

# Spectrum™ Technology Platform

バージョン 12.0 SP1

Spectrum Spatial 管理ガイド



# 目次

## 1 - はじめに

---

このガイドの内容	5
----------	---

## 2 - システムの設定

---

Spectrum Spatial の HTTP ポート番号の変更	7
リポジトリ データベース タイプの変更	8
サードパーティ ツールを使用したリソースのアップロードとアクセス	9
Web サービスの設定	10
測地系変換用の精度ファイルの無効化	11
リクエスト タイムアウトの設定	12
名前付きテーブルの可変属性の設定	12
Spectrum™ Technology Platformを Linux サービスとして実行	13
MRR 用の Linux マシンの設定	16
デフォルトの HTTP キャッシュ制御ヘッダの無効化	17

## 3 - セキュリティの管理

---

Spectrum™ Technology Platform のセキュリティ	19
Location Intelligence モジュールのセキュリティ	54

## 4 - システムの監視

---

システム イベントの表示	62
Spatial のログ	63

メール サーバーの設定	65
有効期限通知を受信する項目の選択	66
バージョン情報の表示	67
ライセンス情報の表示とエクスポート	67
JMX コンソールによるパフォーマンスのモニタリング	68
JMX コンソールによる、ファイルハンドルキャッシュ統計の監視	69
メモリ使用量の監視	69

## 5 - パフォーマンス チューニング

---

リモート コンポーネントの構成設定	72
データ ソース プーリングの設定	73
距離ベースの操作のパフォーマンスの向上	73

## 6 - クラスタの管理

---

Location Intelligence モジュール用のクラスタアーキテクチャ	76
クラスタでの Enterprise Designer の使用	78
クラスタの起動	79
クラスタの停止	80
クラスタからのノードの削除	80
Location Intelligence モジュール用のクラスタの管理	81

## 7 - 管理ユーティリティの使用

---

管理ユーティリティを使用する前に	88
管理ユーティリティをスクリプトから使用する	89

Location Intelligence モジュール	91
Enterprise Routing モジュール	97

## 8 - Enterprise Routing モジュール

### ル

---

デフォルト サービス/ステージ オプションの指定	121
サービス/ステージのプレビュー	121
Management Console を用いたルート データの取得	124

## 9 - システムのトラブルシュー

### ティング

---

破損したりポジトリ インデックスの再構築	126
反応がなくなったサーバーのメモリ使用量の監視	126

# 1 - はじめに

## このセクションの構成

---

このガイドの内容

5

## このガイドの内容

『*Spectrum Spatial* 管理ガイド』へようこそ。このガイドは、各種 Web サービス、機能、ツール、およびサンプルコードを使用して、Web マッピング アプリケーションを作成したり、既存のアプリケーションにマッピングを組み込むのに役立ちます。

このガイドでは、次の内容について説明します。

- デフォルトのポート番号またはリポジトリ データベースを変更することによるシステムの設定。リポジトリへのアクセス。リソースのアクセスとアップロード。Web サービスの設定。Linux サービスとしての Spectrum™ Technology Platform の実行。
- Management Console を使用したセキュリティの管理。ユーザおよび役割を追加する方法、セキュリティ エンティティのオーバーライドを適用する方法を含みます。
- システムの監視。ログ記録、バージョンとライセンスの情報の表示、JMX コンソールを使用したパフォーマンスの監視、メモリの使用状況の監視などです。
- メモリとスレッドの監視。JVM のパフォーマンスのチューニング、プールサイズの調整、ヒープメモリの追加などです。
- 耐障害性または高いキャパシティのための空間サービスの負荷分散。
- システムのトラブルシューティング。壊れたリポジトリ インデックスの再構築、応答しないサーバーのメモリ使用状況の注意などです。
- User Management Service (将来のリリースで非推奨) を使ったセキュリティの管理

Spectrum™ Technology Platform および Location Intelligence モジュールの他のドキュメントは、[support.pb.com](https://support.pb.com) からオンラインで入手できます。

# 2 - システムの設定

## このセクションの構成

---

Spectrum Spatial の HTTP ポート番号の変更	7
リポジトリ データベース タイプの変更	8
サードパーティ ツールを使用したリソースのアップロードとアクセス	9
Web サービスの設定	10
測地系変換用の精度ファイルの無効化	11
リクエスト タイムアウトの設定	12
名前付きテーブルの可変属性の設定	12
Spectrum™ Technology Platformを Linux サービスとして実行	13
MRR 用の Linux マシンの設定	16
デフォルトの HTTP キャッシュ制御ヘッダの無効化	17

## Spectrum Spatial の HTTP ポート番号の変更

HTTP ポートは、すべての Spectrum™ Technology Platform Web サービスにアクセスするために使用されます。REST と SOAP のどちらにも対応し、ようこそページ、サンプルアプリ、および Spatial Manager のアクセスにも使用します。

Spectrum™ Technology Platform のインストール後、グローバル、起動用、または個別のサービス設定ファイルを手動で編集すれば、インストール中に割り当てられた既存のポート設定を変更できます。ポート番号の変更が必要となる理由はいくつかあります。

- インストール後に、ポートの競合が発生した場合。
- 古い Spectrum™ Technology Platform を削除せずに新バージョンを試してみたい場合。両方をインストールすることはできないため、既存のバージョンを無効にして、別のポートを使用する Spectrum™ Technology Platform イメージをインストールします。
- ポート 8080 にプロキシが必要だが、外部に公開するポート数に制限があるため、設定やデータフローを作成し直すことなく Spectrum™ Technology Platform を移動したい場合。

注：この作業は、アプリケーションサーバーのポート番号を変更したことのある経験豊富な管理者のみが実行してください。ネットワークポートに競合があると、モジュールコンポーネントが起動に失敗する恐れがあります。コンポーネントが Management Console に表示されない場合は、コンポーネントが起動に失敗したことを表しています。この問題をトラブルシューティングするには、Spectrum™ Technology Platform サーバーの wrapper log を確認してください。このログには、どのポートが問題を起こしているかが示されています。wrapper.log は、

<install\_folder>\server\app\repository\logs\wrapper.log にあります。

Spectrum™ Technology Platform を新しい HTTP ポートで実行するには、プロパティと設定ファイルの項目の番号を変更する必要があります。サービス設定を変更するには、WebDAV ファイルの編集がサーバーで有効になっている必要があります。WebDAV は Windows サーバーと Linux サーバーで使用できますが、インストールされている必要があります。

ポート番号を変更するには

1. spectrum-container.properties で、spectrum.http.port の値を新しいポート番号に変更します。このファイルは、<install\_folder>/server/app/conf に配置されています。
2. java.properties ファイルで、すべての repository.host ポートと image.webapp.url を変更します。このファイルは <install\_folder>/server/modules/spatial. にあります。
3. で、次のサービス設定のポート番号を変更します。

- Mapping (SOAP 経由でマッピング サービスにアクセスし、RenderMap 要求の ReturnImage パラメータが False の場合にのみ必要)
- WFS
- WMS
- WMTS

手順については、『*Spectrum Spatial* ガイド』の「ユーティリティ」セクションにある「Spatial Manager ガイド」を参照してください。

別のポートが使えるようにサーバーを移動させる場合は、Spectrum™ Technology Platform サーバーが実行していない可能性があります。このサーバーを実行しないと、サービス設定 ファイルを編集できません。サーバーを起動し、設定を編集し、サーバーを再起動する必要があります。

4. Spectrum™ Technology Platform を再起動してポートとプロパティの変更を適用します。

## リポジトリ データベース タイプの変更

Location Intelligence モジュールは、名前付きリソース (マップ、レイヤ、テーブル、スタイル)、地理メタデータ、構成をリポジトリに保存します。デフォルトのシングルサーバー構成では、組み込みデータベースを使用して、ローカルサーバー上にこれらのリソースを保存します。場合によっては、次のような理由から、組み込みの Derby データベース以外のデータベースを使用することが必要になります。

- 耐障害性を備えた独立したデータベースを使用してスケーラブルなソリューションを構築する場合。
- 会社が推奨または指示する社内データベースを使用する場合。

このリリースではリポジトリ データベースとして Oracle、PostgreSQL/PostGIS、Microsoft SQL Server がサポートされています。手順については、「[共通リポジトリ データベースのセットアップ](#) (81ページ)」を参照してください。



## サードパーティ ツールを使用したリソースのアップロードとアクセス

名前付きリソース ファイルはリポジトリに保存されます。Spectrum™ Technology Platform に同梱されている多数のサンプル ファイルは、<http://<サーバー>:<ポート>/RepositoryService/repository/default/Samples> の特定のフォルダの下にあります。例:

- NamedLayers
- NamedMaps
- NamedStyles
- NamedTables
- NamedTiles

独自の名前付きリソースに対しては、任意のフォルダ名を作成できます。

これらのファイルには、JCR リポジトリにアクセスするための WebDAV プロトコル ツール (WebFolders など) を使用して、手動でアクセスできます。

## WebFolders を使用して Spectrum Spatial リポジトリ リソースにアクセスする

名前付きリソースを追加または変更するには、WebDAV ツールを使用してリソースをリポジトリへ、またはリポジトリからコピーすることができます。WebFolders を使用すると、Spectrum Spatial リポジトリとそれに含まれるリソースに簡単にアクセスできます。

注：リポジトリにアクセスするには、Spectrum™ Technology Platform とリポジトリがインストールされているマシンと同じマシンにログインする必要があります。

Windows 7 で WebFolder を設定するには

1. Windows エクスプローラで、[ネットワークドライブの割り当て] を選択します。
2. ポップアップウィンドウで、[ドキュメントと画像の保存に使用できるWebサイトに接続します] のリンクをクリックして、ネットワークの場所の追加ウィザードを開きます。
3. [次へ] をクリックし、[カスタムのネットワークの場所を選択] を選択します。[次へ] をクリックします。
4. [インターネットまたはネットワークのアドレス] フィールドに、リポジトリ URL を追加します。例えば、<http://<サーバー>:<ポート>>

>/RepositoryService/repository/default/ と入力します。[次へ] をクリックします。

5. 入力を求められた場合は、資格情報 (ユーザ名とパスワード) を入力します。
6. この接続に名前を付けます。例えば、Spectrum Spatial Repository とします。[次へ] をクリックします。

完了すると、指定したネットワーク プレースの下にリポジトリのコンテンツへのフォルダ接続が作成されます。

リポジトリへの WebFolder 接続は、他のすべての Windows エクスプローラと同じように使用できます。

**注：**もしあなたが WebDAV を使って名前付きリソースやメタデータリソースのレコードを変更する場合、同じベース名を同じフォルダには配置しない様に注意してください。そうしないと、リソースに対する移動・名前変更・削除などの操作が、Spatial Manager のメタデータ更新に反映されなくなります。

## Web サービスの設定

各 Web サービスの設定ファイル内の設定値で、Location Intelligence モジュール Web サービスの期待する動作を明示的に指定することができ、時には変更する必要があります。Location Intelligence モジュールでは Web サービスの設定ファイルが、名前付き設定として Location Intelligence モジュールリポジトリに保持されます。

**注：**名前付き設定は、リポジトリ内に保持されている他の名前付きリソースとは異なりません。Named Resource Service を使用して名前付き設定にアクセスすることはできません。代わりに WebFolders などの WebDAV ツールを使用する必要があります。

Mapping、Feature、Map Tiling、WFS、WMS および WMTS の各サービスについては、設定ファイルがリポジトリにあらかじめ読み込まれます。これらの設定ファイルは、<http://hostname:port/RepositoryService/repository/default/Configuration/> に配置されています。

リポジトリ内にある各 Web サービスの名前付き設定の名前と場所、および各 Web サービスの設定パラメータの一覧は、『Spectrum Spatial 開発者ガイド』の「空間サービスの使用」の章を参照してください。

## 測地系変換用の精度ファイルの無効化

Spectrum Spatial は、より正確な座標シフトに役立つアルゴリズムを用いることによって、特定の測地系の間の変換をサポートします。Spectrum のインストール場所 `server\app\types` ディレクトリにある測地系変換ごとに、これらのアルゴリズムを含む個別の jar ファイルがデフォルトでインストールされます。

- `midev-core-coordsys-irishtm-バージョン番号-onprem.jar` (アイルランドの横メルカトル図法用)
- `midev-core-coordsys-jgd2000-バージョン番号-onprem.jar` (更新バージョンである JGD2011 も有効にします) (日本の測地系用)
- `midev-core-coordsys-nadcon-バージョン番号-onprem.jar` (米国 Nad27-Nad83 用)
- `midev-core-coordsys-ntv2-バージョン番号-onprem.jar` (NTV2 用、多数の国の複数の変換を含む)

注：この jar に含まれる XML ファイルによって、使用される変換が制御されます。そのファイルの中の特定の変換を無効にするには、サーバーを停止し、jar ファイルからその XML ファイルを取り出します。エディターを使用して、無効にする各変換のエントリに "false" を設定します。編集した XML ファイルを jar ファイルに追加し直して、サーバーを再起動します。

- `midev-core-coordsys-rgf93-バージョン番号-onprem.jar` (フランスのランベルト変換用)

デフォルトで、これらすべての jar ファイルがロードされますが、それらを使用すると一部の操作のパフォーマンスが低下する可能性があります。これらの変換は、場合によって無効にすることができます。例えば、特定のタイプの変換が不要な場合 (日本の測地系を変換する必要がない場合など) や、ズームレベルが低くて精度よりもパフォーマンスが高い方がメリットがある場合などです。

特定の変換を無効にするには

1. サーバーを停止します。
2. ディレクトリから jar ファイルを削除します。または、jar ファイルを別の拡張子 (.jar~ など) に変更して、読み込みから除外することもできます。
3. サーバーを再起動します。

## リクエスト タイムアウトの設定

Spectrum Spatial では、Mapping Service および Feature Service へのリクエストの一部として SOAP および REST 操作に対するタイムアウトを設定できます。デフォルトでは、このタイムアウトが有効になっていて、300 秒 (5 分) という値に設定されています。

このタイムアウトを適用するには、エン트리および中間のポイントカットを設定しておく必要があります。この設定は、`server/modules/spatial/`にある `aop.xml`で行います。このファイルには、使用できるいくつかの実装が含まれています。エン트리ポイントは、タイムアウトの時間計測が開始されるポイントです。中間ポイントは、操作がタイムアウトしたかどうかのチェックが行われるポイントです。

これを使用するのは、例えば、SOAP および REST の `renderMap` メソッドや一部の中間ステップ (データベースに対する呼び出し、テーブルの検索、候補の取得) にタイムアウトを適用する場合です。

300 秒というデフォルトのタイムアウト値を調整するには、`/server/modules/spatial` の下の `java.properties` 内にある Mapping Service や Feature Service 用のタイムアウト プロパティを編集します。

```
timeout.mapping.value=300
```

```
timeout.feature.value=300
```

指定されたタイムアウト値が 0 以下の場合、タイムアウトは無効になります。

タイムアウト値を変更した後は、Spectrum™ Technology Platform を再起動してください。

## 名前付きテーブルの可変属性の設定

可変性は、データソースからの情報がいつでも変化できることを Spectrum Spatial に対して示します。TAB、SAP HANA、JDBC ベース (Oracle、SQL Server、PostGIS) の名前付きテーブルに対するデフォルト値は `true` に設定されています。これは、クエリや挿入のようなデータアクセス操作のたびに、Spectrum Spatial がそのデータソースをチェックして、テーブルが可変かどうか、可変の場合はデータが変化したかどうかを確認します。データが変化している場合、データアクセス操作を続行する前に、キャッシュはフラッシュされ、テーブルは再ロードされます。テーブルが変化していない場合、クエリやその他の操作はキャッシュ内のデータに対して実行されます。

各データソースに変化を引き起こす要因の詳細については、『*Spectrum Spatial* ガイド』の「リソースおよびデータ」セクションの「サポートされるデータソース」を参照してください。

**Map Uploader** を使用して MapInfo Professional からアップロードされた名前付きテーブルは、可変性が有効になります (true に設定されます)。**Spatial Manager** によって作成されたすべての名前付きテーブルについても、可変性は有効です。リポジトリ内の古い名前付きテーブルは可変であると見なされますが、**Spatial Manager** のテーブルの詳細ページに表示される場合にはこの情報が示されません。

変更しないテーブルに対してしか、可変性を無効にはしてはいけません。例えば、可変 TAB ファイルからタイルを生成する操作は、非常に時間がかかります。PostGIS を使用する場合は、可変性を無効にして **Spatial Manager** での接続エラーの発生を避けることも検討するとよいでしょう (例えば、テーブルの詳細ページでサンプル行を表示する場合)。

可変性は、**Spatial Manager** のテーブルの詳細ページで無効にできます。**Spatial Manager** での名前付きテーブルの作成と変更については、『*Spectrum Spatial* ガイド』の「ユーティリティ」セクションを参照してください。

既存の名前付きテーブルの可変性設定を変更したり、それまで false に設定されていた (つまり可変性が無効だった) データベース テーブルに基づいて新しい名前付きテーブルを作成したりしたときには、サーバーを再起動する必要があります。

注：テキスト エディタ内でのこの値の変更、または WebDAV を介してアクセスした名前付きテーブル定義の手動編集には、Named Resource Service の updateNamedResource 操作を使用しないでください。

## Spectrum™ Technology Platform を Linux サービスとして実行

このチュートリアルでは、Spectrum™ Technology Platform を Linux サービスとして実行するために必要な手順を説明します。

### Spectrum™ Technology Platform を Linux サービスとして実行する方法

この手順は、Spectrum™ Technology Platform を Linux サービスとして実行する方法を説明しています。

1. 提供されている pbspectrum スクリプト (**PBSpectrum スクリプト** (15ページ) にあります) を変更します。
  - a) 5行目の chkconfig パラメータを変更します。デフォルトでは、このパラメータは次のようになっています。 # chkconfig: 35 90 10  
1 番目の値 (35) はランレベルです。詳細については、'man init' を使用してください。  
2 番目の値 (90) は起動の優先順位です。  
3 番目の値 (10) は停止の優先順位です。  
起動と停止の優先順位は、依存関係にあるサービスに応じて設定します。例えば、同じコンピュータで実行される Oracle サーバーを Spectrum™ Technology Platform で使用する場合、Spectrum™ Technology Platform の起動の優先順位を Oracle サービスより低く、停止の優先順位を Oracle サービスより高くする必要があります。詳細については、'man chkconfig' を使用してください。
  - b) 11 行目の SPECTRUM\_ROOT 変数を Spectrum™ Technology Platform のインストールディレクトリに変更します。
  - c) SUSE Linux を使用している場合は、デフォルトの優先ユーザを su から runuser に変更する必要があります。
2. 変更した pbspectrum スクリプトを、RedHat Linux の場合は /etc/rc.d/init.d に、Suse Linux の場合は /etc/init.d にコピーします。
3. pbspectrum スクリプトのモードを実行可能に変更します。RedHat Linux の場合は /etc/rc.d/init.d、Suse Linux の場合は /etc/init.d です。  
cd /etc/init.d 使用する Linux のバージョンによっては cd /etc/rc.d/init.d の場合もあります。  
実行 `chmod +x pbspectrum`
4. 実行 `chkconfig --add pbspectrum`
5. コンピュータを再起動してスクリプトの動作を確認します。シェルから再起動するには、`shutdown -r now` を使用します。

完了後は、次の操作も利用できます。

- `service pbspectrum start` Spatial Server の起動
- `service pbspectrum stop` Spatial Server の停止
- `service pbspectrum restart` Spatial Server の再起動

注：提供されているスクリプトで実行される 'ulimit -n 8192' コマンドは、Linux でオープンファイルの数を増加させるために必要なコマンドです。

## PBSpectrum スクリプト

次のスクリプトは、この手順の基盤として使用されます。 [Spectrum™ Technology Platform を Linux サービスとして実行する方法 \(13ページ\)](#)

```
#!/bin/bash
#
# pbspectrum Bring up/down PB Spectrum platform
#
# chkconfig: 35 90 10
# description: Starts and stops the spectrum
#
# /etc/rc.d/init.d/pbspectrum
# See how we were called.

SPECTRUM_ROOT=/root/PBSpectrum

start() {
    su - spectrum -c ". $SPECTRUM_ROOT/server/bin/setup;
    ulimit -n 8192;
    $SPECTRUM_ROOT/server/bin/server.start"
    RETVAL=$?
    return $RETVAL
}

stop() {
    su - spectrum -c ". $SPECTRUM_ROOT/server/bin/setup;
    $SPECTRUM_ROOT/server/bin/server.stop"
    RETVAL=$?
    return $RETVAL
}

# See how we were called.
case "$1" in
    start)
        start
        ;;
    stop)
        stop
        ;;
    restart)
        stop
        start
        ;;
    *)
        echo $"Usage: pbspectrum {start|stop|restart}"
        exit 1
esac
```



```
exit $RETVAL
```

## MRR 用の Linux マシンの設定

Linux 環境の Spectrum Spatial 上で MRR (Multi Resolution Raster) ファイルを使用するには、GCC と LIBC を適切なバージョンにアップグレードする必要があります。

MRR 用に Linux マシンを設定するには

1. UUID パッケージをインストールします。これによって、LIBC v.2.17 がインストールされます。

例えば、Cent OS 上に UUID をインストールするには次のようにします。

- wget  
http://ftp.riken.jp/Linux/centos/6/os/x86\_64/Packages/libuuid-2.17.2-12.18.el6.x86\_64.rpm
- sudo yum -y install libuuid-2.17.2-12.18.el6.x86\_64.rpm
- sudo yum -y install libuuid-devel

2. devtoolset-3 をインストールします。これによって、GCC v.4.9 がインストールされます。詳細については、<https://www.softwarecollections.org/en/scls/rhscsl/devtoolset-3/>を参照してください。

3. GCC v.4.9 と LIBC v.2.17 (またはそれ以降) がインストールされていることを確認します。

4. 上記のステップで、すべての依存関係が解決されていることを確認します。解決されていない依存関係がある場合は、それをインストールしてステップ 2 を繰り返します。

例えば以下は、OEL 6.5 コンピュータに対して必要な依存関係の一部です。

- wget https://www.softwarecollections.org/en/scls/mizdebsk/maven30-rhel-6/epel-6-x86\_64/download/mizdebsk-maven30-rhel-6-epel-6-x86\_64.noarch.rpm
- sudo yum -y install mizdebsk-maven30-rhel-6-epel-6-x86\_64-1-2.noarch.rpm
- wget https://www.softwarecollections.org/en/scls/rhscsl/maven30/epel-6-x86\_64/download/rhscsl-maven30-epel-6-x86\_64.noarch.rpm
- sudo yum -y install rhscsl-maven30-epel-6-x86\_64-1-2.noarch.rpm
- sudo yum -y install maven30
- wget https://www.softwarecollections.org/en/scls/mbooth/eclipse-luna/fedora-20-x86\_64/download/mbooth-eclipse-luna-fedora-20-x86\_64.noarch.rpm



- `sudo yum -y install mbooth-eclipse-luna-fedora-20-x86_64-1-2.noarch.rpm`
- `sudo yum -y install --skip-broken eclipse-luna`

## デフォルトの HTTP キャッシュ制御ヘッダの無効化

デフォルトで、Spectrum™ Technology Platform Web サービスは、次の HTTP ヘッダをキャッシュに使用します。

```
Cache-Control: no-cache,no-store,no-transform,must-revalidate
Expires: Wed, 07 Jan 2015 15:38:03 GMT //48 hours in the past
Pragma: no-cache
```

これらの HTTP ヘッダは、Map Tiling サービスには適していません。ただし、これらのデフォルトの HTTP ヘッダを無効にして、個々の Web サービスで定義されるヘッダで HTTP キャッシュ動作を設定することができます。

注：この変更をクラスタに適用する場合は、クラスタ内の各ノードに対して以下の手順を繰り返す必要があります。

デフォルトの HTTP キャッシュ制御ヘッダを無効にするには

1. Spectrum™ Technology Platform サーバーを停止します。
2. この設定ファイルをテキスト エディタで開きます。  
`SpectrumFolder\server\app\conf\spectrum-advanced.properties`
3. 次のプロパティを `true` から `false` に変更します。

```
spectrum.cache.control.headers.enable=false
```

4. プロパティ ファイルを保存して閉じます。
5. Spectrum™ Technology Platform サーバーを開始します。

# 3 - セキュリティの管理

Location Intelligence モジュールは、Spectrum™ Technology Platform に対して使用されるものと同じ役割ベースのセキュリティ モデルを使用します。セキュリティはプラットフォーム レベルで処理されるので、Location Intelligence モジュールのすべてのセキュリティ アクティビティの管理には Management Console を使用できます。

## このセクションの構成

---

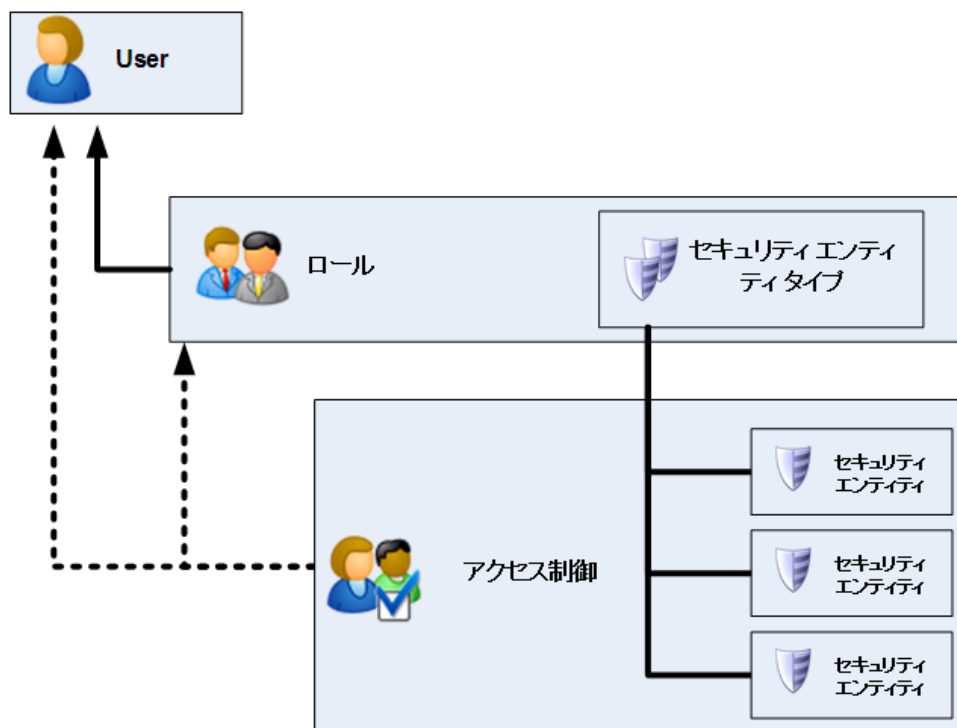
Spectrum™ Technology Platform のセキュリティ	19
Location Intelligence モジュールのセキュリティ	54

## Spectrum™ Technology Platform のセキュリティ

ここでは、すべてのモジュールに関連するプラットフォームレベルでのセキュリティモデルおよび手順について説明します。このモジュールに固有のセキュリティ情報の詳細については、[Location Intelligence モジュールのセキュリティ](#)（54ページ）を参照してください。

### セキュリティ モデル

Spectrum™ Technology Platform は、役割ベースのセキュリティ モデルを使用して、システムへのアクセスを制御します。以下の図に、Spectrum™ Technology Platform セキュリティ モデルの重要な概念を示します。



ユーザとは、各個人に割り当てられるアカウントのことで、Spectrum™ Technology Platform に対する認証、つまり Enterprise Designer や Management Console などのクライアント ツールに対する認証、または Web サービスや API を介してサービス呼び出すときの認証で使用されます。

ユーザには 1 つ以上の役割が割り当てられます。役割とは、システムのさまざまな部分に対するアクセスを付与または拒否する権限の集合です。通常、役割は、特定のタイプのユーザがシステ

ムとの間で行う対話操作の種類を表します。例えば、データフローの作成および変更アクセスを付与する、データフロー設計者向けの役割や、既存のデータフローを利用したデータ処理のみを必要とするユーザ向けの役割があります。

役割は、セキュアエンティティタイプに権限を付与します。セキュアエンティティタイプとは、アクセスを付与または拒否するアイテムのカテゴリです。例えば、"データフロー"というセキュアエンティティタイプは、システム上のすべてのデータフローに対するデフォルトの権限を制御します。

アクセスをきめ細かく制御する必要がある場合は、アクセス制御を設定すれば役割またはユーザの設定をオーバーライドすることができます。アクセス制御設定は、役割と連動して、ユーザの権限を定義します。役割は、すべてのデータフローやすべてのデータベースリソースなど、エンティティのカテゴリに対する権限を定義し、アクセス制御設定は、セキュアエンティティと呼ばれる特定のエンティティに対する権限を定義します。セキュアエンティティの例として、特定のジョブや特定のデータベース接続などがあります。アクセス制御設定の定義は必須ではありません。アクセス制御設定を定義しない場合は、役割に定義された権限により、ユーザの権限が制御されます。

アクセス制御設定は、役割と連動して、ユーザの権限を定義します。役割は、すべてのデータフローやすべてのデータベースリソースなど、エンティティのカテゴリに対する権限を定義し、アクセス制御設定は、セキュアエンティティと呼ばれる特定のエンティティに対する権限を定義します。セキュアエンティティの例として、特定のジョブや特定のデータベース接続などがあります。例えば、ある役割によって"データフロー"というセキュアエンティティタイプに対する変更権限が付与されているが、特定のデータフローをユーザが変更できないようにしたいとします。アクセス制御を使用して変更したくない特定のデータフローへの変更権限を削除することにより、これが可能になります。アクセス制御設定は、ユーザまたは役割に対して指定できます。ユーザに対するアクセス制御設定は、ユーザの役割によって付与された特定のユーザの権限をオーバーライドします。役割に対するアクセス制御設定は、その役割を持つすべてのユーザに適用されません。

## ユーザ

Spectrum™ Technology Platform ユーザアカウントは、ユーザがシステムで実行できるアクションのタイプを制御します。以下のアクションを行うには、ユーザアカウントが必要です。

- Management Console、Enterprise Designer、Metadata Insights、コマンドラインツールなどのツールを使用する。
- ジョブをスケジュールどおりに実行する。
- ジョブをコマンドラインから実行する。
- Web サービスまたは API を介してサービスにアクセスする。


**admin** という管理アカウントが付属しています。このアカウントは、完全アクセスを持ちます。初期パスワードは "admin" です。

**重要：** システムに対して不正な管理者アクセスが行われることを回避するために、Spectrum™ Technology Platform をインストールした後すぐに **admin** パスワードを変更してください。

ユーザ アカウントは必要に応じていくつでも作成できます。

## ユーザの追加

この手順では、Spectrum™ Technology Platform ユーザ アカウントを作成し、そのユーザ アカウントに役割を割り当てる方法について説明します。

1. Management Console を開きます。
2. [システム] > [セキュリティ] を選択します。
3. [追加] ボタン  をクリックします。
4. このユーザ アカウントを使用できるようにする場合は、[有効] スイッチを [オン] のままにしておきます。
5. [ユーザ名] フィールドにユーザ名を入力します。

注：ユーザ名には ASCII 文字のみを使用できます。ユーザ名は大文字と小文字が区別されます。

6. [電子メールアドレス] フィールドにユーザの電子メールアドレスを入力します。電子メールアドレスは、ユーザに通知を送信するために一部のモジュールで使用されます。
7. [説明] フィールドにユーザの説明を入力します。
8. ユーザのパスワードを入力し、確認のためにもう一度入力します。
9. このユーザに付与する役割を選択します。

独自の役割を作成することも、デフォルトの役割を使用することもできます。デフォルトの役割は次のとおりです。

- |                   |  |
|-------------------|--|
| <b>admin</b>      | この役割は、システムのすべての部分に対する完全アクセスを持ちます。  |
| <b>designer</b>   | この役割は、Enterprise Designer でデータフローとプロセスフローを作成するユーザを対象としています。この役割では、データフローを設計および実行できます。  |
| <b>integrator</b> | この役割は、Spectrum™ Technology Platform を介してデータを処理する必要があるが、データフローを作成または変更する必要のないユーザを対象としています。この役割では、Web サービスと API を介してサービスにアクセスし、ジョブを実行できます。 |


**user** これはデフォルトの役割です。この役割は、システムに対するアクセスを持ちません。この役割を持つユーザは、セキュアエンティティオーバーライドを介して権限が付与された場合にのみシステムにアクセスできます。

役割の作成については、[役割の作成](#)（26ページ）を参照してください。

10. **[保存]** をクリックします。

## パスワードの変更

この手順は、ユーザのパスワードの変更方法を示しています。

1. Management Console を開きます。
2. **[システム]** > **[セキュリティ]** を選択します。
3. ユーザを選択し、編集ボタン  をクリックします。
4. **[パスワードの変更]** をクリックします。
5. 新しいパスワードを入力し、確認のためにもう一度入力します。
6. **[保存]** をクリックします。

## 最小パスワード長の設定

この最小パスワード長は、パスワードの作成または変更時に適用されます。既存のパスワードについては、この最小長よりも短いものであっても引き続き有効となります。

1. Web ブラウザを開いて に移動します。 `http://server:port/jmx-console`

WHERE:

*server* は、Spectrum™ Technology Platform サーバーの IP アドレスまたはホスト名です。

*port* は、Spectrum™ Technology Platform が使用する HTTP ポートです。デフォルト値は 8080 です。

2. 管理者アカウントでログインします。
3. "Domain: com.pb.spectrum.platform.config" の下にある **[com.pb.spectrum.platform.config:manager=AccountConfigurationManager]** をクリックします。
4. **[updatePasswordPolicy]** 操作の **[enableAdvanceControl]** オプションを **[True]** に設定します。
5. **[minLength]** フィールドに最小パスワード長を入力します。
6. **[Invoke]** をクリックします。
7. **[Return to MBean View]** をクリックして、アカウント構成マネージャーの画面に戻ります。

## 電子メールアドレスの変更


ユーザ アカウントに関連付けられた電子メールアドレスは、ユーザに通知を送信する一部のモジュールによって使用されます。電子メールアドレスを変更するには、以下の手順に従います。

1. Management Console にログインします。
2. 右上隅のユーザ メニューをクリックします。
3. **[プロフィール]** を選択します。
4. **[電子メール]** フィールドに、新しい電子メールアドレスを入力します。
5. **[保存]** をクリックします。

## ユーザ アカウントの無効化

ユーザ アカウントを無効にすると、そのユーザ アカウントを Spectrum™ Technology Platform へのアクセスに使用できなくなります。無効にされたユーザ アカウントを使用して実行がスケジュールされているジョブもすべて実行されなくなります。

注: "admin" ユーザ アカウントは無効にできません。


1. Management Console を開きます。
2. **[システム]** > **[セキュリティ]** を選択します。
3. 変更するユーザの横にあるチェックボックスをオンにし、編集ボタン  をクリックします。
4. **[有効]** スイッチを **[オフ]** にします。
5. **[保存]** をクリックします。

これで、ユーザ アカウントが無効になり、Spectrum™ Technology Platform へのアクセスに使用できなくなります。

## ユーザの削除

この手順では、Spectrum™ Technology Platform ユーザ アカウントを完全に削除する方法について説明します。

ヒント: また、ユーザ アカウントは無効にすることもできます。ユーザ アカウントを無効にすると、アカウントを削除しなくても、システムへのアクセスにそのアカウントを使用できなくなります。

1. Management Console を開きます。
2. **[システム]** > **[セキュリティ]** を選択します。
3. 削除するユーザの横にあるチェックボックスをオンにし、削除ボタン  をクリックします。

注: "admin" ユーザ アカウントは削除できません。



## ユーザアカウントのロック

セキュリティ対策として、ユーザの認証が5回連続で失敗した場合はユーザアカウントが無効になります。この回数には、Enterprise Designer、Management Console、Web サービス、またはクライアント API への認証失敗も含まれます。

管理者は、Management Console にログインし、ユーザを編集して **[有効]** スイッチを **[オン]** に切り替えてユーザアカウントを再び有効にすることができます。ユーザアカウントは、管理ユーティリティを使って有効にすることもできます。ユーザ自身がアカウントのロックを解除することはできません。

注：認証に LDAP または Active Directory を使っている場合は、それらのサービスのアカウント ロック ルールが適用されます。LDAP または Active Directory のルールでは、許容されるログイン失敗の回数が Spectrum™ Technology Platform よりも少ない場合があります。

## admin アカウントのロック解除

ログインに数回失敗すると、ユーザアカウントがロックされます。大部分のユーザアカウントは Management Console でのロック解除できますが、admin アカウントはこの方法でロック解除できません。admin アカウントをロック解除するには、サーバー上でスクリプトを実行する必要があります。

1. Spectrum™ Technology Platform を実行しているサーバーにログインします。

クラスタで Spectrum™ Technology Platform を実行している場合は、いずれかのノードにログインします。ロック解除スクリプトは、いずれか1つのノードで実行するだけで済みます。

2. コマンド プロンプトを開き、*Spectrum Folder\server\bin* フォルダに移動します。
3. (Unix および Linux のみ) 次のコマンドを実行します。

```
./setup
```

4. 次のコマンドを入力して、enableadmin スクリプトを実行します。

Windows の場合:

```
enableadmin.bat -h HostAndPort -p AdminPassword [-s]
```

UNIX/Linux の場合:

```
./enableadmin.sh -h HostAndPort -p AdminPassword [-s]
```

説明:

**HostAndPort** Spectrum™ Technology Platform が使用するホスト名と HTTP ポートです。例: spectrumserver:8080



- AdminPassword** admin アカウントのパスワードです。ロックされた admin アカウントのパスワードが不明な場合は、Pitney Bowes テクニカル サポートまでご連絡ください。
- s** Spectrum™ Technology Platform が HTTPS を使用するように設定されている場合は、`-s` を指定します。

## 非アクティブ状態による自動ログアウト

Enterprise Designer や Management Console、Relationship Analysis Client、Business Steward Portal などの Web クライアントのユーザは、30 分間動作がない状態が続くと自動的にログアウトされます。

## 役割

役割とは、システムのさまざまな部分に対するアクセスを付与または拒否する権限の集合です。通常、役割は、特定のタイプのユーザがシステムとの間で行う対話操作の種類を表します。例えば、データフローの作成および変更アクセスを付与する、データフロー設計者向けの役割や、既存のデータフローを利用したデータ処理のみを必要とするユーザ向けの役割があります。

Spectrum™ Technology Platform では以下の役割が事前に定義されています。

- admin** この役割は、システムのすべての部分に対する完全アクセスを持ちます。
- designer** この役割は、Enterprise Designer でデータフローとプロセスフローを作成するユーザを対象としています。この役割では、データフローを設計および実行できます。
- integrator** この役割は、Spectrum™ Technology Platform を介してデータを処理する必要があるが、データフローを作成または変更する必要のないユーザを対象としています。この役割では、Web サービスと API を介してサービスにアクセスし、ジョブを実行できます。
- user** これはデフォルトの役割です。この役割は、システムに対するアクセスを持ちません。この役割を持つユーザは、セキュア エンティティ オーバーライドを介して権限が付与された場合にのみシステムにアクセスできます。


注：Location Intelligence モジュールの定義済みの役割については、[Location Intelligence モジュールのセキュリティ](#)（54ページ）を参照してください。


これらの各役割に付与されている権限を表示するには、Management Console を開き、**[セキュリティ]** に移動して **[役割]** をクリックします。次に、表示する役割を選択し、**[表示]** をクリックします。

ヒント：あらかじめ定義されている役割は変更できません。ただし、あらかじめ定義されている役割を基に新しい役割をできます。

## 役割の作成

役割とは、ユーザに割り当てる権限の集合です。Spectrum™ Technology Platform にあらかじめ定義されている役割が組織のニーズに合わない場合は、独自の役割を作成できます。

1. Management Console を開きます。
2. [システム] > [セキュリティ] を選択します。
3. [役割] をクリックします。
4. [追加] ボタン  をクリックします。

ヒント：既存の役割とよく似た役割を作成する場合は、コピーしたい役割の横にあるチェックボックスをオンにしてからコピー ボタン  をクリックすると、既存の役割のコピーを作成できます。続いて、その新しい役割を編集してから以下の手順に進みます。

5. [役割名] フィールドに、この役割に付与する名前を入力します。任意の名前にすることができます。
6. オプション:セキュアエンティティタイプの一覧は長くなることもあり、その場合は、セキュアエンティティタイプの特定のグループのみを表示することもできます。同じ権限をグループ内のすべてのエンティティに適用する場合は、その特定のグループのみを表示すると便利です。例えば、すべてのデータベースリソースから変更権限を削除する場合は、フィルタリングしてデータベースリソースグループのみを表示できます。1つのグループのみを表示して変更するには:
  - a) [グループのフィルタリングを有効にする] ボックスをオンにします。
  - b) [グループ] 列のヘッダにある漏斗（ろうと）アイコンをクリックし、表示するグループを選択します。
  - c) 適用する権限の列ヘッダにあるボックスをオンまたはオフにします。
  - d) セキュアエンティティタイプの完全な一覧に戻るには、フィルタアイコンをクリックし、[[すべて]] を選択して、[グループのフィルタリングを有効にする] ボックスをオフにします。
7. 各エンティティタイプに付与する権限を選択します。権限は以下のとおりです。

**ビュー** ユーザは、エンティティタイプに含まれるエンティティを表示できます。例えば、JDBC 接続エンティティタイプに対する表示権限を許可すると、この役割を持つユーザは Management Console でデータベース接続を表示できます。

**変更** ユーザは、エンティティタイプに含まれるエンティティを変更できます。例えば、JDBC 接続エンティティタイプに対する変更権限を許可すると、この役割を持つユーザは Management Console でデータベース接続を変更できます。

作成	ユーザは、このエンティティ タイプのカテゴリに分類されるエンティティを作成できます。例えば、JDBC 接続エンティティ タイプに対する作成権限を許可すると、この役割を持つユーザは Management Console で新しいデータベース接続を作成できます。
削除	ユーザは、エンティティ タイプに含まれるエンティティを削除できます。例えば、JDBC 接続エンティティ タイプに対する削除権限を許可すると、この役割を持つユーザは Management Console でデータベース接続を削除できます。
実行	ユーザは、ジョブ、サービス、プロセスフローの処理を開始できます。例えば、ジョブエンティティ タイプに対する実行権限を許可すると、この役割を持つユーザはバッチ ジョブを実行できます。例えば、サービス エンティティ タイプに対する実行権限を許可すると、この役割を持つユーザは Spectrum™ Technology Platform 上で実行されるサービスに API または Web サービスを介してアクセスできます。

## 8. [保存] をクリックします。

これで、ユーザに役割を割り当てることができます。

## 役割の削除

どのユーザにも割り当てられていない役割は、削除できます。

注：ただし、admin、user、designer、integrator の役割は削除できません。

1. Management Console を開きます。
2. [システム] > [セキュリティ] を選択します。
3. [ユーザ] タブで、削除したい役割がどのユーザにも割り当てられていないことを確認します。ユーザに割り当てられている役割は、削除できません。
4. [役割] をクリックします。
5. 削除する役割の横にあるチェックボックスをオンにし、削除ボタン  をクリックします。

## 役割ベースのセキュリティの無効化

役割ベースのセキュリティは、デフォルトで有効になっています。つまり、役割を通じてユーザに割り当てられているセキュリティ制約が適用されます。役割ベースのセキュリティを無効にした場合、ユーザに割り当てられているセキュリティ制約は適用されなくなり、すべてのユーザがシステムのすべての部分にアクセスできるようになります。役割ベースのセキュリティを無効にした場合でも、サービスにアクセスするには有効なユーザアカウントが必ず必要であることに注意してください。

この手順は、役割ベースのセキュリティを無効化する方法を示しています。

**警告：** この手順を実行した場合、すべてのユーザが Spectrum™ Technology Platform システムへの完全アクセスを持つようになります。

1. Management Console を開きます。
2. **[システム]** > **[セキュリティ]** を選択します。
3. **[役割による制限アクセス]** スイッチを **[オフ]** にします。

## セキュア エンティティ タイプ - プラットフォーム

エンティティ タイプとは、アクセスを付与または拒否するアイテムのカテゴリです。例えば、"データフロー" というエンティティ タイプは、システム上のすべてのデータフローに対する権限を制御します。プラットフォーム エンティティ タイプは、Spectrum™ Technology Platform のすべてのインストールに適用され、モジュール固有のエンティティ タイプは、その特定のモジュールをインストールした場合にのみ適用されます。プラットフォームレベルのエンティティ タイプを以下に示します。

**監査ログ** Management Console の **[システム]** > **[ログ]** > **[監査ログ]** エリアへのアクセスを制御します。

**データフロー** Enterprise Designer ですべてのデータフロー (ジョブ、サービス、およびサブフロー) へのアクセスを制御します。

**注：** 編集権限がないユーザには、エクスポートされたバージョンと最終保存バージョンのみが Enterprise Designer の **[バージョン]** ウィンドウに表示されます。

**データフロー - エクスポート** データフローを実行のために使用可能にできるかどうかを制御します。

**注：** データフローの最終保存バージョン (Enterprise Designer の **[バージョン]** ウィンドウで常に最上位にあるバージョン) をエクスポートするためには、ユーザが **[データフロー - エクスポート]** セキュア エンティティ タイプに対する編集権限に加えて **[データフロー]** セキュア エンティティ タイプに対する編集権限も持っている必要があります。最終保存バージョンは、エクスポート前にまずバージョンとして保存する必要があります。そのためにはデータフローに対する編集権限が必要になるからです。

**フローのデフォルト - データ タイプの変換** Management Console の **[フロー]** > **[デフォルト]** > **[データ タイプの変換]** エリアへのアクセスを制御します。すべてのユーザがデータ タイプの変換オプションに対する View アクセスを持ちます。View アクセスを削除することはできません。

フローのデフォルト - 形式に誤りのあるレコード	Management Console の <b>[フロー] &gt; [デフォルト] &gt; [形式に誤りのあるレコード]</b> エリアへのアクセスを制御します。すべてのユーザが形式に誤りのあるレコード オプションに対する <b>View</b> アクセスを持ちます。View アクセスを削除することはできません。
フローのデフォルト - レポート	Management Console の <b>[フロー] &gt; [デフォルト] &gt; [レポート]</b> エリアへのアクセスを制御します。すべてのユーザがレポートオプションに対する <b>View</b> アクセスを持ちます。View アクセスを削除することはできません。
フローのデフォルト - ソート パフォーマンス	Management Console の <b>[フロー] &gt; [デフォルト] &gt; [ソート パフォーマンス]</b> エリアへのアクセスを制御します。すべてのユーザがソート パフォーマンス オプションに対する <b>View</b> アクセスを持ちます。View アクセスを削除することはできません。
フロー履歴 - ジョブ	Enterprise Designer と Management Console のジョブ実行履歴への表示アクセスを制御します。
フロー履歴 - プロセス フロー	Management Console と Enterprise Designer でプロセス フロー実行履歴へのアクセスを制御します。
フロー履歴 - トランザクション	Management Console の <b>[フロー] &gt; [履歴] &gt; [トランザクション]</b> エリアへのアクセスを制御します。
フロー スケジューリング	Management Console の <b>[フロー] &gt; [スケジュール]</b> エリアへのアクセスを制御します。
ジョブ	Enterprise Designer、Management Console、Job Executor、および管理ユーティリティでのジョブ実行を制御します。
通知 - ライセンス有効期限	Management Console でライセンス有効期限通知の電子メール設定へのアクセスを制御します。
通知 - SMTP 設定	Management Console の <b>[システム] &gt; [メール サーバー]</b> エリアへのアクセスを制御します。
プロセス フロー	Enterprise Designer でプロセス フローへのアクセスを制御します。  注：編集権限がないユーザには、エクスポートされたバージョンと最終保存バージョンのみが Enterprise Designer の <b>[バージョン]</b> ウィンドウに表示されます。
プロセス フロー - エクスポート	プロセス フローを実行でできるようにする Enterprise Designer の機能を制御します。  注：プロセスフローの最終保存バージョン (Enterprise Designer の <b>[バージョン]</b> ウィンドウで常に最上位にあるバージョン) をエクスポートするためには、ユーザが <b>[プロセス フロー - エクスポート]</b> セキュア エンティティタイプに対する編集権限に加えて <b>[プロセス フロー]</b> セキュア エンティティタイプに対する編集権限も持っている必要があります。最終保存バージョン



は、エクスポート前にまずバージョンとして保存する必要があり、そのためにはデータフローに対する編集権限が必要になるからです。

リソース - データベース接続	Management Console でデータベース接続を設定する機能を制御します。
リソース - 外部の Web サービス	Management Console で外部の Web サービスの管理へのアクセスを制御します。
リソース - ファイル サーバー接続	Management Console でファイルサーバーを設定する機能を制御します。
リソース - JDBC ドライバ	Management Console で JDBC ドライバを設定する機能を制御します。
リソース - リモート サーバー	Management Console の [リソース] > [リモート サーバー] エリアへのアクセスを制御します。
セキュリティ - アクセス制御	Management Console の [システム] > [セキュリティ] > [アクセス制御] エリアのアクセス制御設定へのアクセスを制御します。
セキュリティ - アクセス トークン	ユーザのトークンを表示し、トークンを削除する機能を制御します。トークンは、クライアントとサーバーとの間の認証に使用されます。読み取り権限があれば、それぞれアクティブなセッションを表すアクティブなトークンのリストを表示できます。削除権限があれば、ユーザのトークンを削除し、そのセッションを終了させることができます。
セキュリティ - ディレク トリ アクセス	Management Console の [システム] > [セキュリティ] > [ディレクトリ アクセス] エリアを使ってサーバー ディレクトリ リソースの制限を有効または無効にする機能を制御します。
セキュリティ - ディレク トリ パス	Management Console の [システム] > [セキュリティ] > [ディレクトリ アクセス] エリアでサーバー ディレクトリ リソースを設定する機能を制御します。
セキュリティ - オプション	Management Console の [システム] > [セキュリティ] > [役割] エリアでセキュリティを有効または無効に切り替える機能へのアクセスを制御します。
セキュリティ - 役割	Management Console の [システム] > [セキュリティ] > [役割] エリアの役割設定機能へのアクセスを制御します。
セキュリティ - ディレク トリ パス	Management Console の [システム] > [セキュリティ] > [ディレクトリ アクセス] エリアでサーバー ディレクトリ リソースを設定する機能を制御します。
セキュリティ - ユーザ	Management Console の [システム] > [セキュリティ] > [ユーザ] エリアのユーザ アカウント管理機能へのアクセスを制御します。
サービス	API と Web サービスを介してサービスを実行する機能を制御します。

ステージ	エクスポートされたサブフローを Enterprise Designer のデータフローのステージとして使用できるかどうかを制御します。
システム - ライセンス	Management Console の [システム] > [ライセンスと有効期限] エリアに表示されるライセンス情報へのアクセスを制御します。
システム - バージョン情報	Management Console の [システム] > [バージョン] エリアへのアクセスを制御します。
システム ログ	Management Console のシステム ログへのアクセスを制御します。

## セキュア エンティティ タイプ - Location Intelligence モジュール

エンティティ タイプとは、アクセスを付与または拒否するアイテムのカテゴリです。Location Intelligence モジュールには、以下のモジュール固有のエンティティ タイプがあります。

**名前付きリソース** Location Intelligence モジュール内のすべての名前付きリソースに対する権限を制御します。Location Intelligence モジュール サービスのユーザには、最低限、使用するリソースとその従属リソースに対する読み込み権限が必要です。(Spatial Manager、管理ユーティリティ、Named Resource Service、WebDAV などの任意のツールを使用して) 名前付きリソースが作成されると、新しい LocationIntelligence.Named Resource セキュア エンティティがその名前付きリソースに対して自動的に作成されます。

**Dataset.DML** Location Intelligence モジュール内で使用される、名前付きテーブルに関連付けられたデータセットに対する権限を制御します。(Spatial Manager、管理ユーティリティ、Named Resource Service、WebDAV などの任意のツールを使用して) 名前付きテーブルが作成またはアップロードされると、新しい LocationIntelligence.Dataset セキュア エンティティがその名前付きテーブルに対して自動的に作成されます。ユーザが、書き込み可能 (JDBC ベース) テーブルで DML 操作を行うには、名前付きテーブルに対する表示権限、およびデータセットに対する作成/変更/削除権限が必要です。DML 操作には挿入、更新、および削除の操作があり、Write Spatial Data ステージまたは Feature Service を使用して実行します。

## アクセス制御

アクセス制御設定は、役割と連動して、ユーザの権限を定義します。役割は、すべてのデータフローやすべてのデータベースリソースなど、エンティティのカテゴリに対する権限を定義し、アクセス制御設定は、セキュア エンティティと呼ばれる特定のエンティティに対する権限を定義します。セキュア エンティティの例として、特定のジョブや特定のデータベース接続などがあります。例えば、ある役割によって "データフロー" というセキュア エンティティ タイプに対する変更権限が付与されているが、特定のデータフローをユーザが変更できないようにしたいとします。アクセス制御を使用して変更したくない特定のデータフローへの変更権限を削除することにより、これが可能になります。アクセス制御設定は、ユーザまたは役割に対して指定できます。ユーザ

に対するアクセス制御設定は、ユーザの役割によって付与された特定のユーザの権限をオーバーライドします。役割に対するアクセス制御設定は、その役割を持つすべてのユーザに適用されません。



## アクセス制御の設定

アクセス制御設定は、役割と連動して、ユーザの権限を定義します。役割は、すべてのデータフローやすべてのデータベースリソースなど、エンティティのカテゴリに対する権限を定義し、アクセス制御設定は、特定のジョブや特定のデータベース接続など、特定のエンティティに対する権限を定義します。

アクセス制御を設定するには、これらのセキュアエンティティタイプに対する表示と変更の権限が必要です。

- セキュリティ - アクセス制御
- セキュリティ - 役割
- セキュリティ - ユーザ

アクセス制御を設定するには、次の手順を実行します。

1. Management Console で、**[システム]** > **[セキュリティ]** に移動します。
2. **[アクセス制御]** タブをクリックします。
3. **[追加]** ボタン  をクリックします。
4. 以下のいずれかの方法を実行します。
  - 役割に対するアクセス制御を指定する場合は、**[役割]** をクリックします。指定するアクセス制御権限は、選択した役割を持つすべてのユーザに影響します。
  - 単一のユーザに対するアクセス制御を指定する場合は、**[ユーザ]** をクリックします。指定するアクセス制御権限は、選択したユーザにのみ影響します。
5. アクセス制御を定義する役割またはユーザを選択します。
6. **[追加]** ボタン  をクリックします。
7. 該当するセキュアエンティティが含まれるセキュアエンティティタイプを選択します。例えば、データフローに対するアクセス制御を設定する場合は、**Platform.Dataflows** を選択します。
8. アクセス制御を設定するセキュアエンティティを選択して **>>** ボタンをクリックし、**[選択中のエンティティ]** リストに追加します。
9. **[追加]** をクリックします。

選択したセキュアエンティティが表示されます。チェックボックスにより、選択した役割またはユーザに適用されている権限が示されます。



10. 各セキュア エンティティに付与する権限を指定します。各セキュア エンティティには、次のいずれかの権限を付与できます。

- 権限は、役割から継承されます。

---

- 権限は、役割から継承され、オーバーライドできません。

---

- この権限は、ユーザまたは役割で指定された権限をオーバーライドする形で付与されます。

---

- この権限は、ユーザまたは役割で指定された権限をオーバーライドする形で拒否されます。

### アクセス制御の例

以下に、役割 `RetentionDepartmentDesigner` のアクセス制御設定を示します。

ホーム / システム: セキュリティ / アクセス制御を追加

## アクセス制御を追加

役割

ユーザ

RetentionDepartmentDesigner


▼ Platform.Dataflow	作成	表示	変更	削除	実行
ExampleJob1	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

この例では、プラットフォーム.データフローセキュア エンティティタイプに表示および変更権限が許可されていますが、削除権限は許可されていません。したがって、デフォルトでは、`RetentionDepartmentDesigner` の役割を持つどのユーザも、すべてのデータフローに対して表示および変更権限を持ちます。ただし、この役割を持つユーザに対し、`ExampleJob1` データフローについてのみは変更を許可しないように制限するとします。この場合、`ExampleJob1` の [変更] 列のチェックボックスをオフにします。これで、この役割を持つユーザは、このデータフローを変更できなくなりますが、他のデータフローは引き続き変更可能です。

### アクセス制御設定の削除

ユーザまたは役割のアクセス制御設定を削除すると、そのアクセス制御設定で定義されていた権限オーバーライドがユーザまたは役割から削除されます。ユーザの場合は、ユーザの役割によっ

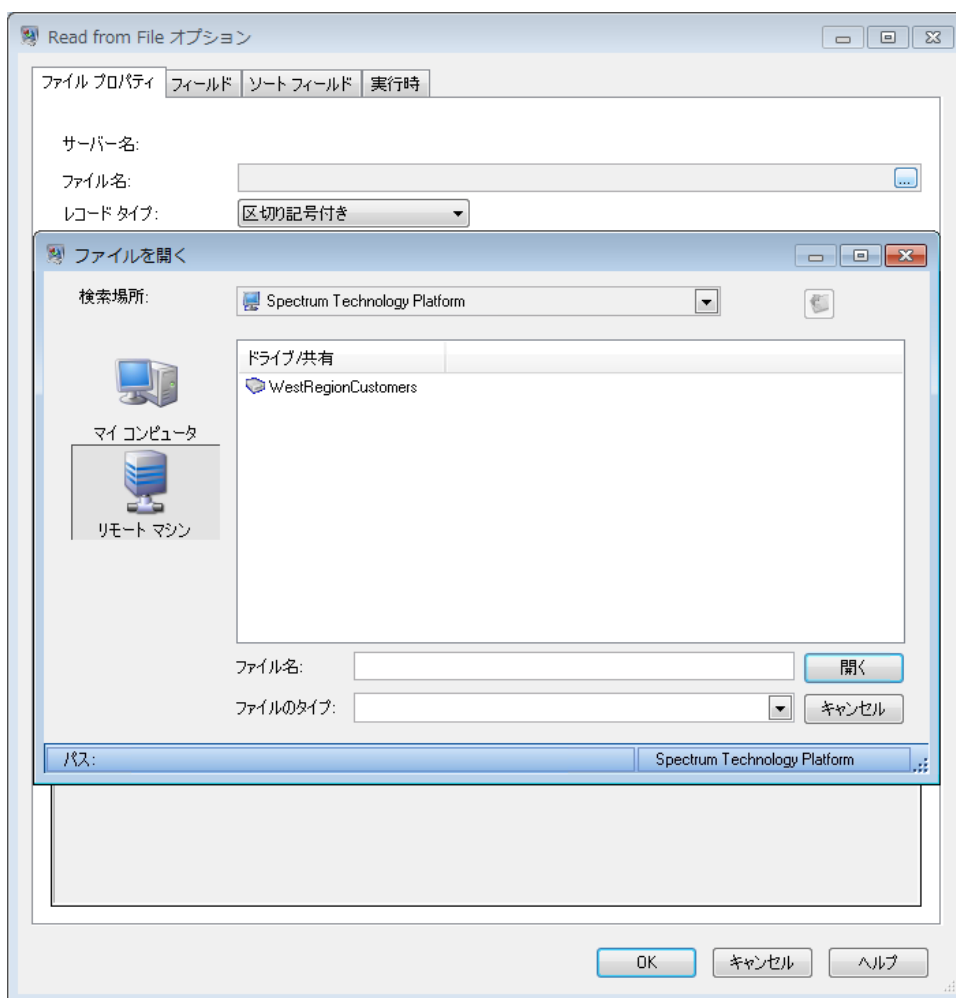
て付与された権限がオーバーライドなしで効力を持つこととなります。役割の場合は、その役割に定義された権限がオーバーライドなしで効力を持つこととなります。

1. **Management Console** を開きます。
2. **[システム] > [セキュリティ]** を選択します。
3. **[アクセス制御]** をクリックします。
4. アクセス制御を削除するユーザまたは役割の横にあるボックスをチェックし、削除ボタン  をクリックします。

## サーバー ディレクトリ アクセスの制限

ファイルの選択が必要なタスクを実行するときに、ユーザは Spectrum™ Technology Platform サーバーのフォルダを参照できます。例えば、ユーザは、Enterprise Designer のソースまたはシンクステージで入力ファイルや出力ファイルを選択する際にサーバーを参照できます。管理者としては、サーバー内の慎重な取り扱いを要する部分を参照したり変更したりできないように、アクセスを制限したいことがあります。


サーバーのファイル システムへのアクセスを防ぐための 1 つの方法は、プラットフォームのセキュリティ権限である **[セキュリティ - ディレクトリ パス]** を決してユーザに与えないことです。これにより、サーバー上のすべてのフォルダへのアクセスを防ぐことができます。サーバー上の一部のフォルダへのアクセスを禁止して、その他のフォルダへのアクセスを許可することもできます。部分的なアクセスを許可する場合、アクセスを許可したフォルダが、ユーザのファイル参照ウィンドウの最上位フォルダとして表示されます。例えば、サーバー上にある **WestRegionCustomers** という名前のフォルダへのアクセスのみを許可している場合、ユーザがそのサーバーを参照すると、次のように、そのフォルダのみが表示されます。



**重要：** 次の2つの場合は、部分的なアクセスしか許可されていなくても、サーバーのファイルシステム全体をユーザが参照できます。

- Management Console で Spectrum データベースを作成するときに、データベース ファイルを参照する場合
  - Management Console でドライバを作成するときに、JDBC ドライバ ファイルを参照する場合
- サーバーのファイルシステム全体をユーザが参照できないようにするには、Spectrum データベースと JDBC ドライバに対するユーザのアクセスを制限する役割を使用します。

サーバー上の一部のフォルダへのアクセスのみを許可するには、以下の手順に従います。

1. Management Console を開きます。
2. [システム] > [セキュリティ] を選択します。
3. [ディレクトリ アクセス] をクリックします。
4. [サーバー ディレクトリへのアクセスを制限] スイッチを [オン] に設定します。
5. [追加] ボタン  をクリックします。

6. **[名前]** フィールドに、アクセスを付与するフォルダの名前を指定します。

ここで指定した名前は、ユーザがサーバーを参照する際に、ディレクトリのルート名として表示されます。このトピックの冒頭で示した例では、アクセス可能なディレクトリに付与された名前は **WestRegionCustomers** です。

7. **[パス]** フィールドに、アクセスを付与するフォルダを指定します。ユーザは、指定されたフォルダ内に含まれるすべてのファイルとサブフォルダにアクセスできます。
8. **[保存]** をクリックします。
9. 他のフォルダへのアクセスを追加で指定する場合は、前の手順を必要に応じて繰り返します。

これで、ユーザは指定されたフォルダにのみアクセスが許可されます。サーバーのディレクトリにアクセスするには、ユーザがプラットフォーム セキュリティ権限 **[セキュリティ - ディレクトリ パス]** を持つ必要があることに注意してください。

注：以前はアクセスできたが、ファイル参照の制限によって現在はアクセスできないファイルがデータフローにある場合、そのデータフローは正しく実行されません。

## HTTPS 通信の設定

デフォルトで Spectrum™ Technology Platform サーバーは、Enterprise Designer との通信、Management Console や Metadata Insights などのブラウザアプリケーションとの通信、Web サービス リクエストや API 呼び出しの処理、リモート サーバーとの通信に HTTP を利用します。これらのネットワーク通信をセキュリティ保護する場合は、HTTPS を使用するように Spectrum™ Technology Platform を設定できます。

注：Spectrum™ Technology Platform では、通信の暗号化に TLS 1.2 を使用します。Spectrum™ Technology Platform の Web サービスや API にアクセスするアプリケーションは、HTTPS を介して接続するために TLS 1.2 をサポートする必要があります。

以下では、HTTPS 通信を Spectrum™ Technology Platform の単一サーバー インストール環境で有効にする手順を説明します。HTTPS を使用する必要があるが、Spectrum™ Technology Platform をクラスタで実行している場合は、この手順を行わないでください。そのようなケースでは、クライアントとの通信に HTTPS を使用するようにロード バランサーを設定します。ロード バランサーと Spectrum™ Technology Platform ノードとの通信、およびノード間の通信は、Spectrum™ Technology Platform クラスタリングが HTTPS をサポートしないことから、暗号化されます。ロード バランサーとクラスタ内の Spectrum™ Technology Platform サーバーは、セキュアな環境を提供するため、ファイアウォールの反対側にある必要があります。

HTTPS 通信を Spectrum™ Technology Platform の単一サーバー インストール環境で設定するには:

1. Spectrum™ Technology Platform サーバーを停止します。
  - Windows 上のサーバーを停止するには、Windows システムトレイの Spectrum™ Technology Platform アイコンを右クリックして、**[Spectrum™ を停止する]** を選択します。または、Windows の [コントロールパネル] の [サービス] を使用して、Pitney Bowes Spectrum™ Technology Platform サービスを停止できます。
  - Unix または Linux 上のサーバーを停止するには、`SpectrumLocation/server/bin/setup` スクリプトをソースとして、`SpectrumLocation/server/bin/server.stop` スクリプトを実行します。

2. 信頼できる認証局 (CA) によって署名された証明書を作成します。

注：証明書は、Spectrum™ Technology Platform で使用されている Java のバージョンに対応する暗号化と長さの要件を満たしている必要があります。Java のバージョンを調べるには、Management Console を開いて、[システム] > [バージョン] に移動します。詳細については、[java.com/en/jre-jdk-cryptoroadmap.html](http://java.com/en/jre-jdk-cryptoroadmap.html) を参照してください。

3. この証明書を JSSE キーストアに読み込みます。詳細については、[www.eclipse.org/jetty/documentation/current/configuring-ssl.html#loading-keys-and-certificates](http://www.eclipse.org/jetty/documentation/current/configuring-ssl.html#loading-keys-and-certificates) を参照してください。
4. `spectrum-override-container-ssl.xml` という XML ファイルを作成し、次の内容を含めます。

```
<beans xmlns="http://www.springframework.org/schema/beans"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:util="http://www.springframework.org/schema/util"
  xsi:schemaLocation="http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans-3.0.xsd

    http://www.springframework.org/schema/util
    http://www.springframework.org/schema/util/spring-util-3.0.xsd">

  <bean id="defaultWebServerConnector"
    class="org.eclipse.jetty.server.ServerConnector">
    <constructor-arg ref="webServer"/>
    <constructor-arg>
      <bean class="org.eclipse.jetty.util.ssl.SslContextFactory">

        <property name="keyStorePath"
          value="/SpectrumKeystore"/>
        <property name="keyManagerPassword" value="password"/>

        <property name="keyStorePassword" value="password"/>
      </bean>
    </constructor-arg>
    <property name="host" value="\${spectrum.bind.address}"/>
    <property name="port" value="\${spectrum.http.port}"/>
  </bean>
</beans>
```

```
<property name="idleTimeout" value="-1"/>
</bean>
</beans>
```

5. 必要に応じて、使用している環境を反映するように以下の行を変更します。

<code>&lt;property name="keyStorePath" value="/SpectrumKeystore"/&gt;</code>	<code>value</code> 属性の値を Java キーストアへのフルパスに変更します。
<code>&lt;property name="keyManagerpassword" value="password"/&gt;</code>	<code>value</code> 属性の値をキーストアに対するパスワードに変更します。
<code>&lt;property name="keyStorePassword" value="password"/&gt;</code>	<code>value</code> 属性の値をキーストア内のキーに対するパスワードに変更します。

6. `spectrum-override-container-ssl.xml` ファイルを `SpectrumLocation/server/app/conf/spring` に保存します。
7. 適当なテキスト エディタでファイル `spectrum-container.properties` を開きます。ファイルの場所は `SpectrumLocation/server/app/conf` です。コメントを解除して、以下のプロパティを設定します。

```
spectrum.http.port=port
spectrum.runtime.port=port
spectrum.runtime.hostname=dnsname
```

`port` はクライアントとの通信に使用するネットワーク ポート (例: 8443) で、`dnsname` は Spectrum™ Technology Platform サーバーのホスト名です。指定するポートは、`spectrum.http.port` と `spectrum.runtime.port` の両方に対して同じでなければなりません。

8. Location Intelligence モジュールと Spectrum Spatial サービスに対して HTTPS 通信を設定する場合は、追加設定を実行してから Spectrum™ Technology Platform サーバーを再起動する必要があります。
- a) `java.properties` ファイル (`SpectrumLocation\server\modules\spatial`) を変更して、すべてのホスト名とポートが Spectrum™ Technology Platform サーバーで使用されているものとまったく同じになるようにします。ホスト名はサーバーの DNS 名と証明書の CN に一致する必要があります。プロパティ `repository.useSecureConnection` を `true` に設定します。例を次に示します。

```
repository.host=www.spectrum.com
repository.port=8443
repository.useSecureConnection=true
```

- b) で、次のサービス設定の URL を、HTTPS を使用するように変更します。

- Mapping (SOAP 経由でマッピング サービスにアクセスし、RenderMap 要求の ReturnImage パラメータが False の場合にのみ必要)
- WFS
- WMS
- WMTS

手順については、『*Spectrum Spatial* ガイド』の「ユーティリティ」セクションにある「Spatial Manager ガイド」を参照してください。

#### 9. Spectrum™ Technology Platform サーバーを開始します。

- Windows 上のサーバーを開始するには、Windows システムトレイの Spectrum™ Technology Platform アイコンを右クリックして、**[Spectrum™ を起動する]** を選択します。または、Windows の [コントロールパネル] の [サービス] を使用して、Pitney Bowes Spectrum™ Technology Platform サービスを開始できます。
- Unix または Linux 上のサーバーを開始するには、`SpectrumLocation/server/bin/server.start` スクリプトを実行します。

#### 10. Location Intelligence モジュールや Spectrum Spatial サービス向けに HTTPS 通信を設定する場合は、変更したファイルを WebDAV を使ってリポジトリにアップロードします (手順については [WebFolders を使用して Spectrum Spatial リポジトリ リソースにアクセスする](#) (9 ページ) を参照)。

## Web サービスの認証

Spectrum™ Technology Platform Web サービスは要求元に対し、有効なユーザ資格情報による認証を求めます。認証方法には、ベーシック認証とトークンベースの認証の 2 種類があります。

### ベーシック認証

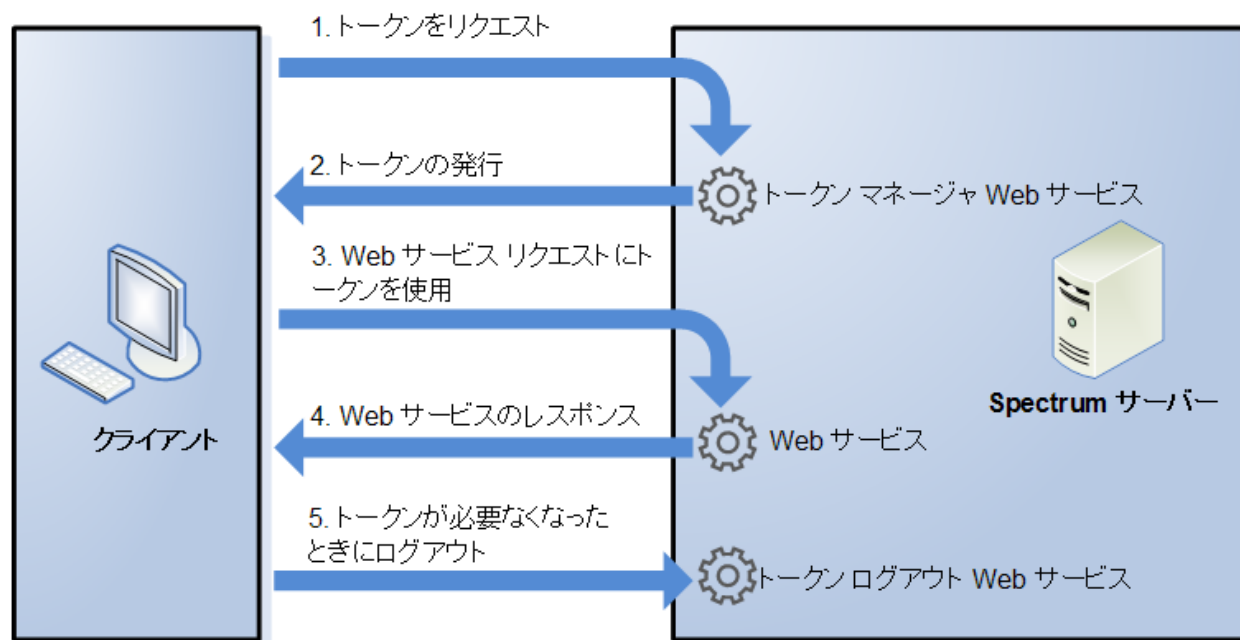
ベーシック認証では、ユーザ ID とパスワードが Web サービスへの各要求の HTTP ヘッダに付加されて Spectrum™ Technology Platform に引き渡されます。ベーシック認証はデフォルトで許可されていますが、管理者によってベーシック認証が無効化されている場合もあります。ベーシック認証が無効化されている場合は、トークンベースの認証を使用して Web サービスにアクセスする必要があります。

### トークンベースの認証

トークンベースの認証では、要求元は Spectrum™ Technology Platform サーバーからトークンを取得し、そのトークンを使用して Web サービスに要求を送信します。要求ごとにユーザの資格情報を送信する代わりに、トークンがサーバーに送信され、サーバーはトークンが有効であるかどうかを確認します。



次の図は、このプロセスを表したものです。



1. トークン管理サービスに要求を送信することにより、Spectrum™ Technology Platformサーバーからトークンを取得します。
2. トークン管理サービスがトークンを発行します。セッショントークンを要求した場合は、セッション ID も発行されます。
3. HTTP ヘッダにトークンを入れて、対象の Web サービスに要求を送信します。セッショントークンの場合は、セッション ID を HTTP ヘッダに含めます。
4. Web サービスが応答を発行します。そのトークンを使用して、同じ Web サービス、または Spectrum™ Technology Platformサーバー上の他の任意の Web サービスに対して、追加の Web サービス要求を送信することができます。1つのトークンで送信可能な Web サービス要求数に上限はありませんが、トークンに有効期限 (Time To Live と呼ばれます) がある場合は、Time To Live の時間が経過するとトークンは無効になります。セッショントークンの場合は、操作がない状態が 30 分間続くと無効になります。
5. トークンが不要になった場合は、トークンログアウト Web サービスに要求を送信することによって、ログアウトする必要があります。それによってトークンは、Spectrum™ Technology Platformサーバー上の有効トークンのリストから削除されます。

### Web サービスのベーシック認証の無効化

Spectrum™ Technology Platform では、Web サービス要求を認証する方法としてベーシック認証とトークン認証の 2 つの方法をサポートしています。デフォルトでは、両方の認証方法が有効です。Web サービス要求に対してベーシック認証ではなくトークン認証を適用したい場合は、以下の手順でベーシック認証を無効にすることができます。

**注：** ベーシック認証を無効にすると、既存クライアントは失敗することに注意してください。Location Intelligence モジュールの場合、WMS、WMTS、およびWFSクライアントでは、ベーシック認証または認証なしが想定されます。トークンベースの認証のみを有効にすると、これらのクライアントは失敗する可能性があります。

1. Spectrum™ Technology Platform サーバーを停止します。
2. 次のファイルをテキスト エディターで開きます。

```
SpectrumLocation/server/app/conf/spectrum-container.properties
```

3. 次のプロパティに **false** を設定します。

```
spectrum.security.authentication.webservice.basicauth.enabled=false
```

4. サーバーを開始します。

### Web サービスの認証の無効化

Spectrum™ Technology Platformで使用するすべてのサービスおよびリソースへのアクセスは、デフォルトで認証がオンになるように設定されています。

すべての SOAP または REST Web サービス (あるいはその両方) のサービス レベル認証を無効にすることができます。これは例えば、Location Intelligence モジュール サービスを使用しているソリューションに独自の高レベル認証を組み込む場合に役立ちます。

Spectrum™ Technology Platform上の Web サービスの認証を無効にするには

1. Spectrum™ Technology Platformサーバーを停止します。
2. この設定ファイルをテキスト エディタで開きます。

```
SpectrumLocation\server\app\conf\spectrum-container.properties
```

3. 各プロパティの値を必要に応じて変更します。例えば、すべての SOAP サービスの認証を無効にする場合は、次のようにします。

```
spectrum.security.authentication.webservice.enabled.REST=true  
spectrum.security.authentication.webservice.enabled.SOAP=false
```

**注：** Location Intelligence モジュールの場合、REST サービスには OGC Web サービスも含まれます。

4. プロパティ ファイルを保存して閉じます。
5. Spectrum™ Technology Platformサーバーを開始します。

以上の操作を完了すると、指定したタイプの Web サービスで認証がオフになります。

## CORS の有効化

Cross-Origin Resource Sharing (CORS) は、ドメイン間のデータ共有を可能にする W3C 標準仕様です。CORS により、1つのドメインで実行している Web アプリケーションが他のドメインからのデータにアクセスできるようになります。Spectrum™ Technology Platform サーバー上で CORS を有効にすることで、他のドメインでホストされている Web アプリケーションに Spectrum™ Technology Platform Web サービスへのアクセスを許可できます。

例えば、**webapp.example.com** でホストされている Web アプリケーションがあるとします。この Web アプリケーションには、**spectrum.example.com** でホストされている Spectrum™ Technology Platform Web サービスを呼び出す JavaScript 関数が含まれています。CORS がない場合は、プロキシサーバーを使用してこの要求を処理する必要があり、それによって実装は複雑になります。CORS があれば、プロキシサーバーは必要ありません。**webapp.example.com** を "allowed origin" (許可された生成元) として指定することで、Spectrum™ Technology Platform がドメイン **webapp.example.com** からの Web サービス要求に応答することを許可します。

Spectrum™ Technology Platform サーバー上で CORS を有効にするには

1. Spectrum™ Technology Platform サーバーを停止します。
2. 次のファイルをテキスト エディターで開きます。

```
SpectrumLocation/server/app/conf/spectrum-advanced.properties
```

3. 次のパラメータを編集します。

### **spectrum.jetty.cors.enabled**

このプロパティに `true` を設定して CORS を有効にします。デフォルトは `false` です。

### **spectrum.jetty.cors.allowedOrigins**

Spectrum™ Technology Platform サーバー上のリソースへのアクセスを許可された生成元のカンマ区切りリスト。デフォルト値は

`http://localhost:8080,http://localhost:443` で、デフォルトの HTTP ポート 8080 とデフォルトの HTTPS ポート 443 を使用したリソースへのアクセスを許可します。

例えば `http://*.domain.com` のように、許可された生成元に 1 つ以上のアスタリスク ("\*") が含まれる場合、アスタリスクは `*` に変換され、ドット文字 (".") は `\.` とエスケープされて、その結果の許可された生成元が正規表現として解釈されます。つまり、許可された生成元として、より複雑な表現が使用できます。例えば `https?://*.domain.[a-z]{3}` は、`http` または `https`、複数のサブドメイン、そして任意の 3 文字のトップレベルドメイン (`.com`、`.net`、`.org` など) にマッチします。

### **spectrum.jetty.cors.allowedMethods**

Spectrum™ Technology Platform サーバー上のリソースにアクセスするときに使用できる HTTP メソッドのカンマ区切りリスト。デフォルト値は POST,GET,OPTIONS,PUT,DELETE,HEAD です。

#### **spectrum.jetty.cors.allowedHeaders**

Spectrum™ Technology Platform サーバー上のリソースにアクセスするときに使用できる HTTP ヘッダのカンマ区切りリスト。デフォルト値は X-PINGOTHER, Origin, X-Requested-With, Content-Type, Accept です。値がアスタリスク ("\*") 1 つである場合、すべてのヘッダが使用可能です。

#### **spectrum.jetty.cors.preflightMaxAge**

クライアントが preflight 要求をキャッシュできる秒数。デフォルト値は 1800 秒、つまり 30 分です。

#### **spectrum.jetty.cors.allowCredentials**

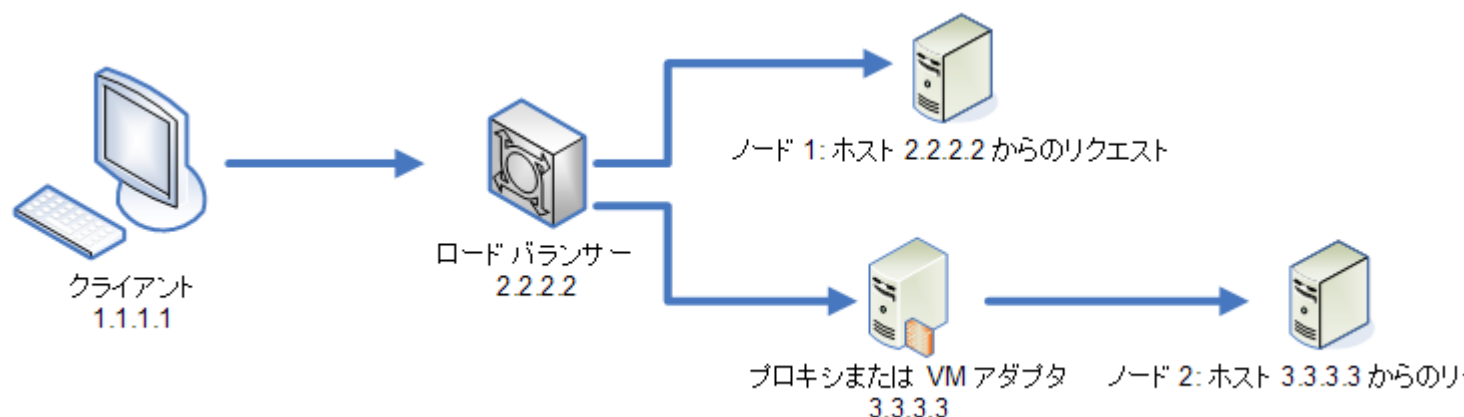
リソースに対し、資格情報付きの要求が可能かどうかを示します。デフォルト値は true です。

4. ファイルを保存して閉じます。
5. Spectrum™ Technology Platform サーバーを開始します。

### トークン認証におけるホスト チェックの無効化

トークン認証において、Spectrum™ Technology Platform サーバーは要求に応答する前に、クライアントによって提示されたトークンをチェックします。サーバーは、トークンが失効していないか、正しく暗号化されているか、また、正しいホストからのものであるかを確認します。セッショントークンの場合は、セッションIDも確認します。そのいずれかのチェックに失敗すると、トークンは拒否され、サーバーはその要求に応答しません。

クラスタ環境では、要求がリダイレクトされることによって、トークンに指定されているホストとは異なるホストが要求元であるように見えてしまい、"無効なトークン" というエラーが発生する可能性があります。例えば、以下に示す 2 つのノードを持つクラスタがあるとします。



クライアントが要求を発行し、それがノード 1 にルーティングされたとします。トークンが作成され、ホスト 2.2.2.2 (ロード バランサー) に結び付けられます。ロード バランサーが要求元であると、ノードが認識するためです。同じクライアントからの次の要求がノード 2 にルーティングされた場合、トークンはホスト 2.2.2.2 に結び付けられたままと見えますが、要求元はプロキシサーバー 3.3.3.3 であるように見えます。この場合、トークンが要求元のホストに関連付けられていないように見えることから、ノードはこのトークンを拒否します。

このような場合には、トークンに含まれるホスト情報を無視するように Spectrum™ Technology Platform サーバーを設定する必要があります。この操作は、ロード バランサーとノードの間に異なるネットワーク デバイスが存在する環境においてのみ、実行してください。すべてのノードが同じネットワーク デバイスの背後にある場合は、ホスト チェックを無効にする必要はありません。

注：以下の手順を実行すると、ホスト チェックは無効になるため、クライアント トークンは実質的にオープン トークンとなります。セッション トークンは、引き続き特定のセッション ID に結び付けられますが、特定のホストには結び付けられません。

1. Spectrum™ Technology Platform サーバー上で以下のプロパティ ファイルを開きます。

```
SpectrumLocation/server/app/conf/spectrum-container.properties
```

2. 次のプロパティを **false** に設定します。

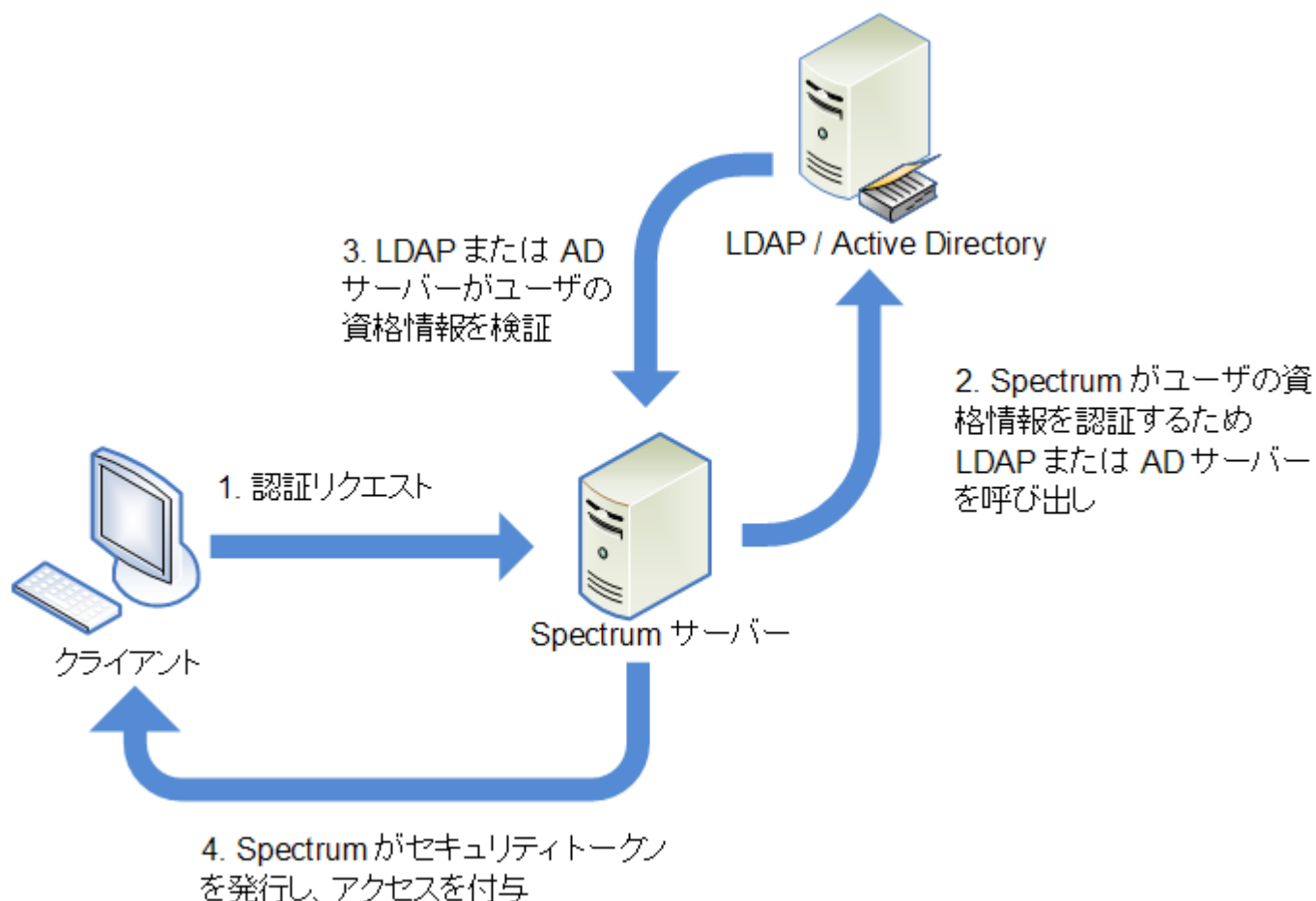
```
spectrum.security.authentication.token.remoteClientCheck.enabled=false
```

3. プロパティ ファイルを保存して閉じます。
4. クラスタ内のすべてのノードに対して、この処理を繰り返します。

## LDAP または Active Directory による認証

Spectrum™ Technology Platform は、LDAP または Active Directory サーバーを認証に使うために設定できます。ユーザが Spectrum™ Technology Platform にログインすると、LDAP または Active Directory を使ってユーザの資格情報が検証されます。その後で、同じ名前の Spectrum™ Technology Platform ユーザが存在するかどうかシステムで確認されます。存在する場合は、ユーザがログインされます。存在しない場合は、そのユーザの Spectrum™ Technology Platform ユーザアカウントが自動的に作成され、役割 user が付与されます。

次の図は、このプロセスを表したものです。



ディレクトリ サービスを使用して認証するように Spectrum™ Technology Platform を設定する前に、お使いのディレクトリ サービスが以下の要件を満たすことを確認してください。

- LDAP の場合、ディレクトリ サーバーは、LDAP バージョン 3 に準拠する必要があります。
- Active Directory サーバーに固有の要件はありません。



注：この処理手順の説明については、Pitney Bowes テクニカル サポートまたはプロフェッショナル サービスにお問い合わせいただくことを推奨します。

1. Management Console で設定した既存のユーザが存在し、LDAP 認証または Active Directory 認証を有効にした後で、それらのユーザを使用する場合は、LDAP システムまたは Active Directory システムにそれらのユーザを作成します。Spectrum™ Technology Platform と同じユーザ名を使うようにしてください。

注：LDAP または Active Directory で "admin" ユーザを作成する必要はありません。このユーザは、LDAP または Active Directory を有効にした後も引き続き、認証に Spectrum™ Technology Platform を使用します。

2. Spectrum™ Technology Platform サーバーを停止します。
3. LDAP または Active Directory の認証を有効にします。
  - a) この設定ファイルをテキスト エディターに開きます。

```
server\app\conf\spectrum-container.properties
```

- b) プロパティ `spectrum.security.authentication.basic.authenticator` を LDAP に設定します。

```
spectrum.security.authentication.basic.authenticator=LDAP
```

設定 LDAP は、LDAP だけでなく Active Directory を有効化する際も使用されます。

- c) ファイルを保存して閉じます。
4. 接続プロパティの設定:
    - a) この設定ファイルをテキスト エディターに開きます。

```
server\app\conf\spring\security\spectrum-config-ldap.properties
```

- b) 次のプロパティを変更します。

#### **spectrum.ldap.url**

LDAP サーバまたは Active Directory サーバの URL (ポートを含む)。例を次に示します。

```
spectrum.ldap.url=ldap://ldapservers.example.com:389/
```

#### **spectrum.ldap.dn.format**

LDAP または Active Directory のユーザアカウントを検索するためのフォーマット。ユーザ名には変数 `%s` を使用します。例を次に示します。



LDAP:

```
spectrum.ldap.dn.format=uid=%s,ou=users,dc=example,dc=com
```

Active Directory:

```
spectrum.ldap.dn.format=%s@example.com
```

### **spectrum.ldap.dn.base**

LDAP または Active Directory のユーザ アカウントを検索するための識別名 (dn)。例を次に示します。

LDAP:

```
spectrum.ldap.dn.base=ou=users,dc=example,dc=com
```

Active Directory:

```
spectrum.ldap.dn.base=cn=Users,dc=example,dc=com
```

### **spectrum.ldap.search.filter**

役割などの属性の検索に使用される検索フィルタ。検索フィルタには以下の変数を含めることができます。

- {user} は Spectrum™ Technology Platform へログインするためのユーザ名
- {dn} は spectrum.ldap.dn.base で指定された識別名

例:

LDAP:

```
spectrum.ldap.search.filter=uid={user}
```

Active Directory:

```
spectrum.ldap.search.filter=userPrincipalName={dn}
```

### **spectrum.ldap.attribute.roles**

これはオプションです。ユーザの Spectrum™ Technology Platform 役割の名前が含まれる LDAP 属性または Active Directory 属性を指定します。LDAP 属性または Active Directory 属性で指定する役割名は、Spectrum™ Technology Platform で定義されている役割の名前と一致する必要があります。

例えば、属性 spectrumroles で定義されている役割を適用するには、次のように指定します。

```
spectrum.ldap.attribute.roles=spectrumroles
```

この属性に `designer` という名前の役割が含まれる場合、`designer` 役割がユーザに付与されます。

指定できる属性は1つだけですが、属性に複数の役割が含まれることがあります。属性内の複数の役割を指定するには、それぞれの属性をカンマで区切ります。属性の各インスタンスに異なる役割が含まれる、複数值の属性を指定することもできます。この1つの属性で指定されている役割だけが Spectrum™ Technology Platform で使用されます。それ以外の LDAP 属性または Active Directory 属性は Spectrum™ Technology Platform の役割に影響を与えません。

Spectrum™ Technology Platform でユーザに役割が割り当てられている場合、ユーザの権限は、LDAP または Active Directory から付与される権限と Spectrum™ Technology Platform からの権限を結合したものになります。

注：初めてログインするユーザが Spectrum™ Technology Platform ユーザ アカウントを持っていない場合は、アカウントが自動的に作成されて、役割 `user` が付与されます。このユーザの有効な権限は、`user` 役割の権限と、`spectrum.ldap.attribute.roles` プロパティに記載された属性で指定されている役割の権限の組み合わせです。

注：Management Console でユーザの役割を表示すると、`spectrum.ldap.attribute.roles` プロパティによってユーザに割り当てられた役割は表示されません。

### **spectrum.ldap.pool.min**

LDAP サーバーまたは Active Directory サーバーに接続するための接続プールの最小サイズ。

### **spectrum.ldap.pool.max**

LDAP サーバーまたは Active Directory サーバーに同時に接続できる接続の最大数。

### **spectrum.ldap.timeout.connect**

LDAP サーバーまたは Active Directory サーバーへの接続が確立されるまで待機する時間 (ミリ秒)。デフォルトでは、1000 ミリ秒に設定されます。

### **spectrum.ldap.timeout.response**

接続が確立された後で、LDAP サーバーまたは Active Directory サーバーからの応答を待機する時間 (ミリ秒)。デフォルトでは、5000 ミリ秒に設定されます。

### **spectrum.ldap.retry.count**

最初の接続試行が失敗した場合に、Spectrum™ Technology Platform サーバーが LDAP サーバーまたは Active Directory サーバーへの接続を試みる回数。接続試行を1回しか許可しない場合は、この値を0に設定します。

ヒント：LDAP サーバーまたは Active Directory サーバーをクラスタ化している場合は、この値を 1 以上に設定し、最初の接続先が利用できなかった場合に LDAP または Active Directory ロード バランサーが接続要求を別のサーバーにリダイレクトできるようにすることをお勧めします。

### spectrum.ldap.retry.wait

接続試行から次の接続試行までの待ち時間 (ミリ秒)。

### spectrum.ldap.retry.backoff

再試行が失敗するたびに待ち時間を増やすために使う乗算計数。

例を次に示します。

```
spectrum.ldap.timeout.connect=1000
...
spectrum.ldap.retry.count=5
spectrum.ldap.retry.wait=500
spectrum.ldap.backoff=2
```

この例では、最初の接続試行時の待ち時間は 1,000 ミリ秒で、5 回の再試行のたびに待ち時間が 2 倍ずつ増えるので、再試行ごとの待ち時間は次のようになります。

- 1 回目の再試行: 500 ミリ秒
- 2 回目の再試行: 1,000 ミリ秒
- 3 回目の再試行: 2,000 ミリ秒
- 4 回目の再試行: 4,000 ミリ秒
- 5 回目の再試行: 8,000 ミリ秒

c) プロパティ ファイルを保存して閉じます。

5. Location Intelligence モジュールを使用しており、役割を ([LDAP 属性値を役割にマッピングする \(50ページ\)](#)) に記載されているように) マッピングする場合は、さらに (*SpectrumFolder*\server\modules\spatial\jackrabbit\workspaces\default\workspace.xml にある) Jackrabbit 設定ファイルを手動で設定し、動的に割り当てられた LDAP または Active Directory の役割を Spectrum Spatial に認識させる必要があります。checkRoles パラメータを次のように追加します。

```
<!--
Spectrum ACL provider.
-->
<WorkspaceSecurity>
  <AccessControlProvider class="com.mapinfo.repository.jackrabbit.acl
    .AccessControlProviderImpl">
    <param name="checkRoles" value="true"/>
  </AccessControlProvider>
</WorkspaceSecurity>
```

## 6. Spectrum™ Technology Platform サーバーを開始します。

Spectrum™ Technology Platform をクラスタ内で実行している場合は、クラスタ内の各サーバー上で `spectrum-container.properties` ファイルと `spectrum-config-ldap.properties` ファイルを変更する必要があります。ファイルを変更する前にサーバーを停止し、ファイルの変更を終えてからサーバーを起動します。LDAP 属性値を役割にマッピングすると、このマッピングはクラスタ内のすべてのノードに複製されるため、このマッピング操作を JMX コンソールで繰り返す必要はありません。

### LDAP 属性値を役割にマッピングする

この手順を実行する前に、LDAP 認証を有効にする必要があります。Location Intelligence モジュールを使用している場合は、Jackrabbit 設定ファイルも変更する必要があります。詳細については、「[LDAP または Active Directory による認証 \(45ページ\)](#)」を参照してください。

LDAP または Active Directory による認証を使用するように Spectrum™ Technology Platform を設定するには、ユーザに付与する役割を決定する値を持つ LDAP 属性を、設定プロパティの 1 つ (ファイル `spectrum-config-ldap.properties` 内の `spectrum.ldap.attribute.roles` プロパティ) で指定する必要があります。デフォルトで、この属性値は Spectrum™ Technology Platform の役割名に正確に一致する必要があります。そうでなければ、役割が付与されません。例えば、`designer` 役割を付与するには、属性に値 `designer` を設定する必要があります。

役割の割り当てに使用する LDAP 属性値が Spectrum™ Technology Platform の役割名に一致しない場合は、LDAP 属性値を役割名にマッピングすることができます。また、Spectrum™ Technology Platform の役割と同じ名前の LDAP 属性値を別の役割にマッピングすることもできます。例えば、組み込まれている役割の 1 つとして `designer` があります。`designer` という名前の LDAP 属性値があり、これを別の役割にマッピングしたい場合は、マッピングを作成することができます。

1. Web ブラウザを開いてに移動します。 `http://server:port/jmx-console`

WHERE:

`server` は、Spectrum™ Technology Platform サーバーの IP アドレスまたはホスト名です。

`port` は、Spectrum™ Technology Platform が使用する HTTP ポートです。デフォルト値は 8080 です。

2. 次のプロパティをクリックします。

```
com.pb.spectrum.platform.common.security.ldap:mappings
```

注：このプロパティは、LDAP 認証を有効にし、サーバが完全に起動した後で表示されます。LDAP 認証を有効にしていない場合は、[LDAP または Active Directory による認証 \(45ページ\)](#) を参照してください。

3. **addMapping** セクションの **ldapValue** フィールドに、Spectrum™ Technology Platform 役割にマッピングする LDAP 属性値を入力します。
4. **roleName** フィールドに、LDAP 属性にマッピングする Spectrum™ Technology Platform 役割を入力します。
5. **[Invoke]** をクリックします。

この LDAP 属性を持つユーザは、次回から Spectrum™ Technology Platform にログインすると、指定された役割が付与されます。

マッピングを削除するには、**ldapValue** フィールド (**removeMapping** セクション内) に、マッピングを解除する LDAP 属性を入力します。

**例**

例えば、`gecos` 属性の値を使用して Spectrum™ Technology Platform の役割を割り当てるケースを想定します。`gecos` に値 `data-quality-user` が設定されている場合に、`designer` 役割を Spectrum™ Technology Platform へのログイン時にユーザに付与したいとします。

これを行うには、役割の割り当てに使用される属性として `gecos` 属性をファイル `spectrum-config-ldap.properties` 内で指定します。

```
spectrum.ldap.attribute.roles=gecos
```

次に、JMX コンソールで、`data-quality-user` 値を `designer` 役割にマッピングします。

### JMX Console

Checking authority: superuser

MBean: com.pb.spectrum.platform.common.security.ldap:mappings=LdapRoleMappings				All MBeans
Description: Handles mapping LDAP role attribute values to Spectrum roles				
Attributes				
Name	Value	Description	Type	
MappingsString		Defined role mappings (ldap -> spectrum)	java.lang.String	
Operations				
Name	Return type	Description		
<b>addMapping</b>	void	Add an LDAP role mapping		
Parameters		<b>Name</b>	<b>Value</b>	<b>Description</b>
		ldapValue	data-quality-user	LDAP attribute value to map from
		roleName	designer	Spectrum role name to map to
		Invoke		
<b>removeMapping</b>	void	Remove an LDAP role mapping		
Parameters		<b>Name</b>	<b>Value</b>	<b>Description</b>
		ldapValue		LDAP attribute value to map from
		Invoke		

これにより、gecos 属性の値が `data-quality-user` に設定されているユーザに役割 `designer` が付与されます。

## LDAP との SSL 通信の有効化

Spectrum™ Technology Platform と LDAP サーバーまたは Active Directory サーバーとの間の通信では、デフォルトで TCP が使用されます。Spectrum™ Technology Platform サーバーと LDAP サーバーまたは Active Directory サーバーとの間の通信をセキュリティで保護する必要がある場合は、LDAP over SSL を使用するように Spectrum™ Technology Platform を設定できます。

1. 以下の場合は、Spectrum™ Technology Platform で使用される Java TrustStore に証明書を追加しなければならない可能性があります。

- デフォルトの Java TrustStore に、使用している認証局のエントリが含まれていない。
- 自己署名証明書を使用している。自己署名証明書の使用は、実稼働環境では推奨されないことに注意してください。

上記のいずれかの条件にあてはまる場合は、以下の手順に従って Java TrustStore に証明書を追加します。

- a) 証明書のコピーを入手します。証明書のコピーは LDAP 管理者から入手できます。また、LDAP Admin などのツールを使用して証明書を表示および保存しても入手できます。
- b) JDK 付属の `keytool` ユーティリティを使用して、証明書を新規または既存の TrustStore に追加します。

例:

```
keytool -import -file X509_certificate_ldap.cer -alias  
server.example.com -keystore ldapTrustStore
```

詳細については、Java のドキュメントを参照してください。

注：証明書は、Spectrum™ Technology Platform で使用されている Java のバージョンに対応する暗号化と長さの要件を満たしている必要があります。Java のバージョンを調べるには、Management Console を開いて、[システム] > [バージョン] に移動します。詳細については、[java.com/en/jre-jdk-cryptoroadmap.html](http://java.com/en/jre-jdk-cryptoroadmap.html) を参照してください。

2. Spectrum™ Technology Platform サーバーを停止します。

- Windows 上のサーバーを停止するには、Windows システムトレイの Spectrum™ Technology Platform アイコンを右クリックして、**[Spectrum™ を停止する]** を選択します。または、Windows の [コントロールパネル] の [サービス] を使用して、Pitney Bowes Spectrum™ Technology Platform サービスを停止できます。



- Unix または Linux 上のサーバーを停止するには、`SpectrumLocation/server/bin/setup` スクリプトをソースとして、`SpectrumLocation/server/bin/server.stop` スクリプトを実行します。

3. 次のファイルをテキスト エディターで開きます。

```
SpectrumLocation\server\app\conf\spring\security\spectrum-config-ldap.properties
```

4. 以下のプロパティを構成します。

#### **spectrum.ldap.url**

LDAP サーバーの URL を指定します。必ず SSL ポート番号を指定してください。番号は通常 636 です。例:

```
spectrum.ldap.url=ldap://server.example.com:636
```

注：URL の末尾にスラッシュ (/) を追加しないでください。

#### **spectrum.ldap.useSSL**

true を指定して LDAP との SSL 通信を有効にします。

#### **spectrum.ldap.trustStore**

LDAP との SSL 通信に使用する証明書が格納されている TrustStore の場所を指定します。Windows の場合:

```
spectrum.ldap.trustStore=file:D:\\Certs\\MyTrustStore
```

Linux および Unix の場合:

```
spectrum.ldap.trustStore=file://Certs//MyTrustStore
```

#### **spectrum.ldap.trustStore.password**

TrustStore パスワードを指定します。

**重要：** Spectrum™ Technology Platform をクラスタ内で実行している場合は、クラスタ内の各サーバーでこの手順を繰り返します。

### LDAP との SSL 通信の無効化

LDAP や Active Directory との通信に SSL を使用するように Spectrum™ Technology Platform が設定されており、TCP を使用するように設定を戻す必要がある場合は、以下の手順に従います。

1. Spectrum™ Technology Platform サーバーを停止します。



- Windows 上のサーバーを停止するには、Windows システムトレイの Spectrum™ Technology Platform アイコンを右クリックして、**[Spectrum™ を停止する]** を選択します。または、Windows の [コントロールパネル] の [サービス] を使用して、Pitney Bowes Spectrum™ Technology Platform サービスを停止できます。
  - Unix または Linux 上のサーバーを停止するには、`SpectrumLocation/server/bin/setup` スクリプトをソースとして、`SpectrumLocation/server/bin/server.stop` スクリプトを実行します。
2. 次のファイルをテキスト エディターで開きます。

`SpectrumLocation\server\app\conf\spring\security\spectrum-config-ldap.properties`

3. 以下のプロパティを構成します。

#### **spectrum.ldap.url**

SSL ポートの代わりに TCP ポートを使用するように LDAP サーバーの URL を変更します。デフォルト値は 389 です。例:

```
spectrum.ldap.url=ldap://ldapservers.example.com:389/
```

注：URL の末尾にスラッシュ (/) を追加する必要があります。

#### **spectrum.ldap.useSSL**

`false` を指定して LDAP との SSL 通信を無効にします。

#### **spectrum.ldap.trustStore**

このプロパティをコメントアウトします。

#### **spectrum.ldap.trustStore.password**

このプロパティをコメントアウトします。

## Location Intelligence モジュールのセキュリティ

Location Intelligence モジュールは、Spectrum™ Technology Platform に対して使用される役割ベースのセキュリティを使用します。セキュリティはプラットフォーム レベルで処理されるので、Location Intelligence モジュールのすべてのセキュリティ アクティビティの管理には Management Console を使用できます。これには、ユーザアカウントの管理 (ユーザアカウントの作成、変更、削除) に加えて、名前付きリソースの権限の設定が含まれます。

## 定義済みの *spatial* の役割

Location Intelligence モジュールをインストールすると、Management Console で 3 つの定義済み役割が使用可能になります。

- spatial-admin** spatial-admin 役割は、すべての名前付きリソースと、名前付きテーブルに関連付けられたすべてのデータセットに対する全権限 (作成/表示/変更/削除) を提供できます。これらの権限は、Location Intelligence モジュールのセキュア エンティティ タイプ Location Intelligence.Named Resources および Location Intelligence.Dataset.DML を使用して制御します。Location Intelligence モジュール サービスのユーザには、最低限、使用するリソースとその従属リソースに対する表示権限が必要です。データセットの権限の制御については、[データセットに対するアクセス制御](#) (56ページ) を参照してください。
- spatial-user** spatial-user 役割は、名前付きリソースの表示権限のみを提供します。これらの権限は、Location Intelligence モジュールのセキュア エンティティ タイプ Location Intelligence.Named Resources を使用して制御されます。Location Intelligence モジュール サービスのユーザには、最低限、使用するリソースとその従属リソースに対する表示権限が必要です。
- spatial-dataset-editor** spatial-dataset-editor 役割は、データセットに対するすべての権限 (作成/表示/変更/削除) を提供します。これらの権限は、Location Intelligence モジュールのセキュア エンティティ タイプ Location Intelligence.Dataset.DML を使用して制御します。この役割と、データセットでの権限の制御については、[データセットに対するアクセス制御](#) (56ページ) を参照してください。

名前付きリソースにアクセスする必要のあるデータフロー設計者は、"designer" 役割では付与されない権限を必要とします。空間データフロー設計者の作成方法については、[空間データフロー設計者の作成](#) (59ページ) を参照してください。

## カスタムの *spatial* の役割とアクセス制御の設定

定義済みの *spatial* の役割を基にしてカスタム役割を作成し、それをユーザ アカウントに割り当てた後、個別の名前付きリソース、データセット、フォルダまたはディレクトリにアクセス制御設定 (オーバーライド) を適用することによって、これらの役割およびユーザの名前付きリソースに対するアクセスを微調整できます。Location Intelligence モジュールにおけるセキュリティの設定については、権限なしの役割を作成して、その役割にアクセス制御設定を適用し (例えば、特定のフォルダ内の名前付きリソースに対する変更および削除権限を許可するなど)、そのカスタム役割や、いずれかの定義済みの *spatial* の役割をユーザに割り当てるとというのが、典型的なシナリオであり、ベスト プラクティスです。よくある別のシナリオとしては、1 人のユーザに対してオーバーライド権限を設定するというものがあります。例えば、名前付きリソースに対する表示権限のみを持つユーザ アカウントを作成してから、特定のフォルダ内の名前付きリソースの変更と削除を許可するアクセス制御設定を、そのユーザに適用します。

## フォルダ

フォルダ権限は、下位のリソースおよびフォルダにそれをオーバーライドする特定のアクセス制御設定がない限り、それらのリソースおよびフォルダに継承されます。これは、一連のリソースに権限を設定したい場合に便利です。指定したユーザまたは役割のみがアクセスできるフォルダを作成できます。このフォルダおよびその下位にあるすべての要素は他のユーザには表示されません。Location Intelligence.Named Resources エンティティ タイプでは、スラッシュ (/) で終わるすべての表示リソースがリポジトリ内のフォルダまたはディレクトリです。

ただし、このフォルダレベルの権限は、下位の個別リソースレベルで設定された権限をオーバーライドしません。例えば、あるフォルダに対する作成権限を持つ特定の役割またはユーザに対し、そのフォルダ内の1つのリソース(名前付きテーブルなど)の表示権限のアクセス制御設定が適用されている場合、その1つのリソースに対する表示(読み取り専用)権限がフォルダに対する作成権限よりも優先されます。

## データセットに対するアクセス制御

### データセットとは

データセットは、テーブル形式のデータ値のコレクションで、一般に行(またはレコード)と列で構成されています。Location Intelligence モジュールでは、データセットは、TAB ファイルのフォーマット、シェイプファイル、GeoPackage ファイル、または MS SQL Server のテーブルなどの JDBC ベースのテーブルを取得できます。

### データセットのアクセス制御の利点

データセットのアクセス制御により、管理者は名前付きテーブルがポイントするデータセットの編集権限から名前付きテーブルのアクセス権の関連付けを切断できます。例えば管理者は、名前付きテーブルへの読み取り専用の権限(表示権限)を変更することなく、データセットへ完全な編集権限(作成、変更、削除の権限)を与えることができます。ユーザが、データ操作言語(DML)の操作(Feature Service と Write Spatial Data ステージのいずれかによる挿入、更新、削除の操作)を実行しようとする、指定された名前付きテーブルに対するそのユーザの権限が、Location Intelligence.Named Resources エンティティ タイプだけでなく、Location Intelligence.Dataset.DML エンティティ タイプに対しても検証されます。表示権限がないと判定された場合、その名前付きテーブルはユーザのリポジトリに表示されません。

### データセット セキュア エンティティとは

LocationIntelligence.Dataset.DML セキュア エンティティは、Location Intelligence モジュールの2種類のセキュア エンティティのうちの一つです。このセキュア エンティティによって、名前付きテーブルに関連付けられたデータセットへのDML権限を制御できます。名前付きテーブルが作成またはアップロードされると(Spatial Manager、管理ユーティリティ、Named Resource サービス、WebDAV などの任意のツールを使用)、新しい LocationIntelligence.Dataset.DML セキュア

エンティティがその名前付きテーブルに関連付けられたデータセットに対して自動的に作成されます。ユーザが、書き込み可能 (JDBC ベース) テーブルで DML 操作を行うには、名前付きテーブルに対する表示権限、およびデータセットに対する作成/変更/削除権限が必要です。DML 操作には挿入、更新、および削除の操作があり、**Write Spatial Data** ステージまたは **Feature Service** を使用して実行します。

**注：** .TAB ファイルやシェイプファイルなど、書き込みできないデータセットのデータセットセキュア エンティティにも作成/変更/削除権限を設定できますが、DML 操作をそれらのデータセットに実行することはできません。

**ヒント：** データセットのセキュア エンティティに対する表示権限は、名前付きリソースに対する権限に影響しません。データセット セキュア エンティティに対する表示権限をオフにした場合も、テーブル内のデータを表示できます。ユーザにテーブルを表示したくない場合は、名前付きリソースのセキュア エンティティに対する表示権限を削除します。

名前付きテーブルの名前変更、移動、削除が行われると、**Spectrum Spatial** はそのデータセットに関連するセキュア エンティティの名前変更または削除を行います。

### spatial 役割とデータセット アクセス

役割はシステムの別の部分へのアクセスを許可または拒否するために使用します。役割によって、権限の管理が容易になります。Management Console では、Location Intelligence モジュールのユーザに次の 3 つの定義済み役割を使用できます。

**spatial-admin** spatial-admin 役割は、すべての名前付きリソースとデータセットに対して、すべての権限 (作成/表示/変更/削除) を提供できます。spatial-admin の役割を持つユーザは、名前付きリソースの表示と、データセットの編集ができます。

**注：** ファイルシステム ベースの名前付き接続のソース フォルダや、サービス設定ファイルの特定の設定 (Mapping サービスのイメージディレクトリなど) を作成または編集するには、追加のファイルサーバー アクセスが必要です。詳細については、「[名前付きリソース管理者の作成 \(58ページ\)](#)」を参照してください。

**spatial-user** spatial-user 役割は、名前付きリソースの表示権限のみを提供します。spatial-user の役割を持つユーザは、リソースを表示できますが、データセットを編集できません。


**spatial-dataset-editor** spatial-dataset-editor 役割は、データセットに対するすべての権限 (作成/表示/変更/削除) を提供します。例えば、管理者は現在 spatial-user 役割を持つユーザに spatial-dataset-editor の役割を追加することにより、すべての権限を容易に与えることができます。

これらの定義済みの役割は変更できません。ただし、定義済みの spatial の役割を基にしてカスタム役割を作成し、それをユーザアカウントに割り当てた後に、データセット、個別の名前付きリ



ソース、または名前付きリソースを含むフォルダにアクセス制御設定 (オーバーライド) を適用することによって、これらの役割およびユーザに対するアクセスを微調整できます。詳細については、[アクセス制御の設定](#) (32ページ) を参照してください。

## 名前付きリソース管理者の作成

Spatial Manager および Management Console を使用してリポジトリ内の名前付きリソースを管理するには、定義済みの `spatial` の役割によって提供されるアクセスに加えて、これらのリソースへのフル アクセスを許可する役割がユーザに割り当てられている必要があります。定義済みの `spatial` の役割は変更できず、定義済みの "名前付きリソース管理者" の役割は Spectrum™ Technology Platform では提供されません。ただし、定義済みの `spatial` の役割を基にしてこのような役割を作成できます。

1. Management Console を開きます。
2. [システム] > [セキュリティ] を選択します。
3. [役割] をクリックします。
4. `spatial-admin` の横のチェックボックスをオンにすることによって開始点として使用する役割を選択し、コピー ボタン  をクリックします。`spatial-admin` 役割には、Location Intelligence Module.Named Resources と Location Intelligence Module.Dataset のセキュア エンティティタイプに対する表示、変更、作成、削除の各権限があります。
5. [役割名] フィールドに、この役割の名前を入力します (例: "resource-admin")。
6. 以下のセキュア エンティティ タイプに対して次のように追加の権限を設定します。  
データベース リソース:
  - **Centrus** データベース リソース: 表示/変更/生成/削除/実行 (必要な場合)
  - **Enterprise Routing**: 表示/変更/生成/削除/実行 (必要な場合)プラットフォーム:
  - サービス: 表示/変更/実行
  - システム - バージョン情報: 表示リソース接続:
  - リソース - ファイル サーバー接続: 表示
  - リソース - JDBC ドライバ: 表示
7. [保存] をクリックして、新しい `resource-admin` 役割を保存します。
8. [ユーザ] をクリックします。



9. 既存のユーザを選択して編集ボタン  をクリックしてそれを変更するか、追加ボタン  をクリックして新しいユーザを作成します。
10. 新しい "resource-admin" 役割をユーザ アカウントに割り当てて、ユーザが名前付きリソースを管理できるようにします。

以上でユーザに、Spatial Manager および Management Console で名前付きリソースを管理するために必要なアクセス権が付与されます。

## 空間データフロー設計者の作成

Location Intelligence モジュールのステージとサービスに対するデータフローを作成するには、ユーザに **designer** と **spatial-user** の両方の役割が割り当てられている必要があります。**spatial-user** の役割は、Location Intelligence.Named Resources セキュア エンティティ タイプの下にある名前付きリソースへの View アクセスを提供します。**designer** の役割は、Platform セキュア エンティティ タイプ (Dataflow など) への必要なアクセスを提供します。

1. Management Console で、[システム] > [セキュリティ] に移動します。
2. 既存のユーザを選択して [編集] ボタン  をクリックするか、[追加] ボタン  をクリックして新しいユーザを作成します。
3. [役割] セクションで、**designer** と **spatial-user** の両方の役割をユーザ アカウントに割り当てます。

これで、ユーザは名前付きリソースを表示したり、Location Intelligence モジュールのステージとサービスのリソースを使用してデータフローを設計したりできる権限を持つことになります。

## リポジトリに対する WebDAV アクセスの制限

WebDAV は、Spectrum Spatial リポジトリ内のリソースにアクセスするためのプロトコルとして使用されます。デフォルトでは、WebDAV を使用したリポジトリへのアクセスは特定のサーバーに制限されるものではなく、リポジトリにアクセス可能なすべてのサーバーに公開されています。空間 java プロパティファイルを変更することによって、アクセスを特定のサーバーに制限することができます。これを行うには、WebDAV を公開するホスト名 (IP) の (カンマ区切りの) リストを含む、次のプロパティを追加します。変更後に、Spectrum™ Technology Platform サーバーを再起動する必要があります。

WebDAV を使用したリポジトリ アクセスを制限するには、次の手順に従います。

1. modules/spatial/java.properties ファイルをエディタで開きます。

2. 次のプロパティをこのファイルに追加します。

```
repository.accesscontrol.allows=
```

3. WebDAV アクセスを許可する IP アドレスのリストを含めます。IP アドレスのカンマ区切りリストを使用して、複数のサーバーを追加できます。このプロパティを空のままにすると、Spectrum™ Technology Platform がインストールされているコンピュータを除くすべてのサーバーに対して、WebDAV を使用したすべてのアクセスが無効になります。

```
repository.accesscontrol.allows=192.168.2.1,192.168.2.2
```

4. サーバを再起動します。

以上の操作を完了すると、リポジトリに対する WebDAV アクセスが制限されます。

## WebDAV を HTTPS で使用する場合

HTTPS を介してサーバーと通信し、ドライブをリポジトリにマッピングする場合、WebDAV クライアントでは TLS v1.2 プロトコルの使用が必要です。クライアント コンピュータが Windows 7 SP1、Windows Server 2008 R2 SP1、および Windows Server 2012 上で稼働している場合、このプロトコルを利用するために、セキュリティパッチとレジストリの更新を適用する必要があります。

1. クライアント コンピュータ上で、使用しているオペレーティングシステムに該当するパッチをマイクロソフト サポート技術情報 (<https://support.microsoft.com/ja-jp/kb/3140245>) から適用します。
2. サポート技術情報の記事に記載されている手順に従って、レジストリを更新し、TLS v1.2 のサポートを追加します。DefaultSecureProtocols の値を少なくとも 0x00000800 にする必要があります。
3. レジストリのエントリを変更した後、クライアント コンピュータを再起動します。



# 4 - システムの監視

## このセクションの構成


---

システム イベントの表示	62
<b>Spatial</b> のログ	63
メール サーバーの設定	65
有効期限通知を受信する項目の選択	66
バージョン情報の表示	67
ライセンス情報の表示とエクスポート	67
JMX コンソールによるパフォーマンスのモニタリング	68
JMX コンソールによる、ファイル ハンドル キャッシュ統計の監視	69
メモリ使用量の監視	69

## システム イベントの表示

システム ログには、Spectrum™ Technology Platform サーバーのラッパー ログからのメッセージが表示されます。これらのメッセージには、サーバー操作や、API および Web サービスからサービスに対して作成されたリクエストに関する情報が含まれます。トラブルが発生し、考えられる原因についての情報を探する場合、システム ログを表示します。

Spectrum™ Technology Platform をクラスタで実行している場合、使用できるシステム ログはたまたま接続先となったノードで生成されたものです。特定のノードのシステム ログを表示するには、そのノードでファイル `ServerLocation\server\app\repository\logs\wrapper.log` をテキスト エディタに開きます。

1. Management Console を開きます。
2. [システム] > [ログ] を選択します。
3. ダウンロード アイコンをクリックして  システム ログ ファイルをダウンロードします。
4. ダウンロードしたファイルをテキスト エディタで開きます。

## サービスのログ レベルの設定

デフォルトのログレベルや、システム上の各サービスのログレベルを指定することができます。ログレベルを変更しても、変更前に作成されたログ エントリにはその変更は反映されません。

注：サービスに対して指定するログレベルは、監査ログには影響しません。Management Console で表示できるイベント ログのログレベルのみを制御します。現時点では、Web 版の Management Console ではイベント ログを表示できません。

1. Management Console を開きます。
2. [システム] > [ログ] を選択します。
3. [システム デフォルト ログレベル] フィールドで、システムのサービスに対するデフォルトのイベント ログレベルを選択します。

**無効**           すべてのイベント ログを無効にします。

**致命的**        最小限のログです。致命的なエラーのみがログに記録されます。致命的なエラーとは、システムを使用不可能にするエラーのことです。

エラー	エラーと致命的なエラーがログに記録されます。エラーとは、システムの一部が使用不可能になる単発的な問題を指します。例えば、1つのサービスが機能しなくなる問題によってエラーが生成されます。
警告	イベント警告、エラー、致命的なエラーがログに記録されます。警告とは、システムの動作を停止させることのない問題を指します。例えば、パラメータに無効な値が設定されているサービスをロードすると、警告が発行され、デフォルトのパラメータが使用されます。サービスの使用時に、結果は返ってきたが問題が存在するという場合に、警告がログに記録されます。
情報	抽象度の高いシステム情報がログに記録されます。これは、実稼働に適した最も詳細なログレベルです。通常、情報イベントは、起動や初期化の際に発生し、バージョン情報やロードされたサービスなどの情報を提供します。
デバッグ	システムの問題のデバッグ時に適した、非常に詳細なログレベルです。
トレース	プログラムの実行(メソッドの開始と終了)をトレースする、最も詳細なログレベルです。デバッグに使用する詳細なプログラムフロー情報を提供します。

各ログレベルは、1つ上のログレベルの内容を含みます。つまり、警告ログレベルが選択されている場合、エラーと致命的なエラーも記録されます。情報ログレベルが選択されている場合は、情報メッセージ、警告、エラー、および致命的なエラーが記録されます。

注：最も詳細なログレベルを選択すると、システムのパフォーマンスに影響が生じる恐れがあります。したがって、必要なログ要件を満たす最小レベルの設定を選択する必要があります。

- サービスごとに異なるログレベルを指定することもできます。

## Spatial のログ

logback.xml ファイルを使うと、ログの動作をより細かく制御できます。例えば、デフォルトでは出力はコンソールに送られ、wrapper.log にリダイレクトされますが、別のログファイルに出力を送ることができます。また、ログレベルも設定できるため、たとえばログを完全にオフにしたり、致命的なエラーのみログに記録したりできます。

デフォルトの logback ファイルは次のとおりです。

(<Installed>\Pitney Bowes\Spectrum\server\modules\spatial\logback.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<!--
=====
-->
<!-- Logger configuration for remote components
-->
<!--
-->
<!-- log to console, redirected to Platform log
(server\app\repository\logs\wrapper.log) -->
<!-- log to files, redirected to (server\modules\spatial\spatial.XXX.log)
-->
<!--
-->
<!-- for general information about the configuration file, check out
the logback manual -->
<!-- at http://logback.qos.ch/manual/configuration.html
-->
<!--
=====
-->
<configuration>
  <appender name="CONSOLE-SPATIAL"
class="ch.qos.logback.core.ConsoleAppender">
  <encoder>
    <pattern>[Spatial] - [%thread] %-5level %logger{35} - %msg%n</pattern>

  </encoder>
</appender>
  <!--appender name="FILE-SPATIAL"
class="ch.qos.logback.core.rolling.RollingFileAppender">
  <file>${gl.server.modules.dir}/spatial/${component.name}.log</file>
  <encoder>
    <pattern>%d [%thread] %-5level %logger{35} - %msg%n</pattern>
  </encoder>
  <append>true</append>
  <triggeringPolicy
class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">
    <maxFileSize>10MB</maxFileSize>
  </triggeringPolicy>
  <rollingPolicy
class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
    <fileNamePattern>${component.name}.log.%i</fileNamePattern>
    <maxIndex>1</maxIndex>
  </rollingPolicy>
</appender-->
  <!-- Level: OFF, ERROR, WARN, INFO, DEBUG -->
  <logger name="com.mapinfo.midev" level="INFO" additivity="false">
    <appender-ref ref="CONSOLE-SPATIAL"/>
    <!-- appender-ref ref="FILE-SPATIAL"/ -->
  </logger>
</configuration>

```

オプション	値
レベル	<ul style="list-style-type: none"> <li>• OFF – ログをオフにします。</li> <li>• ERROR – 実行時エラーや予期せぬエラーをログに記録します。</li> <li>• WARN – 非推奨の API を使用したなどの警告のみをログに記録します。</li> <li>• INFO – 起動やシャットダウンなどの実行時のイベントをログに記録します。[デフォルト]</li> <li>• DEBUG – 詳細なデバッグ情報をログに記録します。</li> </ul>
出力	<ul style="list-style-type: none"> <li>• CONSOLE-SPATIAL – ログ情報をコンソールに送ります。[デフォルト]</li> <li>• FILE-SPATIAL – ログ情報をコンポーネント別にログファイルに記録します (現在の Spectrum Spatial には 1 つのリモートコンポーネントしかないため、このオプションは適用されません)。</li> </ul>

## メールサーバーの設定

Spectrum™ Technology Platform では、電子メールアラートを送信して重要なイベントを通知できます。電子メール通知は、データフローおよびプロセスフロー内の状況に従って送信したり、ライセンスやデータベースなどのアイテムが有効期限切れになりそうなタイミングで送信したりできます。

Spectrum™ Technology Platform にはメールサーバーが組み込まれていないので、電子メール通知を有効にするためには、外部の SMTP サーバーを使用できるように設定を行う必要があります。

1. Management Console を開きます。
2. [システム] > [メールサーバー] を選択します。
3. [ホスト] フィールドに、電子メール通知の送信に使用する SMTP サーバーのホスト名または IP アドレスを入力します。

4. **[ポート]** フィールドに、Spectrum™ Technology Platform サーバーと SMTP サーバーとの間のネットワーク通信で使用するポートの番号または範囲を入力します。  
デフォルトのポート番号は 25 です。
5. **[ユーザ名]** および **[パスワード]** フィールドに、Spectrum™ Technology Platform サーバーが SMTP サーバーでの認証に使用する資格情報を入力します。
6. **[送信者アドレス]** フィールドに、通知の送信元電子メール アドレスを入力します。
7. メール サーバーが正しく設定されていることを確認するためには、テスト用の電子メールを送信できます。テスト メールを送信先電子メール アドレスを **[テスト アドレス]** フィールドに入力し、**[テスト]** をクリックします。
8. **[保存]** をクリックします。

これで、Spectrum™ Technology Platform サーバーが SMTP サーバーに接続され、SMTP サーバーを使用して通知メールを送信できます。

#### 例: メール サーバーの設定

mail.example.com という SMTP サーバーがあるとします。このメール サーバーを使用して、Spectrum™ Technology Platform サーバーから送信される電子メール通知を処理したいとします。この SMTP サーバー上には Spectrum123 というアカウントを Example123 というパスワードで作成してあります。このアカウントの電子メールは spectrum.notification@example.com です。

以上の情報に従って通知を設定するには、各フィールドに次のように入力します。

ホスト	mail.example.com
送信者アドレス	spectrum.notification@example.com
ユーザ名	Spectrum123
パスワード	Example123

## 有効期限通知を受信する項目の選択

Spectrum™ Technology Platform は、ライセンス、データベース、ソフトウェア コンポーネントの有効期限が切れる前に電子メール通知を送付できます。これによって、期限切れによってビジネス プロセスに支障をきたすことなく、必要な措置を講じることができます。有効期限があるコンポーネントには、以下のようなものがあります。

- ライセンス

注：電子メール通知は、トランザクションベースのライセンスに対しては提供されていません。ライセンスのトランザクション数が上限に近づいている場合は、Management Console のシステム ログにメッセージが表示されます。

- データベース (CASS 処理に使用される米国郵便データベースなど)
- 一部のソフトウェア コンポーネント (Universal Addressing モジュールにおいて米国住所の検証に用いられるエンジンなど)

ヒント：有効期限がある項目を参照するには、Management Console を開き、[システム]>[ライセンスと有効期限] を選択します。

通知を希望する項目が選択できるため、気になる項目についてのみ通知を受け取ることができます。

1. Management Console を開きます。
2. [システム]>[ライセンスと有効期限] を選択します。
3. ある項目に対する有効期限通知メールを受信するには、[通知を送る] 列のチェックボックスをオンにします。デフォルトよりも早く、または遅く通知を受け取りたい場合は、有効期限の何日前に通知を希望するかを、有効期限までの日数で指定します。

## バージョン情報の表示

1. Web ブラウザで次の URL を表示します。

`http://server.port/managementconsole`

ここで `server` は、Spectrum™ Technology Platform サーバーの名前または IP アドレスで、`port` は Spectrum™ Technology Platform が使用する HTTP ポートです。デフォルトの HTTP ポートは 8080 です。

2. [システム]>[バージョン]をクリックします。

## ライセンス情報の表示とエクスポート

ライセンスに関する情報を XML ファイルにエクスポートできます。この機能は、テクニカル サポートとともにライセンスの問題を解決する際に必要となる場合があります。

1. Web ブラウザで次の URL を表示します。



`http://server:port/managementconsole`

ここで `server` は、Spectrum™ Technology Platform サーバーの名前または IP アドレスで、`port` は Spectrum™ Technology Platform が使用する HTTP ポートです。デフォルトの HTTP ポートは 8080 です。

2. [システム] > [ライセンスと有効期限] をクリックします。
3. エクスポート アイコンをクリックします。

ライセンス情報は、拡張子が `.lic` である XML ファイルに保存されます。

## JMX コンソールによるパフォーマンスのモニタリング

JMX コンソールはブラウザベースのツールで、データフロー内の各ステージのパフォーマンス統計を記録するパフォーマンス モニタリング ツールを提供します。

1. Web ブラウザを開いてに移動します。 `http://server:port/jmx-console`

WHERE:

`server` は、Spectrum™ Technology Platform サーバーの IP アドレスまたはホスト名です。

`port` は、Spectrum™ Technology Platform が使用する HTTP ポートです。デフォルト値は 8080 です。

2. 管理者アカウントでログインします。
3. "Domain: com.pb.spectrum.platform.performance" の下にある **com.pb.spectrum.platform.performance:service=PerformanceMonitorManager** をクリックします。
4. **[enable]** の横にある **[Invoke]** ボタンをクリックします。
5. **[Return to MBean View]** をクリックして、**[PerformanceMonitorManager]** 画面に戻ります。

これでパフォーマンス モニタリングが有効になります。データフローを実行すると、パフォーマンス統計が **[PerformanceMonitorManager]** 画面の上部に表示されます。次のことに注意してください。

- 更新を確認するには、画面を再表示する必要があります。
- カウンタをリセットするには、**[reset]** の横にある **[Invoke]** ボタンをクリックします。
- Spectrum™ Technology Platform サーバーを停止すると、パフォーマンス モニタリングはオフになります。サーバーを再起動したら、パフォーマンス モニタリングを再度オンにする必要があります。

## JMX コンソールによる、ファイルハンドル キャッシュ統計の監視

JMX コンソールは、ネイティブ TAB およびシェイプファイルのファイルハンドル キャッシュ統計を含む、パフォーマンスやレコード統計を監視するブラウザベースのツールです。

1. Web ブラウザを開いて `http://server:port/jmx-console` に移動します。

説明：

`server` は、Spectrum™ Technology Platform サーバーの IP アドレスまたはホスト名です。

`port` は、Spectrum™ Technology Platform が使用する HTTP ポートです。デフォルト値は 8080 です。

2. 管理者アカウントでログインします。
3. [Domain: Spatial] の下で、**[Spatial:name=TABFileHandlePool,type=Remote Component]** または **[Spatial:name=ShapeFileHandlePool,type=Remote Component]** をクリックして、TAB ファイルまたはシェイプファイルのファイルハンドル キャッシュ統計を表示します。

注：このページ上で、サーバーを再起動することなく、ファイルハンドル キャッシュを無効にするか、クリアすることもできます。

4. **[すべての MBean]** をクリックして、メインの JMX コンソールに戻ります。

## メモリ使用量の監視

JMX コンソールでは、空間リモート コンポーネントの JVM ヒープ使用量を監視できます。

# Spectrum Technology Platform JMX Console

**MBean: Spatial:name=Process,type=Remote Feature Component**

**Description: The Managed Bean of Remote Feature Component containing FeatureService  
UserManagementService for process monitoring**

## Attributes

Name	Value
HeapMemoryUsage	javax.management.openmbean.CompositeDataSupport(compositeType=javax.management.openmbean.CompositeType(name=((itemType=committed,itemType=javax.management.openmbean.SimpleType(name=java.lang.Long)),(itemType=init,itemType=javax.management.openmbean.SimpleType(name=java.lang.Long)),(itemType=max,itemType=javax.management.openmbean.SimpleType(name=java.lang.Long)),(itemType=used,itemType=javax.management.openmbean.SimpleType(name=java.lang.Long))))),contents={committed=99463920,used=20588920})
NonHeapMemoryUsage	javax.management.openmbean.CompositeDataSupport(compositeType=javax.management.openmbean.CompositeType(name=((itemType=committed,itemType=javax.management.openmbean.SimpleType(name=java.lang.Long)),(itemType=init,itemType=javax.management.openmbean.SimpleType(name=java.lang.Long)),(itemType=max,itemType=javax.management.openmbean.SimpleType(name=java.lang.Long)),(itemType=used,itemType=javax.management.openmbean.SimpleType(name=java.lang.Long))))),contents={committed=45088704,used=44770368})
RuntimeName	1252@spectrum8-sp2

## Operations

Name	Return type	Description
------	-------------	-------------

メモリ使用量 (HeapMemoryUsage と NonHeapMemoryUsage) は、標準の JVM メモリ MBean をベースとしています。リモートコンポーネントを実行している JVM のメモリ使用量を表します。この中では、init、max、committed、used の各メモリ容量が示されます。

RuntimeName にはプロセス ID が含まれているため、オペレーティングシステムで詳細情報を確認したり (たとえば Windows のタスク マネージャーを使用)、プロセスを停止するために使用できません。

ヒープセクションの `{committed=143130624, init=134217728, max=1908932608, used=23483928}` はバイト単位の値です。

init は JVM が最初に割り当てる容量で、`-Xms` で指定する値に相当します。max は `-Xmx` で指定する値に相当します。used は JVM がオブジェクトに使用しているメモリ容量です。これらの関連性を表すと、`-Xms < committed < -Xmx` および `used < committed` となります。

ヒープメモリを変更するには、spatial フォルダにある `java.vmargs` ファイル (<インストール先>\Pitney Bowes\Spectrum\server\modules\spatial\java.vmargs) で `-Xm` を変更します。詳細については、「[ヒープメモリの増加](#)」を参照してください。

# 5 - パフォーマンス チューニング

このセクションでは、メモリ管理とスレッド処理によってパフォーマンスを向上する方法について説明します。また、Location Intelligence モジュールのパフォーマンスを最適化するためのベスト プラクティスを紹介します。経験豊富な管理者向けの内容です。

Spectrum では、サーバーのパフォーマンスを最適化するためのいくつかのチューニング オプションが提供されています。どの設定が最適であるかは、展開の性質に依存します。適切にチューニングされたサーバー環境を構築するために、展開済み環境においてパフォーマンス テストを実施して、最適な設定を判断することを推奨します。このセクションでは、パフォーマンス チューニングに関する一般的な指針を示します。

## このセクションの構成

---

リモート コンポーネントの構成設定	72
データ ソース プーリングの設定	73
距離ベースの操作のパフォーマンスの向上	73

## リモート コンポーネントの構成設定

Spectrum™ Technology Platform のすべての空間サービスは、プラットフォーム ランタイムから独立したリモート コンポーネント (JVM インスタンス) に展開されます。これによって、プラットフォームはそれに含まれるモジュールに依存せず、また、JVM 構成は空間サービスごとに適用可能となるため、メモリ割り当ての柔軟性が高まり、サービスの特性に応じたパフォーマンス チューニングが可能になります。

リモート コンポーネントは、空間サービス (Feature Service や Mapping Service など) とステージ (Spatial Calculator や Query Spatial Data など) に空間機能を提供します。リモート コンポーネントのプール サイズは、コンポーネントが同時に処理できる要求の数です。この設定は、空間サービスと空間ステージの両方のスループットに影響を与えます。

空間リモート コンポーネントの権限を管理するには、他のセキュア エンティティ タイプと同じ方法で Management Console を使用します。空間リモート コンポーネントは、"Spatial Component" セキュア エンティティ タイプとして **[データベース リソース]** グループに表示されます。空間リモート コンポーネントの権限は、役割の作成時または編集時に設定したり、アクセス制御設定を使用して設定することができます。詳細については、[セキュリティの管理](#) (18ページ) を参照してください。

## プール サイズの変更

JVM の調整に加えて、空間リモート コンポーネントのプール サイズも調整できます。リモート コンポーネントのプール サイズは、コンポーネントが同時に処理できる要求の数です。この設定は、Spectrum™ Technology Platform からのサービス要求を待機したり、Location Intelligence モジュールのステージを実行したりするコンポーネント上のスレッド数(つまり管理対象の接続の最大数)を表します。

すべての Web サービスリクエストは、プラットフォームから Spectrum に入り、コンポーネントへと引き渡されます。デフォルト値の 1 より大きくすることで、高いリクエスト負荷に対応できます。CPU の数に一致するプール サイズが推奨されます。この最大値を、CPU コア数の 2 倍以上に設定しないでください。例えば、CPU が 4 基のマシンでは、全サービスのスレッドの合計数が 8 を超えないようにします。使用状況に対して最適なパフォーマンスが得られるように、さまざまな設定でパフォーマンス テストを実施する必要があります。

空間リモート コンポーネントのプール サイズは Management Console で調整できます。

1. Management Console を開きます。

2. [リソース] > [Location Intelligence] に移動します。
3. 矢印を使うか値を入力して、リモート コンポーネントのプール サイズを変更します。最小値は 1 で、最大値は 64 です。
4. [保存] をクリックします。
5. プール サイズを小さくした場合は、サーバーを再起動します。プール サイズを増やした場合はすぐに有効になるので、サーバーを再起動する必要はありません。

## データ ソース プーリングの設定

pooling-datasource-factory.properties ファイル (\server\modules\spatial にあります) を使用して、JDBC ベースのデータ ソース (Oracle、SQL Server など) で使用される接続のプーリングを、最適なパフォーマンスが得られるように設定できます。

ほとんどの場合に対し、バリデータ クラスを有効にすることが推奨されます。これにより、プールから借用する前にオブジェクトを検証できます。検証に失敗した場合は、その接続がプールから破棄され、別の接続の借用を試みます。検証クエリは、カスタム データ プロバイダを使用する場合など、特殊なケースに対しても使用できます。検証クエリとバリデータ クラスの両方を有効にした場合は、バリデータ クラスが使用されます。

検証を有効にすると、パフォーマンスが若干低下する可能性があります。ただし、このテストクエリにより、Spectrum Spatial と外部データベース間の通信に信頼性がない場合でも、コネクションプール内のすべての接続の整合性が維持されます。検証によるパフォーマンスへの影響が少なくなるように、検証間隔を設定します。ある接続を検証するタイミングになっても、この間隔内に検証が既に済んでいる場合は、その接続が再び検証されることはありません。

## 距離ベースの操作のパフォーマンスの向上

PGD インデックス ファイルは、TAB ファイル セットを補足するファイルであり、ネイティブ、ネイティブ拡張 (NativeX)、およびシームレス TAB で GSB ファイルに匹敵するパフォーマンスを発揮することができます。ラインとポリゴンを含むネイティブ データセットに対する、特定の距離ベースの操作のパフォーマンスを向上させるために、この特殊なインデックス ファイルを作成する、PGD Builder というコマンドライン ユーティリティが用意されています。PGD Builder を使用して作成したインデックスは、ラインとリージョンに基づくデータを検索していて、以下を使用している場合に有効です。



- Point in Polygon ステージで、距離を含める場合
- Find Nearest ステージで、(距離を含めるかどうかにかかわらず) 入力ポイントである場合
- Feature サービスの SearchNearest 操作で、入力ポイントと、ラインまたはポリゴンの検索テーブルを使用する場合

PGD Builder ユーティリティは、ようこそページの [Spectrum Spatial] セクションにある、[ユーティリティ] タブの **[PGD Builder]** からダウンロードできます。ドキュメントへのリンクも、ようこそページのユーティリティ ダウンロード リンクの横にあります。

注：PGD ファイルは、TAB 用の .MAP ファイルの 5 ～ 6 倍のサイズになります。1 つの TAB ファイルに対して 1 つの PGD ファイルが生成されます。ただし、シームレス TAB の場合は、各サブ TAB に対して PGD ファイルが作成されます。

また、TAB 内のデータを変更すると (行の追加や削除、または、TAB の MAP 部分のジオメトリの変更)、PGD ファイルはシステムによって使用されなくなります。警告が有効である場合は (「[Spatial のログ \(63ページ\)](#)」を参照)、有効期限切れの PGD ファイルに関するメッセージが、wrapper.log、または、Spatial のログ用に設定されたログ ファイルがある場合はそれに表示されます。メッセージが表示された場合は、更新された TAB ファイルに対する PGD を再生成する必要があります。

# 6 - クラスタの管理

## このセクションの構成

---

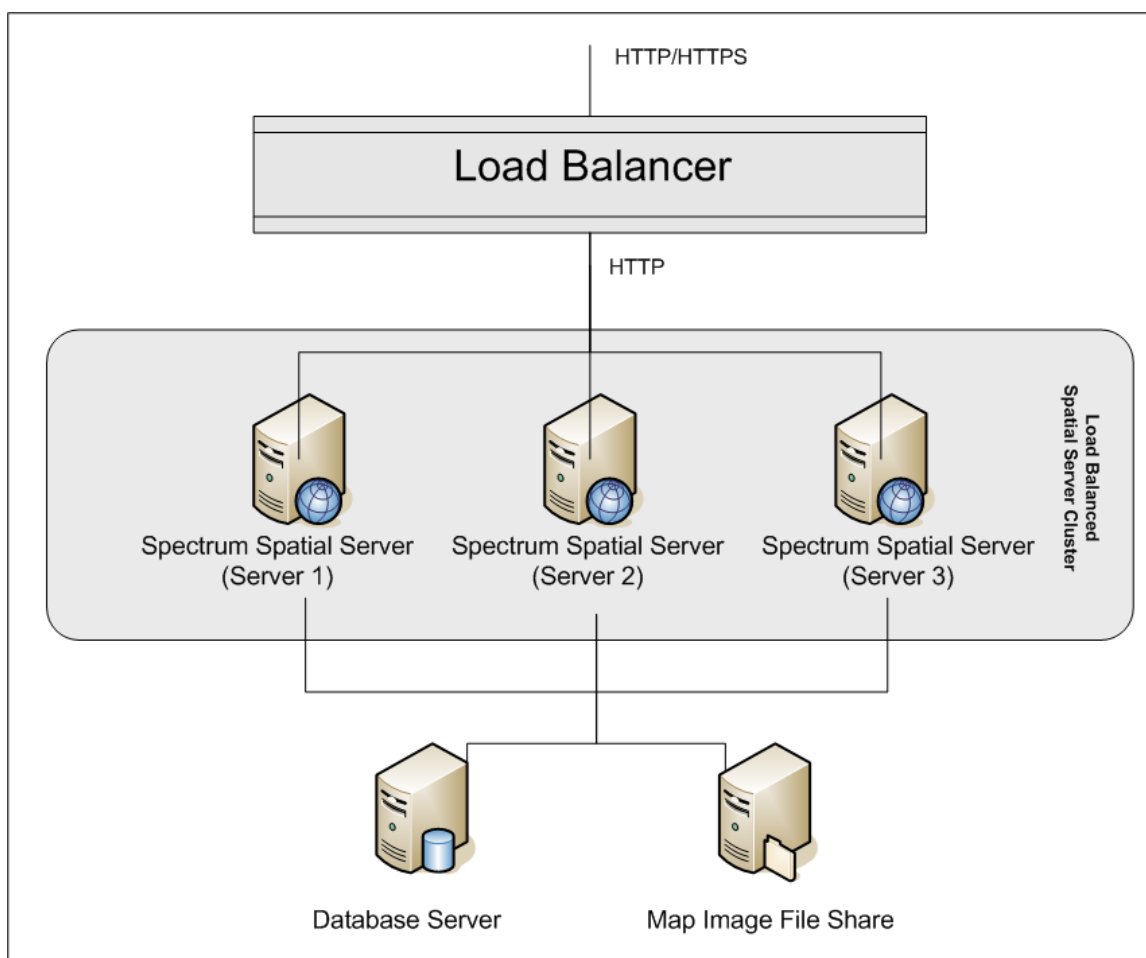
Location Intelligence モジュール用のクラスタ アーキテクチャ	76
クラスタでの Enterprise Designer の使用	78
クラスタの起動	79
クラスタの停止	80
クラスタからのノードの削除	80
Location Intelligence モジュール用のクラスタの管理	81

## Location Intelligence モジュール用のクラスタ アーキテクチャ

クラスタ環境では、サーバーの複数のインスタンスが処理を共有します。以下の図は、このような構成の展開アーキテクチャを示したものです。負荷分散を利用することによって、高い可用性とスケーリングをサポートすることができます。展開アーキテクチャは、ロード バランサー (負荷分散サーバー)、Spectrum Spatial クラスタ、データベース、ファイル共有で構成されます。このアプローチによって、水平および垂直の両方のスケーリングが可能になります。Location Intelligence モジュールは、プラットフォームをクラスタリングしてもしなくてもクラスタ化できます。

**注：** Spectrum™ Technology Platform クラスタと Location Intelligence モジュール クラスタの両方を設定することを推奨します。これには、以下に示すいくつかの利点があります。

- 名前付きリソースのセキュリティ (ACL) 同期が自動的に行われます。
- 1つのノード上で作成されたデータフロー、ユーザ、役割が、すべてのノードに自動的に同期されます。
- Location Intelligence モジュールのすべてのデモ ページとユーティリティ (Spatial Manager など) がロード バランサーを参照でき、参照する必要があります。



### ロード バランサー

ロードバランサーは、Spectrum Spatial インスタンス間にリクエストを分散します。HTTP/HTTPS リクエストの負荷分散をサポートする任意のロード バランサーが使用できます。

### Spectrum Spatial クラスタ

クラスタは、LIM 共有管理、名前付きリソース、地理メタデータ コンテンツ、および構成設定を伴う Spectrum インスタンスのコレクションです。クラスタにさらにノードを追加することによって、耐障害性を高めたり、さらに高い負荷に対応したりすることができます。各ノードは、ハードウェアリソースを追加するか、または、大規模リソースを伴うハードウェアに対して必要である場合はインスタンスを追加することによって、垂直にスケーリング可能です。Spectrum は、限られた数の CPU を使用するように構成できます。

### データベース

Spectrum は名前付きリソース(マップ、レイヤ、テーブル、スタイル)、地理メタデータ、構成をリポジトリに保存します。デフォルトのシングルサーバー構成では、組み込みデータベースを使

用して、ローカルサーバー上にこれらのリソースを保存します。耐障害性に優れたスケラブルなソリューションを構築するには、この組み込みデータベースを耐障害性を備えた独立したデータベースに置き換える必要があります。Oracle、PostgreSQL/PostGIS、Microsoft SQL Server が、リポジトリ データベースとしてサポートされています。

負荷分散構成において、Spectrum ノードはローカルキャッシュにこれらのリソースをキャッシュし、クラスタの各ノード内のインデックスを検索します。Spectrum ノードはリクエストを受信すると、ローカル キャッシュとインデックスを使用してリソースを検索します。名前付きリソースは、クラスタ内の任意のノードを介して追加できます。各ノードは、ローカル キャッシュと中央データベースとの違いをチェックして、キャッシュを最新の状態に保ちます。このチェックは、デフォルトで2秒ごとに実行されます。チェック時間の間隔は設定可能です。このアーキテクチャによって、サーバーは高いパフォーマンスでトランザクションを処理することができ、リポジトリ データベースの負荷は最小限に抑えられることとなります。新しい Spectrum ノードがクラスタに追加されると、キャッシュとインデックスが自動的に作成されます。新しい Spectrum の追加は、障害が発生したノードを復旧したい場合や展開のキャパシティを拡大したい場合に、行われる可能性があります。

### ファイル共有

ファイル共有は、Spectrum によって生成されたマップ イメージを格納するためのフォルダを提供します。Web サービスを使用してマップをレンダリングする際、サーバーは、URL を介して返されるマップ イメージまたは base 64 エンコード イメージとして返されるマップ イメージをサポートします。URL が返される場合、マップ イメージはファイルとして保存され、その URL がリクエストされた場合に提供されます。任意の Spectrum ノードがマップ イメージを返せるようにするために、イメージの保存にファイル シェアが使用されます。

## クラスタでの Enterprise Designer の使用

1. Enterprise Designer を起動します。
2. **[サーバー名]** フィールドに、ロード バランサーのサーバー名を入力します。
3. **[ポート]** フィールドに、ロード バランサーがリッスンするように設定したポートを入力します。

注：入力ファイル、出力ファイル、およびデータベース リソースは、共有ドライブ、ファイルサーバー、またはその他の共通にアクセスできる場所に置く必要があります。そうしない場合は、Spectrum™ Technology Platform サーバーをホストする各サーバーにすべてのファイルをロードする必要があり、すべてのファイルが同じパスに存在する必要があります。

ログインした後は、普通に Enterprise Designer を使用できます。実行するアクションは、ログインしているクラスタのすべての Spectrum™ Technology Platform インスタンスに対して適用されます。

## クラスタの起動

クラスタ内のすべてのノードが停止した場合は、次の手順に従ってクラスタを安全に起動し、データの損失を回避する必要があります。

1. 最後に停止したノード上で、シード ノードを削除し、サーバーを起動します。

**警告：**最後に停止したノードを最初に起動する必要があります。また、このノードがシードノードである必要もあります。別のノードを最初に起動すると、ジョブ履歴や構成設定などのデータが失われる恐れがあります。最後に停止したノードがわからない場合は、各ノードの `wrapper.log` でシャットダウン メッセージのタイム スタンプを確認します。`wrapper.log` は、`Spectrum Location\server\app\repository\logs\wrapper.log`にあります。

- a) 次のファイルをテキスト エディターで開きます。

```
server/app/conf/spectrum-container.properties
```

- b) `spectrum.cluster.seeds` プロパティから、このサーバーのものを除くすべてのホスト名と IP アドレスを削除します。ホスト名と IP アドレスは、後で再度追加できるように保存しておきます。
- c) ファイルを保存します。
- d) サーバーを開始します。
- e) Spectrum™ Technology Platformサーバーが完全に起動するまで待ちます。

Spectrum™ Technology Platformサーバーが完全に起動したかどうかは、`wrapper.log` (`Spectrum Location\server\app\repository\logs\wrapper.log`) の中を確認すればわかります。サーバーが完全に起動している場合は、次のメッセージが表示されません。

```
Pitney Bowes Spectrum(TM) Technology Platform (Version Version Number) Started.
```

- f) プロパティファイル `spectrum-container.properties` の `spectrum.cluster.seeds` プロパティに、削除したホスト名または IP アドレスをそれぞれカンマで区切って追加します。
  - g) ファイルを保存して閉じます。サーバーを再起動する必要はありません。
2. クラスタ内の他のノードを起動します。



**警告:** 必ず最初のノードが完全に起動するのを待ってから、他のノードを起動してください。最初のノードが起動する前に他のノードを起動すると、データが失われる恐れがあります。

## クラスタの停止

クラスタ全体を停止するには

1. シード ノードとして使用されているノードを識別します。これを行うには、`SpectrumFolder/server/app/conf/spectrum-container.properties` ファイルを開き、`spectrum.cluser.seeds` プロパティに示される一連のノードに注目します。
2. クラスタに含まれる Spectrum™ Technology Platform サーバーを 1 つずつ停止します。最後に停止するノードが必ずシード ノードになるようにします。

- Unix または Linux では、作業ディレクトリを Spectrum™ Technology Platform サーバーの `bin` ディレクトリに変更し、セットアップ ファイルのソースを指定して、コマンド `./server.stop` を入力します。
- Windows の場合は、Windows システム トレイの Spectrum™ Technology Platform アイコンを右クリックして、**[Spectrum™ を停止する]** を選択します。

**警告:** データが失われるのを防ぐため、最後に停止するノードはシード ノードである必要があります。

3. 最後に停止したノードをメモに書き留めておいてください。この情報は、クラスタの起動時に必要になります。

**警告:** クラスタの起動時にデータが失われるのを防ぐため、最後に停止したノードを最初に起動する必要があります。また、そのノードはシード ノードである必要があります。

## クラスタからのノードの削除

クラスタからノードを削除するには、Spectrum™ Technology Platform サーバーを停止します。

1. 削除するノードを、次の手順で停止します。
  - Unix または Linux では、作業ディレクトリを Spectrum™ Technology Platform サーバーの `bin` ディレクトリに変更し、セットアップ ファイルのソースを指定して、コマンド `./server.stop` を入力します。

- Windows の場合は、システムトレイの Spectrum™ Technology Platform アイコンを右クリックして、**[Spectrum™ を停止する]** を選択します。
2. ファイル `server/app/conf/spectrum-container.properties` をテキスト エディターで開き、`spectrum.cluster.enabled` を `false` に設定します。
  3. クラスタ内の他の各ノード上で、ファイル `spectrum-container.properties` を開き、`spectrum.cluster.seeds` プロパティからノードを削除します。

**Location Intelligence モジュールを使用する場合:** ノードをスタンドアロンのままにして、クラスタの外部で実行できるようにするには、元の `repository.xml` ファイルをコピーして戻し、Spectrum™ Technology Platform の各インスタンスの `/server/modules/spatial/jackrabbit` ディレクトリから、`repository`、`version`、`workspaces` というフォルダを削除します。サーバーを再起動し、リポジトリのコンテンツをインポートします。

## Location Intelligence モジュール用のクラスタの管理

### 共通リポジトリ データベースのセットアップ

クラスタに対して共通リポジトリ データベースを使用するように、Location Intelligence モジュールを設定する必要があります。これによって、名前付きリソース、地理メタデータ、構成設定が、クラスタ全体で管理されるようになります。

リポジトリは、一連の名前付きリソース、地理メタデータ、構成ファイルとともにインストールされています。これらのリソースを共通データベース リポジトリに移行するには、リソースをデフォルトの内部リポジトリ データベースから新しい共有リポジトリ データベースへとエクスポートする必要があります。

リポジトリ コンテンツを一括でエクスポートまたはインポートする場合は `limrepo import`、管理ユーティリティの `limrepo export` コマンドと `limrepo import` コマンドを使用します。これらのコマンドには、権限を維持するオプションがあります (手順については『*Spectrum Spatial ガイド*』の「管理」セクションを参照してください)。

PostgreSQL、Oracle、または Microsoft SQL Server のいずれかの共通データベースにリポジトリをセットアップするには、次の手順を実行します。

1. 管理ユーティリティの `limrepo export` コマンドを使用して、すべてのリポジトリ リソースをローカルフォルダにエクスポートします (手順については『*Spectrum Spatial ガイド*』の「管理」セクションを参照してください)。

インストールしたりリポジトリのコンテンツをエクスポートする必要があります。このステップは 1 回だけ実行します。Spectrum™ Technology Platform のすべてのインスタンスに対し、この時点でのリポジトリのコンテンツを同一にするためです。

2. すべてのノード上で Spectrum™ Technology Platform サーバーを停止します (手順については [クラスタの停止](#) (80ページ) を参照してください)。
3. Spectrum™ Technology Platform のすべてのノード上で、共通データベースを指定するように設定を変更します。
  - a) repository.<databaseType>.xml の内容を、repository.xml にコピーします。このファイルは server/modules/spatial/jackrabbit フォルダにあります。ここで <databaseType> は、お使いのデータベースに対する適切なタイプです (postgres、oracle、または mssql)。
  - b) repository.xml で、次の処理を行います。
    - DataSource セクションを、サーバーのホスト名、ポート、データベース、ユーザ、パスワードで変更します。
    - Cluster セクションを変更して、Node1 のような個別のクラスタ ID を割り当てます。クラスタ内の後続のすべてのノードに一意的 ID を割り当てます (Node2、Node3 など)。
    - 変更を repository.xml に保存します。
  - c) /server/modules/spatial/jackrabbit フォルダから、repository、version、workspaces というフォルダを削除します。
4. データベースにこれまでにリポジトリ コンテンツが含まれていたことがある場合は、クリーンなリポジトリが作成できるようにデータベースからテーブルを削除しておく必要があります。以下のテーブルを削除する必要があります。

default_binval	security_binval
default_bundle	security_bundle
default_names	security_names
default_refs	security_refs
rep_fsentry	version_binval
rep_global_revision	version_bundle
rep_journal	version_names

rep\_local\_revisions

version\_refs

Oracle を使用している場合は、version\_seq\_names\_id, security\_seq\_names\_id、default\_seq\_names\_id も削除します。

5. シード ノード上にものみ、バックアップしたりポジトリ コンテンツをインポートします。
  - a) Spectrum™ Technology Platform サーバーを起動します (手順については「[クラスタの起動 \(79ページ\)](#)」を参照してください)。
  - b) limrepo import コマンドでシード ノードを指定して、コンテンツをインポートします。
6. クラスタ内の残りのノードを起動します (手順については「[クラスタの起動 \(79ページ\)](#)」を参照してください)。

## システムの設定

Spectrum™ Technology Platform をインストールして共通リポジトリを設定した後、別の仮想マシンに複製する前に、インスタンスを構成設定する必要があります。仮想マシン環境を使用していない場合は、インストールされている各 Spectrum™ Technology Platform に対して以下の手順を実行する必要があります。

### マップ ファイル共有の設定

Spectrum™ Technology Platform にマップ ファイル共有 (共有イメージフォルダ) を設定するにはまず、共有マップ イメージ ディレクトリが必要です。マップ ファイル共有の作成については、「[Unix/Linux 上でのマップ イメージ ファイル共有の作成 \(84ページ\)](#)」または「[Windows 上でのマップ イメージ ファイル共有の作成 \(85ページ\)](#)」を参照してください。

マップ イメージ ディレクトリを作成したら、以下の手順でマップ ファイル共有を設定します。

1. マッピング サービス構成を、共有イメージフォルダと負荷分散サーバーを使用するように変更します。ImageCache において、Directory パラメータを共通イメージ ディレクトリに変更し、AccessBaseURL パラメータをロード バランサー マシンのイメージ URL に変更します。

仮想マシン環境を使用している場合は、この IP アドレスを覚えておいてください。ロード バランサー VM をこの IP に設定する必要があるためです。

Unix/Linux インストールの場合:

```
<ImageCache>
<Directory>/<spatial server
root>/server/modules/spatial/images</Directory>
```

```
<AccessBaseURL>http://<loadbalance_IP_address>/rest/Spatial/MappingService/internal/imageCache</AccessBaseURL>
<FileExpire>30</FileExpire>
<ScanInterval>30</ScanInterval>
</ImageCache>
```

Windows インストールの場合:

```
<ImageCache>
<Directory>\\server\Share\images</Directory>
<AccessBaseURL>http://<loadbalance_IP_address>/rest/Spatial/MappingService/internal/imageCache</AccessBaseURL>
<FileExpire>30</FileExpire>
<ScanInterval>30</ScanInterval>
</ImageCache>
```

2. Unix/Linux インストールの場合は、マップイメージを共有ファイルシステムに配置できるようにシンボリックリンクを設定する必要があります。

マウントされた共有フォルダに images サブフォルダを作成します。例: /mnt/<linux mount>/images。

```
cd /<spatial server root>/server/modules/spatial
rm -Rf images
ln -s /mnt/<linux mount>/images ./images
```

### Unix/Linux 上でのマップイメージファイル共有の作成

ファイル共有は、Spectrum Spatial によって生成されたマップイメージを格納するためのフォルダを提供します。すべての Spectrum ノードからアクセス可能な共有フォルダを作成します。マップが Base64 エンコード イメージとして Web サービスから返される場合は、ファイル共有は必要ありません。

Unix/Linux 上でのマップイメージファイル共有を作成するには

1. Spectrum をホスティングする各オペレーティングシステム上で、共有フォルダをマウントします。以下のコマンドによって、Microsoft Windows Server または CIFS をサポートするネットワーク ドライブ上にドライブをマウントします。

```
mkdir /mnt/<linux mount>
mount -t cifs //<windows host>/<windows share> /mnt/<linux mount>-o
username=shareuser,password=sharepassword,domain=pbj
```

2. /etc/fstab において、起動時にイメージ シェアをロードするように設定します。

```
//<windows ip address for share>/share /path_to/mount cifs  
username=server_user,password=secret,_netdev 0 0
```

### Windows 上でのマップ イメージ ファイル共有の作成

ファイル共有は、Spectrum Spatial によって生成されたマップ イメージを格納するためのフォルダを提供します。すべての Spectrum ノードからアクセス可能な共有フォルダを作成します。マップが Base64 エンコード イメージとして Web サービスから返される場合は、ファイル共有は必要ありません。

Windows 上でのマップ イメージ ファイル共有を作成するには

1. Windows エクスプローラで、共有するイメージ フォルダを選択します。
2. 右クリックして **[共有]** をクリックします。
3. イメージフォルダに書き込みを行うユーザを選択します。これらのユーザは読み取り/書き込み権限を持つ必要があります。

### クラスタ用の OGC サービス設定の変更

Spectrum™ Technology Platform クラスタと Location Intelligence モジュール クラスタの両方がある場合のクラスタ環境が正しく動作するように、Spatial Manager によって OGC サービス設定ファイルを変更する必要があります。WFS、WMS、WMTS の設定ページで、オンライン リソース (サービス) URL をロード バランサーの IP アドレスおよびポートに変更します。詳細については、『Spectrum Spatial ガイド』の「ユーティリティ」セクションにある「Spatial Manager ガイド」を参照してください。

### 全ノードの Java プロパティの変更

クラスタのすべてのノードの java プロパティ ファイルを変更する必要があります。Spectrum™ Technology Platform の Java プロパティを変更するには、次の手順を実行します。

1. java.properties ファイル (<spectrum>/server/modules/spatial/java.properties) を変更して、repository.host が localhost を指すようにします。
2. images.webapp.url と、すべてのサービス ホストおよびポート番号を変更して、負荷分散サーバーを指すようにします。

### 複数の Spectrum インスタンスのポートの構成

単一のマシン上に複数の Spectrum™ Technology Platform のインスタンスがある場合は、ポート番号を変更する必要があります。



Spectrum™ Technology Platform の各インスタンスのポート番号を変更するには、次の手順を実行します。

1. <Spectrum root>/server/app/conf/spectrum-container.properties 内のすべてのポートを、使用されていない新しいポートに変更します。http ポートは、インストーラで入力されたポート番号に対応します。
2. /<spectrum root>/server/modules/spatial フォルダにある bootstrap.properties の rmi ポートを更新します (11099 など)。デフォルトは 1099 です。

### 共有される Spectrum ローカル データ

ファイル システム上の TAB ファイル データを使用する場合は、このデータを負荷分散環境に含まれるすべての Spectrum インスタンスがアクセス可能な共有場所に配置する必要があります。また、ファイル システム上のデータにアクセスするリポジトリ内のすべての名前付きリソースが、この共有場所を指す必要があるということも、重要な点です。

Spectrum をホスティングする各 VM またはマシンは、マウントされた共有ドライブにアクセスできる必要があります。

注：データベース テーブルを指す名前付きリソースを使用するのに、共有ドライブは必要ではありません。リポジトリ内の名前付きリソースは、ファイル パスを使用してデータにアクセスするのではなく、データベース内のデータへの名前付き接続を使用するためです。

# 7 - 管理ユーティリティの使用

## このセクションの構成

---

管理ユーティリティを使用する前に	88
管理ユーティリティをスクリプトから使用する	89
Location Intelligence モジュール	91
Enterprise Routing モジュール	97

## 管理ユーティリティを使用する前に

管理ユーティリティでは、管理機能をコマンドラインから実行できます。この機能はスクリプトで利用できるため、特定の管理タスクを自動化できます。対話式の操作で機能を実行することもできます。すべての管理機能を管理ユーティリティで利用できるわけではありません。Management Console を使用すると、管理ユーティリティでは利用できない機能にアクセスできます。

注：管理ユーティリティには Java 8 以降が必要です。管理ユーティリティを実行する前に、Java 8 がシステムのパスに含まれていることを確認してください。

1. Web ブラウザを起動し、次の Spectrum™ Technology Platform の Welcome ページを開きます。

`http://<サーバー名>:<ポート>`

例えば、Spectrum™ Technology Platform が "myspectrumplatform" という名前のコンピュータにインストールされており、デフォルトの HTTP ポート 8080 を使用している場合は、次のアドレスに移動します。

`http://myspectrumplatform:8080`

2. **[プラットフォーム クライアント ツール]** をクリックします。
3. **[コマンドライン]** をクリックします。
4. **[管理ユーティリティ]** で **[ダウンロード]** をクリックして、管理ユーティリティを使用するコンピュータに zip ファイルをダウンロードします。
5. zip ファイルの内容を解凍します。
6. コマンドライン インターフェイスを起動するには、次のいずれかの操作を実行します。
  - サーバーが Unix または Linux システムで実行されている場合は、`cli.sh` を実行します。
  - サーバーが Windows システムで実行されている場合は、`cli.cmd` を実行します。

注：必要に応じて、Java のインストールパスが正しく使用されるように `.sh` または `.cmd` ファイルを変更します。

7. 次のコマンドを入力して Spectrum™ Technology Platform サーバーに接続します。

```
connect --h servername:port --u username --p password --s SSLTrueOrFalse
```

例を次に示します。

```
connect --h myserver:8080 --u admin --p myPassword1--s true
```

8. 接続が完了すると、コマンドを実行できます。コマンドのヒントを紹介します。

- 使用できるコマンドの一覧を表示するには、`help` と入力するか、**Tab** キーを押します。
  - コマンドを自動補完するには、最初の数文字を入力してから **Tab** キーを押します。例えば、`us` と入力してから **Tab** キーを押すと、自動補完されて `user` コマンドになります。もう一度 **Tab** キーを押すと、`user` コマンドの全一覧が表示されます。
  - 空白を含むオプション値を指定する場合は、値を二重引用符で囲みます。
9. 作業が終わったら、`exit` コマンドを入力して管理ユーティリティを終了します。

## 管理ユーティリティをスクリプトから使用する

管理ユーティリティのいくつかのコマンドは、スクリプト ファイルから実行できます。管理ユーティリティや **Management Console** でコマンドを手動で実行する代わりにスクリプトを使用して管理業務を自動化または規格化したい場合に、これは便利です。

1. テキストエディタでスクリプトファイルを作成します。スクリプトファイルには、実行したいコマンドを記述します。

コマンドをスクリプト ファイルに追加するには、コマンドプロンプトでコマンドを入力する場合と同様にコマンドと必要なパラメータを入力します。コマンドは、1 行に 1 つ入力します。

スクリプト ファイルにコメントを挿入する場合は、次の表記方法を使用します。

- `/*`            コメント ブロックの開始を示します。
- `*/`            コメント ブロックの終了を示します。
- `//`            インライン コメントを示します。行の先頭のみを使用します。
- `;`            インライン コメントを示します。行の先頭のみを使用します。

2. スクリプトは、管理ユーティリティを実行するコンピュータに保存するか、管理ユーティリティを実行するコンピュータからアクセスが可能な場所に保存します。ファイル名と拡張子は任意に選択できます。ファイル拡張子 `.cli` の使用を推奨します。
3. スクリプトを実行するには、次のいずれかの操作を行います。

### オプション

### 説明

コマンドラインでスクリプトを実行するには

コマンドライン、またはバッチ スクリプトやシェル スクリプトの中で、以下のように指定します。

```
cli.cmd --cmdfile ScriptFile
```

## オプション

## 説明

管理ユーティリティでスクリプトを実行するには 管理ユーティリティを開き、Spectrum™ Technology Platform コマンドを使用して connect サーバーに接続します。続いて script コマンドを使用してスクリプトを実行します。このコマンドの詳細については、[script](#)を参照してください。

**例: データフローをステージングから実稼働に移動する**

Deduplication、AddressValidation、および DrivingDirections の3つのデータフローがあります。これらのデータフローに修正を加え、テストするためのステージングサーバーと、データフローを実行可能にする実稼働環境があります。これらのデータフローをステージングサーバーから実稼働サーバーに移動する作業に一貫性を与え、自動化する必要があるため、管理ユーティリティ スクリプトを利用することにしました。使用するスクリプトは、次のような内容です。

```
// Connect to the staging server
connect --h stagingserver:8080 --u allan12 --p something123

// Export from staging
dataflow export --d "Deduplication" --e true --o exported
dataflow export --d "AddressValidation" --e true --o exported
dataflow export --d "DrivingDirections" --e true --o exported

// Close connection to the staging server
close

// Connect to the production server
connect --h productionserver:8080 --u allan12 --p something123

// Import to production
dataflow import --f exported\Deduplication.df
dataflow import --f exported\AddressValidation.df
dataflow import --f exported\DrivingDirections.df

// Close the connection to the production server
close
```

## Location Intelligence モジュール

### limrepo export

注：管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#)（88ページ）」を参照してください。

limrepo export コマンドは、名前付きリソース (名前付きテーブルなど) を Spectrum Spatial リポジトリからローカル ファイル システムへエクスポートします。このコマンドを使うには、Location Intelligence モジュールがインストールされている必要があります。

リソースはリポジトリのフルパスでターゲットのフォルダにエクスポートされます。例えば、limrepo export --s /Samples/NamedTables --o C:\export を実行すると、C:\export\Samples\NamedTables\WorldTable が作成され、NamedTables フォルダ (ディレクトリ) の下に名前付きテーブルがエクスポートされます。

注：limrepo export コマンドは必ず、空のフォルダを含むすべてのフォルダを再帰的にエクスポートします。

#### 使用方法

```
limrepo export --s SourceRepositoryPath --o OutputFilePath
```

注：パラメータのリストを表示するには、“help limrepo export” と入力します。

必須	引数	説明
はい	--s <i>SourceRepositoryPath</i>	エクスポートするリソースまたはフォルダへのパスを指定します。
はい	--o <i>OutputFilePath</i>	エクスポート先のローカル ファイル システム上のフォルダへのパスを指定します。新規フォルダでも既存フォルダでもかまいませんが、既存フォルダの場合は中身を空にしておく必要があります。空でない場合、エクスポートは失敗します。



必須	引数	説明
いいえ	<code>--q</code> or <code>--quiet</code>	<p>エクスポート中にコピー済みリソースの表示を無効にします (すなわち、クワイエットモードで動作します)。</p> <p>このフラグを指定した場合のデフォルト値は <b>true</b> です。このフラグを指定しない場合のデフォルト値は <b>false</b> です。</p>
いいえ	<code>--f</code> or <code>--fullpaths</code>	<p>エクスポート元と出力先のフルパスをプリントします。</p> <p>このフラグを指定した場合のデフォルト値は <b>true</b> です。このフラグを指定しない場合のデフォルト値は <b>false</b> です。</p>
いいえ	<code>--r</code> or <code>--recursive</code>	<p>サブフォルダ (指定されたエクスポート元フォルダの下にあるフォルダ) を再帰的にエクスポートします。</p> <p>このフラグを指定した場合のデフォルト値は <b>true</b> です。このフラグを指定しない場合のデフォルト値は <b>true</b> です。</p>
いいえ	<code>--c</code> or <code>--continueonerror</code>	<p>エラーが発生した場合もエクスポートを続行します。</p> <p>このフラグを指定した場合のデフォルト値は <b>true</b> です。このフラグを指定しない場合のデフォルト値は <b>false</b> です。</p>
いいえ	<code>--a</code> or <code>--acl</code>	<p>エクスポートするリソースの既存の権限をローカルファイルシステム上のエクスポート先ファイルでも維持します。アクセス制御リスト (ACL) は、ユーザーまたは役割が名前付きリソースに実行できる操作 (作成、表示、編集、削除など) を定義します。</p> <p>このフラグを指定した場合のデフォルト値は <b>true</b> です。このフラグを指定しない場合のデフォルト値は <b>false</b> です。</p>

**例**

この例は、リポジトリの \Samples フォルダにある名前付きリソースをローカルファイルシステムの C:\myrepository\samples にエクスポートします。

```
limrepo export --s /Samples --o C:\myrepository\samples
```

## limrepo import

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

limrepo import コマンドは、名前付きリソース (名前付きテーブルなど) をローカルファイルシステムから **Spectrum Spatial** リポジトリへインポートします。このコマンドを使うには、**Location Intelligence** モジュールがインストールされている必要があります。

リソースをインポートするには、以前にリソースをエクスポートしたフォルダまたはディレクトリを指定する必要があります。例えば、limrepo export --s /Samples/NamedTables --o C:\export を実行すると、C:\export\Samples\NamedTables\WorldTable が作成され、NamedTables フォルダ (ディレクトリ) の下に名前付きテーブルがエクスポートされます。リソースはリポジトリのフルパスでターゲットのフォルダにエクスポートされます。limrepo import --s C:\export を実行すると、WorldTable が /Samples/NamedTables/WorldTable へインポートされます。

**注：**limrepo import コマンドは必ず、空のフォルダを含むすべてのフォルダを再帰的にインポートします。

インポートを行った後は多くの場合、**Spatial Manager** を使用してその新しいパスを指すように名前付き接続を調整する必要があります。例えば、テスト インスタンス上で “C:\myfiles” にインストールされていたネイティブ TAB ファイルが “E:\ApplicationData\Spectrum\Spatial\Spring2016” にインストールされた場合、インポート後に **Spatial Manager** でその接続を修正する必要があります。**Spatial Manager** で名前付き接続を編集する手順については、『*Spectrum Spatial ガイド*』の「ユーティリティ」セクションを参照してください。

**注：**Spectrum™ Technology Platform の 12.0 以前のバージョンからエクスポートされたサービス設定ファイルを復元するために limrepo import を使用すると、そのファイルが自動的に修正され、バージョン 12.0 以降と互換性が成立します (例えばリポジトリ URL が削除されます)。

### 使用方法

```
limrepo import --s SourceFilePath
```

注：パラメータのリストを表示するには、“help limrepo import”と入力します。

必須	引数	説明
はい	<code>--s SourceFilePath</code>	ローカル ファイル システムからインポートするリソースまたはフォルダへのパスを指定します。これは、ローカル ファイル システム上の以前のエクスポートに使用したルートフォルダである必要があります。
いいえ	<code>--q</code> or <code>--quiet</code>	インポート中にコピー済みリソースの表示を無効にします (すなわち、クワイエットモードで動作します)。  このフラグを指定した場合のデフォルト値は <b>true</b> です。このフラグを指定しない場合のデフォルト値は <b>false</b> です。
いいえ	<code>--u</code> or <code>--update</code>	同じ名前のリソースがサーバーに既に存在する場合、この既存のリソースを上書きするかどうかを指定します。  <b>true</b> インポートするリソースと同じ名前のリソースがサーバーに存在する場合、サーバーにあるリソースを上書きされます。このフラグを指定しない場合、または値を設定しないでこのフラグを指定した場合、この動作がデフォルト設定です。  <b>false</b> インポートするリソースと同じ名前のリソースがサーバーに存在する場合、リソースはインポートされません。
いいえ	<code>--f</code> or <code>--fullpaths</code>	エクスポート元と出力先のフルパスをプリントします。  このフラグを指定した場合のデフォルト値は <b>true</b> です。このフラグを指定しない場合のデフォルト値は <b>false</b> です。

必須	引数	説明
いいえ	<code>--c</code> or <code>--continueonerror</code>	<p>エラーが発生した場合もインポートを続行します。</p> <p>このフラグを指定した場合のデフォルト値は <b>true</b> です。このフラグを指定しない場合のデフォルト値は <b>false</b> です。</p>
いいえ	<code>--a</code> or <code>--acl</code>	<p>以前にエクスポートされた権限を維持し、リソースのインポート中にそれらを既存の権限と統合します。アクセス制御リスト (ACL) は、各ユーザーまたは役割が名前付きリソースに実行できる操作 (作成、表示、変更、削除など) を定義します。</p> <p>例えば、エクスポートするリソースに対して、あるユーザーが読み取りと書き込みの権限を持っています。このリソースのインポート時に、このユーザーがそのリソースの読み取り権限しか持っていない場合、インポートが正常に完了した後で書き込み権限が与えられます。</p> <p>競合する権限は統合できず、無視されます。ターゲットのリポジトリに存在しないユーザーや役割の ACL 項目は無視されます。</p> <p>このフラグを指定した場合のデフォルト値は <b>true</b> です。このフラグを指定しない場合のデフォルト値は <b>false</b> です。</p> <p><b>ヒント:</b> このフラグを使用する場合、エクスポート元のサーバー上のユーザが、インポート先のサーバー上にも存在する必要があります。例えば、アクセス制御が設定された "testuser" が存在し、ACL 付きのリソースをあるサーバーからエクスポートして、"testuser" が存在しない別のサーバーに、それらの名前付きリソースをインポートするとします。この場合、名前付きリソースは ACL なしでアップロードされます。</p>

**例**

この例は、名前付きリソースをローカル ファイル システムの `C:\myrepository\samples` からインポートします。

```
limrepo import --s C:\myrepository\samples
```

## limrepo mwsimport

注：管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#)（88ページ）」を参照してください。

Spectrum™ Technology Platform管理ユーティリティの `limrepo mwsimport` により、MapInfo Pro または MapXtreme Workspace Manager で作成された MapInfo Workspace (MWS) ファイル内のマップを Spectrum Spatial リポジトリにプロビジョニングすることができます。インポートにより、名前付きマップとそのすべての従属リソース (レイヤ、テーブル、接続) が作成されます。接続には、マップ名に 'Connection' を付加した名前が付けられます。名前付きテーブルと名前付きレイヤは、サブフォルダに作成されます (それぞれ、NamedTables と NamedLayers)。

このコマンドを使うには、Location Intelligence モジュールがインストールされている必要があります。

### 使用方法

```
limrepo mwsimport --s MWSFilePath --o Output --p ServerPath
```

注：パラメータのリストを表示するには、“`help limrepo mwsimport`”と入力します。

必須	引数	説明
はい	<code>--s <i>MWSFilePath</i></code>	ローカル ファイル システムからインポートする MWS ファイルへのパスを指定します。
はい	<code>--o <i>Output</i></code>	リポジトリ上の名前付きマップへのパスを指定します。すべてのリソースが、名前付きマップとして同じフォルダ内に作成されます。
はい	<code>--p <i>ServerPath</i></code>	データが配置されているサーバー上の場所へのファイルパスを指定します。このパスを使用して名前付き接続が作成され、この名前付き接続は、作成されるすべての名前付きテーブルによって参照されます。これらのテーブルは、その名前付き接続への相対ファイルパスを使用します。
いいえ	<code>--l <i>LocalPath</i></code>	MWSにサーバーファイルシステム上に存在しないファイルシステムが含まれる場合に、データが配置されているローカルファイルシステム上の場所へのファイルパスを指定します。指定した

必須	引数	説明
		値がMWSファイル内に出現すると、それが指定したサーバーパスで置き換えられます。MWSファイルの中に部分パスがある場合は、これは不要です。MapXtremeで作成されたものならば通常、MWSファイルの中に部分パスがあります。

**例**

この例では、D:ドライブ上のMWSファイルをインポートし(サーバー上のデータはC:\mydataに存在します)、名前付きリソースをリポジトリ内の/Europe/Countriesに配置します。

```
limrepo mwsimport --s D:\europe.mws --o /Europe/Countries --p
C:\mydata
```

**結果**

以下の名前付きリソースが作成されます。

```
/Europe/Countries/Europe (名前付きマップ)
/Europe/Countries/EuropeConnection (名前付き接続)
/Europe/Countries/NamedTables/austria (名前付きテーブル)
/Europe/Countries/NamedTables/belgium (名前付きテーブル)
。
/Europe/Countries/NamedLayers/austria (名前付きレイヤ)
/Europe/Countries/NamedLayers/belgium (名前付きレイヤ)
..
```

## Enterprise Routing モジュール

### ermdb list

注：管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

ermdb list コマンドは、サーバー上の既存のルーティング データベース リソースの全一覧を取得します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

### 使用方法

ermdb list

#### 例

この例では、サーバー上のすべてのデータベース リソースを返します。

```
ermdb list
```

## ermdb get

注：管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#) (88ページ)」を参照してください。

ermdb get コマンドでは、サーバー上に設定されているルーティング データベースに関する情報を返すことができます。返される情報は、データベースの名前、ファイルシステム (パス) 上のデータベースの場所、データベース用に設定されているプール サイズです。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

### 使用方法

ermdb get --name *database\_name*

注：パラメータのリストを表示するには、"help ermdb get" と入力します。

必須	引数	説明
はい	--name or --n <i>database_name</i>	情報を返すデータベース リソースの名前を指定します。名前は、サーバー上で一意でなければなりません。既存のルーティング データベース リソースの一覧を表示するには、ermdb list コマンドを使用します。

#### 例

この例は、データベース リソース US の情報をサーバーから返します。

```
ermdb get --name US
```



## ermdb add

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#) (88ページ)」を参照してください。

ermdb add コマンドは、新しいルーティング データベース リソースをサーバー上に作成します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

**注：**ermdb add コマンドでは、追加する各データベースに対して一意の名前を使用する必要があります。

### 使用方法

```
ermdb add --name database_name --poolsize pool_size --path database_path
```

**注：**パラメータのリストを表示するには、“help ermdb add”と入力します。

必須	引数	説明
はい	--name or --n <i>database_name</i>	追加するデータベースリソースの名前を指定します。名前は、サーバー上で一意でなければなりません。既存のルーティング データベース リソースの一覧を表示するには、ermdb list コマンドを使用します。
いいえ	--poolsize or --s <i>pool_size</i>	データベースで処理する同時要求の最大数を指定します。指定しない場合のデフォルト値は 4 です。同時要求数としては、1～128 の任意の整数が指定可能です。
はい	--path <i>database_path</i>	ファイル サーバー上のルーティング データベースの場所を指定します。

### 例

この例では、データベース リソース US を E:  
/ERM-US/2014.09/driving/south からサーバーに追加します。

```
ermdb add --name US --poolsize 10 --path E:  
/ERM-US/2014.09/driving/south
```

## ermdb delete

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#)（88ページ）」を参照してください。

ermdb delete コマンドは、サーバー上の既存のルーティング データベース リソースを削除します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

### 使用方法

```
ermdb delete --name database_name
```

**注：**パラメータのリストを表示するには、“help ermdb delete” と入力します。

必須	引数	説明
はい	--name or --n <i>database_name</i>	削除するデータベースリソースの名前を指定します。既存のルーティング データベース リソースの一覧を表示するには、ermdb list コマンドを使用します。

### 例

この例では、データベース リソース **US** をサーバーから削除します。

```
ermdb delete --name US
```

## ermdb modify

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#)（88ページ）」を参照してください。

ermdb modify コマンドは、サーバー上の既存のルーティング データベース リソースを変更します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

### 使用方法

```
ermdb modify --name database_name --poolsize pool_size --path database_path
```

注：パラメータのリストを表示するには、“help ermdb modify”と入力します。

必須	引数	説明
はい	<code>--name or --n <i>database_name</i></code>	変更するデータベースリソースの名前を指定します。既存のルーティング データベース リソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
いいえ	<code>--poolsize or --s <i>pool_size</i></code>	データベースで処理する同時要求の最大数を指定します。同時要求数としては、1～128の任意の整数が指定可能です。新しいプールサイズと新しいデータベースパスのいずれかを指定する必要があります。
いいえ	<code>--path <i>database_path</i></code>	ファイル サーバー上のルーティング データベースの新しい場所を指定します。新しいプールサイズと新しいデータベースパスのいずれかを指定する必要があります。

#### 例

この例では、新しいバージョンのプール サイズとデータベース パスの両方を変更します。

```
ermdb modify --name US --poolsize 20 --path E:
/ERM-US/2015.03/driving/south
```

## ermdb import

注：管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#) (88ページ)」を参照してください。

`ermdb import` コマンドでは、ルーティング データベース設定を含むファイルをインポートして、データベース リソースをサーバー上に作成できます。インポート ファイルを作成するか、`ermdb export` コマンドで作成したファイルを使用できます。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

インポート ファイルの形式は以下のとおりです。

```
[ { "product": "Spatial", "module": "routing", "name": "US", "maxActive": 4, "properties": {
  "DatasetPaths": "E:/ERM-US/2014.09/driving/northeast" } } ]
```

ここで、product と module はそれぞれ **Spatial** と **routing** でなければなりません。name はデータベース名、maxActive はこのデータベースで処理する同時リクエストの最大数 (またはプールサイズ)、DatasetPaths はデータベース リソースのデータ セットへのパスです。

1つのインポート ファイルに複数のデータベースを追加できます (上の例をコピーします)。また、各データベース リソースに対して複数のデータセットをセミコロンで区切って追加できます。

**注:** インポート ファイルで UTF-8 文字を指定するには、CLI コマンド プロンプトの起動時に、値 UTF-8 を設定した JVM パラメータ file.encoding を追加する必要があります。例えば、-Dfile.encoding=UTF-8 とします。

## 使用方法

```
ermdb import --file file_name
```

**注:** パラメータのリストを表示するには、“help ermdb import”と入力します。

必須	引数	説明
はい	--file or --f <i>file_name</i>	インポート ファイルのディレクトリと名前を指定します。

### 例

この例では、それぞれ複数のデータセットで構成される、US1 と US2 という 2 つのデータベースをインポートします。

```
ermdb import --file E:/ERM-US/export/ermDbResource.txt
```

入力ファイルは次のように定義されています。

```
[{"product": "Spatial", "module": "routing", "name": "US1", "maxActive": 4, "properties":
{"DatasetPaths":
"E:/ERM-US/2014.09/driving/northeast;E:/ERM-US/2014.09/driving/south" }}, {
"product": "Spatial", "module": "routing", "name": "US2", "maxActive": 4, "properties":
{"DatasetPaths":
"E:/ERM-US/2014.09/driving/northwest;E:/ERM-US/2014.09/driving/central" } }]
```

## ermdb export

**注:** 管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#) (88ページ)」を参照してください。

ermdb export コマンドでは、サーバー上に設定されているルーティング データベースをファイルにエクスポートできます。このファイルはその後、バックアップや、あるインスタンスから

別のインスタンスへの移行を目的とした、`ermdb import` コマンドによる別のインスタンスへのインポートに使用できます。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

注：`ermdb export` コマンドは必ず、`ermDbResource.txt` という名前のエクスポートファイルを作成します。

### 使用方法

```
ermdb export --directory directory_name
```

注：パラメータのリストを表示するには、“`help ermdb export`”と入力します。

必須	引数	説明
いいえ	<code>--directory</code> or <code>--o</code> <code>directory_name</code>	データベース ファイルのエクスポート先となるファイルシステム上のディレクトリ名を指定します。このエクスポート コマンドは必ず、 <code>ermDbResource.txt</code> という名前のエクスポートファイルを作成します。このパラメータを指定しない場合は、このエクスポート コマンドを実行したディレクトリにエクスポートファイルが作成されます。

#### 例

この例では、`E:/ERM-US/export` ディレクトリにエクスポート データベース ファイルを作成します。

```
ermdb export --directory E:/ERM-US/export
```

## erm getpointdata

注：管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#) (88ページ)」を参照してください。

`erm getpointdata` コマンドは、ポイントに対するセグメント情報を返します。指定されたポイントに最も近い1つ以上のセグメントが返されます。返される情報としては、セグメント ID、道路タイプ、長さ、速度、方向、時間、道路名などがあります。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

## 使用方法

```
erm getpointdata --datasource db_resource --point "x,y,coordsys"
```

注：パラメータのリストを表示するには、“help erm getpointdata”と入力します。

必須	引数	説明
はい	--datasource <i>db_resource</i>	データを返すデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、ermdb list コマンドを使用します。
はい	--point " <i>x,y,coordsys</i> "	最も近いセグメントの情報を返すポイントを指定します。ポイントは、“ <i>x,y,coordsys</i> ”という形式で指定します。ここで、 <i>coordsys</i> はポイントの座標系です。

### 例

この例では、サーバー上に設定されている US\_NE データベースリソースから、指定されたポイントに最も近いセグメントのデータを返します。

```
erm getpointdata --datasource US_NE --point "-72,40,epsg:4326"
```

## erm getsegmentdata

注：管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#)（88ページ）」を参照してください。

erm getsegmentdata コマンドは、指定されたセグメント ID に対するセグメント情報を返します。返される情報としては、セグメント ID、道路タイプ、長さ、速度、方向、時間、道路名などがあります。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

## 使用方法

```
erm getsegmentdata --datasource db_resource --segmentid "segment_id"
```

注：パラメータのリストを表示するには、“help erm getsegmentdata”と入力します。

必須	引数	説明
はい	<code>--datasource <i>db_resource</i></code>	データを返すデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
はい	<code>--segmentid "<i>segment_id</i>"</code>	情報を返すセグメントを指定します。セグメントは、データ内で指定されている形式で指定します。例: <code>"7e3396fc:6e5251"</code> 。

**例**

この例では、サーバー上に設定されている `US_NE` データベースリソースから、指定されたセグメントのデータを返します。

```
erm getpointdata --datasource US_NE --segmentid
"7e3396fc:6e5251"
```

## erm createpointupdate

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#) (88ページ)」を参照してください。

`erm createpointupdate` コマンドは、指定されたポイントに最も近いセグメントのルーティングデータをオーバーライドします。このコマンドにより、速度を設定または変更したり、ルートのセクションを除外したりできます。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

**注：**永続更新タイプは特定のデータリソースに対してのみ有効で、データ更新後には有効でなくなる場合があります。

### 使用方法

```
erm createpointupdate --datasource db_resource --point "x,y,coordsys" --exclude
--velocity velocity_value --velocityunit velocity_unit --velocityadjustment
velocity_adjustment_value --velocitypercentage velocity_percentage_value
```

**注：**パラメータのリストを表示するには、“`help erm createpointupdate`”と入力します。



必須	引数	説明
はい	<code>--datasource <i>db_resource</i></code>	データをオーバーライドするデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
はい	<code>--point "<i>x,y,coordsys</i>"</code>	最も近いセグメントの情報をオーバーライドするポイントを指定します。ポイントは、" <i>x,y,coordsys</i> " という形式で指定します。ここで、 <i>coordsys</i> はポイントの座標系です。
いいえ	<code>--exclude</code>	指定されたポイントをすべてのルート計算から除外します。コマンドにこのパラメータを含めることにより、そのポイントを除外するかどうかを指定します。
いいえ	<code>--velocity <i>velocity_value</i></code>	速度更新を定義します。新しい速度を指定することにより、ポイントの新しい速度を指定します。 <code>velocityunit</code> パラメータを指定しない限り、デフォルトの単位は <b>mph</b> (マイル/時) です。
いいえ	<code>--velocityunit <i>velocity_unit</i></code>	<code>velocity</code> または <code>velocityadjustment</code> オーバーライドの速度単位を定義します。デフォルト値は <b>mph</b> (マイル/時) です。速度更新の場合、速度単位には <b>kph</b> (キロメートル/時)、 <b>mps</b> (メートル/秒)、 <b>mph</b> (マイル/時) のいずれかの値を指定できます。
いいえ	<code>--velocityadjustment <i>velocity_adjustment_value</i></code>	速度更新を定義します。速度(単位と値)の変更を指定することにより、ポイントの速度の変化を定義します。速度値は増加(正の値)または減少(負の値)させることができます。 <code>velocityunit</code> パラメータを指定しない限り、デフォルトの単位は <b>mph</b> (マイル/時) です。
いいえ	<code>--velocitypercentage <i>velocity_percentage_value</i></code>	速度更新を定義します。速度を増加(正の値)または減少(負の値)させる割合を指定することにより、ポイントの速度の増加を定義します。

**例**

この例では、サーバー上に設定されている **US\_NE** データベースリソースから、ポイントの速度を **15 mph** にオーバーライドします。

```
erm createpointupdate --datasource US_NE --point
"-72,40,epsg:4326" --velocity 15 --velocityunit mph
```

この例では、サーバー上に設定されている **US\_NE** データベースリソースから、指定されたポイントを除外します。

```
erm createpointupdate --datasource US_NE --point
"-72,40,epsg:4326" --exclude
```

この例では、サーバー上に設定されている **US\_NE** データベースリソースから、ポイントの速度を **45 kph** だけ増加させることによってオーバーライドします。

```
erm createpointupdate --datasource US_NE --point
"-72,40,epsg:4326" --velocityadjustment 45 velocityunit kph
```

この例では、サーバー上に設定されている **US\_NE** データベースリソースから、ポイントの速度を **60%** だけ減少させることによってオーバーライドします。

```
erm createpointupdate --datasource US_NE --point
"-72,40,epsg:4326" --velocitypercentage -60
```

## erm resetpointupdate

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に](#) (88ページ)」を参照してください。

erm resetpointupdate コマンドは、データの元の状態に対するオーバーライドがあれば、それを返します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

### 使用方法

```
erm resetpointupdate --datasource db_resource --point "x,y,coordsys" --resettype reset_type
```

**注：**パラメータのリストを表示するには、“help erm resetpointupdate”と入力します。

必須	引数	説明
はい	--datasource <i>db_resource</i>	オーバーライドを持つデータベースリソースの名前を指定します。既存のルーティングデータベ

必須	引数	説明
		スリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
はい	<code>--point "x,y,coordsys"</code>	既存のオーバーライドが位置するポイントを指定します。ポイントは、 <code>"x,y,coordsys"</code> という形式で指定します。ここで、 <code>coordsys</code> はポイントの座標系です。
はい	<code>--resettype reset_type</code>	削除 (取り消し) するオーバーライドのタイプです。 <ul style="list-style-type: none"> <li><b>speed</b>            速度更新を削除します。</li> <li><b>exclude</b>        除外更新を削除します。</li> </ul>

**例**

この例では、指定されたポイントの既存の除外オーバーライドを、サーバー上に設定されている `US_NE` データベース リソースからリセットします。

```
erm resetpointupdate --datasource US_NE --point
"-72,40,epsg:4326" --resettype exclude
```

## erm createsegmentupdate

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

`erm createsegmentupdate` コマンドは、指定されたセグメントのルーティング データをオーバーライドします。このコマンドにより、速度を設定または変更したり、ルート of セクションを除外したり、道路タイプを変更したりできます。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

**注：**永続更新タイプは特定のデータ リソースに対してのみ有効で、データ更新後には有効でなくなる場合があります。

### 使用方法

```
erm createsegmentupdate --datasource db_resource --segmentid "segment_id"
--exclude --velocity velocity_value --velocityunit velocity_unit --velocityadjustment
velocity_adjustment_value --velocitypercentage velocity_percentage_value --roadtype
road_type
```

注：パラメータのリストを表示するには、“help erm createsegmentupdate” と入力します。

必須	引数	説明
はい	--datasource <i>db_resource</i>	データをオーバーライドするデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
はい	--segmentid " <i>segment_id</i> "	オーバーライドするセグメントを指定します。セグメントは、データ内で指定されている形式で指定します。例: " <b>7e3396fc:6e5251</b> "。
いいえ	--exclude	指定されたセグメントをすべてのルート計算から除外します。コマンドにこのパラメータを含めることにより、そのセグメントを除外するかどうかを指定します。
いいえ	--velocity <i>velocity_value</i>	速度更新を定義します。新しい速度を指定することにより、セグメントの新しい速度を指定します。velocityunit パラメータを指定しない限り、デフォルトの単位は mph (マイル/時) です。
いいえ	--velocityunit <i>velocity_unit</i>	velocity または velocityadjustment オーバーライドの速度単位を定義します。デフォルト値は mph (マイル/時) です。速度更新の場合、速度単位には kph (キロメートル/時)、mps (メートル/秒)、mph (マイル/時) のいずれかの値を指定できます。
いいえ	--velocityadjustment <i>velocity_adjustment_value</i>	速度更新を定義します。速度(単位と値)の変更を指定することにより、セグメントの速度の変化を定義します。速度値は増加(正の値)または減少(負の値)させることができます。velocityunit パラメータを指定しない限り、デフォルトの単位は mph (マイル/時) です。
いいえ	--velocitypercentage <i>velocity_percentage_value</i>	速度更新を定義します。速度を増加(正の値)または減少(負の値)させる割合を指定することにより、セグメントの速度の増加を定義します。
いいえ	--roadtype <i>road_type</i>	セグメントの新しい道路タイプを定義します。

**例**

この例では、サーバー上に設定されている **US\_NE** データベースリソースから、セグメントの速度を **15 mph** にオーバーライドします。

```
erm createsegmentupdate --datasource US_NE --segmentid
"7e3396fc:6e5251" --velocity 15 --velocityunit mph
```

この例では、サーバー上に設定されている **US\_NE** データベースリソースから、指定されたセグメントを除外します。

```
erm createsegmentupdate --datasource US_NE --segmentid
"7e3396fc:6e5251" --exclude
```

この例では、サーバー上に設定されている **US\_NE** データベースリソースから、セグメントの速度を **45 kph** だけ増加させることによってオーバーライドします。

```
erm createsegmentupdate --datasource US_NE --segmentid
"7e3396fc:6e5251" --velocityadjustment 45 velocityunit kph
```

この例では、サーバー上に設定されている **US\_NE** データベースリソースから、セグメントの速度を **60%** だけ減少させることによってオーバーライドします。

```
erm createsegmentupdate --datasource US_NE --segmentid
"7e3396fc:6e5251" --velocitypercentage -60
```

この例では、サーバー上に設定されている **US\_NE** データベースリソースから、セグメントの道路タイプをフェリーにオーバーライドします。

```
erm createsegmentupdate --datasource US_NE --segmentid
"7e3396fc:6e5251" --roadtype ferry
```

## erm resetsegmentupdate

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

erm resetsegmentupdate コマンドは、データの元の状態に対するオーバーライドがあれば、それを返します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

### 使用方法

```
erm resetsegmentupdate --datasource db_resource --segmentid "segment_id"
--resettype reset_type
```

**注：**パラメータのリストを表示するには、“help erm resetsegmentupdate” と入力します。

必須	引数	説明
はい	<code>--datasource <i>db_resource</i></code>	オーバーライドを持つデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
はい	<code>--segment "<i>segment_id</i>"</code>	既存のオーバーライドが位置するセグメントを指定します。セグメントは、データ内で指定されている形式で指定します。例: <code>"7e3396fc:6e5251"</code> 。
はい	<code>--resettype <i>reset_type</i></code>	削除 (取り消し) するオーバーライドのタイプです。 <ul style="list-style-type: none"> <li><b>speed</b>            速度更新を削除します。</li> <li><b>exclude</b>        除外更新を削除します。</li> <li><b>roadType</b>       道路タイプ更新を削除します。</li> </ul>

**例**

この例では、指定されたセグメントの既存の道路タイプ オーバーライドを、サーバー上に設定されている `US_NE` データベース リソースからリセットします。

```
erm resetsegmentupdate --datasource US_NE --point
"7e3396fc:6e5251" --resettype roadtype
```

## erm getsegmentupdates

**注:** 管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

`erm getsegmentupdates` コマンドは、指定されたセグメントのルーティング データにおけるオーバーライドのリストを返します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

**注:** `segmentids` はオプションのパラメータです。セグメント ID を指定しない場合は、使用可能なすべてのセグメントに対するオーバーライドが返されます。

### 使用方法

```
erm getsegmentupdates --datasource db_resource --segmentids "segment_ids"
--velocityunit velocityunit
```

注：パラメータのリストを表示するには、“help erm getsegmentupdates”と入力します。

必須	引数	説明
はい	--datasource <i>db_resource</i>	オーバーライドを持つデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
いいえ	--segmentids " <i>segment_ids</i> "	オーバーライド情報を返すセグメント ID のカンマ区切りリスト。セグメントは、データ内で指定されている形式で指定します。例: " <b>7e3396fc:6e5251</b> "。
いいえ	--velocityunit <i>velocityunit</i>	レスポンスに表示される速度単位を指定します (mph - マイル/時、kph - キロメートル/時、mtps - メートル/秒、mtpm - メートル/分)。デフォルトは mph です。

#### 例

この例では、サーバー上に設定されている `US_NE` データベースリソースから、セグメントのオーバーライドを返します。

```
erm getsegmentupdates --datasource US_NE --segmentids
"7e3396fc:6e5251" --velocityunit kph
```

## erm createroadtypeupdate

注：管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

`erm createroadtypeupdate` コマンドは、指定された道路タイプのルーティングデータをオーバーライドします。このコマンドにより、特定の道路タイプのルートのを速度を設定または変更できます。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

注：永続更新タイプは特定のデータ リソースに対してのみ有効で、データ更新後には有効でなくなる場合があります。



## 使用方法

```
erm createroadtypeupdate --datasource db_resource --roadtype "road_type"
--velocity velocity_value --velocityunit velocity_unit --velocityadjustment
velocity_adjustment_value --velocitypercentage velocity_percentage_value --roadtype
road_type
```

注：パラメータのリストを表示するには、“help erm createroadtypeupdate”と入力します。

必須	引数	説明
はい	--datasource <i>db_resource</i>	データをオーバーライドするデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、ermdb list コマンドを使用します。
はい	--roadtype " <i>road_type</i> "	オーバーロードする道路タイプ。道路タイプには、次の種類があります。 <ul style="list-style-type: none"> <li>• 進入路</li> <li>• 裏道路</li> <li>• 接続道路</li> <li>• フェリー</li> <li>• 小道</li> <li>• 出入制限道路 (密集都市部)</li> <li>• 出入制限道路 (農村部)</li> <li>• 出入制限道路 (郊外)</li> <li>• 出入制限道路 (都市部)</li> <li>• 地方道路 (密集都市部)</li> <li>• 地方道路 (農村部)</li> <li>• 地方道路 (郊外)</li> <li>• 地方道路 (都市部)</li> <li>• 主要地方道路 (密集都市部)</li> <li>• 主要地方道路 (農村部)</li> <li>• 主要地方道路 (郊外)</li> <li>• 主要地方道路 (都市部)</li> <li>• 幹線道路 (密集都市部)</li> <li>• 幹線道路 (農村部)</li> <li>• 幹線道路 (郊外)</li> <li>• 幹線道路 (都市部)</li> <li>• 補助地方道路 (密集都市部)</li> </ul>

必須	引数	説明
		<ul style="list-style-type: none"> <li>• 補助地方道路 (農村部)</li> <li>• 補助地方道路 (郊外)</li> <li>• 補助地方道路 (都市部)</li> <li>• 一般道路 (密集都市部)</li> <li>• 一般道路 (農村部)</li> <li>• 一般道路 (農村部)</li> <li>• 一般道路 (都市部)</li> <li>• 主要高速道路 (密集都市部)</li> <li>• 主要高速道路 (農村部)</li> <li>• 主要高速道路 (郊外)</li> <li>• 主要高速道路 (都市部)</li> <li>• 出入路 (密集都市部)</li> <li>• 出入路 (出入制限道路)</li> <li>• 出入路 (幹線道路)</li> <li>• 出入路 (主要高速道路)</li> <li>• 出入路 (農村部)</li> <li>• 出入路 (一般高速道路)</li> <li>• 出入路 (都市部)</li> <li>• 出入路 (郊外)</li> <li>• 一般高速道路 (密集都市部)</li> <li>• 一般高速道路 (農村部)</li> <li>• 一般高速道路 (郊外)</li> <li>• 一般高速道路 (都市部)</li> </ul>
いいえ	<code>--velocity <i>velocity_value</i></code>	速度更新を定義します。新しい速度を指定することにより、道路タイプの新しい速度を指定します。velocityunit パラメータを指定しない限り、デフォルトの単位は mph (マイル/時) です。
いいえ	<code>--velocityunit <i>velocity_unit</i></code>	velocity または velocityadjustment オーバーライドの速度単位を定義します。デフォルト値は mph (マイル/時) です。速度更新の場合、速度単位には kph (キロメートル/時)、mps (メートル/秒)、mph (マイル/時) のいずれかの値を指定できます。
いいえ	<code>--velocityadjustment <i>velocity_adjustment_value</i></code>	速度更新を定義します。速度 (単位と値) の変更を指定することにより、道路タイプの速度の変化を

必須	引数	説明
		定義します。速度値は増加 (正の値) または減少 (負の値) させることができます。velocityunit パラメータを指定しない限り、デフォルトの単位は mph (マイル/時) です。
いいえ	<code>--velocitypercentage</code> <code>velocity_percentage_value</code>	速度更新を定義します。速度を増加 (正の値) または減少 (負の値) させる割合を指定することにより、道路タイプの速度の増加を定義します。

**例**

この例では、サーバー上に設定されている US\_NE データベースリソースから、指定された道路タイプの速度を 25 mph にオーバーライドします。

```
erm createroadtypeupdate --datasource US_NE --roadtype "normal road suburban" --velocity 25 --velocityunit kph
```

この例では、サーバー上に設定されている US\_NE データベースリソースから、指定された道路タイプの速度を 50 kph だけ増加します。

```
erm createroadtypeupdate --datasource US_NE --roadtype "normal road suburban" --velocityadjustment 50 --velocityunit mph
```

この例では、サーバー上に設定されている US\_NE データベースリソースから、指定された道路タイプの速度を 65% だけ減少させることによってオーバーライドします。

```
erm createroadtypeupdate --datasource US_NE --roadtype "normal road suburban" --velocitypercentage -65
```

## erm resetroadtypeupdate

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

erm resetroadtypeupdate コマンドは、データの元の状態に対するオーバーライドがあれば、それを返します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

### 使用方法

```
erm resetroadtypeupdate --datasource db_resource --roadtype "road_type"
```

注：パラメータのリストを表示するには、“help erm resetroadtypeupdate”と入力します。

必須	引数	説明
はい	<code>--datasource <i>db_resource</i></code>	オーバーライドを持つデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
はい	<code>--roadtype "<i>road_type</i>"</code>	既存のオーバーライドを持つ道路タイプを指定します。道路タイプの一覧については、「 <a href="#">erm createroadtypeupdate (112ページ)</a> 」を参照してください。

#### 例

この例では、“normal road suburban” (一般道路 (郊外)) の道路タイプ オーバーライドを、サーバー上に設定されている `US_NE` データベースリソースからリセットします。

```
erm resetroadtypeupdate --datasource US_NE --roadtype "normal road suburban"
```

## erm getroadtypeupdates

注：管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

`erm getroadtypeupdates` コマンドは、指定された道路タイプのルーティング データにおけるオーバーライドのリストを返します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

注：`roadtypes` はオプションのパラメータです。道路タイプを指定しない場合は、使用可能なすべての道路タイプに対するオーバーライドが返されます。

#### 使用方法

```
erm getroadtypeupdates --datasource db_resource --roadtypes "road_types"
--velocityunit velocityunit
```

注：パラメータのリストを表示するには、“help erm getroadtypeupdates”と入力します。

必須	引数	説明
はい	<code>--datasource <i>db_resource</i></code>	オーバーライドを持つデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
いいえ	<code>--roadtypes "<i>road_types</i>"</code>	オーバーライド情報を返す道路タイプのカンマ区切りリスト。道路タイプの一覧については、「 <a href="#">erm createroadtypeupdate (112ページ)</a> 」を参照してください。
いいえ	<code>--velocityunit <i>velocityunit</i></code>	レスポンスに表示される速度単位を指定します (mph - マイル/時、kph - キロメートル/時、mtps - メートル/秒、mtpm - メートル/分)。デフォルトは mph です。

**例**

この例では、"normal road urban" (一般道路 (都市部)) の道路タイプのオーバーライドを、サーバー上に設定されている US\_NE データベースリソースから返します。

```
erm getroadtypeupdates --datasource US_NE --roadtypes "normal
road urban" --velocityunit kph
```

## erm getallupdates

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

`erm getallupdates` コマンドは、指定されたルーティング データベース リソースに対するすべてのオーバーライドのリストを返します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

### 使用方法

```
erm getallupdates --datasource db_resource "segment_ids" --velocityunit velocityunit
```

**注：**パラメータのリストを表示するには、「`help erm getallupdates`」と入力します。

必須	引数	説明
はい	<code>--datasource <i>db_resource</i></code>	オーバーライドを持つデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。
いいえ	<code>--velocityunit <i>velocityunit</i></code>	レスポンスに表示される速度単位を指定します (mph - マイル/時、kph - キロメートル/時、mtps - メートル/秒、mtpm - メートル/分)。デフォルトは mph です。

**例**

この例では、サーバー上に設定されている `US_NE` データベースリソースからのすべてのオーバーライドを返します。

```
erm getallupdates --datasource US_NE --velocityunit kph
```

## erm resetallupdates

**注：**管理ユーティリティをインストールして実行する手順については、「[管理ユーティリティを使用する前に \(88ページ\)](#)」を参照してください。

`erm resetallupdates` コマンドは、データの元の状態に対するすべてのオーバーライドを返します。このコマンドを使うには、Enterprise Routing モジュールがインストールされている必要があります。

### 使用方法

```
erm resetallupdates --datasource db_resource
```

**注：**パラメータのリストを表示するには、“`help erm resetallupdates`” と入力します。

必須	引数	説明
はい	<code>--datasource <i>db_resource</i></code>	オーバーライドを持つデータベースリソースの名前を指定します。既存のルーティングデータベースリソースの一覧を表示するには、 <code>ermdb list</code> コマンドを使用します。

**例**

この例では、サーバー上に設定されている US\_NE データベースリソースからのすべてのオーバーライドをリセットします。

```
erm resetallupdates --datasource US_NE
```



# 8 - Enterprise Routing モジュール

## このセクションの構成

---

デフォルト サービス/ステージ オプションの指定	121
サービス/ステージのプレビュー	121
Management Console を用いたルート データの取得	124

## デフォルト サービス/ステージ オプションの指定

デフォルト オプションは、システム上の各サービスまたはステージのデフォルトの動作を制御します。各オプションに対して、デフォルト値を指定できます。デフォルト オプションは、要求においてオプションの値が明示的に定義されていない場合に適用されます。また、デフォルト オプションは、Enterprise Designer でこのサービスを使用してデータフローを作成する場合に、デフォルトとして使用される設定でもあります。

オプションの詳細については、『*Spectrum Spatial* ガイド』の Enterprise Routing モジュールに該当する「ステージ」セクションと「リソースおよびデータ」セクションを参照してください。

**注：** 永続更新は、Management Console では管理されません。永続更新を行うには、管理ユーティリティの Spectrum コマンドライン機能を使用します。

**注：** Management Console の Get Route Data サービスでは、デフォルト オプションは設定されません。このサービスはインタラクティブに動作して、セグメントに対するルーティング データを返します。Get Route Data の詳細については、「[Management Console を用いたルート データの取得 \(124ページ\)](#)」を参照してください。

1. Management Console を開きます。
2. [サービス] をクリックします。
3. 対象のモジュール (Enterprise Routing モジュール) をクリックします。
4. 左側のリストで設定するサービスをクリックします。
5. サービスのオプションを設定します。ほとんどのサービスにさまざまな種類のオプションがあり、それらが複数のタブに表示されます。
6. [保存] をクリックします。

## サービス/ステージのプレビュー

Management Console でサービスの [プレビュー] タブを使うと、サービスの結果をプレビューできます。プレビューは、どのオプションを指定するかを決める際に便利です。異なるオプションが、サービスまたはステージから返されるデータに与える影響を直ちに確認できるためです。

1. Management Console を開きます。
2. [サービス] メニューを開き、プレビューするサービスを選択します。

3. **[プレビュー]** タブをクリックします。
4. テスト用のデータを各フィールドに入力します。


プレビューを使用する際には、次の点に注意してください。



- すべてのフィールドにデータを入力する必要はありません。フィールドを空のままにすると、空の文字列がプレビューに使用されます。
- フィールドに null 値を引き渡した場合の影響をプレビューするには、そのフィールドの横の無効アイコンをクリックします。

#### 入力レコード



#### ▼ 入力レコード 1

AddressLine1   
AddressLine2

- 複数のレコードを同時にプレビューできます。レコードを追加するには、追加ボタン  をクリックします。
- テスト用のデータはファイルからインポートできます。データをインポートするには、インポート ボタン  をクリックします。**[ファイル名]** と **[フィールド区切り文字]** を選択します。次のことに注意してください。
  - ファイルの最初の行は、ヘッダーレコードでなければなりません。ヘッダー内のフィールド名は、サービスで必要とされるフィールド名と一致する必要があります。
  - インポートできるレコードは、最大 5 個です。
  - フィールドの区切り文字がスペースの場合、フィールド値を引用符で囲む必要があります。スペースをフィールドの区切り文字に使うファイルの例を以下に示します。

```
AddressLine1 AddressLine2 City StateProvince PostalCode
"One Global View" "" "Troy" "NY" "12180"
"3001 Summer St" "" "Stamford" "CT" "06926"
"224 N Michigan Ave" "Suite 300" "Chicago" "IL" ""
```

- すべてのレコードを削除するには、プレビュー エリアの上部にある削除ボタンをクリックします。



- 特定のレコードを削除するには、入力レコード名 ("Input Record 1" など) の上にカーソルを合わせてレコード名の横の削除ボタンをクリックします。



- サービスが階層化されたデータを入力として受け取る場合
  - 子レコードを追加するには、親レコードの上にカーソルを合わせて追加ボタンをクリックします。
  - 親からすべての子レコードを削除するには、親レコードの上にカーソルを合わせて削除ボタンをクリックします。
  - 特定の子レコードを削除するには、そのレコードの上にカーソルを合わせて削除ボタンをクリックします。

5. **[プレビューを実行]** をクリックします。

サービスは入力レコードを処理し、結果を表示します。

6. 出力データをプレビューして、サービスやステージから適切な結果が得られることを確認します。必要に応じて、オプションに変更を加えて、**[プレビューを実行]** を再度クリックすることができます(データを再度入力する必要はありません)。

## Management Console を用いたルート データの取得

Management Console を使用して、最も近いポイントまたはセグメント ID からのセグメント情報をプレビューおよび保存することができます。GetRouteData サービスは、ポイントまたはセグメント ID に対するセグメント情報を返します。ポイントを指定すると、最も近いルート セグメントが返されます。セグメント ID を指定すると、指定されたルート セグメントのルート データが返されます。

ルート データをプレビューまたは保存するには

1. Management Console を開きます。
2. **[サービス]** メニューから、Enterprise Routing モジュールを選択します。
3. サービス リストから **[Get Route Data]** を選択します。
4. **[入力タイプ]** フィールドで、**[ポイント データ]** または **[セグメント データ]** を選択します。
5. **[データベース]** フィールドから、ルーティング データベース リソースを選択します。

新しいルーティング データベース リソースを追加する必要がある場合は、「[ルーティング データベース リソースの追加](#)」を参照してください。

6. 選択した入力タイプに必要な情報を入力します。

ポイント データを選択した場合は、ポイント座標と座標系を入力します。セグメント データを選択した場合は、セグメント ID を入力します。

7. **[プレビュー]** をクリックします。

**[出力データ]** セクションに、ルート セグメント データが返されます。入力に関連するセグメントが複数存在する場合は、複数のセグメントが **[セグメント詳細 1]**、**[セグメント詳細 2]** などのように表示されます。

8. **[保存]** ボタンをクリックしてルーティング データ結果をテキスト ファイルとして保存するか、**[クリア]** ボタンをクリックして **[出力データ]** セクションから結果を削除します。

# 9 - システムのトラブルシューティング

## このセクションの構成

---

破損したりポジトリ インデックスの再構築	126
反応がなくなったサーバーのメモリ使用量の監視	126

## 破損したりポジトリ インデックスの再構築

手動操作または停電によって、サーバーが突然シャットダウンした場合や Java プロセスが停止した場合には、リポジトリが破損することがあります。この場合、以前は検索可能だったリソースを取得できなくなることがありますが、ログにエラーや警告は記録されません。権限の変更が原因でないことを確認したら、次の手順でインデックスを再構築してこの問題を修正します。

1. サーバーをシャットダウンします。
2. 次の場所のインデックス ディレクトリを削除します。
  - <Spectrum>\server\modules\spatial\jackrabbit\workspaces\default
  - <Spectrum>\server\modules\spatial\jackrabbit\workspaces\security
  - <Spectrum>\server\modules\spatial\jackrabbit\repository
3. サーバーを再起動します。  
起動時に Jackrabbit が上記の場所にインデックスを再作成します。

インデックスを再構築した後は、検索は正常な動作に戻ります。

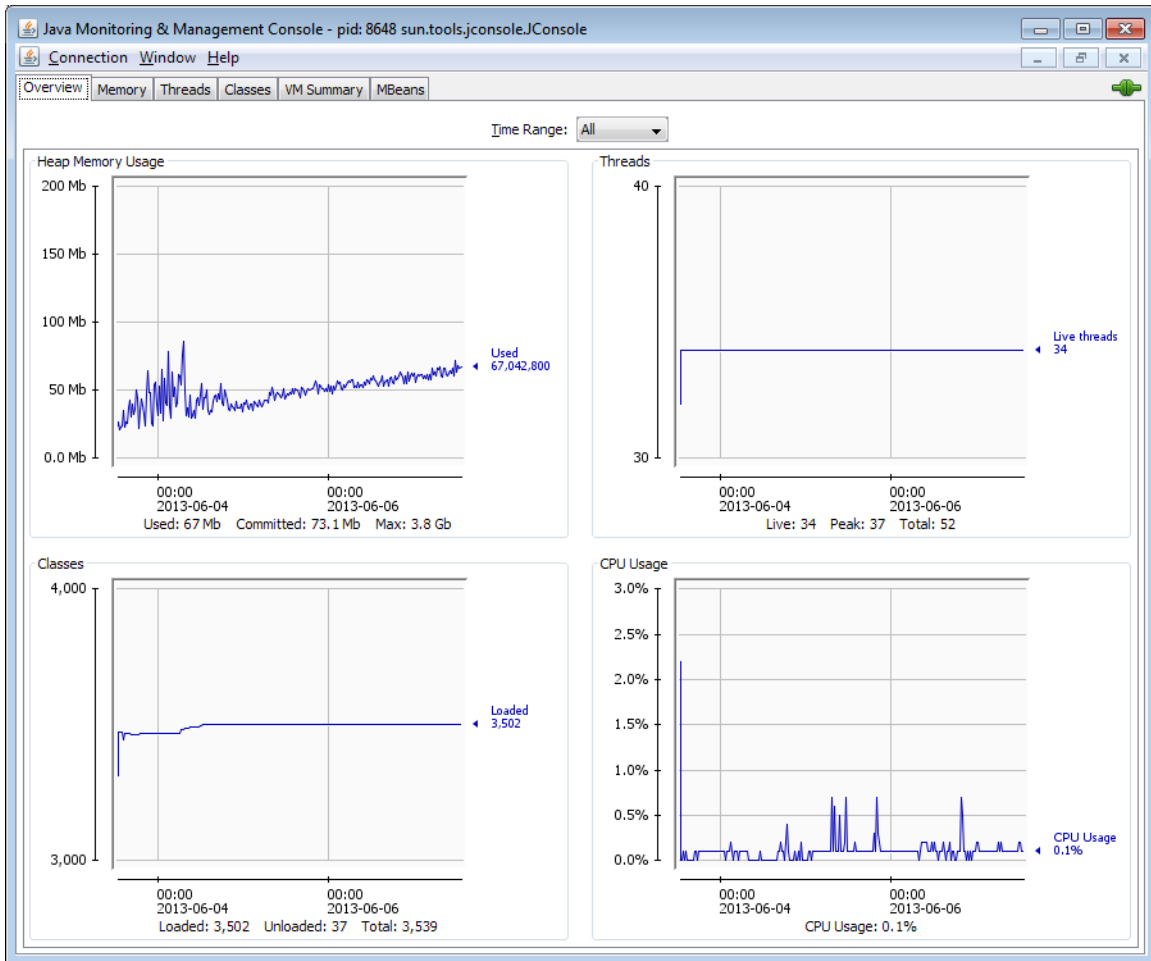
## 反応がなくなったサーバーのメモリ使用量の監視

Spectrum サーバーの反応が停止した場合は、次の手順でパフォーマンスとリソース消費を監視できます。この監視で得られる情報を使用して、メモリとスレッドの使用を調整できます。

1. Mapping Service 以外のサービスが動作しているかどうか確認します。例えば、デモ ページ `http://<サーバー名>:<ポート>/Spatial/FeatureService//DemoPage.html` で Feature Service を開始します。これにより、サーバー全体が停止しているのか、Mapping Service だけが停止しているのかを判断できます。
2. 次の構成ファイルに基づいて、Mapping と Map Tiling の両方のイメージを格納する十分なディスク領域があることを確認します。
  - **Mapping:** `http://<サーバー>:<ポート>/RepositoryService/repository/default/Configuration/MappingConfiguration` の "`<Directory> C:\Program Files\Pitney Bowes\Spectrum/server/modules/spatial/images </Directory>`"
  - **Map Tiling:** "`<Property name="diskPath" value="C:/Program Files/Pitney Bowes/Spectrum/server/modules/spatial/TileCache"/>`" の "`http://<server>:<port>/RepositoryService/repository/default/Configuration/MapTilingConfiguration`"



3. Spectrum サーバーを停止します。
4. テキスト エディタで、**java.vargs** ファイルを <Installed>\Pitney Bowes\Spectrum\server\modules\spatial\java.vargs から開きます。
5. **vargs** の値を変更します。デフォルトは **2GB (2048MB)** です。例えば、リモート コンポーネントのメモリを **4GB** に増やすには、**vargs** をデフォルトの **-Xmx2048m** から **-Xmx4096m** に変更します。お使いのオペレーティング システムで使用可能な最大メモリ容量を超えないようにし、またオペレーティング システム自体の処理のための適切な容量を残すようにしてください。
6. **java.vargs** ファイルを保存します。
7. 次の手順でサーバー ラッパーを開始します。
  - a) 管理者としてコマンド プロンプトを開きます。
  - b) <Installed>\Pitney Bowes\Spectrum\server\bin\wrapper ディレクトリに移動し、「**wrapper.exe -c**」と入力します。Spectrum サーバーが数分で起動します。
8. サーバーが起動したら、デモ ページから次の手順で要求を実行します。
  - a) **http://<サーバー名>:<ポート>/Spatial/MappingService/DemoPage.html** を開き、名前付き マップ リストのリクエストを実行します。
  - b) **http://<サーバー名>:<ポート>/Spatial/FeatureService/DemoPage.html** を開き、テーブル名 リストのリクエストを実行します。
9. <インストール先>\Pitney Bowes\Spectrum\java64\bin に移動し、**jconsole.exe** を実行します。
10. ローカル プロセスの中からラッパー プロセスを選択します。
11. Jconsole で、新しいセッションを追加し、**Feature Service** プロセスを選択します。
12. Jconsole で、新しいセッションを追加し、**Mapping Service** プロセスを選択します。
13. Jconsole を実行したままにして、Spectrum プラットフォーム ラッパーの **Feature Service** と **Mapping Service** のメモリ、CPU、スレッドなどを監視します。



# 著作権に関する通知

© 2017 Pitney Bowes Software Inc. All rights reserved. MapInfo および Group 1 Software は Pitney Bowes Software Inc. の商標です。その他のマークおよび商標はすべて、それぞれの所有者の資産です。

### USPS® 情報

Pitney Bowes Inc. は、ZIP + 4® データベースを光学および磁気媒体に発行および販売する非独占的ライセンスを所有しています。CASS、CASS 認定、DPV、eLOT、FASTforward、First-Class Mail、Intelligent Mail、LACS<sup>Link</sup>、NCOA<sup>Link</sup>、PAVE、PLANET Code、Postal Service、POSTNET、Post Office、RDI、Suite<sup>Link</sup>、United States Postal Service、Standard Mail、United States Post Office、USPS、ZIP Code、および ZIP + 4 の各商標は United States Postal Service が所有します。United States Postal Service に帰属する商標はこれに限りません。

Pitney Bowes Inc. は、NCOA<sup>Link</sup>® 処理に対する USPS® の非独占的ライセンスを所有しています。

Pitney Bowes Software の製品、オプション、およびサービスの価格は、USPS® または米国政府によって規定、制御、または承認されるものではありません。RDI™ データを利用して郵便送料を判定する場合に、使用する郵便配送業者の選定に関するビジネス上の意思決定が USPS® または米国政府によって行われることはありません。

### データ プロバイダおよび関連情報

このメディアに含まれて、Pitney Bowes Software アプリケーション内で使用されるデータ製品は、各種商標によって、および次の 1 つ以上の著作権によって保護されています。

© Copyright United States Postal Service. All rights reserved.

© 2014 TomTom. All rights reserved. TomTom および TomTom ロゴは TomTom N.V. の登録商標です。

© 2016 HERE

Fuente: INEGI (Instituto Nacional de Estadística y Geografía)

電子データに基づいています。© National Land Survey Sweden.

© Copyright United States Census Bureau

© Copyright Nova Marketing Group, Inc.

このプログラムの一部は著作権で保護されています。© Copyright 1993-2007 by Nova Marketing Group Inc. All Rights Reserved

© Copyright Second Decimal, LLC

© Copyright Canada Post Corporation

この CD-ROM には、Canada Post Corporation が著作権を所有している編集物からのデータが収録されています。

© 2007 Claritas, Inc.

Geocode Address World データ セットには、  
<http://creativecommons.org/licenses/by/3.0/legalcode> に存在するクリエイティブ コモンズ アトリビューション ライセンス (「アトリビューション ライセンス」) の下に提供されている GeoNames Project ([www.geonames.org](http://www.geonames.org)) からライセンス供与されたデータが含まれています。お客様による GeoNames データ (Spectrum™ Technology Platform ユーザ マニュアルに記載) の使用は、アトリビューション ライセンスの条件に従う必要があります。お客様と Pitney Bowes Software, Inc. との契約と、アトリビューション ライセンスの間に矛盾が生じる場合は、アトリビューション ライセンスのみに基づいてそれを解決する必要があります。お客様による GeoNames データの使用に関しては、アトリビューション ライセンスが適用されるためです。



3001 Summer Street  
Stamford CT 06926-0700  
USA

[www.pitneybowes.com](http://www.pitneybowes.com)