

Hyperion(R) Enterprise(R) Reporting

リリース 6.5.1

テクニカルリファレンスガイド

ORACLE®
ENTERPRISE PERFORMANCE
MANAGEMENT SYSTEM

Hyperion Enterprise Reporting テクニカルリファレンスガイド, 6.5.1

Copyright © 1991, 2009, Oracle and/or its affiliates. All rights reserved.

著者: Enterprise Reporting Information Development Team

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクル社までご連絡ください。

このソフトウェアまたは関連ドキュメントが、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供される場合は、次の通知が適用されます。

U.S. GOVERNMENT RIGHTS:

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

このソフトウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアは、危険が伴うアプリケーション(人的傷害を発生させる可能性があるアプリケーションを含む)への用途を目的として開発されていません。このソフトウェアを危険が伴うアプリケーションで使用する際、このソフトウェアを安全に使用するために、適切な安全装置、バックアップ、冗長性、その他の対策を講じることは使用者の責任となります。このソフトウェアを危険が伴うアプリケーションで使用したことに起因して損害が発生しても、オラクル社およびその関連会社は一切の責任を負いかねます。

Oracle は、Oracle Corporation またはその関連会社、あるいはその両方の登録商標です。他の名称は、それぞれの所有者の商標である可能性があります。

このソフトウェアおよびドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても、一切の責任を負いかねます。

目次

第 1 章 Hyperion Enterprise Reporting Web のダイナミックリンクライブラリについて	7
関数呼び出しの書式	7
関数のリスト	8
SPIDER.DLL - 例	9
Essbase の構文	10
AGENTFORM 関数 - エージェント購読の表示／更新	10
AGENTFORM 関数の構文	10
AGENTLIST 関数 - エージェントの一覧	11
AGENTLIST 関数の構文	11
AGENTLIST 関数の例	12
AGENTSSSENT 関数 - エージェント通知の表示	12
AGENTSSSENT 関数の構文	12
AGENTSSSENT 関数の例	12
CHANGEID 関数 - Hyperion Enterprise アプリケーションのログオン情報の変更	13
CHANGEID 関数の構文	13
CHANGEID 関数の例	14
CHART 関数 - 動的チャート	14
CHART 関数の構文	15
CHART 関数の例	19
CRITERIA 関数 - 条件スクリプトの表示	20
CRITERIA 関数の構文	20
CRITERIA 関数の例	20
DATAENTRY 関数 - レポートの編集	21
DATAENTRY 関数の構文	21
DATAENTRY 関数の例	23
EDITPOV 関数 - データの視点の変更	24
EDITPOV 関数の構文	25
EDITPOV 関数の例	25
EXCELREP 関数 - スプレッドシートの実行	26
EXCELREP 関数の構文	26
EXCELREP 関数の例	28

ログアウト	28
ログアウト関数の例	28
REPCHART 関数 - レポートとチャートの並列表示	28
REPCHART 関数の構文	29
REPCHART 関数の例	30
REPORT 関数 - 動的レポート	31
REPORT 関数の構文	31
REPORT 関数の例	34
PRINTREP 関数 - PDF ファイルへのレポート出力	35
PRINTREP 関数の構文	35
PRINTREP 関数の例	36
SCHEDREP 関数 - Hyperion Distributed Schedules の起動	37
SCHEDREP 関数の構文	37
SCHEDREP 関数の例	38
ライブラリ関数	39
SYSLIST (製品情報)	39
APPLIST (アプリケーション一覧)	41
SETLIST (レポートセotto一覧)	43
REPSETLIST (レポート一覧)	45
第2章 オブジェクト	49
レポートオブジェクト - 概要	49
上級 OLE ユーザ用の情報	50
GetAppName メソッド	50
GetCellDouble メソッド	51
GetCellPovString メソッド	52
GetCellString メソッド	53
GetCellStringHTML メソッド	53
GetCellStatus メソッド	54
GetName メソッド	55
GetNumRows メソッド	56
GetNumColumns メソッド	56
GetSystemName メソッド	56
GetTitle メソッド	57
エージェントオブジェクト	58
GetNumReports メソッド	58
GetReportDisp メソッド	59
エージェント評価オブジェクト - 概要	63
AddLink メソッド	64

ClearLinks メソッド	65
GetAgent メソッド	65
Notify メソッド	66
URLEncode メソッド	67
HTMLEncode メソッド	67
テンプレートオブジェクト	67
Sub メソッド	67
条件オブジェクト	68
上級 OLE ユーザ用の情報	68
GetTemplate メソッド	69
ServerUserName メソッド	69
SetErrorNum メソッド	69
SetErrorMsg メソッド	70
URLEncode メソッド	71
HTMLEncode メソッド	71
URLParam メソッド	72
URLParamNum メソッド	72
アクション情報オブジェクト	73
上級 OLE ユーザ用の情報	73
DataText プロパティ	74
MessageText プロパティ	74
UserMessageText プロパティ	75
OptionalText プロパティ	75
ErrorText プロパティ	76
UserName プロパティ	76
AgentName プロパティ	77
AgentDescription プロパティ	77
GetNumLinks メソッド	78
GetLinkText メソッド	78
GetLinkURL メソッド	79
第 3 章 データベース表のリファレンス	81
データベース表	81
表間の関係	82
エージェント表	83
Agent_Definition	83
Action_Definition	84
DataObjs	86
Agent_Schedule	87

Scheduled_ObjPOVS	87
Agent_Definition_Security	88
Available_Actions	89
Subscriptions	89
Action_Subscription	90
Agent_Notifications	91
Report_Links	92
Retry_Actions	92
Retry_Links	93
第 4 章 トラブルシューティング	95
SpCheck 構成チェッカーの実行	95
ログファイルの使用	96
SMARTPOV のエラー処理	97
例	97
Hyperion Enterprise Reporting Web の Web ウィザードのエラーメッセージ	98
DCOMCNFG.EXE プログラムの使用	101
DCOMCNFG.EXE でデフォルトのアクセス許可を設定する方法	101
DCOMCNFG.EXE で Microsoft Excel のアクセス許可を設定する方法	102
Windows NT 4.0 でエージェント用に HYPPROXY.CDAPROXY を設定する方 法	103
第 5 章 Spider.ini ファイル	105
Spider.ini ファイル内のオプション	105
構文	109
Spider.ini ファイルの例	109
第 6 章 追加情報	111
索引	113

1

Hyperion Enterprise Reporting Webのダイナミック リンクライブラリについて

この章の内容

関数呼び出しの書式	7
AGENTFORM 関数 - エージェント購読の表示／更新	10
AGENTLIST 関数 - エージェントの一覧	11
AGENTSSENT 関数 - エージェント通知の表示	12
CHANGEID 関数 - Hyperion Enterprise アプリケーションのログオン情報の変更	13
CHART 関数 - 動的チャート	14
CRITERIA 関数 - 条件スクリプトの表示	20
DATAENTRY 関数 - レポートの編集	21
EDITPOV 関数 - データの視点の変更	24
EXCELREP 関数 - スプレッドシートの実行	26
ログアウト	28
REPCHART 関数 - レポートとチャートの並列表示	28
REPORT 関数 - 動的レポート	31
PRINTREP 関数 - PDF ファイルへのレポート出力	35
SCHEDREP 関数 - Hyperion Distributed Schedules の起動	37
ライブラリ関数	39

オラクル社の Hyperion(R) Enterprise(R)アプリケーションには、Hyperion Enterprise Reporting Web サーバにインストールするオラクル社の Hyperion(R) Enterprise(R) Reporting Web ダイナミックリンクライブラリ (SPIDER.DLL) に含まれている関数を使用してアクセスします。

この関数は、次のいずれかの方法で呼び出します。

- SPIDER.DLL への呼び出しの一部としてブラウザの URL 行に関数を入力する
- HREF パラメータの一部として関数を含む HTML 文書に URL 行を作成する

関数呼び出しの書式

関数の呼び出しでは、ここで説明する書式を使用します。

注： 関数の構文の太字部分は、示されているとおりに正確に入力する必要があります。

http://Virtual Path/spider.dll?Function&Parameters

変数	説明
Virtual Path	SPIDER.DLL ファイルの仮想パス。例えば、一般的な仮想パスとして server/HSPIDER などがあります。ここで、server はサーバ名です。
Function	SPIDER.DLL 内で呼び出す関数。関数の前には常に疑問符 (?) を配置します。 利用可能な SPIDER.DLL 関数については、次の表を参照してください。
Parameters	使用する任意の関数パラメータ。必須のパラメータとオプションのパラメータがあります。パラメータリストはアンパサンド (&) で開始します。複数のパラメータを指定する場合は、個々のパラメータをアンパサンド (&) で区切ります。 それぞれのパラメータにはラベルもあります。通常、このラベルは短縮されたパラメータ名で、等号 (=) が後ろに続きます。例えば、sys= は、System パラメータのラベルです。 各関数のパラメータについては、該当する関数のトピックを参照してください。

次の要素は、すべての関数呼び出しで必要です。

- http:// - 関数呼び出しの標準ヘッダー
- spider.dll - 関数を含む Hyperion Enterprise Reporting Web ダイナミックリンクライブラリの名前
- ? - 関数名の前に配置する文字

関数のリスト

次の表で Hyperion Enterprise Reporting Web の関数について説明します。

表 1 Hyperion Enterprise Reporting(TM) Web アプリケーションの関数

関数	用途
AGENTFORM	[Subscriptions (購読)] Web ページで [Agent Subscription (エージェント購読)] フォームまたは [Modifications (変更)] Web ページで [Modification (変更)] フォームを表示します。
AGENTLIST	ユーザがまだ購読していないすべてのエージェント、ユーザが購読したすべてのエージェント、またはユーザが表示していない通知があるすべてのエージェントを一覧にする Web ページを生成します。
AGENTSSENT	表示されていない通知の数を一覧にする Web ページを生成します。
APPLIST	指定したオラクル社の Enterprise Performance Management ソフトウェアの全アプリケーションを一覧にする Web ページが生成されます。Oracle Essbase の場合、サーバ、アプリケーション、またはデータベースも一覧にできます。
CHANGEID	現在の Hyperion Enterprise アプリケーションのログイン情報を変更します。
CHART	レポートのチャートを表示する Web ページを生成します。
CRITERIA	条件スクリプトを含む Web ページを生成します。
DATAENTRY	ユーザがデータを入力できるレポートを含む Web ページを生成します。

関数	用途
EDITPOV	特定のアプリケーションの [Change Point of View (データの視点の変更)] フォームを表示して、データの視点を設定します。
EXCELREP	レポートデータをスプレッドシートとして表示します。
LOGOUT	ユーザが接続しているすべてのアプリケーションとの接続を切断します。
REPCHART	左右に分割された表示領域を含む Web ページを生成します。それぞれのフレームには、個別のレポートまたはチャートが表示されます。
REPORT	レポート用の Web ページを生成します。
PRINTREP	レポートを Adobe Acrobat .PDF ファイルに出力します。
REPSETLIST	指定したオラクル社の Enterprise Performance Management ソフトウェア、アプリケーション、およびレポートセットのレポートをすべて一覧にする Web ページを生成します。
SCHEDREP	レポートデータを Hyperion Distributed Schedules に表示します。 これは SHEXT.DLL 関数です。
SETLIST	指定したオラクル社の Enterprise Performance Management ソフトウェアおよびアプリケーションのすべてのレポートセットを一覧にする Web ページが生成されます。
SYSLIST	Hyperion Enterprise Reporting からアクセスできる Web サーバにインストールされたオラクル社の Enterprise Performance Management ソフトウェアをすべて一覧にする Web ページを生成します。これは、Windows ディレクトリにある REPENG.INI ファイルで指定されます。

SPIDER.DLL - 例

これらの例は、SPIDER.DLL を URL アドレスおよびリンクの HREF ステートメントとして使用する方法を示します。

例 1 : Web ブラウザの URL アドレスを使用する

この例は、SPIDER.DLL の REPORT 関数を使用して、SERVER1 という Web サーバにインストールされた Hyperion Enterprise SALES アプリケーションの Forecast レポート用の Web ページを生成します。

```
http://server1/HSPIDER/spider.dll?report&enterprise&sales&forecast
```

例 2 : リンクの HREF ステートメントを使用する

この例も同じコマンドを示しますが、この例では HTML 文書のリンクの HREF タグとしてコマンドを使用します。

```
<A HREF="/HSPIDER/spider.dll?report&enterprise&sales&forecast">
```

Essbase の構文

Hyperion Enterprise Reporting Web を Essbase で使用している場合は、いくつかの SPIDER.DLL 関数の使用に必要な構文が他のオラクル社の Enterprise Performance Management ソフトウェアとは異なります。

System は、SPIDER.DLL 関数の構文でパラメータとして使用される場合は、Hyperion Enterprise アプリケーションなど、オラクル社の Enterprise Performance Management ソフトウェアを指します。Essbase ユーザは、SPIDER.DLL 関数で System の代わりに次の構文を使用してください。

Essbase-servername\Essbase application name

変数	説明
Essbase-servername	REPENG.INI ファイル内の Essbase ドライバ名と Hyperion Essbase サーバ名をハイフンで区切ったもの。
Essbase application name	Essbase アプリケーションの名前。

Application は、SPIDER.DLL 関数の構文でパラメータとして使用される場合は、Hyperion Enterprise アプリケーションなど、オラクル社の Enterprise Performance Management ソフトウェアを指します。Essbase ユーザは、SPIDER.DLL 関数の構文で Application を Essbase データベースの名前に置き換えてください。

AGENTFORM 関数 - エージェント購読の表示／更新

AGENTFORM 関数は、ユーザがエージェントを購読できる Web ページまたは既存のエージェント購読を変更するページを表示するために使用されます。システムは、この関数を使用して、ユーザが購読するエージェントを選択または変更した後に Web ページを表示します。表示される Web ページには、選択したエージェントに対してユーザが設定する必要があるオプションを含むフォームが表示されます。

注： この関数はシステム管理者が使用すべきものではありません。この関数は、Hyperion Enterprise Reporting Web の内部で使用されます。

AGENTFORM 関数の構文

[Subscriptions (購読)] または [Modifications (変更)] ページを表示するには、次の構文を使用します。

<http://Path/spider.dll?AgentForm&type=FormType&buttons=UseButtons &Agent=ID>

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。

変数	説明
FormType	次のいずれかの値。 <ul style="list-style-type: none"> ● Modify : [Subscription Modification (購読の変更)] フォームを含む Web ページが表示されます。 ● Subscribe : [Subscription (購読)] フォームを含む Web ページが表示されます。
UseButtons	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : ページ上部に [通知の表示]、[エージェントの登録] および [Modify Subscriptions (購読の変更)] ボタンが表示されます。 ● No : [通知の表示]、[エージェントの登録] および [Modify Subscriptions (購読の変更)] ボタンはページ上部に表示されません。ページの別のフレームにボタンが表示される場合は、No を指定します。
ID	Agent_Definition データベース表で定義されているエージェントの数値 ID

AGENTLIST 関数 - エージェントの一覧

AGENTLIST 関数は、以下のいずれかの項目を一覧にする Web ページを表示するために使用します。

- ユーザがセキュリティ権限を持っていて、まだ購読していないすべてのエージェント
- ユーザが現在購読しているすべてのエージェント
- ユーザが表示していない通知があるすべてのエージェント

システムは、この関数を使用して、ユーザが最初のエージェント Web ページで該当するボタンを選択するたびに適切な Web ページを表示します。

AGENTLIST 関数の構文

エージェントの一覧を生成するには、次の構文を使用します。

`http://Path/spider.dll?AgentList&type=ListType&buttons=UseButtons`

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
ListType	次のいずれかの値。 <ul style="list-style-type: none"> ● Available : ユーザがまだ購読していないすべてのエージェントが表示されます。 ● Subscribed : 現在購読しているすべてのエージェントが表示されます。 ● Triggered : 表示されていない通知のあるすべてのエージェントの一覧が表示されます。
UseButtons	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : ページ上部に [通知の表示]、[エージェントの登録] または [Modify Subscriptions (購読の変更)] ボタンが表示されます。

変数	説明
	<ul style="list-style-type: none"> No : [通知の表示]、[エージェントの登録] および [Modify Subscriptions (購読の変更)] ボタンはページ上部に表示されません。ページの別のフレームにボタンが表示される場合は、No を指定します。

AGENTLIST 関数の例

この例は、AGENTLIST 関数を使用してユーザにエージェントの一覧を表示する Web ページの一部を示します。

```
<H2>情報マネージャ</H2><P>
<B>表示する情報を選択してください。</B><BR>
<A HREF="/HSPIDER/spider.dll?AgentList&type=Available
&buttons=yes">未購読のすべてのエージェントを表示</A><BR>
<A HREF="/HSPIDER/spider.dll?AgentList&type=Subscribed&buttons=yes">購読
しているすべてのエージェントを表示</A><BR>
<A HREF="/HSPIDER/spider.dll?AgentList&type=Triggered&buttons=yes">表示
待ちの通知を受けているすべてのエージェントを表示</A><BR>
```

AGENTSSSENT 関数 - エージェント通知の表示

AGENTSSSENT 関数は、ユーザが待機しているエージェント通知の数を一覧にする Web ページを表示するために使用します。表示される Web ページの概観をカスタマイズするには、関数のパラメータを使用します。システムは、ユーザがブラウザに初めてエージェントページを入力したときに、この関数を使用します。

AGENTSSSENT 関数の構文

購読ユーザに送信されたエージェント通知の数を含む Web ページを生成するには、次の構文を使用します。

http://Path/spider.dll?AgentsSent&buttons=UseButtons

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
UseButtons	Yes : [通知の表示]、[エージェントの登録] または [Modify Subscriptions (購読の変更)] ボタンが表示されます。ページの別のフレームにボタンが表示される場合は、No を指定します。

AGENTSSSENT 関数の例

この例は、ユーザが待機しているエージェント通知の数を一覧にするために AGENTSSSENT 関数を使用する Web ページの一部を示します。

```
<H2>情報マネージャ</H2><P>
<B>表示する情報を選択してください。</B><BR>
```


エージェント通知を表示する

CHANGEID 関数 - Hyperion Enterprise アプリケーションのログオン情報の変更

CHANGEID 関数を使用して、Hyperion Enterprise アプリケーションの現在のログオン情報を変更できます。ユーザが CHANGEID 関数を呼び出すと、ブラウザにオラクル社の Enterprise Performance Management ソフトウェアの [アプリケーションのログオン] ページが表示され、アプリケーションの新しいユーザ名およびパスワードを入力できます。

Web ブラウザから URL として関数呼び出しを直接発行すると、自分のログオン情報を変更できます。ユーザが自分のログオン情報を変更できるようにするには、HREF パラメータの値として URL を含むリンクを HTML 文書に作成します。

CHANGEID 関数の構文

ユーザのアプリケーションログイン情報を変更するには、次の構文を使用します。

`http://Path/spider.dll?changeid&[sys=]System&[app=]Application[&file=Template.htm]`

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	アクセスするシステムまたはオラクル社の Enterprise Performance Management ソフトウェアのラベル。CHANGEID リクエストでシステムを指定する必要があります。システムラベルは、REPENG.INI ファイルの[DRIVERS32]セクションで定義します。これは必須のパラメータです。 Essbase 構文について詳しくは、 10 ページの「Essbase の構文」 を参照してください。
Application	アクセスする Hyperion Enterprise アプリケーションのラベル。System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。これは必須のパラメータです。
Template.htm	使用可能なオプションの HTML テンプレート。CHANGEID 関数が実行された後に表示される Web ページを変更するには、ユーザ定義テンプレートを指定します。一般的に、テンプレートは、実行する関数で%PREVQUERY%変数が置き換えられた SIGNON.HTX のカスタムバージョンです。

CHANGEID パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application の順で指定する必要があります。

- 1つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

CHANGEID 関数の例

この例は、ユーザによる Hyperion Enterprise SALES および TAX アプリケーションのログオン ID を変更できるようにするために CHANGEID 関数を使用する Web ページを示します。

```
<H2>ログオンIDの変更</H2><P>
<B>アプリケーションの選択:</B><BR>
<A HREF="/HSPIDER/spider.dll
?changeid&enterprise&sales">
Salesアプリケーション</A>
<A HREF="/HSPIDER/spider.dll
?changeid&enterprise&tax">
Taxアプリケーション</A>
```

CHART 関数 - 動的チャート

CHART 関数を使用して SPIDER.DLL への関数呼び出しを行うことにより、Hyperion Enterprise アプリケーションの任意のレポートのチャートを表示する Web ページを動的に生成できます。Web ブラウザから URL として関数呼び出しを直接発行するか、HREF パラメータの値として URL を含むリンクを HTML 文書に作成できます。

CHART 関数で指定できるオプションを次に示します。

- 個別のフレームまたはチャート本体にデータの視点ボタンを含める
- 別のテンプレートまたはユーザ定義テンプレートを使用する
- データの視点要素を設定する
- 水平軸に行データを表示するか列データを表示するかを指定する
- 最大 23 種類のチャートから選択する
- チャートの幅を変更する
- チャートラベルのフォントの種類とサイズを変更する
- 凡例を表示する
- チャートタイトルを含める
- チャート作成ツールを指定する
- カスタムチャート用の Microsoft Excel のオートフォーマットテンプレートを指定する

CHART 関数の構文

レポートのチャートを動的に生成するには、次の構文を使用します。

```
http://Path/spider.dll?chart&[sys=]System&[app=]Application&[rpt=]Report  
[&frame=UseFrames][&tf=Template.htx][&tview=Template2.htx]  
[&tpov=Template3.htx][&Keyname=Keyvalue][&piv=Pivot][&ct=ChartType]  
[&wid=Width][&fon=Font][&fos=Size][&leg=Legend][&ctitle=Text]  
[&chrteng=ChartTool][&exceltempl=Temp][&ht = Height]
```

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	アクセスするシステムまたはオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPENG.INI ファイルの[DRIVERS32]セクションで定義します。 Essbase 構文について詳しくは、 10 ページの「Essbase の構文」 を参照してください。
Application	チャートを生成する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。
Report	実行するレポートのラベル。これは必須のパラメータです。Application パラメータで指定されているアプリケーションの任意のレポートを選択できます。
UseFrames	次のいずれかの値。 <ul style="list-style-type: none">● Yes : 個別のフレームにデータの視点ボタンが表示されます。これがデフォルト値です。● No : レポートまたはチャートの本体にデータの視点ボタンが表示されず。ブラウザがフレームをサポートしていない場合は、No を指定します。
Template.htx	レポートに使用する HTML テンプレート (UseFrames が Yes に設定されている場合)。デフォルトは FRAMES.HTX です。
Template2.htx	レポートに使用する HTML テンプレート。UseFrames が Yes に設定されている場合のデフォルトは FREPORT.HTX です。UseFrames が No に設定されている場合のデフォルトは REPORT.HTX です。
Template3.htx	レポートのデータの視点オプションで使用する HTML テンプレート (UseFrames が Yes に設定されている場合)。デフォルトは FPOV.HTX です。
Keyname	設定するデータの視点ディメンション。
Keyvalue	Keyname に割り当てる値。有効な URL 文字を使用する必要があります。例えば、データの視点ディメンション値のスペースはプラス記号 (+) に置き換えることができます。従って、「JLY 97」の期間は、チャートリクエストで「JLY +97」になります。 注： 現在の月を指定するには、期間を@curmonth に設定します。SPIDER.INI ファイルの[Default]セクションで@curmonth パラメータを設定した場合、CHART 関数呼び出しの期間は、その月に設定されます。例えば、SPIDER.INI ファイルで@curmonth=january と設定し、ダイナミックリンクの CHART 関数呼び出しでperiod=@curmonth と設定した場合、表示される動的レポートのデータの視点は 1 月 (january) に設定されます。SPIDER.INI ファイルで@curmonth を設定しない

変数	説明
	場合、表示される動的レポートのデータの視点は、Windows NT の現在の月に設定されます。SPIDER.INI ファイルで@curmonth を設定しない場合、アプリケーションで使用する月の表記は、コンピュータが使用する長い形式の月の表記に正確に一致している必要があります。
Pivot	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : データの行および列がスワップまたはピボット（再配置）されます。これがデフォルト値です。 ● No : 水平軸に沿って行が、垂直軸に沿って列が表示されます。
ChartType	作成するグラフの種類。デフォルト値は 2（3 次元線）です。これらのチャートの種類については、17 ページの「 ChartType パラメータの値 」を参照してください。
Width	ピクセル単位でのチャートの幅。デフォルト値は 500 です。
Font	ページのテキストに使用するフォントの名前。デフォルト値は Arial です。
Size	チャート軸のラベルに使用するポイント単位でのフォントサイズ。デフォルト値は 12 です。
Legend	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : 色分けされた凡例が含まれます。 ● No : 凡例は含まれません。これがデフォルト値です。
Text	チャートタイトル。有効な URL 文字を使用する必要があります。例えば、チャートタイトルに含まれるスペースはプラス記号（+）に置き換えることができます。例えば、「Key Stocks」というタイトルは、チャートリクエストでは「Key+Stocks」になります。デフォルトはタイトルなしです。
ChartTool	次のいずれかの値。 <ul style="list-style-type: none"> ● 1 : Visual Components First Impression バージョン 2.0 チャート作成ツールが使用されます。これがデフォルト値です。 ● 2 : Microsoft Excel のチャート作成ツールが使用されます。Excel のチャート作成ツールを使用する場合、Hyperion Enterprise Reporting サーバに Microsoft Excel がインストールされている必要があります。
Temp	使用する Microsoft Excel カスタムチャートまたはオートフォーマットテンプレートの名前。このオプションは、ChartType、Font、Size、Legend、および Text パラメータよりも優先され、チャートリクエストで ChartTool を 2 に設定した場合にのみ有効です。デフォルトはテンプレートなしです。
Height	ピクセル単位でのチャートの高さ。

CHART パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application、Report の順で指定する必要があります。

- 1つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

ChartType パラメータの値

ChartType パラメータは、生成するチャートの種類を指定するために使用します。選択できるチャートの種類は Visual Components First Impression チャート作成ツールを使うか、Microsoft Excel を使うかで異なります。

次の場合に Visual Components First Impression チャート作成ツールを使用します。

- Microsoft Excel が Hyperion Enterprise Reporting サーバにインストールされていない場合。
- 組み合わせチャート、バブルチャート、ガントチャートなど Excel がサポートしないチャートを作成する場合。
- First Impression のチャートの外観が望ましい場合

次の場合は Microsoft Excel のチャート作成ツールを使用します。

- Microsoft Excel が Hyperion Enterprise Reporting サーバにインストールされている場合。
- 等高線グラフ、二次元ドーナツグラフ、ユーザ定義のグラフなど First Impression がサポートしないチャートを作成する場合。
- Excel のチャートの外観が望ましい場合

Visual Components First Impression のチャート作成ツールを使用する場合は、次の表にあるチャートの種類を選択できます。

表 2 First Impression のチャートの種類

チャートの種類	値
3次元縦棒	0
2次元縦棒	1
3次元折れ線	2
2次元折れ線	3
3次元面	4
2次元面	5
3次元階段状	6
2次元階段状	7
3次元複合	8

チャートの種類	値
2 次元複合	9
3 次元横棒	10
2 次元横棒	11
3 次元集合棒	12
3 次元円	13
2 次元折れ線	14
3 次元ドーナツ	15
2 次元 XY 線	16
2 次元ポーラー	17
2 次元レーダー	18
2 次元バブル	19
2 次元等高線	20
2 次元ガント	21
3 次元ガント	22

Microsoft Excel のチャート作成ツールを使用する場合は、次の表にあるチャートの種類を選択できます。

表 3 Excel チャートの種類

チャートの種類	値
3 次元縦棒	0
2 次元縦棒	1
3 次元折れ線	2
2 次元折れ線	3
3 次元面	4
2 次元面	5
3 次元横棒	10
2 次元横棒	11
3 次元円	13
2 次元円	14
2 次元 XY 散布図	16

チャートの種類	値
2次元レーダー	18
2次元ドーナツ	24
カスタム	25

CHART 関数の例

この例は、Hyperion Enterprise Reporting Web を使用して、ユーザ定義のテンプレート、さまざまなデータの視点要素またはキー、およびさまざまなチャートオプションを指定し、フレームのあるチャートとフレームのないチャートを表示する方法を示します。

例 1：チャートの実行

CHART 関数を使用して Web ページにチャートを表示できます。この例では、Hyperion Enterprise DEMO アプリケーションの RPPL1 レポートのチャートが生成されます。

```
<A HREF="http://spider/HSPIDER/spider.dll
?chart&EnterpriseEnterprise&Demo&RPPL1">
```

例 2：フレームのないチャートの実行

フレームを指定するパラメータを使用して、フレームをサポートしないブラウザで表示可能なチャートを作成します。チャートページの本体にすべてのデータの視点ボタンが表示されます。次にその例を示します。

```
<A HREF="http://spider/HSPIDER/spider.dll
?chart&Enterprise&Demo&RPPL1&frame=no">
```

例 3：ユーザ定義テンプレートの使用

tview パラメータを使用して、ユーザ定義のテンプレートを指定できます。この例は、ユーザ定義テンプレートの MYTEMP.HTX を使用してチャートを作成します。

```
<A HREF="http://spider/HSPIDER/spider.dll
?chart&Enterprise&Demo&RPPL1&tview=mytemp.htx">
```

例 4：データの視点ディメンションのリセット

任意のデータの視点ディメンションのデフォルトのデータの視点を上書きできます。この例では、チャートのエンティティ、データ種別および期間のデータの視点ディメンションを設定します。

```
<A HREF="http://spider/HSPIDER/spider.dll
?chart&Enterprise&Demo&RPPL1&entity=uk.total
&category=lastyr&period=jly+97">
```

CRITERIA 関数 - 条件スクリプトの表示

CRITERIA 関数は、条件スクリプトの結果を表示する Web ページを生成するために使用します。システムはテンプレートをスキャンして変数を取得し、ユーザがテンプレートの各変数に定義した条件を評価して、条件が満たされているかどうかに基づいて表示を変更します。

CRITERIA 関数の構文

条件スクリプト Web ページを生成するには、この構文を使用します。

`http://Path/spider.dll?criteria&Cbp_id[&Keyname=Keyvalue] [&Frame=yes/no]`

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
Cbp_id	Hyperion Enterprise Reporting Web Application Administration プログラムで定義された条件スクリプトのラベル。このページがアプリケーションディレクトリにある場合は、ファイル名の拡張子を省略できます。このページがアプリケーションディレクトリ以外にある場合は、ファイル名と拡張子を含むフルパスを参照する必要があります。
Keyname	<p>設定するデータの視点ディメンション。keyname は、アプリケーションまたは製品とアプリケーションのいずれかで修飾できます。例えば、keyname に次の値を指定できます。</p> <p>Entity=Europe</p> <p>この場合、Ent5 は製品（Hyperion Enterprise 5）、Demo はアプリケーション、および Entity=Europe はデータの視点ディメンションです。</p>
Keyvalue	<p>Keyname に割り当てる値。ピリオドはすべて、データの視点ディメンション値に表示されるとおりに入力してください。また、データの視点ディメンション値のスペースはプラス記号（+）に置き換える必要があります。例えば、「JLY 97」の期間は、レポートリクエストで「JLY+97」になります。</p> <p>注： 現在の月を指定するには、期間を@curmonth に設定します。SPIDER.INI ファイルの[Default]セクションで@curmonth パラメータを設定した場合、CRITERIA 関数呼び出しの期間は、その月に設定されます。例えば、SPIDER.INI ファイルで @curmonth=january と設定し、リンクの CRITERIA 関数呼び出しで period=@curmonth と設定した場合、表示されるレポートのデータの視点は 1 月（january）に設定されます。SPIDER.INI ファイルで@curmonth を設定しない場合、表示されるレポートのデータの視点は、Windows NT の現在の月に設定されます。SPIDER.INI ファイルで @curmonth を設定しない場合、アプリケーションで使用する月の表記は、コンピュータが使用する長い形式の月の表記に正確に一致している必要があります。</p>
Yes/No	フレームを使用するための引数。

CRITERIA 関数の例

この例は、CRITERIA 関数を使用してエンティティが Europe に設定された条件スクリプト SALES_PERF を表示する Web ページを示します。

http://Path/spider.dll?criteria&sales_perf&entity=
europe

DATAENTRY 関数 - レポートの編集

DATAENTRY 関数は、SPIDER.DLL への関数呼び出しを行って、Hyperion Enterprise アプリケーションの任意のレポートをデータ入力モードで動的に表示する Web ページを生成するために使用します。データ入力モードでは、ユーザは、表をサポートするブラウザを使用して、インターネットまたはイントラネットから Hyperion Enterprise または Essbase アプリケーションでデータを入力または編集できます。Web ブラウザから URL として DATAENTRY 関数呼び出しを直接発行するか、HREF パラメータの値として URL を含むリンクを HTML 文書に作成できます。

注： データ入力レポートをセットアップするときは、セルに特殊文字が含まれないように注意する必要があります。例えば、セルに\$文字が含まれている場合、データ入力には正常に機能しません。

DATAENTRY 関数で指定できるオプションを次に示します。

- 個別のフレームまたはレポート本体にデータの視点ボタンを含める
- 別のテンプレートまたはユーザ定義のレポートテンプレートを使用する
- データの視点ディメンションを設定する
- Web ブラウザの種類を指定する（Microsoft Internet Explorer 6.0 または Mozilla Firefox 2.0.x のみ）
- レポートをスクロールするときに行と列の見出しを保護する（Microsoft Internet Explorer 6.0 またはそれ以降のみ）

注： データ入力モードでは複合レポートは表示できません。

DATAENTRY 関数の構文

データ入力モードでレポートを動的に生成するには、次の構文を使用します。

```
http://Path/spider.dll?dataentry&[sys=]System&[app=]Application  
&[rpt=]Report[&frame=UseFrames][&file=Template.htx] [&view=Template2.htx]  
[&tpov=Template3.htx][&Keyname=Keyvalue] [&browser=BrowserType]  
[&lockhdrs=LockHeaders]
```

DATAENTRY パラメータを使用する際のルール

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。

System	<p>アクセスするシステムまたはオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPENG.INI ファイルの[DRIVERS32]セクションで定義します。</p> <p>Essbase 構文について詳しくは、10 ページの「Essbase の構文」を参照してください。</p>
Application	<p>データ入力モードでレポートを生成する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。</p> <p>System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。</p>
Report	<p>データ入力モードで実行するレポートのラベル。これは必須のパラメータです。</p> <p>Application パラメータで指定されているアプリケーションの任意のレポートを選択できます。</p>
UseFrames	<p>次のいずれかの値。</p> <ul style="list-style-type: none"> ● Yes : 個別のフレームにデータの視点ボタンが表示されます。これがデフォルト値です。 ● No : レポートの本体にデータの視点ボタンが表示されます。ブラウザがフレームをサポートしていない場合は、No を指定します。
Template.htx	<p>レポートで使用するテンプレート（UseFrames が Yes に設定されている場合）。デフォルトは FRAMES.HTX です。</p>
Template2.htx	<p>レポートに使用する HTML テンプレート。</p> <p>このパラメータのデフォルト値については、23 ページの「Template2.htx パラメータのデフォルト値」を参照してください。</p>
Template3.htx	<p>レポートのデータの視点オプションで使用する HTML テンプレート（UseFrames が Yes に設定されている場合）。デフォルトは FPOV.HTX です。</p>
Keyname	<p>設定するデータの視点ディメンション。</p>
Keyvalue	<p>Keyname に割り当てる値。ピリオドはすべて、データの視点ディメンション値に表示されるとおりに入力してください。また、データの視点ディメンション値のスペースはプラス記号 (+) に置き換える必要があります。例えば、「JLY 97」の期間は、レポートリクエストで「JLY+97」になります。</p> <p>注： 現在の月を指定するには、期間を@curmonth に設定します。SPIDER.INI ファイルの[Default]セクションで@curmonth パラメータを設定した場合、DATAENTRY 関数呼び出しの期間は、その月に設定されます。例えば、SPIDER.INI ファイルで@curmonth=january と設定し、ダイナミックリンクの DATAENTRY 関数呼び出しで period=@curmonth と設定した場合、表示される動的レポートのデータの視点は 1 月 (january) に設定されます。SPIDER.INI ファイルで@curmonth を設定しない場合、表示される動的レポートのデータの視点は、Windows NT の現在の月に設定されます。SPIDER.INI ファイルで@curmonth を設定しない場合、アプリケーションで使用する月の表記は、コンピュータが使用する長い形式の月の表記に正確に一致している必要があります。</p>
BrowserType	<p>次のいずれかの値。</p> <ul style="list-style-type: none"> ● IE40 : Microsoft Internet Explorer 6.0.またはそれ以降を Web ブラウザとして指定します。 ● Netscape40 : Mozilla Firefox 2.0.x 以降を Web ブラウザに指定します。 ● 任意の値 : 別のブラウザを指定します。例えば、古いブラウザを使用する場合は、OTHER を指定します。

	<ul style="list-style-type: none"> ● BrowserType パラメータを使用すると、管理プログラムのデフォルトブラウザの設定を上書きできます。このオプションの設定については、Administration プログラムのオンラインヘルプを参照してください。
LockHeaders	<p>次のいずれかの値。</p> <ul style="list-style-type: none"> ● Yes : ユーザがレポートをスクロールする際に行と列の見出しが保護されます。 ● No : 行と列の見出しは保護されません。 <p>LockHeaders パラメータを使用すると、管理プログラムの行および列見出しのデフォルトの動作を上書きできます。このオプションの設定については、Administration プログラムのオンラインヘルプを参照してください。</p>

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application、Report の順で指定する必要があります。
- 1つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

Template2.htx パラメータのデフォルト値

次の表に DATAENTRY 関数の Template2.htx パラメータのデフォルト値を一覧表示します。デフォルト値は、フレームを使用するかどうか、または行と列の見出しを保護するかどうかは、使用するブラウザによって異なります。

表 4 DATAENTRY 関数の Template2.htx パラメータのデフォルト値

	Microsoft Internet Explorer 6.0 以降	Mozilla Firefox 2.0.x
見出しの保護なし、フレームなし	DEREPIE.HTX	DEREPNS.HTX
見出しの保護なし、フレームあり	FDEREPIE.HTX	FDEREPIE.HTX
見出しの保護あり、フレームなし	DELHRIE.HTX	DELHRNS.HTX
見出しの保護あり、フレームあり	FDELHRIE.HTX	FDELHRNS.HTX

DATAENTRY 関数の例

この例は、各種 Web ブラウザを使って Hyperion Enterprise Reporting Web のデータ入力モードでレポートを表示する方法を示します。

例 1：データ入力モードでのレポートの表示

管理者によって Administration プログラムでデータ入力モードが有効にされ、デフォルトブラウザの自動検出が設定されています。管理者は、Web ブラウザの URL リクエストで次の関数呼び出しを入力して、Hyperion Enterprise DEMO アプリケーションで ENTRY レポートをデータ入力モードで表示します。

```
http://spider/HSPIDER/spider.dll?dataentry  
&Enterprise&DEMO&entry
```

例 2：行と列の見出しが保護されたレポートのデータ入力モードでの表示

管理者によって Administration プログラムでデータ入力モードが有効にされ、行と列の見出しの保護が無効に設定されています。しかし、管理者は、Hyperion Enterprise ENTRY レポートをデータ入力モードで表示し、このレポートの行と列の見出しを保護したいと考えています。管理者は、次の関数呼び出しを URL に入力します。

```
http://spider/HSPIDER/spider.dll?dataentry  
&Enterprise&DEMO&entry&lockhdrs=yes
```

例 3：行と列の見出しが保護されないレポートのデータ入力モードでの表示

管理者によって Administration プログラムでデータ入力モードが有効にされ、行と列の見出しの保護有効に設定されています。しかし、管理者は、Hyperion Enterprise ENTRY レポートをデータ入力モードで表示し、このレポートの行と列の見出しの保護を解除したいと考えています。管理者は、次の関数呼び出しを URL に入力します。

```
http://spider/HSPIDER/spider.dll?dataentry  
&Enterprise&DEMO&entry&lockhdrs=no
```

EDITPOV 関数 - データの視点の変更

EDITPOV 関数を使用して、アプリケーションの現在のデータの視点を変更できます。ユーザが EDITPOV 関数を使用すると、指定したアプリケーションの [Change Point of View (データの視点の変更)] フォームが表示されます。ブラウザから URL として関数呼び出しを直接発行するか、HREF パラメータの値として URL を含むリンクを HTML 文書に作成できます。

新しいデータの視点を選択して [完了] をクリックすると、変更を確認する Web ページが表示されます。このページの外観は、テンプレート CHPOV.HTX によって定義されます。このファイルを編集して確認ページのデフォルトの外観を変更することや、CHPOV.HTX の代わりに使用するユーザ定義のテンプレートを作成することができます。テンプレートについて詳しくは、『管理者用リファレンスガイド』を参照してください。

EDITPOV 関数の構文

アプリケーションのデータの視点を変更するには、次の構文を使用します。

`http://Path/spider.dll?editpov&[sys=]System&[app=]Application [&tfile=Template.htx]`

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	アクセスするシステムまたはオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPEING.INI ファイルの[DRIVERS32]セクションで定義します。 Essbase 構文について詳しくは、 10 ページの「Essbase の構文」 を参照してください。
Application	データの視点を変更する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。 System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。
Template.htx	データの視点の変更ページを表示するために使用されるテンプレート。デフォルトは POVEDIT.HTX です。 ユーザが [完了] を選択した後に表示される確認ページの外観を変更するには、POVEDIT.HTX（またはその代りに使用するテンプレート）の[Form Action]セクションに次の行を挿入します。 <INPUT TYPE="hidden" NAME="pfile" VALUE="ConfirmationTemplate.htx"> ここで、ConfirmationTemplate.htx は、ユーザが [完了] をクリックした後に表示されるユーザ定義の HTML ページです。

EDITPOV パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application の順で指定する必要があります。
- 1 つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

EDITPOV 関数の例

この例は、ユーザによる Hyperion Enterprise SALES および TAX アプリケーションのデータの視点を変更できるようにするために EDITPOV 関数を使用する Web ページを示します。

```

<H2>データの視点</H2><P>
<B>アプリケーションの選択:</B><BR>
<A HREF="/HSPIDER/spider.dll
?editpov&enterprise&sales">
Salesアプリケーション</A>
<A HREF="/HSPIDER/spider.dll
?editpov&enterprise&tax">
Taxアプリケーション</A>

```

EXCELREP 関数 - スプレッドシートの実行

EXCELREP 関数は、レポートデータをスプレッドシートとして表示するために使用します。システムでスプレッドシートリクエストが実行されると、要求されたレポートデータは、ユーザのローカルスプレッドシートツールのメニューおよびコントロールを備えたスプレッドシートとして表示されます。ユーザは、スプレッドシートのデータを移動して、データを変更できます。これらの変更は、データベースに保存されません。

EXCELREP 関数の構文

レポートデータをスプレッドシートとして表示するには、次の構文を使用します。

**http://Path/spider.dll?excelrep&[sys=]System&[app=]Application
&[rpt=]Report[&fmt=Format][&Keyname=Keyvalue]**

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	<p>アクセスするシステムまたはオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPENG.INI ファイルの[DRIVERS32]セクションで定義します。</p> <p>Essbase 構文について詳しくは、10 ページの「Essbase の構文」を参照してください。</p>
Application	<p>データの視点を変更する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。</p> <p>System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。</p>
Report	<p>実行するレポートのラベル。これは必須のパラメータです。</p> <p>Application パラメータで指定されているアプリケーションの任意のレポートを選択できます。</p>
書式	表示するスプレッドシートの書式。書式のリストについては、 27 ページの「Format パラメータ値」 を参照してください。
Keyname	設定するデータの視点ディメンション。
Keyvalue	Keyname に割り当てる値。ピリオドはすべて、データの視点ディメンション値に表示されるとおりに入力してください。また、データの視点ディメンション値

変数	説明
	<p>のスペースはプラス記号 (+) に置き換える必要があります。例えば、「JLY 97」の期間は、レポートリクエストで「JLY+97」になります。</p> <p>注： 現在の月を指定するには、期間を@curmonth に設定します。SPIDER.INI ファイルの[Default]セクションで@curmonth パラメータを設定した場合、EXCELREP 関数呼び出しの期間は、その月に設定されます。例えば、SPIDER.INI ファイルで @curmonth=january と設定し、リンクの EXCELREP 関数呼び出しで period=@curmonth と設定した場合、表示されるスプレッドシートのデータの視点は 1 月 (january) に設定されます。SPIDER.INI ファイルで@curmonth を設定しない場合、表示されるスプレッドシートのデータの視点は、Windows NT の現在の月に設定されます。SPIDER.INI ファイルで@curmonth を設定しない場合、アプリケーションで使用する月の表記は、コンピュータが使用する長い形式の月の表記に正確に一致している必要があります。</p>

EXCELREP パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application、Report の順で指定する必要があります。
- 1 つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

Format パラメータ値

次の表に EXCELREP 関数の Format パラメータで使えるスプレッドシートのすべての書式を一覧表示します。

表 5 EXCELREP 関数で使用可能な書式

書式	説明
XLS (デフォルト), -4143	サーバに現在インストールされている Excel のバージョン
XLS5	Excel 5/95 (.XLS)
CSV, 6	カンマ区切りテキスト
TXT, -4158	タブ区切りテキスト
20	タブ区切りテキスト (Windows) (.TXT)
23	カンマ区切りテキスト (Windows) (.CSV)

EXCELREP 関数の例

この例は、EXCELREP 関数を使用して、ユーザが Hyperion Enterprise SALES アプリケーションのレポート REVENUE を Microsoft Excel または Lotus WK4 書式のスプレッドシートとして表示できる Web ページを示します。

```
<H2>REVENUEレポート</H2><P>
<B>スプレッドシートの書式を選択してください。</B><BR>
<A HREF="/HSPIDER/spider.dll
?excelrep&Enterprise&sales&REVENUE&fmt=XLS">
Microsoft Excel Format</A>
<A HREF="/HSPIDER/spider.dll
?excelrep&Enterprise&sales&REVENUE&fmt=WK4">
Lotus WK4書式</A>
```

ログアウト

ログアウト関数は、ユーザが接続しているすべてのアプリケーションとの接続を切断するために使用します。ログアウトするには、以下の構文を使用します。

http://path/spider.dll?logout

ここで、Path はサーバ名および SPIDER.DLL ファイルのパスです。SPIDER.DLL は HSPIDER ディレクトリにあります。

ログアウト関数の例

この例は、接続しているすべてのアプリケーションから切断するユーザの例を示します。この関数は、Microsoft IE ブラウザを終了する前に使用する必要があります。IE ブラウザを閉じて、ユーザはアプリケーションから自動的にログアウトされません。URL に次の行を入力します。

http://localhost/HSPIDER/spider.dll?logout

Hyperion Enterprise Reporting Administrator では、ユーザがブラウザを終了する前にクリックしてログアウト関数を呼び出す [ログアウト] ボタンを作成できます。

REPCHART 関数 - レポートとチャートの並列表示

REPCHART 関数は、左右に分割された表示領域を含む Web ページを生成するために使用します。それぞれのフレームには、個別のレポートまたはチャートが表示されます。

3つのレイアウトから選択できます。

- 左フレームのレポートおよび右フレームのチャート。
- 左フレームのチャートおよび右フレームのレポート。
- 両フレームのレポート。
- 両フレームのチャート。

REPCHART 関数の構文

並列表示のレポートとチャートを生成するには、次の構文を使用します。

```
http://Path/spider.dll?repchart&[sys=]System&[app=]Application  
&[rpt=]Report[&chartrpt=Report2][&tfile=Template.htx] [&tview=Template2.htx]  
[&tpov=Template3.htx][&Keyname=Keyvalue] [ChartOptions]
```

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	アクセスするシステムまたはオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPENG.INI ファイルの[DRIVERS32]セクションで定義します。 Essbase 構文について詳しくは、 10 ページの「Essbase の構文」 を参照してください。
Application	並列表示のレポートとチャートを生成する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。 System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。
Report	左フレームのチャートに表示するレポートのラベル。これは必須のパラメータです。 Application パラメータで指定されているアプリケーションの任意のレポートを選択できます。
Report2	右フレームのチャートに表示するレポートのラベル。RRFRAMES.HTX テンプレートを使用する場合、このパラメータは右フレームに表示するレポートを指定します。このパラメータを使用しない場合、Report パラメータで指定されているレポートが右フレームと左フレームに使用されます。
Template.htx	ページの生成に使用する HTML テンプレートファイル。デフォルトテンプレートは RCFRAMES.HTX で、左フレームにレポートを、右フレームにチャートを表示します。その他のテンプレートも使用できます。 <ul style="list-style-type: none">● CRFRAMES.HTX：左フレームにチャートを、右フレームにレポートを表示します。● RRFRAMES.HTX：両方のフレームに個別のレポートを表示します。● CCFRAMES.HTX：両方のフレームに個別のチャートを表示します。
Template2.htx	レポートに使用する HTML テンプレート。デフォルトは FREPORT.HTX です。
Template3.htx	レポートのデータの視点オプションに使用する HTML テンプレート。デフォルトは FPOV.HTX です。
Keyname	設定するデータの視点ディメンション。
Keyvalue	Keyname に割り当てる値。ピリオドはすべて、データの視点ディメンション値に表示されるとおりに入力してください。また、データの視点ディメンション値のスペースはプラス記号 (+) に置き換える必要があります。例えば、「JLY 97」の期間は、レポートリクエストで「JLY+97」になります。 注： 現在の月を指定するには、期間を@curmonth に設定します。SPIDER.INI ファイルの[Default]セクションで@curmonth パラメータを設定した場合、REPCHART 関数呼び出しの期間は、その月に設定されます。例えば、SPIDER.INI ファイルで

変数	説明
	@curmonth=january と設定し、ダイナミックリンクの REPCHART 関数呼び出しで period=@curmonth と設定した場合、表示される動的レポートのデータの視点は 1 月 (january) に設定されます。SPIDER.INI ファイルで @curmonth を設定しない場合、表示される動的レポートのデータの視点は、Windows NT の現在の月に設定されます。SPIDER.INI ファイルで @curmonth を設定しない場合、アプリケーションで使用する月の表記は、コンピュータが使用する長い形式の月の表記に正確に一致している必要があります。
ChartOptions	チャートオプション。使用可能なチャートオプションについては、 17 ページの「ChartType パラメータの値」 を参照してください。

REPCHART パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application、Report の順に指定する必要があります。
- 1 つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

REPCHART 関数の例

この例は、Hyperion Enterprise Reporting Web を使用してレポートとチャートを並列表示する方法を示します。

例 1：チャートとレポートの並列表示

この例では、REPCHART 関数を使用して、左フレームにレポート PLINE を、右フレームに PLINE のチャートを表示する Web ページを生成します。

```
<A HREF="http://spider/HSPIDER/spider.dll?
repchart&Enterprise&sales&PLINE">
```

次の例に示すように、tfile パラメータを使用して CRFRAMES.HTX テンプレートを選択することにより、左フレームにチャートを、右フレームにレポートを表示することもできます。

```
<A HREF="http://spider/HSPIDER/spider.dll
?repchart&Enterprise&sales&PLINE&tfile=
crframes.htx">
```

例 2 : レポートの並列表示

レポートを並列表示するには、tfile パラメータを使用して RRFRAMES.HTX テンプレートを選択し、CHARTRPT パラメータを使用して右フレームに表示するレポートを指定します。この例は、左フレームにレポート Revenue を、右フレームにレポート Costs を表示します。

```
<A HREF="http://spider/HSPIDER/spider.dll  
?repchart&Enterprise&sales&REVENUE&tfile=  
rrframes.htx&charttrpt=costs">
```

例 3 : チャートの並列表示

チャートを並列表示するには、tfile パラメータを使用して CCFRAMES.HTX テンプレートを選択し、CHARTRPT パラメータを使用して右フレームに表示するレポートを指定します。この例は、左フレームにレポート REVENUE を、右フレームにレポート COSTS を表示します。

```
<A HREF="http://spider/HSPIDER/spider.dll  
?repchart&Enterprise&sales&REVENUE&tfile=  
ccframes.htx&charttrpt=costs">
```

REPORT 関数 - 動的レポート

REPORT 関数は、SPIDER.DLL への関数呼び出しを行って、Hyperion Enterprise アプリケーションの任意のレポートを動的に表示する Web ページを生成するために使用します。Web ブラウザから URL として関数呼び出しを直接発行するか、HREF パラメータの値として URL を含むリンクを HTML 文書に作成できます。

REPORT 関数で指定できるオプションを次に示します。

- 個別のフレームまたはレポート本体にデータの視点ボタンを含める
- 別のテンプレートまたはユーザ定義のレポートテンプレートを使用する
- データの視点ディメンションを設定する
- Web ブラウザの種類を指定する (Microsoft Internet Explorer 6.0 または Mozilla Firefox 2.0.x のみ)
- レポートをスクロールするときに行と列の見出しを保護する (Microsoft Internet Explorer 6.0 以降または Mozilla Firefox 2.0.x 以降のみ)

注： 複合レポートの行と列の見出しを保護することはできません。

REPORT 関数の構文

レポートを動的に生成するには、次の構文を使用します。

```
http://Path/spider.dll?report&[sys=]System&[app=]Application  
&[rpt=]Report[&frame=UseFrames][&tfile=Template.htx] [&view=Template2.htx]  
[&tpov=Template3.htx][&Keyname=Keyvalue] [&drill=Investigations]
```

[&expand=Expansions][&browser=BrowserType] [&lockhdrs=LockHeaders]
 [&fcode=povbar][&webname = MyWebname]

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	<p>アクセスするシステムまたはオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPENG.INI ファイルの[DRIVERS32]セクションで定義します。</p> <p>Essbase 構文について詳しくは、10 ページの「Essbase の構文」を参照してください。</p>
Application	<p>レポートを生成する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。</p> <p>System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。</p>
Report	<p>実行するレポートのラベル。これは必須のパラメータです。</p> <p>Application パラメータで指定されているアプリケーションの任意のレポートを選択できます。</p>
UseFrames	<p>次のいずれかの値。</p> <ul style="list-style-type: none"> ● Yes : 個別のフレームにデータの視点ボタンが表示されます。これがデフォルト値です。 ● No : レポートの本体にデータの視点ボタンが表示されます。ブラウザがフレームをサポートしていない場合は、No を指定します。
Template.htx	レポートで使用するテンプレート（UseFrames が Yes に設定されている場合）。デフォルトは FRAMES.HTX です。
Template2.htx	レポートに使用する HTML テンプレート。このパラメータのデフォルト値については、 34 ページの「REPORT 関数 - Template2 パラメータのデフォルト値」 を参照してください。
Template3.htx	レポートのデータの視点オプションで使用する HTML テンプレート（UseFrames が Yes に設定されている場合）。デフォルトは FPOV.HTX です。
Keyname	設定するデータの視点ディメンション。
Keyvalue	<p>Keyname に割り当てる値。ピリオドはすべて、データの視点ディメンション値に表示されるとおりに入力してください。また、データの視点ディメンション値のスペースはプラス記号 (+) に置き換える必要があります。例えば、「JLY 97」の期間は、レポートリクエストで「JLY+97」になります。</p> <p>注： 現在の月を指定するには、期間を@curmonth に設定します。SPIDER.INI ファイルの[Default]セクションで@curmonth パラメータを設定した場合、REPORT 関数呼び出しの期間は、その月に設定されます。例えば、SPIDER.INI ファイルで@curmonth=january と設定し、ダイナミックリンクの REPORT 関数呼び出しでperiod=@curmonth と設定した場合、表示される動的レポートのデータの視点は 1 月 (january) に設定されます。SPIDER.INI ファイルで@curmonth を設定しない場合、表示される動的レポートのデータの視点は、Windows NT の現在の月に設定されます。SPIDER.INI ファイルで@curmonth を設定しない場合、アプリケーションで使用する月の表記は、コンピュータが使用する長い形式の月の表記に正確に一致する必要があります。</p>

変数	説明
Investigations	<p>次のいずれかの値。</p> <ul style="list-style-type: none"> ● Yes : レポートで調査が有効になります。この設定がデフォルトです。 ● No : レポートで調査が無効になります。このパラメータを No に設定した場合、レポート定義で調査が有効になっていても調査は無効になります。
Expansions	<p>次のいずれかの値。</p> <ul style="list-style-type: none"> ● Yes : レポートで展開が有効になります。この設定がデフォルトです。 ● No : レポートで展開が無効になります。このパラメータを No に設定した場合、レポート定義で展開が有効になっていても展開は無効になります。No を指定すると、プラス記号 (+) やマイナス記号 (-) が表示されないの でブラウザで使用する領域が小さくなります。
BrowserType	<p>次のいずれかの値。</p> <ul style="list-style-type: none"> ● IE40 : Microsoft Internet Explorer 6.0.またはそれ以降を Web ブラウザとして指定します。 ● Netscape40 : Mozilla Firefox 2.0.x 以降を Web ブラウザに指定します。 ● 任意の値 : 別のブラウザを指定します。例えば、古いブラウザを使用する場合は、OTHER を指定します。 <p>BrowserType パラメータを使用すると、管理プログラムのデフォルトブラウザの設定を上書きできます。このオプションの設定については、Administration プログラムのオンラインヘルプを参照してください。</p>
LockHeaders	<p>次のいずれかの値。</p> <ul style="list-style-type: none"> ● Yes : ユーザがレポートをスクロールする際に行と列の見出しが保護されます。 ● No : 行と列の見出しは保護されません。 <p>LockHeaders パラメータを使用すると、管理プログラムの行および列見出しのデフォルトの動作を上書きできます。このオプションの設定方法について詳しくは、Administration プログラムのオンラインヘルプを参照してください。</p>
view/pov	<p>View は、POV バーなしでレポートを生成する場合に使用します。POV は、データの視点バーのみを生成する場合に使用します。この場合、フレームは使用されません。</p> <p>注 : View および POV はオプションの値です。</p>
MyWebname	<p>FrontPage Web が保存されているディレクトリの名前（スペースは除外します）。Web サイトウィザードによって FrontPage Web が生成されるときに、FrontPage Web で使用される REPORT 関数に必要なテンプレート（.HTX ファイル）がコピーされます。生成された FrontPage Web でレポートへのリンクを選択すると、MyWebname パラメータによって正しいテンプレートの場所が識別されます。</p> <p>このパラメータは、主に FrontPage Web サイトで使用されます。</p>

REPORT パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application、Report の順に指定する必要があります。

- 1つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

REPORT 関数 - Template2 パラメータのデフォルト値

次の表に REPORT 関数の Template2.htx パラメータのデフォルト値を一覧表示します。デフォルト値は、フレームを使用するかどうか、または行と列の見出しを保護するかどうかは、使用するブラウザによって異なります。

表 6 REPORT 関数の Template2 パラメータのデフォルト値

	すべてのブラウザ、見出しの保護なし	Microsoft Internet Explorer 6.0.x またはそれ以降、見出しの保護あり	Mozilla Firefox 2.0.x 以降、見出しの保護あり
フレームなし	REPORT.HTX	JSREPIE.HTX	JSREPNS.HTX
フレームあり	FREPORT.HTX	FJSREPIE.HTX	FJSREPNS.HTX

REPORT 関数の例

この例は、Hyperion Enterprise Reporting Web を使用して、ユーザ定義のテンプレートおよびさまざまなデータの視点ディメンションを指定して、フレームのあるレポートとフレームのないレポートを表示する方法を示します。

例 1：レポートの実行

REPORT 関数を使用して Web ページにレポートを表示できます。この例は、Hyperion Enterprise DEMO アプリケーションの RPPL1 レポートのレポートを生成します。

```
<A HREF="http://spider/HSPIDER/spider.dll
?report&Enterprise&Demo&RPPL1">
```

例 2：フレームのないレポートの実行

フレームを指定するパラメータを使用して、フレームをサポートしないブラウザで表示可能なレポートを作成します。レポートの本体にすべてのデータの視点ボタンが表示されます。次にその例を示します。

```
<A HREF="http://spider/HSPIDER/spider.dll
?report&Enterprise&Demo&RPPL1&frame=no">
```

例 3 : ユーザ定義テンプレートの使用

tview パラメータを使用して、レポートの作成に使用するユーザ定義のテンプレートを指定できます。この例は、ユーザ定義テンプレートの MYTEMP.HTX を使用してレポートを作成します。

```
<A HREF="http://spider/HSPIDER/spider.dll  
?report&Enterprise&Demo&RPPL1&tview=mytemp.htx">
```

例 4 : データの視点ディメンションのリセット

任意のデータの視点ディメンションのデフォルトのデータの視点を上書きできます。レポートのデータの視点を変更すると、新しいデータの視点が保持され、そのレポートにアクセスするたびに、そのデータの視点が使用されます。この例は、レポートのエンティティ、データ種別および期間のデータの視点ディメンションをリセットします。

```
<A HREF="http://spider/HSPIDER/spider.dll  
?report&Enterprise&Demo&RPPL1&entity=uk.total  
&category=lastyr&period=jly+97">
```

PRINTREP 関数 - PDF ファイルへのレポート出力

PRINTREP 関数を使用して、Adobe Acrobat PDF ファイルにレポートを出力できます。ブラウザの出力機能を使用してレポートを出力する場合は、Hyperion Enterprise Reporting で設定されているレポート書式の一部がサポートされません。PRINTREP 関数を使用すると、Hyperion Enterprise Reporting レポートエンジンを使用して、レポートが PDF ファイルに出力されます。この PDF ファイルは、ユーザのクライアントマシンに保存されます。この PDF ファイルは Adobe Acrobat Reader で表示され、ユーザは、このファイルを印刷できます。PDF ファイルに出力する場合は、Hyperion Enterprise Reporting の書式設定がすべてサポートされます。

注 : この機能を使用して出力するには、サーバに Adobe Acrobat PDF Writer がインストールされていること、およびユーザのクライアントマシンに Adobe Acrobat Reader がインストールされていることが必要です。Adobe Acrobat Reader のダウンロードについては、次の Adobe の Web サイト (<http://www.adobe.com/prodindex/acrobat/readstep.html>) を参照してください。Adobe Acrobat Reader は、Hyperion Enterprise Reporting Web の CD-ROM からインストールできます。

PRINTREP 関数の構文

PDF ファイルに出力するには、次の構文を使用します。

```
http://Path/spider.dll?printrep&[sys=]System&[app=]Application&[rpt=]Report
```

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	アクセスするシステムまたはオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPENG.INI ファイルの[DRIVERS32]セクションで定義します。 Essbase 構文について詳しくは、10 ページの「Essbase の構文」を参照してください。
Application	レポートを生成する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。 System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。
Report	実行するレポートのラベル。これは必須のパラメータです。 Application パラメータで指定されているアプリケーションの任意のレポートを選択できます。

PRINTREP パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。

これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application、Report の順で指定する必要があります。
- 1 つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。

PRINTREP 関数の例

この例は、PRINTREP 関数を使用して、ユーザが Hyperion Enterprise SALES アプリケーションの REVENUE および COSTS レポートを.PDF ファイルに出力できる Web ページを示します。ユーザがリンクを選択すると、レポート.PDF ファイルが Adobe Acrobat Reader で開きます。

```
<H2>Salesアプリケーションのレポート</H2><P>
<B>Adobe Acrobat Readerで表示するレポートを選択してください。</B><BR>
<A HREF="/HSPIDER/spider.dll
?printrep&enterprise&sales&REVENUE">
RevenueレポートのPDFファイル</A>
<A HREF="/HSPIDER/spider.dll
?printrep&enterprise&sales&COSTS">
CostsレポートのPDFファイル</A>
```

SCHEPREP 関数 - Hyperion Distributed Schedules の起動

SCHEPREP 関数は、Hyperion Distributed Schedules を起動する URL リンクを提供するために使用します。SCHEPREP 関数は、SCHEXT.DLL への関数呼び出しを行って Hyperion Enterprise アプリケーションのデータ入力レポートを表示します。SCHEXT.DLL は、Hyperion Distributed Applications サーバの一部として HSPIDER ディレクトリにインストールされます。Web ブラウザから URL として関数呼び出しを直接発行するか、HREF パラメータの値として URL を含むリンクを HTML 文書に作成できます。

注： この関数は、Hyperion Distributed Applications がクライアントのサーバにインストールされている場合にのみ使用できます。

SCHEPREP 関数で Keyname パラメータを使用して、データの視点の設定を指定することもできます。また、オプションのサーバ名を指定して、クライアントが別の Distributed Applications サーバに接続するように設定できます。

SCHEPREP 関数の構文

Hyperion Distributed Schedules を起動するには、次の構文を使用します。

```
http://Path/schext.dll?schedrep&[sys=]System&[appset=]  
ApplicationSet&[app=]Application&[rpt=]Report[&Keyname=Keyvalue]  
[&server=ServerName]
```

変数	説明
Path	サーバ名と SCHEXT.DLL ファイルのパス。SCHEXT.DLL は HSPIDER ディレクトリにあります。
System	アクセスするシステムまたはオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPEING.INI ファイルの[DRIVERS32]セクションで定義します。 Essbase 構文について詳しくは、 10 ページの「Essbase の構文」 を参照してください。
ApplicationSet	レポートを生成する Hyperion Enterprise アプリケーションセットのラベル。これは必須のパラメータです。 System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアを使用する場合は、表示、入力、または編集するデータのあるアプリケーションが含まれたセットを選択します。指定したオラクル社の Enterprise Performance Management ソフトウェアがアプリケーションセットを使用しない場合は、NONE を指定します。
Application	レポートを生成する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。 System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアと、Appset パラメータで指定されているアプリケーションセットのアプリケーションを選択できます。
Report	実行するレポートのラベル。これは必須のパラメータです。

変数	説明
	Application パラメータで指定されているアプリケーションの任意のレポートを選択できます。
Keyname	設定するデータの視点ディメンション。
Keyvalue	Keyname に割り当てる値。ピリオドはすべて、データの視点ディメンション値に表示されるとおりに入力してください。また、データの視点ディメンション値のスペースはプラス記号 (+) に置き換える必要があります。例えば、「JLY 97」の期間は、レポートリクエストで「JLY+97」になります。
ServerName	Distributed Applications サーバのサーバ名。Hyperion Distributed Schedules クライアントが接続するサーバは、Path パラメータに表示されているサーバとは異なる場合があります。ServerName パラメータを省略すると、Hyperion Distributed Schedules クライアントは、Path パラメータで指定されているサーバに接続します。

SCHEDREP パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、ApplicationSet、Application、Report の順で指定する必要があります。
- 1 つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

SCHEDREP 関数の例

例 1

この例では、Hyperion Enterprise レポート RPPL1 を含む Hyperion Distributed Schedules を起動します。

```
<A HREF="http://spider/HSPIDER/schext.dll
?SchedRep&enterprise&none&demo&rppl1">
```

例 2

この例では、Hyperion Enterprise レポート RPPL1 を含む Hyperion Distributed Schedules を起動し、データ種別ディメンションを BUDGET に、勘定科目ディメンションを COSTGDS に設定します。

```
<A HREF="http://spider/HSPIDER/schext.dll
?SchedRep&enterprise&none&demo&rppl1
&category=budget&account=costgds">
```

ライブラリ関数

Hyperion Enterprise Reporting Web ライブラリ関数を使用すると、アプリケーションの一覧、レポートセットおよびレポートをオラクル社の Enterprise Performance Management ソフトウェアから動的に生成できます。次の表でライブラリ関数について説明します。

表 7 ライブラリ関数

関数	用途
SYSLIST	Web サーバ上のオラクル社の Enterprise Performance Management の一覧を生成します。
APPLIST	指定した製品のアプリケーションの一覧を生成します。
SETLIST	指定した製品とアプリケーションのレポートセットの一覧を生成します。
REPSETLIST	指定した製品、アプリケーション、およびレポートセット内のレポートの一覧を生成します。

SYSLIST（製品情報）

SYSLIST 関数は、Windows ディレクトリ内の REPENG.INI ファイルで指定されているように、Web サーバにインストールされ Hyperion Enterprise Reporting からアクセスできるオラクル社の Enterprise Performance Management ソフトウェアをすべて一覧にする Web ページを生成します。

製品一覧内の各製品はハイパーリンクとして表示されます。ユーザがページから製品へのリンクを選択すると、選択した製品のアプリケーションを一覧表示する Web ページが動的に生成されます。ユーザはアプリケーション一覧から、アプリケーションのレポートセットとレポートを一覧にするページへと、さらにドリルダウンできます。レポート一覧ページからは動的レポートを実行できます。

Hyperion Enterprise Reporting Web では、テンプレートファイルを使用して、SYSLIST で生成された Web ページの外観を定義します。SYSLIST で使用されるテンプレートファイルは SYSLIST.HTX です。テンプレートを変更するか、独自のカスタムテンプレートを作成できます。テンプレートについては、『管理者用リファレンスガイド』の「テンプレートを使用した Web ページの作成」の章を参照してください。

SYSLIST 関数の構文

SYSLIST 関数には以下の構文を使用します。

```
http://Path/spider.dll?syslist[&frame=UseFrames][&file=Template1]
[&htxapplist=Template2][&htxsetlist=Template3][&htxrList=Template4]
[&htxrep=Template5][&bitmap=UseBitmaps][&webname=MyWebname]
```

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
UseFrames	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : 個別のフレームにデータの視点ボタンが表示されます。これがデフォルト値です。 ● No : レポートの本体にデータの視点ボタンが表示されます。ブラウザがフレームをサポートしていない場合は、No を指定します。
Template1	システム一覧 Web ページに使用する HTML テンプレート。デフォルトは SYSLIST.HTX です。
Template2	アプリケーション一覧 Web ページに使用する HTML テンプレート。デフォルトは APPLIST.HTX です。
Template3	レポートセッター一覧 Web ページに使用する HTML テンプレート。デフォルトは SETLIST.HTX です。
Template4	レポート一覧 Web ページに使用する HTML テンプレート。デフォルトは REPLIST.HTX です。
Template5	ユーザがレポート一覧ページからレポートを選択したときに動的レポートの生成に使用する HTML テンプレート。デフォルトは FRAMES.HTX です。
UseBitmaps	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : レポート一覧のレポート名の横にレポートアイコンが表示されます。これがデフォルト値です。 ● No : レポートアイコンは表示されません。
MyWebname	FrontPage Web が保存されているディレクトリの名前（スペースは除外します）。Web サイトウィザードによって FrontPage Web が生成されるときは、FrontPage Web で使用される SYSLIST 関数に必要なテンプレート（.HTX ファイル）がコピーされます。生成された FrontPage Web でレポートへのリンクを選択すると、MyWebname パラメータによって正しいテンプレートの場所が識別されます。 このパラメータは、主に FrontPage Web サイトで使用されます。

SYSLIST パラメータを使用する際のルール

この関数には、オプションのパラメータが含まれます。これらのパラメータを使用する場合は、次のルールに従ってください。

- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

SYSLIST 関数の例

例 1 : 製品一覧の生成

この例では、Hyperion Enterprise Reporting サーバ上の製品一覧を生成します。

```
<A HREF="http://spider/HSPIDER/spider.dll?syslist">
```

例 2 : SYSLIST 関数でのユーザ定義テンプレートの使用

この例では、システム一覧ページのユーザ定義テンプレートとして SYS.HTX、アプリケーション一覧ページのユーザ定義テンプレートとして APPS.HTX を指定します。

```
<A HREF="http://spider/HSPIDER/spider.dll?syslist  
&sys=enterprise&tfile=sys.htx&htxapplist=apps.htx">
```

APPLIST（アプリケーション一覧）

APPLIST 関数を使用すると、指定したシステムのすべてのアプリケーションを一覧表示する Web ページが生成されます。アプリケーション一覧ページの各アプリケーションはハイパーテキストリンクとして表示されます。リンクを選択すると、APPLIST 関数の最後のレベルに達するまで、次のレベルの情報が表示されます。

Essbase を使用している場合は、指定した製品に使用できるサーバの一覧が APPLIST 関数によって生成されます。サーバを選択すると、ログオンプロンプトが表示され、そのサーバで使用可能なアプリケーションの一覧がブラウザに表示されます。アプリケーションを選択すると、使用可能なデータベースの一覧が生成されます。

APPLIST 関数を使用して Essbase で初めてサーバにアクセスするとき、有効なユーザ名とパスワードを入力する必要があります。

Hyperion Enterprise Reporting Web では、HTML テンプレートファイルを使用して、APPLIST で生成された Web ページの外観を定義します。APPLIST で使用されるテンプレートファイルは APPLIST.HTX です。テンプレートを変更するか、独自のカスタムテンプレートを作成できます。詳しくは、『管理者用リファレンスガイド』の「テンプレートを使用した Web ページの作成」の章を参照してください。

APPLIST 関数の構文

APPLIST 関数には以下の構文を使用します。

```
http://Path/spider.dll?applist&[sys=]System[&frame=UseFrames] [&tfile=Template1]  
[&htxsetlist=Template2][&htxrslst=Template3] [&htxrep=Template4]  
[&bitmap=UseBitmaps][&webname=MyWebname]
```

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	<p>アプリケーション一覧を生成するオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。</p> <p>Essbase を使用している場合は、次の構文を使用して、サーバ、アプリケーションまたはデータベースの一覧を取得できます。</p> <ul style="list-style-type: none">● サーバの一覧を取得するには、REPENG.INI ファイルの Essbase ドライバの名前を使用します。例えば、&SYS=Essbase と指定します。● アプリケーションの一覧を取得するには、REPENG.INI ファイルの Essbase ドライバの名前、および Essbase サーバの名前をハイフンで区切って指定します。例えば、&SYS=Essbase-ServerName と指定します。

変数	説明
	データベースの一覧を取得するには、REPEING.INI ファイルの Essbase ドライバの名前、および Essbase サーバの名前をハイフンで区切って指定し、その後に円記号 (\) および Essbase アプリケーションの名前を指定します。例えば、 &SYS=Essbase-ServerName\ApplicationName と指定します。
UseFrames	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : 個別のフレームにデータの視点ボタンが表示されます。これがデフォルト値です。 ● No : レポートの本体にデータの視点ボタンが表示されます。ブラウザがフレームをサポートしていない場合は、No を指定します。
Template1	アプリケーション一覧 Web ページに使用する HTML テンプレート。デフォルトは APPLIST.HTX です。
Template2	レポートセッター一覧 Web ページに使用する HTML テンプレート。デフォルトは SETLIST.HTX です。
Template3	レポート一覧 Web ページに使用する HTML テンプレート。デフォルトは REPLIST.HTX です。
Template4	ユーザがレポート一覧ページからレポートを選択したときに動的レポートの生成に使用する HTML テンプレート。デフォルトは FRAMES.HTX です。
UseBitmaps	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : レポート一覧のレポート名の横にレポートアイコンが表示されます。これがデフォルト値です。 ● No : レポートアイコンは表示されません。
MyWebname	FrontPage Web が保存されているディレクトリの名前（スペースは除外します）。Web サイトウィザードによって FrontPage Web が生成されるときは、FrontPage Web で使用される APPLIST 関数に必要なテンプレート（.HTX ファイル）がコピーされます。生成された FrontPage Web でレポートへのリンクを選択すると、MyWebname パラメータによって正しいテンプレートの場所が識別されます。 このパラメータは、主に FrontPage Web サイトで使用されます。

APPLIST パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

APPLIST 関数の例

この例では、指定した製品のアプリケーション一覧を動的に生成する方法を示します。

例 1 : Hyperion Enterprise のアプリケーション一覧の生成

この例は、Hyperion Enterprise Reporting サーバ上の Hyperion Enterprise のアプリケーション一覧を生成します。

```
<A HREF="http://spider/HSPIDER/spider.dll?applist&sys=enterprise">
```

例 2 : APPLIST 関数でのユーザ定義テンプレートの使用

この例では、アプリケーション一覧ページのユーザ定義テンプレートとして APPS.HTX を、レポート一覧ページのユーザ定義テンプレートとして REPS.HTX を指定します。

```
<A HREF="http://spider/HSPIDER/spider.dll?applist&sys=enterprise  
&tfile=apps.htx&htxrslst=reps.htx">
```

SETLIST（レポートセotto一覧）

SETLIST 関数は、指定した製品とアプリケーションのレポートセットをすべて一覧にする Web ページを生成します。レポートセotto一覧ページのレポートの各レポートセottoはハイパーテキストリンクとして表示されます。ユーザがこのページからレポートセottoへのリンクを選択すると、そのレポートセotto内のレポートを一覧表示する Web ページが動的に生成されます。ユーザはレポートセotto一覧ページから動的レポートを実行できます。

Hyperion Enterprise Reporting Web では、HTML テンプレートファイルを使用して、SETLIST で生成された Web ページの外観を定義します。SETLIST で使用されるテンプレートファイルは SETLIST.HTX です。テンプレートを変更するか、独自のカスタムテンプレートを作成できます。詳しくは、『管理者用リファレンスガイド』の「テンプレートを使用した Web ページの作成」の章を参照してください。

SETLIST 関数の構文

SETLIST 関数には以下の構文を使用します。

```
http://Path/spider.dll?setlist&[sys=]System&[app=]Application [&frame=UseFrames]  
[&tfile=Template1][&htxrslst=Template2] [&htxrep=Template3]  
[&bitmap=UseBitmaps][&webname=MyWebname]
```

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	アプリケーション一覧を生成するオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPENG.INI ファイルの[DRIVERS32]セクションで定義します。 Essbase 構文について詳しくは、 10 ページの「Essbase の構文」 を参照してください。
Application	レポートを生成する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。

	System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。
UseFrames	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : 個別のフレームにデータの視点ボタンが表示されます。これがデフォルト値です。 ● No : レポートの本体にデータの視点ボタンが表示されます。ブラウザがフレームをサポートしていない場合は、No を指定します。
Template1	レポートセotto一覧 Web ページに使用する HTML テンプレート。デフォルトは SETLIST.HTX です。
Template2	レポート一覧 Web ページに使用する HTML テンプレート。デフォルトは REPLIST.HTX です。
Template3	ユーザがレポート一覧ページからレポートを選択したときにレポートの動的な生成に使用する HTML テンプレート。デフォルトは FRAMES.HTX です。
UseBitmaps	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : レポート一覧のレポート名の横にレポートアイコンが表示されます。これがデフォルト値です。 ● No : レポートアイコンは表示されません。
MyWebname	FrontPage Web が保存されているディレクトリの名前（スペースは除外します）。Web サイトウィザードによって FrontPage Web が生成されるときは、FrontPage Web で使用される SETLIST 関数に必要なテンプレート（.HTX ファイル）がコピーされます。生成された FrontPage Web でレポートへのリンクを選択すると、MyWebname パラメータによって正しいテンプレートの場所が識別されます。 このパラメータは、主に FrontPage Web サイトで使用されます。

SETLIST パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application の順で指定する必要があります。
- 1 つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

SETLIST 関数の例

この例は、指定したアプリケーションのレポートセotto一覧を動的に生成する方法を示します。

Hyperion Enterprise DEMO アプリケーションのレポートセッター一覧の生成

この例では、Hyperion Enterprise Reporting サーバ上の Hyperion Enterprise DEMO アプリケーションのレポートセッター一覧を生成します。

```
<A HREF="http://spider/HSPIDER/spider.dll?
setlist&sys=enterprise&app=demo">
```

例 2 : SETLIST 関数でのユーザ定義テンプレートの使用

この例は、セッター一覧ページのユーザ定義テンプレートとして SETS.HTX を、レポート一覧ページのユーザ定義テンプレートとして REPS.HTX を指定します。

```
<A HREF="http://spider/HSPIDER/spider.dll?
setlist&sys=enterprise&app=demo&tfile=sets.htx&htxrslst=reps.htx">
```

REPSETLIST（レポート一覧）

REPSETLIST 関数は、指定した製品、アプリケーション、およびレポートセット内のレポートをすべて一覧にする Web ページを生成します。レポート一覧ページの各レポートはハイパーテキストリンクとして表示されます。ユーザがこのページからレポートへのリンクを選択すると、そのレポートを表示する Web ページが動的に生成されます。

Hyperion Enterprise Reporting Web では、HTML テンプレートファイルを使用して、REPSETLIST で生成された Web ページの外観を定義します。REPLIST で使用されるテンプレートファイルは REPLIST.HTX です。テンプレートを変更するか、独自のカスタムテンプレートを作成できます。詳しくは、『管理者用リファレンスガイド』の「テンプレートを使用した Web ページの作成」の章を参照してください。

REPSETLIST 関数の構文

REPSETLIST 関数には以下の構文を使用します。

```
http://Path/spider.dll?repsetlist&[sys=]System&[app=]Application
&[set=]Set[&frame=UseFrames][&tfile=Template1][&htxrslst=Template2]
[&bitmap=UseBitmaps][&webname=MyWebname]
```

変数	説明
Path	サーバ名と SPIDER.DLL ファイルのパス。SPIDER.DLL は HSPIDER ディレクトリにあります。
System	アプリケーション一覧を生成するオラクル社の Enterprise Performance Management ソフトウェアのラベル。これは必須のパラメータです。システムラベルは、REPENG.INI ファイルの[DRIVERS32]セクションで定義します。 Essbase 構文について詳しくは、 10 ページの「Essbase の構文」 を参照してください。
Application	レポートを生成する Hyperion Enterprise アプリケーションのラベル。これは必須のパラメータです。

	System パラメータで指定されているオラクル社の Enterprise Performance Management ソフトウェアの任意のアプリケーションを選択できます。
Set	レポーター一覧を生成するレポートセットのラベル。これは必須のパラメータです。 Application パラメータで指定されているアプリケーションの任意のレポートセットを選択できます。
UseFrames	次のいずれかの値。 <ul style="list-style-type: none"> ● Yes : 個別のフレームにデータの視点ボタンが表示されます。これがデフォルト値です。 ● No : レポートの本体にデータの視点ボタンが表示されます。ブラウザがフレームをサポートしていない場合は、No を指定します。
Template1	レポーター一覧 Web ページに使用する HTML テンプレート。デフォルトは REPLIST.HTX です。
Template2	ユーザがレポーター一覧ページからレポートを選択したときにレポートの動的な生成に使用する HTML テンプレート。デフォルトは FRAMES.HTX です。
UseBitmaps	レポーター一覧 Web ページに使用する HTML テンプレート。デフォルトは FRAMES.HTX です。
MyWebname	FrontPage Web が保存されているディレクトリの名前（スペースは除外します）。Web サイトウィザードによって FrontPage Web が生成されるときは、FrontPage Web で使用される REPSETLIST 関数に必要なテンプレート（.HTX ファイル）がコピーされます。生成された FrontPage Web でレポートへのリンクを選択すると、MyWebname パラメータによって正しいテンプレートの場所が識別されます。 このパラメータは、主に FrontPage Web サイトで使用されます。

REPSETLIST パラメータを使用する際のルール

この関数には、必須とオプションの両方のパラメータが含まれます。必須パラメータについては、前述の表を参照してください。これらのパラメータを使用する場合は、次のルールに従ってください。

- 必須パラメータは、System、Application、Set の順に指定する必要があります。
- 1 つの必須パラメータでラベルを使用する場合、すべての必須パラメータでラベルを使用する必要があります。
- オプションパラメータは、必須パラメータの後に配置する必要があります。
- オプションパラメータは任意の順序で使用できます。
- それぞれのオプションパラメータでは、ラベルを使用する必要があります。

REPSETLIST 関数の例

この例は、指定したレポートセットのレポートセット一覧を動的に生成する方法を示します。

例 1 : レポート一覧の生成

この例は、Hyperion Enterprise Reporting サーバ上の Hyperion Enterprise DEMO アプリケーションの Income レポートセットのレポートの一覧を生成します。

```
<A HREF="http://spider/HSPIDER/spider.dll?setlist  
&sys=enterprise&app=demo&set=income">
```

例 2 : REPSETLIST 関数でのユーザ定義テンプレートの使用

この例は、レポート一覧ページのユーザ定義テンプレートとして REPS.HTX を、動的レポートのユーザ定義テンプレートとして CUSTOM.HTX を指定します。

```
<A HREF="http://spider/HSPIDER/spider.dll?setlist&sys=enterprise  
&app=demo&set=income&tfile=sets.htx&htxrslst=reps.htx">
```


この章の内容

レポートオブジェクト - 概要	49
エージェントオブジェクト	58
エージェント評価オブジェクト - 概要	63
テンプレートオブジェクト	67
条件オブジェクト	68
アクション情報オブジェクト	73

Hyperion Enterprise Reporting Web Administration プログラムを使って条件スクリプトやエージェント条件を作成するときには、OLE オートメーションオブジェクトを参照します。OLE オートメーションオブジェクトはブラウザに表示するページをシステムが作成するときに使用されます。条件スクリプトが定義されているページやエージェント条件が実行されたページにユーザがアクセスするたびに、システムは VBScript エンジンで次のオブジェクトを登録します。

- レポートオブジェクト
- エージェントオブジェクト
- エージェント評価オブジェクト
- テンプレートオブジェクト
- 条件オブジェクト
- アクション情報オブジェクト

各オブジェクトには関連メソッドがあり、プロパティを持つものもあります。以下の項では、各メソッドとプロパティの概略を説明します。

レポートオブジェクト - 概要

レポートオブジェクトは、Hyperion Enterprise Reporting からアクセスできるオラクル社の Enterprise Performance Management ソフトウェアのデータです。レポートオブジェクトは Hyperion Enterprise Reporting Web によって次の 2 つの場合に登録されます。

- ユーザが条件ベースのページにアクセスすると、システムはその条件スクリプトで使用されているレポートを登録します。
- システムが財務エージェントを実行するときに、そのエージェントに関連付けられている個々のレポートが登録されます。

レポートの登録にはレポート名が使用され、その名前を呼び出すことによってアクセスできます。登録されたレポートに次のメソッドを適用できます。

表 8 レポートオブジェクトのメソッド

メソッド	戻り値
GetAppName	参照先のレポートを作成した Hyperion Enterprise アプリケーションの名前。
GetCellDouble	レポート内の行と列に関連付けられている、書式設定されていない数値。
GetCellString	レポート内の特定の行と列の値の文字列表現。
GetCellStringHTML	レポート内の特定の行と列の HTML 形式に変換された値の文字列表現。
GetCellPovString	レポート内の特定セルのデータの視点。
GetCellStatus	レポート内の特定セルの値の元になっているデータの状態に関する情報。
GetName	レポート名。
GetNumRows	指定したレポートオブジェクト内の行数。
GetNumColumns	指定したレポートオブジェクト内の列数。
GetSystemName	参照されているレポートを作成したオラクル社の Enterprise Performance Management ソフトウェアの名前。

上級 OLE ユーザ用の情報

レポートオブジェクトを参照するときに次の情報を使用します。

OLE ProgID : RHYPEDA.HYPREPORT

OLE GUID : {185FB2E1-72AC-11CF-809D-0800091AE62F}

GetAppName メソッド

GetAppName メソッドは、条件スクリプトやエージェント条件内で参照されているレポートを作成した Hyperion Enterprise アプリケーションの名前を取得するために使用します。このメソッドにパラメータはありません。アプリケーションの名前はテキスト文字列として戻ります。

構文 :

object.GetAppName

但し、object は参照先のレポートオブジェクトの名前です。

戻り値 : STRING - Hyperion Enterprise アプリケーションの名前

適用対象 : レポートオブジェクト

例 1 :

レポート RPPL1 は、Hyperion Enterprise 製品の DEMO アプリケーションによって作成されました。この情報を取得するには、次のように入力します。

```
RPPL1.GetAppName
```

システムは、「DEMO」という文字列を戻します。

例 2 :

次の例は、カスタム OLE オブジェクト内でこのメソッドを使用する方法を示します。

```
Dim AppName
AppName = RPPL1.GetAppName
```

GetCellDouble メソッド

GetCellDouble メソッドは、レポート内の行と列で定義されるセル内の数値を取得するために、条件スクリプトやエージェント条件内で使用します。GetCellString とは異なり、GetCellDouble メソッドは、書式設定されていない数値を返します。

構文 :

```
object.GetCellDouble(lRow as Long, lCol as Long)
```

変数	説明
object	参照先のレポートオブジェクトの名前。
lRow as Long	値を取得する行の番号。行番号はゼロから始まります。レポートの左上の最初の行の番号がゼロになります。
lCol as Long	値を取得する列の番号。列番号はゼロから始まります。レポートの左上の最初の列の番号がゼロになります。

戻り値 : DOUBLE - レポートの数値

適用対象 : レポートオブジェクト

例 1 :

Hyperion Enterprise の RPPL1 レポートには、一連の勘定科目の予算と実際の数値の比較が含まれています。この例では、「Operating Expenses (経費)」行の「Actual (実際)」列から数値を取得します。「Operating Expenses」データは行 22、「Actual」データは列 1 にあるので、次のように入力します。

```
RPPL1.GetCellDouble(22,1)
```

「3156」が指定されたセルの値として戻ります。

例 2 :

次の例は、カスタム OLE オブジェクト内でこのメソッドを使用する方法を示します。

```
Dim CellValue
CellValue = RPPL1.GetCellDouble(22,2)
```

GetCellPovString メソッド

GetCellPovString メソッドは、条件スクリプトやエージェント条件内で、レポート内の行と列で定義されるセルのデータの視点を取得するために使用します。GetCellPovString メソッドは STRING 値を返します。

構文 :

```
object.GetCellPovString(lRow as Long, lCol as Long)
```

変数	説明
object	参照先のレポートオブジェクトの名前。
lRow as Long	データの視点を取得する行の番号。行番号はゼロから始まります。レポートの左上の最初の行の番号がゼロになります。
lCol as Long	データの視点を取得する列の番号。列番号はゼロから始まります。レポートの左上の最初の列の番号がゼロになります。

戻り値 : STRING - レポートのセルのデータの視点的値

適用対象 : レポートオブジェクト

例 1 :

レポート RPPL1 の最初の行と列のデータ視点は現在、エンティティが「USA」、データ種別が「ACTUAL」、期間が「97 年 1 月」、期間単位が「月次」、勘定科目が「GROSSSALES」に設定されています。レポートのデータ視点を取得するために、この例では、次のように入力します。

```
RPPL1.GetCellPovString(1,1)
```

次のような文字列が返されます。

```
&エンティティ=USA&データ種別=ACTUAL&期間=97 年 1 月&期間単位=M.CTD&  
勘定科目=GRSALES&サブエンティティ=なし&サブ勘定科目 1=なし&サブ勘定科  
目 2=なし&フィールド=VAL
```

例 2 :

次の例は、カスタム OLE オブジェクト内でこのメソッドを使用する方法を示します。

```
Dim CellPOV
CellPOV = RPPL1.GetCellPovString(22,2)
```

GetCellString メソッド

GetCellString メソッドは、レポート内の行と列で定義されるセル内の値を文字列として取得するために、条件スクリプト内で使用します。戻される文字列にはそのレポートに定義されている書式が適用されます。例えば、そのレポートでドル記号 (\$) や負の値をカッコに入れる書式が使用されている場合、戻される文字列でも、ドル記号 (\$) や負の値をカッコに入れる書式が使用されます。このメソッドで行または列の番号にゼロを指定すると、行や列の見出しが文字列として戻ります。

構文：

object.GetCellString(lRow as Long, lCol as Long)

変数	説明
object	参照先のレポートオブジェクトの名前。
lRow as Long	値を取得する行の番号。行番号はゼロから始まります。レポートの左上の最初の行の番号がゼロになります。
lCol as Long	値を取得する列の番号。列番号はゼロから始まります。レポートの左上の最初の列の番号がゼロになります。

戻り値：STRING - 行と列で指定したセルの内容を文字列で表記したもの。

適用対象： レポートオブジェクト

例 1：

Hyperion Enterprise の RPPL1 レポートには、一連の勘定科目の予算と実際の数値の比較が含まれています。この例では、「Operating Expenses (経費)」行の「Variance (差)」列にある数値を取得します。「Operating Expenses」は行 22 に、「Variance」は列 3 にあるので、次のように入力します。

```
RPPL1.GetCellString(22,3)
```

経費は予算を\$500 下回っているなので、差が-\$500 になっています。そのレポートでは負の値をカッコに入れる書式が使用されているので、書式を適用した文字列「(\$500)」が返されます。

例 2：

次の例は、カスタム OLE オブジェクト内でこのメソッドを使用する方法を示します。

```
Dim CellString  
CellString = RPPL1.GetCellString(22,2)
```

GetCellStringHTML メソッド

GetCellStringHTML メソッドは、レポート内の行と列で定義されるセルの値を HTML コードの形式に変換して取得するために、条件スクリプト内で使用します。

HTML 形式に変換された値を取得するには、[管理] ダイアログボックスで [Convert special characters in reports to HTML (レポート内の特殊文字を HTML 用に変換)] チェックボックスがオンになっている必要があります。

構文：

object.GetCellStringHTML(lRow as Long, lCol as Long)

変数	説明
object	参照先のレポートオブジェクトの名前。
lRow as Long	値を取得する行の番号。行番号はゼロから始まります。レポートの左上の最初の行の番号がゼロになります。
lCol as Long	値を取得する列の番号。列番号はゼロから始まります。レポートの左上の最初の列の番号がゼロになります。

戻り値：行と列で指定されたセルの値の文字列表記を HTML 互換のコードに変換したもの。

例：

この例では、条件ベースの Web ページの作成で、列見出しの 1 つが「Profit<Loss>」となっているレポートから、すべての列見出しを抽出して条件ベースのページに表示します。特殊文字を含むセルでは GetCellStringHTML を呼び出します。GetCellStringHTML は特殊文字を HTML 互換のコードに変換します。GetCellString を使うと変換されていない値が返されます。

ProfitLoss.GetCellStringHTML(1, 0) ' 列1の列見出しは「Profit<Loss>」

この呼び出しは、値「Profitc;Losse;」を返します。

GetCellStatus メソッド

GetCellStatus メソッドは、セルの値の元になっているデータに関する情報を取得するために使用します。例えば、Hyperion Enterprise アプリケーションで連結が実行され、そのデータが保護済みに設定された場合には、そのデータを参照しているセルに GetCellStatus メソッドを適用すると、保護済みのステータスを表す整数が返されます。GetCellStatus メソッドの戻り値は INTEGER です。

GetCellStatus メソッドが返す整数は 16 進数です。データが保護されているかどうかを調べるために GetCellStatus メソッドを使用するときには、その整数のどの部分を読み取ろうとしているのかを指定する必要があります。これは保護済みデータ用のマスク (&H0400) を構文の終わりに入れることによって行います。

構文：

object.GetCellStatus(lRow as Long, lCol as Long) And &H0400

変数	説明
object	参照先のレポートオブジェクトの名前。

lRow as Long	ステータスの値を取得する行の番号。行番号はゼロから始まります。レポートの左上の最初の行の番号がゼロになります。
lCol as Long	ステータスの値を取得する列の番号。列番号はゼロから始まります。レポートの左上の最初の列の番号がゼロになります。

戻り値：Integer

適用対象：レポートオブジェクト

例 1：

次の例は、カスタム OLE オブジェクト内でこのメソッドを使用する方法を示します。

```
Dim bDataLocked
bDataLocked = RPPL1.GetCellStatus(22,2) And &H0400
```

GetName メソッド

GetName メソッドは、レポートの名前を取得するために使用します。GetName メソッドにパラメータはありません。戻り値は STRING 値です。

構文：

object.GetName

但し、object は参照先のレポートオブジェクトの名前です。

戻り値：STRING - レポート名

適用対象：レポートオブジェクト

例

```
IF 1 = 1 Then ' Specifies condition to test
Agent.DefMsg = "Testing Names" + Chr(13) + Chr(10)
Agent.DefMsg = Agent.DefMsg + "GetName: " + RPPL1.GetName + Chr(13)
+ Chr(10)
Agent.DefMsg = Agent.DefMsg + "GetTitle: " + RPPL1.GetTitle
AgentEval.Notify ' 通知を送る
End If
```

上記の例では、次のテキストを HTML 形式の通知として送ります。

```
Agent Notifications
These agent notifications have been triggered:
Testing Names
GetName: RPPL1
GetTitle: Consolidating P and L
Triggered by:
testTitle Date:
April 8, 1998 Time:
14:34 Priority:1
```

GetNumRows メソッド

GetNumRows メソッドは、レポート内の行の数を取得するために使用します。GetNumRows メソッドには引数はありません。戻り値は数値です。

構文：

`object.GetNumRows`

但し、`object` は参照先のレポートオブジェクトの名前です。

戻り値：LONG - レポート内の行の数

適用対象：レポートオブジェクト

例 1：

次の例は、カスタム OLE オブジェクト内でこのメソッドを使用する方法を示します。

```
Dim NumberOfRows  
NumberOfRows = RPPL1.GetNumRows
```

GetNumColumns メソッド

GetNumColumns メソッドは、レポート内の列の数を取得するために使用します。GetNumColumns メソッドには引数はありません。戻り値は数値です。

構文：

`object.GetNumColumns`

但し、`object` は参照先のレポートオブジェクトの名前です。

戻り値：LONG - レポート内の列の数

適用対象：レポートオブジェクト

例 1：

次の例は、カスタム OLE オブジェクト内でこのメソッドを使用する方法を示します。

```
Dim NumberOfColumns  
NumberOfColumns = RPPL1.GetNumColumns
```

GetSystemName メソッド

GetSystemName メソッドは、参照先のレポートを作成した Hyperion システムの名前を取得するために、条件スクリプトとエージェント条件内で使用します。このメソッドにパラメータはありません。戻り値はシステム名を含むテキスト文字列です。

構文：

`object.GetSystemName`

但し、`object` は参照先のレポートオブジェクトの名前です。

戻り値：STRING - レポートを作成した Hyperion システムの名前

適用対象：レポートオブジェクト

例 1：

レポート RPPL1 は Hyperion Enterprise 製品の REL4 システムで作成されました。この情報を取得するには、次のように入力します。

```
RPPL1.GetSystemName
```

システムは「REL4」という文字列を返します。

例 2：

次の例は、カスタム OLE オブジェクト内でこのメソッドを使用する方法を示します。

```
Dim SystemName  
SystemName = RPPL1.GetSystemName
```

GetTitle メソッド

GetTitle メソッドは、レポートのタイトル（説明）を取得するために使用します。GetTitle メソッドにパラメータはありません。戻り値は文字列です。

構文：

`object.GetTitle`

但し、`object` は参照先のレポートオブジェクトの名前です。

戻り値：STRING - レポートのタイトル

適用対象：レポートオブジェクト

例：

```
IF 1 = 1 Then ' Specifies condition to test  
Agent.DefMsg = "Testing Names" + Chr(13) + Chr(10)  
Agent.DefMsg = Agent.DefMsg + "GetName: " + RPPL1.GetName + Chr(13)  
+ Chr(10)  
Agent.DefMsg = Agent.DefMsg + "GetTitle: " + RPPL1.GetTitle  
AgentEval.Notify ' 通知を送る  
End If
```

上記の例では、次のテキストを HTML 形式の通知として送ります。

```
Agent Notifications  
These agent notifications have been triggered:  
Testing Names  
GetName: RPPL1  
GetTitle: Consolidating P and L
```

Triggered by:
testTitle Date:
April 8, 1998 Time:
14:34 Priority:1

エージェントオブジェクト

エージェントオブジェクトは現在実行されているエージェントの定義です。エージェントオブジェクトプロパティは、エージェント定義のコンポーネントです。エージェントオブジェクトは、エージェント条件の実行中に自動的に VBScript から「Agent」という名前でアクセスできるようになります。

次のプロパティはエージェント定義の個々のコンポーネントです。

表 9 エージェントオブジェクトのプロパティ

プロパティ	説明
AgentDesc	エージェントの詳細説明を返します。
AgentName	短い形式のエージェント名を返します。
AgentID	エージェントデータベースでそのエージェントを識別する一意のデータベースキーを返します。
DefMsg	システム管理者が定義したデフォルトメッセージがそのエージェントの通知で使用されるように設定します。

次のメソッドがエージェントオブジェクトで使用できます。

表 10 エージェントオブジェクトメソッド

メソッド	説明
GetNumReports	指定されたエージェントによって使用されているレポートの数を返します。
GetReportDisp	評価対象のエージェントオブジェクトに関連付けられているレポートオブジェクトの数を返します。

上級 OLE ユーザ用の情報

エージェントオブジェクトは、エージェント評価オブジェクトで **GetAgent** メソッドを呼び出すことによってアクセスできます。エージェントオブジェクトを参照するときは次の情報を使用してください。

OLE ProgID : "SCHD.AGENTOBJ"

OLE GUID : {CEED4650-3CBF-11D0-A941-0800091B6B1C}

GetNumReports メソッド

GetNumReports メソッドは、スクリプトまたはエージェントによって使用されているレポートの数を取得するために、条件スクリプトやエージェント条件内で使用します。このメソッドにパラメータはありません。戻り値は整数です。

注： このメソッドは OLE オブジェクト内でのみ使用されます。VBScript 内で定義されたエージェントでは関連付けられているレポートの名前がわかるので、このメソッドを VBScript で使用する必要はありません。

構文：

object.GetNumReports

但し、object は参照先のエージェントオブジェクトの名前です。エージェントを呼び出すときの object は「Agent」です。

戻り値： INTEGER - 使用されているレポートの数

適用対象： エージェントオブジェクト、条件オブジェクト

例：

この例では、レポートの一覧を作成し、通知に入れて送ります。

```
Dim Report1
Dim Report2
Dim Report3
If Agent.GetNumReports = 3 Then
    Set Report1 = Agent.GetReportDisp(1)
    Agent.DefMsg = Agent.DefMsg & Chr(13) & Report1.GetName
    Set Report2 = Agent.GetReportDisp(2)
    Agent.DefMsg = Agent.DefMsg & Chr(13) & Report2.GetName
    Set Report3 = Agent.GetReportDisp(3)
    Agent.DefMsg = Agent.DefMsg & Chr(13) & Report3.GetName &
Chr(13)
    AgentEval.Notify
End If
```

GetReportDisp メソッド

GetReportDisp メソッドは、現在評価されているエージェントオブジェクトまたは条件オブジェクトで指定されているレポートオブジェクトを取得するために、条件スクリプトやエージェント条件内で使用します。このメソッドはオブジェクトを返します。

構文：

object.GetReportDisp(ReportNum As Integer)

変数	説明
object	参照先のオブジェクトの名前。エージェントを呼び出すときの object は「Agent」です。
ReportNum	返すレポートの番号。可能な値は 1 から最大レポート数までです。

戻り値： OBJECT - 指定されたレポートオブジェクト

適用対象： エージェントオブジェクト、条件オブジェクト

例：

次の例は、カスタム OLE オブジェクト内でこのメソッドを使用する方法を示します。

```
Dim Report1
Dim Report2
Dim Report3
If Agent.GetNumReports = 3 Then
    Set Report1 = Agent.GetReportDisp(1)
    Agent.DefMsg = Agent.DefMsg & Chr(13) & Report1.GetName
    Set Report2 = Agent.GetReportDisp(2)
    Agent.DefMsg = Agent.DefMsg & Chr(13) & Report2.GetName
    Set Report3 = Agent.GetReportDisp(3)
    Agent.DefMsg = Agent.DefMsg & Chr(13) & Report3.GetName &
Chr(13)
    AgentEval.Notify
End If
```

AgentDesc プロパティ

AgentDesc プロパティは、エージェントの定義の一部として定義されるエージェントの詳細説明に対応します。このプロパティは詳細説明の設定や取得に使用できます。

構文：

値を設定するときに使用する構文：

object.AgentDesc= String

変数	説明
object	参照先のオブジェクト。エージェントを呼び出すときの object は「Agent」です。
String	エージェントに設定する詳細説明。

値を取得するときに使用する構文

String = object.AgentDesc

変数	説明
String	取得した詳細説明を入れる変数の名前。
object	参照先のエージェントオブジェクト。エージェントを呼び出すときの object は「Agent」です。

注： VBScript ではエージェントオブジェクトは「Agent」として登録されます。

データタイプ：String

適用対象：エージェントオブジェクト

例 1 :

この例では、エージェント「BUDGET1」の現在の詳細説明を「Budget Watch Agent」に変更するために、そのエージェントのスクリプト内で次のようにプロパティを入力します。

```
Agent.AgentDesc="Budget Watch Agent"
```

例 2 :

次の例は、VBScript やカスタム OLE オブジェクトから詳細説明を取得するときのプロパティの使用方法を示します。

```
Dim Description  
Description = Agent.AgentDesc
```

AgentName プロパティ

AgentName プロパティは、エージェント定義の一部として定義される短い形式のエージェント名に対応します。このプロパティはエージェント名の設定や取得に使用できます。

構文 :

名前を設定するときに使用する構文 :

```
object.AgentName=String
```

変数	説明
object	参照先のエージェントオブジェクト。エージェントを呼び出すときの object は「Agent」です。
String	短い形式のエージェント名。

名前を取得するときに使用する構文 :

```
String=object.AgentName
```

変数	説明
String	取得した名前を入れる変数の名前。
object	参照先のエージェントオブジェクト。エージェントを呼び出すときの object は「Agent」です。

注 : VBScript ではエージェントオブジェクトは「Agent」として登録されます。

データタイプ : String

適用対象 : エージェントオブジェクト

例 1 :

この例では、エージェント「BUDGET1」の現在の名前を「BUDWATCH」に変更するために、そのエージェントのスクリプト内で次のようにプロパティを入力します。

```
Agent.AgentName="Budwatch"
```

例 2 :

次の例は、VBScript やカスタム OLE オブジェクトから名前を取得するときのプロパティの使用方法を示します。

```
Dim Name  
Name = Agent.AgentName
```

AgentID プロパティ

AgentID プロパティは、データベース表内でそのエージェントを識別する一意のデータベースキーに対応します。AgentID プロパティは ID を取得するだけで、新しい ID を設定することはできません。

構文 :

```
object.AgentID=Long
```

変数	説明
object	参照先のオブジェクト。エージェントを呼び出すときの object は「Agent」です。
Long	エージェントの一意の ID。

注 : VBScript ではエージェントオブジェクトは「Agent」として登録されます。

データタイプ : Long

適用対象 : エージェントオブジェクト

例 1 :

「BUDGET1」エージェントの ID を取得する場合、そのエージェントのスクリプト内に次のようにプロパティを入力します。

```
Agent.AgentID
```

例 2 :

次の例は、VBScript やカスタム OLE オブジェクトから ID を取得するときのプロパティの使用方法を示します。

```
Dim ID  
ID = Agent.AgentID
```

DefMsg プロパティ

DefMsg プロパティは、エージェントの通知に含めるデフォルトテキストに対応します。このテキストはエージェント定義の一部として定義されます。このプロパティはデフォルトメッセージの設定と取得に使用できます。

構文：

デフォルトメッセージを設定するときに使用する構文：

```
object.DefMsg=String
```

変数	説明
object	参照先のオブジェクト。エージェントを呼び出すときの object は「Agent」です。
String	そのエージェントのデフォルトメッセージテキスト。

デフォルトメッセージを取得するときに使用する構文：

```
String=object.DefMsg
```

変数	説明
String	取得したメッセージを入れる変数の名前。
object	参照先のオブジェクト。エージェントを呼び出すときの object は「Agent」です。

注： VBScript ではエージェントオブジェクトは「Agent」として登録されます。

データタイプ：String

適用対象：エージェントオブジェクト

例 1：

「BUDGET1」エージェントのデフォルトメッセージを「Expenses are over budget by more than 5%」に変更するは、そのエージェントのスクリプト内に次のようにプロパティを入力します。

```
Agent.DefMsg="Expenses are over budget by more than 5%"
```

例 2：

次の例は、VBScript やカスタム OLE オブジェクトからデフォルトメッセージを取得するときのプロパティの使用方法を示します。

```
Dim Message  
Message = Agent.DefMsg
```

エージェント評価オブジェクト - 概要

エージェント評価オブジェクトは、エージェントが実行されるコンテキストを扱います。エージェント評価オブジェクトは、エージェントがデータベースを調べ

る必要があるときに、スケジューラによって呼び出されます。エージェント評価オブジェクトのメソッドは、エージェント条件を実行してエージェントのアクションを処理するために使用されます。

エージェント評価オブジェクトは、エージェント条件の実行中に自動的に VBScript から「AgentEval」という名前でアクセスできるようになります。さらに、エージェント評価オブジェクトは、独自に作成したカスタムエージェント OLE オブジェクトに渡すことのできる唯一のオブジェクトです。

次のメソッドがエージェント評価オブジェクトで使用できます。

表 11 エージェント評価のメソッド

メソッド	説明
AddLink	エージェントのユーザ通知に含まれているリンクリストに URL リンクを追加します。
ClearLinks	ユーザに送られるエージェントの通知に含まれている URL リンクをすべて削除します。
GetAgent	評価されるエージェントオブジェクトを返します。
Notify	エージェントに関連付けられている購読ユーザのためのアクションをすべて実行します。
HTMLEncode	特殊文字を HTML 互換タグに変換します。
URLEncode	特殊文字を URL 互換タグに変換します。

上級 OLE ユーザ用の情報：

エージェント評価オブジェクトを参照するときは、次の情報を使用してください。

OLE ProgID："SCHD.AGENTEVAL"

OLE GUID：{DE5819C4-3294-11D0-A93B-0800091B6B1C}

AddLink メソッド

AddLink メソッドは、エージェントの通知でユーザが受け取るメッセージに URL リンクを追加します。リンク先には、動的レポート、静的レポート、イントラネットやインターネット上のあらゆる文書を指定できます。

構文：

object.AddLink(URL, Linktext)

変数	説明
object	参照先のオブジェクトの名前を指定する文字列。エージェント評価オブジェクトを呼び出すときの object は「AgentEval」です。
URL	追加するリンクの URL。引用符で囲んだ文字列を使用。
Linktext	リンク先を記述するテキスト。

戻り値：処理が正常に行われた場合は「TRUE」、そうでなければ「FALSE」

適用対象： エージェント評価オブジェクト

注： URL に含まれているスペース文字はプラス記号 (+) で置き換え、特殊文字 (ANSI 文字コードで 127 以上またはダブルバイト文字) は「%xx」で置き換える必要があります。但し、xx は元の文字の 16 進数値です。

例：

次の例では、AddLink メソッドを使って「RPPL1」レポートに URL リンクを追加します。ユーザへの通知では「Expense Report」がリンクテキストとして表示されます。

```
AgentEval.AddLink "http://HSPIDER/spider.dll?report
&Enterprise&Demo&RPPL1", "Expense Report"
```

ClearLinks メソッド

ClearLinks メソッドは、ユーザに送られるエージェントの通知に含まれる URL リンクをすべて削除するために使用します。デフォルトでは、エージェント定義で指定されているすべての動的レポートへのリンクがそのエージェントの通知に含まれます。ClearLinks メソッドにパラメータはありません。

構文：

`object.ClearLinks`

但し、object は現在参照されているオブジェクトです。エージェント評価オブジェクトを呼び出すときの object は「AgentEval」です。

戻り値： なし

適用対象： エージェント評価オブジェクト

例：

次の例では、ClearLinks メソッドを使って既存のリンクをすべて削除した後、AddLink メソッドを使って静的レポートページへの独自のリンクを追加します。

```
AgentEval.ClearLinks
AgentEval.AddLink "/jansales.htm", "Gross Sales for January by Division"
```

GetAgent メソッド

GetAgent メソッドは、現在評価されているエージェントオブジェクトの名前を取得するために使用します。GetAgent メソッドにパラメータはありません。

構文：

`object.GetAgent`

但し、object は現在の参照先のエージェント評価オブジェクトの名前です。エージェント評価オブジェクトを呼び出すときの object は「AgentEval」です。

戻り値： OBJECT - エージェントオブジェクトの名前

適用対象： エージェント評価オブジェクト

例：

次の例では、`GetAgent` メソッドを使って現在のエージェントオブジェクトの名前を取得します。

```
AgentEval.GetAgent
```

Notify メソッド

`Notify` メソッドは、エージェントが監視している条件が「TRUE」と評価されたときに、そのエージェントに関連付けられているすべてのアクションを実行するために、エージェント条件内で使用します。購読ユーザがその結果作成される通知を見るためには、次の条件が満たされている必要があります。

- エージェントを定義したときに、[Agent Options (エージェントオプション)] ダイアログボックスでそのアクションが設定されている必要があります。
- エージェントの定義で、少なくとも 1 つのアクションが有効にされている必要があります。
- ユーザはそのエージェントの購読を設定したときにそのアクションを選択している必要があります。

構文：

```
object.Notify(Optionaltext)
```

変数	説明
object	参照先のエージェント評価オブジェクトの名前。エージェント評価オブジェクトを呼び出すときの object は「AgentEval」です。
Optionaltext	評価とともに渡す任意のテキスト、STRING 形式。例えば、データの視点を渡すことができます。このテキストはアクション情報オブジェクトの OptionalText プロパティによって取得することができます。

戻り値：

- 0 は正常に処理された場合
- -1 はそのエージェント評価オブジェクトに関連付けられているエージェントオブジェクトがない場合
- -2 はそのエージェント評価オブジェクトに関連付けられているエージェントオブジェクトが無効の場合
- -999 はアクションでエラーが発生した場合

適用対象： エージェント評価オブジェクト

例：

次の例では、`Notify` メソッドを使って現在のエージェントオブジェクトのアクションを開始します。

```
AgentEval.Notify()
```

URLEncode メソッド

URLEncode メソッドは、特殊文字を含むテキスト文字列を URL 互換文字列に変換するために使用します。

構文：

`object.URLEncode`

但し、`object` は参照先のエージェント評価オブジェクトの名前です。

戻り値：変換する必要がある URL 固有の特殊文字を含む URL の文字列。

HTMLEncode メソッド

HTMLEncode メソッドは、特殊文字を含むテキスト文字列を HTML 互換文字列に変換するために使用します。

構文：

`object.HTMLEncode`

但し、`object` は参照先のエージェント評価オブジェクトの名前です。

戻り値：変換する必要がある HTML 固有の特殊文字を含む HTML の文字列。

テンプレートオブジェクト

条件ベースのページをブラウザからアクセスするときに、システムは条件スクリプト内でそのページに適用されるテンプレートを登録します。テンプレートはそのファイル名によって登録されますが、拡張子またはパスは登録に含まれません。テンプレートに含まれている変数は評価されて、適切な値が代入されます。その後で、テンプレートオブジェクトの最終的な値が戻されます。Sub メソッドを使用して、テンプレートの変数に代入する内容を変更できます。

Sub メソッド

Sub メソッドは、条件スクリプト内で参照されているテンプレートファイル内の 1 つの変数の全インスタンスに値を代入するために使用します。

構文：

`object.Sub (Variable, Value)`

変数	説明
<code>object</code>	参照先のテンプレートオブジェクトの名前。
変数	対象の変数の名前を指定する文字列。
値	変数に代入する値の文字列。

適用対象：テンプレートオブジェクト

条件オブジェクト

条件オブジェクトは、条件スクリプト Web ページ用に定義された条件からなるオブジェクトです。条件オブジェクトは VBScript 内で登録されます。条件オブジェクトから条件の定義に含まれるテンプレートオブジェクトやレポートオブジェクトを取得できます。条件オブジェクトは、条件の定義に含まれるカスタム OLE 関数にパラメータとして渡されます。

次のメソッドが条件オブジェクトに適用されます。

表 12 条件オブジェクトのメソッド

メソッド	説明
GetTemplate	HTML テンプレートオブジェクトを返します。
GetNumReports	指定されたエージェントによって使用されているレポートの数を返します。
GetReportDisp	評価対象の条件オブジェクトに関連付けられているレポートオブジェクトの数を返します。
ServerUserName	Web サーバのログオン ID を返します。
SetErrorNum	エラー発生時に、システムがオブジェクトから受け取り、ブラウザに表示するエラー番号を設定します。
SetErrorMsg	エラー発生時に、システムがユーザのブラウザに表示するエラーメッセージを設定します。
HTMLEncode	特殊文字を HTML 互換タグに変換します。
URLEncode	特殊文字を URL 互換タグに変換します。
URLParam	GET または POST 要求で、CRITERIA 関数の呼び出しに渡される文字列パラメータの値を返します。
URLParamNum	GET または POST 要求で、CRITERIA 関数の呼び出しに渡される数値パラメータの値を返します。

VBScript で有効なメソッドは、HTMLEncode、URLEncode、URLParam、および URLParamNum のみです。

上級 OLE ユーザ用の情報

条件オブジェクトは VBScript では登録されません。定義されている OLE 関数の呼び出しにオブジェクトの参照を渡すことによってのみアクセスできます。条件オブジェクトの参照では次の情報を使用してください。

OLE ProgID : "CRITERIA.CRITERIA"

OLE GUID : {5D511245-397C-11D0-ADC9-080009A80DEF}

GetTemplate メソッド

GetTemplate メソッドは、条件定義で使用するために定義されているテンプレートオブジェクトを取得するために使用します。テンプレートオブジェクトは条件ベースのページをブラウザに表示するときのページ書式に使用されます。

構文：

`object.GetTemplate`

但し、object はカスタム関数に渡す条件オブジェクトの名前。

戻り値：OBJECT - 条件定義で使用するために定義されているテンプレートオブジェクト

適用対象：条件オブジェクト

例：

次の例では、渡された条件オブジェクトからテンプレートオブジェクトを取得します。

```
Dim htxTemp as Object
set htxTemp = critobj.GetTemplate
```

ServerUserName メソッド

ServerUserName メソッドは、条件の定義で使用する Web サーバのユーザ名を取得するために使用します。

構文：

`object.ServerUserName()`

但し、object はカスタム関数に渡す条件オブジェクトの名前。

戻り値：Web サーバのユーザ名を示す文字列。

適用対象：条件オブジェクト

例：次の条件 VBScript の例では、ServerUserName()の戻り値に基づいて、実際の数値を表示するかテキストメッセージを返すかを決定します。

```
if CBP2.GetCellDouble( 1, 3)>0 then
if Criteria.ServerUserName()="FREDJONES" then
MyCriteriaPage.Sub "%UserVariable%",
CBP2.GetCellDouble( 1, 3)
Else
MyCriteriaPage.Sub "%UserVariable%","Over Budget"
end if
end if
```

SetErrorNum メソッド

OLE 関数が要求されたアクションを完了できない場合、SetErrorNum メソッドと SetLastErrorMsg メソッドを使用してエラー情報を返すことができます。エラー発生時に、システムは SetErrorNum メソッドで非ゼロ値をチェックします。ゼロ以外の

値が戻った場合、システムはエラー番号を HTML でユーザに報告し、SetErrorMsg メソッドを使って設定されたエラーメッセージを報告に追加します。

SetErrorNum メソッドを SetErrorMsg メソッドなしで使用すると、システムはエラーメッセージの番号を返しますが、説明のテキストは付きません。SetErrorMsg を SetErrorNum なしで使用すると、システムはエラーを報告しません。

構文：

object.SetErrorNum(ErrNum as Integer)

変数	説明
object	カスタム関数に渡す条件オブジェクトの名前。
ErrNum	システムによってユーザに報告されるエラー番号。この数値がゼロより大きな値のときにシステムはエラーを報告します。

戻り値：なし

適用対象：条件オブジェクト

例：

次の例では、渡された条件オブジェクトのエラー番号とメッセージを設定します。

```
If (nRetVal < > 0) then
critobj.SetErrorNum nRetVal
critobj.SetErrorMsg "Could Not Load required <BR>
information          from resources<BR>"
End if
```

SetErrorMsg メソッド

OLE 関数が要求されたアクションを完了できない場合、SetErrorNum メソッドと SetErrorMsg メソッドを使用してエラー情報を返すことができます。エラー発生時に、システムは SetErrorNum メソッドで非ゼロ値をチェックします。ゼロ以外の値が戻った場合、システムはエラー番号を HTML でユーザに報告し、SetErrorMsg メソッドを使って設定されたエラーメッセージを報告に追加します。

SetErrorNum メソッドを SetErrorMsg メソッドなしで使用すると、システムはエラーメッセージの番号を返しますが、説明のテキストは付きません。SetErrorMsg を SetErrorNum なしで使用すると、システムはエラーを報告しません。

構文：

object.SetErrorMsg(ErrMsg)

変数	説明
object	カスタム関数に渡す条件オブジェクトの名前。
ErrMsg	ユーザに報告するエラーメッセージの文字列。エラーメッセージには HTML 書式を適用する必要があります。

戻り値：なし

適用対象： 条件オブジェクト

例：

次の例では、渡された条件オブジェクトのエラー番号とメッセージを設定します。

```
If (nRetVal < > 0) then
critobj.SetErrorNum nRetVal
critobj.SetErrorMsg "Could Not Load required <BR>
information                      from resources<BR>"
End if
```

URLEncode メソッド

URLEncode メソッドは、特殊文字を含むテキスト文字列を URL 互換文字列に変換するために使用します。

構文：

object.URLEncode

但し、object はカスタム関数に渡す条件オブジェクトの名前。

戻り値： 変換する必要がある URL 固有の特殊文字を含む URL の文字列。

例：

次の例では、条件ベースの Web ページで、「Profit & Loss」というレポートへのリンクを作成します。但し、アンパサンド記号「&」は URL の区切り文字として使用されるため、そのまま使用することはできません。このレポート名を使用するには、アンパサンドの特殊文字コードを調べて入力できますが、URLEncode を呼び出して URL 互換文字列に変換した方が確実です。

```
LinkString"\Hspider\spider.dll?report&Enterprise
&Demo&"LinkString = LinkString + Criteria.URLEncode("Profit & Loss")
```

この呼び出しが処理された後、LinkString には次の文字列が入ります。"\Hspider\spider.dll?report&Enterprise&Demo&Profit+%26+Loss"。これは Sub メソッドを使ってテンプレートオブジェクトの HTML テンプレートに代入できます。

HTMLEncode メソッド

HTMLEncode メソッドは、特殊文字を含むテキスト文字列を HTML 互換文字列に変換するために使用します。

構文：

object.HTMLEncode

但し、object はカスタム関数に渡す条件オブジェクトの名前。

戻り値： HTML 固有の特殊文字を含むため、変換する必要がある HTML の文字列。

例：

条件ベースの Web ページで、「Profit<Loss」というテキストを表示しようとしていますが、「<」という記号は HTML で使用されるので、そのまま使用できません。

このテキストを表示するには、小なり記号「<」の特殊文字コードを調べることができますが、`HTMLEncode` を呼び出して HTML 互換文字列に変換した方が確実です。

```
SomeString = Criteria.HTMLEncode("Profit<Loss")
```

この呼び出しが処理された後、`SomeString` には「Profitc;Loss」が入り、`Sub` メソッドをテンプレートオブジェクトに適用して HTML テンプレートに代入できます。

URLParam メソッド

`URLParam` メソッドは、GET または POST 要求で条件関数の呼び出しに文字列として渡されるパラメータの値を取得するために使用します。

構文：

`object.URLParam`

但し、`object` は条件オブジェクトの名前です。

戻り値： `STRING` - パラメータの値の文字列。パラメータが見つからない場合は、空の文字列が返されます。

適用対象： 条件オブジェクト

例：

次の例では、条件オブジェクトから URL パラメータを取得します。

```
mytemplate.Sub "%status%", criteria.URLParam("status")
```

URLParamNum メソッド

`URLParamNum` メソッドは、GET または POST 要求で条件関数の呼び出しに数値として渡されるパラメータの値を取得するために使用します。

構文：

`object.URLParamNum`

但し、`object` は条件オブジェクトの名前です。

戻り値： `LONG` - パラメータの数値。パラメータが見つからない場合は、0 が返されます。

適用対象： 条件オブジェクト

例：

次の例では、条件オブジェクトから URL パラメータを取得します。

```
if myreport.GetCellDouble(1,1) >
criteria.URLParamNum("tolerance") then      mytemplate.Sub "%somegif%",
"Good.gif"
Else
    mytemplate.Sub "%somegif%", "bad.gif"
end if
```

アクション情報オブジェクト

アクション情報オブジェクトは、監視していたデータ条件を検出したエージェントに関する情報を含むオブジェクトです。システムは、そのエージェントに定義されていて、購読ユーザへ通知する必要のあるエージェントアクションオブジェクトにアクション情報オブジェクトを渡します。

注： アクション情報オブジェクトは、OLE オートメーションで作成されたカスタムエージェントアクションによってパラメータとして受け取られる必要があります。カスタムエージェントアクションの定義については、『Hyperion Enterprise Reporting 管理者用リファレンスガイド』を参照してください。

次のプロパティがアクション情報オブジェクトに適用されます。

表 13 アクション情報オブジェクトのプロパティ

プロパティ	説明
DataText	エージェントへの購読を設定するときにユーザが入力したデータ。
OptionalText	エージェントがトリガされたときに、Notify コマンドに渡すテキスト。
MessageText	エージェントの通知に含めるデフォルトメッセージのテキスト。これはエージェント定義の一部として定義されたものです。
UserMessageText	購読設定時に、購読ユーザがエージェント通知で使用する個人的なメッセージとして入力したテキスト。
UserName	購読ユーザの名前。
AgentName	アクションがトリガされる原因となったエージェントの名前。
AgentDescription	アクションの原因となったエージェントの説明。
ErrorText	エラー発生時に呼び出し元のオブジェクトに返すエラーメッセージ。

次のメソッドがアクション情報オブジェクトに適用されます。

表 14 アクション情報オブジェクトのメソッド

メソッド	説明
GetNumLinks	そのエージェントに関連付けられている URL リンクの数进行返します。
GetLinkText	エージェントの URL リンクのリンクテキストとして指定されているテキストを返します。
GetLinkURL	エージェントの URL リンクの URL の值を返します。

上級 OLE ユーザ用の情報

アクション情報オブジェクトは、ディスパッチポインタをエージェントアクション OLE 関数の呼び出しに渡すことによってのみアクセスできます。アクション情報オブジェクトの参照では次の情報を使用してください。

OLE ProgID : SCHD.ACTIONINFO

OLE GUID : {322DC5F1-5441-11D0-BC05-080009C30B9A}

DataText プロパティ

DataText プロパティは、ユーザがエージェントへの購読を設定したときに入力したアクション関連のデータです。例えば、電子メールでの通知用に入力された電子メールアドレスは、DataText プロパティの一部として保存されます。このプロパティはユーザが入力したアクション関連のテキストに対してのみ使用できます。

構文 :

String = object.DataText

変数	説明
String	取得したテキストを入れる変数の名前。
object	参照先のアクション情報オブジェクトの名前。

データタイプ : String

適用対象 : アクション情報オブジェクト

例 :

次の例では、VBScript またはカスタム OLE オブジェクトから購読ユーザが入力したデータのみを取得するためにプロパティを使用します。

```
Dim StrToField
StrToField = ActInfo.DataText
```

MessageText プロパティ

MessageText プロパティは次の 2 つを含みます。

- エージェントの定義の一部として入力されたデフォルトメッセージ。
- そのエージェントが評価されたときに、エージェント評価オブジェクトの DefMsg プロパティに設定された値。

MessageText プロパティはメッセージのみを取得するために使用できます。

構文 :

String = object.MessageText

変数	説明
String	取得したデフォルトメッセージを入れる変数の名前。
object	参照先のアクション情報オブジェクトの名前。

データタイプ : String

適用対象 : アクション情報オブジェクト

例：

次の例は、VBScript やカスタム OLE オブジェクトからデータを取得するときのプロパティの使用方法を示します。

```
Dim MsgValue
MsgValue = ActInfo.MessageText
```

UserMessageText プロパティ

UserMessageText プロパティは、ユーザがエージェントへの購読を設定したときに入力する個人的なメッセージです。UserMessageText プロパティは個人的なメッセージのみを取得するときに使用できます。

構文：

```
String = object.UserMessageText
```

変数	説明
String	取得した個人的なメッセージを入れる変数の名前。
object	参照先のアクション情報オブジェクトの名前。

データタイプ：String

適用対象：アクション情報オブジェクト

例：

次の例は、VBScript やカスタム OLE オブジェクトからデータを取得するときのプロパティの使用方法を示します。

```
Dim MsgValue
MsgValue = ActInfo.UserMessageText
```

OptionalText プロパティ

OptionalText プロパティは、システムが Notify 関数にパラメータとして渡したテキストです。現在、電子メール通知でも Web 通知でもこの値を使用するオブジェクトはありません。この値は評価の情報を AgentEval.Notify メソッドに渡すために使用できます。

構文：

```
String = object.OptionalText
```

変数	説明
String	取得した値を入れる変数の名前。
object	参照先のアクション情報オブジェクトの名前。

データタイプ：String

適用対象：アクション情報オブジェクト

例：

次の例は、VBScript やカスタム OLE オブジェクトからデータを取得するときのプロパティの使用方法を示します。

```
Dim OptValue
OptValue = ActInfo.OptionalText
```

ErrorText プロパティ

ErrorText プロパティは、参照先のカスタムアクションオブジェクトでエラーが発生したときに、システムが使用できるエラー情報を指定するために使用します。このプロパティはエラーテキストを設定するためにのみ使用します。

構文：

```
object.OptionalText = String
```

変数	説明
object	参照先のアクション情報オブジェクトの名前。
String	エラー発生で返すエラーメッセージ。

データタイプ：String

適用対象：アクション情報オブジェクト

例：

次の例では、このプロパティを使ったエラーメッセージの設定方法を示します。

```
ActionInfo.ErrorText = "Could not Open file"
```

UserName プロパティ

UserName プロパティは、参照先のエージェントを購読しているユーザ名です。システムは、設定する必要があるユーザ通知ごとにアクション情報オブジェクトのインスタンスを作成します。このプロパティは、現在のアクション情報オブジェクトの通知でのユーザ名を取得するためにのみ使用できます。

構文：

```
String = object.UserName
```

変数	説明
String	取得したユーザ名を入れる変数の名前。
object	参照先のアクション情報オブジェクトの名前。

データタイプ：String

適用対象：アクション情報オブジェクト

例：

次の例は、VBScript やカスタム OLE オブジェクトからデータを取得するときのプロパティの使用方法を示します。

```
Dim SubsName
SubsName = ActInfo.UserName
```

AgentName プロパティ

AgentName プロパティは、アクション情報オブジェクトが参照しているエージェントの名前です。このプロパティはエージェントの名前を取得するために使用できます。

構文：

String=object.AgentName

変数	説明
String	取得したエージェント名を入れる変数の名前。
object	参照先のアクション情報オブジェクトの名前。

データタイプ：String

適用対象：アクション情報オブジェクト

例：

次の例は、VBScript やカスタム OLE オブジェクトからデータを取得するときのプロパティの使用方法を示します。

```
Dim agName
agName = ActInfo.AgentName
```

AgentDescription プロパティ

AgentDescription プロパティは、エージェントの説明です。このプロパティはエージェントの説明を取得するためにのみ使用できます。

構文：

String = object.AgentDescription

変数	説明
String	取得したエージェント名を入れる変数の名前。
object	参照先のアクション情報オブジェクトの名前。

データタイプ：String

適用対象：アクション情報オブジェクト

例：

次の例は、VBScript やカスタム OLE オブジェクトからデータを取得するときのプロパティの使用方法を示します。

```
Dim agdescription
agdescription = ActInfo.AgentDescription
```

GetNumLinks メソッド

GetNumLinks メソッドは、エージェントに関連付けられている URL リンクの数を取得するために使用します。このリンク数には、エージェントで使用されている元のレポートへのリンクに加えてエージェント評価で追加されたリンクが含まれます。このメソッドにパラメータはありません。戻り値は整数です。

構文：

object.GetNumLinks

但し、object は参照先のアクション情報オブジェクトの名前です。

戻り値： LONG INTEGER - 参照先のエージェントに関連付けられているリンクの数

適用対象： アクション情報オブジェクト

例：

次の例では、「ACTINFO」というアクション情報オブジェクトに関連付けられているリンクの数を取得します。

```
Dim INumLinks
lNumLinks = ActInfo.GetNumLinks
```

GetLinkText メソッド

GetLinkText メソッドは、エージェントに関連付けられている URL リンクの 1 つからリンクテキストを取得するために使用します。対象のリンクには、エージェントで使用されている元のレポートへのリンクに加えてエージェント評価で追加されたリンクが含まれます。

構文：

object.GetLinkText(nIndex)

変数	説明
object	参照先のアクション情報オブジェクトの名前。
nIndex	取得すべきリンクテキストのインデックス番号。

戻り値： STRING - リンクテキスト

適用対象： アクション情報オブジェクト

例：

次の例では、GetLinkURL メソッドと GetLinkText メソッドを使ってレポートへの HTML リンクを作成します。

```
Dim strURL
strURL = "<A HREF=" ActInfo.GetLinkURL(1) "> " & ActInfo.GetLinkText(1)
"</A>"
```

GetLinkURL メソッド

GetLinkURL メソッドは、エージェントに関連付けられているリンクの 1 つからリンク URL を取得するために使用します。対象のリンクには、エージェントで使用されている元のレポートへのリンクに加えてエージェント評価で追加されたリンクが含まれます。

構文：

object.GetLinkURL(nIndex)

変数	説明
object	参照先のアクション情報オブジェクトの名前。
nIndex	取得すべきリンク URL のインデックス番号。

戻り値：STRING - リンク URL

適用対象：アクション情報オブジェクト

例：

次の例では、GetLinkURL メソッドと GetLinkText メソッドを使ってレポートへの HTML リンクを作成します。

```
Dim strURL
strURL = "<A HREF=" & ActInfo.GetLinkURL(1) & "> " & ActInfo.GetLinkText(1)
"</A>"
```


3

データベース表のリファレンス

この章の内容

データベース表	81
表間の関係.....	82
エージェント表	83

この章はプログラマやシステム管理者のためのリファレンスです。Hyperion Enterprise Reporting Web で使用されるすべてのデータベース表についての説明が記載されています。これらの表はエージェント専用です。これらの表は、使用するシステムのカスタム拡張機能の開発に必要な場合に参照してください。

注： これらの表はシステム管理者が変更すべきものではありません。

各表の説明にはその表の列に関する詳しい情報が含まれています。列は、次の項目に分けて説明します。

- 内容：その列に含まれる情報の種類。デフォルト値とその列の検証に使用される表がある場合は、それらも説明に含まれます。
- データタイプ： CHARACTER や INTEGER のように、列に入るデータの型。
- ルール：その列に入るデータのルール。その列での有効な値を示すルールです。
- 例：その列の値の例。

データベース表

次の表では、Hyperion Enterprise Reporting Web データベース表の内容を説明します。さらに、その表にデータを入れる前にデータを入れるべき他の表があれば、その情報も説明に含まれます。表間の関係について詳しくは、[82 ページの「表間の関係」](#)を参照してください。

表 15 Hyperion Enterprise Reporting Web データベース表

表	内容	検証表
Action_Definition	現在システム内で定義されているすべてのアクション。アクションはエージェントがトリガされたときに実行される一連の処理です。	

表	内容	検証表
Action_Subscription	エージェントに購読を設定したときにユーザが選択したアクション。	Subscriptions、Action_Definition
Agent_Definition	エージェントの説明。エージェントの名前、説明、およびデフォルトメッセージが含まれます。	
Agent_Definition_Security	エージェントのセキュリティ設定。エージェント定義の一部です。	Agent_Definition
Agent_Notifications	エージェントがトリガされたときに各購読ユーザに送られた通知の記録。	Agent_Definition、Subscriptions
Agent_Schedule	エージェントを実行する頻度。エージェント定義の一部です。	Agent_Definition
Available_Actions	特定のエージェントに関連付けられているアクションのリスト。ユーザがエージェントに購読の設定をするときに選択できるアクションのリストです。	Action_Definition、Agent_Definition
DataObjs	エージェントに関連付けられているレポートへのポインタ。	Agent_Definition
Report_Links	通知の中の各レポートと一緒に送信するテキストメッセージと、レポートを開くために必要な URL。	Agent_Definition、Agent_Notifications、Subscriptions
Retry Actions	正常に完了しなかったエージェント通知に関する情報。システムは定期的にこの表に記録されている通知の再実行を試みます。	Available_Actions、Subscriptions
Retry Links	正常に完了しなかったエージェント通知に関連付けられているレポートに関する情報。	Retry_Actions
Scheduled_ObjPOVs	エージェントがエージェント条件を調べるためにデータベースをチェックすべき日付と時刻のすべてのエントリ。	Agent_Definition、Agent_Schedule、DataObjs
Subscriptions	各エージェントを購読しているユーザのリスト。	Agent_Definition

表間の関係

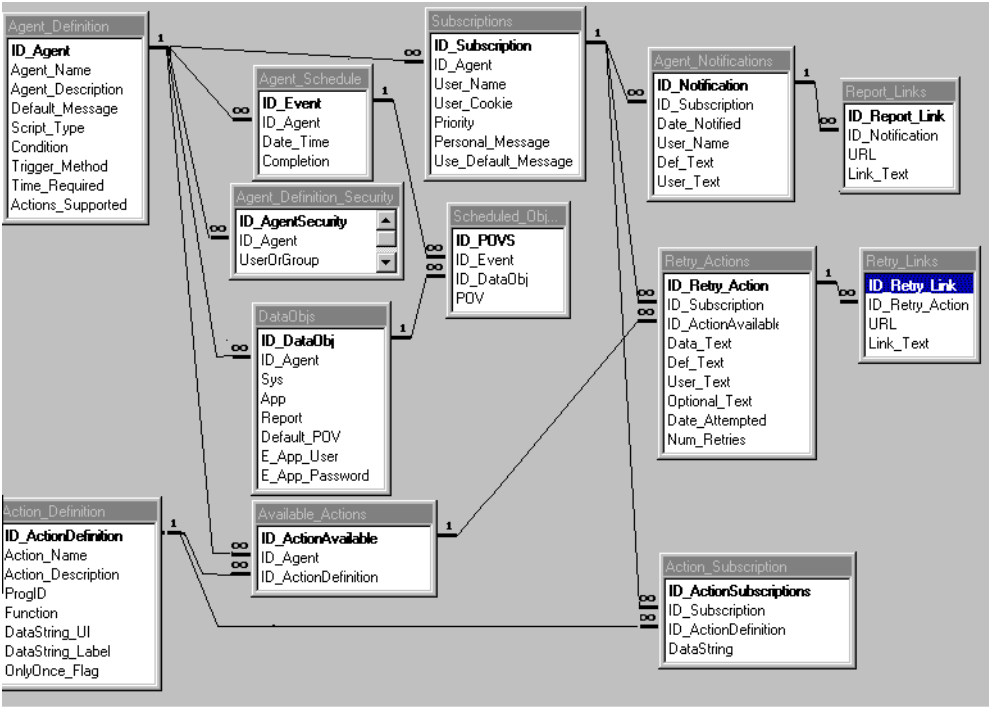
次の図は、Hyperion Enterprise Reporting Web のデータベース表間の関係を示します。表の読み方を以下で説明します。

- 2つの表を結ぶ線は、その2つの表間でデータが共有されていることを示します。例えば、Agent_Definition 表に線で結ばれている5つの表は、すべて Agent_Definition 表とデータを共有し、エージェント定義の一部となっています。

- 表間の関係を示す 1 と ∞（多数）の記号は、データベース関係が 1 から多数へか、多数から 1 へかを示します。例えば、Agent_Definition 表から Subscriptions 表への関係は、図によると 1 から多数への関係であることが示されています。これは、1 つのエージェントが複数の購読ユーザを持つことができることを意味します。逆に、Subscriptions 表から Agent_Definition 表への関係は、多数から 1 への関係になります。これは、Subscriptions 表内の個々のレコードは Agent_Definition 表内の 1 つのレコードにのみ関連することを意味します。

次の図でさらに詳しい表の内容を見ることができます。

図 1 データベース表



エージェント表

以下の項では、Hyperion Enterprise Reporting Web 内のエージェント表を説明します。

Agent_Definition

Agent_Definition 表にはシステム内のすべてのエージェントが含まれています。[管理] プログラムでエージェントが定義されるたびに、システムはこの表にレコードを追加します。

表 16 Agent_Definition 表

フィールド	説明
ID_Agent	内容：エージェントを識別する一意の ID。 データタイプ：NUMBER (LONG)

フィールド	説明
	例 : 1058
Agent_Name	内容 : エージェントの一意の名前。 データタイプ : TEXT ルール : 50 字以内 例 : 1,000 個あたりの製品欠陥数
Agent_Description	内容 : エージェントの説明。 データタイプ : TEXT ルール : 255 字以内 例 : スポーツ用品での製品ごとの 1,000 個あたりの欠陥数
Default_Message	内容 : エージェントの通知に含めるメッセージ。 データタイプ : MEMO 例 : 「GetRowHead」列の欠陥数が許容値を 5 パーセント以上超えています。
Script_Type	内容 : そのエージェントの定義に使用されたのは OLE かまたは VBScript メソッドの 1 つかを示すフラグ。 データタイプ : BYTE ルール : 1 は VBScript、3 は OLE オートメーション。デフォルトは 1 です。 例 : 1
Condition	内容 : エージェントがデータベースをチェックしたときに評価するデータ条件。VBScript ステートメントまたは OLE オブジェクトへの参照を 1 つ含むことができます。 データタイプ : MEMO 例 : IF EXPCOMP.GetCellIDouble(1, 3) > 5 THEN AgentEval.Notify END IF.
Trigger_Method	このリリースでは未使用。
Time_Required	このリリースでは未使用。
Actions_Supported	内容 : このオプションのオンオフのフラグ。 データタイプ : BYTE ルール : 1 はオン、0 はオフ。デフォルトは 0 です。 例 : 1

Action_Definition

Action_Definition 表にはシステム内で定義されているすべてのアクションが記録されています。アクションは、エージェントがトリガされたときに実行される一連の処理です。アクションには購読ユーザに電子メールを送信する、Web ページを作成する、FAX を作成する、その他があります。[Financial Agent Management (財務エージェントの管理)] ダイアログボックスの中の [オプション] 設定を使用してアクションを定義するたびに、この表にレコードが追加されます。アクション

をエージェントに関連付けるときには、この表に含まれている情報が使用されます。

表 17 Action_Definition 表

フィールド	説明
ID_ ActionDefinition	<p>内容：アクションを識別する一意の ID。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：8503</p>
Action_Name	<p>内容：アクションの一意の名前。</p> <p>データタイプ：TEXT</p> <p>ルール：50 字以内</p> <p>例：FAX 通知</p>
Action_Description	<p>内容：アクションの説明。</p> <p>データタイプ：TEXT</p> <p>ルール：255 字以内</p> <p>例：指定された番号への FAX による通知。</p>
ProgID	<p>内容：このアクションが実行されるときに呼び出される OLE オートメーションオブジェクトの ID。</p> <p>データタイプ：TEXT</p> <p>ルール：100 字以内</p> <p>例：HYPEXT.HYPAGENTS</p>
Function	<p>内容：このアクションが実行されたときに OLE オートメーションオブジェクト内で呼び出される関数の ID。</p> <p>データタイプ：TEXT</p> <p>ルール：255 字以内</p> <p>例：FAX</p>
Datastring_UI	<p>内容：このアクションのために購読フォームに追加のユーザインターフェイス要素を含めるかどうかを示します。オプションには、「なし」、「Check Box（チェックボックス）」、「Edit Box（編集ボックス）」があります。</p> <p>データタイプ：TEXT</p> <p>ルール：50 字以内</p> <p>例：Edit Box（編集ボックス）</p>
Datastring_Label	<p>内容：追加のユーザインターフェイス要素のラベル。</p> <p>データタイプ：TEXT</p> <p>ルール：50 字以内</p> <p>例：FAX 番号</p>

DataObjs

DataObjs 表には、システム内の各エージェントが使用する個々のレポートのレコードが入っています。エージェントの定義でそのエージェントに関連付けられた個々のレポートに対して、この表内に 1 つのレコードが作成されます。エージェントがデータの変更を調べるためにデータベースをチェックするときに、この表のレコードを使ってレポートへアクセスします。

表 18 DataObjs 表

フィールド	説明
ID_DataObj	内容 ：この表内のレコードを識別する一意の ID。 データタイプ ：NUMBER (LONG) 例 ：6802
ID_Agent	内容 ：エージェントを識別する一意の ID。 データタイプ ：NUMBER (LONG) 例 ：1058
Sys	内容 ：Hyperion システムを識別する ID。エージェントがデータベース内の Hyperion レポートをチェックするときにアクセスするデータが入っているシステムです。 データタイプ ：TEXT ルール ：50 字以内 例 ：リリース 4
App	内容 ：オラクル社の Enterprise Performance Management ソフトウェアのアプリケーションを識別する ID。エージェントが Hyperion レポートをチェックするときにアクセスするデータを扱うアプリケーションです。 データタイプ ：TEXT ルール ：50 字以内 例 ：DEMO
Report	内容 ：Hyperion レポートを識別する ID。エージェントがデータベースをチェックするときにアクセスするデータが入っているレポートです。 データタイプ ：TEXT ルール ：50 字以内 例 ：RPPL2.RPT
Default_POV	内容 ：デフォルトのデータ視点。エージェントが Hyperion レポートをチェックするときにシステムによって使用されるデータ視点です。 データタイプ ：MEMO 例 ：&エンティティ=USA&データ種別=ACTUAL&期間=97 年 1 月&期間単位=M.CTD&勘定科目=GRSALES&サブエンティティ=なし&サブ勘定科目 1=なし&サブ勘定科目 2=なし&フィールド=VAL
E_App_User	内容 ：暗号化されたユーザ名。エージェントが Hyperion レポートをチェックするときにシステムによって使用されるユーザ名です。 データタイプ ：TEXT

フィールド	説明
	ルール : 255 字以内 例 : Janet Matthews
E_App_Password	内容 : 暗号化されたパスワード。エージェントが Hyperion レポートをチェックするときにシステムによって使用されるパスワードです。 データタイプ : TEXT ルール : 255 字以内

Agent_Schedule

Agent_Schedule 表には、エージェントがデータベースをチェックする日程が個々のレコードとして入っています。エージェントの定義で日付と時刻を設定するたびに、この表にレコードが 1 つ追加されます。日程に従ってチェックが実行されると、完了ステータスを示す値が記録されます。

表 19 Agent_Schedule 表

フィールド	説明
ID_Event	内容 : この表内のレコードを識別する一意の ID。 データタイプ : NUMBER (LONG) 例 : 0087
ID_Agent	内容 : エージェントを識別する一意の ID。 データタイプ : NUMBER (LONG) 例 : 1058
Date_Time	内容 : エージェントを実行する日付と時刻。 データタイプ : Date/Time 例 : 1997 年 1 月 1 日、10:00:00 PM
Completion	内容 : そのチェックが完了したかどうか、チェックのステータスを示します。 データタイプ : NUMBER (LONG) ルール : 「-1」は未定義、「0」は予定、「1」は完了、「2」は削除済み、「3」はエラー、「4」は保留中。デフォルトは 0 です。 例 : 1

Scheduled_ObjPOVS

Scheduled_ObjPOV 表には、エージェントが実行されるときに使用されるデータの視点が入っています。エージェントの定義で、日程を組んだチェックに対してデータの視点を設定すると、システムはこの表の中にそのレコードを作成します。

表 20 Scheduled_ObjPOV 表

フィールド	説明
ID_POVs	<p>内容：この表内のレコードを識別する一意の ID。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：0008</p>
ID_Event	<p>内容：チェックの日程を識別する一意の ID。そのエージェントの Agent_Schedule 表に定義されている ID_Event です。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：0067</p>
ID_DataObj	<p>内容：エージェントが日程に従って実行されたときにチェックするレポートを識別する一意の ID。DataObjs 表で定義されている ID_DataObj です。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：6057</p>
POV	<p>内容：日程に従ってエージェントがチェックを実行するときにシステムが使用するデータの視点。</p> <p>例：MEMO</p> <p>例：&エンティティ=USA&データ種別=ACTUAL&期間=97 年 1 月&期間単位=M.CTD&勘定科目=GRSALES&サブエンティティ=なし&サブ勘定科目 1=なし&サブ勘定科目 2=なし&フィールド=VAL</p>

Agent_Definition_Security

Agent_Definition_Security 表には、個々のエージェントに対するアクセス権を持つ NT ユーザおよびユーザグループを示すレコードが入っています。エージェントの定義でそのエージェントにユーザやユーザグループを関連付けると、この表にそのレコードが追加されます。この表はエージェントの購読や実行で使用されます。

表 21 Agent_Definition_Security 表の列

フィールド	説明
ID_AgentSecurity	<p>内容：この表内のレコードを識別する一意の ID。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：3114</p>
ID_Agent	<p>内容：エージェントを識別する一意の ID。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：1058</p>
UserorGroup	<p>内容：この表にアクセス権を持つ NT ユーザやユーザグループの名前。</p> <p>データタイプ：TEXT</p> <p>ルール：255 字以内</p> <p>例：Joe Williams</p>

UGF	<p>内容 : UserOrGroup 列の値がユーザであるかユーザグループであることを示す値。</p> <p>データタイプ : Integer</p> <p>ルール : 0 はユーザ、1 はグループ。</p> <p>例 : 1</p>
-----	--

Available_Actions

Available_Actions 表には特定のエージェントに関連付けられているすべてのアクションのリストが入っています。アクションはエージェントがトリガされたときに実行される処理です。エージェントに使用可能なアクションの 1 つを関連付けると、この表にレコードが追加されます。エージェントがトリガされると、システムはこの表の中のレコードを使ってどのアクションを実行すべきかを調べます。

表 22 Available_Actions 表の列

フィールド	説明
ID_ActionAvailable	<p>内容 : エージェントに関連付けられたアクションを識別する一意の ID。</p> <p>データタイプ : NUMBER (LONG)</p> <p>例 : 0016</p>
ID_Agent	<p>内容 : エージェントを識別する一意の ID。</p> <p>データタイプ : NUMBER (LONG)</p> <p>例 : 1058</p>
ID_ActionDefinition	<p>内容 : アクションを識別する一意の ID。</p> <p>データタイプ : NUMBER (LONG)</p> <p>例 : 0135</p>

Subscriptions

Subscriptions 表には、特定のエージェントを購読しているユーザとユーザグループがすべて入っています。ユーザが購読設定 Web ページからエージェントへの購読を設定すると、この表にそのレコードが追加されます。エージェントがトリガされると、システムはこの表に入っている情報を使って、どのユーザにどの方法で通知すべきかを調べます。

表 23 Subscriptions 表

フィールド	説明
ID_Subscription	<p>内容 : 購読レコードを識別する一意の ID。</p> <p>データタイプ : NUMBER (LONG)</p> <p>例 : 0013</p>
ID_Agent	<p>内容 : エージェントを識別する一意の ID。</p> <p>データタイプ : NUMBER (LONG)</p>

フィールド	説明
	例 : 1058
User_Name	内容 : そのエージェントを購読している NT ユーザやユーザグループの名前。 データタイプ : TEXT ルール : 255 字以内 例 : Darlene Emerson
User_Cookie	このリリースでは未使用。
Priority	内容 : ユーザがエージェントとその通知に割り当てた優先順位。 データタイプ : NUMBER (LONG) 例 : 1
Personal_Message	内容 : 購読ユーザが通知に含める個人的メッセージとして定義したもの。 データタイプ : MEMO 例 : この通知を受け取ったらアカウントマネージャに電子メールを送ること。
Use_Default_Message	内容 : そのエージェントに定義されているデフォルトメッセージを含めるべきかどうかを示す値。 データタイプ : BOOLEAN ルール : 1 はデフォルトメッセージを含め、0 は含めない。 例 : 1

Action_Subscription

Action_Subscription 表には、各ユーザが個々のエージェントへの購読を設定したときに選択したアクションのリストが入っています。ユーザが購読設定 Web ページでアクションを選択すると、この表にレコードが追加されます。エージェントが実行されると、システムはこのレコードを使って各購読ユーザにどのように通知すべきかを調べます。

表 24 Action_Subscription 表

フィールド	説明
ID_ActionSubscriptions	内容 : ユーザが選択した購読アクションを識別する一意の ID。 データタイプ : NUMBER (LONG) 例 : 0216
ID_Subscription	内容 : 購読レコードを識別する一意の ID。 データタイプ : NUMBER (LONG) 例 : 0035
ID_ActionDefinition	内容 : アクションを識別する一意の ID。 データタイプ : NUMBER (LONG)

フィールド	説明
	例 : 3045
DataSource	内容 : 購読ユーザが電子メール通知を選択した場合に指定する電子メールアドレス。 データタイプ : TEXT ルール : 255 字以内 例 : janedoe@bigcorp.com

Agent_Notifications

Agent_Notifications 表には、送信されたエージェント通知のリストが入っています。通知がユーザに送られると、この表にそのレコードが追加されます。ユーザが通知設定 Web ページで通知をクリアすると、この表からそのレコードが削除されます。

表 25 Agent_Notifications 表

フィールド	説明
ID_Notification	内容 : 通知レコードを識別する一意の ID。 データタイプ : NUMBER (LONG) 例 : 0045
ID_Subscription	内容 : 購読レコードを識別する一意の ID。 データタイプ : NUMBER (LONG) 例 : 0016
Date_Notified	内容 : 通知が送られた日付と時刻。 データタイプ : Date/Time 例 : 1997 年 12 月 1 日、10:00:00 PM
User_Name	内容 : 通知が送られた先の NT ユーザ名。 データタイプ : TEXT ルール : 50 字以内 例 : Barbara Flint
Def_Text	内容 : 通知とともに送られるデフォルトテキスト。Agent_Definition 表で定義されているものです。 データタイプ : MEMO 例 : 1,000 個あたりの欠陥数が許容値を 5 パーセント以上超えています。
User_Text	内容 : 通知とともに送られるユーザ定義のテキスト。ユーザごとに Subscriptions 表で定義されているものです。 データタイプ : MEMO 例 : この通知を受け取ったら、アカウントスーパーバイザに連絡すること。

Report_Links

Report_Links 表には、通知に含まれる個々のレポートへのリンクで使用するリンクテキストが入っています。リンク先レポートへのアクセスに必要な URL もこの表に入っています。エージェントの定義でエージェントとレポートの関連付けが定義されると、この表にそのレコードが追加されます。

表 26 Report_Links 表

フィールド	説明
ID_Report_Link	内容 ：この表内のレコードを識別する一意の ID。 データタイプ ：NUMBER (LONG) 例 ：8225
ID_Notification	内容 ：この表のレコードが関連付けられている通知レコードを識別する一意の ID。 データタイプ ：NUMBER (LONG) 例 ：8608
URL	内容 ：ユーザがレポートのリンクをクリックしたときに開くレポートの URL。 データタイプ ：MEMO 例 ：\HSPIDER\spider.dll?report& enterprise&demo
Link_Text	内容 ：通知に表示されるレポートへのリンクのリンクテキスト。 データタイプ ：TEXT ルール ：255 字以内 例 ：欠陥製品レポート

Retry_Actions

Retry_Actions 表には、正常に完了しなかったエージェント通知のリストが入っています。通知ができないと、この表にそのレコードが追加されます。Agent Monitor プログラムが定期的に Retry_Actions 表の中をスキャンして、送信できなかった通知の再送信を試みます。通知ができると、そのレコードはこの表から削除されます。

表 27 Retry_Actions 表

フィールド	説明
ID_Retry_Action	内容 ：通知レコードを識別する一意の ID。 データタイプ ：NUMBER (LONG) 例 ：0045
ID_Subscription	内容 ：購読レコードを識別する一意の ID。 データタイプ ：NUMBER (LONG) 例 ：0016

フィールド	説明
ID_ActionAvailable	<p>内容：エージェントに関連付けられたアクションを識別する一意の ID。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：0016</p>
Data_Text	<p>内容：購読ユーザが電子メール通知を選択した場合に指定する電子メールアドレス。</p> <p>データタイプ：TEXT</p> <p>ルール：255 字以内</p> <p>例：janetmatthews@bigcorp.com</p>
Def_Text	<p>内容：通知とともに送られるデフォルトテキスト。Agent_Definition 表で定義されているものです。</p> <p>データタイプ：MEMO</p> <p>例：1,000 個あたりの欠陥数が許容値を 5 パーセント以上超えています。</p>
User_Text	<p>内容：通知とともに送られるユーザ定義のテキスト。ユーザごとに Subscriptions 表で定義されているものです。</p> <p>データタイプ：MEMO</p> <p>例：この通知を受け取ったら、アカウントスーパーバイザに連絡すること。</p>
Optional_Text	このリリースでは未使用。
Date_Attempted	<p>内容：その通知を送信する試みが行われた最新の日付。</p> <p>データタイプ：MEMO</p> <p>例：01/03/97</p>
Num_Retries	<p>内容：通知が試みられた回数。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：3</p>

Retry_Links

Retry_Links 表には、通知に含まれる個々のレポートへのリンクで使用するリンクテキストが入っています。リンク先レポートへのアクセスに必要な URL もこの表に入っています。通知ができないと、この表にそのレコードが追加されます。その通知の再実行が正常に行われると、そのレコードがこの表から削除されます。詳しくは、[92 ページの「Retry_Actions」](#)を参照してください。

表 28 Retry_Links 表

フィールド	説明
ID_Retry_Link	<p>内容：この表内のレコードを識別する一意の ID。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：8056</p>

ID_Retry_ Action	<p>内容：このレコードに関連付けられている、送信できなかった通知のレコードを識別する一意の ID。</p> <p>データタイプ：NUMBER (LONG)</p> <p>例：8608</p>
URL	<p>内容：ユーザがレポートのリンクをクリックしたときに開くレポートの URL。</p> <p>データタイプ：MEMO</p> <p>例：\HSPIDER\spider.dll?report& enterprise&demo</p>
Link_Text	<p>内容：通知に表示されるレポートへのリンクのリンクテキスト。</p> <p>データタイプ：TEXT</p> <p>ルール：255 字以内</p> <p>例：欠陥製品レポート</p>

この章の内容

SpCheck 構成チェッカーの実行	95
ログファイルの使用	96
SMARTPOV のエラー処理	97
Hyperion Enterprise Reporting Web の Web ウィザードのエラーメッセージ	98
DCOMCNFG.EXE プログラムの使用	101

Hyperion Enterprise Reporting Web は、次のようにトラブルシューティングに役立ちます。

- SpCheck はシステム構成を調べるユーティリティプログラムです。
- ログファイルはシステム内で発生した問題に関する情報の追跡に使用できます。
- SMARTPOV エラーメッセージは HTML コードにリストされています。
- エラーメッセージはブラウザ、管理プログラム、Web ウィザード、またはログファイルに報告されるエラーの種類を示します。

SpCheck 構成チェッカーの実行

SpCheck は Hyperion Enterprise Reporting Web とともにインストールされる、システム構成をチェックするためのユーティリティプログラムです。SpCheck を実行して、Hyperion Enterprise Reporting Web で必要なすべてのコンポーネントがインストールされていることを確認できます。SpCheck はチェックの実行中にその結果をダイアログボックスに表示します。結果をテキストファイルに保存することもできます。

- ▶ SpCheck を実行するには、次の手順に従います。
- 1 [スタート] メニューから [ファイル名を指定して実行] を選択し、SpCheck.exe ファイルへの完全パスを入力します。SpCheck.exe は Hyperion Enterprise Reporting Web サーバがインストールされているディレクトリにあります。
 - 2 [Scan local drives (ローカルドライブをスキャン)] をファイルの検索方法として選択します。
 - 3 [チェックの実行] を選択します。
 - 4 設定ファイルを選択するように指示されたら、「SPCHECK.LST」を選択し、[OK] をクリックします。

- 5 出力ファイルを選択するように指示されたら、「SPCHECK.TXT」を選択し、[OK] をクリックします。
- 6 SpCheck が構成のチェックを終了したら、[終了] を選択します。
- 7 出力ファイルの SPCHECK.TXT を [メモ帳] などのテキストエディタで開き、チェックの結果を調べます。または、Hyperion Solutions Support の担当者にその次に何をすべきかを問い合わせます。

ログファイルの使用

Hyperion Enterprise Reporting Web ではシステム内で発生した問題に関する情報の追跡に使用する次のログファイルを作成できます。Web サーバも独自の Web サーバログファイルを作成する機能を備えています。

注： ログファイルの作成を有効にするとパフォーマンスが低下する場合があります。ログファイルは必要な場合のみに使用してください。

- SPIDER.LOG - Hyperion Enterprise Reporting Web 要求で発生するアクティビティに関する情報を記録します。
- SCRIPT.LOG - スクリプトやエージェントを実行したときに、条件スクリプトやエージェント条件内の処理に関する情報を記録します。
- CDA.LOG - Hyperion Enterprise Reporting からレポートやアプリケーションの情報がアクセスされるときに実行される処理に関する情報を記録します。
- CHART.LOG - Hyperion Enterprise Reporting Web チャートがアクセスされるときに実行される処理に関する情報を記録します。
- EMAIL.LOG - 電子メールによる通知に関する一連の詳細メッセージを記録します。

ログファイルの作成と更新に Web サーバも使用できます。

➤ ログファイルを使用するには、次の手順に従います。

- SPIDER.LOG を使用するには、次のエントリを SPIDER.INI ファイルに作成します。

[DEFAULT] SpiderLog=pathname/filename

- SCRIPT.LOG ファイルを使用するには、次のエントリを SPIDER.INI ファイルに作成します。

[DEFAULT] ScriptLog=pathname/filename

- CDA.LOG を使用するには、次のエントリを SPIDER.INI ファイルに作成します。

- [DEFAULT] CDALog=pathname/filename

- CHART.LOG を用するには、次のエントリを SPIDER.INI ファイルに作成します。

[DEFAULT] ChartLog=pathname/filename

- EMAIL.LOG を使用するには、次のエントリを SPIDER.INI ファイルに作成します。

[Agent-Email] LogFile=pathname/filename

但し、pathname/filename は作成するログファイルのパスとファイル名です。

SMARTPOV のエラー処理

動的レポートやチャートに標準のデータの視点バーを表示するだけでなく、Hyperion Enterprise Reporting Web テンプレートで次のカスタム変数を使用してデータの視点バーをカスタマイズできます。

- @SMARTPOV
- @POVDROPBAR
- @POVDROPDOWN
- @EVALUATE
- @QUERY

これらのカスタム変数でエラーが発生した場合は、その HTML コードを見てエラーの種類を調べることができます。以下はエラーのリストです。

- 存在しないテンプレートを使用した場合は、エラーメッセージとそのテンプレートが HTML コードに記載されます。
- 適切でない値を使用した場合は、エラーメッセージとその原因となったステートメントが HTML コードに記載されます。
- @SMARTPOV 変数で使用できないデータの視点を指定した場合は、[POV Edit All Bitmap (POV すべてのビットマップを編集)] が追加されます。
- エラーが 25 以上発生した場合、HTML コードにエラーメッセージが記載され、すべてのデータの視点ドロップダウンメニューが空になります。

例

```
<HTML>
<FORM %@QUERY (POST, -POV) %
<!-- ハードコードされた値を示す -->
%@POVDROPDOWN(%SYSTEM%, %APPLICATION%, CATEGORY, [CATEGORY "ACTUAL2",
"BUDGET"])%
<input type="Submit" Value="Go">
</FORM>
</HTML>
```

「Actual2」が無効なディメンションメンバである場合は、次のエラーが HTML コードに記載されます。

```
<!-- ERROR: Char 12 to 18 of :CATEGORY qACTUAL2q,qBUDGETq HEAD q<option
value=></option>q -->
```

Hyperion Enterprise Reporting Web の Web ウィザードのエラーメッセージ

次の表は、Hyperion Enterprise Reporting Web の Web ウィザードのエラーコード、エラーメッセージ、および推奨対策の一覧です。

エラーコード	エラーメッセージ	推奨対策
100	Unable to validate the Web site on the server. (サーバ上で Web サイトを確認できません。)	Web サービスが正常に実行されていることを確認します。
101	Failed to connect to FrontPage Explorer. (FrontPage エクスプローラへ接続できませんでした。)	FrontPage が正しくインストールされていることを確認します。
102	Failed to create the specified Web site. (指定された Web サイトを作成できませんでした。)	Web サイトを作成する権限があることを確認し、作成された Web サイトを保存するためのディスク領域が十分あることを確認します。1 つの Web サイトには約 1 MB の領域が必要です。追加のレポートセットを選択した場合や大きな.GIF ファイルを使用した場合は、さらに大きなディスク領域が必要です。
103	Failed to load the Web files to the current Web site. (現在の Web サイトに Web ファイルを読み込めませんでした。)	作成された Web サイトを保存するための十分なディスク領域があることを確認します。1 つの Web サイトには約 1 MB の領域が必要です。追加のレポートセットを選択した場合や大きな.GIF ファイルを使用した場合は、さらに大きなディスク領域が必要です。
104	Failed to create the Web site folder. (Web サイトのフォルダを作成できませんでした。)	<p>BORDERS フォルダは作成できません。必要に応じて次の操作を行います。</p> <p>ディスク領域が十分あることを確認します。1 つの Web サイトには約 1 MB の領域が必要です。追加のレポートセットを選択した場合や大きな.GIF ファイルを使用した場合は、さらに大きなディスク領域が必要です。</p> <p>FrontPage で共有枠線の領域を設定します。</p> <p>FrontPage で Web ページを開きます。</p> <p>[ツール] メニューの [共有枠] を選択し、すべてのページに左と下の枠線を適用します。</p>

エラーコード	エラーメッセージ	推奨対策
		<p>[ナビゲーションボタンを含む] オプションがオンになっていることを確認します。</p> <p>FrontPage エディタでナビゲーションバーを含むページを開きます。</p> <p>ナビゲーションバーをダブルクリックして、[ナビゲーションバーのプロパティ] ダイアログボックスでボタンを使うかテキストを使うかを指定します。</p>
105	Failed to open the Web site. (Web サイトを開くことができませんでした。)	<p>ディスク領域が十分あることを確認します。1 つの Web サイトには約 1 MB の領域が必要です。追加のレポートセットを選択した場合や大きな.GIF ファイルを使用した場合は、さらに大きなディスク領域が必要です。</p>
106	Failed to create the Web application object. (Web アプリケーションオブジェクトを作成できませんでした。)	<p>FrontPage が正しくインストールされ実行されていることを確認します。</p>
120	Failed to initialize FrontPage. (FrontPage の初期化ができませんでした。)	<ul style="list-style-type: none"> ● Web サービスが正常に実行されていることを確認します。 ● FrontPage が正しくインストールされ実行されていることを確認します。
121	Failed to create the Web site. (Web サイトを作成できませんでした。)	<p>FrontPage が正しくインストールされ実行されていることを確認します。</p>
122	Failed to load the Web page files. (Web ページファイルを読み込めませんでした。)	<p>Web サイトでどのファイルが欠落しているかを調べ、そのファイルを Web サイトのルートディレクトリにコピーします。レジストリでのエントリは次のようになっています。</p> <p>HKLocal Machine\Software\Microsoft\Shared Tools\Web Server Extension セクションに frontpagroot キーが入っています。この文字列はローカルマシンでの frontpagroot のパスを示します。</p> <p>Web サイトのルートディレクトリに次のファイルをコピーします。</p> <p>Frontpage のナビゲーションビューで、.HTX ファイルが</p>

エラーコード	エラーメッセージ	推奨対策
		ホームページの子になっていることを確認します。
123	Failed to load the image files. (画像ファイルを読み込めませんでした。)	<p>1. 次の.GIF ファイルを INSETPUB\WWWROOT\USRGUIDE ディレクトリから Web サイトのルートディレクトリにコピーします。</p> <ul style="list-style-type: none"> ● rank.cbp、scard.cbp、scard.rep ● rank.rep、dynscard.htx、sp_users.gif ● spider.gif、spider_help.gif、title.gif ● reptpov.gif、inv01.gif、inv02.gif ● arrow.gif、expbtn.gif、expand1.gif ● expand2.gif、expand3.gif、e_pov.gif ● dataentr.gif、colbtn.gif、bookpov.gif ● flow.gif、report.htx、replist.htx ● setlist.htx、syslist.htx、applist.htx ● dynpdf.htx、favorites.htx、dynrank.htx ● combo3.htx、sidbsid.htx、stack2.htx ● stack3.htx、stack4.htx、1u2d.htx ● default.htm、hardcopy.shtm、udfav.shtm ● usrguide.shtm、syslist.shtm、rank.shtm ● scard.shtm、repset1.shtm、repset2.shtm <p>2. 次のファイルを INSETPUB\WWWROOT\USRGUIDE ディレクトリから Web サイトの IMAGE ディレクトリにコピーします。</p> <ul style="list-style-type: none"> ● downarr.gif ● hslogo.gif ● okarr.gif

エラーコード	エラーメッセージ	推奨対策
		<ul style="list-style-type: none"> ● ribbon.jpg ● uparr.gif

DCOMCNFG.EXE プログラムの使用

NTを使用するシステムでHyperion Enterprise Reporting Webを使用するには、OLEアクセス許可を「分散COMの設定のプロパティ」ダイアログボックスで定義する必要があります。DCOM（旧称「Network OLE」）はOLEサーバのリモート実行を可能にします。MicrosoftはNTにセキュリティメカニズムを組み込んで、許可を与えられているユーザやグループのみがOLEサーバの起動やアクセスができるようにしています。システムは通常、Hyperion Enterprise Reporting Webをインストールするときに、これらの権限を定義します。エラーが発生する場合は、これらの権限を手動で設定する必要があるかもしれません。

「分散COMの設定のプロパティ」ダイアログボックスへはDCOMCNFG.EXEプログラムを実行してアクセスします。DCOMCNFG.EXEでは次の作業ができます。

- デフォルトのアクセス許可を設定する。
- Microsoft Excelのアクセス許可を設定する。
- NT 4.0内のエージェント用にHYPPROXY.CDAPROXYを構成する。

DCOMCNFG.EXEでデフォルトのアクセス許可を設定する方法

▶ デフォルトのセキュリティ許可を設定するには、次の手順に従います。

- 1 タスクバーで「スタート」ボタンを選択し、「ファイル名を指定して実行」を選択します。
- 2 DCOMCNFGと入力して、「OK」をクリックします。
- 3 「既定のセキュリティ」を選択します。
- 4 「既定のアクセス許可」の「既定値の編集」を選択します。
- 5 「追加」ボタンをクリックし、次の手順に従います。
 1. 「ユーザーの表示」を選択し、リストをスクロールして「IUSR_computername」を表示します。但し、computernameは対象コンピュータの名前です。
 2. 「IUSR_computername」を選択し、「SYSTEM」を選択して、「追加」ボタンをクリックします。
 3. 「OK」を2回クリックします。
- 6 「既定の起動アクセス許可」の「既定値の編集」を選択し、次の手順に従います。
 1. 「追加」をクリックします。

2. [ユーザーの表示] を選択し、リストをスクロールして「IUSR_computername」を表示します。但し、computername は対象コンピュータの名前です。
3. 「IUSR_computername」を選択し、[SYSTEM] を選択して、[追加] ボタンをクリックします。
4. [OK] を 2 回クリックします。
- 7 DCOMCNFG.EXE を終了するには、[OK] をクリックします。
- 8 コンピュータを再起動します。

DCOMCNFG.EXE で Microsoft Excel のアクセス許可を設定する方法

➤ Microsoft Excel のアクセス許可を設定するには、次の手順に従います。

- 1 タスクバーで [スタート] ボタンを選択し、[ファイル名を指定して実行] を選択します。
- 2 DCOMCNFG と入力して、[OK] をクリックします。
- 3 [アプリケーション] タブで Microsoft Excel ワークシートまたは Microsoft Excel 97 アプリケーションをダブルクリックするか、オプションのいずれかを選択して [プロパティ] ボタンをクリックします。

注： Microsoft Excel 5.0 を使用している場合は、これらのオプションがリストに表示されない可能性があります。その場合には、Excel 7.0 (95) または 8.0 (97) にアップグレードする必要があります。

- 4 [セキュリティ] タブを選択します。
- 5 [独自のアクセス許可を使う] を選択し、最初の [編集] ボタンを選択します。
- 6 [追加] ボタンをクリックし、Everyone をダブルクリックして、[OK] をクリックします。
- 7 [レジストリ値のアクセス許可] ダイアログボックスで [OK] をクリックします。
- 8 [独自の起動アクセス許可を使う] を選択します。
- 9 最初の [編集] ボタンをクリックして、全員をカスタム起動アクセス権に追加します。
- 10 [識別] タブを選択し、[起動したユーザー] を選択します。
- 11 コンピュータを再起動します。

注： 詳しくは、Microsoft のサポート技術情報文書 Q156223 を参照してください。

Windows NT 4.0 でエージェント用に HYPPROXY.CDAPROXY を設定する方法

▶ Windows NT 4.0 でエージェント用に HYPPROXY.CDAPROXY を設定するには、次の手順に従います。

- 1 タスクバーで [スタート] ボタンを選択し、[ファイル名を指定して実行] を選択します。
- 2 DCOMCNFG と入力して、[OK] をクリックします。
- 3 [アプリケーション] リストで、「HYPPROXY.CDAPROXY」をダブルクリックします。
- 4 [セキュリティ] タブを選択します。
- 5 [独自のアクセス許可を使う] を選択し、[編集] ボタンをクリックします。
- 6 次の手順に従います。
 1. [追加] ボタンをクリックします。
 2. エージェントオブジェクトにアクセス許可を与える NT のユーザまたはグループをダブルクリックします。

注： エージェントへのアクセスを必要とするユーザがはっきり分らない場合は、「Everyone」グループを追加します。

3. [アクセスの種類] リストで各ユーザに [アクセスの許可] を選択し、[OK] をクリックします。
- 7 [独自の起動アクセス許可を使う] を選択し、[編集] ボタンをクリックします。
- 8 次の手順に従います。
 1. [追加] ボタンをクリックします。
 2. エージェントオブジェクトにアクセス許可を与えるユーザまたはグループをダブルクリックします。
 3. 追加したユーザに対して [アクセスの種類] リストで、[起動の許可] を選択します。
 4. [OK] をクリックします。
- 9 [OK] をクリックすると、ダイアログボックスが閉じて設定が保存されます。

この章の内容

Spider.ini ファイル内のオプション	105
構文	109
Spider.ini ファイルの例	109

spider.ini ファイルには、システムにインストールされている Hyperion Enterprise Reporting Web の構成オプションが入っています。このファイルに入っているオプションを編集することによって、Hyperion Enterprise Reporting サーバ全体、そのアプリケーション、製品、またはデータベースの構成を変更できます。構成オプションの設定方法について詳しくは、『Hyperion Enterprise セットアップガイド』の「システムの設定」の章を参照してください。spider.ini ファイルは Hyperion Enterprise Reporting サーバ内の Windows ディレクトリにあります。

Hyperion Enterprise Reporting Web を使用する各ユーザは spider.ini ファイルを同時に使用します。問題を防ぐために、このファイルはすべてのユーザがログアウトしてから更新してください。例えば、システム管理者が spider.ini ファイルを開き、変更を加えて保存している間に、ログインしている別のユーザがデータの視点を変更したとします。この場合には、spider.ini にはシステム管理者が加えた変更のみが保存されます。システム管理者がファイルを開いている間に別のユーザが行ったデータ視点への変更は失われます。

spider.ini オプションの構文に関する詳しい説明は、『Hyperion Enterprise Reporting セットアップガイド』または『Hyperion Enterprise Reporting 管理者用ヘルプ』を参照してください。次の表は、spider.ini ファイルで設定できるオプションの要約です。

注： spider.ini オプションを設定した後に、Web サーバサービスを再起動する必要があります。

Spider.ini ファイル内のオプション

次の表は、spider.ini ファイルにある構成オプションの一覧です。

表 29 Spider.ini ファイル内のオプション

オプション	セクション	用途
UseLongNamesInPOVBar = Y または N	[Default]	次の 1 つを指定します。

オプション	セクション	用途
		<ul style="list-style-type: none"> ● 「Y」は長い形式のディメンション名を動的レポート、チャート、条件、および [Change Point of View (データ視点の変更)] ページで表示します。 ● 「N」は短い形式のディメンション名を動的レポート、チャート、条件、および [Change Point of View (データ視点の変更)] ページで表示します。
default_povbar_keys = 数値	[Product Name-Essbase Server \Application Name, Essbase Database] または [Product Name,Application]	動的レポート、チャートおよび条件ページのデータの視点バーに表示するディメンション数を指定します。
default_povedit_keys = 数値	[Product Name-Essbase Server \Application Name, Essbase Database] または [Product Name,Application]	[Change Point of View (データ視点の変更)] ページに表示されるディメンション数を指定します。
B_dimension = Y または N	[Product Name-Essbase Server \Application Name, Essbase Database] または [Product Name,Application]	次の 1 つを指定します。 <ul style="list-style-type: none"> ● 「Y」は指定されたディメンション（例えば勘定科目）を動的レポート、チャート、および条件ページのデータの視点バーに表示します。 ● 「N」は指定されたディメンションをデータの視点バーに表示しません。
E_dimension = Y または N	[Product Name-Essbase Server \Application Name, Essbase Database] または [Product Name,Application]	次の 1 つを指定します。 <ul style="list-style-type: none"> ● 「Y」は指定されたディメンション（例えば、勘定科目）を [Change Point of View (データ視点の変更)] ページに表示します。 ● 「N」は指定されたディメンションを [Change Point of View (データ視点の変更)] ページに表示しません。
smartpov = default	[Default]	データの視点バーを smart（スマート）データの視点バーで置き換えます。
SMPOV_dimension = [レポートスク립ト名]	[Product Name-Essbase Server \Application Name, Essbase Database] または [Product Name,Application]	smart（スマート）データの視点のディメンションのドロップダウンリストを初期化します。
locked_data_only = yes または no	[Product Name, Application]	次の 1 つを指定します。 <ul style="list-style-type: none"> ● 「yes」は保護済みのデータのみを表示します。

オプション	セクション	用途
		<ul style="list-style-type: none"> ● 「no」 は保護済みでないデータも表示します。 <p>注： このオプションは Hyperion Enterprise 専用です。</p>
Proc0 = essbase.exe	[reqstart]	<p>Essbase サーバを起動してからエージェントサーバを起動します。エージェントサーバと Essbase サーバが同じコンピュータ上にある場合に使用します。デフォルトでは、エージェントサーバが Essbase サーバの前に起動されるため、エージェントサーバは Essbase サーバを探しても見つからないので、エラーを報告します。Essbase サーバは essbase.exe です。</p> <p>このオプションは Essbase 専用です。</p>
RemotePort = ポート番号	[Agent-email]	<p>リモートのポート番号を設定します。</p> <p>デフォルトは 25 です。</p>
RemoteHost = ホスト名	[Agent-email]	<p>SMTP メールサーバの名前を設定します。</p>
Template = テンプレート名	[Agent-email]	<p>メッセージテンプレートのパスと名前を設定します。</p>
returnaddress = 返信先アドレス	[Agent-email]	<p>返信先（差出人）の電子メールアドレスを設定します。</p>
logfile = ログファイル名	[Agent-email]	<p>デバッグの目的で使用するログファイルのパスとファイル名を設定します。このパラメータはオプションです。</p>
domainname = ドメイン名	[Agent-email]	<p>ドメインのアドレスを設定します。このパラメータはオプションです。</p>
verify = 検証オプション番号	[Agent-email]	<p>次の 1 つを指定します。</p> <ul style="list-style-type: none"> ● ユーザがシステムの有効なユーザであることを検証するには 1 を指定します。 ● 検証せずにメッセージを送信するには 0 を指定します。デフォルトは 0 です。
SendTimeOut = 秒数	[Agent-email]	<p>メール送信でタイムアウトになるまでに待つ時間を秒単位で設定します。</p>
RecieveTimeOut = 秒数	[Agent-email]	<p>メール受信でタイムアウトになるまでに待つ時間を秒単位で設定します。</p>
MultiThread = 数値	[Essbase]	<p>次のマルチスレッドオプションの 1 つを指定します。</p>

オプション	セクション	用途
		<ul style="list-style-type: none"> ● 「0」 は同時にタスクを実行できるユーザ数を 1 に制限します。 ● 「-1」 は複数のユーザによるタスクの同時実行を可能にします。
PDFFile = 1 または 2	[Options]	<p>次の PDF ファイルオプションの 1 つを指定します。</p> <ul style="list-style-type: none"> ● PDF ファイルを作成してブラウザに送信するには「1」を指定します。 ● 「2」 は PDF ファイルを作成し、Java スクリプトを使ってブラウザに送信します。
SpiderLog = ログファイル名	[Default]	Hyperion Enterprise Reporting のメッセージをそのログファイルに記録します。
CDALog = ログファイル名	[Default]	報告されるメッセージをそのログファイルに記録します。
ChartLog = ログファイル名	[Default]	チャートファイルメッセージをそのログファイルに記録します。
ScriptLog = ログファイル名	[Default]	スクリプトファイルメッセージをそのログファイルに記録します。
CacheReports = Y または N	[Default]	<p>次のオプションの 1 つを指定します。</p> <ul style="list-style-type: none"> ● 「Y」 はレポートのキャッシュメモリを作成します。 ● 「N」 はレポートのキャッシュを作成しません。 <p>キャッシュを作成するとレポートへのアクセス速度を向上できます。</p>
Product Name-Essbase Server \\Application Name, Essbase Database = ユーザ数	[Cached]	特定のアプリケーションへのログインプロファイルがキャッシュメモリに保存されるユーザ数を指定します。これによってアクセス速度が向上します。
ImageCleanupInterval = 時間数	[Default]	チャートと.PDF ファイルをシステムが削除するまでの時間を時間単位で指定します。
@curmonth = 現在の月	[Default]	<p>@curmonth 変数に関連付ける現在の月を設定します。@curmonth は期間ディメンションの現在の月を指定します。</p> <p>このパラメータはオプションです。このパラメータを設定しないと、@curmonth は Windows NT の日付情報を使用します。これを設定しないと、アプリケーションで使用する月の表記は、コンピュータが使用</p>

オプション	セクション	用途
		する長い形式の月の表記に正確に一致している必要があります。
MaxListEntries	[Default]	[Edit POV (データの視点編集)] Web ページに表示されるディメンションメンバの数を設定します。これが設定されていない場合は、デフォルト値の 100 が使用されます。

構文

- spider.ini ファイルのオプションの構文では大文字小文字の区別をしません。
- オプションには余分なスペースを使用しないでください。例えば、次のオプションにはスペースを使用しません。
ImageCleanupInterval=2
- コメントの先頭にはセミコロンを使用します。例えば、次のような行がコメントになります。
;これはコメント

Spider.ini ファイルの例

```
[DEFAULT]
UseLongNamesInPOVBar=y
;ログファイルのオプション - newlog.txtに
;すべてのロギング情報を記録
SpiderLog=d:\testdata\logfiles\newlog.txt
CDALog=d:\testdata\logfiles\newlog.txt
ChartLog=d:\testdata\logfiles\newlog.txt
ScriptLog=d:\testdata\logfiles\newlog.txt
CacheReports=y
ImagecleanupInterval=24
@curmonth=january
[Options]
pdffile=2
[reqstart]
;Agent Monitorを起動する前にEssbaseサーバを起動する
proc0=essbase.exe
[agent-email]
RemotePort = 25
RemoteHost = mail.server.com
Template = d:\inetpub\spider\emailtmp.txt
returnaddress = john_doe@your_company.com
logfile = d:\testdata\logfiles\newlog.txt
domainname = maildomain
verify = 1
SendTimeOut = 60
RecieveTimeOut = 60
[CACHED]
Enterprise,Enterprise_application1 = 10
ESSBASE-Eserver\ESSBASE_APPLICATION2,EDataBase = 25
```

```
[Essbase]
multithread=-1
[ENTERPRISE, ENTERPRISE_APPLICATION1]
;データの視点オプションを設定
default_povbar_keys = 4
B_field=Y
B_product=Y
; 勘定科目ディメンションをデータの視点バーに表示しない
B_account =N
B_entity=Y
default_povedit_keys = 2
E_account =Y
E_entity=Y
; 最終データのみが表示されるようにする
locked_data_only=yes
```

データベースに対するデータの取得および書込みを可能にする Oracle Essbase をインストールして構成できます。Web ユーザーは、オラクル社の Hyperion(R) Enterprise(R) Web クライアントおよびオラクル社の Hyperion(R) Enterprise(R) Reporting Web クライアントを使用して、メッセージの**送受信**を行うことができます。

索引

記号

.PDF ファイル

PRINTREP 関数, PDF ファイルへの出力, 35

@curmonth, SPIDER.INI ファイルでの設定, 108

@curmonth, 関数の構文, 15, 20, 22, 27, 29, 32

A - Z

Action_Definition 表, 84

Action_Subscriptions 表, 90

AddLink メソッド, 64

Adobe Acrobat, 出力関数, 35

Agent_Definition_Security 表, 88

Agent_Definition 表, 83

Agent_Notification 表, 91

Agent_Schedule 表, 87

AgentDescription プロパティ, 77

AgentDesc プロパティ, 60

AGENTFORM 関数の概要, 10

AgentID プロパティ, 62

AGENTLIST 関数

概要, 11

例, 12

AgentName プロパティ, 61, 77

AGENTSSSENT 関数

概要, 12

例, 12

APPLIST ライブラリ関数

概要, 41

例, 42

Available_Actions 表, 89

CHANGEID 関数

概要, 13

例, 14

CHART 関数

概要, 14

チャートの種類を指定するパラメータ, 17

例, 19

CRITERIA 関数

概要, 20

例, 20

DataObjs 表, 86

DataText プロパティ, 74

DCOMCNFG.EXE プログラム

エージェント用に HYPPROXY.CDAPROXY

を設定する, 103

概要, 101

デフォルトのアクセス許可の設定, 101

DefMsg プロパティ, 63

Distributed COM 設定ダイアログボックス, 101

DLL ファイル, 7

EDITPOV 関数

概要, 24

例, 25

ErrorText プロパティ, 76

EXCELREP 関数

概要, 26

スプレッドシートの書式, 27

例, 28

GetAgent メソッド, 65

GetAppName メソッド, 50

GetCellDouble メソッド, 51

GetCellPovString メソッド, 52

GetCellStatus メソッド, 54

GetCellString メソッド, 53

GetName メソッド, 55, 57

GetNumReports メソッド, 58

GetNumRows メソッド, 56

GetReportDisp メソッド, 59

GetSystemName メソッド, 56

MessageText プロパティ, 74

Microsoft Excel アクセス許可の設定, 102

OptionalText プロパティ, 75

PRINTREP 関数, 例, 36

REPCHART 関数

概要, 28

例, 30

Report_Links 表, 92
 REPORT 関数
 概要, 31
 例, 34
 REPSETLIST ライブラリ関数
 概要, 45
 Retry_Actions 表, 92
 Retry_Links 表, 93
 Scheduled_ObjPOV 表, 87
 SETLIST ライブラリ関数
 概要, 43
 例, 44, 46
 SMARTPOV のエラー処理, 97
 SpCheck 構成チェッカー, 95
 spider.ini ファイルのオプション
 一覧, 105
 構文, 109
 Sub メソッド, 67
 SYSLIST ライブラリ関数, 例, 40
 URLParamNum メソッド, 72
 URLParam メソッド, 72
 UserMessageText プロパティ, 75
 Web ウィザードのエラー, 98
 Web ページ
 アプリケーションの表示, 41
 チャート, 14
 フレーム分割, 28
 レポートセット一覧の表示, 43
 レポートの表示, 31

あ行

アクション
 開始する, 66
 表, 89
 アクション情報オブジェクト
 AgentDescription プロパティ, 77
 DataText プロパティ, 74
 ErrorText プロパティ, 76
 GetLinkText メソッド, 78
 GetLinkURL メソッド, 79
 GetNumLinks メソッド, 78
 MessageText プロパティ, 74
 OptionalText プロパティ, 75
 UserMessageText プロパティ, 75
 アクセス許可, DCOMCNFG.EXE プログラムで
 設定, 101

アプリケーション, 一覧の動的な生成, 41
 エラー
 SMARTPOV, 97
 Web ウィザード, 98
 エージェント
 AGENTFORM 関数, 10
 AGENTLIST 関数, 11
 AGENTSSSENT, 12
 GetNumReports メソッド, 58
 GetReportDisp メソッド, 59
 アクション情報オブジェクト, 73
 一覧表示, 11
 エージェントオブジェクト, 58
 エージェント評価オブジェクト, 63
 オブジェクト, 58, 63
 購読, 10
 参照する表, 81
 通知の表示, 12
 表間の関係, 82
 エージェントオブジェクト
 AgentDesc プロパティ, 60
 AgentID プロパティ, 62
 DefMsg プロパティ, 63
 概要, 58
 エージェント評価オブジェクト
 AddLink メソッド, 64
 ClearLinks メソッド, 65
 GetAgent メソッド, 65
 HTMLEncode メソッド, 67
 Notify メソッド, 66
 URLEncode メソッド, 67
 概要, 63
 オブジェクト
 アクション情報オブジェクト, 73
 エージェント, 58
 エージェント評価, 63
 条件, 68
 テンプレート, 67
 レポートオブジェクト, 49

か行

カスタムオブジェクト, 70
 関数
 AGENTFORM, 10
 AGENTLIST, 11, 12
 AGENTSSSENT, 12
 APPLIST, 41, 42

CHANGEID, 13, 14
 CHART, 14, 19
 CRITERIA, 20
 EDITPOV, 24, 25
 EXCELREP, 26, 28
 Hyperion Essbase の構文, 10
 PRINTREP, 35, 36
 REPCHART, 28, 30
 REPORT, 31, 34
 REPSETLIST, 45
 SETLIST, 43, 44, 46
 SYSLIST, 40

購読

エージェント, 10
 エージェントの表, 89

構文

AGENTFORM 関数, 10
 AGENTLIST 関数, 11
 AGENTSSENT 関数, 12
 APPLIST 関数, 41
 CHANGEID 関数, 13
 CHART 関数, 14
 CRITERIA 関数, 20
 EDITPOV 関数, 24
 REPCHART 関数, 28
 REPORT 関数, 31
 SETLIST ライブラリ関数, 43
 関数の呼び出し, 7
 スプレッドシート, 実行, 26

さ行

財務エージェント

AGENTFORM 関数, 10
 AGENTLIST 関数, 11
 AGENTSSENT 関数, 12
 GetNumReports メソッド, 58
 GetReportDisp, 59
 一覧表示, 11
 エージェントオブジェクト, 58
 エージェント評価オブジェクト, 63
 オブジェクト, 58, 63
 購読, 10
 参照する表, 81
 通知の表示, 12
 表間の関係, 82

出力, PDF 出力用の PRINTREP 関数, 35

書式を指定するパラメータ, EXCELREP 関数, 27

条件オブジェクト

GetTemplate メソッド, 69
 HTMLEncode メソッド, 71
 SetErrorMsg メソッド, 70
 SetErrorNum メソッド, 69
 URLEncode メソッド, 71
 URLParamNum メソッド, 72
 URLParam メソッド, 72
 概要, 68

条件に基づいたページの表示, 20

スプレッドシート

開始する, 28
 パラメータ, 27

セキュリティ

Agent_Definition 表, 88
 アクセス許可の設定, 101

セット一覧, 動的な生成, 43

た行

ダイナミックリンクライブラリファイル, 7

チャート

種類, 17
 表示, 14
 並列表示, 28

チャートの種類を指定するパラメータ, 17

テンプレートメソッド, 67

データの視点

カスタマイズ, 97

データの視点, 変更, 24

データベース表, 81

トラブルシューティング

概要, 95

は行

パラメータ

CHART 関数, 17
 スプレッドシートの書式, 27

表

Action_Definition, 84
 Action_Subscriptions, 90
 Agent_Definition, 83
 Agent_Definition_Security, 88
 Agent_Notification, 91
 Agent_Schedule, 87
 Available_Actions, 89

DataObjs, 86
 Report_Links, 92
 Retry_Actions, 92
 Retry_Links, 93
 Scheduled_ObjPOV, 87
 Subscriptions, 89
 関係, 82
 データベース, 81
 含まれている, 81

表示基準

GetCellDouble メソッド, 51
 GetCellString メソッド, 53
 GetNumColumns メソッド, 56
 GetNumRows メソッド, 56
 Sub メソッド, 67
 オブジェクトを参照する, 49
 条件オブジェクト, 68
 条件を使用したページの表示, 20
 テンプレートメソッド, 67
 メソッド, 49, 67
 レポートオブジェクト, 49

プロパティ

AgentDesc, 60
 AgentDescription, 77
 AgentID, 62
 DataText, 74
 DefMsg, 63
 ErrorText, 76
 MessageText, 74
 OptionalText, 75
 UserMessageText, 75
 UserName, 76

ま行

メソッド

AddLink, 64
 ClearLinks, 65
 GetAgent, 65
 GetAppName, 50
 GetCellDouble, 51
 GetCellPovString, 52
 GetCellStatus, 54
 GetCellString, 53
 GetLinkText, 78
 GetLinkURL, 79
 GetName, 55, 57
 GetNumColumns, 56

GetNumLinks, 78
 GetNumReports, 58
 GetNumRows, 56
 GetReportDisp, 59
 GetSystemName, 56
 GetTemplate, 69
 GetTitle, 57
 HTMLEncode, 67, 71
 Notify, 66
 SetErrorMsg, 70
 SetErrorNum, 69
 Sub, 67
 URLEncode, 67, 71
 エージェントオブジェクトの, 58
 エージェント評価オブジェクトの, 63
 テンプレートオブジェクトの, 67
 レポートオブジェクトの, 49

ら行

ライブラリ, ダイナミックリンク, 7

ライブラリ関数

APPLIST, 41
 REPSETLIST, 45
 SETLIST, 43
 クイックリファレンス, 39

例

AGENTLIST 関数, 12
 AGENTSSSENT 関数, 12
 APPLIST ライブラリ関数, 42
 CHANGEID 関数, 14
 CHART 関数, 19
 CRITERIA 関数, 20
 EDITPOV 関数, 25
 EXCELREP 関数, 28
 PRINTREP 関数, 36
 REPCHART 関数, 30
 REPORT 関数, 34
 SETLIST ライブラリ関数, 44, 46
 SYSLIST ライブラリ関数, 40

レポート

REPORT 関数, 31
 一覧の生成, 45
 生成, 31
 並列表示, 28
 並列表示の生成, 28

ログオン, アプリケーションのログイン情報の
 変更, 13

ログファイル, 使用, [96](#)

