



Silk Central 20.0

レポート データマート

Micro Focus
The Lawn
22-30 Old Bath Road
Newbury, Berkshire RG14 1QN
UK
<http://www.microfocus.com>

© Copyright 2004-2019 Micro Focus またはその関連会社。

MICRO FOCUS、Micro Focus のロゴおよび Silk Central は、Micro Focus またはその関連会社の商標または登録商標です。

その他、記載の各名称は、各所有社の知的所有財産です。

2019-04-30

目次

| | |
|--------------------------------|-----------|
| 概要 | 4 |
| アーキテクチャ | 5 |
| データマートを使用してレポートを作成する方法 | 6 |
| データマート クエリを作成する | 6 |
| 実行計画におけるテストの信頼性 | 6 |
| 実行フォルダにおけるすべての失敗したテスト | 7 |
| テスト サイクルのステータス | 8 |
| 実行ツリーのステータス | 10 |
| 構成スイートのステータス | 11 |
| トラブルシューティング | 13 |
| 間違っただまたは欠落したデータ | 13 |
| データマートがシステム速度を低下させる | 13 |
| 参考：データマート テーブルとビュー | 14 |
| DM_TestStatus | 14 |
| RV_TestStatusExtended | 15 |
| RV_LatestTestStatus | 16 |
| RV_LatestFinishedTestStatus | 18 |
| RV_MaxTestRunID | 19 |
| RV_MaxFinishedTestRunID | 20 |
| RV_TestingCycleStatus | 20 |
| RV_ExecutionPlanStatusPerBuild | 21 |
| RV_EPFinishedStatusPerBuild | 22 |
| RV_ExecutionPlanStatusRollup | 23 |
| RV_EPFinishedStatusRollup | 24 |
| RV_ConfigurationSuiteStatus | 24 |
| RV_ConfigSuiteFinishedStatus | 25 |

概要

Silk Central レポート データマートは、レポート目的のデータへのアクセスを容易にします。運用テーブルから詳細レポートの作成に使用される専用のビューにデータを移動します。利点は次の通りです。

- テーブルとビューの名前が明確で、検索するデータの場所を素早く見つけることができます。
- 前処理済みのデータによって、自分で計算せずに集計済みデータにアクセスすることができます。
- レポートがより単純化され、SQL クエリが高速化されるため、パフォーマンスが改善されます。
- 運用データベースの負荷に対する依存性が低下し、パフォーマンスが改善され、負荷スパイクが削除されます。


データマートの現在のバージョンは結果領域をカバーします。レポートの他の領域については、将来のリリースでデータマートに追加される予定です。以下のテーブルとビューが現在利用可能です。

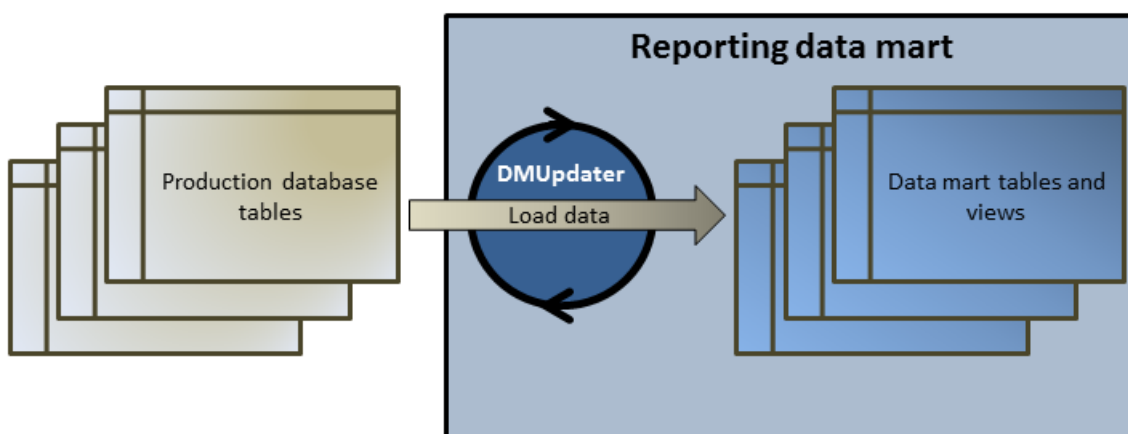
- DM_TestStatus テーブルは、ステータス関連ビューの基礎です。
- RV_TestStatusExtended ビューは、あるテスト実行に対する詳細な情報を提供します。
- RV_LatestTestStatus ビューは、特定の実行計画とビルドのコンテキスト内でテストの最後のテスト実行のステータスと拡張情報を提供します。
- RV_LatestFinishedTestStatus ビューは、特定の実行計画とビルドのコンテキスト内でテストの最後のテスト実行のステータスと拡張情報を提供します。このビューは、RV_LatestTestStatus ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。
- RV_MaxTestRunID ビューは、すべてのテスト、実行計画、およびビルドの組み合わせに対する最後のテスト実行 ID を取得するために役立つビューです。
- RV_MaxFinishedTestRunID ビューは、すべてのテスト、実行計画、およびビルドの組み合わせに対する最後のテスト実行 ID を取得するために役立つビューです。このビューは、RV_MaxTestRunID ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。
- RV_TestingCycleStatus ビューは、テスト サイクルのステータス情報を提供します。
- RV_ExecutionPlanStatusPerBuild ビューは、ビルドのコンテキストにおいて、すべての実行計画に対する最後のテスト ステータスの合計数を取得します。
- RV_EPFinishedStatusPerBuild ビューは、ビルドのコンテキストにおいて、すべての実行計画に対する最後のテスト ステータスの合計数を取得します。このビューは、RV_ExecutionPlanStatusPerBuild ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。
- RV_ExecutionPlanStatusRollup ビューは、ビルドのコンテキストにおいて、実行計画またはフォルダごとの合計、失敗、未実行テストの合計数を取得します。
- RV_EPFinishedStatusRollup ビューは、ビルドのコンテキストにおいて、実行計画またはフォルダごとの合計、失敗、未実行テストの合計数を取得します。このビューは、RV_ExecutionPlanStatusRollup ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。
- RV_ConfigurationSuiteStatus ビューは、すべての構成スイートとビルドごとの構成に対するステータスの合計数を一覧します。
- RV_ConfigSuiteFinishedStatus ビューは、すべての構成スイートとビルドごとの構成に対するステータスの合計数を一覧します。このビューは、RV_ConfigurationSuiteStatus ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。

Silk Central のデータベース スキーマについての詳細な情報を含んだ .zip ファイルをダウンロードできます。メニューから、**ヘルプ > ドキュメント** を選択し、**Silk Central データベース スキーマ** をクリックして、.zip ファイルをダウンロードしてください。

アーキテクチャ

運用データベースのテーブルからバックグラウンドで定期的にデータを抽出し、データマートテーブルとビューに読み込むことで、簡単で高速なクエリを実現します。データベースの読み込みが高価でない場合は、通常は変更がコミットされた後数分以内にこのデータは利用可能になります。Silk Central システム管理者としてログインしている場合、<http://<サーバー>:<ポート>/sctm/check/db> を実行して、**DM_TestStatus Table** を確認することで、データ読み込み処理の現在の状態を確認することができます。



 **注:** データマートを含まない Silk Central のバージョン (バージョン 13.0 より前) からアップデートする場合、データマートテーブルとビューは運用システムからのデータで最初に更新されます。データベースのサイズに応じて、この処理にかかる時間は変わります。この処理が完了すると、データにアクセスできるようになります。



データマートを使用してレポートを作成する方法

以下の例を用いて、データマート ビューを使用して便利なレポートを作成する方法について説明します。

データマート クエリを作成する

1. メニューで **レポート > 詳細ビュー** をクリックします。
2. **レポート ツリー** で、新しいレポートを追加するフォルダを選択します。
これにより、ディレクトリ ツリー内でレポートが保存される場所が決まります。
3. ツールバーの  をクリックします。**レポートの新規作成** ダイアログ ボックスが開きます。
4. 新しいレポートの名前を入力します。
この名前は、**レポート ツリー** に表示されます。
5. 他のユーザーもこのレポートを使用できるようにする場合は、**このレポートを他のユーザーと共有する** チェック ボックスをオンにします。
6. **説明** フィールドにレポートの説明を入力します。
7. **詳細クエリ モード** をクリックすると、**レポートデータ クエリ** フィールドが開きます。既に記述したコードをこのフィールドに挿入するか、新しいコードを直接書き込みます。
プレースホルダの挿入 リストは、定義済みの関数プレースホルダを使用して SQL クエリを編集するときに便利です。詳細については、「カスタム レポート用の SQL 関数」を参照してください。
 **注:** クエリの SQL コードを手作業で編集する場合は、**SQL のチェック** をクリックして作業を確認します。
8. **完了** をクリックして、設定を保存します。

実行計画におけるテストの信頼性

問題

デイリー ビルドをテストするため、およびテスト対象アプリケーションの品質を保証するために、継続的な統合環境において、テストは理想的には 1 日に少なくとも 1 度は実行されるべきです。テスト セットが AUT の品質を計測するためにどの程度信頼できるかを理解するために、時間の経過とともに結果がどのように変化しているかを観察することが時に必要になります。たとえば、頻繁にステータスが変更されるテストがテスト セットに含まれている場合、品質を測る本来の尺度にはなりません。

解決策

指定した実行計画のコンテキストにおいて指定したテストに対する結果を一覧するレポートを作成するために、**RV_TestStatusExtended** データマート ビューを使用します。これを利用して、時間の経過と共にテストの結果がどのように変化したかを観察することができます。便宜上、タグが設定されたビルドに関連したものに結果のリストを限定することによって、テスト対象アプリケーションの特定のマイルストーン ビルドを観察します。このレポートは、実行計画とビルドのコンテキストにおけるテストに対するテスト結果データを収集します。クエリは次のようになります。

- このビューから表示したい列を選択します。
- 調査したいテストの ID とそのテストが属する実行計画の ID によって結果を絞り込みます。

- タグが設定されたビルドのみを調査するために定数を追加します。

```
SELECT TestName, ExecutionPlanName, VersionName, BuildName, TestRunID,
    PassedCount, FailedCount, NotExecutedCount
FROM RV_TestStatusExtended
WHERE TestID = ${TESTID|1|Test ID} AND ExecutionPlanID = ${EXECUTIONPLANID|1|
Execution Plan ID} AND BuildIsTagged = 1
ORDER BY BuildOrderNumber
```

SQL クエリの結果は、選択した実行計画内の選択したテストに対するすべてのテストの実行になります。以下の例では、ビルド 579_Drop2 に対して再実行されたテストを確認できます：

| TestName | Execution PlanName | Version Name | BuildName | TestRunID | Passed Count | FailedCount | NotExecutedCount |
|----------|----------------------|--------------|-------------|-----------|--------------|-------------|------------------|
| UI Tests | EN SQL2012 IE9 IIS | 3.0 | 579_Drop0 2 | 7741797 | 59 | 5 | 0 |
| UI Tests | EN SQL2012 IE9 IIS | 3.0 | 579_Drop0 2 | 7745078 | 63 | 1 | 0 |
| UI Tests | EN SQL2012 IE9 IIS | 3.0 | 593_Drop0 3 | 7787437 | 63 | 1 | 0 |
| UI Tests | EN SQL2012 IE9 IIS | 3.0 | 605_Drop0 4 | 7848720 | 63 | 1 | 0 |

実行フォルダにおけるすべての失敗したテスト

問題

通常、すべての実行計画はフォルダ階層に構造化されています。各フォルダは実行計画やそのテストが関連するさまざまな分野や目的によって分類されます。複数の実行計画が継続的な統合環境で定期的に行われたり、ときにはリリース期間に実行され、洗練された実行統計情報を得ることができますが、残念ながら単一の実行計画毎の情報になります。

しかし、特定の分野や目的に応じて実行したすべてのテストがどのような点で脆弱かを知るために全体的な情報が必要になる場合があります。

解決策

データマート ビュー [RV_LatestTestStatus](#) を使用して、特定の実行計画の階層レベルに対してすべての失敗したテストのリストを返すレポートを作成します。

以下のクエリは、実行計画フォルダ内のテストを選択し、実行計画名とビルド名などのコンテキスト情報を取得します。

```
SELECT TestID, TestName, ExecutionPlanID, ExecutionPlanName, BuildName
FROM RV_LatestTestStatus Its
INNER JOIN TM_ExecTreePaths ON Its.ExecutionPlanID = TM_ExecTreePaths.NodeID_pk_fk
WHERE TM_ExecTreePaths.ParentNodeID_pk_fk = ${executionFolderID|2179|Execution
Folder ID}
AND StatusID = 2
ORDER BY TestName
```

クエリは次の動作を行います。

- ビュー `RV_LatestTestStatus` を使用して最新のテスト実行の結果を取得します。
- 階層内のすべての実行計画からすべてのテストをクエリできるようにするために、実行ツリー階層 (`TM_ExecTreePaths`) を含めます。
- トップレベルのフォルダ ID を使って、一致する `ParentNodeID_pk_fk` のノードから分析を開始します。
- 失敗したテストだけ (`StatusID = 2`) を含めます。

StatusID をテーブル `TM_TestDefStatusNames` から検索します。

SQL クエリの結果は、選択した実行フォルダ内の最後の実行に失敗したすべてのテストになります。

| TestID | TestName | ExecutionPlanID | ExecutionPlanName | BuildName |
|--------|------------------|-----------------|-------------------|-----------|
| 14073 | JUnitTestPackage | 2184 | CI Testing | 352 |
| 14107 | Volatile Tests | 2191 | Volatile Test | 352 |

テスト サイクルのステータス

問題

テスト サイクルは手動テスト担当者、テスト、製品のさまざまなビルドやバージョン、そして場合によっては構成についての情報さえ含まれるため、複雑なオブジェクトになり得ます。テスト サイクルの状況を見失わないようにするため、次のような質問に回答することは重要です。

- いくつかのテストが完了したか。
- そのうちいくつかのテストがビルドに対して合格あるいは失敗したか。
- 手動テスト担当者は忙しいか、または追加の作業を行うことが可能か。

解決策

データマート ビュー `RV_TestingCycleStatus` を使用して、テスト サイクルのステータスをテスト担当者 とビルドごとに表示するレポートを作成します。このレポートは、いくつかのテストが合格、失敗、未実行かを手動テスト担当者、構成、ビルドによってグループ化された概要を提供します。

```
SELECT BuildName, TesterLogin, TesterExecutionName,
       PassedCount, FailedCount, NotExecutedCount
FROM RV_TestingCycleStatus
WHERE TestingCycleID = ${testingCycleID|3|Testing Cycle ID}
ORDER BY BuildOrderNumber, TesterLogin
```

クエリは次の動作を行います。

- `RV_TestingCycleStatus` ビューをデータ ソースとして使用します。このビューには、`BuildName`、`TesterLogin`、および `TesterExecutionName` (テスト担当者、構成、およびテストを反映した自動生成された名前) が含まれています。
- データを表示したいテスト サイクル ID で制限します。

SQL クエリの結果は、ビルドとテスト担当者ごとのステータスを表示します。

| BuildName | TesterLogin | TesterExecutionName | PassedCount | FailedCount | NotExecutedCount |
|-----------|-------------|----------------------------------|-------------|-------------|------------------|
| 352 | | No specific tester (Test Assets) | 0 | 0 | 1 |

| BuildName | TesterLogin | TesterExecutionName | PassedCount | FailedCount | NotExecutedCount |
|-----------|--------------|---|-------------|-------------|------------------|
| 351 | admin | admin (English SQL2008 FF Tomcat - Test Assets) | 0 | 1 | 0 |
| 352 | admin | admin (English SQL2008 FF Tomcat - Test Assets) | 0 | 0 | 1 |
| 352 | gmazzuchelli | gmazzuchelli (English Oracle10g IE8 Tomcat - Test Assets) | 0 | 1 | 1 |
| 352 | jallen | jallen (German Oracle11g FF Tomcat - Test Assets) | 1 | 1 | 0 |
| 352 | smiller | smiller (German SQL2008 IE8 IIS - Test Assets) | 1 | 1 | 0 |

未割り当てテストに対して、「テスト担当者なし」グループが列 (TesterLogin、TesterFirstName、TesterLastName) に対して空の値で作成されます。

テストサイクルが手動テスト担当者のパフォーマンスによってどのくらい行われているかを表示したい場合、次の多少異なるクエリが役立ちます。

```
SELECT TesterLogin, TesterExecutionName, SUM(PassedCount) PassedCount,
SUM(FailedCount) FailedCount, SUM(NotExecutedCount) NotExecutedCount
FROM RV_TestingCycleStatus
WHERE TestingCycleID = ${testingCycleID|3|Testing Cycle ID}
GROUP BY TesterLogin, TesterExecutionName
ORDER BY TesterLogin
```

クエリは次の点が拡張されています。

- GROUP BY TesterLogin, TesterExecutionName 残りの列を表示するため。
- SUM() 数値をカウンタに合計するため。

| TesterLogin | TesterExecutionName | PassedCount | FailedCount | NotExecutedCount |
|--------------|---|-------------|-------------|------------------|
| | No specific tester (Test Assets) | 0 | 0 | 1 |
| admin | admin (English SQL2008 FF Tomcat - Test Assets) | 0 | 1 | 1 |
| gmazzuchelli | gmazzuchelli (English Oracle10g | 0 | 1 | 1 |

| TesterLogin | TesterExecution Name | PassedCount | FailedCount | NotExecutedCount |
|-------------|--|-------------|-------------|------------------|
| jallen | IE8 Tomcat - Test Assets) jallen (German Oracle11g FF Tomcat - Test Assets) | 1 | 1 | 0 |
| smiller | smiller (German SQL2008 IE8 IIS - Test Assets) | 1 | 1 | 0 |

実行ツリーのステータス

問題

一般的に、実行計画はさまざまなテスト領域や目的を表す階層構造になります。テストステータス、さらに領域や目的の品質を知るために、合格、失敗、未実行の全体の数を知りたい場合があります。

解決策

指定した実行計画フォルダに対してビルドで分類した合格、失敗、未実行の数を返すレポートを作成するために、[RV_ExecutionPlanStatusRollup](#) データマートビューを使用します。

```
SELECT BuildName, PassedCount, FailedCount, NotExecutedCount
FROM RV_ExecutionPlanStatusRollup
WHERE ExecutionFolderID = ${executionPlanID|43|Execution Plan ID}
```

クエリは次の動作を行います。

- BuildName を選択し、ステータスを RV_ExecutionPlanStatusRollup ビューから計算します。
- ステータスを取得したいトップレベルのフォルダを指定します (ExecutionFolderID)。

SQL クエリの結果は、ビルドごとに集約され、選択したフォルダのすべての実行計画のテスト実行のステータスを表示します。

| BuildName | PassedCount | FailedCount | NotExecutedCount |
|-----------|-------------|-------------|------------------|
| 351 | 0 | 0 | 2 |
| 352 | 15 | 7 | 1 |

選択した階層内の各実行計画に対するステータスの数のように、より詳細について知りたい場合は、[RV_ExecutionPlanStatusPerBuild](#) データマートビューを使用できます。

```
SELECT eps.BuildName, eps.ExecutionPlanID, SUM(eps.PassedCount) PassedCount,
SUM(eps.FailedCount) FailedCount, SUM(eps.NotExecutedCount) NotExecutedCount
FROM RV_ExecutionPlanStatusPerBuild eps
INNER JOIN TM_ExecTreePaths etp ON eps.ExecutionPlanID = etp.NodeID_pk_fk
WHERE etp.ParentNodeID_pk_fk = ${execFolderID|44|Execution Folder ID}
GROUP BY eps.ExecutionPlanID, eps.BuildOrderNumber, eps.BuildName
ORDER BY eps.BuildOrderNumber, eps.ExecutionPlanID
```

クエリは次の動作を行います。

- RV_ExecutionPlanStatusPerBuild ビューを使用して、実行計画固有のデータにアクセスします (ExecutionPlanID および ExecutionPlanName)。実行計画だけでなく、フォルダ ノードに対する結果

も取得する必要があるため、前で使用した RV_ExecutionPlanStatusRollup ビューは、あらかじめ集計済みのデータ (合計したデータ) を含んでおり、ここでの目的には適していません。

- 階層情報を取り入れるため、JOIN テーブルの TM_ExecTreePath を使用して指定したフォルダ内のすべてのノードを選択します。
- ExecutionFolderID を使ってトップレベルのフォルダを指定します。テーブル TM_ExecutionTreePaths もすべての実行計画に対する自己参照を含んでいるため、ParentNodeID_pk_fk に対する実行計画 ID を使ってこのクエリを実行することもできます。
- ORDER BY BuildOrderNumber と ExecutionPlanID を追加して、古いビルドとその実行計画を最初に表示するように結果を分かりやすく並べ替えます。

SQL クエリの結果は、選択したフォルダのすべての実行計画のテスト実行のステータスを表示します。

| BuildName | ExecutionPlanID | PassedCount | FailedCount | NotExecutedCount |
|-----------|-----------------|-------------|-------------|------------------|
| 351 | 2307 | 0 | 0 | 2 |
| 352 | 2184 | 11 | 2 | 0 |
| 352 | 2185 | 0 | 3 | 0 |
| 352 | 2186 | 2 | 1 | 0 |
| 352 | 2187 | 1 | 0 | 0 |
| 352 | 2191 | 0 | 1 | 0 |
| 352 | 2307 | 1 | 0 | 1 |

構成スイートのステータス

問題

構成スイートを使用すると、複数のブラウザやオペレーティングシステムといった複数の構成に対して同じテストセットを実行することができます。テスト対象アプリケーションの品質と信頼性に関して適切な状態に保つことができるようにするために、個々それぞれの構成に対する結果を把握することを考えます。

解決策

データマートビュー [RV_ConfigurationSuiteStatus](#) を使用して、ビルドごとの各構成に対して合格、失敗、未実行の数を返すレポートを作成します。

```
SELECT BuildName, ConfigurationName, PassedCount, FailedCount, NotExecutedCount
FROM RV_ConfigurationSuiteStatus
WHERE ConfigurationSuiteID = ${configSuiteID|97|Configuration Suite ID}
ORDER BY BuildOrderNumber, ConfigurationName
```

クエリは次の動作を行います。

- RV_ConfigurationSuiteStatus ビューからテスト事項のビルドごとのステータスの数を取得します。
- 構成スイート (ConfigurationSuiteID) で結果を絞り込みます。

SQL クエリの結果は、それぞれの構成に対するテスト実行のステータスを表示します。

| BuildName | ConfigurationName | PassedCount | FailedCount | NotExecutedCount |
|-----------|-------------------|-------------|-------------|------------------|
| 350 | Chrome | 0 | 1 | 0 |
| 350 | Firefox | 0 | 1 | 0 |

| BuildName | ConfigurationName | PassedCount | FailedCount | NotExecutedCount |
|-----------|-------------------|-------------|-------------|------------------|
| 350 | Internet Explorer | 0 | 1 | 0 |
| 351 | Chrome | 1 | 0 | 0 |
| 351 | Firefox | 1 | 0 | 0 |
| 351 | Internet Explorer | 0 | 1 | 0 |
| 352 | Chrome | 1 | 0 | 0 |
| 352 | Firefox | 1 | 0 | 0 |
| 352 | Internet Explorer | 1 | 0 | 0 |

この例では、すべての構成を取得するために構成スイートの ID を使用します。特定のビルドを指定して結果を絞り込むことも可能です。この場合、where 節に BuildID、BuildName、または BuildOrderNumber を指定する必要があります。



注: *RV_ConfigurationSuiteStatus* ビューは、テスト固有のデータは持たず、ステータスの数の合計のみを含みます。さらにテスト固有のデータを取得するには、*RV_LatestTestStatus* ビューを使用することができます。

トラブルシューティング

間違っただまたは欠落したデータ


問題

データマート テーブルやビューからデータをクエリする場合、一覧された結果が最新でないかたり欠落していることがあります。

解決策

データマート テーブルとビューは、バックグラウンドで定期的に更新されますが、リアルタイムではありません。このため、データがデータマート テーブルにロードされるまでに数秒から数分かかります。システムが高負荷状態で実行されている場合、こもデータをロードするバックグラウンド処理のパフォーマンスに影響を及ぼします。他のプロセスがより高い優先度を持っていたり、DataMartUpdater バックグラウンド ジョブが一時的にブロックされてりする可能性があるためです。しばらく時間をおいてから再びクエリを実行して、更新されたデータを取得してください。

Silk Central システム管理者としてログインしている場合、`http://<サーバー>:<ポート>/sctm/check/db` を実行して、**DM_TestStatus Table** を確認することで、データ読み込み処理の現在の状態を確認することができます。

 **注:** テスト担当者と依存するテスト実行はテストが削除されるとデータマートから削除されます。これは、テスト パッケージをクリーンアップするために削除されたテストにも適用されます。

データマートがシステム速度を低下させる

問題

データマートを実行して以来、システム全体のパフォーマンスが悪くなった、または動作が不安定になったように感じられます。

解決策

このような状態が発生すべきではありませんが、実際にデータマートが影響を与えているのかどうかを確認するために、データマートを無効化することができます。

1. **インスタンス管理** ページで、変更するインスタンスのアプリケーション サーバーを停止します。
2. `SccAppServerBootConf.xml` ファイルをテキスト エディタで開きます。このファイルのデフォルトパスは、アプリケーション サーバー上の `C:%Program Files (x86)%Silk%Silk Central 20.0%instance_<インスタンス番号>_<インスタンス名>%Conf%AppServer` になります。
3. `Config/DataMart/Enabled` XML タグに移動し、その値を `false` に設定します。
4. XML ファイルを保存して閉じます。
5. アプリケーション サーバーを再起動します。

参考：データマート テーブルとビュー

以下のデータマート テーブルとビューは、レポート作成時に容易に素早く利用できます。

DM_TestStatus

DM_TestStatus テーブルは、ステータス関連ビューの基礎です。

このテーブルからは、テストの名前などの情報へ直接アクセスすることができないため、通常は他のデータマート ビューで、詳細データへの簡易アクセスを提供します。このテーブルのキーは TestID、ExecutionPlanID、BuildID、および TestRunID 列の組み合わせです。


| 行 | 説明 |
|----------------------------|---|
| TestID | テストの識別子。 |
| ExecutionPlanID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |
| TestRunID | テストの実行の識別子。 |
| ExecutionRunID | この結果を生成した実行計画の実行の識別子。 |
| StatusID | このテストの実行のステータス (TM_TestDefStatusNames を参照)。 |
| EDRStatusID | 実行計画の実行のステータス。詳細については、 TM_TestDefStatusNames を参照してください。た とえば、ステータス 7 = 手動実行の保留、ステータス 10 = 自動実行の保留です。 |
| ReasonID | このテストの実行のステータスの理由 (TM_ResultStatusReasons を参照)。null になる場 合があります。 |
| PassedCount | すべての合格したテストの合計数。通常のテストでは 0 か 1 ですが、パッケージ テストのルートではそれ以上にな る場合があります。 |
| FailedCount | すべての失敗したテストの合計数。通常のテストでは 0 か 1 ですが、パッケージ テストのルートではそれ以上にな る場合があります。 |
| NotExecutedCount | すべての未実行テストの合計数。通常のテストでは 0 か 1 ですが、パッケージ テストのルートではそれ以上にな る場合があります。 |
| ProjectID | この行が属するプロジェクトの ID。 |
| TestStartTime | テストの実行が開始された時間 (UTC)。 |
| ExecutionStartTime | テストの実行計画の実行が開始された時間 (UTC)。 |
| TestDurationInMilliseconds | テストの実行の継続時間 (ミリ秒)。 |
| IsBlocked | テストの実行がブロックされているか、ブロック解除され ているかのフラグ。 |

| 行 | 説明 |
|-------------|--------------------------------|
| DbChangedAt | この行がレポート データマートによって最後に更新された日時。 |

RV_TestStatusExtended

RV_TestStatusExtended ビューは、あるテスト実行に対する詳細な情報を提供します。

[RV_LatestTestStatus](#) ビューは実行計画と特定のビルドのコンテキスト内でのテストの最後のテスト実行のみを含みますが、このビューはすべてのテスト実行を含みます。[タグが設定されたビルドのすべてのテスト実行をリストするレポート](#)を作成するためにこのビューを使用できます。このテーブルのキーは TestID、ExecutionPlanID、BuildID、および TestRunID 列の組み合わせです。

 **注:** テスト担当者と依存するテスト実行はテストが削除されるとデータマートから削除されます。これは、テスト パッケージをクリーンアップするために削除されたテストにも適用されます。

| 行 | 説明 |
|----------------------------|---|
| TestID | テストの識別子。 |
| ExecutionPlanID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |
| TestRunID | テストの実行の識別子。 |
| ExecutionRunID | この結果を生成した実行計画の実行の識別子。 |
| StatusID | このテストの実行のステータス (TM_TestDefStatusNames を参照)。 |
| ReasonID | このテストの実行のステータスの理由 (TM_ResultStatusReasons を参照)。null になる場合があります。 |
| PassedCount | すべての合格したテストの合計数。通常のテストでは 0 か 1 ですが、パッケージ テストのルートではそれ以上になる場合があります。 |
| FailedCount | すべての失敗したテストの合計数。通常のテストでは 0 か 1 ですが、パッケージ テストのルートではそれ以上になる場合があります。 |
| NotExecutedCount | すべての未実行テストの合計数。通常のテストでは 0 か 1 ですが、パッケージ テストのルートではそれ以上になる場合があります。 |
| ProjectID | この行が属するプロジェクトの ID。 |
| TestStartTime | テストの実行が開始された時間 (UTC)。 |
| ExecutionStartTime | テストの実行計画の実行が開始された時間 (UTC)。 |
| TestDurationInMilliseconds | テストの実行の継続時間 (ミリ秒)。 |
| IsBlocked | テストの実行がブロックされているか、ブロック解除されているかのフラグ。 |
| DbChangedAt | この行がレポート データマートによって最後に更新された日時。 |

| 行 | 説明 |
|--------------------------|-----------------------------------|
| TestName | このテストの名前。 |
| TestDescription | テストの説明。 |
| TestParentID | テストの親の ID。 |
| PlannedTimeInMinutes | このテストの予定時間 (分)。 |
| Reason | 理由の名前。その間に削除された理由も含まれます。 |
| ExecutionPlanName | 実行計画の名前。 |
| ExecutionPlanDescription | 実行計画の説明。 |
| ExecutionParentFolderID | 実行計画の親の ID。 |
| Priority | 実行計画の優先度。0 = 低、1 = 中、2 = 高。 |
| BuildName | このテスト実行に使用したビルドの名前。 |
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |

RV_LatestTestStatus

RV_LatestTestStatus ビューは、特定の実行計画とビルドのコンテキスト内でテストの最後のテスト実行のステータスと拡張情報を提供します。

[RV_TestStatusExtended](#) ビューを使用して、すべてのテスト実行についての情報を取得します。[実行フォルダにおけるすべての失敗したテストをリストするレポート](#)を作成する方法を説明します。このテーブルのキーは TestID、ExecutionPlanID、BuildID、および TestRunID 列の組み合わせです。

| 行 | 説明 |
|-----------------|-----------------------|
| TestID | テストの識別子。 |
| ExecutionPlanID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |
| TestRunID | テストの実行の識別子。 |
| ExecutionRunID | この結果を生成した実行計画の実行の識別子。 |

| 行 | 説明 |
|----------------------------|--|
| StatusID | このテストの実行のステータス (TM_TestDefStatusNames を参照)。 |
| ReasonID | このテストの実行のステータスの理由 (TM_ResultStatusReasons を参照)。null になる場合があります。 |
| PassedCount | すべての合格したテストの合計数。通常のテストでは 0 か 1 ですが、パッケージテストのルートではそれ以上になる場合があります。 |
| FailedCount | すべての失敗したテストの合計数。通常のテストでは 0 か 1 ですが、パッケージテストのルートではそれ以上になる場合があります。 |
| NotExecutedCount | すべての未実行テストの合計数。通常のテストでは 0 か 1 ですが、パッケージテストのルートではそれ以上になる場合があります。 |
| ProjectID | この行が属するプロジェクトの ID。 |
| TestStartTime | テストの実行が開始された時間 (UTC)。 |
| ExecutionStartTime | テストの実行計画の実行が開始された時間 (UTC)。 |
| TestDurationInMilliseconds | テストの実行の継続時間 (ミリ秒)。 |
| IsBlocked | テストの実行がブロックされているか、ブロック解除されているかのフラグ。 |
| DbChangedAt | この行がレポート データマートによって最後に更新された日時。 |
| TestName | このテストの名前。 |
| TestDescription | テストの説明。 |
| TestParentID | テストの親の ID。 |
| PlannedTimeInMinutes | このテストの予定時間 (分)。 |
| Reason | 理由の名前。その間に削除された理由も含まれます。 |
| ExecutionPlanName | 実行計画の名前。 |
| ExecutionPlanDescription | 実行計画の説明。 |
| ExecutionParentFolderID | 実行計画の親の ID。 |
| Priority | 実行計画の優先度。0 = 低、1 = 中、2 = 高。 |
| BuildName | このテスト実行に使用したビルドの名前。 |
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |

| 行 | 説明 |
|--------------------|----------------|
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |

RV_LatestFinishedTestStatus

RV_LatestFinishedTestStatus ビューは、特定の実行計画とビルドのコンテキスト内でテストの最後のテスト実行のステータスと拡張情報を提供します。このビューは、RV_LatestTestStatus ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。

現在実行中の実行計画を無視するレポートを作成するために、このビューを使用できます。

[RV_TestStatusExtended](#) ビューを使用して、すべてのテスト実行についての情報を取得します。このテーブルのキーは TestID、ExecutionPlanID、BuildID、および TestRunID 列の組み合わせです。

| 行 | 説明 |
|--------------------|---|
| TestID | テストの識別子。 |
| ExecutionPlanID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |
| TestRunID | テストの実行の識別子。 |
| ExecutionRunID | この結果を生成した実行計画の実行の識別子。 |
| StatusID | このテストの実行のステータス。詳細については、Silk Central データベーススキーマの TM_TestDefStatusNames テーブルを参照してください。 |
| EDRStatusID | 実行計画の実行のステータス。たとえば、ステータス 7 = 手動実行の保留、ステータス 10 = 自動実行の保留です。詳細については、Silk Central データベーススキーマの TM_TestDefStatusNames テーブルを参照してください。 |
| ReasonID | このテストの実行のステータスの理由。null になる場合があります。詳細については、Silk Central データベーススキーマの TM_ResultStatusReasons テーブルを参照してください。 |
| PassedCount | すべての合格したテストの合計数。通常のテストでは 0 か 1 ですが、パッケージテストのルートではそれ以上になる場合があります。 |
| FailedCount | すべての失敗したテストの合計数。通常のテストでは 0 か 1 ですが、パッケージテストのルートではそれ以上になる場合があります。 |
| NotExecutedCount | すべての未実行テストの合計数。通常のテストでは 0 か 1 ですが、パッケージテストのルートではそれ以上になる場合があります。 |
| ProjectID | この行が属するプロジェクトの ID。 |
| TestStartTime | テストの実行が開始された時間 (UTC)。 |
| ExecutionStartTime | テストの実行計画の実行が開始された時間 (UTC)。 |

| 行 | 説明 |
|----------------------------|-------------------------------------|
| TestDurationInMilliseconds | テストの実行の継続時間 (ミリ秒)。 |
| IsBlocked | テストの実行がブロックされているか、ブロック解除されているかのフラグ。 |
| DbChangedAt | この行がレポート データマートによって最後に更新された日時。 |
| TestName | このテストの名前。 |
| TestDescription | テストの説明。 |
| TestParentID | テストの親の ID。 |
| PlannedTimeInMinutes | このテストの予定時間 (分)。 |
| Reason | 理由の名前。その間に削除された理由も含まれます。 |
| ExecutionPlanName | 実行計画の名前。 |
| ExecutionPlanDescription | 実行計画の説明。 |
| ExecutionParentFolderID | 実行計画の親の ID。 |
| Priority | 実行計画の優先度。0 = 低、1 = 中、2 = 高。 |
| BuildName | このテスト実行に使用したビルドの名前。 |
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |

RV_MaxTestRunID

RV_MaxTestRunID ビューは、すべてのテスト、実行計画、およびビルドの組み合わせに対する最後のテスト実行 ID を取得するために役立つビューです。

このテーブルのキーは TestID、ExecutionPlanID、および BuildID 列の組み合わせです。

| 行 | 説明 |
|-----------------|-----------|
| TestID | テストの識別子。 |
| ExecutionPlanID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |

| 行 | 説明 |
|--------------|-------------------------------------|
| MaxTestRunID | 実行計画とビルドのコンテキスト内にある、最後のテスト実行を識別します。 |

RV_MaxFinishedTestRunID

RV_MaxFinishedTestRunID ビューは、すべてのテスト、実行計画、およびビルドの組み合わせに対する最後のテスト実行 ID を取得するために役立つビューです。このビューは、RV_MaxTestRunID ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。

現在実行中の実行計画を無視するレポートを作成するために、このビューを使用できます。

このテーブルのキーは TestID、ExecutionPlanID、および BuildID 列の組み合わせです。

| 行 | 説明 |
|-----------------|-------------------------------------|
| TestID | テストの識別子。 |
| ExecutionPlanID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |
| MaxTestRunID | 実行計画とビルドのコンテキスト内にある、最後のテスト実行を識別します。 |

RV_TestingCycleStatus

RV_TestingCycleStatus ビューは、テストサイクルのステータス情報を提供します。

[テストサイクルの現在のステータスを表示するレポート](#)を作成するためにこのビューを使用できます。

TestingCycleID はテストサイクルを表し、TesterExecutionID (同様に TesterExecutionName、UserID、CapacityInCycle、TesterLogin、TesterFirstName、TesterLastName) は、テストサイクルに割り当てられたテスト担当者を識別するために使用されます。特定のテスト担当者が割り当てられていないテストの場合、UserID、CapacityInCycle、TesterLogin、TesterFirstName、および TesterLastName は null になります。このテーブルのキーは TesterExecutionID、および BuildID 列の組み合わせです。

| 行 | 説明 |
|--------------------------|--|
| TestingCycleID | テストサイクルの識別子。 |
| TesterExecutionID | 特定のテスト担当者に割り当てられているテストのグループを識別します。 |
| TesterExecutionName | 特定のテスト担当者に割り当てられているテストのグループに対して生成した名前。 |
| UserID | テスト担当者のユーザー ID。 |
| CapacityInCycleInMinutes | このテストサイクルでのこのユーザーの最大利用可能時間 (秒) |
| TesterLogin | テスト担当者のログイン名。 |
| TesterFirstName | テスト担当者の名前。 |
| TesterLastName | テスト担当者の姓。 |
| PassedCount | すべての合格したテストの合計数。 |

| 行 | 説明 |
|--------------------|-----------------------------------|
| FailedCount | すべての失敗したテストの合計数。 |
| NotExecutedCount | すべての未実行のテストの合計数。 |
| ProjectID | プロジェクトの識別子。 |
| BuildID | ビルドの識別子。 |
| BuildName | このテスト実行に使用したビルドの名前。 |
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |

RV_ExecutionPlanStatusPerBuild

RV_ExecutionPlanStatusPerBuild ビューは、ビルドのコンテキストにおいて、すべての実行計画に対する最後のテスト ステータスの合計数を取得します。

フォルダと子ノードは対象としません。[フォルダの各実行計画に対して、テスト実行のステータスを表示するレポート](#)を作成するためにこのビューを使用できます。[RV_ExecutionPlanStatusRollup](#)とは異なり、データの取得において階層を考慮しないため、このビューのパフォーマンス上の利点はほとんどありません。このテーブルのキーは ExecutionPlanID、および BuildID 列の組み合わせです。

| 行 | 説明 |
|-------------------------|---------------------|
| ExecutionPlanID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |
| ExecutionPlanName | 実行計画の名前。 |
| ExecutionParentFolderID | 実行計画の親の ID。 |
| PassedCount | すべての合格したテストの合計数。 |
| FailedCount | すべての失敗したテストの合計数。 |
| NotExecutedCount | すべての未実行のテストの合計数。 |
| ProjectID | 実行計画が属するプロジェクトの ID。 |
| BuildName | このテスト実行に使用したビルドの名前。 |

| 行 | 説明 |
|--------------------|-----------------------------------|
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |

RV_EPFinishedStatusPerBuild

RV_EPFinishedStatusPerBuild ビューは、ビルドのコンテキストにおいて、すべての実行計画に対する最後のテストステータスの合計数を取得します。このビューは、RV_ExecutionPlanStatusPerBuild ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。

現在実行中の実行計画を無視するレポートを作成するために、このビューを使用できます。フォルダと子ノードは対象としません。[フォルダの各実行計画に対して、テスト実行のステータスを表示するレポート](#)を作成するためにこのビューを使用できます。[RV_ExecutionPlanStatusRollup](#)とは異なり、データの取得において階層を考慮しないため、このビューのパフォーマンス上の利点はほとんどありません。このテーブルのキーは ExecutionPlanID、および BuildID 列の組み合わせです。

| 行 | 説明 |
|-------------------------|-----------------------------------|
| ExecutionPlanID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |
| ExecutionPlanName | 実行計画の名前。 |
| ExecutionParentFolderID | 実行計画の親の ID。 |
| PassedCount | すべての合格したテストの合計数。 |
| FailedCount | すべての失敗したテストの合計数。 |
| NotExecutedCount | すべての未実行のテストの合計数。 |
| ProjectID | 実行計画が属するプロジェクトの ID。 |
| BuildName | このテスト実行に使用したビルドの名前。 |
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |

| 行 | 説明 |
|--------------------|-------------------|
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |

RV_ExecutionPlanStatusRollup

RV_ExecutionPlanStatusRollup ビューは、ビルドのコンテキストにおいて、実行計画またはフォルダごとの合計、失敗、未実行テストの合計数を取得します。

フォルダの場合、カウンタにはすべての子の数値が含まれます。[フォルダのすべてのテスト実行のステータスを表示するレポート](#)を作成するためにこのビューを使用できます。このテーブルのキーは ExecutionFolderID、および BuildID 列の組み合わせです。

| 行 | 説明 |
|--------------------|-----------------------------------|
| ExecutionFolderID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |
| PassedCount | すべての合格したテストの合計数。 |
| FailedCount | すべての失敗したテストの合計数。 |
| NotExecutedCount | すべての未実行のテストの合計数。 |
| ProjectID | 実行計画が属するプロジェクトの ID。 |
| BuildName | このテスト実行に使用したビルドの名前。 |
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |

RV_EPFinishedStatusRollup

RV_EPFinishedStatusRollup ビューは、ビルドのコンテキストにおいて、実行計画またはフォルダごとの合計、失敗、未実行テストの合計数を取得します。このビューは、RV_ExecutionPlanStatusRollup ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。

現在実行中の実行計画を無視するレポートを作成するために、このビューを使用できます。

フォルダの場合、カウンタにはすべての子の数値が含まれます。このテーブルのキーは ExecutionFolderID、および BuildID 列の組み合わせです。

| 行 | 説明 |
|--------------------|-----------------------------------|
| ExecutionFolderID | 実行計画の識別子。 |
| BuildID | ビルドの識別子。 |
| PassedCount | すべての合格したテストの合計数。 |
| FailedCount | すべての失敗したテストの合計数。 |
| NotExecutedCount | すべての未実行のテストの合計数。 |
| ProjectID | 実行計画が属するプロジェクトの ID。 |
| BuildName | このテスト実行に使用したビルドの名前。 |
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |

RV_ConfigurationSuiteStatus

RV_ConfigurationSuiteStatus ビューは、すべての構成スイートとビルドごとの構成に対するステータスの合計数を一覧します。

構成スイートの各構成に対して、すべてのテスト実行のステータスを表示するレポートを作成するためにこのビューを使用できます。このテーブルのキーは ConfigurationID、および BuildID 列の組み合わせです。

| 行 | 説明 |
|----------------------|-------------|
| ConfigurationSuiteID | 構成スイートの識別子。 |

| 行 | 説明 |
|--------------------|-----------------------------------|
| ConfigurationID | 構成の識別子。 |
| ConfigurationName | 構成の名前。 |
| BuildID | ビルドの識別子。 |
| PassedCount | すべての合格したテストの合計数。 |
| FailedCount | すべての失敗したテストの合計数。 |
| NotExecutedCount | すべての未実行のテストの合計数。 |
| ProjectID | この行が属するプロジェクトの ID。 |
| BuildName | このテスト実行に使用したビルドの名前。 |
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |

RV_ConfigSuiteFinishedStatus

RV_ConfigSuiteFinishedStatus ビューは、すべての構成スイートとビルドごとの構成に対するステータスの合計数を一覧します。このビューは、RV_ConfigurationSuiteStatus ビューと同じ情報を提供していますが、完了した実行計画の実行内のテスト実行のみが対象となります。

現在実行中の実行計画を無視するレポートを作成するために、このビューを使用できます。

このテーブルのキーは ConfigurationID、および BuildID 列の組み合わせです。

| 行 | 説明 |
|----------------------|------------------|
| ConfigurationSuiteID | 構成スイートの識別子。 |
| ConfigurationID | 構成の識別子。 |
| ConfigurationName | 構成の名前。 |
| BuildID | ビルドの識別子。 |
| PassedCount | すべての合格したテストの合計数。 |
| FailedCount | すべての失敗したテストの合計数。 |
| NotExecutedCount | すべての未実行のテストの合計数。 |

| 行 | 説明 |
|--------------------|-----------------------------------|
| ProjectID | この行が属するプロジェクトの ID。 |
| BuildName | このテスト実行に使用したビルドの名前。 |
| BuildDescription | ビルドの説明。 |
| BuildOrderNumber | ビルドの順位番号。 |
| BuildIsTagged | ビルドにタグが設定されている場合は 1 を、そうでない場合は、0。 |
| VersionID | ビルドが属するバージョンの ID。 |
| VersionName | バージョンの名前。 |
| VersionDescription | バージョンの説明。 |
| VersionOrderNumber | バージョンの順位番号。 |
| ProductID | ビルドが属する製品の ID。 |
| ProductCode | 製品の名前。 |
| ProductDescription | 製品の説明。 |
| ProductOrderNumber | 製品の順位番号。 |