CA Performance Management Data Aggregator

アップグレード ガイド - サイレント モード

2.4

ca technologies

CA への連絡先

テクニカル サポートの詳細については、弊社テクニカル サポートの Web サイト（http://www.ca.com/jp/support/）をご覧ください。

目次

第 1 章: アップグレード要件および考慮事項

このセクションには、以下のトピックが含まれています。

サポートされているアップグレード パス

Data Aggregator の以前のリリースからアップグレードする場合は、ご使用のコンポーネントをアップグレードします。 CA Performance Center、Data Aggregator、および Data Collector のコンポーネントは常にアップグレードする必要があります。 以下の表で識別されたリリースにアップグレードしている場合、Data Repository をアップグレードします。

重要: リリース 2.0.00 からリリース 2.3.4 にアップグレードする場合は、最初に、リリース 2.1.00 に、続けてリリース 2.2.x にアップグレードした後、リリース 2.3 にアップグレードします。

以下の表に、サポートされるアップグレード パスと、アップグレードが必要なコンポーネントを示します。

リリース	CA Performance Center コンポーネント	Data Aggregator コンポーネント	Data Collector コンポーネント	Data Repository コンポーネント
リリース 2.0.00 からリリース 2.1.00	アップグレードが必要	アップグレードが必要	アップグレードが必要	アップグレードは *不要*
リリース 2.1.00 からリリース 2.2.00	アップグレードが必要	アップグレードが必要	アップグレードが必要	アップグレードが必要
リリース 2.2.00 からリリース 2.2.1	アップグレードが必要	アップグレードが必要	アップグレードが必要	アップグレードは *不要*

リリース	CA Performance Center コンポーネント	Data Aggregator コンポーネント	Data Collector コンポーネント	Data Repository コンポーネント
リリース 2.2.00/2.2.1 から 2.2.2	アップグレードが必要	アップグレードが必要	アップグレードが必要	アップグレードが必要
リリース 2.2.[1、2、3] から 2.3.[0、1、2、3]	アップグレードが必要	アップグレードが必要	アップグレードが必要	アップグレードは*不要*
リリース 2.2.x から 2.3.4	アップグレードが必要	アップグレードが必要	アップグレードが必要	アップグレードが必要 **注**: Vertica Release 7 はリリース 2.3.4 で導入されています。
リリース 2.3.[0、1、2、3] から 2.3.4	アップグレードが必要	アップグレードが必要	アップグレードが必要	アップグレードが必要 **注**: Vertica Release 7 はリリース 2.3.4 で導入されています。

注: Data Aggregator コンポーネントのアップグレードの詳細については、「Data Aggregator インストール ガイド」を参照してください。 2.3.x より前のリリース用のアップグレード要件および考慮事項の詳細については、アップグレードしているリリースの「リリース ノート」または「修正された問題」ファイルを参照してください。

Data Repository アップグレードの準備方法

Data Repository をインストールする前に、以下の前提条件を満たしている必要があります。

1. Data Repository のインストール先コンピュータに少なくとも 2 GB のスワップ領域があることを確認します。
2. データおよびカタログのディレクトリに ext3 または ext4 ファイル システムを使用していることを確認します。

3. データおよびカタログのディレクトリに論理ボリューム マネージャ（LVM）を使用していないことを確認します。

第 2 章: アップグレード

このセクションには、以下のトピックが含まれています。

CA Performance Management Data Aggregator のアップグレード方法 - サイレント モード

Data Aggregator の以前のリリースからアップグレードする場合は、ご使用のコンポーネントをアップグレードします。

注: 各コンポーネント用の *installation_directory*/logs ディレクトリ内の .history ファイルを調べることにより、製品のどのバージョンがインストールされているかを確認できます。

以下の手順を、推奨される記載の順序に従って実行します。

1. Data Aggregator プロセスの自動復旧を無効にします (P. 12)。
2. Data Collector を停止して、Data Aggregator を停止します (P. 13)。
3. CA Performance Center をインストールしているユーザの ulimit 値が 65536 以上であることを確認します。

   注: ulimit 値の設定の詳細については、「CA Performance Center インストール ガイド」を参照してください。
4. CA Performance Center をアップグレードします。

   注: CA Performance Center のアップグレードの詳細については、「CA Performance Center インストール ガイド」を参照してください。
5. Data Repository ホスト上のオープン ファイルの上限を確認します (P. 13)。

6. Data Repository をアップグレードします (P. 15)。

   注: Data Repository のアップグレードは、CA Performance Management Data Aggregator の特定のリリースにアップグレードしている場合にのみ実行してください。 サポートされているアップグレード パスと、Data Repository をアップグレードするタイミングの詳細については、「Data Aggregator リリース ノート」を参照してください。

7. Data Aggregator をインストールしているユーザの ulimit 値が 65536 以上であることを確認します。 (P. 20)
8. すべてのデータベース テーブル予測がセグメント化されていることを確認します (P. 21)。
9. セグメント データベース テーブル予測をセグメント化します (P. 22)。
10. Data Aggregator をアップグレードします (P. 30)。
11. Data Collector をアップグレードします (P. 33)。
12. Data Aggregator プロセスの自動復旧を再度有効化します (P. 39)。
13. アップグレード後の手順を実行します (P. 39)。

Data Aggregator プロセスの自動復旧の無効化

Data Aggregator のアップグレード前に、Data Aggregator プロセスの自動復旧を無効化します。 あらかじめ停止しておくことで、cron ジョブによってシステムが中断されることなくアップグレードを実行できます。

次の手順に従ってください:

1. Data Aggregator がインストールされているコンピュータに root ユーザとしてログインします。
2. コンソールを開き、以下のコマンドを入力します。

   ```
   crontab -e
   ```

   vi セッションが開始されます。
3. 以下の行をコメントアウトします。

   ```
   * * * * * /etc/init.d/dadaemon start > /dev/null
   ```

   たとえば、以下のようになります。

   ```
   # * * * * * /etc/init.d/dadaemon start > /dev/null
   ```

   Data Aggregator プロセスの自動復旧が無効化されます。

Data Collector および Data Aggregator の停止

Data Aggregator をアップグレードする前に、インストールされている Data Collector および Data Aggregator を停止します。

次の手順に従ってください:

1. Data Collector がインストールされている各コンピュータで、コマンド プロンプトを開き、以下のコマンドを入力します。

   ```
   /etc/init.d/dcmd stop
   ```

2. Data Aggregator がインストールされているコンピュータで、コマンド プロンプトを開き、以下のコマンドを入力します。

   ```
   /etc/init.d/dadaemon stop
   ```

Data Repository ホスト上のオープン ファイル数の上限を確認

Data Repository をインストールしているユーザのオープン ファイル数が 65536 以上の値であることを確認します。 この値を永続的に設定します。

注: クラスタ環境では、オープン ファイル数の値がすべてのノード上で同じである必要があります。

次の手順に従ってください:

1. root ユーザまたは sudo ユーザとして、Data Repository をインストールする各コンピュータにログインします。 コマンド プロンプトを開き、以下のコマンドを入力して、オープン ファイル数が正しく設定されていることを確認します。

   ```
   ulimit -n
   ```

   このコマンドは ulimit 数を返します。 この数は 65536 以上である必要があります。

2. この数が 65536 より小さい場合は、以下の手順を実行します。

   a. コマンド プロンプトを開いて次のコマンドを入力し、ulimit のオープン ファイルの上限を少なくとも 65536 に変更します。

      ```
      ulimit -n ulimit_number
      ```

      例 :

      ```
      ulimit -n 65536
      ```

b. Data Repository をインストールする各コンピュータ上で /etc/security/limits.conf ファイルを開き、以下の行を追加します。

```
# Added by Vertica
* soft nofile 65536
# Added by Vertica
* hard nofile 65536
# Added by Vertica
*       soft    fsize   unlimited
# Added by Vertica
*       hard    fsize   unlimited
```

c. Data Repository をインストールする予定の各コンピュータで、以下のコマンドを入力します。

```
service sshd restart
```

注: 引数として restart を指定しない場合、次のコマンドを入力して sshd の停止と起動を行います。

```
service sshd stop
```

```
service sshd start
```

d. Data Repository をインストールする各コンピュータ上で、オープン ファイルの数が適切に設定されていることを確認するには、以下のコマンドを入力します。

```
ulimit -n
```

このコマンドは、以前に指定した ulimit 値を返します。

e. （クラスタ インストールのみ）root ユーザまたは sudo ユーザとしてログインします。 あるノードから別のノードへ ssh 接続して、オープン ファイルの数が各コンピュータ上で正しく設定されていることを確認します。

```
ulimit -n
```

このコマンドは、以前に指定した ulimit 値を返します。

Data Repository をインストールしているユーザの ulimit 値は、少なくとも 65536 であることが必要です。ulimit 値が永続的に設定されます。 Data Repository がインストールされているコンピュータが再起動された場合でも、この値の設定は維持されます。

f. いずれかのホスト上で ulimit 値が少なくとも 65536 に設定されていない場合は、そのホスト上で以下の手順を実行します。

/etc/security/limits.conf ファイルを開き、以下の行を追加します。

```
# Added by Vertica
* soft nofile 65536
# Added by Vertica
* hard nofile 65536
```

Data Repository ホスト上のオープン ファイル数の上限が確認されました。

Data Repository のアップグレード

Data Repository をアップグレードします。アップグレード処理の一部として、以下のスクリプトを順番に実行する必要があります。

- dr_validate.sh - Data Repository の前提条件が満たされていることを確認するのに役立ちます。
- dr_install.sh - Vertica データベースをインストールします。

注: サポートされているアップグレード パスと、Data Repository をアップグレードするタイミングの詳細については、「Data Aggregator リリース ノート」を参照してください。

次の手順に従ってください:

1. Data Repository に使用するデータベース サーバに Vertica Linux データベース管理者ユーザとしてログインし、Vertica が稼働しているホストを判断します。
   a. 以下のコマンドを入力します。

   ```
   /opt/vertica/bin/adminTools
   ```

   ［Administration Tools］ダイアログ ボックスが表示されます。
   b. オプション 6 （Configuration Menu）を選択します。
   c. オプション 3 （View Database）を選択します。
   d. データベースを選択します。
   e. ホスト名をメモしておきます。 この手順の後半で、これらのホスト名が必要になります。
   f. adminTools ユーティリティを終了します。

2. コンソールを開き、root ユーザとして Data Repository をインストールするコンピュータにログインします。sudo インストール手順が必要な場合は、CA サポートにお問い合わせください。

   重要: クラスタ インストールでは、クラスタに含まれている 3 つのホストのいずれからでも、Data Repository のインストールを開始できます。 必要なソフトウェア コンポーネントは、インストール中に他の 2 つのノードにプッシュされます。

3. installDR.bin ファイルをローカルにコピーします。

4. 以下のコマンドを入力して、インストール ファイルの権限を変更します。

   ```
   chmod u+x installDR.bin
   ```

5. インストール ファイルを展開するには、以下のコマンドを入力します。

   ```
   ./installDR.bin
   ```

   重要: installDR.bin ファイルは Data Repository をインストールしません。 このファイルは、Data Repository の rpm およびライセンス ファイルを解凍します。 Data Repository は、後続の手順でインストールします。

   installDR.bin インストール中に選択されたディレクトリには、すべてのユーザがアクセスできる必要があります。 **chmod** を使用して、ユーザ ホーム ディレクトリ内のディレクトリの読み取り/書き込みを有効にすることができます（たとえば chmod -R 755）。

   使用許諾契約が表示されます。

   Data Repository のインストール ファイルをセキュア シェルまたはコンソールから解凍し、Data Repository をインストールするコンピュータで X Window System を実行していない場合、使用許諾契約はコンソール モード（コマンド ライン）で表示されます。それ以外の場合、使用許諾契約はユーザ インターフェースに表示されます。

6. ユーザ インターフェース内にいる場合は、使用許諾契約を読み、使用許諾契約に同意して、［次へ］をクリックします。 コンソール モードの場合は Enter キーを押します。

7. プロンプトが表示されたら、Data Repository のインストール パッケージおよび Vertica ライセンス ファイルを展開するインストール ディレクトリを入力するか、またはデフォルトのインストール ディレクトリ /opt/CA/IMDataRepository_vertica7/ をそのまま使用します。 ユーザ インターフェース内にいる場合は、［インストール］をクリックし、［完了］をクリックします。コンソール モードの場合は、Enter キーを 2 回押します。

   Data Repository のインストール パッケージおよびライセンス ファイルが、選択したディレクトリに解凍されます。 インストールを完了するのに必要な 3 つのインストール スクリプトも解凍されます。

8. Data Repository の手動バックアップを実行するには、以下のコマンドを入力します。

   ```
   /opt/vertica/bin/vbr.py --task backup --config-file configuration_filename
   ```

   configuration_filename

   自動バックアップを最初にセットアップしたときに作成した設定ファイルのディレクトリ パスとファイル名を示します。 このファイルは、バックアップ ユーティリティを実行した場所（/opt/vertica/bin/vbr.py）にあります。

   以下に例を示します。

   ```
   /opt/vertica/bin/vbr.py --task backup --config-file
   /home/vertica/vert-db-production.ini
   ```

   ホストの信ぴょう性に関するプロンプトが表示された場合は、「はい」と回答します。

   注: クラスタ インストールでは、クラスタに含まれているホストのいずれかにのみ、この手順を実行します。

   重要: Data Repository をバックアップする場合、それまでに Data Repository を定期的にバックアップしていない場合は、Data Repository のバックアップに数時間かそれ以上かかることがあります。

9. 使用可能な場合は、/opt/CA/IMDataRepository_vertica6 から /opt/CA/IMDataRepository_vertica7 に既存の drinstall.properties をコピーします。

   注: 上記のパスはデフォルト値です。 実際の場所は異なる可能性があります。

10. drinstaller.properties ファイル内のパラメータがすべて正しいことを確認します。 以下のパラメータを確認します。

    - DbAdminLinuxUser=*Vertica データベース管理者として使用される Linux ユーザ*

      デフォルト： dradmin

    - DbAdminLinuxUserHome=*Vertica Linux データベース管理者ユーザ ホーム ディレクトリ*

      デフォルト： /export/dradmin

    - DbDataDir= データ ディレクトリの場所

      デフォルト： /data

      注: データ ディレクトリがどれかはっきりわからない場合は、次の手順に従います： /opt/vertica/config/admintools.conf ファイルを開きます。 ［Nodes］セクションまでスクロールします。 v_*dbname*_node*XXXX* で始まる行の 1 つを見つけます。この行には、ノードの IP アドレス、カタログ ディレクトリの場所、およびデータ ディレクトリの場所が、この順にカンマ区切りで含まれています。 データ ディレクトリの場所を確認します。

    - DbCatalogDir= カタログ ディレクトリの場所

      デフォルト： /catalog

      注: カタログ ディレクトリがどれかはっきりわからない場合は、次の手順に従います： /opt/vertica/config/admintools.conf ファイルを開きます。 ［Nodes］セクションまでスクロールします。 v_*dbname*_node*XXXX* で始まる行の 1 つを見つけます。この行には、ノードの IP アドレス、カタログ ディレクトリの場所、およびデータ ディレクトリの場所が、この順にカンマ区切りで含まれています。 カタログ ディレクトリの場所を確認します。

    - DbHostNames= Data Repository のホスト名のカンマ区切りリスト

      デフォルト： yourhostname1,yourhostname2,yourhostname3

- DbName= データベース名

  デフォルト： drdata

  注: データベース名がはっきりわからない場合は、Vertica Linux データベース管理者として Admintools を実行します。 メイン メニューから［6 Configuration Menu］を選択肢、次に［3 View Database］を選択します。 使用するデータベースの名前は、［Select database to view］ダイアログにあります。 この値は、DbName に対して指定された値に一致する必要があります。データベース名を確認し、［キャンセル］を選択します。

- DbPwd= データベース パスワード

  デフォルト： dbpass

注: drinstall.properties ファイル内に InstallDestination パラメータがある場合、このパラメータは今後、使用されませんので、削除しても問題ありません。

11. Data Repository が稼働中であることを確認してから、以下のコマンドを入力して、インストール前のスクリプトを実行します。

    ```
    ./dr_validate.sh -p properties_file
    ```

    以下に例を示します。

    ```
    ./dr_validate.sh -p drinstall.properties
    ```

    インストール前スクリプトは、クラスタ内のすべてのホスト間で、パスワードなしの SSH を確立します。パスワードなしの SSH が存在しない場合、ユーザにパスワードが要求されます。

    注: インストール前スクリプトで再起動が必要になることがあります。

12. エラーまたは警告用の画面上で出力を確認します。システム設定の前提条件がすべて正しく設定されることを確認するため、このスクリプトを複数回実行できます。

13. インストール スクリプトを実行するには、以下のコマンドを入力します。

    ```
    ./dr_install.sh -p properties_file
    ```

    以下に例を示します。

    ```
    ./dr_install.sh -p drinstall.properties
    ```

インストール スクリプトは、データ リポジトリをアップグレードし、不要な vertica プロセスを無効にします。 vertica Linux データベース管理者ユーザ パスワードがユーザに要求される場合があります。

注: パスワードを入力し、Enter キーを 2 回押して続行します。

14. エラーがある場合は調査して解決してください。
15. 以下の手順に従って、Data Repository が正しくアップグレードされたことを確認します。
    a. 以下のコマンドを入力します。

       ```
       /opt/vertica/bin/adminTools
       ```

       ［Administration Tools］ダイアログ ボックスが表示されます。
    b. バナーの上部に、データベース バージョンが 7.0.1-2 と表示されていることを確認します。
16. Vertica Linux データベース管理者ユーザとして［Administration Tools］ダイアログ ボックスのメイン メニューから、オプション 3 （Start Database）を選択して、Data Repository を再起動します。

    Data Repository がアップグレードされます。

Data Aggregator 上のオープン ファイル数の上限を確認

Data Aggregator をインストールしているユーザのオープン ファイル数が 65536 以上の値であることを確認します。 この値を永続的に設定します。

次の手順に従ってください:

1. root ユーザまたは sudo ユーザとして、Data Aggregator をインストールするコンピュータにログインします。コマンド プロンプトを開いて次のコマンドを入力し、ulimit のオープン ファイルの上限を少なくとも 65536 に変更します。

   ```
   ulimit -n ulimit_number
   ```

   例：

   ```
   ulimit -n 65536
   ```

2. Data Aggregator をインストールするコンピュータ上で /etc/security/limits.conf ファイルを開き、以下の行を追加します。

```
# Added by Data Aggregator
* soft nofile 65536
# Added by Data Aggregator
* hard nofile 65536
```

注: これらの変更を有効にするには、Data Aggregator を再起動します。アップグレード中である場合、アップグレード処理により自動的に Data Aggregator が再起動されます。

3. Data Aggregator をインストールするコンピュータ上で、オープン ファイルの数が適切に設定されていることを確認するには、以下のコマンドを入力します。

```
ulimit -n
```

このコマンドは、以前に指定した ulimit 値を返します。Data Aggregator 上のオープン ファイル数の上限が設定されました。

すべてのデータベース テーブルのセグメント化の確認

データベース テーブルがすべてセグメント化されていることを確認します。テーブルをセグメント化することにより、データベースに必要なディスク容量を減らすことができます。 テーブルをセグメント化すると一般にクエリのパフォーマンスも向上します。

次の手順に従ってください:

1. Vertica Linux データベース管理者ユーザとして、Data Repository がインストールされているクラスタ内のコンピュータの 1 つにログインします。

2. segment.py スクリプトをダウンロードし、インストール メディアを抽出します。 Vertica Linux データベース管理者ユーザが書き込み可能なディレクトリ内にスクリプトを置きます。 この手順では、segment.py スクリプトが Vertica Linux データベース管理者ユーザのホーム ディレクトリにあると仮定します。

3. コマンド プロンプト ウィンドウを開き、以下のコマンドを入力します。

```
./segment.py --task tables --pass database_admin_user_password  [--name
database_name] [--port database_port]
```

database_admin_user_password

Vertica Linux データベース管理者ユーザ パスワードを設定します。

database_name

データベースの名前を示します。 データベース名がデフォルトの drdata でない場合のオプションです。

database_port

Vertica に接続するために使用するポートを示します。 ポート番号がデフォルトの 5433 でない場合のオプションです。

以下に例を示します。

```
./segment.py --task tables --pass password  --name mydatabase
```

現在セグメント化されていないすべてのテーブル予測値が、大きい順に並べられて返されます。

4. セグメント化されていないデータベース テーブル予測が返される場合は、テーブルをセグメント化 (P. 22) します。

データベース テーブルのセグメント化（クラスタ インストールのみ）

アップグレード中に、またはアップグレード後の任意の時点で、すべてのデータベース テーブルがセグメント化されていることを確認します (P. 21)（まだの場合）。 セグメント化されていないデータベース テーブル予測が返される場合は、それらをセグメント化します。 また、アップグレードの後にいつでもデータベース テーブルをセグメント化することができます。

重要: データベース テーブルをセグメント化しない場合、Data Aggregator コンポーネントのアップグレード中に警告メッセージが表示されます。

テーブルをセグメント化することにより、データベースに必要なディスク容量を減らすことができます。 テーブルをセグメント化すると一般にクエリのパフォーマンスも向上します。Data Aggregator および Data Collector が稼働している場合、またはこれらのコンポーネントがダウンしている場合のいずれもデータベース テーブルをセグメント化できます。

注: セグメント化はリソースを大量に消費するプロセスです。 Data Aggregator コンポーネントをアップグレードする前に、Data Aggregator および Data Collector がダウンしているときにデータベース テーブルをセグメント化することを強くお勧めします。 Data Aggregator および Data Collector の実行中にデータベース テーブルをセグメント化することはできますが、推奨していません。

Data Aggregator および Data Collector がダウンしている間にデータベース テーブルをセグメント化する場合、Data Aggregator コンポーネントをアップグレードする前に以下の情報を考慮してください。

- このスクリプトは、データベース内の大規模なテーブルに実行する場合は数時間かかる可能性があります。内部のセグメント化テストおよび顧客データベース テストでは、100 GB 以上のテーブルのセグメント化が完了するまでに 10 時間以上かかりました。 セグメント化の時間はテーブル サイズに対して一定ではありません。 行数、列数、データの圧縮、マシンの仕様などの多くの要因によって時間は変わります。 Data Aggregator および Data Collector がダウンしている場合、インフラストラクチャ環境のアクティブな監視は発生しません。

重要: Data Aggregator が実行されていない場合でも、セグメント化のディスク使用率の合計が使用可能なディスク容量の 90 パーセントを超えることはできません。 セグメント化の間にディスク使用率が 90 パーセントを超える場合、それ以上テーブルは処理されません。

Data Aggregator コンポーネントをアップグレードした後、Data Aggregator および Data Collector の実行中にデータベース テーブルをセグメント化する場合は、以下の情報を考慮してください。

- データベース テーブルをセグメント化している間は、Data Aggregator 管理機能を何も実行しないでください。たとえば以下のようなものです。
  - 監視プロファイルの変更
  - 監視プロファイルへのコレクションの関連付け
  - ポーリング レートの増加
  - 新しいディスカバリの実行

    注: ここに記述したものがすべてではありません。
- レポートの負荷は最小化することをお勧めします。

重要: データベース内のテーブルのセグメント化では、Data Aggregator が実行されている場合、使用可能なディスク容量の少なくとも 40 パーセントはクエリ処理および他のデータベース アクティビティ用に空いている必要があります。

セグメント化が完了した後、バックアップ用のディスク容量は、作成された新しいセグメント化テーブル予測でのデータ量の分だけ増加します。セグメント化が完了した後、バックアップが実行される前に、使用可能なディスク容量が十分にあることを確認してください。

セグメント化されていない古いテーブル予測に対するバックアップ領域内のデータは、restorePointLimit （このエントリはバックアップ設定ファイル内にあります）の時間プラス 1 日が経過した後に削除されます。

古いデータが削除されるのにかかる時間を回避するには、バックアップ設定ファイル内のスナップショット名を変更し、セグメント化が完了した後にフル バックアップを実行します。 次に、古いバックアップをアーカイブし、バックアップ ディスクからバックアップを削除できます。 セグメント化が完了した後に作成されたバックアップを使用できない場合にのみ、セグメント化前のバックアップを使用してください。 セグメント化前のバックアップを使用する必要がある場合、テーブル予測を再度セグメント化する必要があります。

データベース テーブルのセグメント化の準備

データベース テーブルのセグメント化を準備するには、以下の手順に従います。

- Data Repository をバックアップします。
- データのないデータベース テーブルをセグメント化します。
- 残りのデータベース テーブルをセグメント化するために必要な時間の量を予測します。

Data Repository をバックアップするには、以下の手順に従います。

1. Data Repository をバックアップします。 バックアップの実行は時間のかかるプロセスです。 以下のコマンドを実行します。

   ```
   backup_script_directory_location/backup_script.sh
   >/backup_directory_location/backup.log  2>&1
   ```

   以下に例を示します。

   ```
   /home/vertica/backup_script.sh >/tmp/backup.log  2>&1
   ```

   注: Data Repository を自動的にバックアップするためにこのスクリプトを作成していた場合の詳細については、「CA Performance Management 管理者ガイド」を参照してください。

データのないデータベース テーブルをセグメント化するには、以下の手順に従います。

1. Vertica Linux データベース管理者ユーザとして、Data Repository がインストールされているクラスタ内のコンピュータの 1 つにログインします。
2. segment.py スクリプトをダウンロードし、インストール メディアを抽出します。 Vertica Linux データベース管理者ユーザが書き込み可能なディレクトリ内にスクリプトを置きます。 この手順では、segment.py スクリプトが Vertica Linux データベース管理者ユーザのホーム ディレクトリにあると仮定します。
3. Data Aggregator の実行中に以下のコマンドを入力します。

   ```
   ./segment.py --task zerotables --pass database_admin_user_password [--name database_name] [--port database_port]
   ```

   database_admin_user_password

   Vertica Linux データベース管理者ユーザ パスワードを設定します。

   database_name

   データベースの名前を示します。 データベース名がデフォルトの drdata でない場合のオプションです。

   database_port

   Vertica に接続するために使用するポートを示します。 ポート番号がデフォルトの 5433 でない場合のオプションです。

   データのないデータベース テーブルがセグメント化されました。

残りのデータベース テーブルをセグメント化するために必要な時間の量を判断するには、ベースラインを計算します。

1. テーブル名が大きい順に並べられて返されるようにするには、以下のコマンドを入力します :

   ```
   ./segment.py --task tables --pass database_admin_user_password [--name database_name] [--port database_port]
   ```

2. セグメント化が完了するまで、スケジュールされたバックアップを無効にします。バックアップはセグメント化プロセスの邪魔になる可能性があります。

3. 約 5 GB のサイズがあるテーブルを手順 1 から選択します。以下のコマンドを入力して、テーブルをセグメント化します。

   ```
   ./segment.py --task segment --table rate_table_name --pass
   database_admin_user_password [--name database_name] [--port database_port]
   ```

   注: このコマンドは、Data Aggregator の実行中に実行できますが、2 ～ 3 時間の保守ウィンドウ中にこのコマンドを実行することをお勧めします。

4. スケジュールされたバックアップを再度有効にします。

5. 5 GB のテーブルをセグメント化するためにかかった時間を使用して、100 GB 未満のテーブルをすべてセグメント化するためにかかる時間を判断します。

   注: データベース テーブルをセグメント化するために必要となる実際の時間は、テーブル内のデータのタイプおよび圧縮に基づいて変わる可能性があります。 ここで計算されるのは概算の値です。 定期的な保守ウィンドウを計画する場合は、セグメント化される 10 ～ 15 GB のデータベース テーブルごとに余分な時間を追加してください。

   大きなデータベースについては、データベース全体をセグメント化するために十分な長さのある保守ウィンドウを 1 回ではスケジュールできない可能性があります。 この場合、複数の保守ウィンドウにわたってデータベース テーブルをセグメント化することができます。

データベース テーブルのセグメント化

次の手順に従ってください:

1. Vertica Linux データベース管理者ユーザとして、Data Repository がインストールされているクラスタ内のコンピュータの 1 つにログインします。

2. 前の手順でのテーブル予測セグメント化の検証中に、10 を超えるゼロレングス テーブル予測が検出された場合は、以下のコマンドを入力してそれらをセグメント化します。

   ```
   ./segment.py --task segment --pass database_admin_user_password --zerotables
   [--name database_name] [--port database_port]
   ```

   database_admin_user_password

   Vertica Linux データベース管理者ユーザ パスワードを設定します。

database_name

データベースの名前を示します。 データベース名がデフォルトの drdata でない場合のオプションです。

database_port

Vertica に接続するために使用するポートを示します。 ポート番号がデフォルトの 5433 でない場合のオプションです。

以下に例を示します。

```
./segment.py --task segment --pass password --zerotables --name mydatabase --port 1122
```

3. サイズが 100 GB を超えるテーブル予測がある場合は、以下のコマンドを入力し、最初に 100 GB 未満であるテーブル予測をセグメント化するスクリプトを作成します。

```
./segment.py --task script --pass database_admin_user_password --lt100G [--name database_name] [--port database_port]
```

database_admin_user_password

Vertica Linux データベース管理者ユーザ パスワードを設定します。

database_name

データベースの名前を示します。 データベース名がデフォルトの drdata でない場合のオプションです。

database_port

Vertica に接続するために使用するポートを示します。 ポート番号がデフォルトの 5433 でない場合のオプションです。

以下に例を示します。

```
./segment.py --task script --pass password --lt100G --name mydatabase --port 1122
```

4. セグメント化が完了するまで、スケジュールされたバックアップを無効にします。バックアップはセグメント化プロセスの邪魔になる可能性があります。

5. segment-script.sh スクリプトを実行するには、以下のコマンドを入力します。

```
nohup ./segment-script.sh
```

このスクリプトは、セグメント化されていない 100 GB 未満のテーブル予測をセグメント化し、小さい順に並べます。 出力は nohup.out に送信されます。 シェルが予期せず閉じられた場合、スクリプトの実効は継続されます。

実施されている保守ウィンドウのサイズ、および 100 GB 未満のテーブルのすべての合計サイズに応じて、保守ウィンドウでセグメント化されるテーブルを判断します。データベース テーブルのセグメント化を準備したときに計算された概算時間に基づいて、保守ウィンドウ内に収まらないテーブルを削除することによって、生成されたスクリプトを変更します。 生成された segment-script.sh を保守ウィンドウ内で実行します。100 GB 未満のテーブルのすべてを保守ウィンドウ内でセグメント化できなかった場合は、スクリプトを再生成して次の保守ウィンドウ内で segment-script.sh を実行し、テーブルがすべてセグメント化されるまで続けます。

重要: スクリプトを実行すると、ディスク使用率が 90 パーセントを超える原因となるテーブルにはエラー メッセージが表示され、それらのテーブルはセグメント化されません。 これらのテーブルをセグメント化するには、使用可能なディスク容量を増やす必要があります。

ディスク使用率が 60 パーセントを超える原因となる各テーブルごとにプロンプトが示されます。 これらのテーブルをセグメント化する前に、Data Aggregator をダウンさせることを強くお勧めします。

また、このスクリプトの実行には数時間かかる場合があることに注意してください。 いったん開始されたら、データベースの破損を回避するためにスクリプトの実行を中断しないでください。

6. さらにセグメント化が必要で、今後の保守ウィンドウで実行される場合にのみ、スケジュールされたバックアップを再度有効にします。

7. 100 GB を超える残りのテーブル予測をセグメント化するスクリプト（segment-script.sh）を生成するには、以下のコマンドを入力します。

```
./segment.py --task script --pass database_admin_user_password [--name
database_name] [--port database_port]
```

database_admin_user_password

Vertica Linux データベース管理者ユーザ パスワードを設定します。

database_name

データベースの名前を示します。 データベース名がデフォルトの drdata でない場合のオプションです。

database_port

Vertica に接続するために使用するポートを示します。 ポート番号がデフォルトの 5433 でない場合のオプションです。

以下に例を示します。

```
./segment.py --task script --pass password --name mydatabase --port 1122
```

重要: スクリプトが生成されると、ディスク使用率が 60 パーセントおよび 90 パーセントを超える原因となる可能性があるすべてのテーブルが示されます。

8. スケジュールされたバックアップを無効にします（まだの場合）。
9. segment-script.sh スクリプトを実行するには、以下のコマンドを入力します。

   ```
   nohup ./segment-script.sh
   ```

   このスクリプトは、未分割のテーブルをすべてセグメント化し、小さい順に並べます。

   重要: スクリプトを実行すると、ディスク使用率が 90 パーセントを超える原因となるテーブルにはエラー メッセージが表示され、それらのテーブルはセグメント化されません。 これらのテーブルがセグメント化されるようにするには、使用可能なディスク容量を増やす必要があります。

   ディスク使用率が 60 パーセントを超える原因となる各テーブルごとにプロンプトが示されます。 これらのテーブルをセグメント化する前に、Data Aggregator をダウンさせることを強くお勧めします。

   このスクリプトは、データベース内の大規模なテーブルに実行する場合は数時間かかる可能性があります。 内部のセグメント化テストおよび顧客データベース テストでは、100 GB 以上のテーブルのセグメント化が完了するまでに 10 時間以上かかりました。 セグメント化の時間はテーブル サイズに対して一定ではありません。 行数、列数、データの圧縮、マシンの仕様などの多くの要因によって時間は変わります。 実施されている保守ウィンドウのサイズに応じて、保守ウィンドウごとのテーブルのセグメント化を計画します。

10. すべてのテーブルがセグメント化されたことを確認するには、以下のコマンドを入力します。

    ```
    ./segment.py --task tables --pass database_admin_user_password  [--name database_name] [--port database_port]
    ```

    以下の 内容のメッセージが表示されます。

    セグメント化されていないテーブルはありません。

11. スケジュールされたバックアップを再度有効にします。
12. Data Aggregator および Data Collector がダウンしているときにデータベース テーブルをセグメント化した場合は、これらのコンポーネントを起動します。
    a. Data Aggregator を起動するには、以下のコマンドを入力します。

       ```
       service dadaemon start
       ```

    b. Data Collector を起動するには、以下のコマンドを入力します。

       ```
       service dcmd start
       ```

前述の手順では、segment.py スクリプトの使用について、および環境をマイグレートする際のさまざまな考慮事項について簡単に説明しています。スクリプトの使用に関してご不明な点がある場合、またはマイグレーションを計画する際にヘルプが必要な場合は、CA サポートまでお問い合わせください。

Data Aggregator インストールのアップグレード - サイレント モード

Data Aggregator の既存のインストール環境をアップグレードすると、カスタム プロファイルおよび以下の機能の設定を維持できます。

- ベンダー認定
- ベンダー認定優先度

  注: 新しいベンダー認定は、対応するメトリック ファミリの［ベンダー認定優先度］リストの最後に追加されます。 新しいベンダー認定を利用するには、ベンダー認定優先度を手動で変更します。 たとえば、F5 CPU ベンダー認定は通常の CPU としてモデル化されますが、F5 はホスト リソースもサポートするため、検出されません。アップグレード後に、ホスト リソース CPU 優先度エントリは、この優先度リストの最後に新しく追加された F5 エントリより上位になります。 F5 CPU のデバイスおよびコンポーネントを検出するには、CPU メトリック ファミリのベンダー認定優先度を更新します。 新規インストールの場合、この問題は存在しません。
- 監視プロファイル

- 以下のポーリング制御設定：
  - コンパイルされた MIB
  - インターフェースのフィルタ設定
  - 検出された監視対象デバイスおよびコンポーネント
  - 収集されたポーリング データ
  - SNMP プロファイル
  - ディスカバリ プロファイル

インストールされているソフトウェアのアップグレードは、既存のソフトウェアをアンインストールせずにインストールされます。 インストーラは、既存のインストール環境があるかどうかを検出し、続行するかどうかを確認します。

重要：Data Aggregator インストールをアップグレードする前に、Data Repository データベースをバックアップしてください。 また、Data Aggregator インストールをアップグレードする前に CA Performance Center をアップグレードします。

次の手順に従ってください:

1. 必要に応じて、レスポンス ファイルを変更します。
2. Data Aggregator をインストールするコンピュータに、root ユーザまたは sudo ユーザとしてログインします。
3. /tmp フォルダにインストール パッケージをコピーします。
4. 以下のコマンドを入力して、インストール ファイルの権限を変更します。

   ```
   chmod a+x installDA.bin
   ```

5. 以前に作成したレスポンス ファイルを使用してサイレント インストールを実行するには、以下のいずれかの手順を実行します。
   - root ユーザとしてインストールを実行するには、以下のコマンドを入力します。

     ```
     ./installDA.bin -i silent -f レスポンス ファイル名
     ```

     レスポンス ファイル名

     レスポンス ファイルのファイル名を示します。

     デフォルト： installer.properties

- sudo ユーザとしてインストールを実行するには、以下のコマンドを入力します。

  ```
  sudo ./installDA.bin -i silent -f レスポンス ファイル名
  ```

  レスポンス ファイル名

  レスポンス ファイルのファイル名を示します。

Data Aggregator がサイレント モードでインストールされます。サイレント モードでは、インストールの結果を示すメッセージは表示されません。

6. CA_Infrastructure_Management_Data_Aggregator_Install_*タイムスタンプ*.log ファイルの情報を確認することにより、インストールが成功したことを確認します。このログ ファイルは、Data Aggregator をインストールしたディレクトリ（たとえば、/opt/IMDataAggregator/Logs）にあります。
7. （新規インストール） Data Aggregator を CA Performance Center のデータ ソースとして登録します。

   注: データ ソースの登録の詳細については、「*CA Performance Center 管理者ガイド*」を参照してください。
8. Data Aggregator が CA Performance Center と自動的に同期されるまで数分かかります。

   あるいは、自動同期が行われるまで待機できない場合は、CA Performance Center と Data Aggregator を手動で同期できます。

   注: インストールが完了すると、インストーラにより Data Aggregator が自動的に再起動されます。

詳細:

CA Performance Management Data Aggregator のアップグレード方法 - サイレント モード (P. 11)

Data Collector インストールのアップグレード - サイレント モード

インストールされている Data Collector はアップグレードできます。 インストールされているソフトウェアのアップグレードは、既存のソフトウェアをアンインストールせずにインストールされます。

重要: Data Collector インストールをアップグレードする前に、Data Aggregator が稼働している必要があります。http://*ホスト名:ポート*/rest にアクセスしてください。 *ホスト名:ポート*には Data Aggregator のホスト名とポート番号を指定します。 このページが正常に表示される場合、Data Aggregator は稼働中です。

次の手順に従ってください:

1. 必要に応じて、レスポンス ファイルを変更します。
2. Data Collector をインストールするコンピュータに、root ユーザまたは sudo ユーザとしてログインします。
3. 以下のいずれかの操作を実行して、Data Collector のインストール パッケージにアクセスします。
   - Data Aggregator がインストールされているコンピュータに HTTP でアクセスできる場合は、Data Collector をインストールするコンピュータ上で Web ブラウザを開きます。以下のアドレスに移動し、インストール パッケージをダウンロードします。

     `http://data_aggregator:port/dcm/install.htm`

     data_aggregator:*port*

     Data Aggregator のホスト名および必要なポート番号を指定します。

     デフォルト： 8581 （Data Aggregator のインストール時に非デフォルト値を指定しなかった場合）。

     /tmp ディレクトリにインストール パッケージを保存します。

- Data Aggregator がインストールされているコンピュータに HTTP でアクセス できない 場合、HTTP でアクセス できるコンピュータ上でコマンド プロンプトを開きます。 以下のコマンドを入力して、インストール パッケージを Desktop ディレクトリにダウンロードします。

  ```
  wget -P /Desktop -nv
  http://data_aggregator:port/dcm/InstData/Linux/VM/install.bin
  ```

  data_aggregator:port

  Data Aggregator のホスト名および必要なポート番号を指定します。

  デフォルト： 8581 （Data Aggregator のインストール時に非デフォルト値を指定しなかった場合）。

  Data Collector をインストールするコンピュータの /tmp ディレクトリに install.bin ファイルを転送します。

注: Data Aggregator がインストールされているコンピュータに HTTP でアクセスできる場合に、Data Collector インストール パッケージを非対話モードでダウンロードするには、wget コマンドを使用します。

4. 以下のコマンドを入力して、/tmp ディレクトリに移動します。

   ```
   cd /tmp
   ```

5. 以下のコマンドを入力して、インストール ファイルの権限を変更します。

   ```
   chmod a+x install.bin
   ```

6. 以前に作成したレスポンス ファイルを使用してサイレント インストールを実行するには、以下のいずれかの手順を実行します。

   - root ユーザとしてインストールを実行するには、以下のコマンドを入力します。

     ```
     ./install.bin -i silent -f レスポンス ファイル名
     ```

     レスポンス ファイル名

     レスポンス ファイルのファイル名を示します。

     デフォルト： installer.properties

- sudo ユーザとしてインストールを実行するには、以下のコマンドを入力します。

  ```
  sudo ./install.bin -i silent -f レスポンス ファイル名
  ```

  レスポンス ファイル名

  レスポンス ファイルのファイル名を示します。

Data Collector がサイレント モードでインストールされます。サイレント モードでは、インストールの結果を示すメッセージは表示されません。

7. Data Collector をインストールしたコンピュータで、/opt/IMDataCollector/Logs/CA_Infrastructure_Management_Data_Collector_*タイムスタンプ*.log ファイルを確認します。 インストールに成功した場合、ログに 0 Warnings、0 NonFatalErrors、および 0 FatalErrors と表示されます。
8. 以下の手順に従い、インストール後に Data Collector の接続が成功していることを確認します。
   a. デフォルト テナント管理者として CA Performance Center にログインします。
   b. Data Aggregator の管理ビューに移動し、［システム ステータス］ビューを展開します。
   c. メニューから［Data Collector］を選択します。
   d. Data Collector がリストに表示されることを確認します。 この Data Collector をデフォルト テナントに関連付けるべきかどうか確認された場合に 'n' を選択していたら、そのテナントと IP ドメインは空白になります。

      注: リストがリフレッシュされ、新しくインストールした Data Collector が表示されるまで数分かかる場合があります。

9. テナントと IP ドメインが空白の場合は、各 Data Collector にテナントと IP ドメインを割り当てます。

   a. Data Collector インスタンスを選択して、［割り当て］をクリックします。

   b. ［Data Collector の割り当て］ダイアログ ボックスで、この Data Collector のテナントおよび IP ドメインを選択し、［保存］をクリックします。

   Data Collector がインストールされます。

   注: Data Collector がインストールされているコンピュータを再起動すると、Data Collector が自動的に再起動されて Data Aggregator に接続されます。

Data Collector が再インストールされると、Data Collector は Data Aggregator からデバイスとコンポーネントの情報を取得し、以前にポーリングされていたデバイスとコンポーネントのポーリングを再開します。

詳細:

CA Performance Management Data Aggregator のアップグレード方法 - サイレント モード (P. 11)

埋め込み CAMM を備える CA Performance Management 2.3.3 の、CAMM 2.4 を備える CA Performance Management 2.4 へのアップグレード

埋め込み CAMM （CA Mediation Manager）を備える CA Performance Management 2.3.3 を、CAMM 2.4 を備える CA Performance Management 2.4 にアップグレードする場合と、CAMM 2.2.6 を備える CA Performance Management 2.3.4 を、CAMM 2.4 を備える CA Performance Management 2.4 にアップグレードする場合とでは、プロセスが異なります。 これらの手順に従えば、CA Performance Management および CAMM の最新のバージョンへの円滑な移行を確実に行うことができます。

次の手順に従ってください:

1. マイグレーション スクリプトを実行する前に CAMM 2.4 MC / LC をインストールします。 CAMM のインストールの詳細については、次の URL にある CAMM Wiki を参照してください。

   https://wiki.ca.com/display/CAMM23/Install+CA+Mediation+Manager
   https://wiki.ca.com/display/CAMM23/Install+CA+Mediation+Manager

   重要: 必ず CAMM 2.4 の新規インストールを実行してください。

2. LC を DC にインストールした後、デバイス パックのマイグレーション スクリプトを実行します。

   デバイス パックのマイグレーションの詳細については、次の URL にある CAMM Wiki を参照してください。

   https://wiki.ca.com/display/CAMM23/Device+Pack+Migration
   https://wiki.ca.com/display/CAMM23/Device+Pack+Migration

   重要: デバイス パック マイグレーション スクリプトは、必ず -d オプションで実行してください。-d オプションを使用するとデバイス パックがマイグレートされますが、自動的には開始されません。

3. Data Aggregator を停止します。

4. CA Performance Management 2.4 に対して新しいディレクトリ構造に新しいタグでデバイス パックを展開するには、camm-tools-cert-migration-<version>-SNAPSHOT-jar-with-dependencies.jar スクリプトを実行して認定をマイグレートします。 このスクリプトは Data Aggregator ファイル（CA_DA_<version>_Linux.tar.gz）にパッケージ化されています。 スクリプトを実行するには、CA_DA_<version>_Linux.tar.gz を CA サポート サイトの CA Performance Management ページからダウンロードして解凍します。

5. CA Performance Management 2.4 にアップグレードします。 「CA Performance Management 2.4 インストール ガイド」を参照してください。

6. CAMM MC Web UI で、マイグレートしたデバイス パック エンジンを起動します。

CAMM 2.2.6 を備える CA Performance Management 2.3.4 の、CAMM 2.4 を備える CA Performance Management 2.4 へのアップグレード

CAMM（CA Mediation Manager）2.2.6 を備える CA Performance Management 2.3.4 を、CAMM 2.4 を備える CA Performance Management 2.4 にアップグレードする場合と、埋め込み CAMM を備える CA Performance Management 2.3.3 を、CAMM 2.4 を備える CA Performance Management 2.4 にアップグレードする場合では、プロセスが異なります。 これらの手順に従えば、CA Performance Management および CAMM の最新のバージョンへの円滑な移行を確実に行うことができます。

1. Data Aggregator を停止します。
2. CA Performance Management 2.4 に対して新しいディレクトリ構造に新しいタグでデバイス パックを展開するには、camm-tools-cert-migration-<version>-SNAPSHOT-jar-with-dependencies.jar スクリプトを実行して認定をマイグレートします。 このスクリプトは Data Aggregator ファイル（CA_DA_<version>_Linux.tar.gz）にパッケージ化されています。 スクリプトを実行するには、CA_DA_<version>_Linux.tar.gz を CA サポート サイトの CA Performance Management ページからダウンロードして解凍します。
3. CA Performance Management 2.4 にアップグレードします。 「CA Performance Management 2.4 インストール ガイド」を参照してください。
4. CAMM 2.4 にアップグレードします。CAMM 2.4 へのアップグレード方法については、以下の URL の CAMM Wiki を参照してください。

   https://wiki.ca.com/display/CAMM23/Upgrade+CA+Mediation+Manager
   https://wiki.ca.com/display/CAMM23/Upgrade+CA+Mediation+Manager

Data Aggregator プロセスの自動復旧の再有効化

Data Aggregator プロセスの自動復旧を再度有効化します。 Data Aggregator のアップグレード前に、自動復旧が無効化されました。 有効化すると、データベース サーバがメモリを使い果たすか、または Data Repository が一時的に利用不可になると、Data Aggregator はデータの整合性を保つために自動的にシャットダウンします。

次の手順に従ってください:

1. Data Aggregator がインストールされているコンピュータに root ユーザとしてログインします。
2. コンソールを開き、以下のコマンドを入力します。

   ```
   crontab -e
   ```

   vi セッションが開始されます。
3. 以下の行の先頭にあるシャープ記号（#）を削除して、コメントを解除します。

   ```
   # * * * * * /etc/init.d/dadaemon start > /dev/null
   ```

   たとえば、以下のようになります。

   ```
   * * * * * /etc/init.d/dadaemon start > /dev/null
   ```

   Data Aggregator プロセスの自動復旧が再度有効化されました。

アップグレード後の手順の実行

Data Aggregator のアップグレード後に、以下の手順を実行します。

- セキュリティ ポリシーの安全性を強化するために Java 6 に Java Cryptography Extension（JCE）を適用していた場合は、Data Aggregator での Java 7 の使用を検討します。 このセキュリティ強化が必要な場合は、アップグレード後に最新の JCE を再適用してください。

  注: JCE の Java 7 バージョンについては、Oracle のサイトを参照してください。

- Data Aggregator のアップグレードでは、/opt/IMDataAggregator/apache-karaf-X.X.X ディレクトリを /opt/IMDataAggregator/backup/apache-karaf ディレクトリにバックアップします。/opt/IMDataAggregator/apache-karaf-X.X.X/etc/ ディレクトリに含まれているカスタマイズ（デフォルトのログ レベルやその他の設定）は、バックアップされますが、インストール ディレクトリに自動的にリストアされません。これらのカスタマイズが失われないようにするには、アップグレードが正常に行われた後にカスタマイズを手動でリストアします。

  たとえば、local-jms-broker.xml を更新したカスタムの設定が /opt/IMDataAggregator/apache-karaf-X.X.X/deploy ディレクトリに存在するとします。アップグレードの後、/opt/IMDataAggregator/apache-karaf-X.X.X/deploy ディレクトリ内の local-jms-broker.xml は、最新のインストーラから取得されています。カスタマイズされた jms-broker ファイルは、/opt/IMDataAggregator/backup/apache-karaf ディレクトリにバックアップされています。カスタム変更を保持するには、バックアップ ファイルを特定し、インストール ディレクトリにそのカスタマイズを再適用します。

- Data Collector のアップグレードでは、/opt/IMDataCollector/apache-karaf-X.X.X ディレクトリを /opt/IMDataCollector/backup/apache-karaf ディレクトリにバックアップします。/opt/IMDataCollector/apache-karaf-X.X.X/etc/ ディレクトリに含まれているカスタマイズ（デフォルトのログ レベルやその他の設定）は、バックアップされますが、インストール ディレクトリに自動的にリストアされません。これらのカスタマイズが失われないようにするには、アップグレードが正常に行われた後にカスタマイズを手動でリストアします。

- ベンダー認定の優先度を設定します。 Data Aggregator のアップグレード後に、新しいベンダー認定が、対応するメトリック ファミリの［ベンダー認定優先度］リストの最後に追加されます。 新しいベンダー認定を利用するには、ベンダー認定優先度を手動で変更します。 たとえば、F5 CPU ベンダー認定は通常の CPU としてモデル化されますが、F5 はホスト リソースもサポートするため、検出されません。 アップグレード後に、ホスト リソース CPU 優先度エントリは、この優先度リストの最後に新しく追加された F5 エントリより上位になります。F5 CPU のデバイスおよびコンポーネントを検出するには、CPU メトリック ファミリのベンダー認定優先度を更新します。

  注: ベンダー認定の優先度設定の詳細については、「Data Aggregator 自己認定ガイド」を参照してください。

- CA Performance Center 上のメモリ設定を再適用します。 大規模展開では、デフォルトの最大メモリ使用率設定をカスタマイズすることを推奨します。 これらのカスタマイズされた設定は、アップグレード時に自動的に再適用されません。 カスタム メモリ設定を利用するには、アップグレード後にそれらを手動で再適用します。

  以下の手順を実行します。

  1. <インストール ディレクトリ>/PerformanceCenter/*SERVICE*/conf/wrapper.conf.old を開きます。

     注: *SERVICE* は、以下のサービス用のサブディレクトリを示します。

     - PC （Performance Center コンソール サービス）
     - DM （Performance Center デバイス マネージャ サービス）
     - EM （Performance Center イベント マネージャ サービス）

     例： /opt/CA/PerformanceCenter/PC/conf/wrapper.conf.old

  2. "wrapper.java.maxmemory" プロパティを見つけて、指定されている値をメモします。

  3. <インストール ディレクトリ>/PerformanceCenter/*SERVICE*/conf/wrapper.conf を開きます。

     例： /opt/CA/PerformanceCenter/PC/conf/wrapper.conf

  4. "wrapper.java.maxmemory" プロパティを見つけて、手順 2 でメモした値に設定します。 保存します。

以下のコマンドを入力して、各デーモンを停止し、再起動します。

```
service service name stop
service service name start
```

5. 各サービスについて、手順 1 - 5 を繰り返します。

カスタム メモリ設定が再適用されます。

第 3 章: トラブルシューティング

このセクションには、以下のトピックが含まれています。

トラブルシューティング: Data Aggregator の同期の失敗

症状：

Data Aggregator を CA Performance Center と同期させようとしたら、「同期失敗」というメッセージが表示されました。［データ ソースの管理］ダイアログ ボックスの Data Aggregator の［ステータス］列に、［同期失敗］と表示されます。

解決方法:

同期失敗は、同期中に Data Aggregator が送信されたデータを処理できないことを示す可能性があります。 DMService.log と呼ばれるデバイス マネージャ アプリケーション ログ ファイルを確認します。 このファイルは、CA/PerformanceCenter/DM/logs ディレクトリにあります。 同期中に Data Aggregator が CA Performance Center から受信したデータを処理できなかった場合、ログ エントリには一般的な SOAP 例外が表示されます。

同期の以下の段階で例外およびスタック トレースを探します。

- プル
- グローバル同期
- バインド（データ ソースと最初に同期する場合のみ実行する）
- プッシュ

CA テクニカル サポートに問い合わせて、この情報を伝えてください。

トラブルシューティング : CA Performance Center が Data Aggregator に接続できない

症状:

Data Aggregator は正常にインストールされましたが、［データ ソースの管理］ダイアログ ボックスのステータスに「'接続できません。' ＜perf> は Data Aggregator に接続できません」と表示されます。

解決方法:

以下の手順を実行します。

1. Data Aggregator ホストのコンピュータにログオンします。 コンソールを開き、以下のコマンドを入力して、Data Aggregator が実行されていることを確認します。

   ```
   service dadaemon status
   ```

2. Data Aggregator が実行されている場合、CA Performance Center が Data Aggregator に接続するのを妨げているのは、ほとんどの場合、ネットワークの問題です。 ネットワークの問題をすべて解決します。

3. Data Aggregator が実行されていない場合は、Data Aggregator を開始します。 root ユーザ、または制限されたコマンド セットにアクセスできる sudo ユーザとして、Data Aggregator ホストのコンピュータにログインします。 コンソールを開き、以下のコマンドを入力します。

   ```
   service dadaemon start
   ```

トラブルシューティング : Data Collector をインストールしたが、[Data Collector リスト]メニューに表示されない

症状:

Data Collector は正常にインストールしましたが、［Data Collector リスト］メニューに Data Collector が表示されません。

解決方法:

以下の手順を実行します。

1. *Data Collector のインストール ディレクトリ* /apache-karaf-2.3.0/shutdown.log ファイルを確認して、Data Collector が自動的にシャットダウンされていないことを確認します。 Data Collector のインストール時に、Data Aggregator ホスト、テナント、または IP ドメインを誤って指定した場合、Data Collector は自動的にシャットダウンされます。 shutdown.log ファイルに、Data Collector がシャットダウンされた理由についてのエラー情報があります。 Data Collector がシャットダウンされる主な理由には、以下の 2 つがあります。
   - Data Collector のインストール時に指定された Data Aggregator ホスト情報、テナント、または IP ドメインが正しくなかった。
     - Data Aggregator のホスト情報を誤って指定した場合は、Data Collector をアンインストールして再インストールします。
     - テナントを誤って指定した場合は、Data Collector をアンインストールして再インストールします。
     - IP ドメインを誤って指定した場合は、Data Collector をアンインストールして再インストールします。
   - Data Aggregator との接続が確立できなかった。
2. 以下のコマンドを入力して、Data Aggregator への接続が確立されていることを確認します。

```
netstat –a | grep 61616
```

3. Data Aggregator への接続が存在しない場合は、以下の手順に従います。

   a. Data Collector ホストの *Data Collector のインストール ディレクトリ*/apache-karaf-2.3.0/deploy/local-jms-broker.xml ファイルを表示します。 このファイルには、Data Collector のインストール時に指定した Data Aggregator ホストのホスト名または IP アドレスが含まれています。

   b. broxer.xml ファイルの「networkConnector」セクションを探します。このセクションには、以下のような行が含まれている必要があります。

   ```
   <networkConnector name="manager"
          uri="static:(tcp://test:61616)"
          duplex="true"
         suppressDuplicateTopicSubscriptions="false"/>
   ```

   「networkConnector」セクションに指定されている Data Aggregator ホスト名が正しく、DNS または /etc/hosts のエントリを介して解決されることを確認します。 Data Collector のインストール時に Data Aggregator のホスト名を誤って入力した場合、Data Collector は Data Aggregator と通信できません。

   c. 以下のコマンドを入力して、ポート 61616 で Data Aggregator ホストへの telnet 接続が正常に開くことを確認します。

   ```
   telnet dahostname 61616
   ```

   このコマンドにより、Data Aggregator がそのポートをリスンしていることが確認されます。

d. telnet 接続が正常に開かない場合は、以下の理由が考えられます。

- Data Aggregator が実行されていません。Data Aggregator が実行されていることを確認します。コンソールを開き、以下のコマンドを入力します。

  ```
  service dadaemon status
  ```

  Data Aggregator が実行されていない場合は、Data Aggregator を開始します。 root ユーザ、または制限されたコマンドセットにアクセスできる sudo ユーザとして、Data Aggregator ホストのコンピュータにログインします。コンソールを開き、以下のコマンドを入力します。

  ```
  service dadaemon start
  ```

- 接続を開始するリクエストが、Data Collector から Data Aggregator に対して正常に実行されていません。 broxer.xml ファイルの「networkConnector」セクションに指定されているポートが、Data Aggregator で受信接続に対して開かれていることを確認します。この接続を妨げるファイアウォールのルールがないことを確認します。