

# HP Service Manager Software

Windows®およびUnix®オペレーティングシステム向け

ソフトウェアバージョン : 9.20

---

HP Universal CMDB-HP Service Manager インテグレーションガイド

ドキュメントリリース日 : 2010年6月

ソフトウェアリリース日 : 2010年6月



## ご利用条件

### 保証

HP 製品およびサービスの保証は、各製品およびサービスに添付された保証書に記載の明示保証のみとなります。本書のいかなる内容も当該保証に新たに保証を追加するものではありません。HP は、本書中の技術的あるいは校正上の誤り、省略に対して責任を負いかねます。

本文書に記載の事項は、予告なく変更されることがあります。

### 限定保証条項

機密コンピュータソフトウェア。所有、使用、コピーには、HP による有効なライセンスが必要です。FAR 12.211 および 12.212 準拠。商用コンピュータソフトウェア、コンピュータソフトウェアマニュアル、技術データは、ベンダの標準商用ライセンスに基づき、米国政府にライセンス供与されています。

### 著作権

© Copyright 1994-2010 Hewlett-Packard Development Company, L.P.

本製品には、Eric Young (eay@cryptsoft.com) が作成した暗号ソフトウェアが含まれています。本製品には、Tim Hudson (tjh@cryptsoft.com) が作成したソフトウェアが含まれています。Smack ソフトウェア copyright© Jive Software, 1998-2004。SVG Viewer、Mozilla JavaScript-C (SpiderMonkey)、および Rhino ソフトウェア Copyright©1998-2004 The Mozilla Organization。本製品には、OpenSSL ツールキットで使用するため OpenSSL Project が開発したソフトウェアが含まれています。

([http:// www.openssl.org](http://www.openssl.org))。OpenSSL ソフトウェア copyright 1998-2005 The OpenSSL Project. All rights reserved. 本製品には、MX4J project (<http://mx4j.sourceforge.net>) が開発したソフトウェアが含まれています。MX4J ソフトウェア copyright©2001-2004 MX4J Team. All rights reserved. JFreeChart ソフトウェア© 2000-2004、Object Refinery Limited. All rights reserved. JDOM ソフトウェア copyright©2000 Brett McLaughlin, Jason Hunter. All rights reserved. LDAP、OpenLDAP、Netscape Directory SDK Copyright© 1995-2004 Sun Microsystems, Inc.。Japanese Morphological Analyzer© 2004 Basis Technology Corp.。Sentry Spelling-Checker Engine Copyright© 2000 Wintertree Software Inc.。Spell Checker copyright© 1995-2004 Wintertree Software Inc.。CoolMenu ソフトウェア copyright© 2001 Thomas Brattli. All rights reserved. Coroutine Software for Java は Neva Object Technology, Inc. が所有しており、米国および国際著作権法で保護されています。Crystal Reports Pro および Crystal RTE ソフトウェア© 2001 Crystal Decisions, Inc., All rights reserved. Eclipse ソフトウェア© Copyright 2000, 2004 IBM Corporation およびその他。All rights reserved. Copyright 2001-2004 Kiran Kaja and Robert A. van Engelen, Genivia Inc. All rights reserved. Xtree copyright 2004 Emil A. Eklund。本製品には、インディアナ大学 Extreme! Lab (<[http:// www.extreme.indiana.edu/](http://www.extreme.indiana.edu/)>) が開発したソフトウェアが含まれています。Portions copyright Daniel G. Hyans, 1998. cbg.editor Eclipse plug-in copyright 2002, Chris Grindstaff. 本製品に組み込まれたソフトウェアの一部は gSOAP ソフトウェアです。gSOAP が作成した部分は、copyright© 2001-2004 Robert A. van Engelen, Genivia Inc です。All Rights Reserved. Copyright© 1991-2005 Unicode, Inc. All rights reserved. [http:// www.unicode.org/copyright.html](http://www.unicode.org/copyright.html) の利用規約の下で配布されます。

### 商標

Java™ およびすべての Java 関連の商標およびロゴは、米国およびその他の国における Sun Microsystems, Inc. の商標または登録商標です。

Microsoft® および Windows® は、Microsoft Corporation の米国における登録商標です。

Oracle® は、Oracle Corporation (カリフォルニア州レッドウッド市) の米国における登録商標です。

UNIX® は、The Open Group の登録商標です。

## ドキュメントの更新

本ドキュメントのタイトルページには、次の識別情報が含まれます。

- ソフトウェアバージョン番号。ソフトウェアバージョンを示します。
- ドキュメントのリリース日。ドキュメントが更新されるたびに変更されます。
- ソフトウェアのリリース日。このバージョンのソフトウェアのリリース日を示します。

最新の更新について確認する場合や、最新版ドキュメントを使用していることを確認する場合は、次のサイトを参照してください。

**<http://h20230.www2.hp.com/selfsolve/manuals>**

このサイトを利用するには、HP Passportに登録した上でサインインする必要があります。HP Passport IDは、次のURLでご登録いただけます。

**<http://h20229.www2.hp.com/passport-registration.html>**

または、[HP Passport login (HP Passport ログイン)] ページの [New user registration (新規ユーザの登録)] リンクをクリックしてください。

該当製品のサポートサービスを受けている場合は、更新版または最新版ドキュメントも入手することができます。詳細については、お近くのHP営業窓口までご連絡ください。

## サポート

次のHP Software サポート Webサイトを参照してください。

**[www.hp.com/go/hpssoftwaresupport](http://www.hp.com/go/hpssoftwaresupport)**

このWebサイトには、連絡先の情報をはじめ、HP Softwareの提供する製品、サービス、サポートに関する詳細な情報が記載されています。

HP Software オンラインサポートには、お客様によるセルフソルブ機能があります。業務の管理に必要な対話型の技術支援ツールにすばやく効率的にアクセスすることができます。サポートカスタマのお客様は、サポート Webサイトで次のサービスをご利用いただけます。

- 必要なナレッジドキュメントの検索
- サポートケースやエンハンスメント要求の提出およびトラッキング
- ソフトウェアパッチのダウンロード
- サポート契約の管理
- HPサポート連絡先の検索
- 利用可能なサービスに関する情報の確認
- 他のソフトウェアカスタマとのディスカッションへの参加
- ソフトウェアトレーニングの調査および登録

ほとんどの場合、サポートを受けるには、HP Passport ユーザとして登録した上でサインインする必要があります。また多くの場合、サポート契約も必要となります。HP Passport IDは、次のURLでご登録いただけます。

**<http://h20229.www2.hp.com/passport-registration.html>**

アクセスレベルの詳細については、次のサイトを参照してください。

**[http://h20230.www2.hp.com/new\\_access\\_levels.jsp](http://h20230.www2.hp.com/new_access_levels.jsp)**

# 目次

|   |                                       |    |
|---|---------------------------------------|----|
| 1 | はじめに                                  | 9  |
|   | インテグレーションの目的                          | 10 |
|   | マルチテナントの要件                            | 11 |
|   | UCMDBに格納されるマルチテナント情報について              | 12 |
|   | Service Managerに格納されるマルチテナント情報について    | 12 |
|   | 一意な論理名                                | 12 |
|   | 会社レコードの同期                             | 13 |
|   | UCMDB顧客ID                             | 15 |
|   | UCMDBユーザIDとパスワード                      | 15 |
|   | 会社コード                                 | 15 |
|   | CIレコードと関係に複製される会社情報                   | 15 |
|   | インシデントレコードに複製される会社情報                  | 15 |
|   | スケジュールレコード                            | 16 |
| 2 | インテグレーションのセットアップ                      | 17 |
|   | インテグレーションの要件                          | 18 |
|   | インテグレーションポイントの作成                      | 19 |
|   | 各テナントの独立Data Flow Probeのインストール        | 20 |
|   | テナント固有Data Flow Probeの起動              | 21 |
|   | テナント固有Data Flow ProbeのIPアドレス範囲の設定     | 22 |
|   | HP Universal CMDB 7.52～8.xでのセットアップの概要 | 23 |
|   | Service Manager インテグレーションパッケージのインストール | 23 |
|   | RMIアダプタデータストアの追加                      | 23 |
|   | 新規変更アダプタデータストアの追加                     | 25 |
|   | Service Deskアダプタデータストアの追加             | 28 |
|   | HP Service Managerセットアップの概要           | 31 |
|   | UCMDB接続情報の追加                          | 31 |
|   | UCMDB 7.52システムをサポートするためのプロセス定義レコードの編集 | 32 |
|   | インテグレーションのマルチテナントサポートの有効化             | 32 |
|   | Service Managerシステム情報レコードの設定          | 33 |
|   | テナント固有UCMDBユーザIDとパスワード値の追加            | 34 |
|   | 既存の会社レコードへのUCMDB顧客IDの追加               | 34 |
|   | Service ManagerからUCMDBへの既存の会社の同期      | 35 |
|   | 会社レコードがUCMDBにあるかどうかの確認                | 35 |
|   | UCMDBとの既存の会社の再同期                      | 36 |
|   | 同期されている会社の非アクティブ化                     | 36 |
|   | 非アクティブな会社の再アクティブ化                     | 37 |
|   | 調停ルールの追加                              | 37 |
|   | 調整での結合テーブルの使用                         | 37 |
|   | 調整の順序                                 | 38 |

|   |           |
|---|-----------|
| DEM 調停ルールの作成 .....                          | 38        |
| 検出イベントマネージャルールの追加 .....                     | 38        |
| 変更とインシデントレコードに表示されるCI属性 .....               | 39        |
| インテグレーションによってオープンした変更とインシデントレコードの検索 .....   | 40        |
| UCMDB 9.0での変更データのプッシュの定義 .....              | 40        |
| UCMDB 7.52～8.xでの変更レプリケーションジョブの定義 .....      | 41        |
| UCMDB 9.0でのRMIデータのプッシュの定義 .....             | 42        |
| UCMDB 7.52～8.xでのRMIレプリケーションジョブの定義 .....     | 43        |
| データプッシュ、またはレプリケーションジョブのスケジュール .....         | 44        |
| UCMDB 9.0データプッシュジョブのスケジュール .....            | 44        |
| UCMDB 7.52と8.xのレプリケーションジョブのスケジュール .....     | 46        |
| <b>3 インテグレーションのカスタマイズ .....</b>             | <b>47</b> |
| インテグレーションのアーキテクチャ .....                     | 48        |
| インテグレーションのクラスモデル .....                      | 48        |
| UCMDB 9.0 .....                             | 48        |
| UCMDB 7.52と8.x .....                        | 48        |
| インテグレーションTQLクエリ .....                       | 51        |
| TQLクエリの要件 .....                             | 54        |
| Service Manager Web サービス .....              | 54        |
| 管理フィールド .....                               | 55        |
| Service Manager 調停ルール .....                 | 58        |
| パフォーマンスについて .....                           | 59        |
| DEMルールでの依存関係 .....                          | 59        |
| Service Manager 検出イベントマネージャルールの追加 .....     | 59        |
| DEMルールが実行される条件の変更 .....                     | 59        |
| テナント固有DEMルールの追加 .....                       | 60        |
| DEMルールが行うアクションの変更 .....                     | 61        |
| CIタイプの管理フィールドリストの更新 .....                   | 61        |
| 変更やインシデントレコードをオープンするカスタムJavaScriptの作成 ..... | 61        |
| UCMDBでのインテグレーションコンポーネントの表示 .....            | 63        |
| リンクルート設定の変更 .....                           | 63        |
| UCMDB 9.0 .....                             | 63        |
| UCMDB 7.52と8.x .....                        | 63        |
| オブジェクトルート設定の変更 .....                        | 64        |
| UCMDB 9.0 .....                             | 64        |
| UCMDB 7.52と8.x .....                        | 65        |
| カスタマイズのオプション .....                          | 67        |
| UCMDB クラスモデルへのCI属性の追加 .....                 | 67        |
| CI属性の計算の有効化 .....                           | 70        |
| CI属性の [変更をモニタ] オプションの有効化 .....              | 71        |
| Web サービスオブジェクトへのCI属性のマッピング .....            | 73        |
| インテグレーションアダプタのコードベースの再ロード .....             | 75        |
| CI属性をサポートする Web サービスプロジェクトの作成 .....         | 76        |
| CI属性をサポートするための管理フィールドの追加 .....              | 78        |
| インテグレーションへのCIタイプの追加 .....                   | 81        |
| UCMDB クラスモデルへのCIタイプの追加 .....                | 82        |

|   |     |
|---|-----|
| UCMDB 9.00.....                                   | 82  |
| UCMDB 7.52 と 8.x.....                             | 82  |
| サンプルのCIタイプ.....                                   | 82  |
| CIタイプの属性を同期するためのTQLクエリの作成.....                    | 85  |
| TQLクエリ内のすべての関係に追跡修飾子が存在することの確認.....               | 89  |
| CIタイプ属性の計算の有効化.....                               | 90  |
| CIタイプTQLクエリのSL変換ファイルへのマップ.....                    | 92  |
| CIタイプ属性のWebサービスオブジェクトへのマップ.....                   | 94  |
| ServiceDeskAdapterのコードベースの再ロード.....               | 97  |
| インテグレーション変更アダプタへのカスタムTQLクエリの追加.....               | 98  |
| インテグレーションデータプッシュ（レプリケーションジョブ）へのカスタムTQLクエリの追加..... | 99  |
| UCMDB 9.00データプッシュジョブの定義.....                      | 99  |
| UCMDB 7.52 と 8.xのレプリケーションジョブ.....                 | 100 |
| CIタイプをサポートするWebサービスオブジェクトの作成.....                 | 101 |
| CIタイプをサポートするための管理フィールドの追加.....                    | 103 |





# 1 はじめに

本章では、HP Universal CMDBとHP Service Managerのインテグレーションに関する以下の項目を説明します。

- インテグレーションの目的 (10ページ)
- 本ガイドの対象者 (10ページ)
- UCMDBに格納される情報について (10ページ)
- Service Managerに格納される情報について (11ページ)
- マルチテナント (マルチカンパニー) のサポート (11ページ)

## インテグレーションの目的

HP Universal CMDB (UCMDB) と HP Service Manager 間のインテグレーションにより、UCMDB システムと Service Manager システム間で、構成アイテム (CI) の実際のステータスに関する情報を共有できるようになります。ITIL プロセスにおける構成管理と変更管理のベストプラクティスを実装する組織は、このインテグレーションを用いることにより、CI が、組織がサポートすることに同意した属性値を実際に取り替えていることを確認できます。

このインテグレーションを用いることにより、Service Manager の変更やインシデントのレコードを自動的に作成し、予期しない属性値を取る CI を更新したり、切り戻すことができます。Service Manager では、CI の実際のステータスが CI レコードで定義された予期されるステータスと一致しない場合に実行するアクションをプログラムにより定義できます。

インテグレーションにより、ユーザはいくつかの方法で CI の実際のステータス情報を表示できます。

- デフォルトでは、インテグレーションにより、定期的な UCMDB 同期スケジュール中に Service Manager CI レコードの管理フィールドが自動的に更新されます。また、変更やインシデントのレコードを自動的に作成するようにインテグレーションを設定することもできます。
- Service Manager ユーザは、CI レコードの [実際のステータス] タブを確認することにより、CI の現在の実際のステータスを表示できます。[実際のステータス] タブをクリックすると、Service Manager は UCMDB に対する Web サービス要求を行い、要求が返したすべての CI 属性を表示します。Service Manager が Web サービスコールを作成するのは、ユーザがタブをクリックした場合のみです。
- Service Manager ユーザは [UCMDB のビュー] オプションを使用することで、UCMDB システムにログインし、UCMDB からの現在の CI 属性を表示できます。Service Manager ユーザは、UCMDB システムにログインするための有効な UCMDB ユーザ名とパスワードを必要とします。

## 本ガイドの対象者

本ガイドは、UCMDB、および Service Manager システム間の接続を確立、保守を行うシステム実装者やシステム管理者を対象としています。本ガイドでは、読者に両システムへの管理アクセス権があることを想定しています。本ガイドにある手順は、UCMDB と Service Manager のヘルプシステムにある情報と重複する場合がありますが、便宜上掲載してあります。

## UCMDB に格納される情報について

UCMDB システムは、CI の実際のステータスと CI 関係を CI 属性として格納します。通常、UCMDB は 1 つまたは複数のディスカバリメカニズム (フィーダ) を使用して、CI 属性値を自動的に検出します。UCMDB と Service Manager のインテグレーションでは、UCMDB システムで利用できる CI 属性のサブセットのみが使用されます。

詳細については、[インテグレーションのカスタマイズ](#) (47 ページ) を参照してください。

## Service Managerに格納される情報について

Service Managerシステムは、CIの管理ステータスや予期ステータス、およびCI関係を、CIレコード内の属性値として格納します。インテグレーションに含めるには、UCMDBシステムのCI属性をService Manager CIレコードの管理フィールドにマップする必要があります。インテグレーションを管理するService Manager Webサービスをカスタマイズすることにより、インテグレーションに含まれる管理フィールドの追加、削除、または更新を行えます。

CIの実際のステータスがCIレコードで定義されている予期されるステータスと一致しない場合、システムはアクションを実行します。Service Managerは、このアクションを定義するルールのセットに従って動作します。これらのルールは、Service Managerの検出イベントマネージャ (DEM) で定義します。次の操作を行います。

- 実際のステータスに一覧される属性値に一致するように、CIレコードを自動的に更新する (これがデフォルトの動作です)。
- 実際のステータスと管理ステータス間の差異を確認するための変更レコードを自動的に作成する。
- 実際のステータスと管理ステータス間の差異を確認するためのインシデントレコードを自動的に作成する。

## マルチテナント (マルチカンパニー) のサポート

HP Universal CMDB (UCMDB) と HP Service Manager のインテグレーションはマルチテナント設定をサポートしており、これにより、Service Manager と UCMDB の両システムが、会社IDで構成アイテム (CI) と構成アイテム関係 (CIR) を追跡します。マルチテナント設定では、インテグレーションをカスタマイズして、各テナントが自社の会社IDに一致するCIとCIRのみを表示して作業できるようにできます。マルチテナントは、複数のテナントにサービスとして構成管理を提供する管理サービスプロバイダ (MSR) 向けです。

### マルチテナントの要件

インテグレーションがマルチテナントをサポートするには、システムは以下の要件に一致する必要があります。

- HP Universal CMDBバージョン8.02以降のシステム
- HP Service Managerバージョン9.20のシステム
- UCMDBとService Manager間のインテグレーションが有効であること
- Service Managerシステム上でマルチカンパニーモードが有効であること
- Service Manager上でproblemスケジュールプロセスが動作していること

マルチテナントインテグレーションに関するその他の情報については、<http://support.openview.hp.com>にあるHP SoftwareサポートWebサイトを参照してください。次の関連マニュアルもダウンロードできます。

- HP Service Manager ヘルプ。バージョン 9.20

## UCMDBに格納されるマルチテナント情報について

UCMDBシステムは、各CIとCIRの会社ID属性を格納します。会社IDにより、UCMDBシステムがCIデータの更新に使用するアダプタと同期スケジュールが決定します。各CIと関係レコードは、1つの会社IDのみを取ることができます。UCMDBシステムはService Managerシステムから会社IDを取得します。

複数のテナント（会社）が同一のCIを共有する場合、各テナントはCIを記述する独自の一意なCIレコードを取ります。実際には、UCMDBシステムは1つの管理資産を追跡するために複数のCIレコードを作成します。各テナントCIレコードはそのテナント固有であり、会社の一意な会社IDを一覧します。

## Service Managerに格納されるマルチテナント情報について

Service Managerは、マルチテナント設定での各テナントを記述する会社レコードを格納します。Service Managerシステムは、会社IDの確定的なソースであり、UCMDBシステムに新規情報と更新された情報をプッシュします。

Service Managerは、マルチテナント設定で、各CIと関係の会社IDを追跡します。CIレコードは、そのCIレコードを検出したUCMDBフィールドの会社IDを継承します。関係レコードは、関係の親CIの会社IDを継承します。

ベストプラクティスの実装では、Service Managerはデータ制限を用いて、CIの会社IDがオペレータの会社IDに一致するCIと関係レコードのみをオペレータが表示できるようにします。データ制限でビューを制限すると、Service Managerにより、変更要求とインシデントなどのその他すべての関連レコードへのビューも制限されます。

## 一意な論理名

Service Managerでは、すべてのCIに一意の論理名があることが必要です。論理名生成プロセスで重複する論理名の値が生成されると、Service Managerは、論理名の最後にアンダースコアと番号を付記して論理名を一意にします。たとえば、2つのCIが論理名mytesthostを取る場合、2番目のCIの名前はmytesthost\_1になります。重複するCIがもう1つあれば、その名前はmytesthost\_2になります。

## 会社レコードの同期

システムが、マルチテナントサポートのための条件をすべて満たす場合、Service ManagerはUCMDBシステムに会社レコードの会社IDをプッシュするためのスケジュールレコードを作成します。Service Managerは、UCMDBシステムに会社IDをプッシュするかどうかの判定に次のルールを用います。

表1 Service ManagerがUCMDBと会社IDを同期する条件

| 条件   | テナント情報が同期されるかどうか | 作成されるスケジュールレコードと、UCMDBで行われるアクション   |
|--|------------------|--|
| <ul style="list-style-type: none"> <li>UCMDBとService Managerのインテグレーションが<b>有効</b></li> <li>Service Managerでマルチカンパニーモードが<b>有効</b></li> <li>Service Managerで新しい会社レコードを作成する</li> </ul>                   | はい               | Synch Company with UCMDB - <UCMDB 会社ID> <ul style="list-style-type: none"> <li>新しい会社IDの追加</li> </ul>           |
| <ul style="list-style-type: none"> <li>UCMDBとService Managerのインテグレーションが<b>有効</b></li> <li>Service Managerでマルチカンパニーモードが<b>有効</b></li> <li>UCMDBと同期していない既存会社レコードを更新する</li> </ul>                       | はい               | Synch Company with UCMDB - <UCMDB 会社ID> <ul style="list-style-type: none"> <li>新しい会社IDの追加</li> </ul>           |
| <ul style="list-style-type: none"> <li>UCMDBとService Managerのインテグレーションが<b>有効</b></li> <li>Service Managerでマルチカンパニーモードが<b>有効</b></li> <li>UMCDBと同期している会社に対し、マルチカンパニーリストで会社を表示するオプションを無効にする</li> </ul> | はい               | Inactivate Company with UCMDB - <UCMDB 会社ID> <ul style="list-style-type: none"> <li>既存の会社IDの非アクティブ化</li> </ul> |
| <ul style="list-style-type: none"> <li>UCMDBとService Managerのインテグレーションが<b>有効</b></li> <li>Service Managerでマルチカンパニーモードが<b>有効</b></li> <li>既存会社レコードに対し、UMCDBとの再同期を行うオプションを選択する</li> </ul>              | はい               | Synch Company with UCMDB - <UCMDB 会社ID> <ul style="list-style-type: none"> <li>新しい会社IDの追加</li> </ul>           |

表1 Service ManagerがUCMDBと会社IDを同期する条件

| 条件  | テナント情報が同期されるかどうか | 作成されるスケジュールレコードと、UCMDBで行われるアクション                           |
|---|------------------|--|
| <ul style="list-style-type: none"> <li>UCMDBとService Managerのインテグレーションが<b>有効</b></li> <li>Service Managerでマルチカンパニーモードが<b>有効</b></li> <li>非アクティブな会社に対し、マルチカンパニーリストで会社を表示するオプションを有効にする</li> </ul>       | はい               | Synch Company with UCMDB -<br><UCMDB 会社ID><br>会社IDの再アクティブ化 |
| <ul style="list-style-type: none"> <li>UCMDBとService Managerのインテグレーションが<b>無効</b></li> <li>Service Managerでマルチカンパニーモードが<b>有効</b></li> <li>UCMDBと同期している既存会社レコードを更新する</li> </ul>                         | いいえ              | なし   |
| <ul style="list-style-type: none"> <li>UCMDBとService Managerのインテグレーションが<b>無効</b></li> <li>Service Managerでマルチカンパニーモードが<b>有効</b></li> <li>Service Managerで新しい会社レコードを作成する</li> </ul>                    | いいえ              | なし   |
| <ul style="list-style-type: none"> <li>UCMDBとService Managerのインテグレーションが<b>有効</b></li> <li>Service Managerでマルチカンパニーモードが<b>有効</b></li> <li>UCMDBと同期していない会社に対し、マルチカンパニーリストで会社を表示するオプションを無効にする</li> </ul> | いいえ              | なし   |
| <ul style="list-style-type: none"> <li>UCMDBとService Managerのインテグレーションが<b>有効</b></li> <li>Service Managerでマルチカンパニーモードが<b>無効</b></li> <li>Service Managerで新しい会社レコードを作成する</li> </ul>                    | いいえ              | なし   |

## UCMDB顧客ID

マルチテナントインテグレーションを有効にすると、各会社レコードに [UCMDB 顧客ID] という新しいフィールドが表示されます。UCMDB と会社レコードを同期するには、まずこのフィールドの値を入力する必要があります。UCMDB 顧客ID 値を入力すると、このフィールドは読み取り専用になります。一度設定すると、会社の UCMDB 顧客ID は変更できません。

このフィールドには、10 文字までの数字データのみを入力できます。フィールド値は一意的な正の整数である必要があります。重複した値を入力したり、小数、負の数、ゼロを使用することはできません。

UCMDB システムは、単一テナントモードで実行時には UCMDB 顧客ID として自動的に 1 を使用します。Service Manager の会社にこの UCMDB 顧客ID 値を割り当てることで、マルチテナント実装でこのデフォルト値を再使用できます。出荷時設定では、UCMDB 顧客ID が 1 の Service Manager 会社はありません。

## UCMDB ユーザID とパスワード

マルチテナントインテグレーションを有効にすると、各会社レコードに [UCMDB ユーザID] と [UCMDB パスワード] という 2 つの新しいフィールドが表示されます。これらのフィールドを使用することで、[実際のステータス] タブの情報を要求するときに Service Manager が使用する接続情報を指定できます。これらのフィールドに入力するユーザ名とパスワードは、UCMDB システムで有効である必要があります。

会社情報レコードに入力されるユーザ名とパスワードは、システム情報レコードに入力されるユーザ名とパスワードより優先されます。これにより、管理サービスプロバイダーは、テナント別ベースで UCMDB システムへのアクセスを制御できます。会社固有の UCMDB ユーザ名とパスワードを入力しない場合、システム情報レコードに入力された資格情報が使用されます。

## 会社コード

マルチテナントインテグレーションでは、各会社レコードが一意的な会社コード（会社フィールド）値を取る必要があります。会社コードは必須フィールドであるため、既存の会社レコードには会社コード値が必要です。ただし、各会社レコードに一意的な会社コード値があることを確認する必要があります。



マルチテナントインテグレーションを有効にした後は、会社コード値は変更できません。変更すると、Service Manager データが同期されなくなるためです。

## CIレコードと関係に複製される会社情報

マルチテナントインテグレーションを有効にすると、Service Manager は、レプリケーション中に CI と関係レコードに SM 会社コード値を挿入します。Service Manager は UCMDB 顧客ID を使用して、一致する SM 会社コード値を検索します。

## インシデントレコードに複製される会社情報

マルチテナントインテグレーションを有効にし、UCMDB が新規 CI、更新された CI、削除された CI を検出したときにインシデントを作成するオプションを選択すると、Service Manager は、レプリケーション中にインシデントレコードに SM 会社コード値を挿入します。Service Manager は UCMDB 顧客ID を使用して、一致する SM 会社コード値を検索します。

## スケジュールレコード

Service Managerはproblemスケジュールプロセッサを使用して、UCMDBシステムへの会社IDの同期を管理します。システムのsm.cfgファイルに「sm system.start」行を追加することで、problemスケジュールプロセッサを自動的に有効にできます。また、[システムステータス] フォームで手動で有効にすることもできます。

同期の条件が上記の表を満たすと、UCMDB - <UCMDB 会社ID>の会社の同期スケジュールレコード (Synch Company with UCMDB - 1234567890、など) が作成されます。会社を非アクティブにすると、UCMDB - <UCMDB 会社ID>の会社の非アクティブ化スケジュールレコード (Inactivate Company with UCMDB - 1234567890、など) が作成されます。problemスケジュールプロセッサは、次のバックグラウンド処理の際に新しいスケジュールレコードを処理します。

Service Managerシステムが何らかの理由でUCMDBシステムに接続できない場合、次のスケジュール間隔 (出荷時設定での間隔は5分間) に会社の同期を再スケジュールします。problemスケジュールプロセッサは、再スケジュールされたステータスでスケジュールレコードを更新します。Service ManagerシステムがUCMDBシステムに接続中にその他のエラーメッセージを受信すると、スケジュールレコードは、「application failed due to error - check msglog for possible messages」ステータスに更新されます。



## 2 インテグレーションのセットアップ

本章では、HP Service Manager と HP UCMDB 間のインテグレーションをセットアップするための次のトピックについて説明します。

- [インテグレーションの要件](#) (18ページ)
- [インテグレーションのセットアップの概要](#) (18ページ)
- [HP Universal CMDB 9.00でのセットアップの概要](#) (19ページ)
- [HP Universal CMDB 7.52～8.xでのセットアップの概要](#) (23ページ)
- [HP Service Manager セットアップの概要](#) (31ページ)
- [Service Manager への UCMDB CI の転送](#) (40ページ)

## インテグレーションの要件

UCMDBとService Managerの間のインテグレーションを確立するには、次の必須コンポーネントをセットアップする必要があります。

- HP Universal CMDB 7.52以降のインストール
  - 次のCI属性を検出するためのUCMDBフィードとUCMDB間の同期の設定：
    - DDM（これはUCMDBのディスカバリフィードメカニズムです）
    - DDMi（これは、オプションである外部ディスカバリフィードメカニズムへのインテグレーションです）
  - CI属性を送信するためのUCMDBとService Manager間の同期の設定：
    - Service ManagerにCI属性を送信するためのService Deskターゲットアダプタの定義（すべてのインテグレーションで必要）

表2 バージョン別UCMDB同期要件

| UCMDBバージョン | 要件   |
|------------|--|
| 7.52～8.xx  | <ul style="list-style-type: none"><li>• 複合関係を同期するためのRMIソースアダプタを定義する（DDMiにのみ必要）</li><li>• CI変更を同期するための変更ソースアダプタを定義する（DDMiにのみ必要）</li></ul> |
| 9.00以上     | CI変更と複合関係を同期するためのDDMプローブを定義する。   |

- HP Service Manager 9.2以降のインストール
  - システム情報レコードへのUCMDB URLの追加
- HP Universal CMDBとHP Service Managerシステム間のネットワーク接続

システムのインストールと設定の方法については、UCMDBとService Managerのマニュアルを参照してください。

## インテグレーションのセットアップの概要

インテグレーションには、UCMDBとService Managerの両システムでセットアップが必要になります。

**タスク1: UCMDBシステムをセットアップします。**

HP Universal CMDB 9.00でのセットアップの概要（19ページ）を参照してください。

HP Universal CMDB 7.52～8.xでのセットアップの概要（23ページ）を参照してください。

**タスク2: Service Managerシステムをセットアップします。**

HP Service Managerセットアップの概要（31ページ）を参照してください。

**タスク3: UCMDBレプリケーションジョブを実行して、Service ManagerにCIを転送します。**

Service ManagerへのUCMDB CIの転送（40ページ）を参照してください。

# HP Universal CMDB 9.00でのセットアップの概要

UCMDB 9.00システムでインテグレーションをサポートするには、次のタスクを実行する必要があります。

**タスク1:** UCMDBとService Manager間にインテグレーションポイントを作成します。

インテグレーションポイントの作成（19ページ）を参照してください。

**タスク2:** マルチテナント設定をサポートするかどうか？

はい。手順3に移動します。

いいえ。UCMDBのセットアップは完了です。

**タスク3:** インテグレーションがサポートする各テナントの独立Data Flow Probeをインストールします。

各テナントの独立Data Flow Probeのインストール（20ページ）を参照してください。

**タスク4:** テナント固有Data Flow Probeを起動します。

テナント固有Data Flow Probeの起動（21ページ）を参照してください。

**タスク5:** テナント固有Data Flow ProbeのIPアドレス範囲を設定します。

テナント固有Data Flow ProbeのIPアドレス範囲の設定（22ページ）を参照してください。

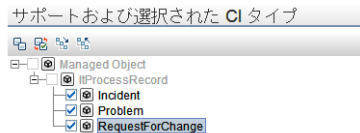
## インテグレーションポイントの作成


デフォルトのUCMDB 9.00インストールには、Service Manager インテグレーションパッケージが含まれています。インテグレーションパッケージを使用するには、インテグレーションの接続プロパティを一覧するインテグレーションポイントを作成する必要があります。

- 1 管理者としてUCMDB 9.00システムにログインします。
- 2 **[マネージャ]** > **[データフロー管理]** > **[Integration Studio]** をクリックします。UCMDBに、インテグレーションポイントのリストが表示されます。
- 3 **[新規インテグレーションポイントの作成]** ボタンをクリックします。    
UCMDBに、新規インテグレーションポイントプロパティの作成ウィンドウが表示されます。
- 4 **[インテグレーション名]** に一意のアダプタ名を入力します。「**SM Integration**」などです。
- 5 **[アダプタ]** で、**[Service Manager 7.1x]** を選択します。
- 6 **[インテグレーションはアクティブ化されています]** オプションを有効にします。
- 7 **[資格情報]** で、Service Managerに接続するのにインテグレーションが使用するユーザの資格情報を選択するか、作成します。たとえば、出荷時設定のService Manager System.Adminユーザアカウントを使用する**genericprotocol: System.Admin**資格情報を作成します。
- 8 **[Hostname/IP]** に、Service Managerシステムの名前を入力します。「**mysmserver**」などです。
- 9 **[ポート]** に、Service Managerシステムの通信ポートを入力します。「**13080**」などです。
- 10 **[OK]** をクリックします。

11 [サポートおよび選択された CI タイプ] には、[Managed Object] > [ItProcessRecord] から次のオプションを選択します：

- Incident
- Problem
- RequestForChange



12 [インテグレーションの保存] ボタンをクリックします。 

## 各テナントの独立 Data Flow Probe のインストール

マルチテナント設定をサポートする場合、各テナント用に独立データプローブをインストールする必要があります。出荷時設定では、UCMDB インストーラは1つの Data Flow Probe とサービスのみをインストールします。次の手順により、追加の Data Flow Probe をインストールして、オペレーティングシステムのコマンドプロンプトから起動できます。

- 1 管理者として UCMDB システムのホストにログインします。
- 2 システムディスクドライブに HP Universal CMDB 9.00 Setup Windows DVD を挿入します。
- 3 Data Flow Probe インストーラ (HPUCMDB\_DataFlowProbe\_90.exe) を起動します。
- 4 画面の指示に従ってウィザードを完了します。この際、インストールする各 Data Flow Probe には次の値を使用します。
  - a 各インストールフォルダの一意なパスを入力します。
  - b 各 Data Flow Probe に、同一の UCMDB アプリケーションサーバアドレスを使用します。
  - c 有効な Data Flow Probe アドレスを入力します。
  - d 各 Data Flow Probe 識別子の一意な名前を入力します。
  - e 各プローブに、一意な顧客の Data Flow Probe ドメインを作成します ([標準設定の UCMDB ドメインを使用] オプションをクリアします)。


- f 各プローブには、同じプローブゲートウェイとプローブマネージャ設定を使用します（結合プロセス、または独立プロセスの使用など）。

全体的なインストール方法については、『HP Universal CMDB デプロイメントガイド』を参照してください。

- 5 インストールする各 Data Flow Probe について、手順3から手順4を繰り返します。
- 6 テキストエディタで、プローブの DiscoveryProbe.properties ファイルを開きます。デフォルトで、このファイルは次のフォルダに配置されています：

<UCMDBのインストールフォルダ>\<Data Flow Probeのインストールフォルダ>\conf

C:\hp\UCMDB\DataFlowProbe\conf などです。

 <Data Flow Probeのインストールフォルダ>は、テナントごとに一意である必要があります。

- 7 設定ファイルで次のプロパティを編集します。

**表3 各テナントに設定するディスカバリプローブプロパティ**

| プロパティ  | 値  |
|--|--|
| serverName                                     | UCMDB サーバの名前を確認します                             |
| customerId                                     | この Data Flow Probe がサポートするテナントの顧客ID を入力します     |
| appilog.collectors.probe.name                  | プローブ名が、サーバ+テナント ID などのように一意であることを確認します         |
| appilog.collectors.domain                      | Data Flow Probe 名を確認します                        |
| appilog.collectors.local.ip                    | Data Flow Probe ゲートウェイ名を確認します                  |
| appilog.collectors.probe.ip                    | Data Flow Probe マネージャ名を確認します                   |
| appilog.collectors.rmi.port                    | 各プローブの一意なポートを入力します                             |
| appilog.collectors.rmi.gw.port                 | 各プローブの一意なポートを入力します                             |
| appilog.collectors.probe.html.port             | 各プローブの一意なポートを入力します                             |
| appilog.collectors.local.html.port             | 各プローブの一意なポートを入力します                             |
| appilog.collectors.ProbeUseSpecificRMIPortFrom | 各プローブの一意なポートを入力します。「0」を入力するとシステムが自動的にポートを選択します |
| appilog.collectors.bigBrother.port             | 各プローブの一意なポートを入力します                             |

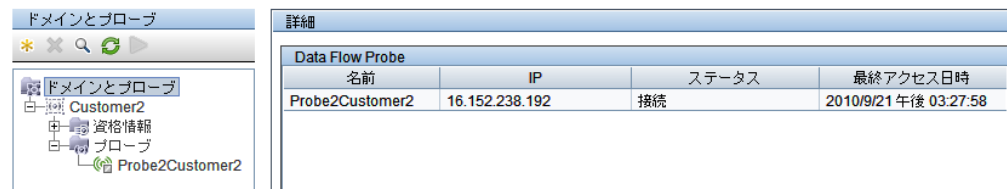
- 8 設定ファイルを保存します。
- 9 各テナントの Data Flow Probe について、手順6から手順8を繰り返します。

## テナント固有 Data Flow Probe の起動

- 1 OS のコマンドプロンプトを開き、プローブの bin フォルダに移動します。  
C:\hp\UCMDB\DataFlowProbe1\bin などです。
- 2 「gateway console」と入力します。
- 3 起動する各 Data Flow Probe について、手順1から手順2を繰り返します。


## テナント固有 Data Flow Probe の IP アドレス範囲の設定

- 1 設定する Data Flow Probe のテナントの会社 ID を使用し、管理者として UCMDB システム にログイン します。
- 2 **[データ フロー管理]** > **[Data Flow Probe 設定]** をクリック します。
- 3 起動するプローブが含まれる Data Flow Probe ドメインを 展開 します。**[Customer2]** などです。
- 4 プローブ ノードを 展開 し、起動する Data Flow Probe を 選択 します。**[Probe2Customer2]** などです。



The screenshot shows the UCMDB interface. On the left, a tree view under 'ドメインとプローブ' shows 'Customer2' expanded to 'プローブ', with 'Probe2Customer2' selected. On the right, a '詳細' (Details) window displays a table for 'Data Flow Probe'.

| 名前              | IP             | ステータス | 最終アクセス日時              |
|-----------------|----------------|-------|-----------------------|
| Probe2Customer2 | 16.152.238.192 | 接続    | 2010/9/21 午後 03:27:58 |

- 5 **[IP 範囲の追加]** ボタンをクリック します。 
- 6 Data Flow Probe がスキャンする IP 範囲を入力 します。オプションで、除外する IP 範囲を追加 します。
- 7 **[OK]** をクリック して、IP 範囲を保存 します。
- 8 設定する各 Data Flow Probe について、**手順 1** から **手順 7** を繰り返 します。

# HP Universal CMDB 7.52～8.xでのセットアップの概要

UCMDB 7.52～8.xシステムでインテグレーションをサポートするには、次のタスクを実行する必要があります。

タスク1: Service Manager インテグレーションパッケージをインストールします。

Service Manager インテグレーションパッケージのインストール (23ページ) を参照してください。

タスク2: UCMDBとソースフィード間にRMIアダプタデータストアを追加します。

RMIアダプタデータストアの追加 (23ページ) を参照してください。

タスク3: UCMDBとソースフィード間に新規変更アダプタデータストアを追加します。

新規変更アダプタデータストアの追加 (25ページ) を参照してください。

タスク4: UCMDBとService Manager間に、Service Deskアダプタデータストアを追加します。

Service Deskアダプタデータストアの追加 (28ページ) を参照してください。

## Service Manager インテグレーションパッケージのインストール

Service Manager インテグレーションコンポーネントは独立したパッケージであり、UCMDB インタフェースからインストールする必要があります。


1 管理者としてUCMDBシステムにログインします。



UCMDB ファイルシステムからログインします。パッケージインストールプロセスが、パッケージ用にクライアントのローカルファイルシステムを検索します。

2 [設定] タブを選択します。

3 [パッケージ マネージャ] をクリックします。インストールされているパッケージのリストが表示されます。

4 [サーバにパッケージをデプロイする(ローカルディスクから)] をクリックします。 

5 [追加] ボタンをクリックします。 

6 インストール先として、`integration_packages`フォルダを参照します。例：

`C:\hp\UCMDB\UCMDBServer\root\lib\integration_packages`

7 [smintegration.zip] パッケージを選択して、[開く] をクリックします。

8 [OK] をクリックして、パッケージを展開します。


## RMIアダプタデータストアの追加

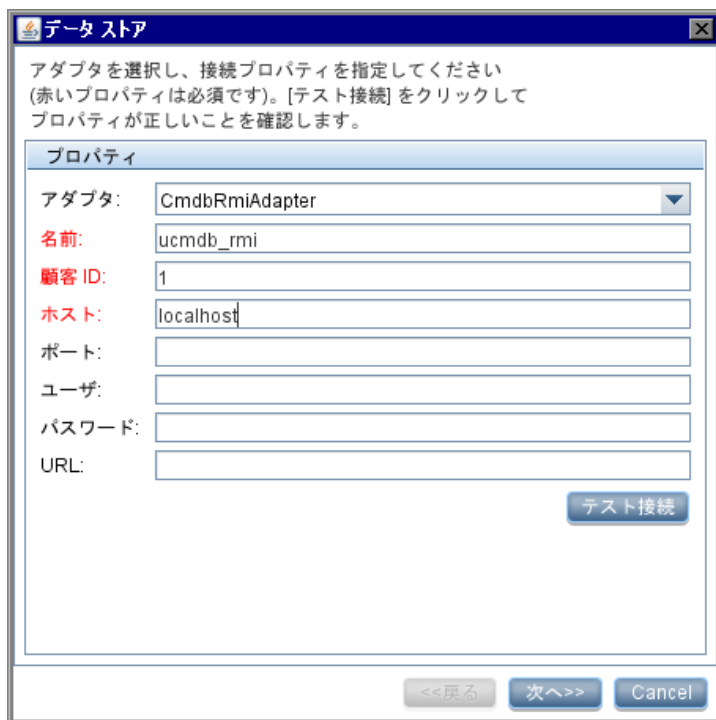
UCMDBとService Managerの同期において、ソースアダプタとして機能するRMIデータストアを作成する必要があります。RMIアダプタはUCMDBシステム上でCI関係を更新します。

1 管理者としてUCMDBシステムにログインします。

2 [設定] タブを選択します。

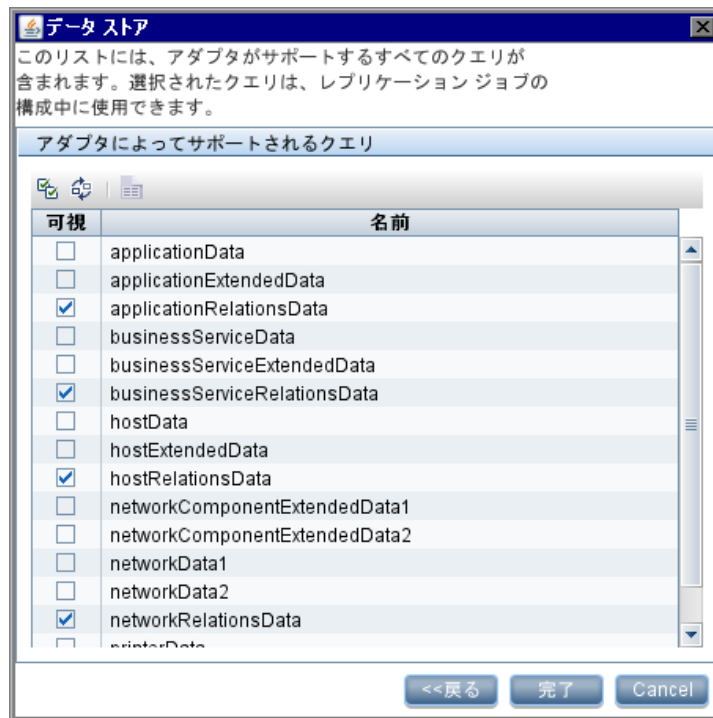
3 [フェデレート CMDB] をクリックします。インストールされているデータストアのリストが表示されます。

- 4 [追加] ボタンをクリックして、新規データストアを追加します。   
[データストア] プロパティウィンドウが表示されます。
- 5 [アダプタ] で [CmdbRmiAdapter] を選択します。
- 6 [名前] に一意のアダプタ名を入力します。「ucmdb\_rmi」などです。
- 7 [顧客 ID] に「1」と入力します。ただし、HP Software-as-a-Service を使用している場合は、お客様の顧客 ID 番号を入力します。
- 8 [ホスト] に、UCMDB システムの名前を入力します。「localhost」などです。



- 9 [次へ] をクリックします。
- 10 [アダプタによってサポートされるクエリ] で、次のクエリを選択します。
  - applicationRelationsData
  - businessServiceRelationsData
  - hostRelationsData
  - networkRelationsData





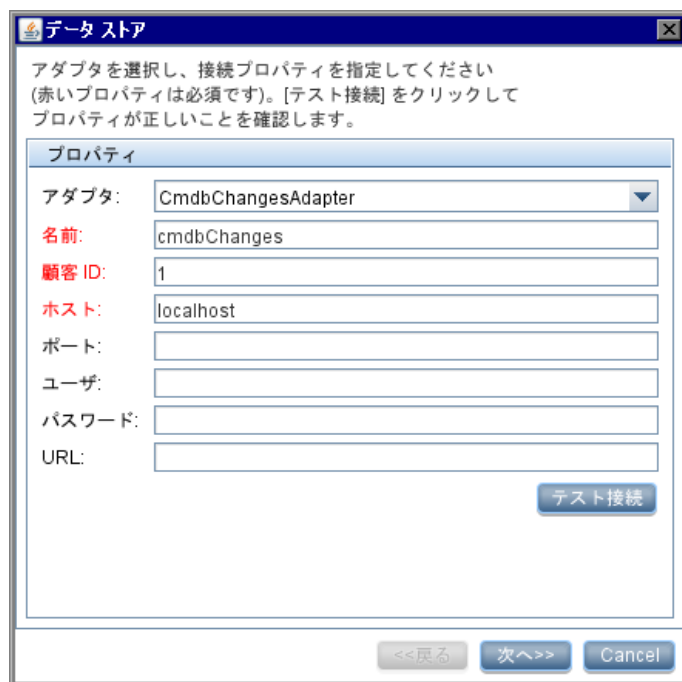
- 11 [完了] をクリックします。データストアが作成され、リスト内に表示されます。

## 新規変更アダプタデータストアの追加

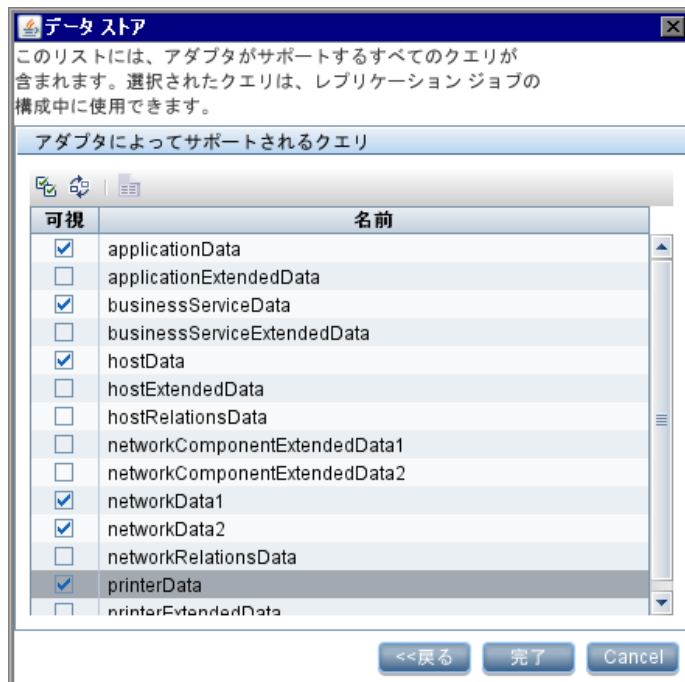
UCMDB から Service Manager の同期で、ソースアダプタとして機能する新規変更アダプタデータストアを作成する必要があります。変更アダプタはUCMDBシステム上でCI属性を更新します。

- 1 管理者としてUCMDBシステムにログインします。
- 2 [設定] タブを選択します。
- 3 [フェデレート CMDB] をクリックします。インストールされているデータストアのリストが表示されます。
- 4 [追加] ボタンをクリックして、新規データストアを追加します。 

[データストア] プロパティウィンドウが表示されます。
- 5 [アダプタ] で [CmdbChangesAdapter] を選択します。
- 6 [名前] に一意のアダプタ名を入力します。「**cmdbChanges**」などです。
- 7 [顧客 ID] に「1」と入力します。ただし、HP Software-as-a-Service を使用している場合は、お客様の顧客ID番号を入力します。
- 8 [ホスト] に、UCMDBシステムの名前を入力します。「**localhost**」などです。



- 9 [次へ] をクリックします。
- 10 [アダプタによってサポートされるクエリ] で、次のクエリを選択します。
  - applicationData
  - businessServiceData
  - hostData
  - networkData1
  - networkData2
  - printerData



- 11 **[完了]** をクリックします。データストアが作成され、リスト内に表示されます。

## Service Deskアダプタデータストアの追加

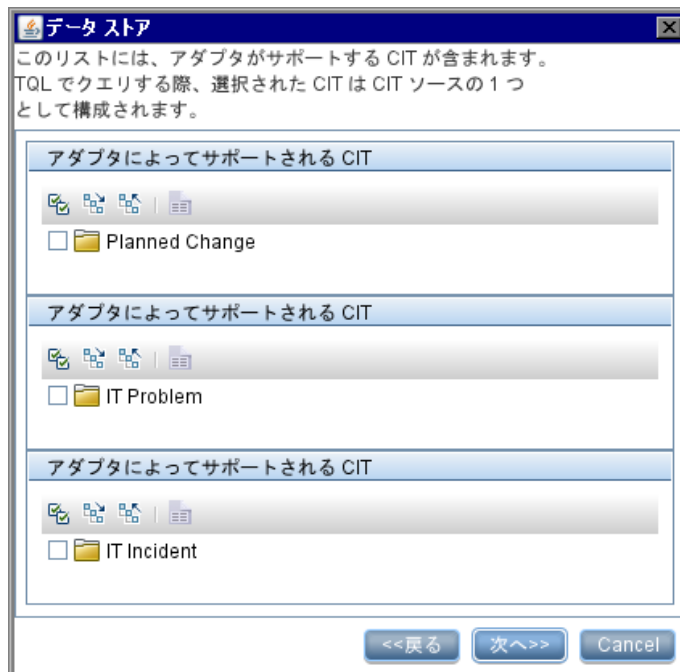
UCMDB から Service Manager へのレプリケーションジョブで、ターゲットアダプタとして機能する Service Desk データストアを作成する必要があります。このアダプタは、Service Manager システムへの接続情報を指定します。UCMDB 7.52～8.xでの変更レプリケーションジョブの定義 (41 ページ) と UCMDB 7.52～8.xでのRMIレプリケーションジョブの定義 (43 ページ) で、このアダプタを使用します。

- 1 管理者としてUCMDBシステムにログインします。
- 2 **設定** タブを選択します。
- 3 **フェデレート CMDB** をクリックします。インストールされているデータストアのリストが表示されます。
- 4 **[+]** ボタンをクリックして、新規データストアを追加します。   
[データストア] プロパティウィンドウが表示されます。
- 5 [アダプタ] で **ServiceDeskAdapter** を選択します。
- 6 [名前] に一意のアダプタ名を入力します。「**sm**」などです。
- 7 [顧客 ID] に「**1**」と入力します。ただし、HP Software-as-a-Service を使用している場合は、お客様の顧客ID番号を入力します。
- 8 [ホスト] に、Service Manager システムの名前を入力します。「**mysmserver**」などです。
- 9 [ポート] に、Service Manager システムの通信ポートを入力します。「**13080**」などです。
- 10 [ユーザ] に、Service Manager システム管理者のオペレータ名を入力します。「**System.Admin**」などです。
- 11 [パスワード] には、Service Manager システム管理者のパスワードを入力します。たとえば、System.Admin では [パスワード] フィールドを空欄にします。

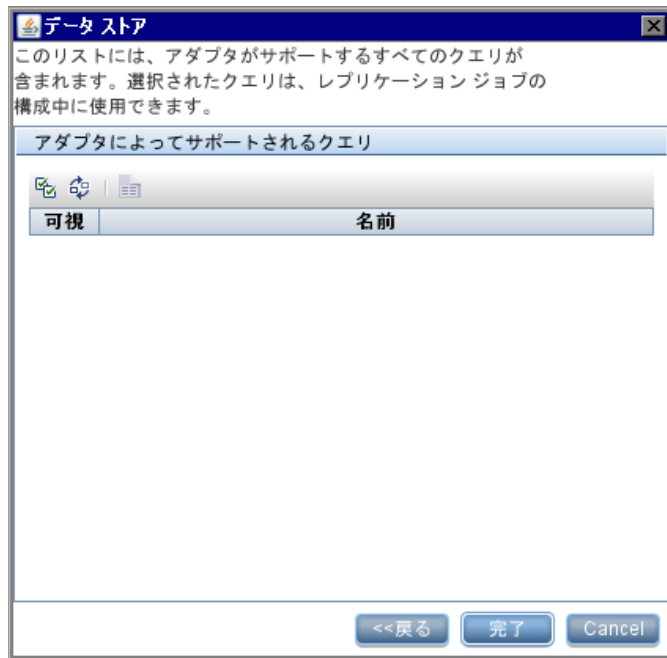


12 [次へ] をクリックします。

13 [アダプタによってサポートされる CIT] では、何も選択せずに、[次へ] をクリックします。



14 [アダプタによってサポートされるクエリ] でも、何も選択しません。



- 15 **完了** をクリックします。データストアが作成され、リスト内に表示されます。

# HP Service Manager セットアップの概要

Service Manager システムでインテグレーションをサポートするには、次のタスクを実行する必要があります。

タスク1: システム情報レコードにUCMDB接続情報を追加します。

UCMDB接続情報の追加 (31ページ) を参照してください。

タスク2: 必要に応じて、UCMDB 7.52システムをサポートするようにプロセス定義レコードを編集します。

UCMDB 7.52システムをサポートするためのプロセス定義レコードの編集 (32ページ) を参照してください。

タスク3: 必要に応じて、マルチテナントサポートを有効にします。

インテグレーションのマルチテナントサポートの有効化 (32ページ) を参照してください。

タスク4: UCMDB側のCIが既存のService Manager CIレコードに一致する場合をシステムがどのように決定するのかを定義する、調停ルールを追加します。

調停ルールの追加 (37ページ) を参照してください。

タスク5: UCMDB側のCI属性値がService Manager側のCI属性値と一致するときのシステムの動作を定義する、検出イベントマネージャールールを追加します。

検出イベントマネージャールールの追加 (38ページ) を参照してください。

## UCMDB接続情報の追加

インテグレーションには、UCMDBシステムからCI属性情報を取得するためのUCMDB接続情報が必要です。



Service Manager ユーザは、Service Manager リンクからUCMDBにログインしようとするとき、UCMDB ユーザ名とパスワードの入力が求められます。

- 1 管理者としてService Managerシステムにログインします。
- 2 **[システム管理]** > **[ベースシステム設定]** > **[その他]** > **[システム情報レコード]** をクリックします。
- 3 **[アクティブ統合]** タブを選択します。
- 4 **[HP Universal CMDB]** オプションを選択します。

フォームに **[UCMDB Web サービス URL]** フィールドが表示されます。

- 5 **[UCMDB Web サービス URL]** フィールドに、HP Universal CMDB Web サービス API への URL を入力します。URL は次の形式です：

`http://<UCMDB サーバ名>:<ポート>/axis2/services/ucmdbSMService`

<UCMDB サーバ名>にはUCMDBサーバのホスト名、<ポート>にはUCMDBサーバが使用する通信ポートを入力します。

- 6 **[ユーザ ID]** と **[パスワード]** に、UCMDBシステム上でCIを管理するのに必要なユーザ資格情報を入力します。たとえば、出荷時設定の資格情報は「**admin/admin**」です。
- 7 **[保存]** をクリックします。Service Managerに「情報レコードは更新されました。」と表示されます。

- 8 Service Managerシステムからログアウトします。
- 9 Service Managerシステムに管理者アカウントでログインします。CIレコードに [実際のステータス] タブが表示されますが、まずレプリケーションを完了し、Service ManagerシステムにUCMDBシステムのCIを入力する必要があります。

## UCMDB 7.52システムをサポートするためのプロセス定義レコードの編集

インテグレーションでUCMDB 7.52システムに接続する場合、プロセス定義レコードを編集してUCMDBシステムへのURLを更新する必要があります。インテグレーションでUCMDB 8.0以降のシステムに接続する場合、このステップはスキップできます。

- 1 管理者としてService Managerシステムにログインします。
- 2 [カスタマイズ] > [文書エンジン] > [プロセス] をクリックします。
- 3 [プロセス名] に「**am.view.ucmdb**」と入力して、[検索] をクリックします。
- 4 [初期 JavaScript] タブを選択します。
- 5 "The following line should be uncommented for use with UCMDB 7.5.x." というテキストを検索します。デフォルトでスクリプトの3行目にあります。
- 6 その次の行の先頭にあるコメント文字 (//) を削除します。デフォルトで4行目にあります。
- 7 "The following lines should be uncommented for use with UCMDB 8.0 and above." というテキストを検索します。デフォルトでスクリプトの7行目と8行目にあります。
- 8 その次の2行の先頭にコメント文字 (//) を追加します。デフォルトでスクリプトの7行目と8行目にあります。
- 9 [保存] をクリックします。
- 10 Service Managerシステムを停止して、再起動します。

## インテグレーションのマルチテナントサポートの有効化

マルチテナントサポートは、テナントに対してサービスとして構成管理を行う管理サービスプロバイダ (MSP) を対象としたインテグレーションのオプション機能です。マルチテナント設定では、各CIとCIRレコードは対応する会社IDを取ります。出荷時設定のService Managerでは、すべてのオペレータはどの会社IDであってもCIデータを表示できます。会社IDでCIデータへのアクセスを制限するには、データ制限を有効にし、[会社ID] フィールドを制限用クエリとして使用する必要があります。マルチカンパニーモードとデータ制限の詳細については、Service Managerヘルプを参照してください。

Service Manager 9.2システムとのインテグレーションでマルチテナントサポートを有効にするには、Service Managerで次のタスクを実行する必要があります。

**タスク1: Service Managerシステム情報レコードを設定します。**

Service Managerシステム情報レコードの設定 (33ページ) を参照してください。

**タスク2: 会社レコードにテナント固有UCMDB IDとパスワード値を追加します (オプション)。**

テナント固有UCMDBユーザIDとパスワード値の追加 (34ページ) を参照してください。

**タスク3: 既存の会社レコードにUCMDB顧客IDを追加します。**

既存の会社レコードへのUCMDB顧客IDの追加 (34ページ) を参照してください。



タスク4: 既存の会社レコードをUCMDBと同期します。

Service ManagerからUCMDBへの既存の会社の同期 (35ページ) を参照してください。

タスク5: Service Managerが会社レコードをUCMDBと同期したことを確認します (オプション)。

会社レコードがUCMDBにあるかどうかの確認 (35ページ) を参照してください。

タスク6: 既存の会社レコードをUCMDBと再同期します (必要な場合)。

UCMDBとの既存の会社の再同期 (36ページ) を参照してください。

タスク7: インテグレーションに含めない会社レコードを非アクティブ化します (必要な場合)。

同期されている会社の非アクティブ化 (36ページ) を参照してください。

タスク8: インテグレーションに含める非アクティブな会社レコードを再アクティブ化します (必要な場合)。

非アクティブな会社の再アクティブ化 (37ページ) を参照してください。

## Service Managerシステム情報レコードの設定

インテグレーションでマルチテナントを有効にするには、Service Managerシステム情報レコードに追加情報を入力する必要があります。



マルチテナントサポートを有効にするには、HP Universal CMDBバージョン8.02以降を使用する必要があります。それよりも前のバージョンのHP Universal CMDBでは、マルチテナントモードで実行しようとすると、エラーメッセージが表示されます。

- 1 管理者としてService Managerシステムにログインします。
- 2 [システム管理] > [ベースシステム設定] > [その他] > [システム情報レコード] をクリックします。
- 3 [全般] タブを選択します。
- 4 [マルチカンパニーモードで実行] オプションを有効にします。
- 5 [アクティブ統合] タブを選択します。

- 6 [HP Universal CMDB] オプションを選択します。

フォームに [UCMDB Web サービス URL] フィールドが表示されます。

- 7 [UCMDB Web サービス URL] フィールドに、CIの同期WebサービスAPIのURLを入力します。URLは次の形式です：

`http://<UCMDB サーバ名>:<ポート>/axis2/services/ucmdbSMService`


<UCMDB サーバ名>にはUCMDBサーバのホスト名、<ポート>にはUCMDBサーバが使用する通信ポートを入力します。

- 8 [ユーザ ID] と [パスワード] に、UCMDBシステム上でCIを管理するのに必要なユーザ資格情報を入力します。たとえば、出荷時設定の資格情報は「**admin/admin**」です。

- 9 [マルチテナント UCMDB Web サービス URL] フィールドに、会社IDの同期WebサービスAPIのURLを入力します。URLは次の形式です：


`http://<UCMDB サーバ名>:<ポート>/axis2/services/UcddbManagementService`

<UCMDB サーバ名>にはUCMDBサーバのホスト名、<ポート>にはUCMDBサーバが使用する通信ポートを入力します。

- 10 UCMDBシステムで会社IDを同期するのに必要なユーザ名とパスワードを入力します。  
たとえば、UCMDB 9.00の出荷時設定のシステム管理者用資格情報は「**sysadmin/sysadmin**」です。  
 UCMDB 8.xシステムを使用している場合、出荷時設定のマネージャ資格情報を使用できます。  
これは、「**manager/manager**」です。
- 11 **[保存]** をクリックします。Service Managerに「情報レコードは更新されました。」と表示されます。
- 12 Service Managerシステムからログアウトし、管理者アカウントでログインし直します。
- 13 **[システムステータス]** > **[表示オプション]** > **[すべてのタスク]** をクリックします。
- 14 problemスケジュールプロセスの隣にある**[コマンド]** フィールドに「**k**」と入力し、**[コマンドの実行]** をクリックします。problemスケジュールプロセスが終了するまで、数分待ちます。
- 15 **[スケジューラの開始]** をクリックします。
- 16 **[problem]** スケジュールプロセスをダブルクリックします。これで、システムはUCMDB向けマルチテナントをサポートするようになります。

## テナント固有UCMDBユーザIDとパスワード値の追加

[実際のステータス] タブの情報を要求するときにService Managerが使用する、テナント固有のUCMDBユーザ名とパスワードを入力できます。資格情報がない場合、Service Managerはすべてのテナントに対して、システム情報レコードにある資格情報を使用します。

 会社レコードに入力する資格情報は、システム情報レコードに入力する資格情報より優先されます。

- 1 管理者としてService Managerシステムにログインします。
- 2 **[システム管理]** > **[ベースシステム設定]** > **[会社]** をクリックします。
- 3 会社レコードの検索に使用する検索基準を入力します。すべての会社レコードを検索するには、検索フォームを空欄にしておきます。
- 4 **[検索]** をクリックします。
- 5 [UCMDB ユーザID]フィールドに、この会社がUCMDBに接続するのに使用するユーザ名を入力します。
- 6 [UCMDB パスワード] フィールドに、このUCMDBユーザ名のパスワードを入力します。
- 7 **[保存]** をクリックします。
- 8 資格情報を入力する各会社について、**手順3**から**手順7**を繰り返します。

## 既存の会社レコードへのUCMDB顧客IDの追加

次のステップを使用して、既存のService Manager会社レコードにUCMDB顧客IDを追加します。

- 1 管理者としてService Managerシステムにログインします。
- 2 **[システム管理]** > **[ベースシステム設定]** > **[会社]** をクリックします。
- 3 会社レコードの検索に使用する検索基準を入力します。すべての会社レコードを検索するには、検索フォームを空欄にしておきます。
- 4 **[検索]** をクリックします。
- 5 [UCMDB 顧客ID] フィールドに、この会社の数値を入力します。
- 6 **[保存]** をクリックします。

- 7 レコードをUCMDBと同期することを確認するプロンプトが表示されます。会社を今すぐ同期するには [はい]、会社を後で同期するには [いいえ] をクリックします。
- 8 [次へ] をクリックして、レコードリストの次の会社に移動します。
- 9 レコードリスト内の各会社について、手順5から手順8を繰り返します。

## Service Manager から UCMDB への既存の会社の同期

Service Manager システムには、マルチテナント インテグレーションで使用する会社レコードが既に含まれている場合があります。[UCMDB と同期] オプションを使用して、会社レコードを UCMDB システムと手動で同期することができます。

UCMDB にまだ同期していない会社レコード内のフィールドを更新する場合、会社を UCMDB に同期するかどうかのプロンプトが表示されます。



マルチカンパニーリストに会社を表示するオプションを無効にしている場合、または会社に関連する保留中のスケジュールレコードがある場合、会社レコードの同期は求められません。詳細については、[同期されている会社の非アクティブ化](#) (36ページ) を参照してください。

- 1 管理者として Service Manager システムにログインします。
- 2 [システム管理] > [ベースシステム設定] > [会社] をクリックします。
- 3 会社レコードの検索に使用する検索基準を入力します。すべての会社レコードを検索するには、検索フォームを空欄にしておきます。
- 4 [検索] をクリックします。
- 5 更新する会社レコードを選択します。
- 6 会社レコードを更新します。
- 7 [保存] をクリックします。レコードを UCMDB と同期することを確認するプロンプトが表示されます。



同期を実行するかどうかにかかわらず、会社レコードは保存されます。

## 会社レコードが UCMDB にあるかどうかの確認


マルチテナント インテグレーションが有効である場合、各会社レコードには、UCMDB 顧客 ID が UCMDB システムと同期済みであるかどうかを一覧する、読み取り専用フィールドが表示されます。このフィールドは、マルチテナント UCMDB インテグレーションが有効である場合のみ表示されます。


- 1 管理者として Service Manager システムにログインします。
- 2 [システム管理] > [ベースシステム設定] > [会社] をクリックします。
- 3 会社レコードの検索に使用する検索基準を入力します。すべての会社レコードを検索するには、検索フォームを空欄にしておきます。
- 4 [検索] をクリックします。
- 5 [UCMDB と同期] フィールドのステータスを確認します。  
チェックボックスがオンである場合、Service Manager と UCMDB システムの会社 ID は同期済みです。  
チェックボックスがオフである場合、Service Manager は UCMDB システムにこの会社をまだ追加していません。

## UCMDB との既存の会社の再同期

なんらかの理由でUCMDB データを失った場合のために、Service Manager では、UCMDB システムと会社レコードを再同期することができます。たとえば、インテグレーションテスト中にUCMDB データを意図的に削除したり、障害後にデータを復旧する必要がある場合などです。UCMDB との再同期オプションを使用することで、強制的にService Manager にUCMDB システムと会社を同期させることができます。

- 1 管理者としてService Manager システムにログインします。
- 2 **[システム管理]** > **[ベースシステム設定]** > **[会社]** をクリックします。
- 3 会社レコードの検索に使用する検索基準を入力します。 すべての会社レコードを検索するには、検索フォームを空欄にしておきます。
- 4 **[検索]** をクリックします。
- 5 同期する会社レコードを選択します。
- 6 **[UCMDB と同期]** チェックボックスの隣にある **[再同期]** ボタンをクリックします。

 **[再同期]** ボタンが利用できる会社レコードは、UCMDB と同期済みで、**[UCMDB と同期]** チェックボックスがオンである会社レコードのみです。

 UCMDB システムにこの ID 値の会社が既に存在している場合、再同期要求は無視されます。UCMDB と会社を再同期する既存のスケジュールレコードが存在する場合にも、再同期要求は無視されます。この場合、「この会社を UCMDB と再同期させるためのスケジュールレコードはすでに追加されています。」というメッセージが表示されます。

## 同期されている会社の非アクティブ化

UCMDB と会社レコードを同期した後は、レコードは削除できなくなります。その代わりに、会社レコードを非アクティブ化することができます。これにより、UCMDB システムはそれ以降、その会社の CI の更新をすべて停止します。その会社のあらゆる既存の CI データはまだ UCMDB システムに存在し、非アクティブな UCMDB 顧客 ID に関連付けられていますが、これ以降は UCMDB システムにおいて会社とその関連 CI は両方とも表示されなくなります。

- 1 管理者としてService Manager システムにログインします。
- 2 **[システム管理]** > **[ベースシステム設定]** > **[会社]** をクリックします。
- 3 会社レコードの検索に使用する検索基準を入力します。 すべての会社レコードを検索するには、検索フォームを空欄にしておきます。
- 4 **[検索]** をクリックします。
- 5 非アクティブ化する会社レコードを選択します。
- 6 **[マルチカンパニーモード時に会社をリストに表示]** で **[いいえ]** を選択します。
- 7 **[保存]** をクリックします。
- 8 この会社が以前に UCMDB と同期されている場合、非アクティブにすることを確認するプロンプトが表示されます。
- 9 非アクティブにするには **[はい]**、変更をキャンセルするには **[いいえ]** をクリックします。

## 非アクティブな会社の再アクティブ化

Service Manager システムで非アクティブな会社を再アクティブ化して、マルチテナントインテグレーションに含めることができます。UCMDBがこの会社の任意のCI更新を処理できるようにするため、UCMDBと会社を同期する必要もあります。

- 1 管理者としてService Managerシステムにログインします。
- 2 [システム管理] > [ベースシステム設定] > [会社] をクリックします。
- 3 会社レコードの検索に使用する検索基準を入力します。すべての会社レコードを検索するには、検索フォームを空欄にしておきます。
- 4 [検索] をクリックします。
- 5 再アクティブ化する会社レコードを選択します。
- 6 [マルチカンパニーモード時に会社をリストに表示] で [はい] を選択します。
- 7 [保存] をクリックします。UCMDBで会社を再アクティブ化することを確認するプロンプトが表示されます。
- 8 [はい] をクリックします。Service Managerにより、会社を再アクティブ化するスケジュールレコードが作成されます。

## 調停ルールの追加

Service Managerシステムには、UCMDBシステム内のCIと一致するCIレコードが既に含まれている場合があります。Service Managerシステムに重複するCIレコードを追加するのではなく、Service Managerを設定して、特定のフィールド値を基に両システム間でCIレコードが調整されるようにします。

Service Managerは必ず、Service Managerテーブルの一意のキーフィールドおよびucmdb.idフィールドを基にして、CIレコードの調整を試みます。[DEM 調停ルール] フォームで、調整の基準とする追加フィールドを指定できます。Service Managerがこれらのフィールドのいずれかに一致する値を検出すると、受信するUCMDBレコードの属性でService Manager CIレコードを更新します。

マルチテナントが有効である場合、Service Managerは、レプリケーションジョブの会社IDと一致する会社IDを持つCIのみを調整します。たとえば、会社2のCIを複製する際は、会社番号2に対応する会社コードのService Manager CIレコードにのみ調停ルールが適用されます。

調整フィールドを指定するには、Service ManagerとUCMDBシステムの両方でテーブルとフィールド名を理解する必要があります。UCMDBシステムの特定の属性について調整する場合、その属性に対応するService Manager管理フィールドが存在することを確認する必要があります。このようなマッピングが存在しない場合、Service Managerは、CIレコード内で一致する値を検索することができません。



Service Manager内に対応するフィールドが存在しないUCMDB属性があります。対応するフィールドが存在しない場合、Service Managerシステムをカスタマイズして、一致フィールドの追加が必要となる場合があります。

## 調整での結合テーブルの使用

調停ルールを設定するときに、調整するデバイスタイプに (devtypeテーブルで定義される) joindef定義がある場合は、deviceテーブルではなく結合テーブル名を使用します。たとえば、コンピュータCIを調整する場合は、deviceテーブルではなくjoincomputerテーブルを使用します。

## 調整の順序

調停ルールでは、一致するCI値を検索するService Managerテーブルとフィールドを指定します。また、Service Managerが調停ルールを処理する順序も指定します。デフォルトで、Service Managerはフィールド名のアルファベット順にルールを処理します。たとえば、Service Managerは、ci.nameフィールドでCIを調整する前に、asset.tagフィールドでCIを調整します。

Service ManagerがCIを調整する順序を変更するには、シーケンスフィールドに数値を追加します。たとえば、次の調停ルールを使用することで、Service Managerはci.nameフィールドでCIを処理してから、asset.tagフィールドでCIを調整するようになります。

表4 シーケンスで順番付けられた調停ルールのサンプル

| テーブル名        | フィールド名    | シーケンス |
|--------------|-----------|-------|
| joincomputer | ci.name   | 1     |
| joincomputer | asset.tag | 2     |

## DEM調停ルールの作成

検出イベントマネージャ（DEM）調停ルールでは、既存のCIレコードがUCMDBシステムにあるCIと一致するかどうかを判断するのに使用するService Managerフィールドを指定できます。管理者は通常、Service Managerによって重複するCIレコードが作成されないように、UCMDBレプリケーションジョブを開始する前に調停ルールを指定します。

DEM調停ルールを作成するには：

- 1 管理者としてService Managerシステムにログインします。
- 2 **[カスタマイズ]** > **[Web サービス]** > **[DEM 調停ルール]** をクリックします。[DEM 調停レコード] フォームが表示されます。
- 3 [テーブル名] に、調整する基準となるフィールドを含むService Managerテーブルの名前を入力します。
- 4 [フィールド名] に、調整する基準となる値を含むService Managerフィールドの名前を入力します。
- 5 [シーケンス] には、Service Managerがこのルールを実行する順番を指定します。



シーケンス値を指定しないと、Service Managerはフィールド名のアルファベット順に処理します。

- 6 **[新規]** をクリックします。調停ルールが作成されます。

## 検出イベントマネージャルールの追加

Service Managerは、検出イベントマネージャ（DEM）を使用して、受信する構成アイテム（CI）レコードの実際のステータスがHP Service Manager内の管理ステータスと異なる場合に、システムが実行すべきアクションを定義します。DEMルールを使用することで、Service Managerシステムが受信するUCMDBデータを基にしてCIレコードの追加、更新、または削除を行うかどうかを定義できます。

Service Managerでは、以下のルールオプションを利用できます：



- **一致するレコードが存在しない場合のアクション**：これは、一致するCIレコードが見つからない場合に Service Manager が実行するアクションです。
  - **レコードの追加**：(デフォルト) Service Manager は、一致するレコードが見つからないときに CI レコードを追加します。Service Manager が CI レコードの照合に使用するフィールドを定義するには、[調停ルールの追加](#) (37 ページ) を参照してください。
  - **インシデントのオープン**：Service Manager は、ユーザが新規 CI レコードを確認できるように、インシデントをオープンします。このインシデントにより、ユーザは新規 CI レコードがビジネスプラクティスに準拠しているかどうかを調査できます。
  - **変更のオープン**：Service Manager は、ユーザが新規 CI レコードを確認できるように、計画外の変更をオープンします。この変更により、ユーザは新規 CI レコードがビジネスプラクティスに準拠しているかどうかを調査できます。CI レコードが準拠している場合、変更を承認できます。CI レコードが準拠していない場合、変更を却下して CI レコードを削除できます。変更レコードには、現在の属性値と推奨される属性値の両方が一覧されます。
- **レコードは存在するが、予期しないデータが見つかった場合のアクション**：これは、一致する CI 属性値が見つからない場合に Service Manager が実行するアクションです。
  - **変更のオープン**：(デフォルト) Service Manager は、CI レコードの実際のステータスを確認するために、計画外の変更を開きます。この変更により、ユーザは新規属性値がビジネスプラクティスに準拠しているかどうかを調査できます。値が準拠している場合、変更を承認できます。値が準拠していない場合、変更を却下して、CI 属性値を元の管理ステータスに戻すことができます。
  - **結果のログの記録とレコードの更新**：Service Manager は、CI レコードの実際のステータスの結果をログに記録し、CI レコードを更新します。
  - **インシデントのオープン**：Service Manager は、CI レコードの実際のステータスを調査するためにインシデントを開き、レコードを Service Manager に準拠させるために実行や開始が必要となるアクションを判断します。
- **レコードが削除予定である場合のアクション**：これは、外部イベントがレコードを削除する必要があると指定している場合に、Service Manager が実行するイベントです。
  - **レコードの削除**：(デフォルト) Service Manager は自動的に CI レコードを削除します。
  - **インシデントのオープン**：Service Manager は、削除される CI レコードを調査するためにインシデントを開き、レコードを Service Manager に準拠させるために実行や開始が必要となるアクションを判断します。
  - **変更のオープン**：Service Manager は、削除される CI レコードを確認させるため、計画外の変更をオープンします。この変更により、ユーザは削除される CI レコードがビジネスプラクティスに準拠しているかどうかを調査できます。CI レコードが準拠している場合、変更を承認できます。CI レコードが準拠していない場合、変更を却下して CI レコードをシステムに戻すことができます。

## 変更とインシデントレコードに表示されるCI属性

Service Manager では、UCMDB インテグレーションを通じて CI 属性の変更を検出したときに、変更レコードやインシデントレコードを開くように DEM を設定すると、その変更の [変更の詳細] タブ、またはそのインシデントの [CMDB 変更] タブが表示されます。Service Manager では、UCMDB インテグレーションが有効であり、検出イベントマネージャで、CI の追加、更新、または削除の際に変更レコードやインシデントレコードを作成するルールを定義している場合のみ、CI 属性のタブが表示されます。

[変更の詳細] タブと [CMDB 変更] タブの両方に、UCMDB によって検出された実際の属性値と一緒に現在の CI 属性値が表示されます。この情報を使用して、変更の承認や却下を行ったり、適切な担当グループにインシデントをエスカレートできます。

## インテグレーションによってオープンした変更とインシデントレコードの検索

次の検索基準を用いて、UCMDB インテグレーションによってオープンされた変更とインシデントレコードを検索できます。

表 5 変更とインシデントレコードに利用できる検索オプション

| レコードタイプ | 利用できる検索オプション                                 |
|---------|--|
| 変更      | カテゴリが「 <b>unplanned change</b> 」のレコードを検索します。 |
| インシデント  | [UCMDB 統合により生成] オプションを使用してレコードを検索します。        |

## Service Manager への UCMDB CI の転送

インテグレーションは、Service Manager システムに CI を入力するため、UCMDB から Service Manager に CI の 1 回限りの転送を必要とします。UCMDB が新規 CI や新規属性値を検出すると、インテグレーションによって Service Manager 内の CI の一覧が更新されます。インテグレーションでは、UCMDB システムのレプリケーションジョブを使用して、CI データの転送が行われます。HP では、CI と CI 属性を最新に保つため、これらのジョブのスケジュールを設定することを推奨します。

**タスク 1: UCMDB と Service Manager 間の変更データのプッシュとレプリケーションジョブを定義します。**

UCMDB 9.0 での変更データのプッシュの定義 (40 ページ) を参照してください。

UCMDB 7.52 ~ 8.x での変更レプリケーションジョブの定義 (41 ページ) を参照してください。

**タスク 2: UCMDB と Service Manager 間の RMI データのプッシュとレプリケーションジョブを定義します。**

UCMDB 9.0 での RMI データのプッシュの定義 (42 ページ) を参照してください。

UCMDB 7.52 ~ 8.x での RMI レプリケーションジョブの定義 (43 ページ) を参照してください。

**タスク 3: CI と CI 属性を最新に保つため、データプッシュとレプリケーションジョブのスケジュールを設定します。**

データプッシュ、またはレプリケーションジョブのスケジュール (44 ページ) を参照してください。

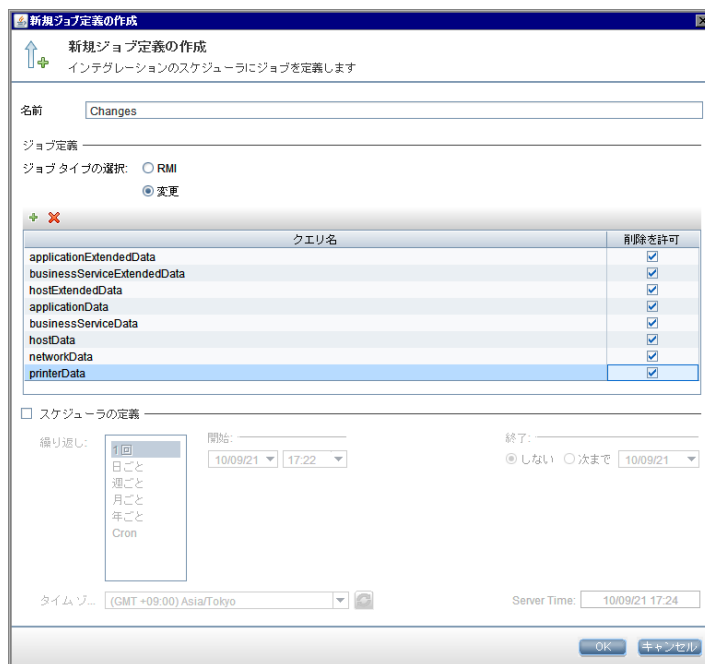
## UCMDB 9.0 での変更データのプッシュの定義

変更データのプッシュジョブは、UCMDB システムから Service Manager システムにすべての CI をコピーします。

- 1 管理者として UCMDB システムにログインします。
- 2 [マネージャ] > [データ フロー管理] > [Integration Studio] をクリックします。UCMDB に、インテグレーションポイントのリストが表示されます。
- 3 [データ プッシュ] タブを選択します。
- 4 Service Manager 向けに作成したインテグレーションポイントを選択します。「SM Integration」などです。
- 5 [+] をクリックして、新規データプッシュジョブを追加します。 
- 6 [名前] にジョブの一意的な名を入力します。「Chnages」などです。
- 7 [ジョブ タイプの選択] で、[変更] を選択します。



- 8 [Root] > [Integration] > [SM Query] から次のクエリを追加します。
- applicationExtendedData
  - businessServiceExtendedData
  - hostExtendedData
- 9 [Root] > [Integration] > [SM Sync] から次のクエリを追加します。
- applicationData
  - businessServiceData
  - hostData
  - networkData
  - printerData
- 10 各クエリで [削除を許可] オプションを選択します。



- 11 [OK] をクリックします。


## UCMDB 7.52～8.x での変更レプリケーションジョブの定義

変更データのレプリケーションジョブは、UCMDBシステムから Service ManagerシステムにすべてのCIをコピーします。

- 1 管理者としてUCMDBシステムにログインします。
- 2 [設定 - フェデレート CMDB] に移動します。
- 3 [レプリケーション ジョブ] タブを選択します。
- 4 [+] をクリックして、新規レプリケーションジョブを追加します。
- 5 [名前] にジョブの一意的な名を入力します。「cmdb2sm」などです。

- 6 [ソース データ ストア] で、**新規変更アダプタデータストアの追加** (25ページ) で作成した変更データストア名を選択します。「**CmdbChanges**」などです。
- 7 [ターゲット データ ストア] で、**Service Deskアダプタデータストアの追加** (28ページ) で作成したService Deskデータストア名を選択します。「**sm**」などです。
- 8 変更レプリケーションジョブに関係クエリが含まれないこと、[ターゲットでの削除を許可] オプションが有効であることを確認します。



- 9 [OK] をクリックして、レプリケーションジョブを保存します。
- 10 [一時的なフルレプリケーション] をクリックして、レプリケーションジョブを実行します。 


## UCMDB 9.0でのRMIデータのプッシュの定義

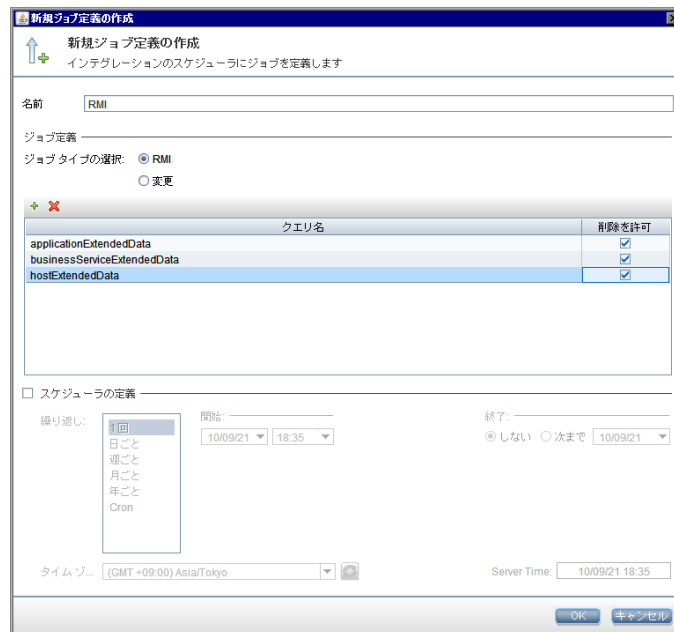
RMIデータのプッシュジョブは、UCMDBシステムからService Managerシステムにすべての関係データをコピーします。



Service ManagerでCI関係レコードを作成するには、有効なCIデータが必要です。RMIデータプッシュジョブの実行前に、少なくとも1回変更データプッシュジョブを実行する必要があります。

- 1 管理者としてUCMDBシステムにログインします。
- 2 [マネージャ] > [データ フロー管理] > [Integration Studio] をクリックします。UCMDBに、インテグレーションポイントのリストが表示されます。
- 3 [データ プッシュ] タブを選択します。
- 4 Service Manager向けに作成したインテグレーションポイントを選択します。「**SM Integration**」などです。
- 5 [+ ] をクリックして、新規データプッシュジョブを追加します。 

- 6 [名前] にジョブの一意的な名を入力します。「**RMI**」などです。
- 7 [ジョブタイプの選択] で、[**RMI**] を選択します。
- 8 [+ ] をクリックして、クエリ名を追加します。 
- 9 [**Root**] > [**Integration**] > [**SM Sync**] から次のクエリを追加します。
  - applicationRelationsData
  - businessServiceRelationsData
  - hostRelationsData
- 10 各クエリで [**削除を許可**] オプションを選択します。



新規ジョブ定義の作成

新規ジョブ定義の作成  
インテグレーションのスケジュールにジョブを定義します

名前

ジョブ定義

ジョブタイプの選択:  RMI  
 変更

| クエリ名                        | 削除を許可                               |
|-----------------------------|-------------------------------------|
| applicationExtendedData     | <input checked="" type="checkbox"/> |
| businessServiceExtendedData | <input checked="" type="checkbox"/> |
| hostExtendedData            | <input checked="" type="checkbox"/> |

スケジュールの定義

繰り返す:  開始:   終了:   しない  次まで

タイムゾーン:  Server Time:

OK キャンセル


- 11 [OK] をクリックします。

## UCMDB 7.52～8.xでのRMIレプリケーションジョブの定義

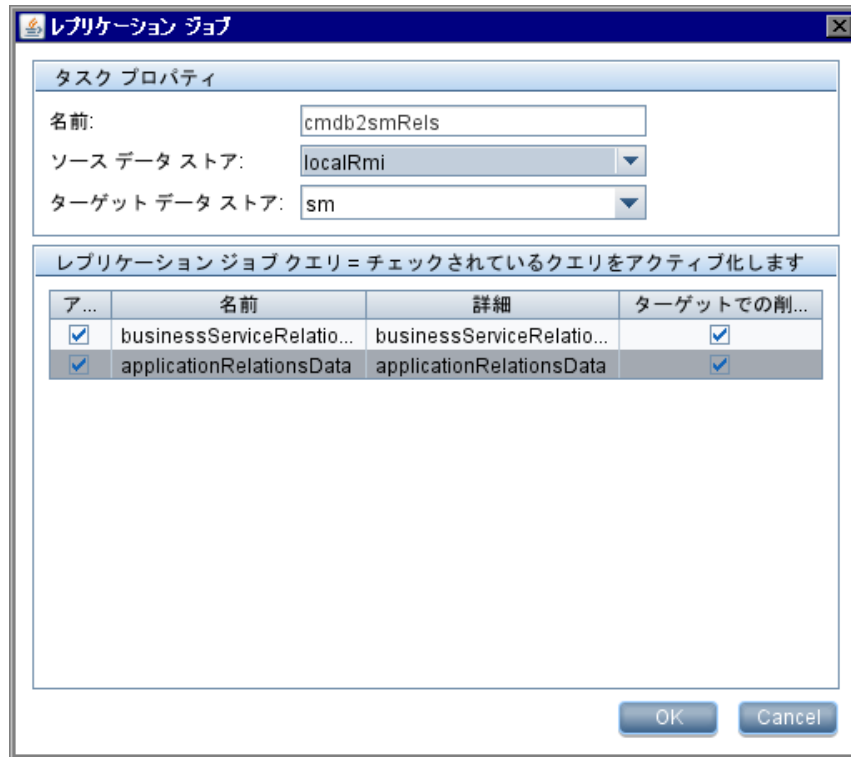
RMIレプリケーションジョブは、UCMDBシステムからService Managerシステムにすべての関係データをコピーします。




Service ManagerでCI関係レコードを作成するには、有効なCIデータが必要です。RMIレプリケーションジョブの実行前に、少なくとも1回変更レプリケーションジョブを実行する必要があります。

- 1 管理者としてUCMDBシステムにログインします。
- 2 [設定 - フェデレート CMDB] に移動します。
- 3 [レプリケーション ジョブ] タブを選択します。
- 4 [+ ] をクリックして、新規レプリケーションジョブを追加します。 
- 5 [名前] にジョブの一意的な名を入力します。「**cmdb2smRe1s**」などです。
- 6 [ソース データ ストア] で、**RMIアダプタデータストアの追加** (23ページ) で作成したRMIデータストア名を選択します。For example, **localRmi**.

- 7 [ターゲット データ ストア] で、Service Deskアダプタデータストアの追加 (28ページ) で作成した Service Deskデータストア名を選択します。「sm」などです。
- 8 変更レプリケーションジョブに関係クエリのみが含まれること、[ターゲットでの削除を許可] オプションが有効であることを確認します。



- 9 [OK] をクリックして、レプリケーションジョブを保存します。

- 10 [一時的なフルレプリケーション] をクリックして、レプリケーションジョブを実行します。 


## データプッシュ、またはレプリケーションジョブのスケジュール

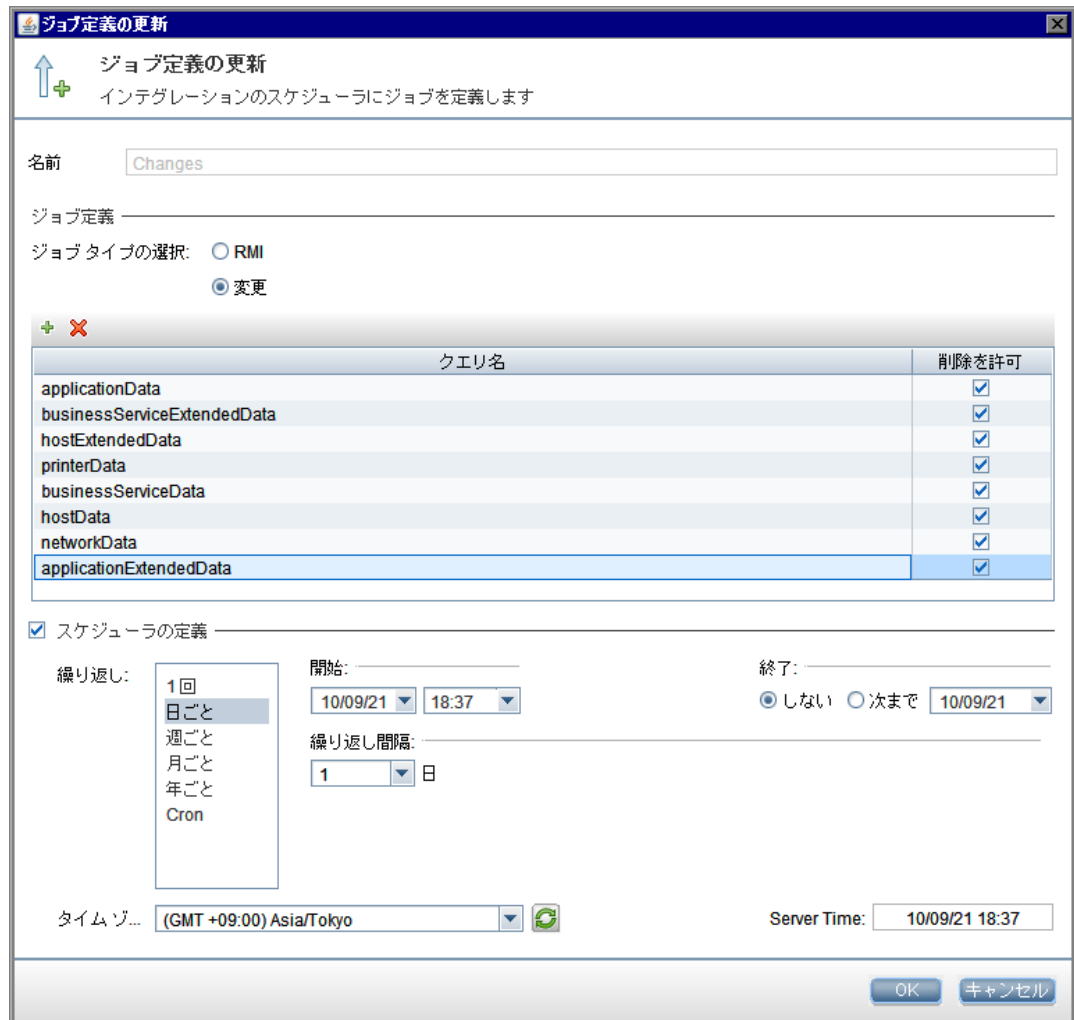
UCMDB フィーダのディスカバリスケジュールと一致するように、変更と RMI の両方のデータプッシュやレプリケーションジョブをスケジュール設定することが重要です。たとえば、UCMDB フィーダが日次スケジュールで CI データの更新を送信する場合は、データプッシュジョブも日次スケジュールで実行する必要があります。一致するスケジュールを用いることにより、Service Manager システムが常に最新の CI データを保持できるようになります。

### UCMDB 9.0 データプッシュジョブのスケジュール

UCMDB 9.0 では、データプッシュジョブからスケジュールを直接更新できます。

- 1 管理者として UCMDB 9.00 システムにログインします。
- 2 [マネージャ] > [データ フロー管理] > [Integration Studio] をクリックします。UCMDB に、インテグレーションポイントのリストが表示されます。
- 3 UCMDB と Service Manager のインテグレーション用に作成したインテグレーションポイントを選択します。「SM Integration」などです。

- 4 [データ プッシュ] タブを選択します。
- 5 インテグレーション用に作成した変更データプッシュジョブを選択します。「Changes」などです。
- 6 [編集] ボタンをクリックします。 
- 7 [スケジューラの定義] オプションを有効にします。
- 8 使用するスケジュールオプションを選択します。たとえば、[繰り返し: 日ごと]、および [終了: しない] を選択します。



ジョブ定義の更新  
インテグレーションのスケジューラにジョブを定義します

名前: Changes

ジョブ定義

ジョブタイプの選択:  RMI  変更

| クエリ名                        | 削除を許可                               |
|-----------------------------|-------------------------------------|
| applicationData             | <input checked="" type="checkbox"/> |
| businessServiceExtendedData | <input checked="" type="checkbox"/> |
| hostExtendedData            | <input checked="" type="checkbox"/> |
| printerData                 | <input checked="" type="checkbox"/> |
| businessServiceData         | <input checked="" type="checkbox"/> |
| hostData                    | <input checked="" type="checkbox"/> |
| networkData                 | <input checked="" type="checkbox"/> |
| applicationExtendedData     | <input checked="" type="checkbox"/> |

スケジューラの定義

繰り返し:   日ごと  週ごと  月ごと  年ごと  Cron

開始: 10/09/21 18:37

終了:  しない  次まで 10/09/21

繰り返し間隔: 1 日

タイムゾーン: (GMT +09:00) Asia/Tokyo



Server Time: 10/09/21 18:37

OK キャンセル

- 9 [OK] をクリックして、データプッシュジョブを保存します。
- 10 インテグレーション用に作成したRMIデータプッシュジョブを選択します。「RMI」などです。
- 11 RMIデータプッシュジョブについて、手順6から手順9を繰り返します。

## UCMDB 7.52 と 8.x のレプリケーションジョブのスケジュール

旧バージョンの UCMDB では、レプリケーションジョブ定義自体とは別にレプリケーションのスケジュールを設定します。

- 1 管理者として HP Universal CMDB システムにログインします。
- 2 **【設定】** タブを選択します。
- 3 **【スケジューラ】** をクリックします。[ジョブ スケジューラ] フォームが開きます。
- 4 **【+】** をクリックして、新規スケジュールジョブを追加します。 
- 5 [名前] にジョブの一意な名を入力します。「**changes\_schedule**」などです。
- 6 [アクション] で、**【+】** をクリックして新規スケジュールアクションを追加します。 
- 7 [アクションを選択] リストから **【レプリケーション ジョブを実行】** を選択して、**【次へ】** をクリックします。
- 8 [レプリケーション ジョブを実行] リストから、スケジュールを設定するレプリケーションジョブを選択します。「**cmdb2sm**」、および「**cmdb2smBusinessRels**」などです。
- 9 **【完了】** を選択します。UCMDB では、アクションリストにアクションが表示されます。
- 10 [スケジューラ] の選択で、アクションの頻度を選択します。「**日ごと**」などです。
- 11 [タイムゾーン] で、UCMDB システムのタイムゾーンを確認します。
- 12 起動時刻として、ジョブを実行する時刻を入力します。たとえば、「**24:00**」などです。
- 13 特定の指定した日付間でのみスケジュールされたアクションを実行する場合、[開始時刻] と [終了時刻] オプションを選択し、スケジュールされたジョブが有効になる日付を選択します。
- 14 **【OK】** をクリックして、スケジュールされたジョブを作成します。

## 3 インテグレーションのカスタマイズ

HP Universal CMDBとHP Service Managerのインテグレーションをカスタマイズすることで、管理CIタイプと属性の追加や削除を行い、ビジネスニーズに合わせるすることができます。本章では、次のカスタマイズトピックについて説明します。

- [インテグレーションのアーキテクチャ](#) (48ページ)
- [UCMDBでのインテグレーションコンポーネントの表示](#) (63ページ)
- [カスタマイズのオプション](#) (67ページ)
- [インテグレーションへのCI属性の追加](#) (67ページ)
- [インテグレーションへのCIタイプの追加](#) (81ページ)

# インテグレーションのアーキテクチャ

インテグレーションをカスタマイズする前に、出荷時のインテグレーションに含まれる次のコンポーネントがどのように機能するのかを把握しておく必要があります。

- インテグレーションのクラスモデル (48ページ)
- インテグレーションTQLクエリ (51ページ)
- Service Manager Web サービス (54ページ)
- Service Manager 調停ルール (58ページ)
- Service Manager 検出イベントマネージャールール (59ページ)

## インテグレーションのクラスモデル

### UCMDB 9.0

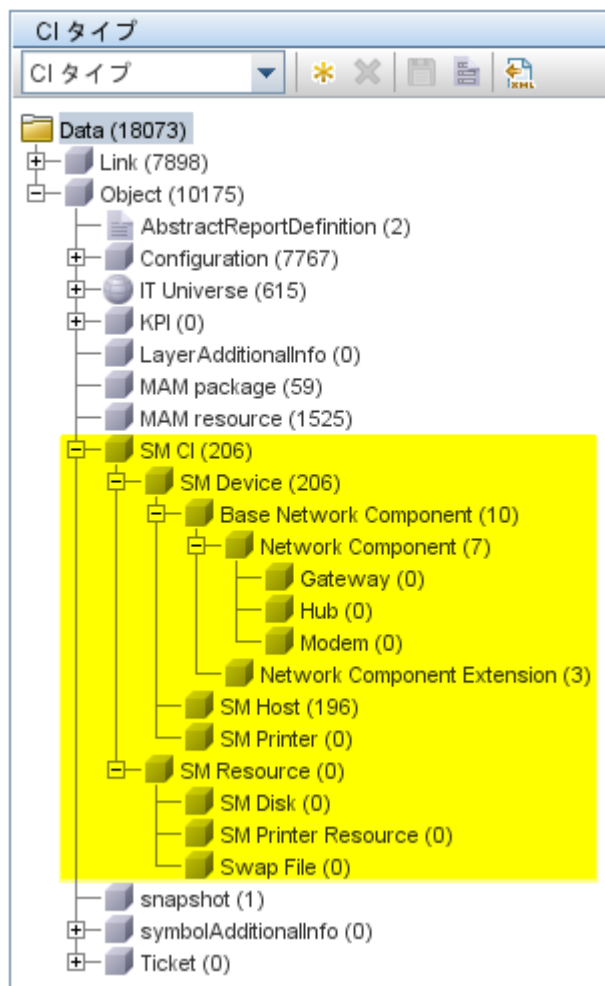
UCMDB 9.0では、インテグレーションCIを管理するのに、CIのプライベートクラスモデルは使用しません。代わりに標準的なUCMDB管理オブジェクトを使用して、それらのオブジェクトをクエリと変換ファイルでService Manager CIタイプと属性にマップします。

### UCMDB 7.52 と 8.x

旧バージョンのインテグレーションには、Service Managerでの管理方法に一致するUCMDB CIタイプ、関係、および属性のプライベートクラスモデルが含まれます。通常、Service ManagerはUCMDBよりも少ないCIタイプしか認識しません。たとえば、Service Managerは、TCP/IPポートなどの特定のUCMDB CIタイプを、独立したCIタイプではなくCI属性として管理します。

UCMDBシステムがService Managerシステムに送信するCIを変更する場合、インテグレーションクラスモデルとモデルをサポートするインテグレーションクエリの両方を編集する必要があります。出荷時設定では、インテグレーションクラスモデルは、次のCIタイプと関係から構成されています。





このバージョンのインテグレーションでは、UCMDBは、IT Universeとは別にService Manager CIを追跡します。インテグレーションのクラスモデルを参照するには、IT Universeのデフォルト値からオブジェクトルートを変更する必要があります。

出荷時設定のUCMDB 7.52と8.xには、インテグレーションに次のCIタイプが用意されています。

表 6 インテグレーションCIタイプ

| CIタイプ                       | 説明   |
|-----------------------------|--|
| SM CI                       | このCIタイプは、出荷時設定のUCMDB インテグレーションで用意されている、すべてのCIタイプと関係のためのコンテナです。このCIタイプは、Object CIタイプから属性を継承し、SM DeviceおよびSM Resource CIタイプのための属性を提供します。                                 |
| SM Device                   | このCIタイプは、Service Managerでデバイスとして管理されているCIのためのコンテナです。このCIタイプは、SM CIタイプから属性を継承し、Base Network Component、SM Host、およびSM Printer CIタイプのための属性を提供します。                          |
| Base Network Component      | このCIタイプは、ネットワークデバイスCIのためのコンテナです。このCIタイプは、SM Device CIタイプから属性を継承し、Network ComponentおよびNetwork Component Extension CIタイプのための属性を提供します。                                    |
| Network Component           | このCIタイプは、ネットワークデバイスCIのためのコンテナです。このCIタイプは、Base Network Component CIタイプから属性を継承し、Gateway、Hub、およびModem CIタイプのための属性を提供します。このCIタイプはインテグレーションに固有であり、UCMDBパブリッククラスモデルには存在しません。 |
| Gateway                     | これは、ネットワークゲートウェイのためのCIタイプです。このCIタイプは、Network Component CIタイプの属性を継承します。このCIタイプはインテグレーションに固有であり、UCMDBパブリッククラスモデルには存在しません。  |
| Hub                         | これは、ネットワークハブのためのCIタイプです。このCIタイプは、Network Component CIタイプの属性を継承します。このCIタイプはインテグレーションに固有であり、UCMDBパブリッククラスモデルには存在しません。  |
| Modem                       | これは、ネットワークモデムのためのCIタイプです。このCIタイプは、Network Component CIタイプの属性を継承します。このCIタイプはインテグレーションに固有であり、UCMDBパブリッククラスモデルには存在しません。   |
| Network Component Extension | これは、ネットワークコンポーネントCIのためのCIタイプです。このCIタイプは、Base Network Component CIタイプの属性を継承し、さらに、SMリンク関係を使用してHost CIタイプの属性を拡張します。   |
| SM Host                     | これは、ホストのためのCIタイプです。このCIタイプは、SM Device CIタイプの属性を継承し、さらに、SMリンク関係を使用してHost CIタイプの属性を拡張します。  |

表 6 インテグレーションCIタイプ

| CIタイプ               | 説明  |
|---------------------|---|
| SM Printer          | このCIタイプはプリンタCIを管理します。このCIタイプは、SM Device CIタイプの属性を継承し、さらに、SMリンク関係を使用してNet Printer CIタイプの属性を拡張します。  |
| SM Resource         | このCIタイプは、Service Managerでリソースとして管理されているCIのためのコンテナです。このCIタイプは、SM CIタイプから属性を継承し、SM Printer Resource、SM Disk、およびSwap File CIタイプのための属性を提供します。         |
| SM Disk             | このCIタイプはハードディスクCIを管理します。このCIタイプは、SM Resource CIタイプの属性を継承し、さらに、SMリンク関係を使用してDisk CIタイプの属性を拡張します。  |
| SM Printer Resource | このCIタイプはプリンタCIを管理します。このCIタイプは、SM Resource CIタイプの属性を継承し、さらに、SMリンク関係を使用してPrinter CIタイプの属性を拡張します。  |
| Swap File           | このCIタイプは、スワップファイルメモリCIを管理します。このCIタイプは、SM Resource CIタイプの属性を継承します。さらに、Host CIタイプへのコンテナリンク関係も備えます。このCIタイプはインテグレーションに固有であり、UCMDBパブリッククラスモデルには存在しません。 |

## インテグレーションTQLクエリ

インテグレーションは、UCMDBからCI属性情報を収集し、それをService Managerシステムに送信するのに、トポロジクエリ言語 (TQL) クエリのコレクションを使用します。これらのクエリはUCMDBインストールの **Integration\SM Sync** フォルダにあります。インテグレーションに含めるCIタイプや属性を変更する場合、更新されたCIタイプと属性をサポートするインテグレーションクエリも編集する必要があります。出荷時設定では、インテグレーションには次のクエリが含まれています。

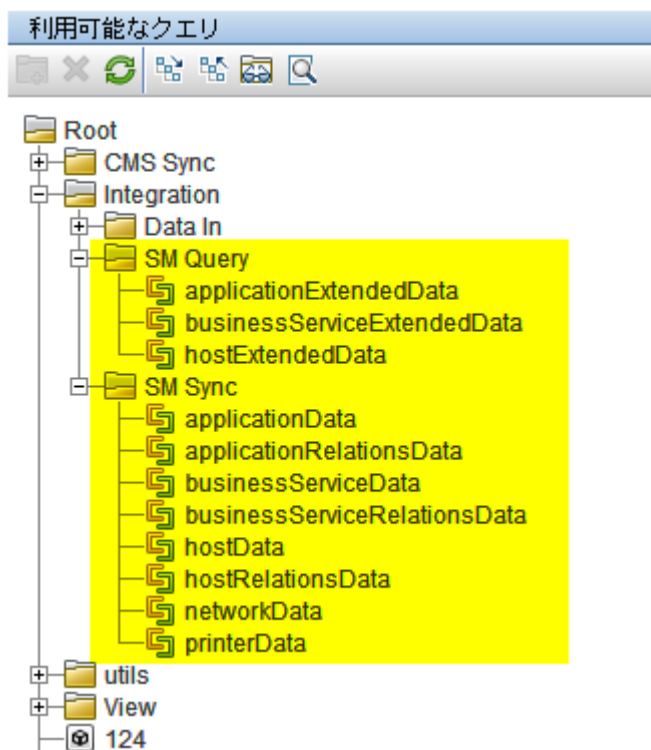


表7 インテグレーションクエリ

| クエリ名                        | 説明   |
|-----------------------------|--|
| applicationData             | このクエリは、アプリケーションCIからCI属性を収集します。   |
| applicationExtendedData     | このクエリは、アプリケーションCIからのリアルタイム拡張情報を収集します。  |
| applicationRelationsData    | このクエリは、次のコンポーネント間の関係を収集します： <ul style="list-style-type: none"> <li>アプリケーションとホストCI</li> <li>アプリケーションとネットワークコンポーネントCI</li> <li>複数のアプリケーションCI</li> </ul> グループを通じて関係が拡張するため、クエリには複合関係が含まれます。 |
| businessServiceData         | このクエリは、ビジネスサービスCIからCI属性を収集します。   |
| businessServiceExtendedData | このクエリは、ビジネスサービスCIからのリアルタイム拡張情報を収集します。  |

表7 インテグレーションクエリ

| クエリ名                         | 説明  |
|------------------------------|---|
| businessServiceRelationsData | <p>このクエリは、次のコンポーネント間の関係を収集します：</p> <ul style="list-style-type: none"> <li>• ビジネスサービスとアプリケーションCI</li> <li>• ビジネスサービスとホストCI</li> <li>• ビジネスサービスとネットワークコンポーネントCI</li> <li>• 複数のビジネスサービス</li> </ul> <p>グループを通じて関係が拡張するため、クエリには複合関係が含まれます。</p>   |
| hostData                     | <p>このクエリは、次のホストCIタイプからCI属性を収集します：</p> <ul style="list-style-type: none"> <li>• <b>UCMDB 9.00</b> : node</li> <li>• <b>UCMDB 7.52 と 8.x</b> : host、host_node、lpar、mainframe、nt、terminalserver、unix、vax</li> </ul> <p>また、コンテナとリンクを経由して、次のCIタイプからも関連CI属性を収集します：IP、Interface、Memory、SM Host</p> |
| hostExpandedData             | <p>このクエリは、次のCIタイプからリアルタイム拡張情報（Asset、Person、WindowsService、Printer、InstalledSoftware、Cpuなど）を収集します：</p> <ul style="list-style-type: none"> <li>• <b>UCMDB 9.00</b> : node</li> <li>• <b>UCMDB 7.52 と 8.x</b> : host、host_node、lpar、mainframe、nt、terminalserver、unix、vax</li> </ul>            |
| hostRelationsData            | <p>このクエリは、次のコンポーネント間の関係を収集します：</p> <ul style="list-style-type: none"> <li>• ホストとインタフェースCI</li> <li>• ホストとIP CI</li> <li>• ホストとネットワークコンポーネントCI</li> <li>• 複数のホストCI</li> </ul> <p>関係のルートクラスはtalkです。</p>   |
| networkData1                 | <p>このクエリは、次のホストCIタイプからCI属性を収集します：atmswitch、marconiatmswitch、ras、lb、concentrator、netdevice、firewall、switch、switchrouter、router</p> <p>また、コンテナとリンクを経由して、次のCIタイプからも関連CI属性を収集します：Interface、IP、Network Component Extension</p>   |

表7 インテグレーションクエリ

| クエリ名                 | 説明   |
|----------------------|--|
| networkData2         | このクエリは、ネットワークコンポーネントCIからCI属性を収集します。<br>また、コンテナとリンクを経由して、次のCIタイプからも関連CI属性を収集します：Interface、IP  |
| networkRelationsData | このクエリは、次のコンポーネント間の関係を収集します： <ul style="list-style-type: none"> <li>ネットワークコンポーネントとホストCI</li> <li>ネットワークコンポーネントとインタフェースCI</li> <li>ネットワークコンポーネントとIP CI</li> <li>複数のネットワークコンポーネントCI</li> </ul> 関係のルートクラスCIタイプはtalkです。 |
| printerData          | このクエリは、ネットプリンタCIからCI属性を収集します。<br>また、コンテナとリンクを経由して、次のCIタイプからも関連CI属性を収集します：Interface、IP、SM Printer   |

## TQLクエリの要件

インテグレーションでは、ユーザが作成するカスタムTQLクエリがフォーマット条件を満たすことが必要です。インテグレーションに含めるどのTQLクエリもこれらの条件を満たす必要があります：

- CIをクエリするには、クエリに**Root**という名前の1つのCIタイプを含める必要があります。このRootノードがUCMDBが同期するメインCIです。その他のCIはすべて、Root CIに含まれます。
- 関係をクエリするには、クエリに**Root**という名前の1つまたは複数の関係を含める必要があります。
- クエリには、Root CIと、Root CIに直接接続しているCIのみを含める必要があります。Root CIは、常にTQL階層の最上位ノードです。
- TQLグラフには循環を含めることはできません。
- 関係を同期するクエリにカーディナリティがある場合、そのカーディナリティは**1...\***である必要があります。カーディナリティエントリを追加する場合、エントリ間はOR条件で接続する必要があります。
- CMDB変更アダプタでは、複合関係はサポートされません。
- インテグレーションで特定CIのみを同期する場合、CI自体に条件を設定してから、TQLクエリに同じ条件を設定する必要があります。

## Service Manager Web サービス

Service Managerは、Web サービスメッセージを使用してUCMDBシステムからCI情報を取得して受信します。出荷時設定のUCMDBは、Service Managerシステムが実際に管理するよりも多くのCI属性情報を送信します。UCMDBシステムが提供するCI属性情報の多くは参照専用です。Service Managerユーザは、CIレコードの[実際のステータス] タブで、UCMDBシステムが送信するCI属性情報をすべて確認できます。

Service Manager は、UCMDB インテグレーションが使用するための Web サービスをいくつか公開します。UCMDB システムは、Web サービスを使用して、UCMDB CI タイプと CI 属性を Service Manager システムが認識する Web サービスオブジェクトにマップします。Service Manager が管理する UCMDB CI タイプや CI 属性を追加する場合、これらの Web サービスの 1 つまたは複数を更新して、Web サービスオブジェクトとして定義する必要があります。Web サービスの公開の詳細については、Service Manager ヘルプを参照してください。

## 管理フィールド

Service Manager 管理フィールドとは、受信する UCMDB Web サービスメッセージ内の CI 属性値と、Service Manager CI レコード内の値とを比較するフィールドのことです。Web サービスメッセージ内の値が CI レコード内の値と一致しない場合、Service Manager は検出イベントマネージャ (DEM) ルールを実行して、実行するアクションを決定します。

Service Manager は、**ucmdbIntegration** Web サービスの管理フィールドの一覧を格納します。ucmdbIntegration Web サービスは、次の Web サービスオブジェクトから構成されます。

表 8 Web サービスオブジェクト、Service Manager テーブル、および DEM ルール間のマッピング

| Web サービスオブジェクト       | フィールドの公開元の Service Manager テーブル | 使用する DEM ルール ID      |
|----------------------|---------------------------------|----------------------|
| Relationship         | cirelationship                  | ucmdbRelationship    |
| ucmdbApplication     | device                          | ucmdbApplication     |
| ucmdbBusinessService | joinbizservice                  | ucmdbBusinessService |
| ucmdbComputer        | joincomputer                    | ucmdbComputer        |
| ucmdbNetwork         | joinnetworkcomponents           | ucmdbNetwork         |
| ucmdbPrinter         | joinofficeelectronics           | ucmdbPrinter         |

DEM ルールは、Web サービスオブジェクトとして公開され、インテグレーションにより管理されるフィールドを決定します。管理フィールドの値を変更するだけで、DEM ルールがトリガされます。

次のセクションは、Web サービスオブジェクトとして公開されるフィールドと、それらのフィールドが出荷時設定の Service Manager システムで管理フィールドであるかどうかを示す一覧です。このリファレンスを参考にして、Web サービスオブジェクトとしてフィールドを公開する必要があるかどうか、また、オブジェクトの DEM ルールを作成する必要があるかどうかを判断できます。

### オブジェクト名 : Relationship

Service Manager は、cirelationship テーブルの次のフィールドを管理します：

表 9 Relationship オブジェクトの Web サービスと管理フィールド

| Web サービスオブジェクトとして公開されるフィールド | Web サービスメッセージで使用されるキャプション | フィールドは管理フィールドである |
|-----------------------------|---------------------------|------------------|
| relationship.name           | RelationshipName          |                  |
| logical.name                | ParentCI                  |                  |

表9 RelationshipオブジェクトのWebサービスと管理フィールド

| Webサービスオブジェクトとして公開されるフィールド | Webサービスメッセージで使用されるキャプション | フィールドは管理フィールドである |
|----------------------------|--------------------------|------------------|
| related.cis                | ChildCIs                 | はい               |
| relationship.type          | RelationshipType         |                  |
| relationship.subtype       | RelationshipSubtype      |                  |

オブジェクト名 : ucmdbApplication

Service Managerは、deviceテーブルの次のフィールドを管理します :

表10 ucmdbApplicationオブジェクトのWebサービスと管理フィールド

| Webサービスオブジェクトとして公開されるフィールド | Webサービスメッセージで使用されるキャプション | フィールドは管理フィールドである |
|----------------------------|--------------------------|------------------|
| ucmdb.id                   | UCMDBId                  |                  |
| ci.name                    | ApplicationName          | はい               |
| subtype                    | Subtype                  |                  |

オブジェクト名 : ucmdbBusinessService

Service Managerは、joinbizserviceテーブルの次のフィールドを管理します :

表11 ucmdbBusinessServiceオブジェクトのWebサービスと管理フィールド

| Webサービスオブジェクトとして公開されるフィールド | Webサービスメッセージで使用されるキャプション | フィールドは管理フィールドである |
|----------------------------|--------------------------|------------------|
| ucmdb.id                   | UCMDBId                  |                  |
| ci.name                    | ServiceName              | はい               |
| vendor                     | ServiceProvider          | はい               |
| subtype                    | Subtype                  | はい               |

オブジェクト名 : ucmdbComputer

Service Managerは、joincomputerテーブルの次のフィールドを管理します :

表12 ucmdbComputerオブジェクトのWebサービスと管理フィールド

| Webサービスオブジェクトとして公開されるフィールド | Webサービスメッセージで使用されるキャプション | フィールドは管理フィールドである |
|----------------------------|--------------------------|------------------|
| ucmdb.id                   | UCMDBId                  |                  |
| default.gateway            | DefaultGateway           | はい               |
| operating.system           | OS                       | はい               |
| os.version                 | OSVersion                | はい               |
| bios.id                    | BIOSId                   | はい               |



表 12 ucldbComputer オブジェクトの Web サービスと管理フィールド

| Web サービスオブジェクトとして公開されるフィールド | Web サービスメッセージで使用されるキャプション | フィールドは管理フィールドである |
|-----------------------------|---------------------------|------------------|
| network.name                | DNSName                   | はい               |
| addIIPAddr[addIIPAddress]   | AddIIPAddress             | はい               |
| addIIPAddr[addISubnet]      | AddISubnet                | はい               |
| addIMacAddress              | AddIMacAddress            | はい               |
| physical.mem.total          | PhysicalMemory            | はい               |
| subtype                     | Subtype                   | はい               |
| building                    | Building                  | はい               |
| room                        | Room                      | はい               |
| floor                       | Floor                     | はい               |
| machine.name                | HostName                  | はい               |

オブジェクト名 : ucldbNetwork

Service Manager は、joinnetworkcomponents テーブルの次のフィールドを管理します :

表 13 ucldbNetwork オブジェクトの Web サービスと管理フィールド

| Web サービスオブジェクトとして公開されるフィールド | Web サービスメッセージで使用されるキャプション | フィールドは管理フィールドである |
|-----------------------------|---------------------------|------------------|
| ucldb.id                    | UCMDBId                   |                  |
| default.gateway             | DefaultGateway            | はい               |
| operating.system            | OS                        | はい               |
| os.version                  | OSVersion                 | はい               |
| network.name                | DNSName                   | はい               |
| addIIPAddr[addIIPAddress]   | AddIIPAddress             | はい               |
| addIIPAddr[addISubnet]      | AddISubnet                | はい               |
| addIMacAddress              | AddIMacAddress            | はい               |
| subtype                     | Subtype                   | はい               |
| building                    | Building                  | はい               |
| room                        | Room                      | はい               |
| floor                       | Floor                     | はい               |

オブジェクト名 : ucmdbPrinter

Service Managerは、joinofficeelectronics テーブルの次のフィールドを管理します :

表 14 ucmdbPrinter オブジェクトの Web サービスと管理フィールド

| Web サービスオブジェクトとして公開されるフィールド | Web サービスメッセージで使用されるキャプション | フィールドは管理フィールドである |
|-----------------------------|---------------------------|------------------|
| ucmdb.id                    | UCMDBid                   |                  |
| default.gateway             | DefaultGateway            | はい               |
| network.name                | DNSName                   | はい               |
| addlIPAddr[addlIPAddress]   | AddlIPAddress             | はい               |
| addlIPAddr[addlSubnet]      | AddlSubnet                | はい               |
| addlMacAddress              | AddlMacAddress            | はい               |
| subtype                     | Subtype                   | はい               |
| building                    | Building                  | はい               |
| room                        | Room                      | はい               |
| floor                       | Floor                     | はい               |

## Service Manager 調停ルール

Service Manager 調停ルールにより、インテグレーションはUCMDBシステム内のCIに一致する Service Manager システム内の CIレコードを特定できます。Service Managerは、UCMDBシステムからのCI属性の各プッシュでCIレコードを調整します。インテグレーションは、次のワークフローを用いてUCMDB CIを Service Manager CIと照合します。

- 1 UCMDBシステムは、Service Managerに最新のCI属性データを含むWebサービスメッセージを送信します。
- 2 Service Managerは、WebサービスメッセージをスキャンしてCI ucmdb.id値を取得します。
  - ▶ 出荷時設定の Service Manager では、ユーザが値を変更できないようにするため、CIレコードフォームにucmdb.idフィールドは表示されません。フォームにこの値を追加するには、device テーブルで定義されている umcdb.idフィールドを検索します。HPでは、このフィールドを読み取り専用フィールドにしておくことを推奨します。
- 3 Service Managerは、同じ umcdb.id値の既存のCIレコードを検索します。
- 4 ucmdb.id値が一致するCIが見つかった場合は、調整は必要ありません。Service ManagerはUCMDB CI属性と Service Manager管理フィールドとを比較し、必要に応じて適切な検出イベントマネージャ (DEM) ルールを実行します。
- 5 ucmdb.id値が一致するCIが見つからなかった場合は、調停ルールを実行します。
- 6 Service Managerは、同じ調整フィールド値の既存のCIレコードを検索します。

- 調整フィールド値が一致するCIが見つかったら、Service Managerは一致するUCMDB CIのucmdb.id値でCIレコードを更新します。Service ManagerはUCMDB CI属性とService Manager管理フィールドとを比較し、必要に応じて適切な検出イベントマネージャ（DEM）ルールを実行します。
- 調整フィールド値が一致するCIが見つからなかった場合、Service Managerは「一致するレコードが存在しない場合のアクション」のDEMルールを実行します。出荷時設定のDEMルールでは、Service Managerは新規CIレコードを作成します。CIレコードの作成には、受信したUCMDB CIのucmdb.id値が使用されます。

## パフォーマンスについて

Service Managerは各プッシュでCIの調整を試みることから、調整フィールド数によりインテグレーションのパフォーマンスが影響を受けます。調停ルールが多いほど、Service ManagerがCIを照合するのに必要となる検索が多くなります。調整の検索パフォーマンスを改善するには、基礎となるService Managerテーブルの一意なキーである調整フィールドを選択する必要があります。たとえば、deviceテーブルのCIレコードを調整する場合、一意なキーであるlogical.nameフィールドを調整フィールドとして使用します。調停ルールを作成するには、[調停ルールの追加](#)（37ページ）を参照してください。

## DEMルールでの依存関係

Service Managerは、CIを調整できないときには常に「一致するレコードが存在しない場合のアクション」ルールを使用します。UCMDBからService Managerに最初のCIのプッシュを行う前に、DEM設定を確認し、DEM設定が会社でのビジネス基準を満たしていることを確認してください。たとえば、[変更のオープン] オプションを選択することで、最初のCIプッシュで各CIの変更依頼が作成されるようになります。

## Service Manager 検出イベントマネージャルール

検出イベントマネージャ（DEM）ルールを作成するだけで、次のカスタムアクションを実行できるようになります。

- DEMルールが実行される条件の変更（59ページ）
- テナント固有DEMルールの追加（60ページ）
- DEMルールが行うアクションの変更（61ページ）
- CIタイプの管理フィールドリストの更新（61ページ）
- 変更やインシデントレコードをオープンするカスタムJavaScriptの作成（61ページ）

## DEMルールが実行される条件の変更

Service Managerは、条件フィールドがtrueに評価される場合にのみDEMを実行します。出荷時設定では、ucmdbApplication DEMルールにのみ、ルールが実行される状況を限定する条件文があります。その他のインテグレーションDEMルールは、デフォルトで常に実行されます。

Service ManagerがDEMルールを実行する状況を限定するには、DEMルールの条件文を更新します。ucmdbComputer DEMルールに次の条件を追加すると、ルールはデスクトップCIにのみ限定されます。

```
subtype in $L.file=Desktop
```

条件フィールドを使用して、同じテーブル名に適用される複数のDEMルールを作成することもできます。たとえば、次のDEMルールは両方とも joincomputer テーブルに適用されます。

表 15 異なる条件を使用して、同じテーブルに影響を与える DEM ルール

| DEMルールID      | テーブル名        | 条件                           |
|---------------|--------------|------------------------------|
| ucmdbComputer | joincomputer | subtype in \$L.file!=Desktop |
| ucmdbDesktop  | joincomputer | subtype in \$L.file=Desktop  |

通常、条件を追加しなければならないのは、ビジネスプロセスによりインテグレーションで特定のCIタイプやSLAに異なるアクションが必要となる場合のみです。

## テナント固有DEMルールの追加

マルチテナント UCMDB インテグレーションでは、条件フィールドを使用することにより、特定のテナントに固有のDEMルールを作成できます。たとえば、Service Managerに直接CIを追加したいと考えるテナントもあれば、各CIについて変更をオープンしたいと考えるテナントもあります。次のサンプルDEMルールで、この方法を示します。

表 16 テナント固有DEMルール

| DEMルールID                | 新規CIに対するアクション | 条件                            |
|-------------------------|---------------|-------------------------------|
| ucmdbComputer_advantage | CIの追加         | company in \$L.file=advantage |
| ucmdbComputer_hp        | 変更の作成         | company in \$L.file=HP        |



各テナントごとに独立したDEMルールを作成することがベストプラクティスです。

## DEMルールが行うアクションの変更

出荷時設定では、インテグレーションDEMルールは次のアクションを実行します：

- CIレコードの追加（UCMDBデータが既存のService Manager CIレコードに一致しない場合）
- 変更のオープン（UCMDB CI属性データがService Manager CIレコードのCI属性データと一致しない場合）
- CIレコードの削除（UCMDBデータにCIが削除されたことが示されている場合）

ビジネスプロセスに合わせて、インテグレーションDEMルールを変更できます。たとえば、インテグレーションが予期しないデータのある非デスクトップUIを検出したときには、ucmdbComputer DEMルールを使用して変更をオープンするように設定したり、インテグレーションが予期しないデータのあるデスクトップUIを検出したときには、ucmdbDesktop DEMルールを使用して結果をログに記録し、レコードを更新するように設定することができます。



変更管理の検証とインテグレーションの変更管理の検証機能を使用する場合は、DEMルールは「レコードは存在するが、予期しないデータが見つかった場合のアクション」イベントに変更のオープンオプションを使用する必要があります。

## CIタイプの管理フィールドリストの更新

インテグレーションに含めるCI属性をUCMDBシステムに追加する場合は、Service Managerにも一致する管理フィールドを作成する必要があります。各管理フィールドには、UCMDBシステムからCI属性の更新を受信するために、対応するWebサービスオブジェクトの定義があります。管理フィールドの追加方法については、[インテグレーションへのCI属性の追加](#)（67ページ）と[インテグレーションへのCIタイプの追加](#)（81ページ）を参照してください。

## 変更やインシデントレコードをオープンするカスタムJavaScriptの作成

Service Managerが変更やインシデントレコードをオープンするときは、discoveryEvent JavaScriptを使用して、CI名を作成したり必須フィールドの値を設定します。出荷時設定では、スクリプトは次のデフォルト値を使用します。

### 新規CIを作成するためのデフォルト値

createCINameとpopulateNewCI関数を更新して、次のCIの値を設定できます。

表 17 新規CIを作成するのに使用するデフォルト値

| CI属性                | discoveryEventで定義されているデフォルト値 |
|---------------------|------------------------------|
| record.logical_name | システムが生成するID番号                |
| record.assignment   | AUTO                         |
| record.istatus      | Installed                    |
| record.os_name      | record.operating_systemの値    |

### 新規変更を作成するためのデフォルト値

populateChange関数を更新して、次の変更の値を設定できます。

表 18 新規変更を作成するのに使用するデフォルト値

| CI属性                  | discoveryEventで定義されているデフォルト値 |
|-----------------------|------------------------------|
| change.category       | Unplanned Change             |
| change.reason         | reasonの値                     |
| change.initial_impact | 3                            |
| change.severity       | 3                            |
| change.coordinator    | Change.Coordinator           |
| change.requested_by   | discovery                    |
| change.status         | initial                      |

### 新規インシデントを作成するためのデフォルト値

populateIncident関数を更新して、次のインシデントの値を設定できます。

表 19 新規インシデントを作成するのに使用するデフォルト値

| CI属性                    | discoveryEventで定義されているデフォルト値 |
|-------------------------|------------------------------|
| incident.category       | incident                     |
| incident.subcategory    | hardware                     |
| incident.product_type   | missing または stolen           |
| incident.assignment     | Hardware                     |
| incident.initial_impact | 3                            |
| incident.severity       | 3                            |
| incident.logical.name   | IDの値                         |
| incident.site_cateogry  | C                            |
| incident.contact_name   | ANALYST, INCIDENT            |
| incident.affected_item  | MyDevices                    |

# UCMDBでのインテグレーションコンポーネントの表示

出荷時設定のUCMDBインストールでは、インテグレーションコンポーネントはセットアップ後には表示されません。これは、インテグレーションコンポーネントがクラスモデル内で、デフォルトのビュー表示よりも高い位置にあるためです。インテグレーションを使用する場合は、インテグレーションコンポーネントを表示する必要はありませんが、インテグレーションをカスタマイズする場合は、インテグレーションコンポーネントを表示する必要があります。UCMDB インタフェースにインテグレーションコンポーネントを表示するには、次のタスクを実行する必要があります。

**タスク1:** リンクルート設定を [link] に変更します。

リンクルート設定の変更 (63ページ) を参照してください。

**タスク2:** オブジェクトルートを [data] に変更します。

オブジェクトルート設定の変更 (64ページ) を参照してください。

## リンクルート設定の変更

UCMDB 7.52 または 8.x システムを使用している場合、インテグレーションコンポーネントを表示するにはリンクルート設定を変更する必要があります。




### UCMDB 9.0


UCMDB 9.0 では、インテグレーションはプライベートリンクタイプは使用しなくなりました。インテグレーションリンクを表示するのに変更は必要ありません。

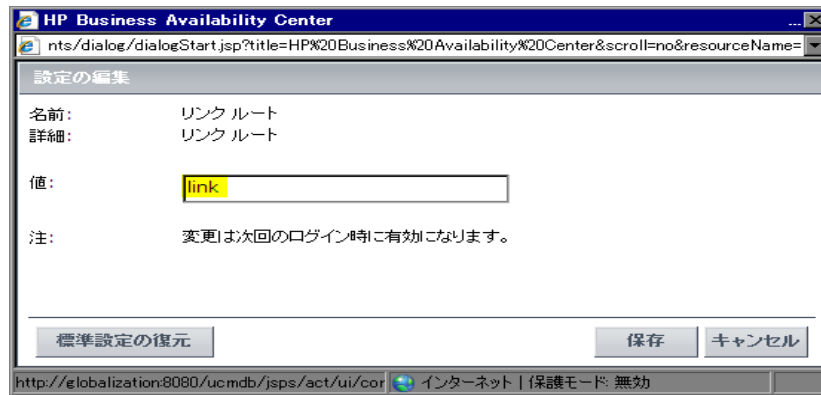
### UCMDB 7.52 と 8.x

出荷時設定のUCMDB 7.52 および 8.x インストールでは、リンクルート値は `it_world_links` です。インテグレーションリンクはすべて SM Link ノードにあるため、`it_world_links` ノードからは表示されません。インテグレーションリンクを表示または変更するには、リンクルート設定の値をクラスモデル内のより高いノードに変更する必要があります。

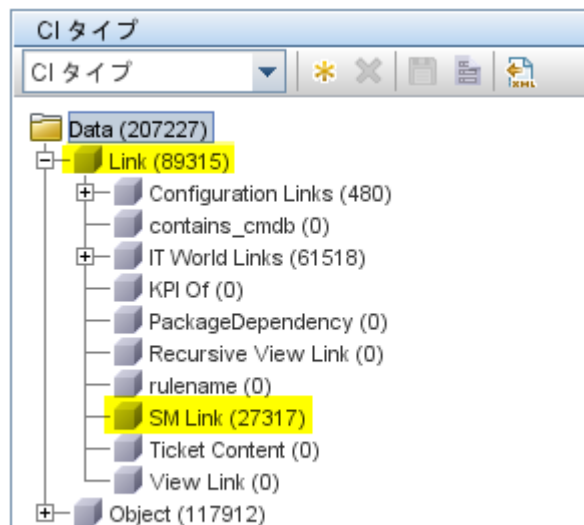
- 1 管理者として HP Universal CMDB システムにログインします。
- 2 **[設定]** タブを選択します。
- 3 **[インフラストラクチャ設定マネージャ]** をクリックします。
- 4 [コンテキストの選択] で、**[すべて]** オプションを選択します。
- 5 [UCMDB 共通 - 全般設定] セクションが表示されるまで下にスクロールします。

| UCMDB 共通 - 全般設定 |            |  |
|-----------------|------------|--|
| 名前 ▲            | 詳細         | 値  |
| オブジェクト ルート      | オブジェクト ルート | it_world        |
| リンク ルート         | リンク ルート    | it_world_links  |
| 削除候補の更新間隔       | 削除候補の更新間隔  | 86400000        |

- 6 リンクルートの編集ボタンをクリックします。 
- 7 リンクルート値を **「link」** に変更します。



- 8 [保存] をクリックします。
- 9 ログアウトしてからUCMDBにログインし直します。
- 10 インテグレーションリンクを表示するには、CI タイプマネージャを開いて、[Data] > [Link] > [SM Link] を参照します。



## オブジェクトルート設定の変更

UCMDB 7.52または8.xシステムを使用している場合、インテグレーションコンポーネントを表示するにはリンクルート設定を変更する必要があります。

### UCMDB 9.0




UCMDB 9.0では、インテグレーションでプライベートCIタイプが使用されなくなりました。インテグレーションCIタイプを表示するのに変更は必要ありません。




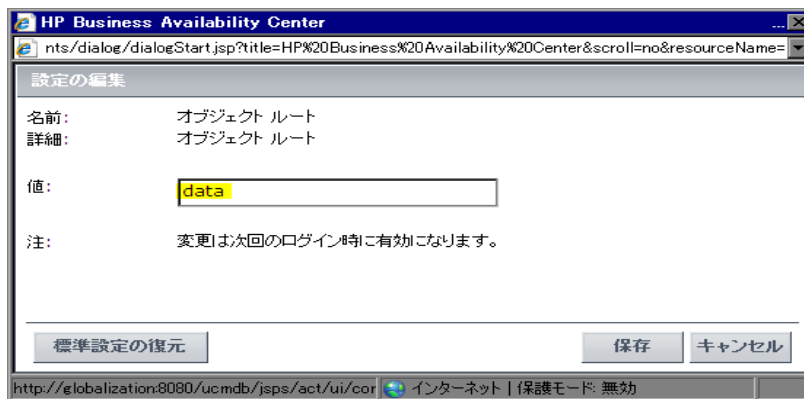
## UCMDB 7.52 と 8.x

出荷時設定のUCMDB 7.52および8.xインストールでは、オブジェクトルート値はit\_worldです。インテグレーションオブジェクトはすべてSM CI ノードにあるため、it\_world ノードからは表示されません。インテグレーションオブジェクトを表示または変更するには、オブジェクトルート設定の値をクラスモデル内のより高いノードに変更する必要があります。

- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 **[設定]** タブを選択します。
- 3 **[インフラストラクチャ設定マネージャ]** をクリックします。
- 4 [コンテキストの選択] で、**[すべて]** オプションを選択します。
- 5 [UCMDB 共通 - 全般設定] セクションが表示されるまで下にスクロールします。

| UCMDB 共通 - 全般設定 |            |  |
|-----------------|------------|--|
| 名前▲             | 詳細         | 値  |
| オブジェクト ルート      | オブジェクト ルート | it_world        |
| リンク ルート         | リンク ルート    | it_world_links  |
| 削除候補の更新間隔       | 削除候補の更新間隔  | 86400000        |

- 6 オブジェクトルートの編集ボタンをクリックします。 
- 7 オブジェクトルート値を「**data**」に変更します。

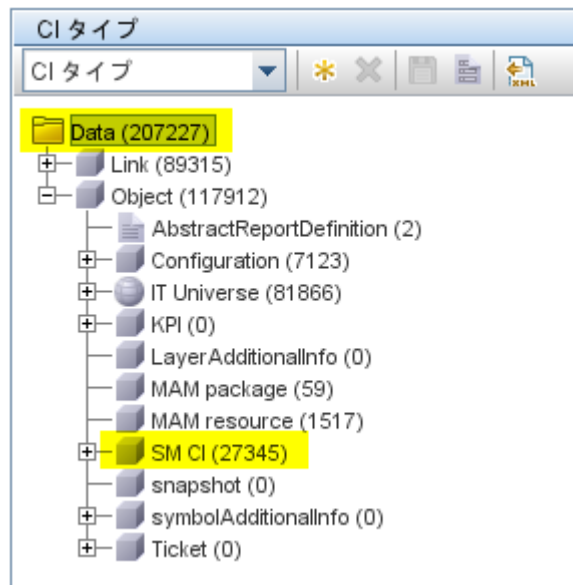


The screenshot shows a web browser window titled "HP Business Availability Center" with the URL "nts/dialog/dialogStart.jsp?title=HP%20Business%20Availability%20Center&scroll=no&resourceName=". The main content area is titled "設定の編集" (Edit Settings). It contains the following fields:

- 名前: オブジェクト ルート
- 詳細: オブジェクト ルート
- 値:
- 注: 変更は次回のログイン時に有効になります。

At the bottom, there are three buttons: "標準設定の復元" (Restore Default Settings), "保存" (Save), and "キャンセル" (Cancel). The status bar at the bottom of the browser shows "http://globalization:8080/ucmdb/jsps/act/ui/cor" and "インターネット | 保護モード: 無効".

- 8 **[保存]** をクリックします。
- 9 ログアウトしてからUCMDBにログインし直します。
- 10 インテグレーション CI タイプを表示するには、CI タイプマネージャを開いて、**[Data]** > **[Object]** > **[SM CI]** を参照します。



# カスタマイズのオプション

インテグレーションには、次のカスタマイズオプションがあります：

- インテグレーションへのCI属性の追加（67ページ）
- インテグレーションへのCIタイプの追加（81ページ）

## インテグレーションへのCI属性の追加

次の手順を用いて、インテグレーションにCI属性を追加できます。

**タスク1: UCMDB クラスモデルにCI属性は既に存在していますか？**

はい。タスク3に移動します。

いいえ。タスク2に移動します。

**タスク2: UCMDB クラスモデルにCI属性を追加します。**

UCMDB クラスモデルへのCI属性の追加（67ページ）を参照してください。

**タスク3: CI属性の計算を有効にします。**

CI属性の計算の有効化（70ページ）を参照してください。

**タスク4: CI属性の「変更をモニタ」オプションを有効にします。**

CI属性の「変更をモニタ」オプションの有効化（71ページ）を参照してください。

**タスク5: WebサービスオブジェクトへのCI属性のマッピング**

WebサービスオブジェクトへのCI属性のマッピング（73ページ）を参照してください。

**タスク6: UCMDB 9.00とインテグレーションしますか？**

はい。手順7に移動します。

いいえ。インテグレーションアダプタのコードベースの再ロード（75ページ）を参照してください。

**タスク7: CI属性をサポートするWebサービスプロジェクトを作成します。**

CI属性をサポートするWebサービスプロジェクトの作成（76ページ）を参照してください。

**タスク8: CI属性の変更を監視するための管理フィールドを追加します。**

CI属性をサポートするための管理フィールドの追加（78ページ）を参照してください。

## UCMDB クラスモデルへのCI属性の追加

インテグレーションクラスモデルでは、UCMDBシステムで利用できるCI属性のサブセットのみが使用されます。出荷時設定では、インテグレーションは、Service Managerシステムで通常管理されるCI属性（ホスト名やホストDNS名など）で構成されます。新規UCMDB CI属性を作成する前に、必要とするディスカバリデータを提供する既存のCI属性がUCMDBシステムにあるかどうかを特定する必要があります。通常は、イ

インテグレーションに追加するディスクバリデータを追跡している既存の属性が存在します。たとえば、ホストCIの属性を確認すると、インテグレーションに追加できる多くの属性が存在することがわかります（インテグレーション属性で、「変更をモニタ」設定が有効である必要があります）。

| 名前                   | 表示名                   | タイプ         | 詳細            | 標準設定値     | 可視 | 編集可能 | 変更をモニタ | 必須 | 小文字 |
|----------------------|-----------------------|-------------|---------------|-----------|----|------|--------|----|-----|
| ack_cleared_time     | ack_cleared_time      | long        |               |           |    |      |        |    |     |
| ack_id               | ack_id                | string      |               |           |    |      |        |    |     |
| root_actualdelet     | Actual Delete Time    | date        | When will...  |           |    |      |        |    |     |
| root_actualdelet     | Actual Deletion Pe... | integer     | What is th... | 40        |    |      |        |    |     |
| root_adminstate      | Admin State           | adminstat.  | Admin St...   | Managed   |    |      |        |    |     |
| data_allow_auto...   | Allow CI Update       | boolean     |               | true      |    |      |        |    |     |
| root_candidatfor...  | Candidate For Del...  | date        | When will...  |           |    |      |        |    |     |
| data_changecorr...   | Change Corr State     | changestat. | Change S...   | No Change |    |      |        |    |     |
| data_changefresh...  | Change Is New         | boolean     | Change S...   | false     |    |      |        |    |     |
| data_changestate     | Change State          | changestat. | Change S...   | No Change |    |      |        |    |     |
| root_class           | CI Type               | string      | Class na...   |           |    |      |        |    |     |
| city                 | City                  | string      | City locat... |           |    |      |        |    |     |
| codepage             | CodePage              | string      | System s...   |           |    |      |        |    |     |
| root_container       | Container             | string      | Container...  |           |    |      |        |    |     |
| contextmenu          | Context Menu          | string_list | Context m...  | iCIs      |    |      |        |    |     |
| country              | Country               | string      | Country lo... |           |    |      |        |    |     |
| root_createtime      | Create Time           | date        | When wa...    |           |    |      |        |    |     |
| data_source          | Created By            | string      |               |           |    |      |        |    |     |
| root_deletioncand... | Deletion Candidat...  | integer     | What is th... | 20        |    |      |        |    |     |
| data_description     | Description           | string      | Data Des...   |           |    |      |        |    |     |
| digest               | Digest                | string      |               |           |    |      |        |    |     |
| display_label        | Display Label         | string      | Used as       |           |    |      |        |    |     |
| document_list        | Documents             | string      | Documents     |           |    |      |        |    |     |
| root_enableaging     | Enable Aging          | boolean     | Is aging e... | false     |    |      |        |    |     |
| data_externalid      | External ID           | string      | external s... |           |    |      |        |    |     |
| FAMILY_ICON          | FAMILY_ICON           | string      | network       |           |    |      |        |    |     |
| host_systemmass...   | Host asset tag        | string      | Asset tag...  |           |    |      |        |    |     |
| host_last_boot_t...  | Host Boot Time        | date        | Host last...  |           |    |      |        |    |     |
| host_dnsname         | Host DNS Name         | string      | DNS Na...     |           |    |      |        |    |     |
| host_iscomplete      | Host Is Complete      | boolean     | True valu...  | false     |    |      |        |    |     |
| host_isdesktop       | Host Is Desktop       | boolean     | Is this ho... |           |    |      |        |    |     |
| host_isroute         | Host Is Route         | boolean     | True valu...  |           |    |      |        |    |     |

次の手順では、既存のCIタイプに新規CI属性を追加する方法を示します。このシナリオはあまり現実的な状況ではありません。通常は、インテグレーションに既存のCI属性を追加します。






インテグレーションでは、UCMDB クラスモデルにCI属性を追加するのに特別な手順は必要ありません。標準的なCI属性作成手順を用いて、CI属性を追加できます。CI属性の作成の詳細については、『*HP Universal CMDB CI Attribute Customization Guide*』を参照してください。

UCMDB クラスモデルにCI属性を追加するには：

- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 **[モデリング]** タブを選択します。
- 3 **[CI タイプ マネージャ]** をクリックします。
- 4 **[CI タイプ]** ナビゲーションツリーから、新規CI属性の追加先となるCIタイプを選択します。例：

表 20 サンプルCIタイプへのパス

| UCMDBバージョン       | パス   |
|------------------|--|
| UCMDB 7.52 と 8.x | <b>[Data]</b> > <b>[Object]</b> > <b>[IT Universe]</b> > <b>[System]</b> > <b>[Software Element]</b> > <b>[Database]</b>             |
| UCMDB 9.00       | <b>[Managed Object]</b> > <b>[ConfigurationItem]</b> > <b>[InfrastructureElement]</b> > <b>[RunningSoftware]</b> > <b>[Database]</b> |

- 5 **[属性]** タブを選択します。
- 6 **[追加]** ボタンをクリックします。   
[属性の追加] ウィンドウが開きます。
- 7 [属性名] に、新規CI属性に使用する一意の名前を入力します。「**database\_owner**」などです。  
 名前に次の文字を含めることはできません： / \ [ ] : | < > + = ; , ? \*
- 8 [表示名] に、インタフェースに表示する名前を入力します。「**Database Owner**」などです。
- 9 [説明] に、新規CI属性の説明を入力します。これはオプションのフィールドです。「**データベースを所有するシステムユーザ**」などです。
- 10 UCMDB 9.00の **[スコープ]** で、**[cms]** を選択します。
- 11 [属性タイプ] で、**[プリミティブ]** または **[列挙/リスト]** を選択します。たとえば、**[プリミティブ]** を選択して、**[文字列]** を選択します。
- 12 [値のサイズ] に、属性に格納できる最大文字数を入力します。「**300**」などです。
- 13 [標準設定値] に、その他の値が入力されない場合に使用される値を入力します。標準設定値を空欄にします。
- 14 [アドバンス] セクションで、**[変更をモニタ]** オプションを有効にします。
- 15 **[OK]** をクリックして、属性を保存します。
- 16 保存ボタンをクリックして、CIタイプに属性の変更を保存します。 

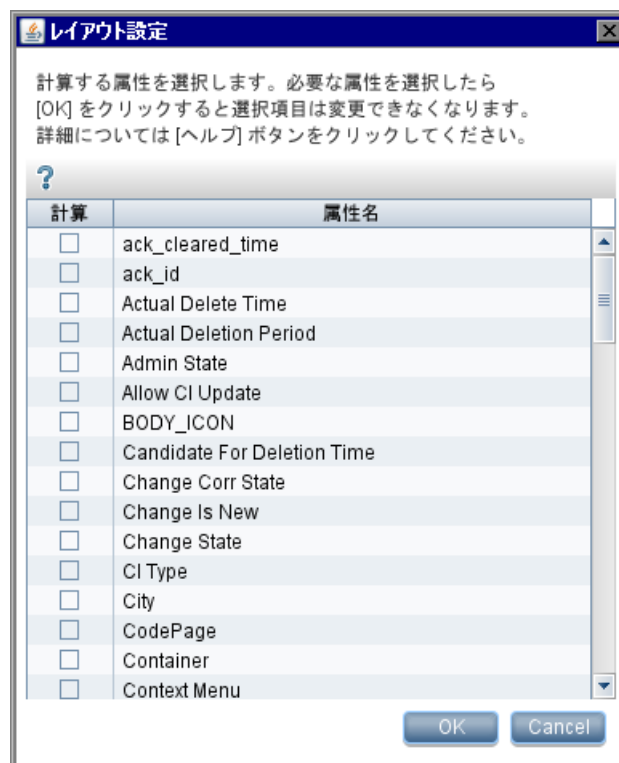
## CI属性の計算の有効化

インテグレーションにCI属性を追加するには、CIタイプを同期するTQLクエリから計算レイアウト設定を有効にする必要があります。インテグレーションに追加するCI属性がどのCIタイプに含まれるのかを把握しておく必要があります。



有効にする属性のリストを用意します。これは、有効にする属性ごとに一致するXSL変換を後で作成する必要があります。

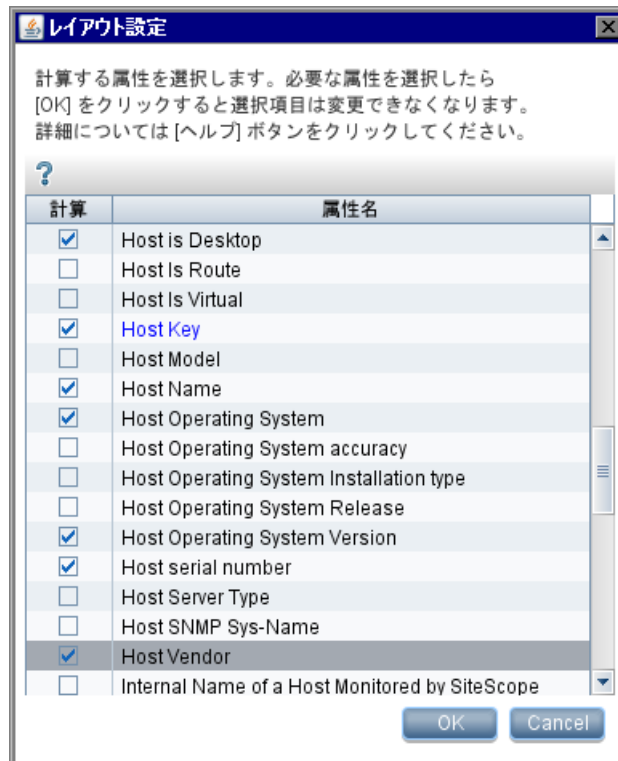
- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 [モデリング] タブをクリックします。
- 3 UCMDDB 9.00の場合、[**モデリングスタジオ**] を選択します。[**リソースタイプ**] ドロップダウンボックスから [**クエリ**] を選択します。  
UCMDDB 7.52と8.xの場合、[**クエリ マネージャ**] をクリックします。
- 4 [クエリ] ナビゲーションツリーから、[**Integration**] > [**SM Sync**] をクリックします。
- 5 インテグレーションに追加する属性があるCIタイプを管理するクエリを選択します。[**hostData**] などは、UCMDDBには、インテグレーションクエリのためのTQLが表示されます。
- 6 インテグレーションに追加するCI属性を含むノードをTQLレイアウトから追加します。「**Root**」などです。
- 7 UCMDDB 9.00の場合、ノードを右クリックして [**クエリ ノードのプロパティ**] を選択します。[クエリ ノードのプロパティ] ウィンドウが開きます。  
UCMDDB 7.52と8.xの場合、[情報] ウィンドウで [**編集**] ボタンをクリックします。[ノードのプロパティ] ウィンドウが表示されます。
- 8 [**詳細レイアウト設定**] をクリックします。[レイアウト設定] ウィンドウが開きます。



9 インテグレーションに含めるCI属性の [計算] オプションを有効にします。例：


表 21 サンプルのCI属性に計算を有効にする

| UCMDBバージョン       | CI属性        |
|------------------|-------------|
| UCMDB 7.52 と 8.x | Host vendor |
| UCMDB 9.00       | OsVendor    |



10 [OK] をクリックして、[レイアウト設定] ウィンドウを閉じます。

11 [OK] をクリックして、ノードのプロパティを保存します。

12 [保存] ボタンをクリックして、TQLクエリを保存します。 

## CI属性の [変更をモニタ] オプションの有効化

インテグレーションがCI属性値の更新を送信するには、インテグレーションに含める各CIの [変更をモニタ] オプションを有効にする必要があります。

- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 [モデリング] タブをクリックします。
- 3 [CI タイプ マネージャ] をクリックします。

4 [CIタイプ] ナビゲーションツリーから、新規CI属性の追加先となるCIタイプを選択します。例：

表 22 サンプルCIタイプへのパス


| UCMDBバージョン       | パス  |
|------------------|---|
| UCMDB 7.52 と 8.x | [Data] > [Object] > [IT Universe] > [System] > [Software Element] > [Database]                    |
| UCMDB 9.00       | [Managed Object] > [ConfigurationItem] > [InfrastructureElement] > [RunningSoftware] > [Database] |

5 [属性] タブを選択します。

6 インテグレーションに含める属性を選択します。例：

表 23 サンプルCIの変更のモニタを有効にする

| UCMDBバージョン       | CI属性        |
|------------------|-------------|
| UCMDB 9.00       | Vendor      |
| UCMDB 7.52 と 8.x | Host Vendor |

7 [編集] ボタンをクリックします。 

[属性の編集] ウィンドウが開きます。


8 UCMDB 9.00の場合、[詳細] をクリックして [変更をモニタ] オプションを有効にします。

UCMDB 7.52 と 8.xの場合、[詳細] セクションから [変更をモニタ] オプションを有効にします。



9 [OK] をクリックして、変更を確定します。



10 [保存] ボタンをクリックして、TQLクエリを保存します。 

## WebサービスオブジェクトへのCI属性のマッピング

インテグレーションは、アダプタを使用して、UCMDB CI属性をService Managerが認識するWebサービスオブジェクトに変換します。アダプタは、UCMDB TQLクエリを適切にフォーマットされたService Manager Webサービスメッセージに変換するのに使用するXSL変換ファイルを指定します。

出荷時設定では、各インテグレーションクエリに、UCMDBの特定CIタイプにマップされる対応XSL変換ファイルがあります。さらに、計算を有効にする属性はそれぞれ、XSL変換ファイル内に独自のエントリを必要とします。XSL変換エントリがない場合、Service ManagerはUCMDBシステムからCI属性の更新を受信できません。

インテグレーションに新規属性を追加する場合、上位CIタイプのXSL変換ファイルを編集して、CI属性のエントリを追加する必要があります。各クエリが管理するCIタイプの詳細については、[インテグレーション TQLクエリ](#) (51 ページ) を参照してください。適切なXSLマッピングを作成するため、Service ManagerがWebサービスとして公開するサービスとオブジェクト名を理解しておく必要があります。Webサービスオブジェクトとして公開するテーブルと列の詳細については、Service Managerヘルプを参照してください。

次の手順では、UCMDB CI属性である vendor (UCMDB 7.52 と 8.x の場合は host\_vendor) を Service Manager Webサービスオブジェクトである OSVendor にマップする方法を示します。

### WebサービスオブジェクトにCI属性をマップするには：

- 1 管理者アカウントでUCMDBファイルシステムにログインします。
- 2 UCMDB 9.0を使用している場合、[手順3](#)に移動します。  
UCMDB 7.52または8.xを使用している場合、[手順7](#)に移動します。
- 3 **[マネージャ]** > **[データフロー管理]** > **[アダプタ管理]** をクリックします。
- 4 Service Managerアダプタに移動します。例：  
**[ServiceManagerAdapter7-1]** > **[構成ファイル]**
- 5 CI属性の上位CIタイプを管理するXSL変換ファイルをダブルクリックします。たとえば、hostData TQLクエリに属性を追加するには、host\_data.xsltを開きます。
- 6 [手順9](#)に移動します。
- 7 ファイルシステムでService Deskアダプタのパスを参照します。例：  
**C:\hp\UCMDB\UCMDBServer\j2f\fcmdb\CodeBase  
\ServiceDeskAdapter**
- 8 テキストエディタを使用して、CI属性の上位CIタイプを管理するXSL変換ファイルを開きます。たとえば、hostData TQLクエリに属性を追加するには、host\_data.xsltを開きます。
- 9 インテグレーションがCI属性値を格納するService Managerテーブル名を定義している要素を見つけます。たとえば、要素<file.device>はService ManagerのdeviceテーブルにCI属性を格納します。
- 10 テーブル命名要素 (<file.device>) 内には、各UCMDB CI属性をWebサービスオブジェクトに変換する方法を定義している、次の形式の要素があります：

```

<xsl:for-each select="@CI_attribute_name">
  <ObjectName><xsl:value-of select="."/></ObjectName>
</xsl:for-each>

```

@CI\_attribute\_nameは、UCMDBシステム内での属性の名前です。

ObjectNameは、Service Managerシステムで公開されるWebサービスオブジェクトの名前です。

図1 host\_data.xslt XSL変換ファイル内のCI属性

```

29 </Room>
30 <xsl:for-each select="@default_gateway_ip_address">
31 <DefaultGateway>
32 <xsl:value-of select="."/>
33 </DefaultGateway>
34 </xsl:for-each>
35 <xsl:for-each select="@discovered_os_name">
36 <OS>
37 <xsl:value-of select="."/>
38 </OS>
39 </xsl:for-each>
40 <xsl:for-each select="@primary_dns_name">
41 <DNSName>
42 <xsl:value-of select="."/>
43 </DNSName>
44 </xsl:for-each>
45 </file.device>
46 <computer>
47 <xsl:for-each select="@name">
48 <HostName>
49 <xsl:value-of select="."/>
50 </HostName>
51 </xsl:for-each>
52 <xsl:for-each select="@discovered_os_version">
53 <OSVersion>
54 <xsl:value-of select="."/>
55 </OSVersion>
56 </xsl:for-each>
57 <xsl:for-each select="@serial_number">
58 <BIOSId>
59 <xsl:value-of select="."/>
60 </BIOSId>
61 </xsl:for-each>
62 <xsl:for-each select="@memory_size">
63 <PhysicalMemory>
64 <xsl:value-of select="."/>
65 </PhysicalMemory>
66 </xsl:for-each>

```

図2 ucmdbComputer Webサービスにある一致するCI属性

**外部アクセス定義**

サービス名:   リリース日  
 名前:   廃止予定  
 オブジェクト名:

◆ 許可されるアクション ◆ 式 ◆ フィールド

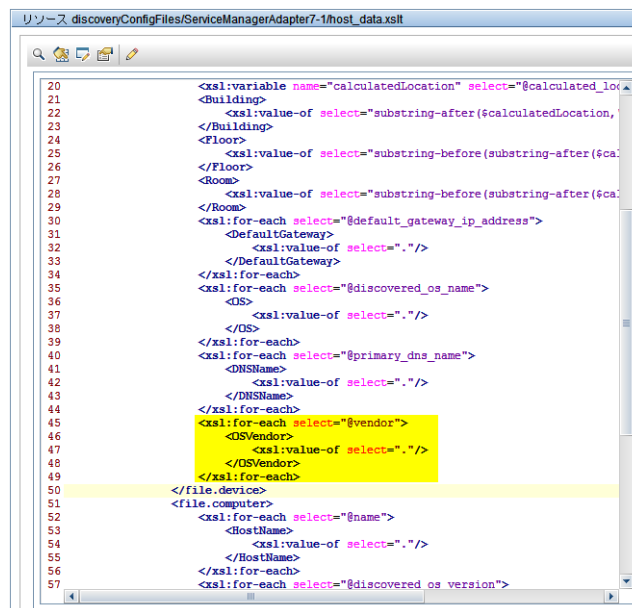
| フィールド                    | キャプション         | タイプ |
|--------------------------|----------------|-----|
| ucmdb.id                 | UCMDBId        |     |
| default.gateway          | DefaultGateway |     |
| operating.system         | OS             |     |
| os.version               | OSVersion      |     |
| bios.id                  | BIOSId         |     |
| network.name             | DNSName        |     |
| addIIPAddr[addIIPAddr... | AddIIPAddress  |     |
| addIIPAddr[addISubnet]   | AddSubnet      |     |
| addIMacAddress           | AddIMacAddress |     |
| physical.mem.total       | PhysicalMemory |     |
| subtype                  | Subtype        |     |
| building                 | Building       |     |
| room                     | Room           |     |
| floor                    | Floor          |     |
| machine.name             | HostName       |     |
| company                  | CustomerId     |     |

- 11 既存の XSL 変換要素をコピーして、新しい変換エントリを作成するためのテンプレートとして使用します。
- 12 適切なテーブル命名要素内に新しい XSL 変換要素を貼り付けます。たとえば、<file.computer> です。
- 13 インテグレーションに追加する属性に一致するように、新しい要素内で CI 属性名と Web サービスオブジェクト名を更新します。たとえば、インテグレーションに属性を追加するには、次の XSL 変換要素を作成します。

表 24

| UCMDBバージョン | 必要な XSL 要素   |
|------------|--|
| 9.0        | <pre>&lt;xsl:for-each select="@vendor"&gt;   &lt;OSVendor&gt;&lt;xsl:value-of select="." /&gt;&lt;/OSVendor&gt; &lt;/xsl:for-each&gt;</pre>      |
| 7.52 と 8.x | <pre>&lt;xsl:for-each select="@host_vendor"&gt;   &lt;OSVendor&gt;&lt;xsl:value-of select="." /&gt;&lt;/OSVendor&gt; &lt;/xsl:for-each&gt;</pre> |

図 3 host\_data.xslt XSL 変換ファイル内の新規属性



- 14 XSL 変換ファイルを保存します。

## インテグレーションアダプタのコードベースの再ロード

この説明は、UCMDB 7.52 または 8.x システムを使用したインテグレーションにのみ適用されます。UCMDB 9.00 ではアダプタを手動で再ロードする必要はありません。

インテグレーションでカスタム CI 属性を使用するには、インテグレーションアダプタのコードベースを再ロードする必要があります。

**ServiceDeskAdapter のコードベースを再ロードするには :**

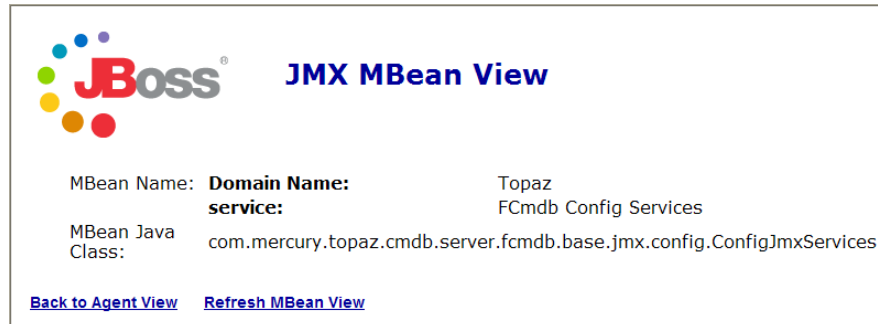
- 1 Web ブラウザを開いて、次の URL にアクセスします :

**http://<UCMDB サーバ>:8080/jmx-console/  
HtmlAdaptor?action=inspectMBean&name=Topaz%3Aservice%3  
DFCmdb+Config+Services**

<UCMDB サーバ>は、UCMDBシステムの完全修飾ドメイン名です。

- 2 入力が求められたら、UCMDB管理者のユーザ名とパスワードを入力します。  
JMX MBean ViewのWebページが開きます。

図 4 JMX MBean View



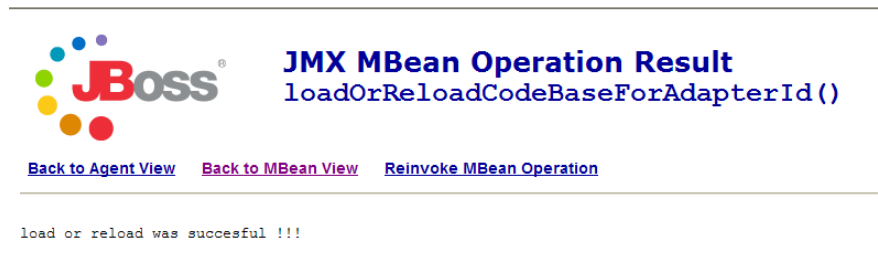
- 3 loadOrReloadCodebaseForAdapterId() MBeanのセクションまで移動します。
- 4 customerIDのParamValueに「1」と入力します。
- 5 adapterIdのParamValueに「ServiceDeskAdapter」と入力します。

### java.lang.String loadOrReloadCodeBaseForAdapterId()

Load or reload adapter code base

| Param      | ParamType        | ParamValue                                      | ParamDescription |
|------------|------------------|---|------------------|
| customerID | int              | <input type="text" value="1"/>                  | Customer id      |
| adapterId  | java.lang.String | <input type="text" value="ServiceDeskAdapter"/> | Adapter id       |

- 6 関数の下にある [Invoke] ボタンをクリックします。成功のウィンドウが開きます。



## CI属性をサポートするWebサービスプロジェクトの作成

UCMDBは、Service ManagerのucmdbIntegration Webサービスを使用してCIデータを送信します。このWebサービスは、UCMDB インテグレーションが提供する出荷時設定のCIタイプとCI属性に一致するオブジェクトを公開します。出荷時設定のWebサービスオブジェクトとそのService Manager テーブルへのマッピングのリストについては、[管理フィールド](#)（55ページ）を参照してください。

UCMDBシステムでインテグレーションにCI属性を追加する場合、Service Managerシステム上で対応Webサービスオブジェクトを作成して、送信されるCIデータを受信する必要があります。各Webサービスオブジェクトは、有効なService Managerテーブルと列にマップされる必要があります。

次の手順では、前のセクションで説明したOSVendor属性のWebサービスオブジェクトを作成する方法を示します。

#### CI属性のWebサービスオブジェクトを作成するには：

- 1 Service Manager サーバに管理者アカウントでログインします。
- 2 **[カスタマイズ]** > **[Web サービス]** > **[WSDL 設定]** をクリックします。[外部アクセス定義] フォームが開きます。
- 3 [サービス名] に「**ucmdbIntegration**」と入力します。
- 4 **[検索]** をクリックします。ucmdbIntegration Webサービスを構成するオブジェクトのレコードリストが表示されます。
- 5 CI属性の追加先となる既存のWebサービスオブジェクトを選択します。[ucmdbComputer] などを選択します。
- 6 **[フィールド]** タブをクリックします。Webサービスオブジェクトとして公開されるフィールドが表示されます。
- 7 [フィールド] リストで空欄の行を選択します。
- 8 [フィールド] で、受信するCI属性値の格納先となるService Manager列の名前を選択します。[os.manufacturer] などです。
  - ▶ Service Manager に、[名前] フィールドに一覧されるテーブルに関連付けられているすべての結合テーブルのフィールドが表示されます。たとえばjoincomputerには、deviceテーブルとcomputerテーブルのフィールドが表示されます。
- 9 [キャプション] に、Service Managerが、Webサービスオブジェクトとしてフィールドを公開するときに使用する名前を入力します。「**OSVendor**」などです。
  - ▶ キャプション名は、UCMDB上のXSL変換ファイルに一覧されているオブジェクト名と一致する必要があります。
- 10 **[保存]** をクリックします。Service Managerシステムの次の再起動時に、新規Webサービスオブジェクトが利用できるようになります。

## 外部アクセス定義


|          |   |                                |
|----------|---|--------------------------------|
| サービス名:   | <input type="text" value="ucmdbIntegration"/> | <input type="checkbox"/> リリース日 |
| 名前:      | <input type="text" value="joincomputer"/>     | <input type="checkbox"/> 廃止予定  |
| オブジェクト名: | <input type="text" value="ucmdbComputer"/>    |                                |

| フィールド                     | キャプション         | タイプ |
|---------------------------|----------------|-----|
| ucmdb.id                  | UCMDBId        |     |
| default.gateway           | DefaultGateway |     |
| operating.system          | OS             |     |
| os.version                | OSVersion      |     |
| bios.id                   | BIOSId         |     |
| network.name              | DNSName        |     |
| addIIPAddr[addIIPAddr...] | AddIIPAddress  |     |
| addIIPAddr[addSubnet]     | AddSubnet      |     |
| addIMacAddress            | AddIMacAddress |     |
| physical.mem.total        | PhysicalMemory |     |
| subtype                   | Subtype        |     |
| building                  | Building       |     |
| room                      | Room           |     |
| floor                     | Floor          |     |
| machine.name              | HostName       |     |
| company                   | CustomerId     |     |
| os.manufacturer           | OSVendor       |     |

## CI属性をサポートするための管理フィールドの追加

インテグレーションに追加するCI属性が、自動化された変更管理の確認と検証プロセスをトリガするには、そのCI属性の管理フィールドを追加する必要があります。Service Manager管理フィールドは、検出イベントマネージャールールの構成要素です。変更管理の確認と検証をトリガするフィールドのリストについては、[Service Manager検出イベントマネージャールール \(59ページ\)](#) を参照してください。

#### インテグレーションに管理フィールドを追加するには：

- 1 Service Manager サーバに管理者アカウントでログインします。
- 2 **[カスタマイズ]** > **[Web サービス]** > **[検出イベントマネージャールール]** をクリックします。レコードを検索するためのフォームが表示されます。
- 3 **[検索]** をクリックして、すべての検出イベントマネージャールールのリストを表示します。
- 4 受信するCI属性をマップした Web サービスオブジェクトと一致するルールIDを選択します。**[ucmdbComputer]** などです。この Web サービスオブジェクトのルールが表示されます。
- 5 **[管理フィールド]** タブをクリックします。変更管理の確認と検証をトリガするフィールドのリストが表示されます。
- 6 **[管理フィールド]** リストで空欄の行を選択します。
- 7 **[フィールド名]** で、先に選択した、受信するCI属性値の格納先となる Service Manager 列のキャプション名を選択します。**[OS 製造業者]** などです。  
 Service Manager に、**[テーブル名]** フィールドに一覧されるテーブルに関連付けられているすべての結合テーブルのフィールドが表示されます。たとえばjoincomputerには、device テーブルと computer テーブルのフィールドが表示されます。
- 8 受信CI属性を格納するのに使用するフィールドが配列構造である場合、**[構造]** フィールドを使用して、列が存在する構造体の配列の名前を選択します。たとえば、**[OS 製造業者]** はプリミティブな文字フィールドであるため、配列構造の名前を指定する必要はありません。
- 9 受信CI属性を格納するのに使用するフィールドが配列構造である場合、**[インデックス]** フィールドを使用して、構造体の配列内にある列を特定するインデックス番号を選択します。たとえば、**[OS 製造業者]** はプリミティブな文字フィールドであるため、配列構造のインデックスを指定する必要はありません。
- 10 **[保存]** をクリックします。Service Manager システムの次の再起動時に、管理フィールドが利用できるようになります。

## 検出イベントマネージャールール

ID:

テーブル名:

条件:

| フィールド名       | 構造        | インデックス | 条件 |
|--------------|-----------|--------|----|
| OS           |           |        |    |
| デフォルトゲートウェイ  |           |        |    |
| OS バージョン     |           |        |    |
| BIOS ID      |           |        |    |
| ネットワーク名      |           |        |    |
| addIPAddress | addIPAddr | 1      |    |
| addSubnet    | addIPAddr | 2      |    |
| 追加 MAC アドレス  |           |        |    |
| 物理メモリ合計      |           |        |    |
| マシン名         |           |        |    |
| サブタイプ        |           |        |    |
| ビル名          |           |        |    |
| 部屋番号         |           |        |    |
| 階            |           |        |    |
| UCMDB ID     |           |        |    |
| OS 製造業者      |           |        |    |



## インテグレーションへのCIタイプの追加

次の手順を用いて、インテグレーションにCIタイプを追加できます。

**タスク1: UCMDB クラスモデルにCIタイプは既に存在していますか?**

はい。タスク3に移動します。

いいえ。タスク2に移動します。

**タスク2: UCMDB クラスモデルにCIタイプを追加します。**

UCMDB クラスモデルへのCIタイプの追加 (82ページ) を参照してください。

**タスク3: 必要に応じて、CIタイプにCI属性を追加します。**

インテグレーションへのCI属性の追加 (67ページ) を参照してください。

**タスク4: TQLクエリを作成して、CIタイプの属性を同期します。**

CIタイプの属性を同期するためのTQLクエリの作成 (85ページ) を参照してください。

**タスク5: TQLクエリ内のすべての関係に追跡修飾子が存在することを確認します。**

TQLクエリ内のすべての関係に追跡修飾子が存在することの確認 (89ページ) を参照してください。

**タスク6: CIタイプの属性の計算を有効にします。**

CIタイプ属性の計算の有効化 (90ページ) を参照してください。

**タスク7: CIタイプのTQLクエリをXSL変換ファイルにマップします。**

CIタイプTQLクエリのXSL変換ファイルへのマップ (92ページ) を参照してください。

**タスク8: WebサービスオブジェクトにCIタイプの属性をマップします。**

CIタイプ属性のWebサービスオブジェクトへのマップ (94ページ) を参照してください。

**タスク9: UCMDB 9.00とインテグレーションしますか?**

はい。手順11に移動します。

いいえ。ServiceDeskAdapterのコードベースの再ロード (97ページ) を参照してください。

**タスク10: インテグレーション変更アダプタにCIタイプのTQLクエリを追加します。**

インテグレーション変更アダプタへのカスタムTQLクエリの追加 (98ページ) を参照してください。

**タスク11: インテグレーションレプリケーションジョブにCIタイプのTQLクエリを追加します。**

インテグレーションデータプッシュ (レプリケーションジョブ) へのカスタムTQLクエリの追加 (99ページ) を参照してください。

**タスク12: CIタイプをサポートするWebサービスプロジェクトを作成します。**

CIタイプをサポートするWebサービスオブジェクトの作成 (101ページ) を参照してください。

**タスク13: CIタイプをサポートするための管理フィールドを追加します。**

CIタイプをサポートするための管理フィールドの追加 (103ページ) を参照してください。

## UCMDB クラスモデルへのCIタイプの追加

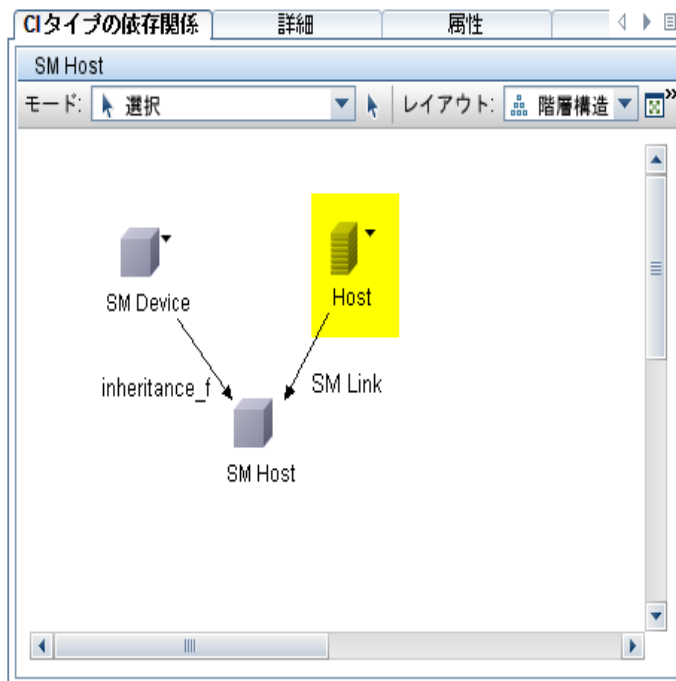
### UCMDB 9.00

インテグレーションでは、UCMDB 9.00のプライベートクラスモデルは使用しません。新規UCMDB CIタイプを作成する前に、必要なCI属性を提供する既存のCIタイプがUCMDBシステムにあるかどうかを特定する必要があります。通常は、1つまたは複数の既存のCIタイプへのリンクを作成することで、インテグレーションで使用する新しい論理CIタイプを作成します。[サンプルのCIタイプ](#)（82ページ）を参照してください。

### UCMDB 7.52 と 8.x

インテグレーションクラスモデルでは、UCMDBシステムで利用できるCIタイプのサブセットのみが使用されます。出荷時設定では、インテグレーションは、Service Managerシステムで通常管理されるCIタイプ（ホストやネットワークコンポーネントなど）で構成されます。新規UCMDB CIタイプを作成する前に、必要なCI属性を提供する既存のCIタイプがUCMDBシステムにあるかどうかを特定する必要があります。通常は、1つまたは複数の既存のCIタイプへのリンクを作成することで、インテグレーションで使用する新しい論理CIタイプを作成します。

たとえば、SM Host CIはHost CIにリンクしている論理CIです。SM Host CIは、Hostとその下位のすべてのCIタイプのCI属性を継承します。



### サンプルのCIタイプ

次の手順では、databaseという既存のCIタイプを基にして、SM RDBMSという新規CIタイプを作成する方法を示します。




インテグレーションでは、UCMDBクラスモデルにCIタイプを追加するのに特別な手順は必要ありません。標準的なCIタイプ作成手順を用いて、CIタイプを追加できます。CIタイプの作成の詳細については、『*HP Universal CMDB CI Attribute Customization Guide*』を参照してください。

UCMDB クラスモデルにCIタイプを追加するには：


- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 [モデリング] タブを選択します。
- 3 [CIタイプマネージャ] をクリックします。
- 4 [CIタイプ] ナビゲーションツリーから、新規CIタイプに対して使用するベースのCIタイプを選択します。例：

表 25 サンプルCIタイプへのパス

| UCMDBバージョン       | パス  |
|------------------|---|
| UCMDB 9.00       | [Managed Object] > [ConfigurationItem] > [InfrastructureElement] > [RunningSoftware] > [Database] |
| UCMDB 7.52 と 8.x | [Data] > [Object] > [IT Universe] > [System] > [Software Element] > [Database]                    |

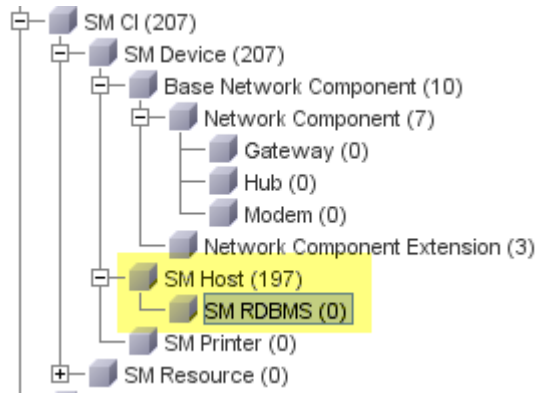
- 5 [新規] ボタンをクリックします。

[構成アイテムタイプを作成] ウィンドウが開きます。
- 6 [名前] に、新規CIタイプに使用する一意の名前を入力します。「sm\_rdbms」などです。
 

 名前に次の文字を含めることはできません： / \ [ ] : | < > + = ; , ? \*
- 7 [表示名] に、インターフェースに表示する名前を入力します。「SM RDBMS」などです。
- 8 [説明] に、新規CIタイプの説明を入力します。これはオプションのフィールドです。「関係データベースを実行するCI」などです。
- 9 [ベースのCIタイプ] で、適切なベースCIタイプが選択されていることを確認します。新規CIタイプは、ここで選択するベースのCIタイプの属性を継承します。例：

| UCMDBバージョン       | ベースのCIタイプ |
|------------------|-----------|
| UCMDB 9.00       | Database  |
| UCMDB 7.52 と 8.x | SM Host   |

- 10 [次へ] をクリックします。ウィザードには、ベースのCIタイプのCI属性のリストが表示されます。
- 11 新規CIタイプの必要に応じて、CI属性を追加、編集、または削除します。たとえば、ベースのCIタイプから継承するデフォルトの属性をそのまま使用します。
- 12 [次へ] をクリックします。ウィザードには、ベースのCIタイプの修飾子のリストが表示されます。
- 13 新規CIタイプの必要に応じて、修飾子を追加または削除します。たとえば、デフォルトの修飾子をそのまま使用します。
- 14 [次へ] をクリックします。ウィザードには、CIタイプに関連付けられたアイコンのリストが表示されます。
- 15 このCIタイプに関連付けるアイコンを選択します。たとえば、デフォルトのアイコンをそのまま使用します。
- 16 [次へ] をクリックして、必要に応じてメニューアイテムのプロパティやラベル定義を追加します。たとえば、ベースのCIタイプのデフォルト設定をそのまま使用します。
- 17 [完了] をクリックして、CIタイプを作成します。



- 18 ツリーから新規CIタイプを選択します。[SM RDBMS] などです。
- 19 リンク先とする既存のCIタイプを参照して、Ctrlキーを押しながらクリックし、そのCIタイプを選択に追加します。例：

▶ 新しい論理CIタイプに追加する属性がある既存のCIタイプを選択します。

#### UCMDBバージョン

#### ベースのCIタイプ

UCMDB 9.00

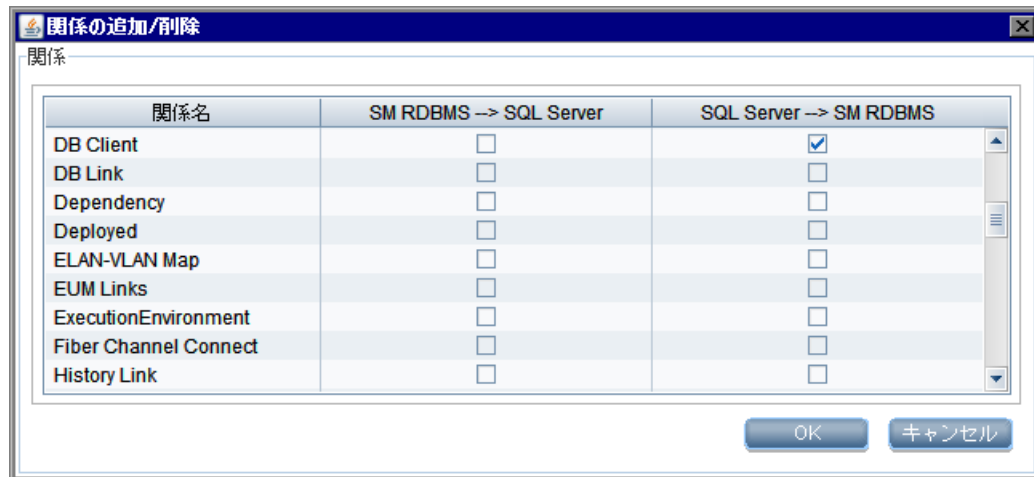
SQL Server

UCMDB 7.52 と 8.x

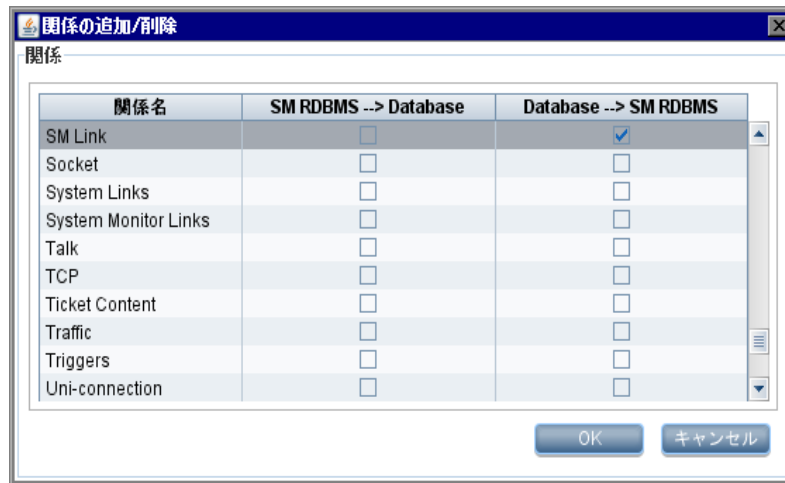
Database

- 20 選択したCIタイプのいずれかを右クリックして、[関係の追加/削除] をクリックします。[関係] ウィンドウが開きます。
- 21 既存のCIタイプから新規CIタイプへの関係を作成します。例：

UCMDB 9.00の場合、**SQL Server**から**SM RDBMS**への**DB Client**関係を作成します。



UCMDB 7.52 と 8.xの場合、**Database**から**SM RDBMS**への**SM Link**関係を作成します。



22 [OK] をクリックして、関係を作成します。

23 [保存] ボタンをクリックして、CIタイプを保存します。



## CIタイプの属性を同期するためのTQLクエリの作成

インテグレーションでは、Topology Query Language (TQL) クエリを使用して、CI属性値を収集してService Managerシステムに渡します。インテグレーションに追加する任意のCIタイプについて、TQLクエリを作成する必要があります。作成するTQLクエリは、いずれも **TQLクエリの要件** (54ページ) に準拠する必要があります。

次の手順では、前のセクションで説明したSM RDBMS CIタイプ向けに、rdbmsDataという新規TQLクエリを作成する方法を説明します。

- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 [モデリング] タブを選択します。
- 3 UCMDDB 9.00の場合、[モデリングスタジオ] を選択します。[リソースタイプ] ドロップダウンボックスから [クエリ] を選択します。  
UCMDDB 7.52と8.xの場合、[クエリ マネージャ] をクリックします。
- 4 [クエリ] ナビゲーションツリーから、[Integration] > [SM Sync] をクリックします。
- 5 UCMDDB 9.00の場合、[新規] ボタンをクリックして [クエリ] を選択します。手順12を参照してください。

UCMDDB 7.52と8.xの場合、[新規] をクリックします。



[新規クエリの作成] ウィンドウが開きます。

- 6 [名前] に、新規クエリに使用する一意の名前を入力します。名前に次の文字を含めることはできません: / \ [ ] : | < > + = ; , ? \* 「rdbmsData」などです。
- 7 [説明] に、新規クエリの説明を入力します。これはオプションのフィールドです。「**関係データベースを実行するホストのクエリ**」などです。
- 8 [クエリタイプ] で、[インテグレーション] を選択します。インテグレーションは、このフォルダに含まれるクエリのCIタイプのみを同期します。
- 9 [クエリの優先度] で、システムがクエリを再実行する頻度を選択します。たとえば、[中間] を選択して、デフォルトの優先度をそのまま使用します。

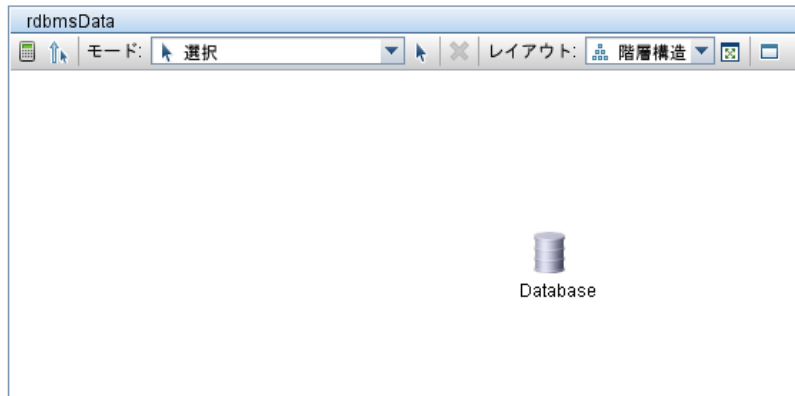
- 10 必要に応じて、永続的にするかどうかを切り替えます。TQLをシステムメモリに永続的に保存する場合は、[永続] を選択します。たとえば、永続が無効なデフォルトのステータスをそのまま使用します。
- 11 [OK] をクリックします。UCMDBが [クエリ] リストに新規クエリを追加します。
- 12 [CIタイプセレクト]から、クエリのルートノードとなるCIタイプを探します。このCIタイプは通常、CIのほとんどの属性を提供するCIタイプです。例：

表 26 ルートCIタイプへのパス

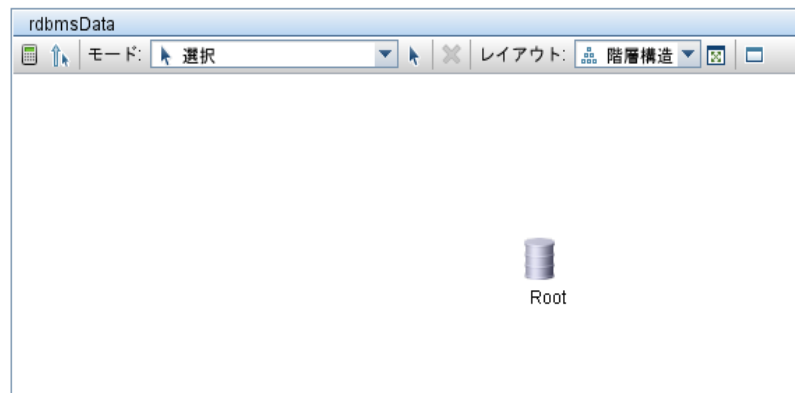
| UCMDBバージョン       | パス  |
|------------------|---|
| UCMDB 9.00       | [Managed Object] > [ConfigurationItem] > [InfrastructureElement] > [RunningSoftware] > [Database] |
| UCMDB 7.52 と 8.x | [Data] > [Object] > [IT Universe] > [System] > [Software Element] > [Database]                    |

▶ UCMDB 7.52 と 8.x では、**Database**CIタイプはより多くのCI属性を提供することから、**SM RDBMS** CIタイプよりも優れたルートノード要素になります。

- 13 [CIタイプセレクト]からルートCIタイプをドラッグして、空の [編集] 表示枠にドラッグします。UCMDBにCIタイプのアイコンが表示されます。



- 14 CIタイプを右クリックして、[クエリ ノードのプロパティ] (UCMDB 9.00) または [編集] (UCMDB 7.52 と 8.x) をクリックします。[ノードのプロパティ] ウィンドウが表示されます。
- 15 要素名を「**Root**」に変更します。
- 16 [OK] をクリックして、ノードのプロパティを保存します。

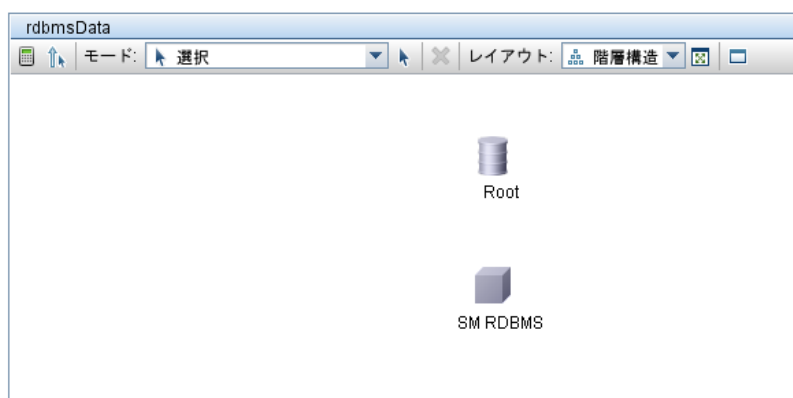


17 [CI タイプ セレクタ] から、クエリに追加する追加CIタイプを探します。これらのCIタイプは通常、追加のCI属性を提供します。例：、

表 27 追加のクエリ CIタイプへのパス

| UCMDBバージョン       | パス  |
|------------------|---|
| UCMDB 9.00       | [Managed Object] > [ConfigurationItem] > [InfrastructureElement] > [RunningSoftware] > [SM RDBMS] |
| UCMDB 7.52 と 8.x | [Data] > [Object] > [SM CI] > [SM Device] > [SM Host] > [SM RDBMS]                                |

18 [CI タイプ セレクタ] から追加のCIタイプをドラッグして、空の [編集] 表示枠にドラッグします。UCMDBに追加のCIタイプのアイコンが表示されます。

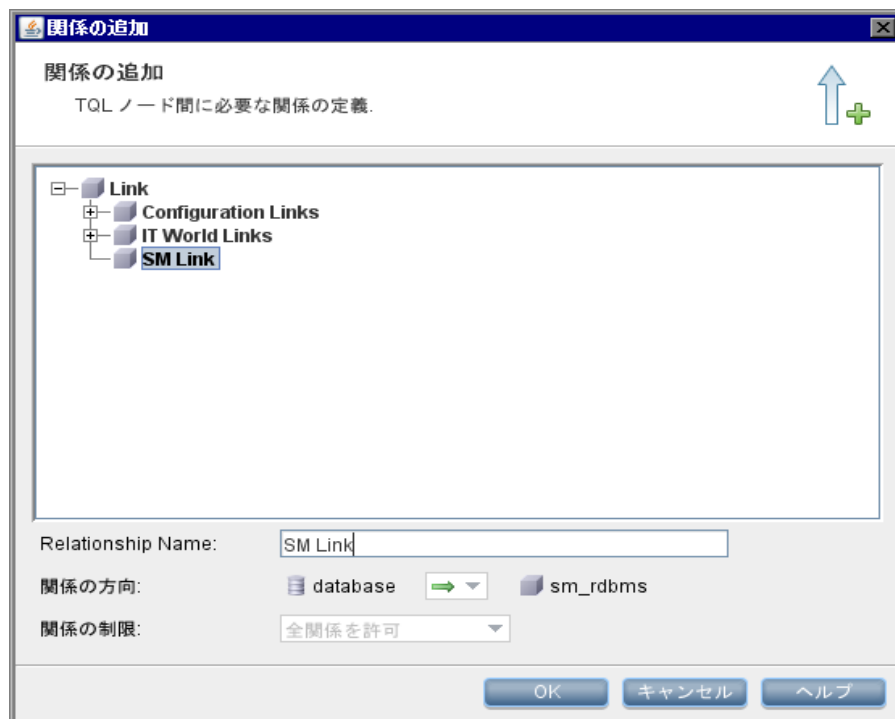


19 必要に応じて、ルートCIタイプと追加のCIタイプ間の関係を作成します。例：、


表 28 ルートクエリと追加クエリ間の関係

| UCMDBバージョン       | 関係                          |
|------------------|-----------------------------|
| UCMDB 9.00       | RootとSM RDBMS間の [DB Client] |
| UCMDB 7.52 と 8.x | RootとSM RDBMS間の [SM Link]   |

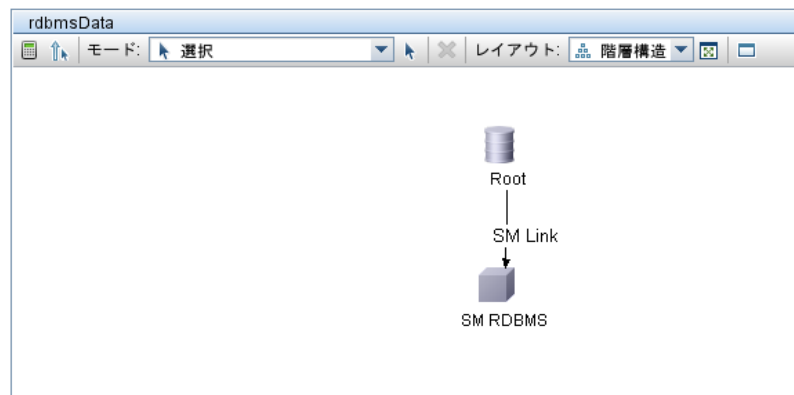
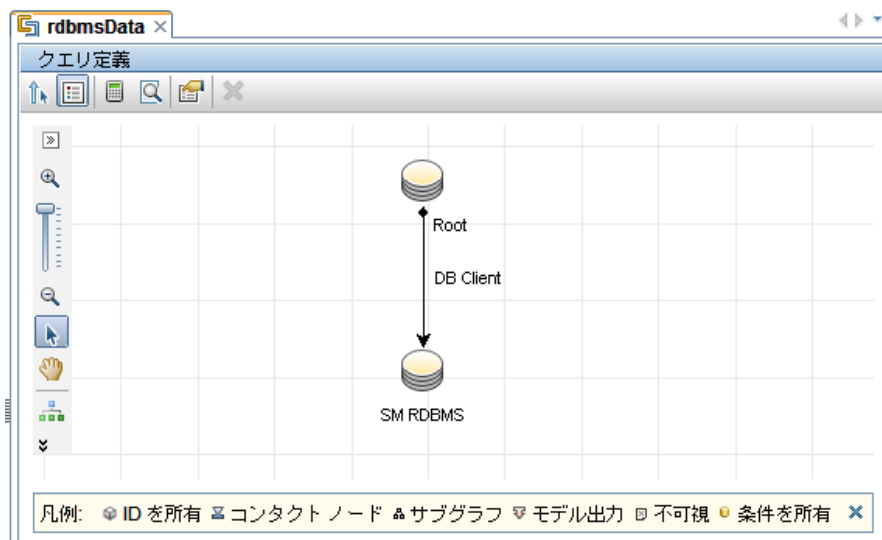
- a [Root] を選択して、Ctrlキーを押しながら追加のCIタイプをクリックします。[SM RDBMS] などです。
- b 選択した項目のいずれかを右クリックして、[関係の追加] をクリックします。[関係の追加] ウィンドウが開きます。
- c UCMDB 9.00の場合、[DB Client] を選択します。  
UCMDB 7.52 と 8.xの場合、[SM Link] をクリックします。
- d 関係の名前を入力します。「DB Client」や「SM Link」などです。
- e 関係がルートCIタイプから追加のCIタイプへの向きになっていることを確認します。RootからSM RDBMS、またはdatabaseからsm\_rdbmsなどです。
- f [OK] をクリックして、関係を追加します。



20 TQLに追加する追加のCIタイプそれぞれに、手順17から手順19を繰り返します。たとえば、SM RDBMSは追加のCIタイプは必要としません。

21 [保存] ボタンをクリックして、TQLクエリを保存します。 





## TQLクエリ内のすべての関係に追跡修飾子が存在することの確認

作成するTQLクエリのほとんどは、1つまたは複数のCIタイプ間クエリから構成されます。インテグレーションで、これらCIの履歴を追跡するため、TQLクエリ内の各関係にTRACK\_LINK\_CHANGES修飾子を追加する必要があります。インテグレーションが関係の更新を受信するようにするため、TQLクエリ内の各関係にTRACK\_LINK\_CHANGES修飾子があることを確認する必要があります。

次の手順では、関係にTRACK\_LINK\_CHANGES修飾子があることを確認する方法を示します。

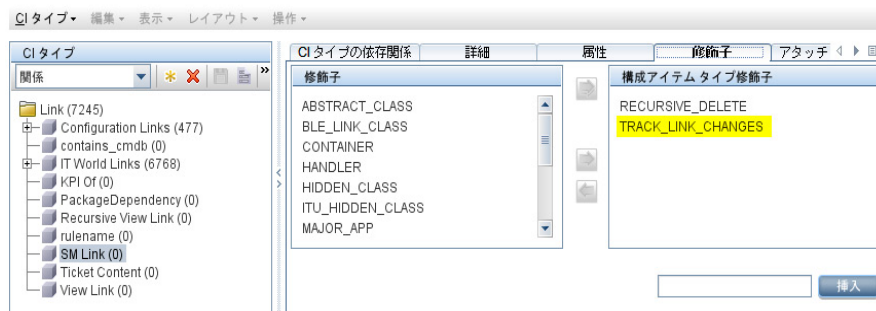
- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 [モデリング] タブから、[CIタイプマネージャ] を選択します。
- 3 [CIタイプ] から [関係] を選択します。

4 ナビゲーションツリーで、編集する関係を選択します。例：

表 29 ルートクエリと追加クエリ間の関係

| UCMDBバージョン       | 関係                                      |
|------------------|---|
| UCMDB 9.00       | [System Links] > [Parent] > [DB Client] |
| UCMDB 7.52 と 8.x | [Link] > [SM Link]                      |

- 5 [修飾子] タブを選択します。
- 6 [構成アイテムタイプ修飾子] 列にTRACK\_LINK\_CHANGES修飾子が含まれることを確認します。
- 7 関係にこの修飾子が含まれない場合、[修飾子] リストで [TRACK\_LINK\_CHANGES] を選択し、右矢印をクリックして [構成アイテムタイプ修飾子] セクションに移動します。



- 8 [保存] ボタンをクリックして、CI関係を保存します。

## CIタイプ属性の計算の有効化

インテグレーションにCI属性を追加するには、CIタイプを同期するTQLクエリから計算レイアウト設定を有効にする必要があります。インテグレーションに追加する各属性で計算を有効にする必要があることから、インテグレーションのCIタイプとそれに含まれるCI属性を理解しておく必要があります。



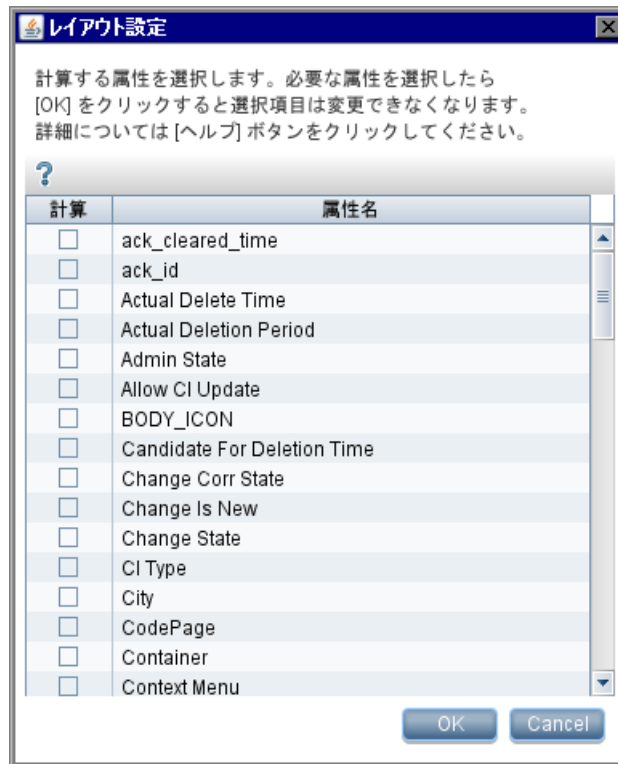
有効にする属性のリストを用意します。これは、属性ごとに一致するXSL変換を作成する必要があるためです。

次の手順では、前のセクションで説明したSM RDBMS CIタイプの属性の計算を有効にする方法を示します。

### CIタイプの属性の計算を有効にするには

- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 [モデリング] タブをクリックします。
- 3 UCMDB 9.00の場合、[モデリングスタジオ] を選択します。[リソースタイプ] ドロップダウンボックスから [クエリ] を選択します。  
UCMDB 7.52 と 8.xの場合、[クエリ マネージャ] をクリックします。
- 4 [クエリ] ナビゲーションツリーから、[Integration] > [SM Sync] をクリックします。

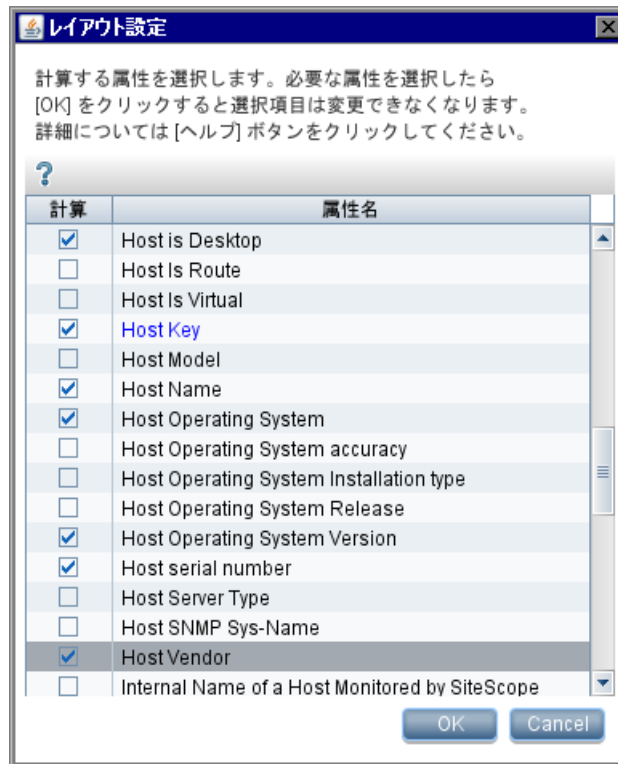
- 5 インテグレーションに追加する属性があるCIタイプを管理するクエリを選択します。[rdbmsData] などです。UCMDBには、インテグレーションクエリのためのTQLが表示されます。
- 6 インテグレーションに追加するCI属性を含むノードをTQLレイアウトから追加します。「Root」などです。
- 7 UCMDB 9.00の場合、ノードを右クリックして[クエリ ノードのプロパティ]を選択します。[クエリ ノードのプロパティ] ウィンドウが開きます。  
UCMDB 7.52と8.xの場合、[情報] ウィンドウで[編集] ボタンをクリックします。[ノードのプロパティ] ウィンドウが表示されます。
- 8 [詳細レイアウト設定] をクリックします。[レイアウト設定] ウィンドウが開きます。



- 9 インテグレーションに含める各CI属性の [計算] オプションを有効にします。例：


表 30 サンプルのCI属性に計算を有効にする

| UCMDBバージョン     | CI属性        |
|----------------|-------------|
| UCMDB 9.00     | OsVendor    |
| UCMDB 7.52と8.x | Host vendor |



10 [OK] をクリックして、[レイアウト設定] ウィンドウを閉じます。

11 [OK] をクリックして、ノードのプロパティを保存します。

12 [保存] ボタンをクリックして、TQL クエリを保存します。 

## CIタイプTQLクエリのSL変換ファイルへのマップ

インテグレーションでは、smSyncConfFile.xml という構成ファイルを使用して、各UCMDB TQL クエリをXSL変換ファイルにマップします。カスタムTQLクエリをインテグレーションに含めるには、各TQLクエリのマッピングエントリを構成ファイルに追加する必要があります。

次の手順では、Service Manager のucmdbApplication Web サービスに、前のセクションで説明したTQLクエリ rdbmsData をマップする方法を示します。

### XSL変換ファイルにTQLクエリをマップするには

- 1 UCMDB 9.00 の場合、[手順2](#)に移動します。  
UCMDB 7.52 と 8.x の場合、[手順4](#)に移動します。
- 2 [マネージャ] > [データフロー管理] > [アダプタ管理] をクリックします。
- 3 [パッケージ] > [ServiceManagerAdapter7-1] > [構成ファイル] > [ServiceManagerAdapter7-1/smSyncConfFile.xml] をクリックします。テキストエディタで構成ファイルが開きます。  
[手順7](#)に移動します。
- 4 管理者アカウントでUCMDBファイルシステムにログインします。
- 5 ファイルシステムでsmSyncConfFile.xmlパスを参照します。例：

C:\hp\UCMDB\UCMDBServer\j2fcmdb\CodeBase  
\ServiceDeskAdapter

- 6 テキストエディタを使用して、smSynchConfFile.xml 構成ファイルを開きます。
- 7 既存の TQL マッピング要素をコピーします。TQL マッピング要素は次の形式を使用します：

```
<tql name="TQL_query" xslFile="XSL_File">
  <!-- Description of mapping -->
  <request type="Create" name="Create_web_service"/>
  <request type="Update" name="Update_web_service"/>
  <request type="Delete" name="Delete_web_service"/>
</tql>
```

*TQL\_query* は、作成した UCMDB TQL クエリの名前です。

*XSL\_File* は、インテグレーションで Service Manager Web サービスオブジェクトに UCMDB 属性をマップするのに使用される XSL 変換ファイルの名前です。

*Create\_web\_service* は、インテグレーションでこの TQL クエリから CI を作成するのに使用する Service Manager Web サービスの名前です。

*Update\_web\_service* は、インテグレーションでこの TQL クエリの CI を更新するのに使用する Service Manager Web サービスの名前です。

*Delete\_web\_service* は、インテグレーションでこの TQL クエリから CI を削除するのに使用する Service Manager Web サービスの名前です。

図 5 smSynchConfFile.xml の一部

```
- <config>
- <global-config>
  <request type="Create" cmdb-id-only="false" />
  <request type="Update" cmdb-id-only="false" />
  <request type="Delete" cmdb-id-only="true" />
</global-config>
- <mapping>
- <tql name="hostData" xslFile="host_data.xslt">
  <!-- this is host->ip,interface,sm_server tql -->
  <request type="Create" name="CreateucmdbComputerRequest" />
  <request type="Update" name="UpdateucmdbComputerRequest" />
  <request type="Delete" name="DeleteucmdbComputerRequest" />
</tql>
- <tql name="applicationData" xslFile="application_data.xslt">
  <!-- this is logical_application tql -->
  <request type="Create" name="CreateucmdbApplicationRequest" />
  <request type="Update" name="UpdateucmdbApplicationRequest" />
  <request type="Delete" name="DeleteucmdbApplicationRequest" />
</tql>
- <tql name="businessServiceData" xslFile="business_service_data.xslt">
  <!-- this is business_service_for_catalog tql -->
  <request type="Create" name="CreateucmdbBusinessServiceRequest" />
  <request type="Update" name="UpdateucmdbBusinessServiceRequest" />
  <request type="Delete" name="DeleteucmdbBusinessServiceRequest" />
</tql>
```

- 8 インテグレーションに追加する各 TQL クエリの TQL マッピング要素を追加または更新します。たとえば、次の TQL では、rdbmsData TQL クエリと rdbms\_data.xslt ファイル間のマッピングが作成されます。

```
<tql name="rdbmsData" xslFile="rdbms_data.xslt">
  <!-- this is database tql -->
  <request type="Create" name="CreateucmdbApplicationRequest"/>
  <request type="Update" name="UpdateucmdbApplicationRequest"/>
```

```

    <request type="Delete" name="DeleteucmdbApplicationRequest" />
</tql>

```

```

- <config>
- <global-config>
  <request type="Create" cmdb-id-only="false" />
  <request type="Update" cmdb-id-only="false" />
  <request type="Delete" cmdb-id-only="true" />
</global-config>
- <mapping>
- <tql name="hostData" xslFile="host_data.xslt">
  <!-- this is host->ip,interface,sm_server tql -->
  <request type="Create" name="CreateucmdbComputerRequest" />
  <request type="Update" name="UpdateucmdbComputerRequest" />
  <request type="Delete" name="DeleteucmdbComputerRequest" />
</tql>
- <tql name="applicationData" xslFile="application_data.xslt">
  <!-- this is logical application tql -->
  <request type="Create" name="CreateucmdbApplicationRequest" />
  <request type="Update" name="UpdateucmdbApplicationRequest" />
  <request type="Delete" name="DeleteucmdbApplicationRequest" />
</tql>
- <tql name="rdbmsData" xslFile="rdbms_data.xslt">
  <!-- this is database tql -->
  <request type="Create" name="CreateucmdbApplicationRequest" />
  <request type="Update" name="UpdateucmdbApplicationRequest" />
  <request type="Delete" name="DeleteucmdbApplicationRequest" />
</tql>

```

9 構成ファイルを保存します。

## CIタイプ属性のWebサービスオブジェクトへのマップ

インテグレーションは、Service Desk アダプタを使用することで、UCMDB CI属性を Service Manager が認識する Web サービスオブジェクトに変換します。Service Desk アダプタは、XSL 変換ファイルを使用して、UCMDB TQL クエリを適切にフォーマットされた Service Manager Web サービスメッセージに変換します。出荷時設定では、インテグレーションクエリごとに対応する XSL 変換ファイルがあります。さらに、[詳細レイアウト設定] から計算を有効にする属性はそれぞれ、XSL 変換ファイル内に独自のエントリを必要とします。

インテグレーションに CI タイプを追加する場合、Service Desk アダプタが Service Manager Web サービスオブジェクトに各 CI タイプを変換する方法を定義する、一致する XSL 変換ファイルを作成する必要があります。各クエリが管理する CI タイプの詳細については、[インテグレーション TQL クエリ](#) (51 ページ) を参照してください。適切な XSL マッピングを作成するため、Service Manager が Web サービスとして公開するサービスとオブジェクト名を理解しておく必要があります。Web サービスオブジェクトとして公開するテーブルと列の詳細については、Service Manager ヘルプを参照してください。

次の手順では、前のセクションで説明した rdbmsData TQL クエリの XSL 変換ファイルを作成する方法を説明します。

**Web サービスオブジェクトに CI タイプをマップするには：**

- 1 管理者アカウントで UCMDB ファイルシステムにログインします。
- 2 ファイルシステムで Service Desk アダプタのパスを参照します。例：

```

C:\hp\UCMDB\UCMDBServer\j2fcmdb\CodeBase
\ServiceDeskAdapter

```

- 3 既存のXSL変換ファイルをコピーして、新しい変換ファイルを作成するためのテンプレートとして使用します。たとえば、application\_data.xsltをコピーします。
- 4 新規変換ファイルを一意の名前に変更します。rdbms\_data.xsltなどです。
- 5 テキストエディタを使用して、新規XSL変換ファイルを開きます。
- 6 CIタイプ定義要素を見つけます。CIタイプ定義要素は次の形式を使用します：

```
<xsl:template match="/CI_type_name">
```

@CI\_type\_nameは、UCMDBシステム内でのCIタイプの名前です。

図6 application\_data.xslt内のCIタイプ定義

```
- <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
- <xsl:template match="/logical_application">
- <model>
  <keys />
  - <instance>
    <!-- import:cmdb_root_attributes.xslt -->
    - <xsl:for-each select="@friendlyType">
      - <Subtype>
        <xsl:value-of select="." />
      </Subtype>
    </xsl:for-each>
    - <xsl:for-each select="@data_name">
      - <ApplicationName>
        <xsl:value-of select="." />
      </ApplicationName>
    </xsl:for-each>
  </instance>
</model>
</xsl:template>
</xsl:stylesheet>
```

- 7 インテグレーションに追加するCIタイプと一致するようにCIタイプ名を更新します。たとえば、次のCIタイプ定義要素を作成して、インテグレーションにデータベースCIタイプを追加します。

```
<xsl:template match="/database">
```

- 8 必要に応じて、テーブル命名要素を追加または更新します。デフォルトで、UCMDBはService ManagerのdeviceテーブルにCI属性データを送信します。deviceのいずれかの結合テーブルにCI属性を送信する場合、<file.table\_name>というフォーマットを使用して、テーブル名を指定する要素を追加する必要があります。たとえば、database CIタイプを定義する場合、追加のjointableを指定する必要はありません。Service Managerは、database CIタイプを管理するのに独立したjointableを使用しないためです。

- 9 Service Manager WebサービスオブジェクトにUCMDB CI属性を変換する要素を見つけます。CI属性変換要素は次の形式を使用します：

```
<xsl:for-each select="@CI_attribute_name">
  <ObjectName><xsl:value-of select="." /></ObjectName>
</xsl:for-each>
```

@CI\_attribute\_nameは、UCMDBシステム内での属性の名前です。

ObjectNameは、Service Managerシステムで公開されるWebサービスオブジェクトの名前です。

図7 application\_data.xslt内のCI属性

```

- <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
- <xsl:template match="/logical_application">
- <model>
  <keys />
  - <instance>
    <!-- import:cmdb_root_attributes.xslt -->
    - <xsl:for-each select="@friendlyType">
      - <Subtype>
        <xsl:value-of select="." />
      </Subtype>
    </xsl:for-each>
    - <xsl:for-each select="@data_name">
      - <ApplicationName>
        <xsl:value-of select="." />
      </ApplicationName>
    </xsl:for-each>
  </instance>
</model>
</xsl:template>
</xsl:stylesheet>

```

図 8 ucmdbApplication Web サービスにある一致するCI属性

**外部アクセス定義**

サービス名:   リリース日  
 名前:   廃止予定  
 オブジェクト名:

◆ 許可されるアクション ◆ 式 ◆ フィールド

| フィールド    | キャプション          | タイプ |
|----------|-----------------|-----|
| ucmdb.id | UCMDBId         |     |
| ciname   | ApplicationName |     |
| subtype  | Subtype         |     |
|          |                 |     |
|          |                 |     |

10 インテグレーションに追加する各CI属性のCI属性変換要素を追加または更新します。たとえば、database CIタイプに次のXSL変換要素を作成します。

表 31 データベースCIのサンプルXSL変換要素

| UCMDB属性        | サンプル変換要素  |
|----------------|---|
| application_ip | <pre> &lt;xsl:for-each select="@application_ip"&gt; &lt;DBIP&gt;&lt;xsl:value-of select="." /&gt;&lt;/DBIP&gt; &lt;/xsl:for-each&gt; </pre> |



表 31 データベース CI のサンプル XSL 変換要素

| UCMDB 属性           | サンプル変換要素  |
|--------------------|---|
| database_dbtype    | <pre>&lt;xsl:for-each select="@database_dbtype"&gt;   &lt;DBType&gt;&lt;xsl:value-of select="."/ &gt;&lt;/DBType&gt; &lt;/xsl:for-each&gt;</pre>          |
| database_dbversion | <pre>&lt;xsl:for-each select="@database_dbversion"&gt;   &lt;DBVersion&gt;&lt;xsl:value-of select="."/ &gt;&lt;/DBVersion&gt; &lt;/xsl:for-each&gt;</pre> |
| user_label         | <pre>&lt;xsl:for-each select="@user_label"&gt;   &lt;UserLabel&gt;&lt;xsl:value-of select="."/ &gt;&lt;/UserLabel&gt; &lt;/xsl:for-each&gt;</pre>         |

図 9 rdbms\_data.xslt 内の新規属性マッピング

```
- <xsl:for-each select="@data_name">
- <ApplicationName>
  <xsl:value-of select="." />
</ApplicationName>
</xsl:for-each>
- <xsl:for-each select="@application_ip">
- <DBIP>
  <xsl:value-of select="." />
</DBIP>
</xsl:for-each>
- <xsl:for-each select="@database_dbtype">
- <DBType>
  <xsl:value-of select="." />
</DBType>
</xsl:for-each>
- <xsl:for-each select="@database_dbversion">
- <DBVersion>
  <xsl:value-of select="." />
</DBVersion>
</xsl:for-each>
- <xsl:for-each select="@user_label">
- <UserLabel>
  <xsl:value-of select="." />
</UserLabel>
</xsl:for-each>
```

11 XSL 変換ファイルを保存します。

## ServiceDeskAdapter のコードベースの再ロード

この説明は、UCMDB 7.52 または 8.x システムを使用したインテグレーションにのみ適用されます。UCMDB 9.00 ではアダプタを手動で再ロードする必要はありません。

インテグレーションでカスタムの TQL クエリと XSL 変換ファイルを使用するには、ServiceDeskAdapter のコードベースを再ロードする必要があります。

ServiceDeskAdapter のコードベースを再ロードするには：

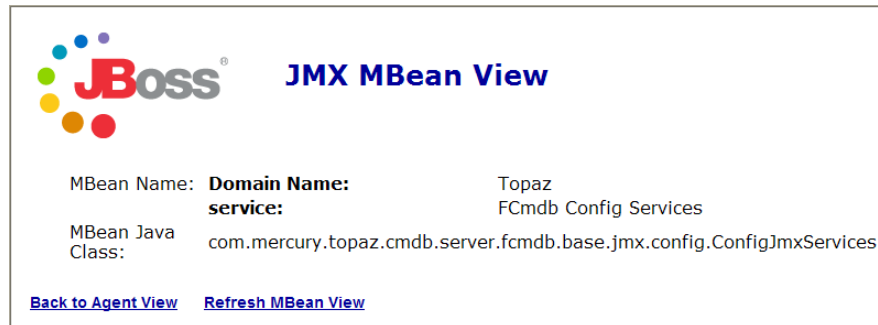
- 1 Web ブラウザを開いて、次の URL にアクセスします：

**http://<UCMDB サーバ>:8080/jmx-console/  
HtmlAdaptor?action=inspectMBean&name=Topaz%3Aservice%3  
DFCmdb+Config+Services**

<UCMDB サーバ>は、UCMDBシステムの完全修飾ドメイン名です。

- 2 入力が求められたら、UCMDB管理者のユーザ名とパスワードを入力します。  
JMX MBean ViewのWebページが開きます。

図 10 JMX MBean View



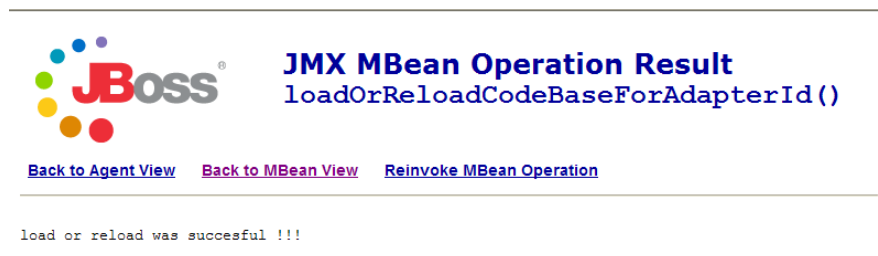
- 3 loadOrReloadCodebaseForAdapterId() MBeanのセクションまで移動します。
- 4 customerIDのParam Valueに「1」と入力します。
- 5 adapterIdのParam Valueに「ServiceDeskAdapter」と入力します。

### java.lang.String loadOrReloadCodeBaseForAdapterId()

Load or reload adapter code base

| Param      | Param Type       | Param Value                                     | Param Description |
|------------|------------------|---|-------------------|
| customerID | int              | <input type="text" value="1"/>                  | Customer id       |
| adapterId  | java.lang.String | <input type="text" value="ServiceDeskAdapter"/> | Adapter id        |


- 6 関数の下にある [Invoke] ボタンをクリックします。成功のウィンドウが開きます。

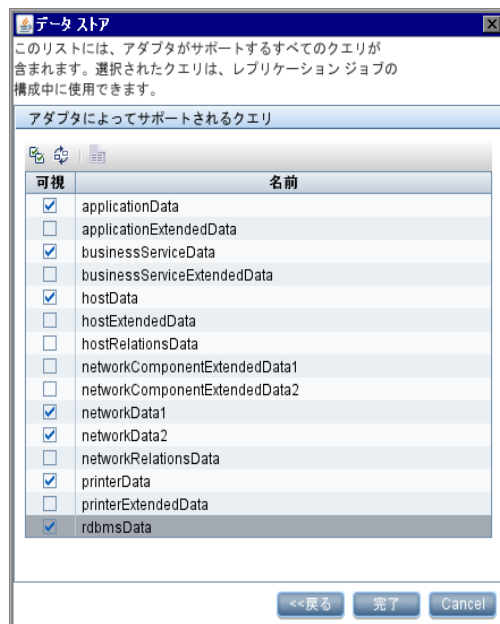


## インテグレーション変更アダプタへのカスタムTQLクエリの追加

インテグレーションでカスタムCIタイプと属性の値を更新するには、変更アダプタにカスタムTQLクエリを追加する必要があります。次の手順では、前のセクションで説明した rdbmsData TQL クエリを変更アダプタに追加する方法を示します。

### 変更アダプタにカスタムTQLクエリを追加するには

- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 [設定] タブをクリックします。
- 3 [データストア] タブを選択します。
- 4 インテグレーションをサポートするために作成した変更アダプタを選択します。 [cmdbChanges] などです。
- 5 [編集] ボタンをクリックします。 
- 6 [データストア] プロパティウィンドウが表示されます。
- 7 [次へ] をクリックして、[アダプタによってサポートされるクエリ] ページに移動します。
- 8 カスタムTQLクエリの [可視] オプションを有効にします。「rdbsData」などです。



- 8 [完了] をクリックして、変更アダプタを更新します。

## インテグレーションデータプッシュ (レプリケーションジョブ) への カスタムTQLクエリの追加

インテグレーションでService ManagerシステムにカスタムのCIタイプと属性を送信する場合、インテグレーションデータプッシュ (レプリケーションジョブ) にカスタムTQLクエリを追加する必要があります。次のステップでは、前のセクションで説明したカスタムrdbsData TQLクエリを追加する方法を示します。

### UCMDB 9.00データプッシュジョブの定義


#### データプッシュジョブ定義にカスタムTQLクエリを追加するには

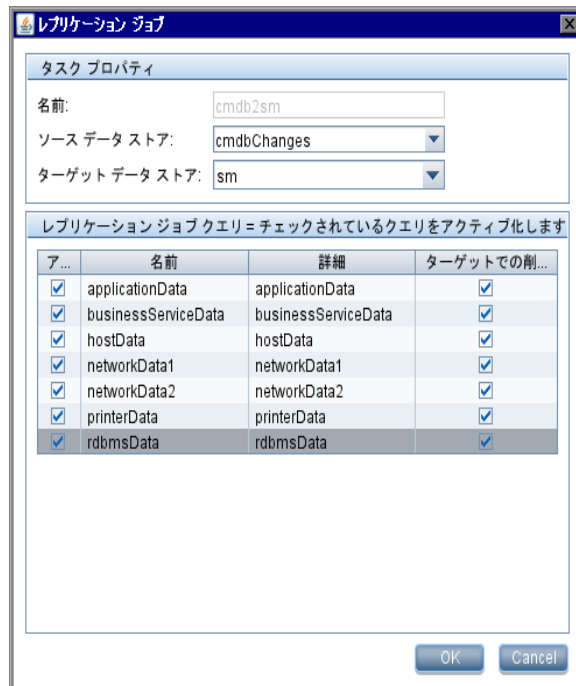
- 1 管理者としてHP Universal CMDBシステムにログインします。
- 2 [マネージャ] > [データフロー管理] > [Integration Studio] をクリックします。
- 3 Service Manager インテグレーションポイントの名前をクリックします。「SM Integration」などです。

- 4 [データ プッシュ] タブを選択します。
- 5 変更レプリケーションジョブの名前をクリックします。「Chnages」などです。
- 6 [編集] ボタンをクリックします。 
- 7 [追加] ボタン  をクリックします。
- 8 [Integration] > [SM Sync] > [rdbmsData] をクリックします。
- 9 [OK] をクリックして、カスタムクエリを追加します。
- 10 クエリの [削除を許可] オプションを選択します。
- 11 [OK] をクリックして、[ジョブ定義の更新] ウィンドウを閉じます。

## UCMDB 7.52 と 8.x のレプリケーションジョブ

レプリケーションジョブにカスタムTQLクエリを追加するには

- 1 管理者として HP Universal CMDB システムにログインします。
  - 2 [設定] タブをクリックします。
  - 3 [レプリケーション ジョブ] タブをクリックします。
  - 4 変更データストアと Service Manager データストア間で CI を同期するために作成したレプリケーションジョブを選択します。[cmdb2sm] などです。
  - 5 [編集] ボタンをクリックします。 
- [レプリケーション ジョブ] ウィンドウが表示されます。
- 6 カスタム TQL クエリの [ターゲットでの削除を許可] オプションを有効にします。「rdbmsData」などです。



- 7 [OK] をクリックして、レプリケーションジョブを更新します。

## CIタイプをサポートするWebサービスオブジェクトの作成

インテグレーションにCIタイプを追加するには、作成したXSL変換の対象となる、UCMDBシステムの各CI属性に対応するService Manager Webサービスオブジェクトを作成する必要があります。Service Managerは、Webサービスオブジェクトを使用して、受信するCI属性値の格納先となるService Managerテーブルと列を特定します。

次の手順では、前のセクションで説明したSM RDBNS CIタイプをサポートするのに必要となる、Webサービスオブジェクトを作成する方法について説明します。

## 新規CIタイプをサポートするWebサービスオブジェクトを作成するには

- 1 Service Manager サーバに管理者アカウントでログインします。
- 2 **[カスタマイズ]** > **[Web サービス]** > **[WSDL 設定]** をクリックします。レコードを検索するためのフォームが表示されます。
- 3 [サービス名] に「**ucmdbIntegration**」と入力して **[検索]** をクリックします。インテグレーションに含まれるすべてのWebサービスオブジェクトが表示されます。
- 4 受信するCIデータを管理するのに使用するWebサービスオブジェクトを選択します。**[ucmdbApplication]** などです。



受信するCI属性の格納先となるService Manager テーブルを使用するWebサービスオブジェクトを選択します。たとえば、ucmdbApplication はService Manager の device テーブルから列を公開します。これは出荷時設定でRDBMS CIに最適な設定になっています。

- 5 [フィールド] タブをクリックします。
- 6 [フィールド] リストで空欄の行を選択します。
- 7 [フィールド] で、受信するCI属性データを格納するのに使用するService Manager を選択します。
- 8 [キャプション] に、XSL変換ファイルで使用したWebサービスオブジェクト名を入力します。



Webサービスオブジェクト名は、XSL変換ファイルで使用した名前と一致する必要があります。名前が一致しない場合、Service ManagerはUCMDBシステムからCI更新を受信しません。

たとえば、SM RDBMS CIタイプをサポートするには、次のフィールド値とキャプション値を入力します。

表 32 SM RDBMS CIタイプのWebサービスオブジェクト定義

| フィールド           | キャプション    |
|-----------------|-----------|
| admin.urlport   | DBIP      |
| addl[addl.type] | DBType    |
| product.version | DBVersion |
| addl[addl.name] | UserLabel |

- 9 **[保存]** をクリックします。

## 外部アクセス定義

|          |   |                                |
|----------|---|--------------------------------|
| サービス名:   | <input type="text" value="ucmdbIntegration"/> | <input type="checkbox"/> リリース日 |
| 名前:      | <input type="text" value="device"/>           | <input type="checkbox"/> 廃止予定  |
| オブジェクト名: | <input type="text" value="ucmdbApplication"/> |                                |

| フィールド           | キャプション          | タイプ |
|-----------------|-----------------|-----|
| ucmdb.id        | UCMDBId         |     |
| ciname          | ApplicationName |     |
| subtype         | Subtype         |     |
| admin.urlport   | DBIP            |     |
| addl[addl.type] | DBType          |     |
| product.version | DBVersion       |     |
| addl[addl.name] | UserLabel       |     |
|                 |                 |     |
|                 |                 |     |

- 10 Service Manager サーバを停止して、再起動します。インテグレーションで、新規 Web サービスオブジェクトが利用できるようになります。

## CIタイプをサポートするための管理フィールドの追加

カスタム CI タイプが、自動化された変更管理の確認と検証プロセスをトリガするには、その CI タイプ内の各 CI 属性の管理フィールドを追加する必要があります。Service Manager 管理フィールドは、検出イベントマネージャールールの構成要素です。変更管理の確認と検証をトリガするフィールドのリストについては、[Service Manager 検出イベントマネージャールール](#)（59 ページ）を参照してください。

次の手順では、前のセクションで説明した SM RDBMS CI タイプの管理フィールドを追加する方法を示します。

CIタイプをサポートするための管理フィールドを追加するには：

- 1 Service Manager サーバに管理者アカウントでログインします。
- 2 **[カスタマイズ]** > **[Web サービス]** > **[検出イベントマネージャールール]** をクリックします。レコードを検索するためのフォームが表示されます。
- 3 **[検索]** をクリックして、すべての検出イベントマネージャールールのリストを表示します。
- 4 受信する CI 属性をマップした Web サービスオブジェクトと一致するルール ID を選択します。**[ucmdbComputer]** などです。この Web サービスオブジェクトのルールが表示されます。
- 5 **[管理フィールド]** タブをクリックします。変更管理の確認と検証をトリガするフィールドのリストが表示されます。
- 6 **[管理フィールド]** リストで空欄の行を選択します。
- 7 **[フィールド名]** で、先に選択した、受信する CI 属性値の格納先となる Service Manager 列のキャプション名を選択します。**[OS 製造業者]** などです。



Service Manager に、**[テーブル名]** フィールドに一覧されるテーブルに関連付けられているすべての結合テーブルのフィールドが表示されます。たとえば joincomputer には、device テーブルと computer テーブルのフィールドが表示されます。

- 8 受信CI属性を格納するのに使用するフィールドが配列構造である場合、[構造] フィールドを使用して、列が存在する構造体の配列の名前を選択します。たとえば、[OS 製造業者]はプリミティブな文字フィールドであるため、配列構造の名前を指定する必要はありません。
- 9 受信CI属性を格納するのに使用するフィールドが配列構造である場合、[インデックス]フィールドを使用して、構造体の配列内にある列を特定するインデックス番号を選択します。たとえば、[OS 製造業者]はプリミティブな文字フィールドであるため、配列構造のインデックスを指定する必要はありません。
- 10 **[保存]** をクリックします。

### 検出イベントマネージャールール

ID:

テーブル名:

条件:

| フィールド名       | 構造        | インデックス | 条件 |
|--------------|-----------|--------|----|
| OS           |           |        |    |
| デフォルトゲートウェイ  |           |        |    |
| OS バージョン     |           |        |    |
| BIOS ID      |           |        |    |
| ネットワーク名      |           |        |    |
| addIPAddress | addIPAddr | 1      |    |
| addSubnet    | addIPAddr | 2      |    |
| 追加 MAC アドレス  |           |        |    |
| 物理メモリ合計      |           |        |    |
| マシン名         |           |        |    |
| サブタイプ        |           |        |    |
| ビル名          |           |        |    |
| 部屋番号         |           |        |    |
| 階            |           |        |    |
| UCMDB ID     |           |        |    |
| OS 製造業者      |           |        |    |

- 11 Service Manager サーバを停止して、再起動します。インテグレーションで、新規管理フィールドが利用できるようになります。