gegenüber Qualität ausbalanciert werden soll. Eine höhere Leistung resultiert in einer niedrigeren Qualität bei der Ausgabe. Anders herum resultiert eine höhere Qualität in einer niedrigeren Leistung. Wenn dieser Schwellenwert erreicht ist, wird bei dem Datensatz keine weitere Verarbeitung mehr durchgeführt.</p> <p>Geben Sie einen Wert zwischen 0 und 100 an. Der Standardwert ist 100.</p>

Kulturoptionen

Die folgende Tabelle führt die Optionen auf, die Namenskulturen steuern.

Tabelle 133: Kulturoptionen für Open Name Parser

optionName	Beschreibung
DefaultCulture	<p>Gibt an, welche Kultur(en) Sie in der Parsing-Grammatik einschließen möchten. Die Standardauswahl ist „Globale Kultur“.</p> <p>Gibt in einer durch Kommas getrennten Liste Kulturen über den zweistelligen Kulturcode in Reihenfolge ihrer Priorität an. Um beispielsweise ein Parsen des Namens unter Verwendung der spanischen gefolgt von der japanischen Kultur zu versuchen, müssten Sie Folgendes angeben:</p> <p><code>es, ja, ,</code></p>

Erweiterte Optionen

Die folgende Tabelle führt die erweiterten Optionen für das Parsen von Namen auf.

Tabelle 134: Erweiterte Optionen für Open Name Parser

Option	Beschreibung
NaturalOrderPersonalNamesDomain	Gibt die Domäne an, die beim Parsen von Personennamen in normaler Reihenfolge verwendet wird. Gültige Werte sind die Domänennamen, die im Open Parser Domain Editor im Enterprise Designer definiert wurden.
NaturalOrderPersonalNamesPriority	<p>Geben Sie eine Zahl zwischen 1 und 5 an, um die Priorität der Domäne von Personennamen in normaler Reihenfolge im Verhältnis zu anderen verwendeten Domänen festzulegen. Hier legen Sie die Reihenfolge fest, in der die Parser ausgeführt werden sollen.</p> <p>Ergebnisse werden für die erste Domäne zurückgegeben, die eine höhere Punktzahl erzielt, als in der Option „Abkürzungsschwellenwert“ angegeben ist. Wenn keine Domäne diesen Schwellenwert erreicht, werden Ergebnisse für die Domäne mit der höchsten Punktzahl zurückgegeben. Wenn mehrere Domänen den Schwellenwert gleichzeitig erreichen, erhält die Domäne Priorität, die zuerst ausgeführt wurde (wie durch die Reihenfolge hier festgelegt). Ergebnisse werden dann für diese Domäne zurückgegeben.</p>
ReverseOrderPersonalNamesDomain	Gibt die Domäne an, die beim Parsen von Personennamen in umgekehrter Reihenfolge verwendet wird. Gültige Werte sind die Domänennamen, die im Open Parser Domain Editor im Enterprise Designer definiert wurden.
ReverseOrderPersonalNamesPriority	<p>Geben Sie eine Zahl zwischen 1 und 5 an, um die Priorität der Domäne von Personennamen in umgekehrter Reihenfolge im Verhältnis zu anderen verwendeten Domänen festzulegen. Hier legen Sie die Reihenfolge fest, in der die Parser ausgeführt werden sollen.</p> <p>Ergebnisse werden für die erste Domäne zurückgegeben, die eine höhere Punktzahl erzielt, als in der Option „Abkürzungsschwellenwert“ angegeben ist. Wenn keine Domäne diesen Schwellenwert erreicht, werden Ergebnisse für die Domäne mit der höchsten Punktzahl zurückgegeben. Wenn mehrere Domänen den Schwellenwert gleichzeitig erreichen, erhält die Domäne Priorität, die zuerst ausgeführt wurde (wie durch die Reihenfolge hier festgelegt). Ergebnisse werden dann für diese Domäne zurückgegeben.</p>

Option	Beschreibung
NaturalOrderConjoinedPersonalNamesDomain	<p>Gibt die Domäne an, die beim Parsen von verbundenen Personennamen in normaler Reihenfolge verwendet wird. Gültige Werte sind die Domänennamen, die im Open Parser Domain Editor im Enterprise Designer definiert wurden.</p>
NaturalOrderConjoinedPersonalNamesPriority	<p>Geben Sie eine Zahl zwischen 1 und 5 an, um die Priorität der Domäne von verbundenen Personennamen in normaler Reihenfolge im Verhältnis zu anderen verwendeten Domänen festzulegen. Hier legen Sie die Reihenfolge fest, in der die Parser ausgeführt werden sollen.</p> <p>Ergebnisse werden für die erste Domäne zurückgegeben, die eine höhere Punktzahl erzielt, als in der Option „Abkürzungsschwellenwert“ angegeben ist. Wenn keine Domäne diesen Schwellenwert erreicht, werden Ergebnisse für die Domäne mit der höchsten Punktzahl zurückgegeben. Wenn mehrere Domänen den Schwellenwert gleichzeitig erreichen, erhält die Domäne Priorität, die zuerst ausgeführt wurde (wie durch die Reihenfolge hier festgelegt). Ergebnisse werden dann für diese Domäne zurückgegeben.</p>
ReverseOrderConjoinedPersonalNamesDomain	<p>Gibt die Domäne an, die beim Parsen von verbundenen Personennamen in umgekehrter Reihenfolge verwendet wird. Gültige Werte sind die Domänennamen, die im Open Parser Domain Editor im Enterprise Designer definiert wurden.</p>
ReverseOrderConjoinedPersonalNamesPriority	<p>Geben Sie eine Zahl zwischen 1 und 5 an, um die Priorität der Domäne von verbundenen Personennamen in umgekehrter Reihenfolge im Verhältnis zu anderen verwendeten Domänen festzulegen. Hier legen Sie die Reihenfolge fest, in der die Parser ausgeführt werden sollen.</p> <p>Ergebnisse werden für die erste Domäne zurückgegeben, die eine höhere Punktzahl erzielt, als in der Option „Abkürzungsschwellenwert“ angegeben ist. Wenn keine Domäne diesen Schwellenwert erreicht, werden Ergebnisse für die Domäne mit der höchsten Punktzahl zurückgegeben. Wenn mehrere Domänen den Schwellenwert gleichzeitig erreichen, erhält die Domäne Priorität, die zuerst ausgeführt wurde (wie durch die Reihenfolge hier festgelegt). Ergebnisse werden dann für diese Domäne zurückgegeben.</p>

Option	Beschreibung
BusinessNamesDomain	Gibt die Domäne an, die beim Parsen von Unternehmensnamen verwendet wird. Gültige Werte sind die Domänennamen, die im Open Parser Domain Editor im Enterprise Designer definiert wurden.
BusinessNamesPriority	Geben Sie eine Zahl zwischen 1 und 5 an, um die Priorität der Unternehmensnamen-Domäne im Verhältnis zu anderen verwendeten Domänen festzulegen. Hier legen Sie die Reihenfolge fest, in der die Parser ausgeführt werden sollen. Ergebnisse werden für die erste Domäne zurückgegeben, die eine höhere Punktzahl erzielt, als in der Option „Abkürzungsschwellenwert“ angegeben ist. Wenn keine Domäne diesen Schwellenwert erreicht, werden Ergebnisse für die Domäne mit der höchsten Punktzahl zurückgegeben. Wenn mehrere Domänen den Schwellenwert gleichzeitig erreichen, erhält die Domäne Priorität, die zuerst ausgeführt wurde (wie durch die Reihenfolge hier festgelegt). Ergebnisse werden dann für diese Domäne zurückgegeben.

Konfigurieren von Optionen zur Laufzeit

Optionen von OpenNameParser können zur Laufzeit konfiguriert und weitergeleitet werden, wenn sie als Datenflussoptionen verfügbar gemacht werden. Das ermöglicht Ihnen, die vorhandene Konfiguration durch Zeichenfolgen für Namensparsing im JSON-Format zu überschreiben. Sie können auch Schrittoptionen festlegen, wenn Sie den Auftrag über einen Prozessfluss oder über das Befehlszeilentool „Job Executor“ aufrufen.

So definieren Sie die Optionen von OpenNameParser zur Laufzeit:

1. Öffnen Sie im Enterprise Designer einen Datenfluss, der den „Open Name Parser“-Schritt verwendet.
2. Speichern Sie diesen Datenfluss und machen Sie ihn verfügbar.
3. Öffnen Sie „Bearbeiten > Datenflussoptionen“.
4. Erweitern Sie Open Name Parser in der Tabelle **Datenflussoptionen zu Schritten zuordnen**, und bearbeiten Sie die Optionen nach Bedarf. Aktivieren Sie das Kästchen für die zu bearbeitende Option, und ändern Sie anschließend den Wert in der Dropdown-Liste **Standardwert**.
5. Optional: Ändern Sie den Namen der Optionen im Feld **Optionsbeschriftung**.
6. Klicken Sie zweimal auf **OK**.

Ausgabe

Tabelle 135: Ausgabe für Open Name Parser

columnName	Format	Beschreibung
AccountDescription	Zeichenfolge	Eine Kontobeschreibung, die Teil des Namens ist. Im Beispiel „Mary Jones Account # 12345“ lautet die Kontobeschreibung „Account#12345“.
Names	Zeichenfolge	Ein hierarchisches Feld, das eine Liste von geparsten Elementen enthält. Dieses Feld wird zurückgegeben, wenn Sie in den Parsing-Optionen das Kästchen Ergebnisse als Liste ausgeben aktivieren.
Mit Unternehmensnamen verwandte Felder		
FirmConjunction	Zeichenfolge	Gibt an, dass der Name eines Unternehmens eine Konjunktion wie „d/b/a“ (doing business as – geschäftlich tätig als), „o/a“ (operating as – tätig als) oder „t/a“ (trading as – handelnd als) enthält.
FirmName	Zeichenfolge	Der Name eines Unternehmens. Zum Beispiel: Pitney Bowes.
FirmSuffix	Zeichenfolge	Das Unternehmenssuffix. Zum Beispiel: „Co.“ und „GmbH“
IsFirm	Zeichenfolge	Gibt an, ob der Name eher ein Unternehmen als eine Einzelperson ist.
Mit den Namen von Einzelpersonen verwandte Felder		
Conjunction	Zeichenfolge	Gibt an, dass der Name eine Konjunktion wie „und“, „oder“ oder „&“ enthält.
CultureCode	Zeichenfolge	Die Kulturcodes, die in den Eingabedaten enthalten sind.

columnName	Format	Beschreibung
CultureCodeUsedToParse	Zeichenfolge	<p>Identifiziert die kulturspezifische Grammatik, die zum Parsen der Daten verwendet wurde.</p> <p>Null (empty) Globale Kultur (Standard)</p> <p>de Deutsch</p> <p>es Spanisch</p> <p>ja Japanisch</p> <p>Anmerkung: Wenn Sie über den Open Parser Domain Editor Ihre eigene Domäne hinzugefügt haben, erscheinen in diesem Feld auch die Kulturen und Kulturcodes für diese Domäne.</p>
FirstName	Zeichenfolge	Der Vorname einer Person.
GeneralSuffix	Zeichenfolge	Das allgemeine/berufliche Suffix einer Person. Zum Beispiel Dr. oder PhD.
IsParsed	Zeichenfolge	Gibt an, ob ein Ausgabedatensatz geparkt wurde. Die Werte sind „wahr“ oder „falsch“.
IsPersonal	Zeichenfolge	Gibt an, ob der Name eher eine Einzelperson als ein Unternehmen ist. Die Werte sind „wahr“ oder „falsch“.
IsReverseOrder	Zeichenfolge	Gibt an, ob der Eingabename in umgekehrter Reihenfolge ist. Die Werte sind „wahr“ oder „falsch“.
LastName	Zeichenfolge	Der Nachname einer Person. Beinhaltet den elterlichen Nachnamen.
LeadingData	Zeichenfolge	Information ohne Namen, die vor einem Namen erscheint.
MaturitySuffix	Zeichenfolge	Das Generationssuffix einer Person. Zum Beispiel Jr. oder Sr.
MiddleName	Zeichenfolge	Der zweite Vorname einer Person.

columnName	Format	Beschreibung
Name.	Zeichenfolge	Der persönliche Name oder Firmenname, der in der Eingabe bereitgestellt wurde.
NameScore	Zeichenfolge	Gibt die durchschnittliche Punktzahl von bekannten und unbekannt Tokens für jeden Namen an. Der Wert von „NameScore“ liegt zwischen 0 und 100 wie in der Parsing-Grammatik definiert. 0 wird zurückgegeben, wenn keine Übereinstimmungen zurückgegeben werden.
SecondaryLastName	Zeichenfolge	In der spanischen Parsing-Grammatik der Nachname der Mutter einer Person.
TitleOfRespect	Zeichenfolge	Eine Information, die vor einem Namen erscheint, z. B. Fr., Hr. oder Dr.
TrailingData	Zeichenfolge	Information ohne Namen, die nach einem Namen erscheint.

Mit verbundenen Namen verwandte Felder

Conjunction2	Zeichenfolge	Gibt an, dass ein zweiter verbundener Name eine Konjunktion wie „und“, „oder“ oder „&“ enthält.
Conjunction3	Zeichenfolge	Gibt an, dass ein dritter verbundener Name eine Konjunktion wie „und“, „oder“ oder „&“ enthält.
FirmName2	Zeichenfolge	Der Name eines zweiten verbundenen Unternehmens. Zum Beispiel: Baltimore Gas & Electric dba Constellation Energy.
FirmSuffix2	Zeichenfolge	Das Suffix eines zweiten verbundenen Unternehmens.
FirstName2	Zeichenfolge	Der Vorname eines zweiten verbundenen Namens.
FirstName3	Zeichenfolge	Der Vorname eines dritten verbundenen Namens.

columnName	Format	Beschreibung
GeneralSuffix2	Zeichenfolge	Das allgemeine/berufliche Suffix für einen zweiten verbundenen Namen. Zum Beispiel Dr. oder PhD.
GeneralSuffix3	Zeichenfolge	Das allgemeine/berufliche Suffix für einen dritten verbundenen Namen. Zum Beispiel Dr. oder PhD.
IsConjoined	Zeichenfolge	Gibt an, ob der Eingabename verbunden ist. Ein Beispiel für einen verbundenen Namen ist „John und Jane Smith“.
LastName2	Zeichenfolge	Der Nachname eines zweiten verbundenen Namens.
LastName3	Zeichenfolge	Der Nachname eines dritten verbundenen Namens.
MaturitySuffix2	Zeichenfolge	Das Generationssuffix für einen zweiten verbundenen Namen. Zum Beispiel Jr. oder Sr.
MaturitySuffix3	Zeichenfolge	Das Generationssuffix für einen dritten verbundenen Namen. Zum Beispiel Jr. oder Sr.
MiddleName2	Zeichenfolge	Der zweite Vorname eines zweiten verbundenen Namens.
MiddleName3	Zeichenfolge	Der zweite Vorname eines dritten verbundenen Namens.
TitleOfRespect2	Zeichenfolge	Eine Information, die vor einem zweiten verbundenen Namen erscheint, z. B. Fr., Hr. oder Dr.
TitleOfRespect3	Zeichenfolge	Eine Information, die vor einem dritten verbundenen Namen erscheint, z. B. Fr., Hr. oder Dr.

9 - Info zu Spectrum™ Technology Platform

In this section

Was ist Spectrum™ Technology Platform?	576
Architektur der Unternehmensdatenverwaltung	577
Architektur von Spectrum™ Technology Platform	581
Module und Komponenten	586

Was ist Spectrum™ Technology Platform?

Spectrum™ Technology Platform ist ein System, mit dem die Vollständigkeit, die Gültigkeit, die Konsistenz, die Zeitgenauigkeit und die Genauigkeit Ihrer Daten durch Datenstandardisierung, -überprüfung und -erweiterung verbessert werden. Durch die Sicherstellung, dass Ihre Daten genau, vollständig und auf dem neuesten Stand sind, kann Ihr Unternehmen Ihre Kunden besser verstehen und besser mit ihnen arbeiten.

Spectrum™ Technology Platform hilft beim Entwerfen und Implementieren von Geschäftsregeln für die Datenqualität, indem folgende Funktionen ausgeführt werden.

Parsing, Namensstandardisierung und Namensüberprüfung

Für eine möglichst genaue Standardisierung müssen Sie gegebenenfalls Datenzeichenfolgen in mehrere Felder aufteilen. Spectrum™ Technology Platform bietet erweiterte Parsing-Features, mit denen Sie Personennamen, Unternehmensnamen und viele andere Begriffe und Abkürzungen parsen können. Zusätzlich können Sie eine eigene Liste mit benutzerdefinierten Begriffen erstellen, die als Grundlage für die Scan-/Extraktionsvorgänge verwendet werden sollen. Das Universal Name-Modul bietet diese Funktionalität.

Deduplizierung und Konsolidierung

Über das Identifizieren eindeutiger Entitäten können Sie Datensätze konsolidieren, Dubletten beseitigen und „Best of Breed“-Datensätze entwickeln. Ein „Best of Breed“-Datensatz ist ein aus Daten anderer Datensätze zusammengesetzter Datensatz. Das Advanced Matching-Modul und das Data Normalization-Modul bieten diese Funktionalität.

Adressenüberprüfung

Die Adressenüberprüfung wendet Regeln der entsprechenden Postbehörde an, um eine Adresse in eine Standardform zu überführen, und überprüft sogar, ob diese Adresse lieferfähig ist. Die Adressenüberprüfung kann Ihnen helfen, sich für Portoermäßigungen zu qualifizieren und die Lieferfähigkeit Ihrer Sendungen zu verbessern. Das Universal Addressing-Modul und das Address Now-Modul bieten diese Funktionalität.

Geocoding

Geocoding ist die Bestimmung der geografischen Koordinaten (Längen- und Breitengrad) einer Adresse. Geocoding kann u. a. zur Erstellung von Karten verwendet werden. Die zugrunde liegenden Standortdaten können z. B. auch bei der Entscheidungsfindung in Unternehmen helfen. In der Umkehrung des Prozesses können Sie auch einen Geocode eingeben – einen durch eine Breiten- und Längengradkoordinate dargestellten Punkt – und Adressinformationen zum Geocode erhalten. Das Enterprise Geocoding-Modul bietet diese Funktionalität.

Location Intelligence

Location Intelligence erstellt neue Informationen zu Ihren Daten, indem geografische Beziehungen bewertet, überprüft, analysiert und modelliert werden. Mit dem Einsatz von Location Intelligence können Sie Standorte überprüfen und Informationen in wertvolle Business Intelligence verwandeln. Das Location Intelligence-Modul bietet diese Funktionalität.

Master Data Management

Mit Master Data Management können Sie beziehungsorientierte Master Data-Ansichten Ihrer wichtigsten Datenbestände erstellen. Das Data Hub-Modul hilft Ihnen, Beeinflusser und nicht offensichtliche Beziehungen zu identifizieren, Betrug zu erkennen und die Qualität, Integration und Verfügbarkeit Ihrer Informationen zu verbessern.

Zuweisung von Steuerhoheit

Die Zuweisung von Steuerhoheit bestimmt die Steuerhoheiten, die für den Standort einer Adresse gelten. Das genaue Zuweisen von richtigen Steuerhoheiten kann das finanzielle Risiko und die gesetzliche Haftung verringern.

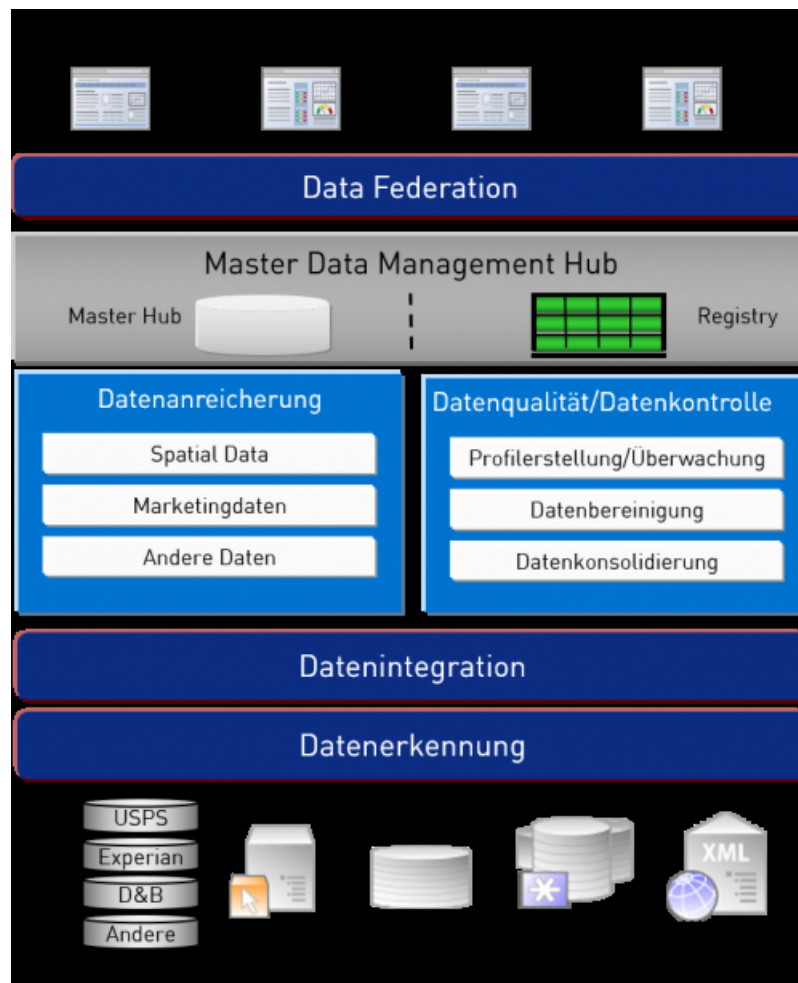
Die Software Spectrum™ Technology Platform von Pitney Bowes integriert die aktuellen Zuständigkeitsgrenzen über die exakten Straßenanschriften in Ihren Kundendatensätzen. So können Sie Ihren Datensätzen die korrekten Steuerinformationen bezüglich Bundesstaat, Bezirk, Township, Gemeinde und besonderer Steuergebiete hinzufügen. Einige Beispielanwendungen für die Zuweisung von Steuerhoheit sind:

- Umsatz- und Gebrauchssteuer
- persönliche Grundsteuer
- Versicherungsbeitragssteuer

Das Enterprise Tax-Modul bietet diese Funktionalität.

Architektur der Unternehmensdatenverwaltung

Mit Spectrum™ Technology Platform können Sie einen umfassenden Verwaltungsprozess für Unternehmensdaten aufbauen oder eine gezieltere Lösung anstreben. Das folgende Diagramm zeigt eine vollständige Lösung, die Quelldaten durch Datenanreicherungs- und Datenqualitätsprozesse leitet und einen Master Data Management Hub damit anreichert, der die Daten über ein einzelne Ansicht für mehrere Unternehmensanwendungen bereitstellt.



Datenerkennung

Datenerkennung ist ein Prozess, bei dem Ihre Datenquellen gescannt werden, um ein vollständiges Inventar Ihrer Datenlandschaft zu erstellen. Spectrum™ Technology Platform kann über eine Vielzahl an Techniken zur Datenprofilierung strukturierte Daten, unstrukturierte Daten und semistrukturierte Daten scannen. Die Ergebnisse des Scans werden verwendet, um automatisch eine Dokumentationsbibliothek zu generieren, die die Datenbestände Ihres Unternehmens beschreibt, und um eine Metadatenbank zu erstellen. Diese Dokumentation und die begleitende Metadatenbank bieten die Einsichten, die Sie benötigen, bevor Sie mit Projekten bezüglich Datenintegration, Datenqualität, Datenkontrolle oder Master Data Management beginnen.

Wenden Sie sich für weitere Informationen zum Spectrum™ Technology Platform Data Discovery-Modul an Ihren Kundenbetreuer.

Data Integration

Sobald Sie über ein Inventar Ihrer Datenlandschaft verfügen, müssen Sie sich überlegen, wie Sie auf die Daten, die Sie verwalten müssen, zugreifen. Spectrum™ Technology Platform kann direkt oder durch Integration in Ihre vorhandenen Datenzugriffstechnologien eine Verbindung mit Daten

in mehreren Quellen herstellen. Batch- und Echtzeitfunktionen für Datenintegration werden für verschiedenste Geschäftsanforderungen, einschließlich Data Warehousing, Datenqualität, Systemintegration und Migration, unterstützt. Spectrum™ Technology Platform kann auf Daten unter anderem in RDBMS-Datenbanken, Data Warehouses, XML-Dateien und Flatfiles zugreifen. Spectrum™ Technology Platform unterstützt SQL-Abfragen mit komplexen Zusammenführungen und Aggregationen und bietet ein visuelles Tool zu Abfragenentwicklung. Zudem kann Spectrum™ Technology Platform über REST- und SOAP-Webservices auf Daten zugreifen.

Spectrum™ Technology Platform kann aufgrund des Vorhandenseins einer oder mehrerer Quelldateien in einem angegebenen Ordner Batch-Verarbeitungen auslösen. Dieses Auslösen anhand eines Überwachungsordners ist nützlich, um Hochladen über FTP zu überwachen und nach Abschluss die Datei sofort zu verarbeiten.

Einige dieser Datenintegrationsfunktionen erfordern eine Lizenz für das Enterprise Data Integration-Modul. Wenden Sie sich für weitere Informationen an Ihren Kundenbetreuer.

Schließlich lässt sich Spectrum™ Technology Platform noch in paketisierte Anwendungen wie SAP und Siebel integrieren.

Datenqualität/Datenkontrolle

Prozesse für Datenqualität und Datenkontrolle überprüfen Ihre Daten auf Datensatzdubletten und auf inkonsistente sowie ungenaue Informationen.

Dublettenabgleich identifiziert mögliche Datensatzdubletten oder Beziehungen zwischen Datensätzen anhand von Namen und Adresse oder einem beliebigen anderen Informationstyp. Mit Spectrum™ Technology Platform können Sie über boolesche Vergleichsmethoden, Bewertungsmethoden, Schwellenwerte, Algorithmen und Gewichtungen ein konsistentes Set geschäftlicher Vergleichsregeln spezifizieren und feststellen, ob eine Gruppe von Datensätzen Dubletten enthält. Spectrum™ Technology Platform ist äußerst anpassungsfähig, sodass Sie die Regeln auf die einzigartigen Anforderungen Ihres Unternehmens zuschneiden können.

Sobald Datensatzdubletten identifiziert wurden, können Sie die Datensätze konsolidieren. Mit Spectrum™ Technology Platform sind Sie in der Lage, zu spezifizieren, wie Datensatzdubletten verknüpft oder zusammengeführt werden sollen, damit Sie den genauesten und vollständigsten Datensatz aus den gesammelten Informationen erstellen können. Beispielsweise kann aus allen Datensätzen für einen Haushalt ein einzelner Best-of-Breed-Datensatz erstellt werden. Das Advanced Matching-Modul wird verwendet, um Dubletten zu erkennen und zu beseitigen.

Datenqualitätsprozesse können zudem Ihre Daten standardisieren. Standardisierung ist ein wichtiger Prozess, da standardisierte Datenelemente erforderlich sind, um beim Abgleichen und Identifizieren von Beziehungen zwischen Datensätzen die bestmöglichen Ergebnisse zu erzielen. Während verschiedene Module eine Standardisierung des einen oder anderen Typs durchführen, bietet das Spectrum™ Technology Platform Data Normalization-Modul das umfassendste Set an Standardisierungsfunktionen. Darüber hinaus bietet das Universal Name-Modul bestimmte Datenqualitätsfunktionen zur Verarbeitung von Personennamen und Unternehmensnamen.

Standardisierte Daten sind nicht notwendigerweise genaue Daten. Spectrum™ Technology Platform kann Ihre Daten anhand bekannter, aktueller Referenzdaten auf ihre Richtigkeit hin überprüfen. Die für diesen Prozess verwendeten Quellen können von Behörden wie dem U.S. Postal Service, von

externen Datenanbietern wie Experian oder D&B oder von internen Referenzquellen Ihres Unternehmens wie Buchhaltungsdaten stammen. Spectrum™ Technology Platform ist besonders leistungsstark bei der Adressdatenüberprüfung. Adressen in 250 Ländern und Territorien auf der ganzen Welt können überprüft und standardisiert werden. Es gibt zwei Module zur Adressenüberprüfung: das Address Now-Modul und das Universal Addressing-Modul.

Besprechen Sie Ihre Anforderungen mit Ihrem Kundenbetreuer, um festzustellen, welches Modul für Sie das richtige ist.

Obwohl Spectrum™ Technology Platform automatisch viele Datenqualitätsprobleme beheben kann, gibt es einige Situationen, in denen eine manuelle Überprüfung durch einen Datenverantwortlichen erfolgen sollte. Um dies zu unterstützen, bietet das Business Steward-Modul eine Möglichkeit, die Regeln zu spezifizieren, die eine manuelle Überprüfung auslösen. Zudem bietet das Modul ein webbasiertes Tool zur Überprüfung von Ausnahmedatensätzen. Es beinhaltet integrierten Zugriff auf Drittanbietertools wie Bing-Karten und Experian-Daten, um Datenverantwortlichen im Prozess der Überprüfung und Auflösung zu helfen.

Datenanreicherung

Prozesse zur Datenanreicherung erweitern Ihre Daten um zusätzliche Informationen. Die Anreicherung kann auf Geodaten, Marketingdaten oder Daten aus anderen Quellen basieren, die Sie verwenden möchten, um Ihre Daten um zusätzliche Details zu erweitern. Wenn Sie beispielsweise über eine Datenbank mit Kundenadressen verfügen, könnten Sie die Adressen geocodieren, um Breitengrad/Längengrad-Koordinaten der Adressen zu ermitteln, und diese Koordinaten als Teil der Datensätze speichern. Sie könnten dann Ihre Kundendaten für verschiedene Geodatenberechnungen verwenden, wie das Suchen nach der nächsten Bankfiliale zu einer Kundenadresse. Mit Spectrum™ Technology Platform können Sie Ihre Daten um verschiedenste Informationen erweitern, einschließlich Geocoding (über das Enterprise Geocoding-Modul), Steuerhoheitszuweisungen (über das Enterprise Tax-Modul), Geodatenberechnungen (über das Location Intelligence-Modul) und Richtungsanweisungen zum Fahren oder Gehen zwischen Punkten (über das Enterprise Routing-Modul).

Master Data Management Hub

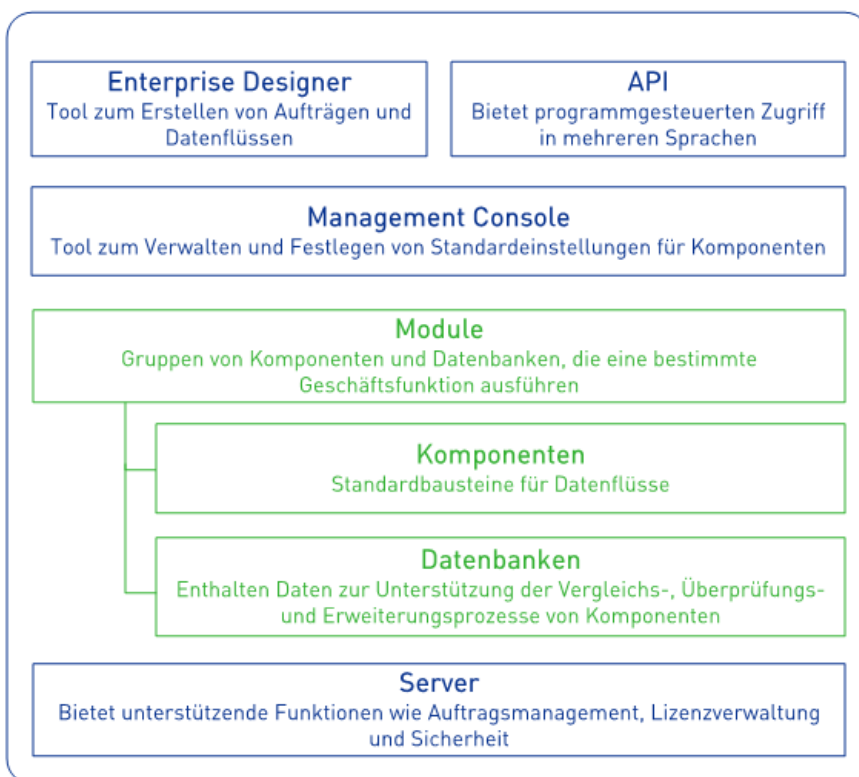
Der Master Data Management (MDM) Hub ermöglicht schnelle Modellierung von Entitäten und ihren komplexen Beziehungen über Rollen, Prozesse und Interaktionen hinweg. Er bietet integrierte Funktionen zur Analyse von soziale Netzwerken, um Ihnen zu helfen, Beeinflusser zu verstehen, Kundenabwanderungen vorherzusagen, nicht offensichtliche Beziehungen und Betrugsmuster zu erkennen und Empfehlungen zu geben.

Spectrum™ Technology Platform unterstützt zwei Ansätze für den MDM Hub. Im Master Hub-Ansatz werden die Daten in einer einzelnen MDM-Datenbank gepflegt und Anwendungen greifen auf Daten in der MDM-Datenbank zu. Im Registrierungsansatz werden die Daten in den einzelnen Unternehmensanwendungen gepflegt und die MDM Hub-Registrierung enthält Schlüssel, die zum Suchen nach zugehörigen Datensätzen verwendet werden. Beispielsweise kann ein Kundendatensatz in einer Auftragserfassungsdatenbank und in einer Kundendienstdatenbank vorhanden sein. Die MDM-Registrierung würde dann einen einzelnen Schlüssel enthalten, der an beiden Speicherorten für den Zugriff auf die Kundendaten verwendet werden kann.

Das Data Hub-Modul bietet MDM-Funktionen.

Architektur von Spectrum™ Technology Platform

Spectrum™ Technology Platform von Pitney Bowes besteht aus einem Server, der eine Anzahl von Modulen ausführt. Diese Module bieten verschiedene Funktionen wie unter anderem Adressenüberprüfung, Geocoding und erweitertes Parsing. Im folgenden Diagramm wird die Architektur von Spectrum™ Technology Platform veranschaulicht.



Server

Die Basis von Spectrum™ Technology Platform ist der Server. Der Server übernimmt die Datenverarbeitung, synchronisiert Datenbankdaten und verwaltet die Kommunikation. Sie bietet Auftragsmanagement- und Sicherheitsfeatures.

Module

Module sind Sets von Features, die eine bestimmte Funktion ausführen. Das Universal Addressing-Modul standardisiert Adressen beispielsweise, damit diese Adressen postalischen Standards entsprechen. Das Enterprise Tax-Modul bestimmt die Steuerhoheit für eine angegebene

Adresse. Module werden zusammen gruppiert, um häufige Geschäftsprobleme zu lösen, und zusammen als Pakete lizenziert.

Komponenten

Module sind aus Komponenten zusammengesetzt, die eine bestimmte Funktion in einem Fluss oder als ein Dienst ausführen. Die Komponente „Geocode US Address“ des Enterprise Geocoding-Moduls gibt beispielsweise den Längen- und Breitengrad für eine Eingabeadresse zurück, die Komponente „Get City State Province“ des Universal Addressing-Moduls gibt den Ort und das Bundesland oder den Kanton einer eingegebenen Postleitzahl zurück.

Welche Komponenten auf Ihrem System verfügbar sind, hängt davon ab, welches Spectrum™ Technology Platform-Bundle Sie lizenziert haben.

Datenbanken

Einige Module sind von Datenbanken abhängig, die Referenzdaten enthalten. Das Universal Addressing-Modul muss beispielsweise Zugriff auf Daten des US-amerikanischen Postdiensts haben, um Adressen in den USA zu überprüfen und zu standardisieren. Datenbanken werden separat installiert und einige von ihnen werden regelmäßig aktualisiert, um Ihnen aktuelle Daten bereitzustellen.

Module verfügen neben erforderlichen Datenbanken über optionale Datenbanken. Optionale Datenbanken enthalten Daten, die für bestimmte Features erforderlich sind, die Ihren Spectrum™ Technology Platform-Prozess erweitern können.

Management Console

Die Management Console ist ein Administrationstool für Spectrum™ Technology Platform. Sie können die Management Console für folgende Aktionen verwenden:

- Definieren der Verbindungen zwischen Spectrum™ Technology Platform und Ihren Daten
- Spezifizieren der Standardeinstellungen für Dienste und Flüsse
- Verwalten von Benutzerkonten einschließlich Berechtigungen und Kennwörtern
- Anzeigen von Protokollen
- Anzeigen von Lizenzen einschließlich Informationen zum Ablauf

Management Console | Flüsse | Dienste | Ressourcen | System | ? admin

Startseite > Ressourcen: Datenquellen

Datenquellen

+ ✎ 🔄 🗑️ 👤

Filter

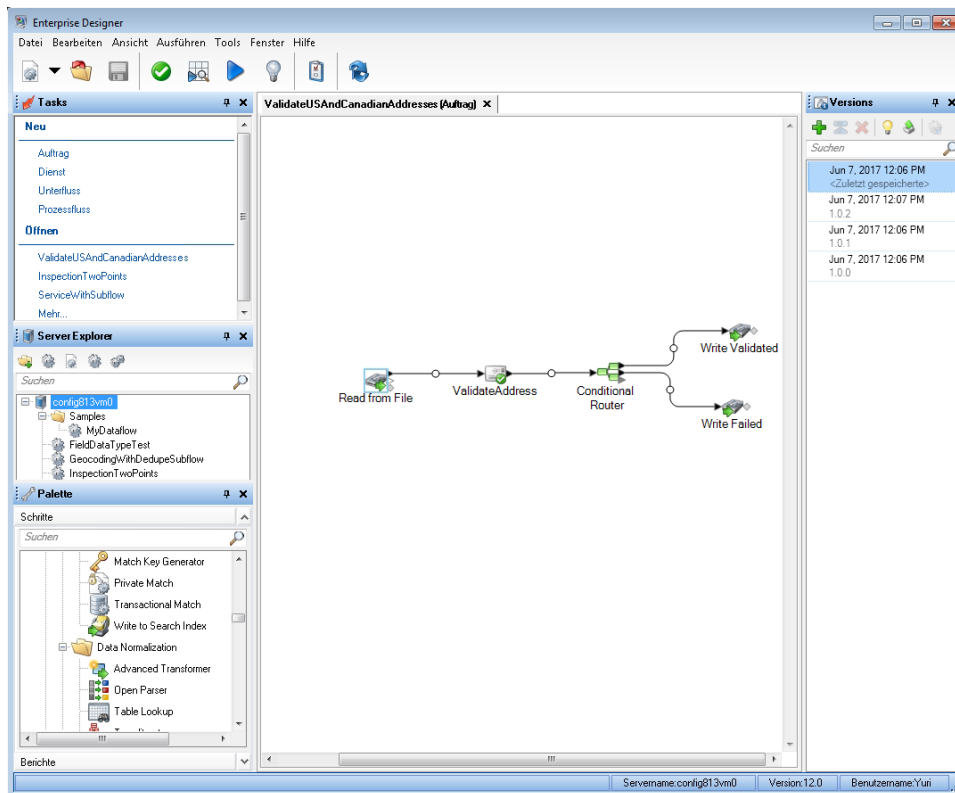
Name	Typ
test1	FTP
test2	FTP
test4	Cloud
test5HDFS	HDFS
GourmetFoods	MSSQLServer

Anzeigen von 5 von 5 Datensätzen | Zeilen pro Seite 10

pitney bowes © 2017 Pitney Bowes Inc.

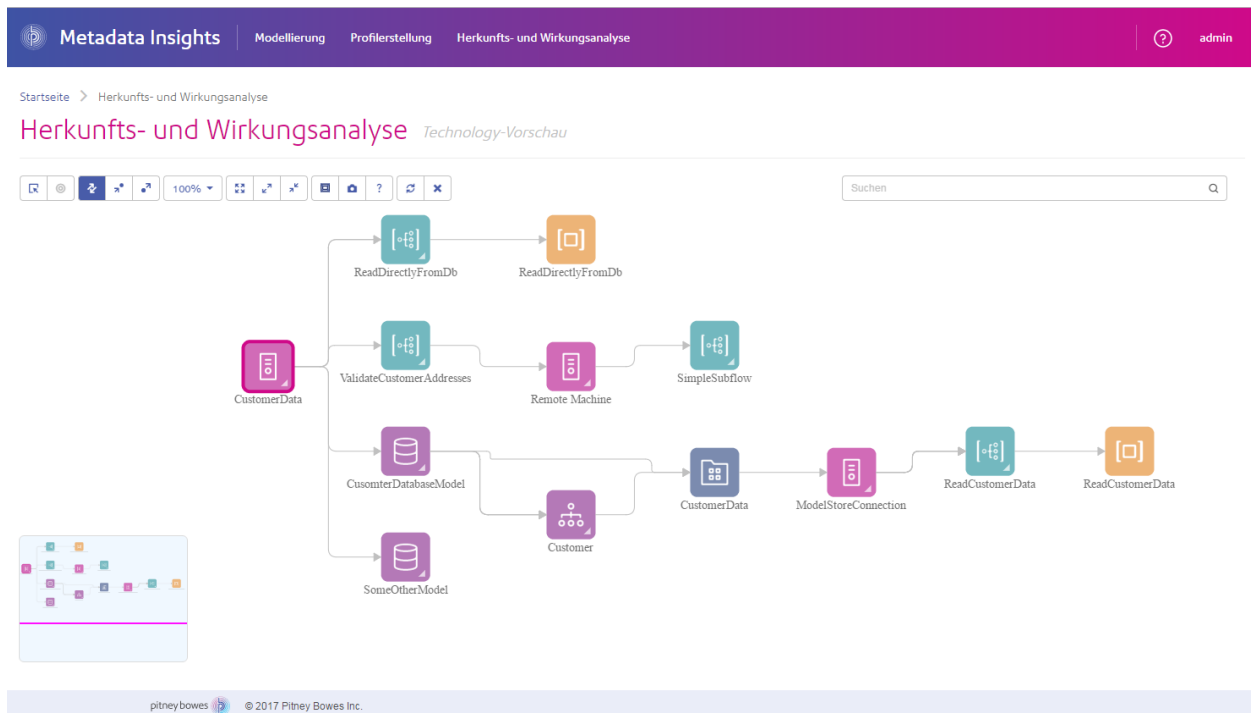
Enterprise Designer

Der Enterprise Designer ist ein Tool zum Erstellen von Aufträgen, Diensten, Unterflüssen und Prozessflüssen für Spectrum™ Technology Platform. Über die einfache Oberfläche können Sie durch Ziehen und Ablegen komplexe Datenflüsse grafisch erstellen.



Metadata Insights

Metadata Insights gibt Ihnen die Kontrolle, die Sie benötigen, um genaue und rechtzeitige datengesteuerte Einblicke in Ihr Geschäft zu geben. Mit Metadata Insights können Sie Datenmodelle entwickeln, den Fluss von Daten von der Quelle bis zur Geschäftsanwendung verfolgen und die Qualität Ihrer Daten durch Profiling bewerten. Mit diesen Einblicken können Sie die Datenressourcen identifizieren, die für die Beantwortung bestimmter Geschäftsfragen, für die Anpassung und Optimierung von Prozessen zur Verbesserung der Nützlichkeit und Konsistenz von Daten in Ihrem gesamten Geschäft und für die Behebung von Datenproblemen verwendet werden.



Webservices und API

Sie können Spectrum™ Technology Platform-Funktionen in Ihre Anwendungen integrieren. Verwenden Sie dazu Webservices und Programmier-APIs. Diese Schnittstellen bieten eine einfache Integration, optimieren die Datenbankverarbeitung und unterstützen die Abwärtskompatibilität für zukünftige Versionen.

Die Spectrum™ Technology Platform-API ist für die folgenden Sprachen verfügbar:

- C
- C++
- COM
- Java
- .NET

Webservices sind über SOAP und REST verfügbar.

Module und Komponenten

Tabelle 136: Module und Komponenten

Moduls	Beschreibung	Komponenten
Address Now-Modul	Es bietet verbesserte Überprüfung und Standardisierung für Adressen außerhalb der USA und weitere Adressenverarbeitungsfunktionen.	Build Global Address Get Global Candidate Addresses Validate Global Address
Advanced Matching-Modul	Vergleicht Datensätze innerhalb und/oder zwischen Eingabedateien.	Best Of Breed Candidate Finder Duplicate Synchronization Filter Interflow Match Intraflow Match Match Key Generator Transactional Match
Advanced Scoring-Modul	Vergleicht Datensätze innerhalb und/oder zwischen Eingabedateien.	PMML Model Scoring Read from Miner Dataset Write to Miner Dataset
Business Steward-Modul	Identifiziert Ausnahmedatensätze und stellt ein browserbasiertes Tool für die manuelle Überprüfung von Ausnahmedatensätzen bereit.	Exception Monitor Read Exceptions Write Exceptions
Country Identifier	Wählt einen Ländernamen oder eine Kombination aus Postleitzahl und Bundesland/Kanton und gibt den zweistelligen ISO-Ländercode, den dreistelligen Universal Postal Union (UPU)-Code und den englischen Ländernamen zurück.	Country Identifier

Moduls	Beschreibung	Komponenten
Data Hub-Modul	Verknüpft und analysiert Daten, identifiziert Beziehungen und Trends.	Write to Hub Read from Hub Query Hub Graph Visualization
Data Integration-Modul	Bietet nützliche Funktionen für Data Warehousing, Datenqualität, Systemintegration und Migration.	Field Selector Generate Time Dimension Query Cache Write to Cache
Data Normalization-Modul	Entfernt Inkonsistenzen in Daten.	Advanced Transformer Open Parser Table Lookup Transliterator
Enterprise Data Integration	Stellt für verschiedene Geschäftsanforderungen, einschließlich Data Warehousing, Datenqualität, Systemintegration und Migration, eine Verbindung zu Daten in mehreren Quellen her.	Call Stored Procedure Field Selector Generate Time Dimension Query Cache Write to Cache
Enterprise Geocoding-Modul	Bestimmt die geografischen Koordinaten einer Adresse. Bestimmt außerdem die Adresse zu gegebenem Längengrad und Breitengrad.	Geocode Address AUS Geocode Address GBR Geocode Address Global Geocode Address World Geocode US Address GNAF PID Location Search Reverse APN Lookup Reverse Geocode Address Global Reverse Geocode US Location

Moduls	Beschreibung	Komponenten
Enterprise Routing-Modul	Ruft Wegbeschreibungen (zum Fahren oder Gehen) ab, berechnet Fahrtzeit und Entfernung und identifiziert Orte innerhalb einer bestimmten Zeitspanne oder Entfernung von einem Startpunkt.	Get Route Data Get Travel Boundary Get Travel Cost Matrix Get Travel Directions Persistent Update
Enterprise Tax-Modul	Bestimmt die Steuerhoheit für einen angegebenen Standort	Assign GeoTAX Info Calculate Distance
GeoConfidence-Modul	Bestimmt die Wahrscheinlichkeit, dass eine Adresse oder Straßenkreuzung innerhalb eines gegebenen Bereichs liegt.	Geo Confidence Surface CreatePointsConvexHull
Global Addressing-Modul	Bietet eine erweiterte Adressenstandardisierung und -prüfung für Adressen außerhalb der USA. Schlägt zudem automatisch Adressen vor, während Sie tippen, und gibt sofort Kandidaten basierend auf Ihrer Eingabe zurück. Teilen Sie postalische Adresszeichenfolgen unter Verwendung von Machine Learning-Techniken in einzelne Adresselemente auf.	Global Address Validation Globaler Typ voraus Globaler Adressen-Parser
Global Geocoding-Modul	Bestimmt die geografischen Koordinaten einer Adresse. Bestimmt außerdem die Adresse zu gegebenem Längengrad und Breitengrad.	Globaler Geocode Globaler Reverse-Geocode
Global Sentry	Versucht, Transaktionen mit von Regierungsbehörden bereitgestellten Suchlisten abzugleichen, die Daten aus verschiedenen Ländern enthalten.	Global Sentry Global Sentry Address Check Global Sentry ID Number Check Global Sentry Name Check Global Sentry Other Data Check

Moduls	Beschreibung	Komponenten
Location Intelligence-Modul	Führt Punkt-in-Polygon-Analysen und radiale Analysen mit verschiedenen Geodaten-Datenbanken durch.	<ul style="list-style-type: none"> Closest Site Find Nearest Point In Polygon Query Spatial Data Read Spatial Data Spatial Calculator Spatial Union
Machine Learning-Modul	Bietet die Möglichkeit, ein Binning für numerische Daten durchzuführen, überwachte und unüberwachte Machine Learning-Modelle anzupassen und Daten in diesen Modellen zu bewerten.	<ul style="list-style-type: none"> Binning Binning Lookup Java Model Scoring K-Means Clustering Linear Regression Logistic Regression Principal Component Analysis Random Forest Classification Random Forest Regression
Metadata Insights	Gibt Ihnen die Kontrolle, die Sie benötigen, um genaue und rechtzeitige datengesteuerte Einblicke in Ihr Geschäft zu geben. Entwickelt Datenmodelle, verschafft Ihnen einen Überblick über den Fluss von Daten von der Quelle bis zur Geschäftsanwendung und bewertet die Qualität Ihrer Daten durch Profilerstellung. So können Sie einfacher die Datenressourcen identifizieren, die Sie für die Beantwortung bestimmter Geschäftsfragen und für die Optimierung von Prozessen zur Verbesserung der Nützlichkeit und Konsistenz von Daten in Ihrem gesamten Geschäft verwenden sollten.	<ul style="list-style-type: none"> Modelle (logische und physische) Modellspeicher Profil Herkunfts- und Wirkungsanalyse

Moduls	Beschreibung	Komponenten
SAP-Modul	Verbindet Spectrum™ Technology Platform über eine Schnittstelle mit Anwendungen des SAP Customer Relationship Management-Moduls.	<ul style="list-style-type: none"> SAP Generate Match Key SAP Generate Match Score SAP Generate Search Key SAP Generate Search Key Constant SAP Generate Search Key Metaphone SAP Generate Search Key Substring SAP Validate Address With Candidates
Siebel-Modul	Verbindet Spectrum™ Technology Platform über eine Schnittstelle mit Anwendungen von Siebel.	<ul style="list-style-type: none"> Siebel Generate Match Key Siebel Generate Match Score Siebel Generate Search Key Siebel Business Name Standardization Siebel Standardize Name Siebel Geocode US Address With Candidates Siebel Geocode US Address With No Candidates Siebel Get Global Candidate Addresses Siebel Validate Address With Candidates Siebel Validate Address With No Candidates
Universal Addressing-Modul	Standardisiert und überprüft Adressen gemäß den Standards der Postbehörde.	<ul style="list-style-type: none"> Get Candidate Addresses Get City State Province Postleitzahlen abrufen Validate Address Validate Address AUS Validate Address Global

Moduls	Beschreibung	Komponenten
Universal Name-Modul	Parst Personennamen, Unternehmensnamen, Adressen sowie viele andere Begriffe und Abkürzungen.	Name Parser (veraltet) Name Variant Finder Open Name Parser

Anhang

In this section

ISO-Ländercodes und Modulunterstützung

593

A - ISO-Ländercodes und Modulunterstützung

[In this section](#)

ISO-Ländercodes und Modulunterstützung

594

ISO-Ländercodes und Modulunterstützung

Diese Tabelle enthält die ISO-Codes für jedes Land sowie die Module, die die Adressierung, das Geocoding und die Routenführung für jedes Land unterstützen.

Beachten Sie, dass das Enterprise Geocoding-Modul Datenbanken für Afrika (30 Länder), den Mittleren Osten (8 Länder) und Lateinamerika (20 Länder) enthält. Diese Datenbanken decken die kleineren Länder in diesen Regionen ab, die nicht über ihre eigenen landesspezifischen Geocoding-Datenbanken verfügen. Die Spalte „Unterstützte Module“ gibt an, welche Länder von diesen Datenbanken für Afrika, den Mittleren Osten und Lateinamerika abgedeckt werden.

Außerdem stellt die Geocode Address World-Datenbank geografisches und begrenztes Post-Geocoding (keine Geocoding auf Straßenebene) für alle Länder bereit.

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Afghanistan	AF	AFG	Universal Addressing-Modul
Ålandinseln	AX	ALA	Universal Addressing-Modul
Albanien	AL oder SQ (Routing)	ALB	Universal Addressing-Modul Enterprise Geocoding-Modul Enterprise Routing-Modul
Algerien	DZ	DZA	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Amerikanisch-Samoa	AS	ASM	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Andorra	AD	AND	Enterprise Geocoding-Modul. (Andorra wird durch den spanischen Geocoder abgedeckt.) Universal Addressing-Modul GeoComplete-Modul
Angola	AO	AGO	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Anguilla	AI	AIA	Universal Addressing-Modul
Antarktika	AQ	ATA	Universal Addressing-Modul
Antigua und Barbuda	AG	ATG	Universal Addressing-Modul
Argentinien	AR	ARG	Enterprise Geocoding-Modul Universal Addressing-Modul Enterprise Routing-Modul
Armenien	AM	ARM	Universal Addressing-Modul
Aruba	AW	ABW	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Australien	AU	AUS	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul
Österreich	AT	AUT	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Aserbaidschan	AZ	AZE	Universal Addressing-Modul
Bahamas	BS	BHS	Enterprise Geocoding-Modul Universal Addressing-Modul Enterprise Routing-Modul
Bahrain	BH	BHR	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul
Bangladesch	BD	BGD	Universal Addressing-Modul
Barbados	BB	BRB	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Weißrussland	BY	BLR	Universal Addressing-Modul Enterprise Routing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Belgien	BE	BEL	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Belize	BZ	BLZ	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Benin	BJ	BEN	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Bermuda	BM	BMU	Universal Addressing-Modul Enterprise Routing-Modul
Bhutan	BT	BTN	Universal Addressing-Modul
Plurinationaler Staat Bolivien	BO	BOL	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Bonaire, Sint Eustatius und Saba	BQ	BES	Universal Addressing-Modul
Bosnien und Herzegowina	BA	BIH	Universal Addressing-Modul Enterprise Routing-Modul Enterprise Geocoding-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Botswana	BW	BWA	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Bouvet-Insel	BV	BVT	Universal Addressing-Modul
Brasilien	BR	BRA	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Britisches Territorium im Indischen Ozean	IO	IOT	Universal Addressing-Modul
Brunei Darussalam	BN	BRN	Enterprise Geocoding-Modul Universal Addressing-Modul
Bulgarien	BG	BGR	Universal Addressing-Modul
Burkina Faso	BF	BFA	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Burundi	BI	BDI	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Kambodscha	KH	KHM	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Kamerun	CM	CMR	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Kanada	CA	CAN	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Kap Verde	CV	CPV	Universal Addressing-Modul
Caymaninseln	KY	CYM	Universal Addressing-Modul
Zentralafrikanische Republik	CF	CAF	Universal Addressing-Modul
Tschad	TD	TCD	Universal Addressing-Modul
Chile	CL	CHL	Enterprise Geocoding-Modul Universal Addressing-Modul Enterprise Routing-Modul GeoComplete-Modul
China	CN oder zh_CN (Routing)	CHN	Enterprise Geocoding-Modul Universal Addressing-Modul Enterprise Routing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Weihnachtsinsel	CX	CXR	Universal Addressing-Modul
Cocosinseln	CC	CCK	Universal Addressing-Modul
Kolumbien	CO	COL	Universal Addressing-Modul
Komoren	KM	COM	Universal Addressing-Modul
Kongo	CG	COG	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Demokratische Republik Kongo	CD	COD	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul Enterprise Routing-Modul
Cook-Inseln	CK	COK	Universal Addressing-Modul
Costa Rica	CR	CRI	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Elfenbeinküste	CI	CIV	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Kroatien	HR	HRV	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul
Kuba	CU	CUB	Enterprise Geocoding-Modul (Lateinamerika) Enterprise Routing-Modul Universal Addressing-Modul
Curaçao	CW	CUW	Universal Addressing-Modul
Zypern	CY	CYP	Enterprise Geocoding-Modul Universal Addressing-Modul
Tschechien	CZ oder CS (Routing)	CZE	Enterprise Geocoding-Modul Universal Addressing-Modul Enterprise Routing-Modul GeoComplete-Modul
Dänemark	DK	DNK	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Dschibuti	DJ	DJI	Universal Addressing-Modul
Dominica	DM	DMA	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Dominikanische Republik	DO	DOM	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Ecuador	EC	ECU	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Ägypten	EG	EGY	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul
El Salvador	SV	SLV	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Äquatorialguinea	GQ	GNQ	Universal Addressing-Modul
Eritrea	ER	ERI	Universal Addressing-Modul
Estland	EE	EST	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul
Äthiopien	ET	ETH	Universal Addressing-Modul
Falklandinseln (Malwinen)	FK	FLK	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Färöer-Inseln	FO	FRO	Universal Addressing-Modul
Fidschi	FJ	FJI	Universal Addressing-Modul
Finnland	FI	FIN	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Frankreich	FR	FRA	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Französisch-Guayana	GF	GUF	Enterprise Geocoding-Modul (<i>Französisch-Guayana wird durch den französischen Geocoder abgedeckt.</i>) Universal Addressing-Modul
Französisch-Polynesien	PF	PYF	Universal Addressing-Modul
Französische Gebiete im südlichen Indischen Ozean	TF	ATF	Universal Addressing-Modul
Gabun	GA	GAB	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Gambia	GM	GMB	Universal Addressing-Modul
Georgien	GE	GEO	Universal Addressing-Modul
Deutschland	DE	DEU	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Ghana	GH	GHA	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul Enterprise Routing-Modul
Gibraltar	GI	GIB	Enterprise Geocoding-Modul (<i>Gibraltar wird durch den spanischen Geocoder abgedeckt.</i>) Universal Addressing-Modul
Griechenland	GR	GRC	Enterprise Geocoding-Modul Universal Addressing-Modul
Grönland	GL	GRL	Universal Addressing-Modul
Grenada	GD	GRD	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Guadeloupe	GP	GLP	Enterprise Geocoding-Modul (<i>Guadeloupe wird durch den französischen Geocoder abgedeckt.</i>) Universal Addressing-Modul
Guam	GU	GUM	Universal Addressing-Modul
Guatemala	GT	GTM	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Guernsey	GG	GGY	Universal Addressing-Modul
Guinea	GN	GIN	Universal Addressing-Modul
Guinea-Bissau	GW	GNB	Universal Addressing-Modul
Guyana	GY	GUY	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Haiti	HT	HTI	Universal Addressing-Modul
Heard und McDonaldinseln	HM	HMD	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Apostolischer Stuhl (Vatikanstaat)	VA	VAT	Enterprise Geocoding-Modul (<i>Der Vatikan wird durch den italienischen Geocoder abgedeckt.</i>) Universal Addressing-Modul
Honduras	HN	HND	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Hongkong	HK	HKG	Enterprise Geocoding-Modul Universal Addressing-Modul
Ungarn	HU	HUN	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Island	IS	ISL	Enterprise Geocoding-Modul Universal Addressing-Modul
Indien	IN	IND	Enterprise Geocoding-Modul Universal Addressing-Modul
Indonesien	ID	IDN	Enterprise Geocoding-Modul Universal Addressing-Modul
Islamische Republik Iran	IR	IRN	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Irak	IQ	IRQ	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul
Irland	IE	IRL	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Isle of Man	IM	IMN	Universal Addressing-Modul
Israel	IL	ISR	Universal Addressing-Modul Enterprise Routing-Modul
Italien	IT	ITA	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Jamaika	JM	JAM	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Japan	JP	JPN	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Jersey	JE	JEY	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Jordanien	JO	JOR	Universal Addressing-Modul Enterprise Geocoding-Modul (Mittlerer Osten) Enterprise Routing-Modul
Kasachstan	KZ	KAZ	Universal Addressing-Modul
Kenia	KE	KEN	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul Enterprise Routing-Modul
Kiribati	KI	KIR	Universal Addressing-Modul
Demokratische Volksrepublik Korea	KP	PRK	Universal Addressing-Modul
Republik Korea	KR	KOR	Universal Addressing-Modul
Kosovo	KS	KOS	Enterprise Geocoding-Modul Universal Addressing-Modul GeoComplete-Modul
Kuwait	KW	KWT	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul
Kirgisistan	KG	KGZ	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Volksdemokratische Republik Laos	LA	LAO	Universal Addressing-Modul
Lettland	LV	LVA	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul
Libanon	LB	LBN	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul
Lesotho	LS	LSO	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul Enterprise Routing-Modul
Liberia	LR	LBR	Universal Addressing-Modul
Libysch-Arabische Dschamahirija	LY	LBY	Universal Addressing-Modul
Liechtenstein	LI	LIE	Enterprise Geocoding-Modul (<i>Liechtenstein wird durch den Schweizer Geocoder abgedeckt.</i>) Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Litauen	LT	LTU	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Luxemburg	LU	LUX	Enterprise Geocoding-Modul (<i>Luxemburg wird durch den belgischen Geocoder abgedeckt.</i>) Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Macao	MO	MAC	Enterprise Geocoding-Modul Universal Addressing-Modul
Ehemalige jugoslawische Republik Mazedonien	MK	MKD	Enterprise Geocoding-Modul Universal Addressing-Modul
Madagaskar	MG	MDG	Universal Addressing-Modul
Malawi	MW	MWI	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Malaysia	MY	MYS	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul
Malediven	MV	MDV	Universal Addressing-Modul
Mali	MT	MLI	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Malta	MT	MLT	Enterprise Geocoding-Modul Universal Addressing-Modul
Marshallinseln	MH	MHL	Universal Addressing-Modul
Martinique	MQ	MTQ	Enterprise Geocoding-Modul (<i>Martinique wird durch den französischen Geocoder abgedeckt.</i>) Universal Addressing-Modul
Mauretanien	MR	MRT	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Mauritius	MU	MUS	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Mayotte	YT	MYT	Enterprise Geocoding-Modul (<i>Mayotte wird durch den französischen Geocoder abgedeckt.</i>) Universal Addressing-Modul
Mexiko	MX	MEX	Enterprise Geocoding-Modul Universal Addressing-Modul
Föderierte Staaten von Mikronesien	FM	FSM	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Republik Moldau	MD	MDA	Universal Addressing-Modul Enterprise Routing-Modul
Monaco	MC	MCO	Enterprise Geocoding-Modul (<i>Monaco wird durch den französischen Geocoder abgedeckt.</i>) Universal Addressing-Modul
Mongolei	MN	MNG	Universal Addressing-Modul
Montenegro	ME	MNE	Enterprise Geocoding-Modul Universal Addressing-Modul
Montserrat	MS	MSR	Universal Addressing-Modul
Marokko	MA	MAR	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Mosambik	MZ	MOZ	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul Enterprise Routing-Modul
Myanmar	MM	MMR	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Namibia	NA	NAM	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Nauru	NR	NRU	Universal Addressing-Modul
Nepal	NP	NPL	Universal Addressing-Modul
Niederlande	NL	NLD	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Neukaledonien	NC	NCL	Universal Addressing-Modul
Neuseeland	NZ	NZL	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul
Nicaragua	NI	NIC	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Niger	NE	NER	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Nigeria	NG	NGA	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Niue	NU	NIU	Universal Addressing-Modul
Norfolkinsel	NF	NFK	Universal Addressing-Modul
Nördliche Marianen	MP	MNP	Universal Addressing-Modul
Norwegen	NO	NOR	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Oman	OM	OMN	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul
Pakistan	PK	PAK	Universal Addressing-Modul
Palau	PW	PLW	Universal Addressing-Modul
Besetztes palästinensisches Gebiet	PS	PSE	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Panama	PA	PAN	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Papua-Neuguinea	PG	PNG	Universal Addressing-Modul
Paraguay	PY	PRY	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Peru	PE	PER	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Philippinen	PH	PHL	Enterprise Geocoding-Modul Universal Addressing-Modul Enterprise Routing-Modul
Pitcairn	PN	PCN	Universal Addressing-Modul
Polen	PL	POL	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Portugal	PT	PRT	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Puerto Rico	PR	PRI	Universal Addressing-Modul
Katar	QA	QAT	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul
Réunion	RE	REU	Enterprise Geocoding-Modul (<i>Réunion wird durch den französischen Geocoder abgedeckt.</i>) Universal Addressing-Modul
Rumänien	RO	ROU	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul
Russische Föderation	RU	RUS	Enterprise Geocoding-Modul Universal Addressing-Modul GeoComplete-Modul
Ruanda	RW	RWA	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Saint Barthelemy	BL	BLM	Universal Addressing-Modul
St. Helena, Ascension und Tristan da Cunha	SH	SHE	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Saint Kitts und Nevis	KN	KNA	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
St. Lucia	LC	LCA	Universal Addressing-Modul
Saint-Martin (französischer Teil)	MF	MAF	Universal Addressing-Modul
Saint Pierre und Miquelon	PM	SPM	Universal Addressing-Modul
St. Vincent / Grenadinen	VC	VCT	Universal Addressing-Modul
Samoa	WS	WSM	Universal Addressing-Modul
San Marino	SM	SMR	Enterprise Geocoding-Modul (<i>San Marino wird durch den italienischen Geocoder abgedeckt.</i>) Universal Addressing-Modul
Sao Tome und Principe	ST	STP	Universal Addressing-Modul
Saudi-Arabien	SA	SAU	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Senegal	SN	SEN	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Serbien	RS	SRB	Enterprise Geocoding-Modul Universal Addressing-Modul
Seychellen	SC	SYC	Universal Addressing-Modul
Sierra Leone	SL	SLE	Universal Addressing-Modul
Singapur	SG	SGP	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Sint Maarten (niederländischer Teil)	SX	SXM	Universal Addressing-Modul
Slowakei	SK	SVK	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Slowenien	SI	SVN	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Salomoninseln	SB	SLB	Universal Addressing-Modul
Somalia	SO	SOM	Universal Addressing-Modul
Südafrika	ZA	ZAF	Enterprise Geocoding-Modul Universal Addressing-Modul GeoComplete-Modul
Südgeorgien und Südliche Sandwichinseln	GS	SGS	Enterprise Geocoding-Modul Universal Addressing-Modul
Südsudan	SS	SSD	Universal Addressing-Modul
Spanien	ES	ESP	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Sri Lanka	LK	LKA	Universal Addressing-Modul
Sudan	SD	SDN	Universal Addressing-Modul
Surinam	SR	SUR	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Svalbard und Jan Mayen	SJ	SJM	Universal Addressing-Modul
Swasiland	SZ	SWZ	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Schweden	SE	SWE	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Schweiz	CH	CHE	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Syrische Arabische Republik	SY	SYR	Universal Addressing-Modul
Taiwan, Provinz Chinas	TW oder zh_TW (Routing)	TWN	Universal Addressing-Modul Enterprise Routing-Modul
Tadschikistan	TJ	TJK	Universal Addressing-Modul
Vereinigte Republik Tansania	TZ	TZA	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul Enterprise Routing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Thailand	TH	THA	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul
Timor-Leste	TL	TLS	Universal Addressing-Modul
Togo	TG	TGO	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Tokelau	TK	TKL	Universal Addressing-Modul
Tonga	TO	TON	Universal Addressing-Modul
Trinidad und Tobago	TT	TTO	Enterprise Geocoding-Modul (Lateinamerika) Universal Addressing-Modul
Tunesien	TN	TUN	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Türkei	TR	TUR	Enterprise Geocoding-Modul Universal Addressing-Modul GeoComplete-Modul
Turkmenistan	TM	TKM	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Turks- und Caicosinseln	TC	TCA	Universal Addressing-Modul
Tuvalu	TV	TUV	Universal Addressing-Modul
Uganda	UG	UGA	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Ukraine	UA	UKR	Enterprise Geocoding-Modul Universal Addressing-Modul
Vereinigte Arabische Emirate	AE	ARE	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul
Großbritannien	GB	GBR	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
Vereinigte Staaten	US	USA	Enterprise Geocoding-Modul Enterprise Routing-Modul Universal Addressing-Modul GeoComplete-Modul
United States Minor Outlying Islands	UM	UMI	Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Uruguay	UY	URY	Enterprise Geocoding-Modul Universal Addressing-Modul
Usbekistan	UZ	UZB	Universal Addressing-Modul
Vanuatu	VU	VUT	Universal Addressing-Modul
Bolivarische Republik Venezuela	VE	VEN	Enterprise Geocoding-Modul Universal Addressing-Modul
Vietnam	VN	VNM	Universal Addressing-Modul
Britische Jungferninseln	VG	VGB	Universal Addressing-Modul
Amerikanische Jungferninseln.	VI	VIR	Universal Addressing-Modul
Wallis und Futuna	WF	WLF	Universal Addressing-Modul
Westsahara	EH	ESH	Universal Addressing-Modul
Jemen	YE	YEM	Enterprise Geocoding-Modul (Mittlerer Osten) Universal Addressing-Modul

ISO-Ländername	ISO 3116-1 Alpha-2	ISO 3116-1 Alpha-3	Unterstützte Module
Sambia	ZM	ZMB	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul
Zimbabwe	ZW	ZWE	Enterprise Geocoding-Modul (Afrika) Universal Addressing-Modul

Notices

© 2017 Pitney Bowes Software Inc. Alle Rechte vorbehalten. MapInfo und Group 1 Software sind Marken von Pitney Bowes Software Inc. Alle anderen Marken und Markenzeichen sind Eigentum ihrer jeweiligen Besitzer.

USPS® Urheberrechtshinweise

Pitney Bowes Inc. wurde eine nicht-ausschließliche Lizenz erteilt, die die Veröffentlichung und den Verkauf von ZIP + 4® Postleitzahl-Datenbanken auf optischen und magnetischen Medien genehmigt. Folgende Marken sind Markenzeichen des United States Postal Service: CASS, CASS Certified, DPV, eLOT, FASTforward, First-Class Mail, Intelligent Mail, LACS^{Link}, NCOA^{Link}, PAVE, PLANET Code, Postal Service, POSTNET, Post Office, RDI, Suite^{Link}, United States Postal Service, Standard Mail, United States Post Office, USPS, ZIP Code, und ZIP + 4. Hierbei handelt es sich jedoch nicht um eine vollständige Liste der Marken, die zum United States Postal Service gehören.

Pitney Bowes Inc. ist nicht-exklusiver Lizenznehmer von USPS® für die Verarbeitungsprozesse von NCOA^{Link}®.

Die Preisgestaltung jeglicher Pitney Bowes Softwareprodukte, -optionen und -dienstleistungen erfolgt nicht durch USPS® oder die Regierung der Vereinigten Staaten. Es wird auch keine Regulierung oder Genehmigung der Preise durch USPS® oder die US-Regierung durchgeführt. Bei der Verwendung von RDI™-Daten zur Berechnung von Paketversandkosten wird die Entscheidung, welcher Paketlieferdienst genutzt wird, nicht von USPS® oder der Regierung der Vereinigten Staaten getroffen.

Datenbereitstellung und Hinweise

Hier verwendete Datenprodukte und Datenprodukte, die in Software-Anwendungen von Pitney Bowes verwendet werden, sind durch verschiedene Markenzeichen und mindestens eines der folgenden Urheberrechte geschützt:

- © Copyright United States Postal Service. Alle Rechte vorbehalten.
- © 2014 TomTom. Alle Rechte vorbehalten. TomTom und das TomTom Logo sind eingetragene Marken von TomTom N.V.
- © 2016 HERE
- Fuente: INEGI (Instituto Nacional de Estadística y Geografía)
- Basierend auf elektronischen Daten © National Land Survey Sweden.
- © Copyright United States Census Bureau
- © Copyright Nova Marketing Group, Inc.
- Teile dieses Programms sind urheberrechtlich geschützt durch © Copyright 1993-2007 Nova Marketing Group Inc. Alle Rechte vorbehalten.
- © Copyright Second Decimal, LLC
- © Copyright Canada Post Corporation

Diese CD-ROM enthält Daten einer urheberrechtlich geschützten Datenerfassung der Canada Post Corporation.

© 2007 Claritas, Inc.

Das Geocode Address World Dataset enthält lizenzierte Daten des GeoNames-Projekts (www.geonames.org), die unter den Bedingungen der Creative Commons Attribution License („Attribution License“) bereitgestellt werden. Die Attribution License können Sie unter <http://creativecommons.org/licenses/by/3.0/legalcode> einsehen. Ihre Nutzung der GeoNames-Daten (wie im Spectrum™ Technology Platform Nutzerhandbuch beschrieben) unterliegt den Bedingungen der Attribution License. Bei Konflikten zwischen Ihrer Vereinbarung mit Pitney Bowes Software, Inc. und der Attribution License hat die Attribution License lediglich bezüglich der Nutzung von GeoNames-Daten Vorrang.



3001 Summer Street
Stamford CT 06926-0700
USA

www.pitneybowes.com