

HP OpenView Internet Services

Web Transaction Recorder ガイド



Manufacturing Part Number: J4511-90009

2006年4月

© Copyright 2004 - 2006 Hewlett-Packard Development Company, L.P.

ご注意

1. 本書に記載した内容は、予告なしに変更することがあります。
2. 当社は、本書に関して特定目的の市場性と適合性に対する保証を含む一切の保証をいたしかねます。
3. 当社は、本書の記載事項の誤り、またはマテリアルの提供、性能、使用により発生した直接損害、間接損害、特別損害、付随的損害または結果損害については責任を負いかねますのでご了承ください。
4. 本製品パッケージとして提供した本書、CD-ROM などの媒体は本製品用だけにお使いください。プログラムをコピーする場合はバックアップ用だけにしてください。プログラムをそのままの形で、あるいは変更を加えて第三者に販売することは固く禁じられています。

本書には著作権によって保護される内容が含まれています。本書の内容の一部または全部を著作者の許諾なしに複製、改変、および翻訳することは、著作権法下での許可事項を除き、禁止されています。

All rights reserved.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

© Copyright 2004 - 2006 Hewlett-Packard Development Company, L.P.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

Trademark Notices

Java™ は、Sun Microsystems, Inc. の商標です。

Microsoft Windows®、Windows NT®、MS Windows®、Windows 2000®、MS-DOS®、XP® は、米国における Microsoft Corporation の登録商標です。

Netscape™ と Netscape Navigator™ は、米国における Netscape Communications Corporation の商標です。

UNIX® は、The Open Group の登録商標です。

Adobe® と Acrobat® は、Adobe Inc. の登録商標です。

その他の製品名は、登録商標を所有する各社に帰属します。

原典

本書は『*HP OpenView Internet Services Web Transaction Recorder Guide (HP Part No. J4511-90007)*』を翻訳したものです。

サポート

HP OpenView Web サイトをご参照ください。

<http://www.managementsoftware.hp.com/>

連絡先の情報および HP OpenView の提供する製品、サービス、サポートの詳細が掲載されています。

また、直接サポート Web サイトにアクセスすることもできます。

<http://support.openview.hp.com/>

HP OpenView オンラインソフトウェアサポートには、お客様がご自身で問題を解決するための機能が備わっています。ビジネスを遂行するのに必要なインタラクティブなテクニカルサポートツールに簡単に効率良くアクセスすることができます。お客様は以下の機能を利用することができます。

- 関心をお持ちの情報の検索
- サポート依頼の送信と進行状況のチェック
- サポート連絡先の管理
- HP サポート担当の一覧
- 利用可能なサービスについての調査
- 他のソフトウェアカスタマとのディスカッション
- ソフトウェアトレーニングの調査と申し込み

大部分のサポートでは、HP Passport ユーザーとして登録して、ログインする必要があります。また、多くのサポートには、サポート契約を結んでいることが必要になります。

アクセスレベルの詳細は、以下の URL を参照してください。

http://support.openview.hp.com/access_level.jsp

HP Passport ID を登録するには、以下の URL にアクセスしてください。

<https://passport.hp.com/hpp2/newuser.do>

目次

第 1 章	OVIS Web Transaction Recorder	9
	はじめに	10
	Web ページの基本的な情報	11
	HTML	11
	JavaScript とプラグインの例	13
	フレーム	14
	HTTP	15
	HTTPS と SSL	15
	認証とデジタル証明書	15
	プロキシサーバー	16
	キャッシング	16
	URL	17
	Cookie	17
	Web トランザクションの記録モード	20
	使用するモードを判断するためのヒント	22
	Web Transaction Recorder の動作	24
	記録	24
	IE モードの記録	24
	URL モードの記録	24
	記録の結果	25
	記録の簡単な例	25
	再生	29
	URL モードの再生	29

IE モードの再生	29
HTTP_TRANS プローブの動作	31
第 2 章 Web Transaction Recorder の使用方法	33
HTTP_TRANS プローブの設定	34
Web トランザクションを記録する手順	36
基本的な手順	37
グローバルなトランザクションプロパティの設定	39
記録のオプション	41
再生のオプション	43
トレースレベル	45
プロキシの設定	45
基本的なトランザクションステップのプロパティ	46
トランザクションステップの詳細プロパティ	49
詳細な使用方法	50
Internet Explorer のセキュリティダイアログボックス	50
動的な URL の代替処理	53
フォームデータ入力用の外部スクリプト (IE モード)	58
可用性チェック用外部スクリプト (IE モード)	59
動的なフレーム名	61
プローブのログオンアカウントの変更	62
Cookie の設定	62
セッション Cookie の読み取りと保存	63
認証	63
ブラウザプラグイン	63
複雑なページのリダイレクション	64
重複する入力変数	65
プローブの結果に影響を与えるレジストリの設定 (IE モード)	66
HTML ダイアログとプルダウンメニューの内部処理	68
位置独立ナビゲーションポイント	77
トランザクションスクリプトのリファレンス	78
IE モード	78
URL/ナビゲーションポイントモード	96

Web Transaction Recorder についてのヒント	102
第 3 章	
トラブルシューティング	105
記録と再生に関する問題点	106
記録に関する問題点	106
IE モードでトランザクションステップが記録されない (メニュー、プラグ イン)	106
IE モードでのポップアップウィンドウ	107
フォームの値が記録されない	107
再生に関する問題点	109
トランザクションがタイムアウトする	109
HTTP_TRANS プロープがリソース使用率に与える影響	110
動的なフレーム名 (エラー: フレームが見つかりません。)	111
初期化エラーダイアログボックス	112
応答時間とシステム負荷の増加	113
Java アプレットが異常終了または不正な動作を引き起こす	113
再生時にトランザクションステップの終了が検出できない	113
再生 / ステップ時の空のメッセージボックス	114
新しいウィンドウが正しく読み込まれない	114
選択ボックスの内容が再生のたびに变化する	115
IE モードのレジストリ設定	116
動的な ID フィールド	117
Active-X コントロール / Java アプレット対応の Web サイトにおけるスクリ プト	117
Windows 2000 での再生時の制約	118
HTTP_TRANS プロープの TIP の設定	118
HTTP_TRANS の注釈付きトレースファイル	119

OVIS Web Transaction Recorder

HTTP_TRANS プロンプは、カタログの検索、ログイン/ログアウト、ショッピングカートのような複数のステップで構成される Web トランザクションを監視するために使います。HTTP_TRANS の監視対象サービスを作成すると、Web Transaction Recorder が自動的に起動されます。Web Transaction Recorder は指定されたユーザーアクションを追跡し、記録します。プロンプがこれを定期的に再生し、典型的なエンドユーザーの動作をシミュレートしたり、サービスの重要な可用性と応答時間のデータを収集します。

Web Transaction Recorder から HTTP_TRANS プロンプを設定すると、間違いが起これにくく設定作業が短縮されます。Web Transaction Recorder を使用すると、多くの URL やページ参照を手動で入力する代わりに、典型的なエンドユーザー トランザクションの各ステップを実行することによって、アクセスしたページのアクションやシーケンス、およびナビゲートしたリンクを自動的に収集していくことができます。その後、トランザクションをテストして検証し、記録したトランザクションステップを変更することができます。



Web Transaction Recorder は、Internet Explorer 5.5 以上が必要です。最新のサービスパックを適用した Internet Explorer 6.0 を使用することをお勧めします。これを使用すると、HTTP ステータスコードをインターセプトしてログに残すことができます。

1 つのサービスグループで、1 つの HTTP_TRANS 監視対象サービスのみ使用できます。最大 100 のステップを記録することができます。

はじめに

Web ページまたは Web サイトの構造と含まれている要素を理解しておく、Web トランザクションを記録する際に役に立ちます。構造と要素によっては、トランザクションの記録方法や Web Transaction Recorder で使用するオプションが異なる場合があります。正しい記録方法の選択やオプションを設定すると、HTTP_TRANS プローブの記録を再生することができます。

Web ページの基本的な情報

以下の Web ページの基本的な情報を説明します。

- HTML
- JavaScript とプラグインの例
- フレーム
- HTTP
- HTTPS と SSL
- 認証とデジタル証明書
- プロキシサーバー
- キャッシング
- URL
- Cookie

HTML

HTML(ハイパーテキスト マークアップ言語)は、SGML(標準汎用マークアップ言語)より派生した言語です。HTMLは、Web ドキュメントの構造を定義するために使用されます。HTMLでは、山形カッコ(<>)で囲んだタグ(キーワードまたは要素)でコンテンツタイプを定義します。マークアップとは、ドキュメントにタグや特定の文字を挿入する記述方法です。タグ自身は表示されず、コンテンツの一部になることはありませんが、ドキュメントの機能を拡張する働きをします。マークアップには、フォーマットタグ、構造タグ、ハイパーテキストリンク機能を含めることができます。タグは属性またはオプションを持つことができます。HTML タグの例は、以下のとおりです。

```
<em>Hello</em>

<table border>



<h1>Hello, <em>world!</em></h1>
```

シンプルな Web ページは、2つのセクション `head` と `body` で構成されます (次に示す例を参照)。 `title` はブラウザウィンドウのタイトルバーに表示されます。
.html ファイルのマークアップは、`<html>` から始まり、`</html>` で終了します。

```
<html>
<head>
<title>A Simple Web Page</title>
</head>
<body>Hello World!</body>
</html>
```

Web ページには以下の要素を含めることができます。

- **カスケーディングスタイルシート (CSS):** CSS は、ドキュメントスタイル (すなわち、ドキュメント内のフォント、色などの具体的な表示方法) を定義します。
- **JavaScript と Visual Basic Script:** 複雑な Web ページでは、Web ブラウザ内で動的な処理を行うスクリプトを使用することがあります (スクリプトはサーバーではなく Web ブラウザで実行されます)。
- **Java アプレットやその他のプラグイン:** アプレットとプラグイン (Macromedia FLASH など) は、ブラウザによって読み込まれるプログラムです。通常は HTML とは無関係の、別個に実行される独立したプログラムです。

JavaScript とプラグインの例

JavaScript は、サーバーとのやり取りは行わずに、直接ブラウザで実行されます。

JavaScript のコードは、HTML `<SCRIPT>` タグで囲みます。



```
<html><body>
<SCRIPT LANGUAGE="JavaScript">
  function f1() {
    alert("Hello");
    return false;
  }
</SCRIPT>
<a href=alink.html onclick="f1()">Link</a>
<INPUT TYPE=button VALUE=Button
  onclick="f1()">
</body></html>
```

上記の例では、スクリプト内の関数呼び出しによって、メッセージダイアログボックスが表示されます。この `f1` 関数は、HTML `OnClick` ハンドラーのリンク (``) と `INPUT` 要素のボタンから呼び出されます。リンクまたはボタンをクリックすると、`script` セクションで定義されたダイアログボックスが表示されます。

`mousemove` や `focus` などのその他のハンドラーも、JavaScript に関連付けることができます。

プラグインを使う場合は、HTML `<object>` タグは使用するコンポーネントを定義します。

以下の例では、Macromedia FLASH プラグインの使い方を示しています。例の後に、各 HTML 要素の定義を示しています。

```
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/
  swflash.cab#version=5,0,0,0" width="500" height="200" align="top">
<param name="movie" value="/images/tbird_new.swf">
```

```
<param name="SCALE" value="noborder">
<embed src="/images/tbird_new.swf" width="500" height="200" align="top"
  pluginspage="http://www.macromedia.com/go/getflashplayer"
  type="application/x-shockwave-flash" scale="noborder"></embed>
</object>
```

codebase パラメータでは、コンポーネントをダウンロードする場所を定義しています。

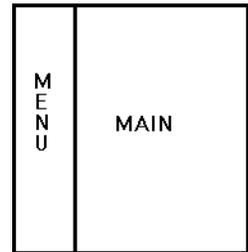
param パラメータは、パラメータをコンポーネントに引き渡します。

<embed> タグは、**Macromedia FLASH** で表示されるマルチメディアコンテンツを定義しています。この例では、このタグの中の SRC 属性が、コンテンツ、タイプ、および (コンテンツの表示に使用する Shockwave コンポーネントを定義している) プラグイン Web ページを定義しています。

フレーム

フレームは、ブラウザウィンドウを複数のビューに分割するための方法として、Web ページの中で使われます。各ビューでは異なるドキュメントが表示されます。1つのビューでリンクをクリックすると、別のビューに新しいドキュメントが表示されるという方法が頻繁に使われます。

たとえば、多くの Web ページでは、2 ペインのデザインが使われています。ナビゲーションペイン (例では MENU) でリンクをクリックすると、メインペインに異なるドキュメントが読み込まれます。



例:

```
<FRAMESET COLS="216,*">
  <FRAME SRC="menu.html" NAME="MENU">
  <FRAME SRC="main.html" NAME="MAIN">
</FRAMESET>
```

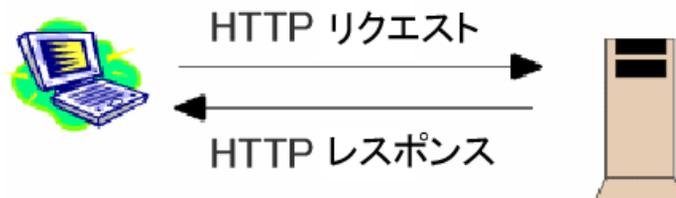
ブラウザは、frameset の定義に従ってドキュメントを読み込みます。frameset タグは、Web ページのレイアウトを定義する属性を持っています。ブラウザは SRC 属性で定義されているフレームを解析して読み込み、frameset 属性に従ってウィンドウを分割します。

HTTP

ハイパーテキスト転送プロトコル (HTTP) は、TCP 上のアプリケーションプロトコルです。HTTP は、インターネット上でブラウザで表示するための、Web オブジェクトの要求、送信、受信に使用されます。HTTP の現在のバージョンは 1.1 です。

ブラウザに URL を入力すると、指定した Web ページの送信を要求する HTTP コマンドが Web サーバーに送信されます。

HTTP は、クライアント/サーバーモデルで動作します。クライアント側では、ブラウザが HTTP リクエストを使って、Web オブジェクトの要求、受信、表示を行います。Web サーバーは、要求への応答として、HTTP レスポンスを使ってオブジェクトを送信します。



HTTPS と SSL

HTTPS は、SSL(セキュアソケットレイヤー)を付加した HTTP です。HTTPS プロトコルは、データを Web 上でセキュアに送信する必要がある場合に使われます。HTTPS は個々のメッセージをセキュアに送信し、SSL は 2 つのコンピュータ間にセキュアな接続を確立します。SSL 接続が必要な Web ページは、規約では、http: ではなく、https: で始めることになっています。

認証とデジタル証明書

認証は、利用する人や物が本人であるかの正当性を検証するプロセスです。認証では、単純に、ログオンのユーザー名とパスワードを使うこともできます。ただし、インターネットビジネスやその他の多くのトランザクションでは、さらに厳密な認証プロセスを必要としています。そのようなトランザクションでは、デジタル証明書が使われます。デジタル証明書は、認証局 (CA) から発行されて確認され、認証を行うために使われます。

Web サーバーが SSL 接続を確立するには、SSL 証明書が必要です。SSL 証明書は Web サーバーを認証し、暗号化のキーを提供します。SSL 証明書は、ブラウザが承認している CA から発行される必要があります。CA は、Internet Explorer や Netscape のようなブラウザにインストールされたルート CA 証明書により認識されます。

プロキシサーバー

プロキシサーバーは、クライアントアプリケーション (たとえば、Web ブラウザ) とサーバーとの間にあるシステムです。プロキシは実際のサーバーへのすべてのリクエストをインターセプトして、リクエストに応えられるか調べます。応えられない場合には、リクエストを実際のサーバーへ転送します。プロキシサーバーはパフォーマンスを改善したり、リクエストをフィルタリングするために使われます。

プロキシはしばしばイントラネット (またはプライベートネットワーク) を保護するためにも使われ、その場合には、インターネットに接続するのはプロキシだけになります。ファイアウォールは、プライベートネットワーク内への (または外への) 承認されていないアクセスを防ぐように設計されたシステムです。ファイアウォール技術の 1 つでは、ネットワークを出入りするすべてのメッセージをインターセプトするためにプロキシサーバーが使われます。プロキシサーバーは、本当のネットワークアドレスを隠すことができます。

キャッシング

キャッシュは情報が一時的に格納される場所です。Web ページを検索しているときに自動的に取り込まれたファイルは、ユーザーのハードディスクのブラウザのディレクトリの下にあるキャッシュサブディレクトリに格納されます。ユーザーが最近見たページに戻る場合、ブラウザはそれらのファイルを元のサーバーからではなくこのキャッシュから取得するので、時間が節約され、ネットワークの余分なトラフィックも節約されます。特定のブラウザに合わせて、通常、キャッシュのサイズは変更することができます。

コンテンツ配信サービスは、Web サイトのページを地理的に分散している複数のサーバーにコピーしておくサービスです。そしてページが要求されると、要求を動的に判断して、ユーザーに最も近いサーバーからページコンテンツを配信します。コンテンツ配信サービスによって、高速な配信が可能になります。コンテンツ配信サービスの対象となった URL をクリックすると、コンテンツ配信ネットワークがユーザーの要求を、サイトの元のサーバーからユーザーに近いキャッシュサーバーにルーティングし直します。キャッシュサーバーは、リクエストされたコンテンツがキャッシュにあればそれを配信し、なければ元の

サーバーから取り込みます。新しいコンテンツもローカルにキャッシュされます。このプロセスは、要求した URL が異なる URL へ転送される(リダイレクションとも呼ばれます)点を除けば、配信時間が短縮されるだけで、一般的にユーザーが気付くことなく処理されます。

URL

URL (Uniform Resource Locator) は、インターネット上のドキュメントその他のオブジェクトを一意に参照する方法です。Web ページをブラウザで表示するときに、URL を使います。構文は以下のとおりです。

<プロトコル>://<ドメイン名>:<ポート番号>/<url またはパス名>

例: **http://www.hp.com/index.html**

http を使用するとき、ブラウザは以下の手順を実行して URL を表示します。

- 1 URL を解析します。
- 2 URL のドメイン名、たとえば、www.hp.com を取得します。
- 3 ドメインネームサーバー (DNS) に対して、ドメイン名から IP アドレスへの変換を依頼します。
- 4 IP アドレスを使って、サーバーに TCP 接続します。
- 5 サーバーに、HTTP で接続します。
- 6 URL のパス名の部分をサーバーに渡します。

Web プロキシの場合には、ブラウザはプロキシに対して URL 全体を送信します。HTTPS や FTP などの、HTTP 以外のプロトコルも、URL で使うことができます。

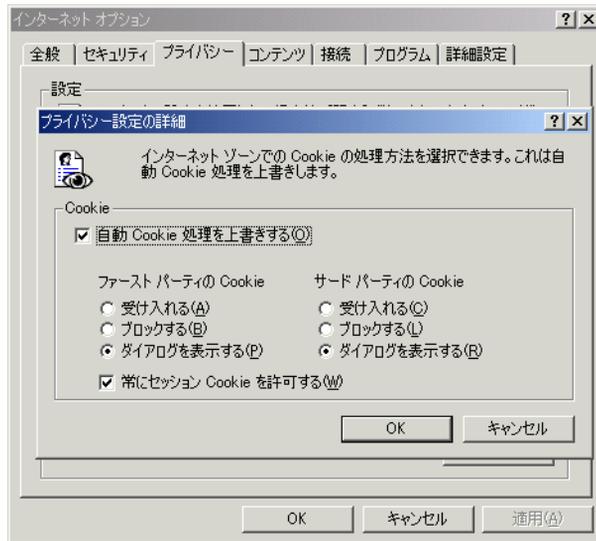
Cookie

Cookie は、サーバーがブラウザに送信する情報です。ブラウザからサーバーへの2回目以降のリクエストには、ブラウザによってヘッダー内に Cookie が(たとえば、http リクエスト + Cookie: OVIS という形式で)提示されます。サーバーは提示された Cookie をサーバーに格納している情報と照合します。

Cookie は、複数の Web トランザクションでの状態を保持するためにも使えます。これは、認証、登録ユーザーのログイン無しでの識別、ユーザー設定の記憶、ショッピングカートの状態保持などに使われます。

Cookie には、セッション Cookie と固定 Cookie の 2 種類があります。セッション Cookie は、1 回のインターネットセッション内だけで使用されます。固定 Cookie はクライアントシステムに格納され、次回以降のインターネットセッションで呼び出されます。固定 Cookie には通常、有効期限が含まれます。

送信されてきた Cookie を見るができます。Internet Explorer 6.0 以降では、[ツール] > [インターネット オプション] > [プライバシー] > [詳細設定] を選択します。[自動 Cookie 処理を上書きする] にチェックを付けて、[ファーストパーティの Cookie] と [サードパーティの Cookie] の下にある、各 [ダイアログを表示する] にチェックを付けます。



サーバーから Cookie を受信するたびに、次のようなダイアログボックスが表示されるので、ダウンロードされた Cookie を見ることができます。



Web トランザクションの記録モード

Web Transaction Recorder には、3 つの記録モードがあります。Web トランザクションの記録を始める前に、記録モード (ナビゲーションモードとも呼ばれます) を選択する必要があります。ナビゲーションモードの種類は、以下のとおりです。

- **IE モードプローブ** – IE モードで作成された HTTP_TRANS プローブが動作するのは、Windows システムだけです。このプローブは、IE のブラウザエンジン呼び出して、記録されたトランザクションを実行します。

このプローブは IE のブラウザエンジンを使うので、プローブで JavaScript やレンダリングのような機能を処理することができます。レンダリングによって Web ページが描画され、スクリプトおよび Java やその他のプラグインのような埋め込みオブジェクトが実行されます。IE モードでは、エンドユーザーの視点に近い応答時間の測定を行うことができます。

- **URL モードプローブ** – URL モードは、HTTP プローブのようなカスタムプローブを使用します。このカスタムプローブは、URL をダウンロードするだけで、IE のブラウザエンジンを使ったレンダリングは行いません。このモードは Web サイトが利用できるかを確認するには十分です。ただしユーザーは IE ブラウザを使用するので、このモードでの応答時間はユーザーが実際に感じている応答時間を反映しません。

URL モードのプローブは、プラットフォームに依存しないので、UNIX、Linux、MS Windows で使うことができます。また、URL モードは IE モードよりも少ないメモリーリソースで動作します。

- **ナビゲーションポイントモード プローブ** – ナビゲーションポイントモードは廃止されました。以前のバージョンとの互換性のためだけに残されています。すべての記録は、IE モードまたは URL モードのいずれかを使って行われます。

IE モードと URL モードとの違いは、プローブで使われる技術と、個々のトランザクションステップを URL として指定するかしないかということです。

HTTP_TRANS プローブは、URL モードのプローブよりも多くのリソースを消費します。特に、1 つのトランザクションシーケンスに多くのステップが含まれている場合には、その傾向が顕著になります。なぜなら、ステップ間の考慮時間が短く (または全く無く)、ステップが次から次へと実行されるからです。このようなトランザクションを数多く同時に実行すると、CPU の使用率は高くなり

ます。また、各プローブの実行には、3～20 MB のメモリーを必要とするプロセスが伴いますが、このプロセスも IE モードの場合、URL モードよりも多くのメモリーを必要とします。



URL モードの方が、より多くの CPU 時間を消費する可能性があります。プローブには各ステップ間で遅延（考慮時間）がないからです。ステップの数が多いと、CPU スパイクが発生します。特に、監視対象のステップが結果をすぐ返すような場合です。URL モードのプローブは、1つのトランザクションについて約3～5 KB の常駐メモリーを使います。IE モードでは、プローブが各ステップ間で1.5秒間停止するので、プローブ自体は少ないCPU時間しか使いません。ただし、IE モードでは、レンダリングするページに、ブラウザで実行する必要がある、クライアント側のブラウザコントロールやロジックが多く含まれている場合には、多くのCPU時間を使います（このようなコントロールはURLモードのプローブでは実行されません）。URLモードのプローブは、1つのトランザクションについて約10～20 KB の常駐メモリーを使います。これは、ダウンロードしたページでレンダリングするコンテンツの種類によって変化します。

プローブシステムが全面的に OVIS プローブに依存する場合には、同時に実行する数に注意する必要があります。同時に実行する数が多いためにシステムやメモリーリソースが飽和状態になり、プローブのリソースが枯渇すると、プローブはタイムアウトになって失敗し、実行を完了させることができなくなります。HTTP_TRANS プローブを、別のコンピューティングタスクやアプリケーションも実行しているプローブシステムで実行する場合にも、同時に実行する数には注意して、プローブによる追加リソースの消費が他のコンピューティング機能に負の影響を及ぼさないようにすることを確認する必要があります。

一般的にデフォルトの同時実行数の値32は、各シーケンスが5ステップ以上で構成される HTTP_TRANS プローブでは多すぎます。このように多くの HTTP_TRANS 監視対象を持つ環境では、同時実行数の値は、10以下にする必要があります。プローブの同時実行数は、1～10の同時実行数を持つネットワークを設定するか、[**プローブロケーション**] ダイアログボックスの [**監視対象の優先度**] オプションを使って、制御することができます。そして、IEモードの各プローブを、このネットワークに割り当てます。確かでない場合は、パフォーマンスツールを使ってプローブシステムを監視して、プローブの影響と、割り当てた間隔内でプローブが完了できるかどうかを判断します。詳細は、『*HP OpenView Internet Services ユーザーリファレンスガイド*』の第7章の「スケーラビリティ情報」の項を参照してください。同時実行数についての詳細は、本書の第3章の110ページの「HTTP_TRANS プローブがリソース使用率に与える影響」を参照してください。

以下の表に、ナビゲーションモードでサポートされている機能の概要を示します。

表 1 Web トランザクションのナビゲーションモード

ナビゲーションモード	Java スクリプト	ActiveX Java	動的 URL	プラグイン	レンダリング
IE モード (デフォルト) probeHttpTrans2	○	○ *	○	○ **	○
URL モード probeHttpTrans			○ ***	○ **	
ナビゲーションポイント probeHttpTrans			○		

* Active-X コントロールや Java アプレットを持つ Web ページでは、Internet Explorer を使ってマウスのクリックやキー入力を記録することができません。Record Mouse and Keyboard Events オプションを使うと、これらの実際のクリックやキー入力を記録して、後で再生することができます (制限については、41 ページの「記録のオプション」の項の「マウスとキーボードのイベントの記録 (IE モード)」を参照)。

** URL を読み込むプラグインのみ。

*** 代替規則を使用。



プローブの実行プログラムは異なります。IE モードでは probeHttpTrans2 で、URL モードでは probeHttpTrans です。

使用するモードを判断するためのヒント

以下の条件に当てはまる場合は、IE モードを使います。

- Web ページに JavaScript または Java アプレットが含まれている (例: オンラインフォーム)
- Web トランザクションが複雑である
- Web ページがユーザーがアクセスするたびに動的に生成される
- 入力がプログラムまたはスクリプトから送られてくる

- ブラウザのエミュレートで実際に近い測定を行いたい
- Web トランザクションを URL モードで記録できない (例 : ページに JavaScript があったり、レンダリングを必要とする場合)

以下の条件に当てはまる場合は、URL モードを使います。

- プローブを UNIX システムや Linux システムから実行する必要がある
- IE モードの機能が不要である

Web Transaction Recorder の動作

Web Transaction Recorder には、記録と再生の機能があります。Web Transaction Recorder の機能は、URL モードと IE モードの、2 つの記録モードで異なります。

記録

IE モードの記録

IE モードで記録すると、Web Transaction Recorder は IE のブラウザエンジン (IE API) を呼び出します。Web トランザクションを実行していくと、IE のブラウザエンジンは、Web Transaction Recorder によって収集された通知イベントを発行します。通知イベントでは、各ステップで読み込まれた情報が送信されます。たとえば、ナビゲーションが行われる前に、BeforeNavigate2 イベントが発行されます。ドキュメント全体が読み込まれると、DocumentComplete イベントが発行されます。ボタンやリンクのような HTML 要素がクリックされると、OnClick イベントが発行されます。

IE モードの記録では、Web Transaction Recorder はユーザーが実際にクリックした HTML 要素を記録します。実際の HTML 要素を記録しているので、コンテンツが変化してもそれに対応した再生が可能で、ユーザーの操作の再現性にも優れています。

URL モードの記録

URL モードで記録すると、Web Transaction Recorder は IE のブラウザエンジンを呼び出します。Web トランザクションを実行していくと、リンクをクリックしたり、ボタンを押すたびに、Internet Explorer によって新しい URL が読み込まれます。Internet Explorer が新しい URL を読み込むたびに、BeforeNavigate2 イベントが発行されます。Web Transaction Recorder は、BeforeNavigate2 イベントで報告された URL を取得します。BeforeNavigate2 イベントはすべての URL を渡しますが、Web Transaction Recorder が取得するのは各ステップの最初の URL だけです。これは記録だけを目的としているためです。実行されているプローブは HTTP プローブです。

記録の結果

httptrans.dat ファイルには、記録したステップの指示だけでなく、HTTP_TRANS プローブ用の設定情報も格納されています。記録したステップを正しく再生するには、十分な情報が必要です。URL モードの記録では、URL のみが必要となります。IE モードの記録では、プローブが再生時に正しい HTML 要素を検出できるよう、各種のパラメータが記録されます。

記録の簡単な例

例の下に、ダイアログボックスの HTML を示します。



フレームセットドキュメントと、HTML ドキュメントが、各フレーム「Frame 1」と「Frame 2」にあります。これらのフレームには、いずれにも **[Submit]** ボタンが付いています。「Frame 1」の **[Submit]** ボタンを押すと、Web Transaction Recorder は次のように記録します。

```
IENAVPNT=INPUT:button "f1" "5" "" "Submit" ""
```

この情報に関する説明は、次の項を参照してください。

frametest.html:

```
<html>
<FRAMESET ROWS="100,*">
<FRAME name="f1" src="f1.html">
<FRAME name="f2" src="f2.html">
</FRAMESET>
```

```
<body>  
</body>  
</html>
```

f1.html:

```
<html>  
<title>Frame 1</title>  
<body>  
Frame 1  
<br>  
<INPUT TYPE=button VALUE=Submit>  
</body>  
</html>
```

f2.html:

```
<html>  
<title>Frame 2</title>  
<body>  
Frame 2  
<br>  
<INPUT TYPE=button VALUE=Submit>  
</body>  
</html>
```

Web Transaction Recorder は、「Frame 1」の **[Submit]** ボタンが押されたときに十分な情報を記録する必要があります。これにより、トランザクションを正確に再生することができます。Web Transaction Recorder は DOM(ドキュメントオブジェクトモデル)の情報を使って、HTML 要素(ユーザーが押したボタン)を一意的に識別します。

上記の例の3つのフレームのDOM(ドキュメントオブジェクトモデル)を、下に示します。

- **Index (インデックス)** – 各 HTML 要素は、DOM 内のインデックス (0000, 0001 など) を持ちます。
- **frame (フレーム)** – フレーム名 (_top, f1, f2 など) が表示されます。
- **html element (html 要素)** – DOM には、HTML 要素 (HEAD、INPUT など) が表示されます。
- **[] 内の Web Transaction Recorder 形式** – Web Transaction Recorder は、各 HTML 要素の各種のプロパティ (タイプ、フレーム名、インデックスなど) を保存します。この情報は、Web Transaction Recorder が再生で DOM 内の要素を検出し直すときに必要です。
- **html source (HTML ソース)** – DOM は要素のソーステキストとそこに含まれるすべてのものを表示します。

Frametest.html:

```
index|frame|html |Web Recorder|      html source      |
      element| format [ ] |
0000: '_top' HTML [# "_top" "0"] '<HTML><HEAD></HEAD><FRAMESET rows=100,*><FRAME name=f1 src='
0001: '_top' HEAD [# "_top" "1"] '<HEAD></HEAD>'
0002: '_top' TITLE [# "_top" "2"] ''
0003: '_top' FRAMESET [# "_top" "3"] '<FRAMESET rows=100,*><FRAME name=f1 src="f1.html"><FRAME nam'
0004: '_top' FRAME [# "_top" "4"] '<FRAME name=f1 src="f1.html">'
0005: '_top' FRAME [# "_top" "5"] '<FRAME name=f2 src="f2.html">'
```

f1.html:

```
0000: 'f1' HTML [# "f1" "0"] '<HTML><HEAD><TITLE>Frame 1</TITLE></HEAD> <BODY>Frame 1 <BR'
0001: 'f1' HEAD [# "f1" "1"] '<HEAD><TITLE>Frame 1</TITLE></HEAD>'
0002: 'f1' TITLE [# "f1" "2"] '<TITLE>Frame 1</TITLE>'
0003: 'f1' BODY [# "f1" "3"] ' <BODY>Frame 1 <BR><INPUT type=button value=Submit> </BODY>'
0004: 'f1' BR [# "f1" "4"] '<BR>'
0005: 'f1' INPUT [INPUT:button "f1" "5" "" "Submit" ""] '<INPUT type=button value=Submit>'
```

f2.html:

```
0000: 'f2' HTML [# "f2" "0"] '<HTML><HEAD><TITLE>Frame 2</TITLE></HEAD> <BODY>Frame 2 <BR'
0001: 'f2' HEAD [# "f2" "1"] '<HEAD><TITLE>Frame 2</TITLE></HEAD>'
0002: 'f2' TITLE [# "f2" "2"] '<TITLE>Frame 2</TITLE>'
0003: 'f2' BODY [# "f2" "3"] ' <BODY>Frame 2 <BR><INPUT type=button value=Submit> </BODY>'
0004: 'f2' BR [# "f2" "4"] '<BR>'
0005: 'f2' INPUT [INPUT:button "f2" "5" "" "Submit" ""] '<INPUT type=button value=Submit>'
```

上記の例で「Frame 1」の **[Submit]** ボタンを押すと、Web Transaction Recorder によって次のように記録されます。

```
IENAVPNT=INPUT:button "f1" "5" "" "Submit" ""
```

httptrans.dat ファイルには、記録したステップの指示が含まれます。情報はスクリプト形式で記録されます。IESTEP と IENAVPNT の指示は、トランザクション内のユーザーアクションを識別するために使われます。この例では、ユーザーアクションは「Frame 1」で **[Submit]** ボタンを押すことに相当します。

Web Transaction Recorder は、再生時の指示が IESTEP または IENAVPNT によって、異なる手順を実行します。詳細は、次の「再生」の項を参照してください。

再生

URL モードの再生

URL モードを使用する場合、Web Transaction Recorder はパイプを確立して、probeHttpTrans.exe (HTTP_TRANS プローブの実行可能プログラム) を起動します。probeHttpTrans.exe は、記録内の現在のステップのダンプ <filename> を「stdout」に出力して、一時停止します。Web Transaction Recorder は一時停止を検出すると、ダンプファイルを読み込み、それを表示します。その後、Web Transaction Recorder は、probeHttpTrans に対して、パイプを通じて再開するように指示します。

IE モードの再生

httptrans.dat ファイルには、記録したステップの指示だけでなく、HTTP_TRANS プローブ用の設定情報も格納されています。情報はスクリプト形式で記録されます。IESTEP と IENAVPNT の指示は、トランザクション内のユーザーアクションを識別するために使われます。

例を、次に示します。

```
[HTTP_TRANS]
CUSTOMER=hp
SERVICENAME=OVIS
FLAGS=cache=0,silent=0,noNewWindow=0,waitTime=1500,incWaitTime=0,runJava=1,about
BlankInit=1,captureWindow=0
IESTEP=www.hp.com
PATTERN="+Welcome to HP"
IENAVPNT=INPUT:image "_top" "466" "submit" "" "http://welcome.hp-ww.com/img/
hpweb_1-2_arrw_sbmt.gif"
PATTERN="+Search HP US - Search results for 'Internet Services'"
FORMDATA=##FPHEGPHACOHDGFGBHC GDGIEGGPHCGNCOHBHEDNEJGOHEGFGHCGOGFHECAFDGFGHCHGGJGDG
FHD
IENAVPNT=A "_top" "231" "" "http://search.hp.com/gwuseng/
redirect.html?url=http%3A//www.openview.hp.com/solutions/
&qt=Internet+Services&pos=1&type=REG"
PATTERN="+Hewlett-Packard OpenView - solutions"
```

IESTEP は、再生時に Internet Explorer に対して、定義されている URL に移動するように指示します。IENAVPNT 指示を検出した場合、Web Transaction Recorder は、以下の動作を行って記録 / ステップを再生する必要があります。

1 HTML 要素の情報の解析

- 2 ドキュメントオブジェクトモデル (DOM) 内の要素の検出
- 3 記録された要素と検出した要素が同じであることを確認
- 4 HTML 要素上のユーザーのクリックのシミュレート
- 5 ステップの最後を検出



Web トランザクション内で測定対象を正しく反映するために、各種のオプションを設定する必要があります。

HTTP_TRANS プローブの動作

HTTP_TRANS プローブには、2つのプローブがあります。

- probeHttpTrans.exe – URL モードを使って Web トランザクションを記録したときに作成される HTTP_TRANS プローブ
- probeHttpTrans2.exe – IE モードを使って Web トランザクションを記録したときに作成される HTTP_TRANS プローブ

URL モードのプローブは、基本的に記録したリスト内の各ステップ (URL) で HTTP プローブコードを呼び出し、プロセスに対応した Cookie のキャッシュをステップにまたがって管理します。このモードは Web サイトが利用できるかを確認するには十分です。ただしユーザーは IE ブラウザを使用するので、このモードでの応答時間はユーザーが実際に感じている応答時間を反映しません。

IE モードのプローブは、IE のブラウザエンジンを呼び出し、Web トランザクションの各ステップを実行することで、あたかもユーザーがブラウザのウィンドウで操作したときのように、ユーザーの操作をシミュレートします。このプローブは IE のブラウザエンジンを使うので、プローブには大きなオーバーヘッドが伴います。しかし、IE のブラウザエンジンはそれに見合った役割を果たします。その結果、このプローブは JavaScript やレンダリングのような機能を処理することができます。同時に実行できる HTTP_TRANS プローブの数については、110 ページの「HTTP_TRANS プローブがリソース使用率に与える影響」を参照してください。

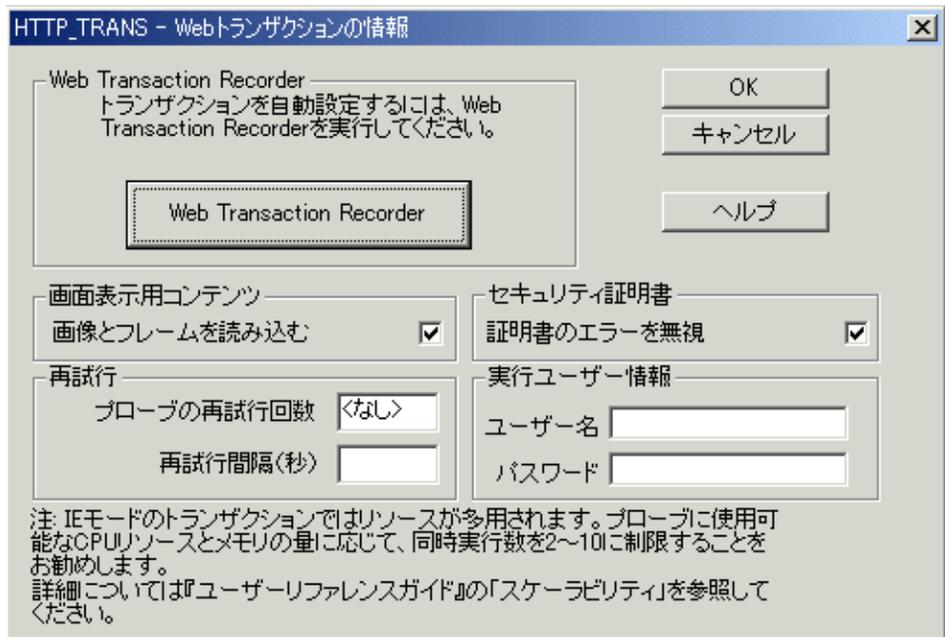
Web Transaction Recorder の使用方法

この章では以下のトピックについて説明します。

- HTTP_TRANS プロープの設定
- Web トランザクションを記録する手順
 - 基本的な手順
 - グローバルなトランザクションプロパティの設定
 - 基本的なトランザクションステップのプロパティ
 - トランザクションステップの詳細プロパティ
- 詳細な使用方法
- トランザクションスクリプトのリファレンス
- Web Transaction Recorder についてのヒント

HTTP_TRANS プローブの設定

Web Transaction Recorder を起動する前に、他のプローブと同様に、OVIS 設定マネージャを使って、HTTP_TRANS プローブを設定する必要があります。すなわち、顧客、サービスグループ、監視対象サービスを設定します。監視対象サービスの作成を開始すると、以下のダイアログボックスが表示されます。



このダイアログボックスでいくつかのオプションを定義して、[Web Transaction Recorder] ボタンをクリックすると、[Web Transaction Recorder] ダイアログが表示されます。

HTTP_TRANS プローブは、OVIS スケジューラを実行するアカウントでではなく、別のユーザーで実行するように設定することができます。[HTTP_TRANS - Web トランザクションの情報] ダイアログボックスの[実行ユーザー情報]セクションに、ユーザー名とパスワードを入力します。

▶ IEモードのHTTP_TRANSプローブは大量のCPUとメモリーのリソースを使うため、このプローブタイプの同時実行の数には制限があります。同時実行の数が多すぎると、プローブプログラム probehttptrans2.exe が異常終了する原因になります。これを避けるには、設定マネージャの[**プローブケーションの情報**]ダイアログで、同時実行の数を1～10の範囲に制限します。詳細は、110ページの「HTTP_TRANS プローブがリソース使用率に与える影響」を参照してください。

以下のような、ナビゲーションモードを選択するためのプロンプトが表示されます。

- IEモード
- URLモード
- ナビゲーションポイントモード(このモードは廃止されました。以前のバージョンとの互換性のためだけに残されています。)

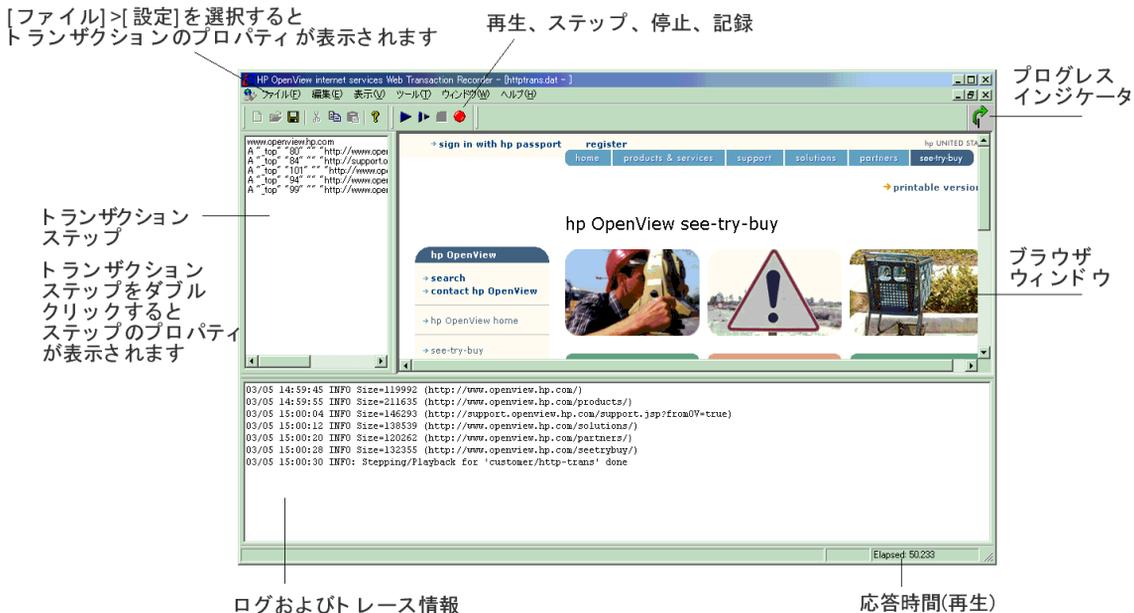
IEモードとURLモードについての詳細と、Web トランザクション用に最適の記録方法を選択する方法は、第1章を参照してください。



Web Transaction Recorder では、トランザクションは個々の独立したステップから成ります。各ステップは、ユーザーが Web ページでクリックする、URL や HTML 要素 (たとえば、リンクやボタン) などです。

Web トランザクションを記録する手順

Web トランザクションの記録には、Web Transaction Recorder の GUI を使います。この GUI は、HTTP_TRANS 監視対象サービスを設定するときに、設定マネージャによって自動的に起動されます。また、実行可能プログラム (<インストールされたディレクトリ>¥bin¥WebRecorder.exe) を直接指定して起動することもできます。以下の図には、GUI の例を示します。



Web Transaction Recorder では、トランザクションは個々の独立したステップから成ります。Web アプリケーションで、リンクまたはボタンをクリックしたり、他の Web ページに移動したりして、トランザクションが実行されます。そのすべてのアクションが記録されます。

記録を再生してトランザクションを確認し、必要に応じて変更します。

HTTP_TRANS プロブは、記録されたデータを使用して Web トランザクションを監視します。プロブによって、トランザクションの可用性と応答時間のメトリックが収集され、OVIS ダッシュボードに表示されます。

基本的な手順

トランザクションを記録するための基本的な手順は、以下のとおりです。

- 1 記録する Web ページを決定します。ブラウザで Web トランザクションのステップを確認して、その順番や選択項目を把握して、正しい記録を行う必要があります。
- 2 トランザクションの記録を開始する前に、Web Transaction Recorder で [**ツール**] メニューの [**キャッシュビューアー**] を使って Cookie を削除します。**[Cookie を無効にする (IE)]** オプションをオンにすると、再生時に Cookie が削除されます (39 ページの「グローバルなトランザクションプロパティの設定」を参照)。
- 3 [**ファイル**] > [**設定**] の順に選択して、[**プロパティ**] ダイアログボックスを開き、グローバルなトランザクションプロパティを設定します。ポップアップ/エラーダイアログ、タイムアウト/待機時間、エラーのキャプチャ、プロキシ設定、トレースレベルなどを Web Transaction Recorder が処理する方法を指定することができます。
- 4 メインウィンドウで、記録ボタンをクリックして、トランザクションステップの記録を開始します。
- 5 最初の URL を入力します。

右ペインのブラウザウィンドウに Web ページが完全に読み込まれたら (右上にある緑の矢印の回転が停止したら)、表示されている Web ページ内で操作し、トランザクションステップを記録することができます。

左ペインにトランザクションステップが表示されます。トランザクションステップのログやトレース情報は、下のペインに表示されます。
- 6 停止ボタンをクリックして、記録を終了します。
- 7 記録した内容を再生して、必要なすべてのステップが含まれているかどうかをテストします。再生時の応答時間は右下に表示されます。
- 8 必要に応じて、Web Transaction Recorder のオプションを使用し、以下のような変更を行います。
 - 左ペインのトランザクションステップを右クリックして、[**プロパティ**] を選択すると表示される [**ステップのプロパティ**] ダイアログボックスで、トランザクションステップのプロパティを変更します。

- 左ペインのトランザクションステップを右クリックして、[**プロパティ**] を選択し、[**詳細**] タブを選択すると表示されるダイアログボックスで、詳細なスクリプト情報を入力します。

- 9 Web Transaction Recorder を終了して、[**HTTP_TRANS - Web トランザクションの情報**] ダイアログの [**OK**] ボタンをクリックします。
- 10 Web Transaction Recorder を使って作成したプローブのプローブロケーションを設定します。
- 11 Web Transaction Recorder を使って作成したプローブのサービスレベル目標値を設定します。

ステップアラーム機能を設定することができます。HTTP_TRANS プローブトランザクションの各ステップにアラームを設定する方法は、『*HP OpenView Internet Services ユーザーリファレンスガイド*』の第3章にある「サービスレベル目標値とアラームの設定」の項を参照してください。ステップアラーム機能には、アラームを設定する機能しかありません。ステップのしきい値は、サービスレベル目標値では使われません。

アラームセットは、1つのステップだけにしか割り当てることはできません。トランザクションのいくつかの異なるステップにアラームを作成するには、それぞれのステップに対して異なるアラーム定義を作成する必要があります。複数のアラームセット定義で同じメトリックを使うことができます。

ステップアラーム機能は、可用性メトリックではなく、応答時間メトリックで使われます。可用性は、トランザクション全体でしか調べることができないためです。いずれかのステップが実行できない場合には、トランザクションには利用不可というマークが付けられます。

ダッシュボードと、設定マネージャの [**ステータス**] ページの詳細に、ステップ番号が表示されます。

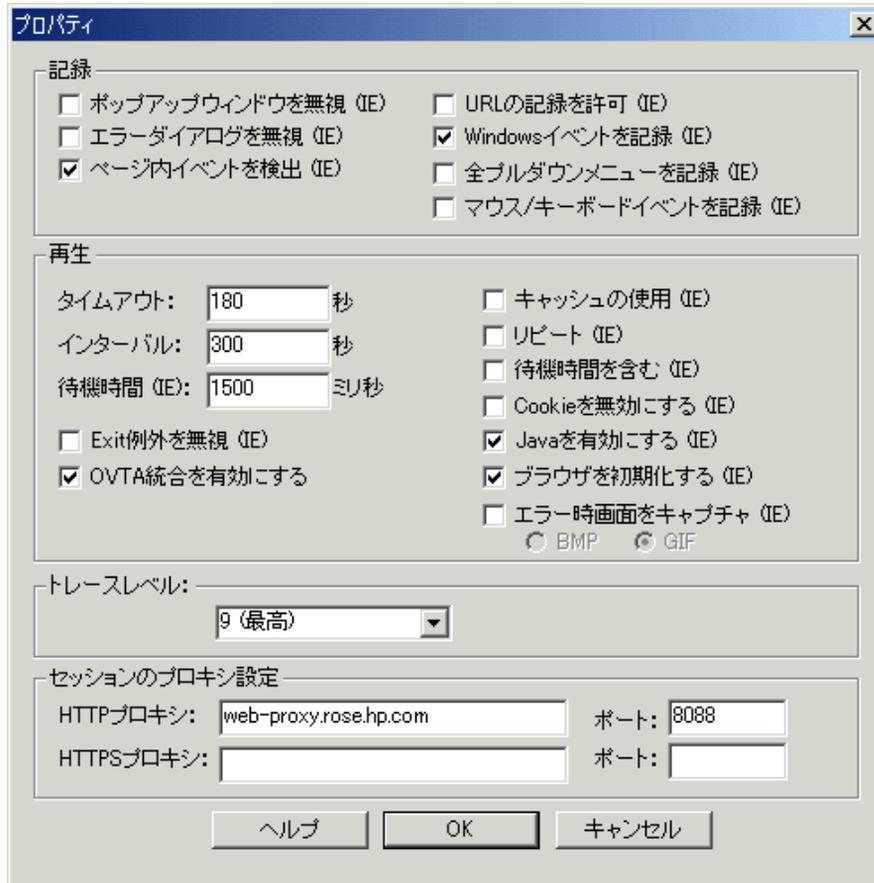
- 12 設定マネージャでのプローブの変更を保存して、終了します。



設定マネージャの IE モードの HTTP_TRANS プローブの [**管理サーバーでテスト**] セクションは、Windows XP 以降でしか使えないことに注意してください。

グローバルなトランザクションプロパティの設定

[**ファイル**] > [**設定**] の順に選択して、[**プロパティ**] ダイアログボックスを開いて、トランザクション全体に適用されるプロパティを指定できます。また、ステップをダブルクリックして[**ステップのプロパティ**] ダイアログボックスを表示して、トランザクションの個々のステップに適用されるプロパティを指定することもできます。各項目については後述します。トランザクション全体に対して値を設定する必要がある場合は、106 ページの「記録と再生に関する問題点」の例を参照してください。



記録のオプション

[**ポップアップウィンドウを無視 (IE)**]: このフラグを設定すると、広告のようなポップアップウィンドウは、記録時に無視されます。このオプションは IE モードの場合のみ有効です。スクリプトオプション `enableIEDialog` を設定した場合、このチェックボックスの指定は無効になります (91 ページの「`enableIEDialog`」を参照)。

[**エラーダイアログを無視 (IE)**]: このフラグを設定すると、JavaScript のエラーのようなダイアログボックスは、記録時に無視されます。このオプションは IE モードの場合のみ有効です。

[**ページ内イベントを検出 (IE)**]: このフラグを設定すると、Web Transaction Recorder は、サーバーからドキュメントを読み込まないステップに対して、自動的に `continueClick=1` フラグを挿入します。このオプションは IE モードの場合のみ有効です。

[**URL の記録を許可 (IE)**]: このフラグを設定すると、Web Transaction Recorder は、ナビゲーションポイントが認識できなかった場合、URL を記録します。このオプションは、マウスイベントを内部で処理する複雑な Web アプリケーションの場合に使います。このオプションは IE モードの場合のみ有効です。



記録された URL が動的な場合には、代替規則を適用する必要があります (53 ページの「動的な URL の代替処理」を参照)。このオプションは IE モードの場合のみ有効です。

[**Windows イベントを記録 (IE)**]: このフラグを設定しない場合には、Web Transaction Recorder は選択された HTML 要素を OVIS-5.0 以前の方法を使って調べます (OnClick 通知イベントを使います)。このオプションは IE モードの場合のみ有効です。

このフラグを設定すると、Web Transaction Recorder は、マウスで示された HTML 要素を OVIS-5.0 以前とは異なるアルゴリズム (IE で使われているアルゴリズムに類似) を使って調べます。

[**全プルダウンメニューを記録 (IE)**]: プルダウンメニューの選択に基づいて動的にコンテンツを読み込む Web アプリケーションを記録する場合は、記録時の、プルダウンメニュー項目を選択する直前にこのフラグを設定する必要があります。Web Transaction Recorder は通常、このようなプルダウンメニューの選択は記録しません。プルダウンメニューの項目は単にフォームの一部と見なされ、フォームの入力完了を待って次のステップが記録されます。このフラグを設定すると、プルダウンメニュー項目の選択を記録することができます。プルダウンメ

ニューの記録が終わった後はこのオプションのチェックを外してください。また、トランザクションの記録の開始時にこのオプションを設定しないことをお勧めします。こうすることで、ナビゲーションポイントが重複したり、他のナビゲーションポイントが記録されることを防ぎます。68 ページの「HTML ダイアログとプルダウンメニューの内部処理」を参照してください。

また、SELECT ナビゲーションポイントに記録された formdata 文を、削除する必要がある場合もあります。Web アプリケーションによっては、特定のメニューに対応する formdata 文のみをナビゲーションポイントに含める必要があります。

たとえば、次のような場合です。Web ページに 2 つのメニュー "m1" と "m2" があるとします。2 番目のメニュー "m2" を記録すると、次のように、"m1" の formdata パラメータが記録されることがあります。

```
ienavpnt=SELECT "_top" "169" "m2" "2"  
formData=_top.form.m1=value_for_m1  
formData=_top.form.m2=value_for_m2
```

これは次のように変更する必要があります。

```
ienavpnt=SELECT "_top" "169" "m2" "2"  
formData=_top.form.m2=value_for_m2
```

また、サブミット機能は SELECT ナビゲーションポイントのデフォルトの動作であるため、Web Transaction Recorder でフォームのサブミット機能を実行しないように指示する必要があります。サブミット機能を実行しないようにするには、SELECT ナビゲーションポイントに nosubmit オプションを指定し、さらにページのナビゲーションを行わないように continueClick=1 を追加します。

```
ienavpnt=SELECT "_top" "169" "m2" "2" "nosubmit"  
formData=_top.form.m2=value_for_m2  
continueClick=1
```

[マウス/キーボードイベントを記録 (IE)]: Web ページには、Active-X コントロールや Java アプレットが存在する場合があります。このような Web ページでは、マウスのクリックやキー入力は IE を介して記録することができません。このオプションを使うと、このような実際のクリックやキー入力を記録したり、後から再生したりすることができるようになります。このオプションは、マウスやキー入力のイベントの記録が必要な特定のステップだけで有効にすることをお勧めします。

Active-X コントロールや Java アプレットでは、マウスやキー入力のイベントが、オペレーティングシステムの下位レベルで発行されることを前提にしているために、これらのクリックの再生ができない場合があります。このような場合は、Web Transaction Recorder ではサポートできません。

再生のオプション

[**タイムアウト**]: 再生用のタイムアウト値を調整します。これはトランザクションをテストする際に役に立ちます。有効なタイムアウト値を調べて、[**プローブロケーションの情報**] ダイアログボックスで設定する必要があります。

[**インターバル**]: 再生を繰り返す間隔です。このオプションは [**リポート (IE)**] オプションをオンにしている場合のみ適用されます。この値は、[**プローブロケーションの情報**] ダイアログボックスの [**測定間隔**] には適用されません。

[**待機時間 (IE)**]: IE モードでプローブがリダイレクションの検出に使う待ち時間です。トランザクションのタイムアウトが早すぎる場合は、この時間を調整します。このオプションは IE モードの場合のみ有効です。

[**Exit 例外を無視 (IE)**]: このチェックボックスにチェックを付けると、Web Transaction Recorder のプローブは、トランザクション完了後に発生した例外による、ランタイムエラーを無視します。他社製の Internet Explorer プラグインでは、トランザクションをクリーンアップして終了するときに、例外が発生するものもあります。このボックスにチェックを付けると、プローブを実行するたびにコンソールにエラーメッセージが表示されることを回避することができます。

チェックボックス

[**キャッシュの使用 (IE)**]: このフラグは、IE ブラウザのオブジェクト (HTML、画像など) のダウンロード方法に影響を与えます。このチェックボックスがオフの場合 (デフォルト)、プローブは、プロキシを無視し HTTP 要求をサーバーに直接送信するよう、IE に指示します (プロキシでのキャッシングがサポートされていても、オブジェクトはプロキシからではなく、サーバーから取り込まれます)。また、ローカルの IE キャッシュも無視され、ダウンロードするオブジェクトについてのアップデート情報は常にサーバーに対して問い合わせが行われます。オブジェクトに変更がない場合 (サーバーからステータスコード "HTTP 1.1 304" を受信) は、キャッシュ内のオブジェクトが使用され、サーバーからはダウンロードされません。このオプションは IE モードの場合のみ有効です。

特定の IE レジストリキーも、キャッシュに影響を与えます。詳細は、66 ページの「[プローブの結果に影響を与えるレジストリの設定 \(IE モード\)](#)」を参照してください。

[リポート (IE)]: このオプションをオンにすると、トランザクションは連続して再生されます。このオプションは、Web Transaction Recorder アプリケーションの記録のリポート再生モードのみ制御します。プローブ動作は、プローブ対象の **[測定間隔]** で制御されるため、変更されません。このオプションは IE モードの場合のみ有効です。

[待機時間を含む (IE)]: このオプションをオンにすると、応答時間の計算に待ち時間 (**[待機時間 (IE)]** フィールド) を含めます。これは各トランザクションステップにユーザーの思考時間を含めるために使われます。このオプションは IE モードの場合のみ有効です。

[Cookie を無効にする (IE)]: このオプションをオンにすると、スケジューラがプローブを実行する前に、すべての固定 Cookie が削除されます。このオプションは IE モードの場合のみ有効です。URL モードでは、プローブは Cookie を格納しません。



Cookie はユーザー固有であり、動作中のすべてのトランザクションで共有されます。したがって、同時に実行されるトランザクションは、同じ Cookie インタフェースを使います。また、Cookie は、(**[プローブプロケーションの情報]** ダイアログボックスで設定された) 間隔が一番長いトランザクションの開始時に削除されます。たとえば、あるトランザクションが 300 秒ごとに実行され、別のトランザクションが 140 秒ごとに実行されるとすると、Cookie は 300 秒ごとにしか削除されません。このような場合には、プローブに別のプローブシステムを使うか、またはすべてのトランザクションを同じ間隔で実行するようにしてください。

[Java を有効にする (IE)]: このオプションをオンにすると、再生時に Java アプレットが読み込まれて実行されます。Java との相互作用はサポートされていません。

[ブラウザを初期化する (IE)]: トランザクションで新しいウィンドウがポップアップされ、そのウィンドウで再生が失敗する場合には、このオプションのチェックを外します。チェックを付けたままにすると、JavaScript の認証エラーが報告されます。Web サーバーまたはプロキシで認証が必要な場合には、このフラグにはチェックを付けておく必要があります。また、noAbout=1 を使って特定のステップにおける初期化を無効にすることもできます。 **[ステップのプロ**

パティ] ダイアログボックスの **[詳細]** タブで、noAbout を指定します。トランザクションスクリプトのリファレンス [89 ページの表](#)の、noAbout を参照してください。

[エラー時画面をキャプチャ (IE)]: このオプションを使用すると、パターンマッチが失敗した場合、またはタイムアウトになった場合に、Web ページの画面イメージが記録されます。これは特定の問題をトラブルシューティングする場合に役に立ちます。画面イメージは、
`<data dir>%data%tmp%probe%capture%<Customer>_<Service Group>_<ProbeID>_<ServiceID>_<TargetID>.bmp` または `.gif` に保存されます。ファイルを `.bmp` と `.gif` のいずれで保存するかを選択します。画像処理アプリケーションで最もはっきりと表示できるのは、`.bmp` イメージです。画面の解像度によっては、MS ペイントで `.bmp` のキャプチャを正しく表示できないことがあります。また、最新の画面イメージのみ表示することができます。既存の画面イメージがある場合には、上書きされます。このデフォルトの動作は、レジストリキーの設定を使って、変更することができます ([116 ページの「IE モードのレジストリ設定」](#)を参照)。

トレースレベル

[トレース]: トレースのレベルを選択します。0= なし、1= 最小トレース、5= 多めのトレース、9= 最大トレース。トレースは Web Transaction Recorder のメッセージウィンドウに書き込まれます。トラブルシューティングに使用するには、トレースレベルを 9 に設定することをお勧めします。使用例は、[119 ページの「HTTP_TRANS の注釈付きトレースファイル」](#)を参照してください。

プロキシの設定

[セッションのプロキシ設定]: 現在の記録セッションに必要なプロキシとポートを設定します。この機能は、HTTP_TRANS プローブの **[プローブローケーションの情報]** ダイアログボックスの設定を一時的に変更するために使用します。

このフィールドが空白の場合は、プローブは設定マネージャで定義された **[プローブローケーションの情報]** のプロキシ設定を使って動作します。

このフィールドが空白で、**[プローブローケーションの情報]** も未設定であれば、プローブは IE のプロキシ設定を使って動作します。

プロキシを無効にするには、このフィールドに `<no proxy>` と入力します。

基本的なトランザクションステップのプロパティ

トランザクションステップをダブルクリックすると、[**ステップのプロパティ**] ダイアログボックスが表示され、各ステップの詳細情報を表示したり、追加情報を指定してプローブがステップを正しく実行できるように設定できます。[**ステップのプロパティ**] を定義して、特定のステップでの問題を解決する例については、102 ページの「Web Transaction Recorder についてのヒント」および 106 ページの「記録と再生に関する問題点」を参照してください。

[**タイプ**]: トランザクションステップのタイプを選択します。[**ターゲット**] フィールドの文は、このタイプと同じにする必要があります。

[**ターゲット**]: タイプに対応して実行するアクションを示す文です。IE モードの場合には、iestep は URL で、ienavpnt は HTML 要素です。

[**ラベル**]: このフィールドに、トランザクションステップのラベルを指定します。ラベルは、OVIS ダッシュボードの [**詳細**] で URL の代わりに表示されるので、見やすくなります。

▶ 動的 URL の場合、ダッシュボードには同じトランザクションステップに関するグラフが複数表示されることがあります。

[**ポストデータ**]: HTTP の POST データです。URL モード (または IE モードの IEstep) では、プローブは、GET メソッドの代わりに、HTTP POST メソッドを実行してデータを取得します。サーバーにデータを送信するには、クエリー文字列を付けた POST または GET を使う 2 通りの方法があります。このオプションにチェックを付けると、HTTP POST のデータを送信します。

[**パターン**]: パターンマッチを行います。プローブがダウンロードしたテキストにパターンが含まれていない場合には、トランザクションは利用不可とマークされます。IE モードでは、ドキュメントのタイトルが事前に設定されています。Recorder/ プローブでダウンロードしたテキストを表示するには、[**表示**] メニューの [**ソースの表示 (HTML)**] をクリックします。

▶ HTML 内の一部の文字は、コード化されています。たとえば、「&」は「&」、 「"」は「"」でコード化されます。パターンを設定する場合には、HTML ソースを見て、ブラウザに表示されている文字列にこのようなコード化された文字がないかどうかを調べます。コード化された文字が文字列に含まれている場合には、それに従ってパターンを調整します。

[**パターン設定**]: 単語間のデフォルトの演算子は、"AND" です。このフィールドを使って、"OR" に変更することができます。単語の比較は、デフォルトでは、大文字と小文字が区別されます。"ICASE" を入力することによって、大文字と小文字を区別しないようにすることができます。たとえば、大文字と小文字を区別せず、論理 OR を使うパターンマッチングを設定するには、このフィールドに "OR ICASE" を入力します。

[**ユーザー名**]: トランザクションステップで HTTP 認証が必要な場合には、ユーザー名を入力します。

[**パスワード**]: トランザクションステップで HTTP 認証が必要な場合には、パスワードを入力します。

[**パスワード確認**]: 前のフィールドで入力したパスワードを再入力します。

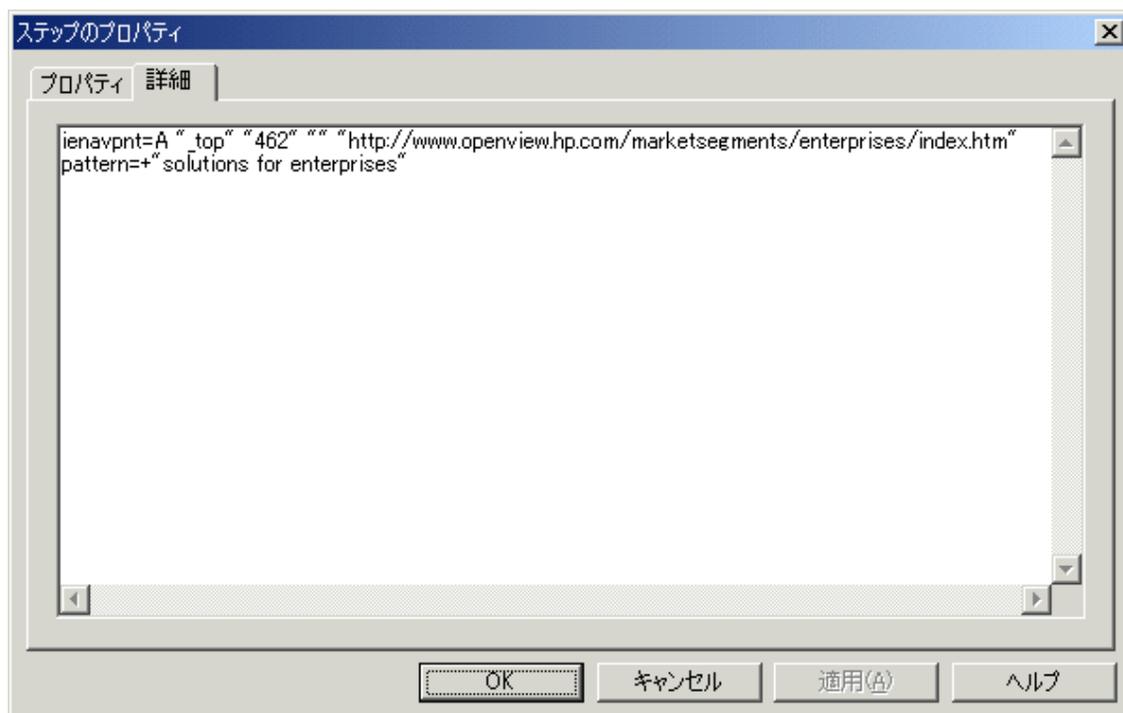
[**プロキシユーザー名**]: トランザクションステップでプロキシによる HTTP 認証が必要な場合には、ユーザー名を入力します。

[**プロキシパスワード**]: トランザクションステップでプロキシによる HTTP 認証が必要な場合には、パスワードを入力します。

[**プロキシパスワード確認**]: 前のフィールドで入力したパスワードを再入力します。

トランザクションステップの詳細プロパティ

[**ステップのプロパティ**] ダイアログボックスの [**詳細**] タブを選択すると、トランザクションステップに詳細なスクリプトオプションを設定できるウィンドウが表示されます。複雑な Web ページの記録を処理できるスクリプト文の種類については、50 ページの「**詳細な使用方法**」と 78 ページの「**トランザクションスクリプトのリファレンス**」を参照してください。通常は、最初にトランザクションを記録し、記録が終了してから、[**ステップのプロパティ**] ダイアログボックスの [**詳細**] タブを使って、記録を正しく再生するために必要なスクリプト文を追加します。



詳細な使用方法

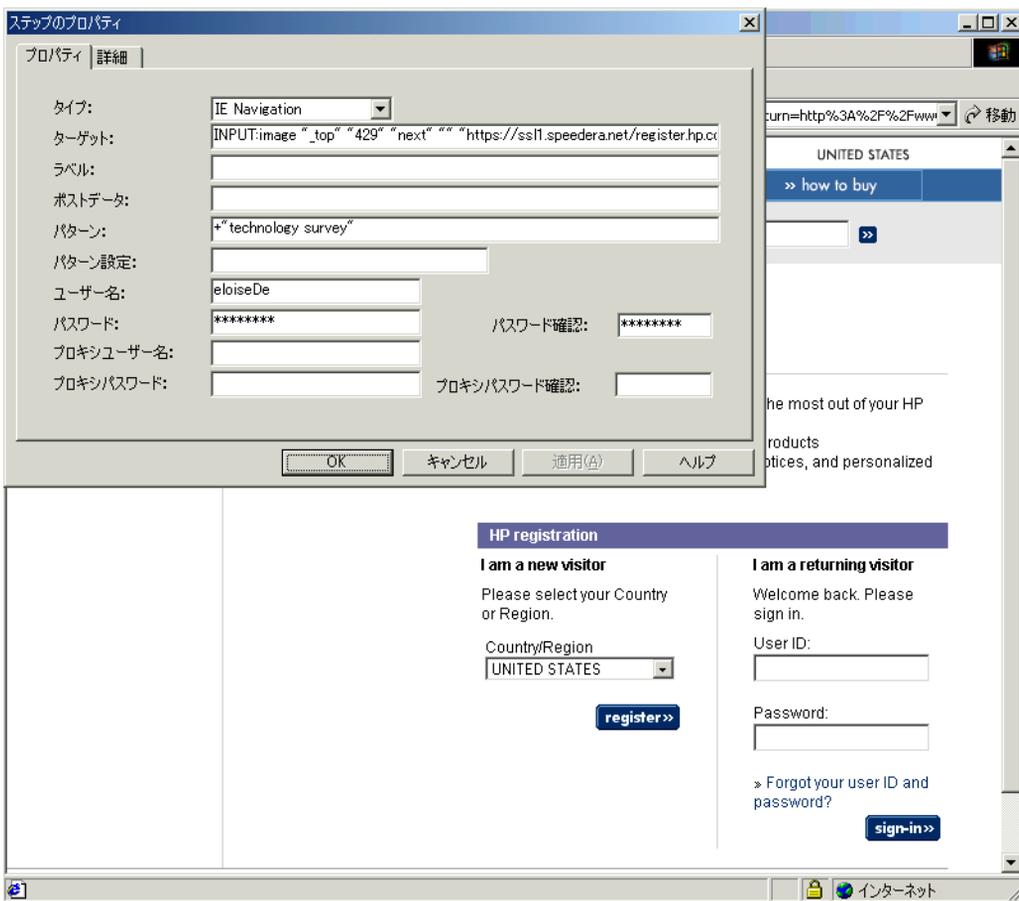
複雑な Web 機能を処理するスクリプト文の使用方法については、以下の詳細項目を参照してください。使用することができるスクリプト文についての詳細は、78 ページの「トランザクションスクリプトのリファレンス」を参照してください。

- Internet Explorer のセキュリティダイアログボックス
- 動的な URL の代替処理
- フォームデータ入力用の外部スクリプト (IE モード)
- 可用性チェック用外部スクリプト (IE モード)
- 動的なフレーム名
- プローブのログオンアカウントの変更
- Cookie の設定
- セッション Cookie の読み取りと保存
- 認証
- ブラウザプラグイン
- 複雑なページのリダイレクション
- 重複する入力変数
- レジストリを設定を使用してプローブの結果に影響を与えることもできます。66 ページの「プローブの結果に影響を与えるレジストリを設定 (IE モード)」を参照してください。
- HTML ダイアログとプルダウンメニューの内部処理
- 位置独立ナビゲーションポイント

Internet Explorer のセキュリティダイアログボックス

トランザクションの記録中に、セキュリティチェックや証明書のポップアップダイアログボックスが表示される場合があります。これらのダイアログボックスへの対応方法は、記録が終了した後で、ユーザー名やパスワードのような入

力情報を手作業で追加します。そのステップの [**ステップのプロパティ**] ダイアログボックスを使って、ユーザー名やパスワードを入力します。以下の例を参照してください。



証明書選択のようなダイアログボックスへの対応方法は、記録が終了した後に、トランザクションステップに dlgCmd 文を追加します。記録が終了した後で、[**ステップのプロパティ**] ダイアログボックスの [**詳細**] タブを使って、ステップにスクリプト文を追加します。これらのスクリプトの追加によって、プローブは、特定のボタンのクリックのような、ユーザーアクションを実行できるようになります (83 ページの表の dlgCmd を参照)。

dlgCmd 文の例は、次のとおりです。

```
dlgCmd=Button " クライアントの認証 " "OK"
```

最初の値は、タイプ (**Button**) を示し、その後にダイアログのタイトル (クライアントの認証) とボタンのテキスト (**OK**) が続きます。この文によって、プローブは、**[クライアントの認証]** ダイアログボックスで **[OK]** ボタンをクリックするというユーザーアクションを実行します。

さらに複雑なダイアログボックスも、`dlgCmd=Sequence <sequence>` コマンドを使って処理することができます。以下の例を参照してください。

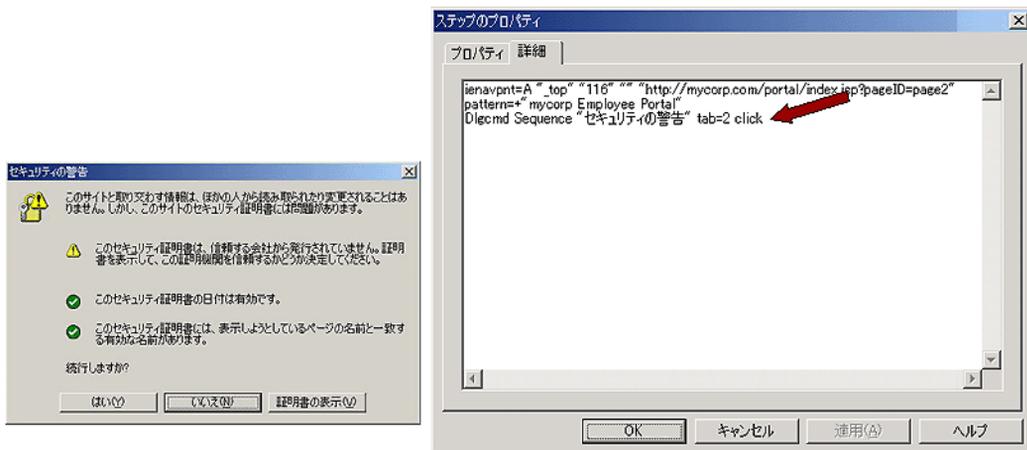
以下のいずれかの構文のコマンドを追加します。

```
dlgCmd=button <ダイアログのタイトル> <ボタンのラベル>
```

```
dlgCmd=Sequence <ダイアログのタイトル> <seq-1>..<seq-n>
```

この例で、`seq` には、`tab=x`、`up=x`、`down=x`、`click`などを指定することができます。

下図の例で、`tab=2 click` は、ダイアログボックスで **[いいえ]** ボタンを選択することを意味しています。



動的な URL の代替処理

ステップがタイプ STEP/IESTEP で構成されるトランザクションを記録している場合、トランザクションステップを変更して動的 URL への対応ができます。動的 URL とは、パスの一部またはクエリー文字列が当該ページにアクセスするたびに変化する URL のことです。

例:

```
http://www.dyn.hp.com/12345/index.html
```

```
http://www.dyn.hp.com/test?sesssionid=123455&arg=login
```

最初の例では、パスには 12345 という要素が含まれています。この要素はメインページにアクセスするたびに変化します。2 番目の例では、クエリー文字列の中の sessionid の値の 123455 は、毎回変化します。このようにセッション ID を、URL や POST データにコード化するような Web アプリケーションが存在し、これらは Web サイトにアクセスするたびに変化します。

動的な URL の代替処理の例

以下の例では、http://shopping.hp.com の URL のクエリー文字列に、2 つの固有の ID を含みます。

```
http://www.shopping.hp.com/cgi-bin/hpdirect/shopping/scripts/  
computer_store/  
computer_landing_view.jsp?BV_SessionID=@@@0722665204.10704  
05063@@@&BV_EngineID=deadckdgjfgjgcfngcfkmdfl1ldfgg.0&lan  
ding=desktops&selectMenu=desktops
```

再生中に 2 つの ID、BV_SessionID と BV_EngineID は、しばらく経つと、タイムアウトになります。記録したトランザクションは、ID が無効になってしまっているので、正しく再生することができません。

ステップが IESTEP(IE モード) または STEP(URL モード) で、これらのセッション ID を使用している場合には、正しい代替文字列を定義するため、以下のコマンドを使ってトランザクションステップを変更します。代替規則は、以下のとおりです。

dynCmdPattern0	置換する文字列の先頭を指定します。
dynCmdDelim0	文字列の末尾を指定します。
dynCmdSubst0	URL または POST のどの部分を文字列で置き換えるのか指定します。

これらのコマンドによって、規則が構成されます。各コマンドの最後の部分が、規則の ID になります(上の表では 0)。これによって 1 つのステップに複数の規則が設定できます。規則処理では前に処理した Web ページの情報を使って、規則が定義されているステップの URL または POST のデータを変更します。

2 つのステップで構成されるトランザクションの記録例を以下に示します。

```
[HTTP_TRANS]
IESTEP=http://www.shopping.hp.com/
IESTEP=http://www.shopping.hp.com/cgi-bin/hpdirect/shopping/
scripts/computer_store/
computer_landing_view.jsp?BV_SessionID=@@@0722665204.1070405
063@@@&BV_EngineID=deadckdgjfgjgcfngcfkmdflldfgg.0&landing=des
ktops&selectMenu=desktops
```

最初のステップの Web ページ (<http://www.shopping.hp.com/>) に、実際は BV_SessionID と BV_EngineID の現在の値が含まれています。[表示]メニューの [ソースの表示 (HTML)] から、Web ページ <http://www.shopping.hp.com> を表示すると、値が 2 番目のステップの URL で記録した値とは異なっていることがわかります。最初のステップの Web ページは、以下のとおりです。

```
...
<SCRIPT language=JavaScript type=text/javascript>
<!--
var menuLink1 = "http://www.shopping.hp.com/cgi-bin/hpdirect/
shopping/scripts/computer_store/
computer_landing_view.jsp?BV_SessionID=@@@0573045601.107041251
8@@@&BV_EngineID=ccdeadckdgjfgjgcfngcfkmdflldfgg.0&landing=des
ktops&selectMenu=desktops";var menuLink2 = ...
```

2 番目のステップで記録されている URL と Web ページ

www.shopping.hp.com の現在の URL が異なっていることに注意してください。動的 URL を処理するには、2 番目のステップで使う BV_SessionID と BV_EngineID の値を定義する代替規則が必要です。

2 番目のステップで BV_SessionID と BV_EngineID を置き換えるには、以下の 2 つの代替規則を追加します。

```

IESTEP=http://www.shopping.hp.com/cgi-bin/hpdirect/shopping/
scripts/computer_store/
computer_landing_view.jsp?BV_SessionID=@@@@1224306641.107041396
3@@@@&BV_EngineID=xxdeadckdggjfgjgcfngcfkmdflldfgg.0&landing=des
ktops&selectMenu=desktops
DYNCMDPATTERN0=BV_SessionID=
DYNCMDDELIM0=&
DYNCMDSUBST0=QUERY BV_SessionID
DYNCMDPATTERN1=BV_EngineID=
DYNCMDDELIM1=&
DYNCMDSUBST1=QUERY BV_EngineID

```

再生時に置き換えられる URL を次に示します。

```

http://www.shopping.hp.com/cgi-bin/hpdirect/shopping/scripts/
computer_store/
computer_landing_view.jsp?BV_SessionID=@@@@1224306641.107041396
3@@@@&BV_EngineID=xxdeadckdggjfgjgcfngcfkmdflldfgg.0&landing=des
ktops&selectMenu=desktops

```

上記の URL が以下の URL で置き換えられます。

```

http://www.shopping.hp.com/cgi-bin/hpdirect/shopping/scripts/
computer_store/computer_landing_view.jsp?BV_SessionID
=@@@@0573045601.1070412518@@@@&BV_EngineID=ccdeadckdggjfgjgcfngc
fkmdflldfgg.0&landing=desktops&selectMenu=desktops

```

修正されたトランザクションは次のようになります。

```

[HTTP_TRANS]
IESTEP=http://www.shopping.hp.com/
IESTEP=http://www.shopping.hp.com/cgi-bin/hpdirect/shopping/
scripts/computer_store/
computer_landing_view.jsp?BV_SessionID=@@@@1224306641.107041396
3@@@@&BV_EngineID=xxdeadckdggjfgjgcfngcfkmdflldfgg.0&landing=des
ktops&selectMenu=desktops
DYNCMDPATTERN0=BV_SessionID=
DYNCMDDELIM0=&
DYNCMDSUBST0=QUERY BV_SessionID

```

```
DYNCMDPATTERN1=BV_EngineID=  
DYNCMDDELIM1=&  
DYNCMDSUBST1=QUERY BV_EngineID
```

値を返すスクリプトの指定

特定の PATH、POST、QUERY に対して、値を返すスクリプトを設定することができます。HTTP_TRANS プローブは、前に処理した Web ページから値を取得することができるだけでなく、スクリプトからも値を取得することができます。そのため、URL、POST、QUERY の文字列の一部をスクリプトが生成する値で置き換えることができます。

たとえば、日付 / 時刻要素が POST データの一部になっていて、将来の特定の時刻に合わせて調整することが必要になる場合には、計算を行って値をプローブに返すようなスクリプトを開発することができます。これは IE モードの FORMDATA のスクリプトの実行に似ています。

要素

```
dynCmdScript1=getid.sh  
dynCmdSubst1=QUERY sessionid
```

sessionid パラメータの値は、スクリプト getid.sh の出力によって置き換えられます。絶対パスが指定されていない場合には、スクリプトはディレクトリ %probes で実行されます。

代替規則の要素と位置インデックス

これらの規則は、[**ステップのプロパティ**] ダイアログボックスの [**詳細**] タブに追加することができます。このダイアログボックス内に新しい行を追加するには、[**CTRL**] キーを押しながら [**ENTER**] キーを押します。

規則は、規則要素の末尾についているインデックスでグループ化されます。たとえば、dynCmdPattern0、dynCmdDelim0、dynCmdSubst0 は同じグループに属します。



パターンをテストするには、動的 URL を生成する前のページまで、トランザクションを 1 つのステップずつ実行します。[**ツール**] メニューの [**パターンのテスト**] を選択すると、現在のドキュメントのダイアログボックスが表示されます。ここでパターンを指定して、テストすることができます。

代替規則の要素には、タイプと置換される変数 / 位置を指定します。以下の値が使用可能です。

QUERY <クエリー文字列内の変数>

PATH <位置インデックス>

POST <POST データ内の変数>

HOST <URL のホスト部分>

例:

```
dynCmdSubst0=QUERY pageGenTime
dynCmdPattern1=AUTHMETHOD' value='
dynCmdDelim1='>dynCmdSubst1=POST AUTHMETHOD
dynCmdPattern2=pageGenTime' value='
dynCmdDelim2='>dynCmdSubst2=PATH 1
```

位置インデックスには負の値も許されます。その場合には、右から数えた要素の位置を指定します。

例:

```
http://www.dyn.hp.com/first/second/third
```

表 2 位置インデックス

0	パス全体の代替 (/first/second/third)
-1	一番右の要素。(この例では、"third")
-2	右から 2 番目の要素。(この例では、"second")
-3	右から 3 番目の要素。(この例では、"first")
1	最初の要素。(この例では、"first")
2	最初から 2 番目の要素。(この例では、"second")
3	最初から 3 番目の要素。(この例では、"third")

高度な代替処理

標準の動的な代替規則では置き換えができない場合には、別の形式オプションを使うことができます。たとえば、以下の URL を例にします。

```
"http://www.dyn.hp.com/dynamic/test&arg=login;sesssionid=123455=";
```

URL の最後のパス要素内の変化するセッション ID

(test&arg=login;sesssionid=123455=;) をプローブが処理できるようにするには、次のような形式オプションを持つ規則を使う必要があります。

```
DynCmdPattern0=test&arg=login  
DynCmdDelim0="  
DynCmdSubst0=PATH -1=test&arg=login%s
```

これは、最後の位置のパターン“test&arg=login”を検索し、見つかった場合には、その後続く文字列（この例では、sessionid）を置き換えることを意味します。DynCmdSubst0 パラメータは、IEStep で指定されるパスの最後のパス要素を、等号の後に続く文字列 (test&arg=login%s) で置き換えます。文字列を実際に置き換える前に、%s をパターンとデリミタによって指定される文字列で置き換えます。たとえば、パターンとデリミタによって ";sessionid=xyz=" が見つかった場合、新しい URL は次のようになります。

```
http://www.dyn.hp.com/dynamic/test&arg=login;sesssionid=xyz=";
```

同様に、POST でも検索文字列を使うことができます。POST が次のような場合には、規則には検索文字列が必要になります。

```
post=Login=Guest&Password=test;sid=123455;IgnorePassword=false&
```

検索文字列は、次のように構成できます。

```
DynCmdPattern0=Password=test  
DynCmdDelim0=;  
DynCmdSubst0=POST Password=test%s
```

フォームデータ入力用の外部スクリプト (IE モード)

IE モードでは、スクリプトを使って入力変数に値を設定することができます。たとえば、時刻表の開始日の入力が必要になる、時刻表の Web アプリケーションを考えます。このアプリケーションでは、過去の日付やかなり先の日付はサポートされていません。そこで、有効な日付を計算するスクリプトを作成します。このスクリプトをフォーム要素を使って指定できます。

例：

Web Transaction Recorder が以下のようなトランザクションステップを記録したとします。

```
IESTEP=http://www.travel.hp.com/schedule?get  
FORMDATA=date1=05-23-2001  
FORMDATA=date2=06-01-2001
```

入力要素 date1 と date2 に対するスクリプトを指定するには、入力要素名の前に @ を追加し、スクリプトを使って値を置き換えます。

```

IESTEP=http://www.travel.hp.com/schedule?get
FORMDATA=@date1=c:¥temp¥t.bat
FORMDATA=@date2=cscript //nologo t.vbs

```

スクリプトが相対値で指定されている場合には、OVIS を使ってプローブロケーションにスクリプトを配布することをお勧めします

(<install dir>¥newconfig から

<data dir>¥bin¥instrumentation¥probe¥scripts へ)。Web Transaction Recorder は、PATH 環境変数に

<data dir>¥bin¥instrumentation¥probe¥scripts ディレクトリを追加し、オペレーティングシステムがスクリプトの場所を特定することができるようにします。また、OVIS_SCRIPTS 環境変数も

<data dir>¥bin¥instrumentation¥probe¥scripts ディレクトリを指すので、次のように指定することもできます。

```
cmd /c perl "%OVIS_SCRIPTS%¥chk.pl" OpenView
```

これは、スクリプトエンジンが、プログラムスクリプトの場所を示すのに、PATH 変数を使っていないときには必要な操作です。

または、スクリプトを <install dir>¥bin ディレクトリに置いておくこともできます。ただし、ファイル名が他の OpenView 製品と一致してしまう可能性があります。

翌日の日付を返す Visual Basic スクリプトの例は、以下のとおりです。

```

Set WshShell = WScript.CreateObject("WScript.Shell")
dteCurrent = Now()+1
dteDay = Day(dteCurrent)
dteMonth = Month(dteCurrent)
dteYear = Year(dteCurrent)
MyDate = dteDay & "-" & dteMonth & "-" & dteYear
Wscript.echo MyDate

```

可用性チェック用外部スクリプト (IE モード)

IE モードのプローブは、高度な可用性チェックを行う外部スクリプトの実行をサポートしています。通常、ステップの可用性は、指定したパターン、タイムアウト値、HTTP ステータスコード (200 以外) によって決められます。

さらに複雑な Web ページの可用性チェックのために、スクリプトを使用することができます。プローブは Web ページ (すべてのフレームを含む) の HTML が含まれたファイル名をスクリプトに渡し、利用可能 ("1") または利用不可 ("0") を示す戻り値を読み取ります。



パターンとスクリプトを指定した場合には、可用性チェックでは、パターンは無視されて、スクリプトだけが使われます。

スクリプトが相対値で指定されている場合には、OVIS を使ってプローブロケーションにスクリプトを配布することをお勧めします

(<install dir>%newconfig から

<data dir>%bin%instrumentation%probe%scripts へ)。Web Transaction Recorder は、PATH 環境変数に

<data dir>%bin%instrumentation%probe%scripts ディレクトリを追加し、オペレーティングシステムがスクリプトの場所を特定することができるようにします。また、OVIS_SCRIPTS 環境変数も

<data dir>%bin%instrumentation%probe%scripts ディレクトリを指すので、次のように指定することもできます。

```
cmd /c perl "%OVIS_SCRIPTS%chk.pl" OpenView
```

これは、スクリプトエンジンが、プログラムスクリプトの場所を示すのに、PATH 変数を使っていないときに必要な操作です。

または、スクリプトを <install dir>%bin ディレクトリに置いておくこともできます。ただし、ファイル名が他の OpenView 製品と一致してしまう可能性があります。

例:

[**ステップのプロパティ**] ダイアログボックスの [**詳細**] タブで、記録済みステップに、checkAvailability スクリプト文を追加することができます。

```
checkAvailability=perl match.pl OpenView
```

match.pl スクリプトは、次のような Perl スクリプトです。

```
-----  
open FH,$ARGV[1];  
undef $/;  
if (defined($block = <FH>)) {  
    if ($block =~ /$ARGV[0]/) {  
        print "1";  
    }  
    else {
```

```

        print "0";
    }
}
else {
    print "0";
}
close FH;
unlink($ARGV[1]);
-----

```

プローブは、コマンド行 `perl match.pl OpenView` の後に、**Web** ページの **HTML** を含むファイル名を追加します。match.pl スクリプトでは、引き数 0 は文字列 "OpenView" であり、引き数 1 は **HTML** テキストを含むファイル名です。スクリプトは **HTML** テキストを読み取ります。スクリプトは、入力ファイルにパターン "OpenView" を検出すると、1 を出力します。検出しなかった場合には、0 を出力します。

最後に、スクリプトはファイルを閉じて、**HTML** を含むファイルを削除します。



このファイルを削除することは重要です。プローブはこのファイルを削除しません。

動的なフレーム名

一部のフレームウィンドウは **JavaScript** によって作成されるため、フレーム名が無作為に生成される可能性があります。フォームデータがこのフレーム内に記録された場合、再生時に問題が発生することがあります。この場合、次のようなエラーメッセージが表示されます。

```
エラー： フレーム "1234567890" が見つかりません。
```

動的なフレーム名を持つトランザクションステップの例は、以下のとおりです。

```

IENAVPNT=A "1234567890" "252" "button"
"javascript:addToCartUsingToggle();"
FORMDATA=1234567890.items.item=Submit=True

```

IENAVPNT 文と FORMDATA 文では、**items** フォームが 1234567890 フレーム上にあります。フォーム名はトランザクション再生時には変化するので、プローブはフォームデータの場所を検出することができなくなっています。

この問題を回避するには、フレーム名を削除する（未定義のままとする）ことにより、最上位のフレームを使用するようにするか、フレームのインデックスを指定します。

例 A:

```
IENAVPNT=A "" "252" "button"  
"javascript:addToCartUsingToggle();"   
FORMDATA=.items.item=Submit=True
```

例 B:

```
IENAVPNT=A "[2]" "252" "button"  
"javascript:addToCartUsingToggle();"   
FORMDATA=[2].items.item=Submit=True
```

例 A では、IENAVPNT と FORMDATA 内のフレーム名は空です。これはプローブに最上位のフレームを使うことを指示しています。例 B では、正しいフレームは、フレームコレクションオブジェクトにあるフレーム 2 です。フレーム番号を調べるには、Web Transaction Recorder のメインウィンドウの [表示] メニューの [ソースの表示 (DOM)] を選択して、すべての HTML 要素のインデックスとフレーム名を表示します。

プローブのログオンアカウントの変更

デフォルトでは、IE モードのプローブは、システムアカウントで動作します。このアカウントは通常はトランザクションを記録するアカウントとは異なります。すべての Cookie とクライアント証明書には、アカウントが関連付けられています。そのため、トランザクションを実行すると、異なる動作が観察される場合があります。トランザクションを異なるアカウントで実行するには、[コントロールパネル] の [サービス] アプレットを実行して、HP Internet Services アカウントの起動パラメータを変更します。記録を行うアカウントのユーザー名とパスワードを入力します。Web Transaction Recorder を起動したときに、最初のダイアログの [実行ユーザー情報] フィールドに、ユーザーを指定しておくこともできます。

Cookie の設定

一部の Web サイトでは、さまざまな設定 (言語の設定など) を保存したり、ユーザーを最初のページから別のページへとリダイレクトしたりするために Cookie を使用します。一貫性のあるトランザクションの結果を取得するには、[

設定 ダイアログボックスで **[Cookie を無効にする (IE)]** フラグを設定することをお勧めします。このフラグは、スケジューラに対して、トランザクションプロンプトを実行する前にすべての Cookie を削除することを指示します。



これはグローバルフラグであり、個々のトランザクションで選択することはできません。

セッション Cookie の読み取りと保存

[ステップのプロパティ] ダイアログボックスの **[詳細]** タブで、SAVESESSION 文または LOADSESSION 文を指定することによって、セッション Cookie をファイルに保存したり、読み取ったりすることができます。78 ページの「トランザクションスクリプトのリファレンス」を参照して、表の中の SAVESESSION 文を参照してください。

認証

Web Transaction Recorder はユーザー名とパスワードで保護されている Web ページの認証ダイアログボックスを取り扱うため、通常の IE 認証ダイアログボックスは表示されません。そのため、ステップは失敗します。ステップが失敗したらステップをダブルクリックして、**[ステップのプロパティ]** ダイアログボックスでユーザー名とパスワードの情報を入力します。トランザクションを再生します。トランザクションの再生後に、記録ボタンをクリックして、記録を再開します。

ブラウザプラグイン

ブラウザプラグイン (Macromedia FLASH など) のナビゲーションは、IE ステップとして手作業で追加する必要があります。

そのためには、まず、Web Transaction Recorder のトレースレベルを 1 に設定します。これにより、ブラウザがナビゲートしている URL 文字列と POST データが表示されます。たとえば、ログに次のように表示されます。

```
Loading: url='http://ww.hp.com/', post=''
```

ページの読み込みが完了したら、新しいステップを作成します (トランザクションステップのリストボックスで右クリックして、ポップアップメニューから **[追加]** を選択します)。ログ内に存在する URL 文字列と POST データを新しく作成したステップにコピーします。

複雑なページのリダイレクション

一部の Web アプリケーションでは、ログイン検証やアカウント検索時に他のページにリダイレクションを行います。このようなアプリケーションでは、Web Transaction Recorder は複数のリダイレクションを行っているうちに、トランザクションステップを未完了のまま終了することがあります。

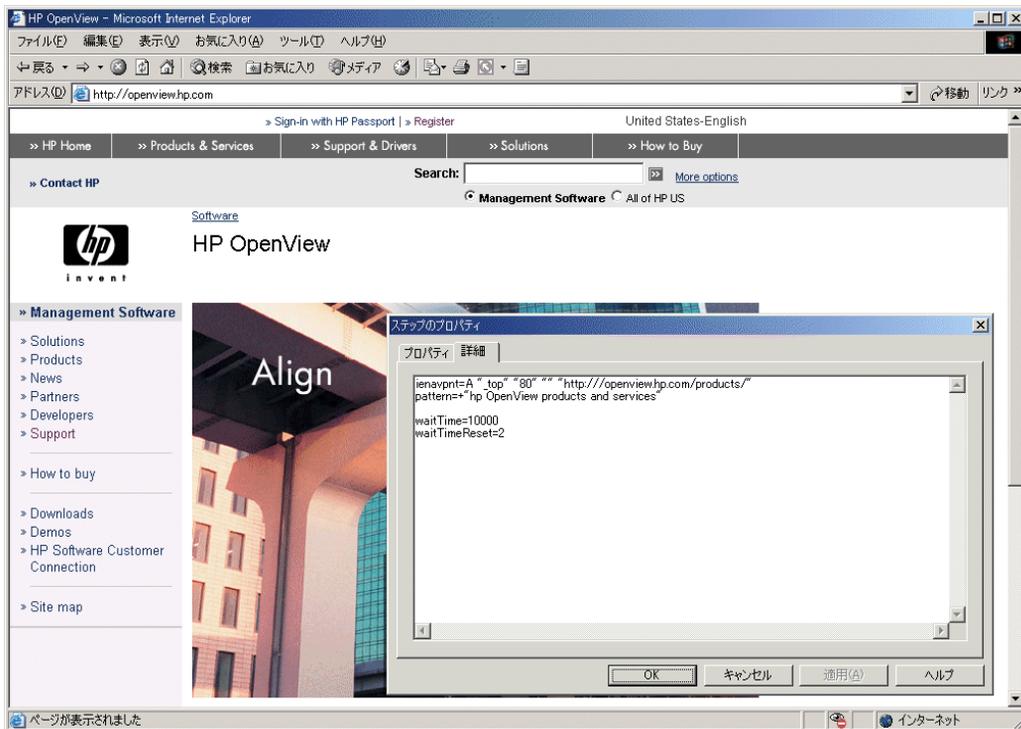
Web Transaction Recorder はデフォルトでは 1.5 秒のタイムアウト値を使って、ページの読み込みの完了を判断します。ページリダイレクションが、このタイムアウト値では不十分な場合は、記録が終わった後に、waitTime 値を追加してタイムアウト値を増加させます。[**ステップのプロパティ**] ダイアログボックスの [**詳細**] タブで、waitTime と waitTimeReset を追加します。waitTimeReset 値は、タイムアウト値を元に戻すリダイレクションの実行回数を指定します。

例:

リダイレクションを検出するための待機時間に 10 秒を設定するには、該当するステップを右クリックして、ポップアップメニューから [**プロパティ**] を選択します。[**詳細**] タブをクリックして、ステップの詳細プロパティを表示します。ダイアログボックスの最後に、"waitTime=10000" を追加します (新しい行を追加するには、[**CTRL**] キーを押しながら [**ENTER**] キーを押します)。waitTime プロパティは、ミリ秒単位の値で指定する必要があります。

または、ステップがサーバーからダウンロードするドキュメントの正確な数を指定して、その数を使ってトランザクションステップの終了を確定するようにプローブに指示することもできます。ダウンロードされた各ドキュメントについては、トレースにより DocCompletes カウンター (例 :DocCompletes: 31) が表示されます。プローブに、これらの DocCompletes の正確な数を使って、トランザクションステップの終了を確定するように指示することができます。[**ステップのプロパティ**] ダイアログボックスの [**詳細**] タブで、"docCompletes=<トレース終了のドキュメント数>" を入力します。

以下の例では、Web ページに2つのリダイレクションがあるので、待機時間には、最初のリダイレクションの後では10秒を設定し、2番目のリダイレクションの後では1.5秒(グローバルなデフォルト)を設定しています。



重複する入力変数

フォームに名前がない場合には、同じ入力要素を持つ複数の `formData` 文が作成され、誤ったナビゲーションの原因になります。これは、[ステップのプロパティ] ダイアログボックスの [詳細] タブで確認できます。

例:

```

formData=_top..index=music
formData=_top..group=e-card-recommendations=True
formData=_top..index=foo
formData=_top..group=bar=True
  
```

この例では、2つの名前のないフォームが、入力要素で同じ名前を使っており、ナビゲーションで問題を起こします。回避策は、問題を起こす formData 文を削除することです (プローブは入力要素を上から下の順番で設定するので、通常は最後の文を削除)。

プローブの結果に影響を与えるレジストリの設定 (IE モード)

IE の [**インターネット オプション**] > [**詳細設定**] タブ、および [**インターネット オプション**] > [**全般**] > [**設定**] ボタンで表示される設定画面での特定のフラグの設定は、IE のコンテンツのダウンロード方法に影響を与えます。

これらの特定のフラグはユーザーごとに保存されるので、スケジューラをシステムアカウント (デフォルト) で実行する場合は、IE モードのプローブではレジストリを変更してこれらの特定のフラグを有効にする必要があります。

IE はこれらの特定のフラグを、以下の2つのレジストリエントリに格納します。

```
HKEY_USERS¥.DEFAULT¥Software¥Microsoft¥Windows¥CurrentVersion¥Internet Settings
```

```
HKEY_USERS¥.DEFAULT¥Software¥Microsoft¥Internet Explorer
```

他のフラグも Microsoft 社のサポートページ上で、または以下のキーの下にある、現在のユーザーのキーを調べることにより見つけることができます。

```
HKEY_CURRENT_USER¥Software¥Microsoft¥Windows¥CurrentVersion¥Internet Settings
```

```
HKEY_CURRENT_USER¥Software¥Microsoft¥Internet Explorer
```

以下のレジストリキーは、IE がコンテンツをダウンロードする方法に影響を与えます。

- [**インターネット オプション**] > [**全般**] タブ > [**設定**] ボタンで表示される [**保存しているページの新しいバージョンの確認**]

以下のキーの下に、SyncMode5 DWORD キーを追加します。

```
HKEY_USERS¥.DEFAULT¥Software¥Microsoft¥Windows¥CurrentVersion¥Internet Settings
```

SyncMode5 の値のデータ値は、以下のとおりです。

- 0: 確認しない
- 2: Internet Explorer を起動するごとに確認する

3: ページを表示するごとに確認する

4: 自動的に確認する

- **[インターネット オプション]>[詳細設定] タブの [HTTP 1.1 を使用する] と [プロキシ接続で HTTP 1.1 を使用する]**

以下のキーの下に、EnableHttp1_1 DWORD キーと ProxyHttp1.1 DWORD キーを追加します。

```
HKEY_USERS¥.DEFAULT¥Software¥Microsoft¥Windows¥CurrentVersion¥Internet Settings
```

このキーのデータ値は、以下のとおりです。

1: 有効

- **[インターネット オプション]>[詳細設定] タブの [スクリプトのデバッグを使用しない]**

以下のキーの下に、文字列値 "Disable Script Debugger" を追加します。

```
HKEY_USERS¥.DEFAULT¥Software¥Microsoft¥Internet Explorer¥Main
```

このキーのデータ値は、以下のとおりです。

yes: スクリプトデバッガを無効にする

no: スクリプトデバッガを有効にする

- **[インターネット オプション]>[詳細設定] タブの [ブラウザを閉じたとき、[Temporary Internet Files] フォルダを空にする]**

以下のキーの下に、"Persistent" DWORD キーを追加します。

```
HKEY_USERS¥.DEFAULT¥Software¥Microsoft¥Windows¥CurrentVersion¥Internet Settings¥Cache
```

"Persistent" キーの値は、以下のとおりです。

1: Temporary Internet Files フォルダを空にしない

0: Temporary Internet Files フォルダを空にする

HTML ダイアログとプルダウンメニューの内部処理

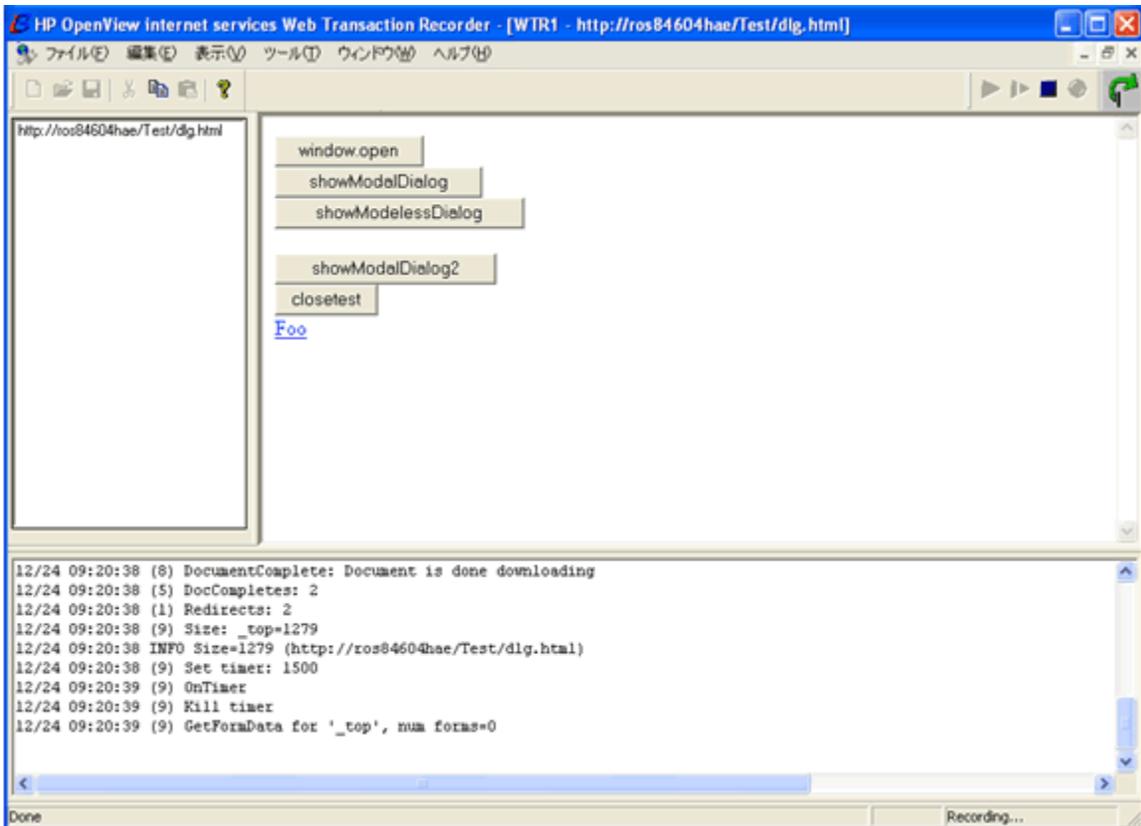
HTML ダイアログ

HTML ダイアログには、以下の JavaScript 機能を介して開かれるモーダルダイアログとモードレスダイアログがあります。

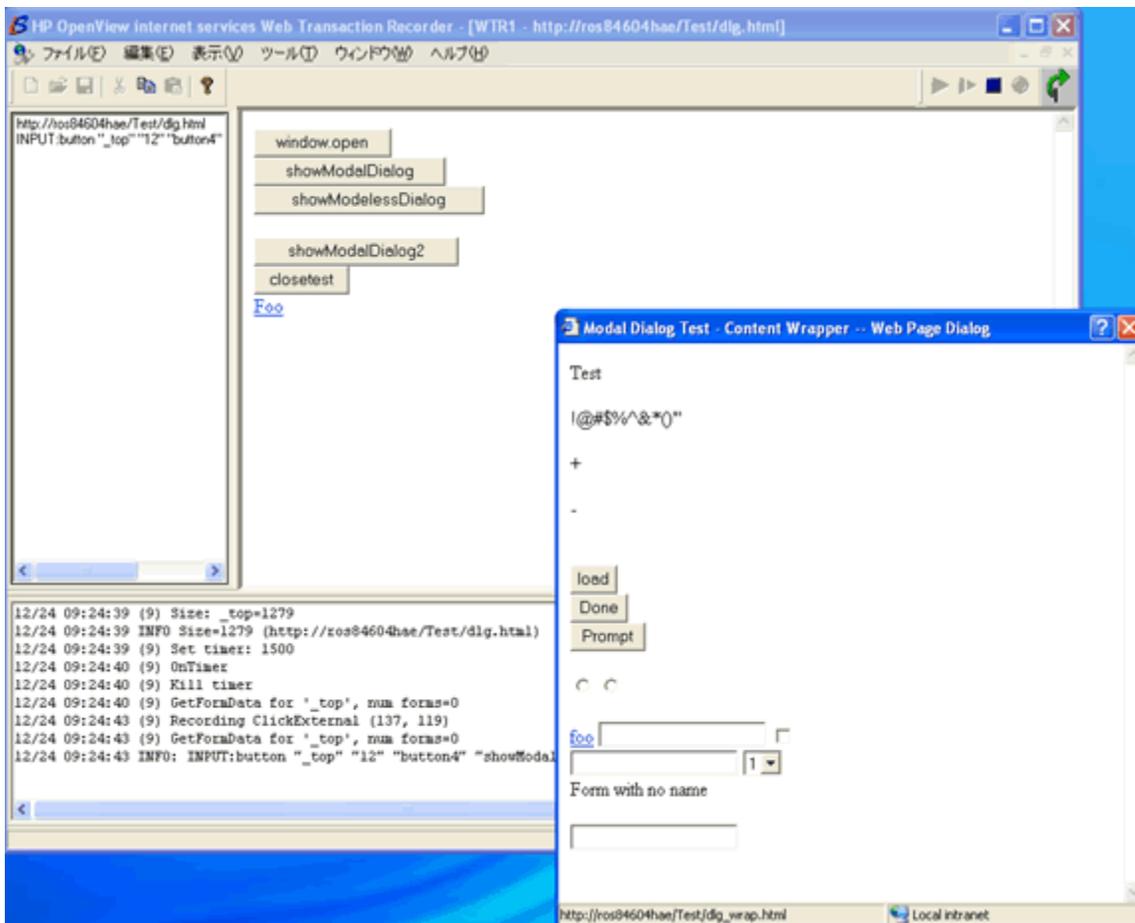
```
window.showModelessDialog("http://www.hp.com");  
window.showModalDialog("http://www.hp.com");
```

この2つの関数を使用して、アプリケーションは指定した HTML ドキュメントを表示するモードレスダイアログボックスまたはモーダルダイアログボックスを作成することができます。この関数を使用して開かれるウィンドウは IE オブジェクトモデルには含まれないため、Web Transaction Recorder はこれらのダイアログを MDI (複数文書インタフェース) ビューの外側に新しいウィンドウとして表示します。

次の画面は、モーダルダイアログを開くトランザクションを示しています。Web Transaction Recorder は、開始 URL に移動した後に、ダイアログを開くことを可能にするテストページウィンドウが MDI ビュー内に表示されます。

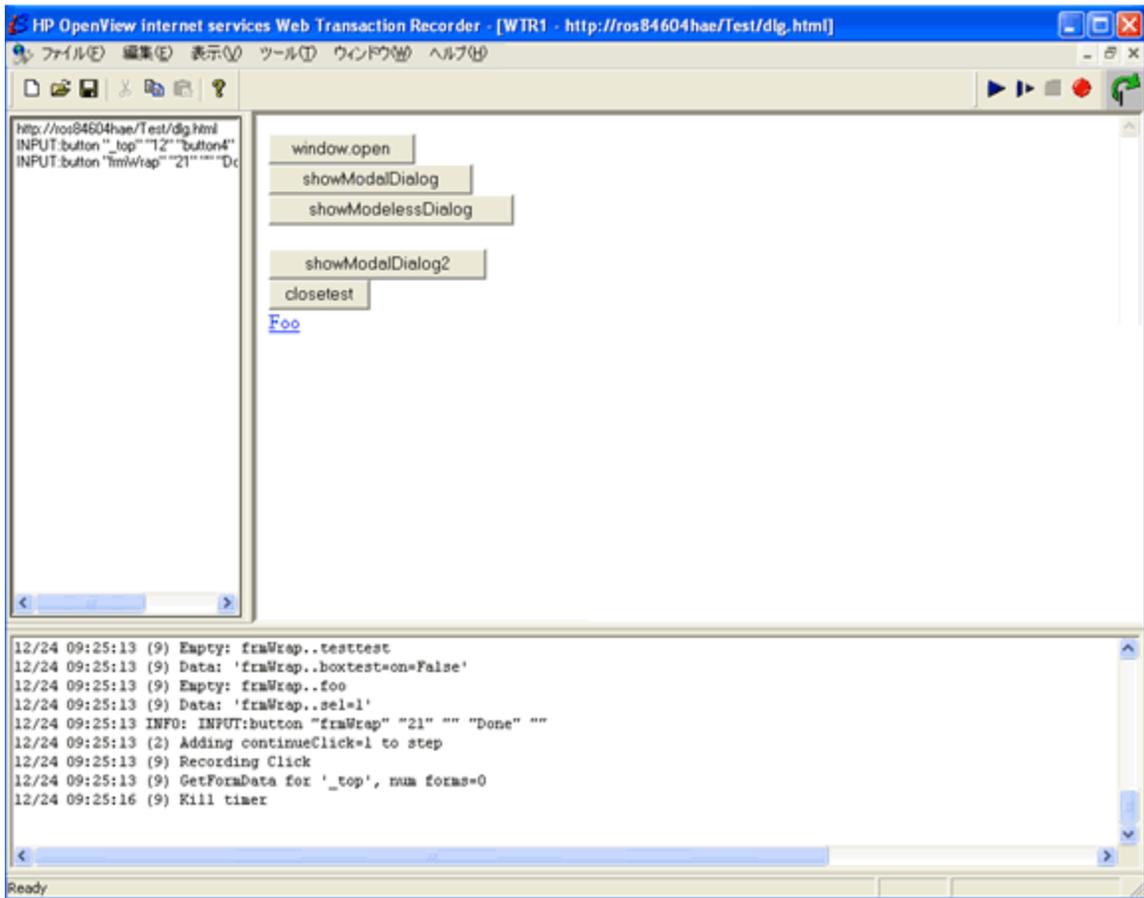


[**showModalDialog2**] ボタンをクリックすると、Web Transaction Recorder ビューの外側に、モーダル HTML ダイアログウィンドウ [**Modal Dialog Test ...**] が表示されます。ダイアログを表示するコードは、`"window.showModalDialog("dlg_wrap.html");"` です。



HTML ダイアログウィンドウの **[Done]** ボタンをクリックすると、ダイアログが閉じます (ボタンの onclick ハンドラーの "window.close()")。

Web Transaction Recorder はウィンドウが閉じたことを検出し、ステップにこのアクションを記録します。



完全なスクリプトは、以下のとおりです。

[HTTP_TRANS]

FLAGS=cache=0,silent=0,noNewWindow=0,waitTime=1500,incWaitTime=0
 ,runJava=1,aboutBlankInit=1,IgnoreExitExceptions=0,captureWindow
 =0

ステップ1 (最初の URL):

IESTEP=http://ros84604hae/Test/dlg.html

PATTERN="+Html Dialog Test Page"

ステップ2 ("showModalDialog2" をクリック):

```
IENAVPNT=INPUT:button "_top" "12" "button4" "showModalDialog2"
""
PATTERN="+Modal Dialog Test - Content Wrapper"
```

ステップ3 ("Done" をクリック):

```
IENAVPNT=INPUT:button "frmWrap" "21" "" "Done" ""
FORMDATA=frmWrap.form2.CARS=on=False
FORMDATA=frmWrap.form2.CARS=on=False
FORMDATA=frmWrap..boxtest=on=False
FORMDATA=frmWrap..sel=1
```

CONTINUECLICK=1

HTMLDIALOG=Modal Dialog Test - Content Wrapper -- Web Page Dialog

ダイアログ内の [Done] ボタンをクリックすると、Web Transaction Recorder はステップに次の2つのパラメータを追加します。

- **HTMLDIALOG=<HTML ダイアログウィンドウのキャプション>**
- **CONTINUECLICK=1** (ページナビゲーションを待たないことの指示)

HTMLDIALOG パラメータは、再生時に、Web Transaction Recorder に対して、指定したキャプション (ウィンドウのタイトル) を持つ **HTML** ダイアログウィンドウを検出する必要があることを指示します。ダイアログウィンドウは **IE** オブジェクトモデルに含まれないため、このようなウィンドウが作成されたことが、**IE API** から **Web Transaction Recorder** には通知されないため、この処理は必要です (標準の "window.open()" 呼び出しのみが、**IE** をホストするアプリケーションに対して "NewWindow" 通知を生成します)。Web Transaction Recorder は、再生時にウィンドウを検出するために、現在のプロセスに属するすべてのウィンドウを調べて、**HTMLDIALOG** パラメータで指定されたキャプションを持つウィンドウを探し出します。Web Transaction Recorder は、ウィンドウを検出した後は、ウィンドウのオブジェクトモデルにアクセスして、ステップ内の指示 (すなわち、フォームへの値の設定とボタンのクリックのシミュレーション) を実行することができます。

CONTINUECLICK=1 パラメータは必要です。**HTML** ダイアログを閉じて、通知が発行されないためです。通常では、Web Transaction Recorder は、現在のステップが終了し、新しいステップを開始することを "BeforeNavigate" イベントの

通知を通じて検出します。HTML ダイアログは "BeforeNavigate" イベントを生成しないため、Web Transaction Recorder にイベントを待たずにそのまま次のステップに移ることを指示するために、このパラメータを追加します。

動的に値を取り込むプルダウンメニューに対する特殊な処理

選択された値に基づいてサーバーから動的に情報を取り込むプルダウンメニューには、特殊な処理が必要になります。以下の例を考えます。

Selection 1:

Selection 2:

Selection 3:

上の例では、プルダウンメニュー **[Selection 1]** の選択に従って、**[Selection 2]** の項目が設定されます。

Selection 1:

Selection 2:

Selection 3:

[Selection 2] の項目を選択すると、プルダウンメニュー **[Selection 3]** の項目が設定されます。

Selection 1:

Selection 2:

Selection 3:

[**Selection 3**] の項目を選択して [**Submit**] ボタンをクリックすると、フォームがサーバーに送られます。

標準的な記録では、プルダウンメニューがすべて選択されて、[**Submit**] ボタンがクリックされたときに、1つのステップで以下の値が記録されます。

```
INPUT:image "_top" "296" "Submit"
```

Web アプリケーションでは各プルダウンメニューが別々に選択されることが必要なので、再生は**失敗**します。しかし、Web Transaction Recorder は、次のように、[**Submit**] ボタンがクリックされる前に、すべてのプルダウンメニューの値を設定します (formdata パラメータ)。

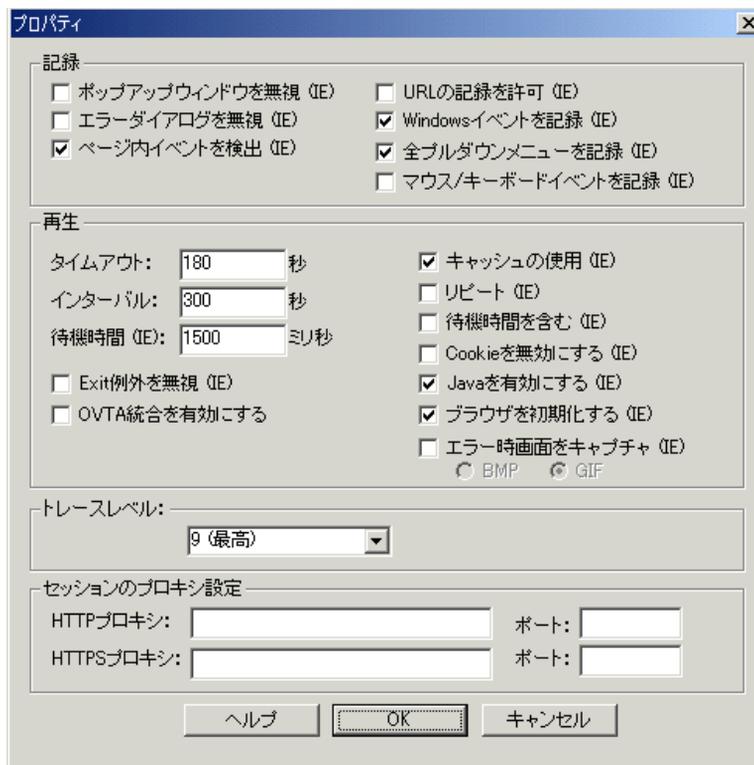
```
ienavpnt=INPUT:image "_top" "296" "submit" "Submit" "http://server/app/submit.gif"
```

```
formData=_top.AForm.Selection1=1 - Value B
```

```
formData=_top.AForm.Selection2=2 - Value A
```

```
formData=_top.AForm.Selection3=3 - Value C
```

Web Transaction Recorder でプルダウンメニューの各動作をシミュレートするには、[**ファイル**] メニューの [**設定**] で [**全プルダウンメニューを記録**] オプションを設定しておく必要があります。



このオプションを選択し、ページへの記入が終わると、そのトランザクションでは、次のように、各プルダウンメニューに対応する追加ステップが記録されます。

```
ienavpnt=SELECT "_top" "125" "Selection1" "1"
```

```
formData=_top.AForm.Selection1=1 - Value B
```

```
ienavpnt=SELECT "_top" "178" "Selection2" "37"
```

```
formData=_top.AForm.Selection1=1 - Value B
```

```
formData=_top.AForm.Selection2=2 - Value A
```

```
ienavpnt=SELECT "_top" "235" "Selection3" "2"
```

```
formData=_top.AForm.Selection1=1 - Value B
```

```
formData=_top.AForm.Selection2=2 - Value A
```

```
formData=_top.AForm.Selection3=3 - Value C
```

上記の3つの"SELECT"ナビゲーションポイントが、最後の送信ステップの前に挿入されていることを確認してください。

そのため、トランザクションを再生すると、Webアプリケーションは最初のSELECTナビゲーションポイントを実行した段階で、エラーを返します。

SELECTナビゲーションポイントでのデフォルトの動作は、SELECTナビゲーションポイントが含まれているHTMLフォームのアクションを実行します。Webページのフォームアクションは、"Submit"ステップを処理するときに、データをサーバーに送信するので、Webアプリケーションは、プルダウンメニューのすべての情報は取得することができません。

SELECTナビゲーションポイントのデフォルトの動作は、SELECTテキストの最後に"nosubmit"を追加することで変更することができます。"nosubmit"フラグはWeb Transaction Recorderに対して、プルダウンメニューを設定するだけで、フォームアクションは実行しないように指示します。

"nosubmit"に変更するだけでなく、ナビゲーションポイントに"continueClick=1"を追加し、対象プルダウンメニューに対応するformdata文以外のすべてのformdata文を削除することも必要です。

3つのSELECTナビゲーションポイントを変更すると、次のようになります。

```
ienavpnt=SELECT "_top" "125" "Selection1" "1" "noSubmit"
```

```
formData=_top.AForm.Selection1=1 - Value B
```

```
continueClick=1
```

```
ienavpnt=SELECT "_top" "178" "Selection2" "37" "noSubmit"
```

```
formData=_top.AForm.Selection2=2 - Value A
```

```
continueClick=1
```

```
ienavpnt=SELECT "_top" "235" "Selection3" "2"
```

```
formData=_top.AForm.Selection3=3 - Value C
```

```
continueClick=1
```

これでトランザクションがプルダウンメニューを正しく再生できるようになります。

位置独立ナビゲーションポイント

ナビゲーションポイント (`ienavpnt`) がインデックスや名前のような属性で一意に特定できない場合には、以下の手順を使って、正しいナビゲーションポイントの場所を特定します。

以下の Web ページを考えます。

```
<a href=x1.html>X1</a><br>
```

```
<a href=x2.html>X2</a><br>
```

```
<a href=x3.html>X3</a><br>
```

Web Transaction Recorder で X2 をクリックすると、以下のナビゲーションポイントが記録されます。

```
ienavpnt=A "_top" "6" "" "http127.0.0.1/Test/moving/x2.html"
```

情報リストが返されてくるときに Web ページが絶えず変化している場合には、再生では正しいリンクが再生できません。たとえば、上記の Web ページが、次のように変化したとします。

```
<a href=x1.html>X2</a><br>
```

```
<a href=x2.html>X1</a><br>
```

```
<a href=x3.html>X3</a><br>
```

このような場合には、再生は失敗します。

この状況を回避するには、ナビゲーションポイントを、指定されたマークアップを検索して見つかったものと置き換え、マークアップが含まれる要素でクリックを実行します。上記の例では、ナビゲーションポイントは次のように変更することができます。

```
ienavpnt=!Find A "_top" "X2"
```

これは再生時に、フレーム "_top" のすべてのリンク ("A" タグ) で X2 マークアップを検索することを指示しています。ナビゲーションポイントで指定したマークアップが、大文字と小文字の区別なしで一致する場合には、Web Transaction Recorder はマークアップを含む要素でクリックを実行します。

!Find の他に、マークアップを標準出力 (stdout) に出力するスクリプトを実行する !FindEx があります。

```
ienavpnt=!FindEx "A" "top" "cmd /c c:¥temp¥ffindex.bat"
```

トランザクションスクリプトのリファレンス

以下の表では、複雑な Web ページの処理で使うことができる各種のスクリプト文を詳しく説明しています。スクリプト文は、トランザクションを記録した後に、[**ステップのプロパティ**] ダイアログボックスの [**詳細**] タブで入力することができます。

IE モード

表 3 IE モードのスクリプト文

文	引き数	説明
iestep	URL	<p>指定された URL に移動します。</p> <p>例：</p> <pre>iestep=http://www.hp.com/go/openview</pre> <p>デフォルトでは iestep で指定された URL はメインフレーム (_top) に読み込まれます。Web アプリケーションによっては、URL を読み込むフレームを指定する必要があります。フレームを指定するには、frame 文を使用します。</p> <p>例：</p> <pre>iestep=http://www.hp.com/go/ frame=salesFrame</pre>
frame	フレーム名	<p>iestep で指定した URL を読み込むフレームを指定します。この文はコンテンツを特定のフレームに読み込むために使用します。</p>

表3 IEモードのスク립ト文(続き)

文	引き数	説明
ienavpnt	IE ナビゲーションポイント	<p>現在の URL 内でナビゲーションポイントを見つけ、検出された HTML 要素でのユーザーのクリックをシミュレートします。</p> <p>形式:</p> <pre>A "frame" "index" "name" "url" AREA "frame" "index" "url" INPUT:type "frame" "index" "name" "value" "url"</pre> <p>(type は BUTTON、IMAGE、SUBMIT のいずれかです)</p> <p>文の最初の部分に、HTML タグを指定し、それに続けて、要素が位置するフレーム (frame) を指定します。index には、要素の場所を指定します。(インデックスによって指定された場所に要素がなかった場合には、自動検索が行われます)。name には、name 属性の値を指定します。url には、要素に含まれる href タグの値を指定します。</p> <p>frame には、名前 (“_top”)、重複したフレーム名の場合にはインデックスが添えられた名前 (“_top[2]”)、またはフレームコレクション内の位置を示すインデックス (“[2]”) のいずれかを指定します。</p> <p>例:</p> <pre>ienavpnt=A "_top" "151" "" "http://www.hp.com/"</pre>

表 3 IE モードのスク립ト文 (続き)

文	引き数	説明
formdata	入力要素	<p>記録時に Web ページに入力されるデータを指定します。スク립トや SELECT 要素内の絶対パスは @ 文字で指定することができます。</p> <p>形式:</p> <pre>frame.form.element=< 値 ></pre> <p>frame には、名前 (“_top”)、重複したフレーム名の場合にはインデックスが添えられた名前 (“_top[2]”)、またはフレームコレクション内の位置を示すインデックス (“[2]”) のいずれかを指定します。</p> <p>例:</p> <pre>formdata=_top.form1.index=blended formdata=_top..group=credit=True formdata=.form2.search=foo formdata=@date1=c:¥temp¥t.bat arg1 arg2 formdata=@date2=cscript //nologo t.vbs formdata=form.sel=@1</pre>
POST	HTTP POST データ	<p>HTTP POST メソッドで使われるデータを指定します (IE モードの iestep のみ)。</p> <p>例:</p> <pre>post=product_oid=427&arg=search</pre>

表3 IEモードのスク립ト文(続き)

文	引き数	説明
pattern	パターン文字列	<p>パターンを指定します。パターンが見つからない場合は、ステップを利用不可とみなします。</p> <p>HTML 内の一部の文字は、コード化されています。たとえば、「&」は「&amp;」、「"」は「&quot;」でコード化されます。パターンを設定する場合には、HTML ソースを見て、ブラウザに表示されている文字列にこのようなコード化された文字がないかどうかを調べます。コード化された文字が文字列に含まれている場合には、それによってパターンを調整します。</p> <p>例： pattern="+HP OpenView"</p>
patternconfig	パターンオプション	<p>単語間のデフォルトの演算子は、"AND" です。この文を使って、"OR" に変更することができます。単語の比較は、デフォルトでは、大文字と小文字が区別されません。"ICASE" を入力することによって、大文字と小文字を区別しないようにすることができます。</p>
username	ユーザー名	<p>HTTP 認証で使用するユーザー名です。</p> <p>例： username=ovros</p>
password	パスワード	<p>HTTP 認証で使用するパスワードです。パスワードは、[ステップのプロパティ] ダイアログボックスの [詳細] タブで入力した場合には、暗号化されています。</p> <p>例： password=artros</p>
proxyUsername	ユーザー名	<p>HTTP プロキシ認証で使用するユーザー名です。</p> <p>例： proxyUsername=secureros</p>

表 3 IE モードのスク립ト文 (続き)

文	引き数	説明
proxyPassword	パスワード	HTTP プロキシ認証で使用するパスワードです。パスワードは、[ステップのプロパティ] ダイアログボックスの [詳細] タブで入力した場合には、暗号化されています。 例： proxyPassword=artros
label	ラベル	トランザクションステップのシンボリック名です。 例： label=Check-out

表3 IEモードのスク립ト文(続き)

文	引き数	説明
dlgCmd	タイプ (Button) ダイアログタイトル ボタンタイトル または タイプ (Sequence) ダイアログタイトル シーケンス	<p>このコマンドを使って、IE ダイアログボックスでボタンをクリックすることができます。以下の2つの形式があります。</p> <ul style="list-style-type: none"> (ダイアログタイトルで指定した)ダイアログボックス内の(ボタンタイトルで指定した)ボタンをクリックします。 <p>例:</p> <pre>dlgCmd=Button "クライアントの認証" "OK"</pre> <ul style="list-style-type: none"> (ダイアログタイトルで指定した)ダイアログボックスで、タブキー、上向き/下向き矢印キー、マウスのクリックのシーケンスを実行します。 <p>形式:</p> <pre>dlgCmd=Sequence "<ダイアログタイトル>" <seq1> <seq2> ...</pre> <p>シーケンス (seq1 と seq2...) には、以下のいずれか1つを指定します。</p> <p>tab=<タブキーをシミュレートする回数(正の数)またはシフト+タブキーをシミュレートする回数(負の数)></p> <p>up=<上向き矢印キーをシミュレートする回数></p> <p>down=<下向き矢印キーをシミュレートする回数></p> <p>click</p> <p>text="現在フォーカスされているテキストフィールドに入力するテキスト"</p> <p>close</p>

表 3 IE モードのスク립ト文 (続き)

文	引き数	説明
		<p>例：以下のシーケンスは、下向き矢印キーを 2 回押し、次のダイアログ要素に移動し、それをクリックする動作をシミュレートします。</p> <pre>dlgCmd=Sequence "クライアントの認証" down=2 tab=1 click</pre>

表3 IEモードのスク립ト文(続き)

文	引き数	説明
dynCmdPattern<idx> dynCmdDelim<idx> dynCmdSubst<idx>	代替規則(パターン、デリミタ、代替処理)	<p>動的 URL に対する代替規則を指定します。文は、<idx>によってグループ化されます。たとえば、dynCmdPattern3、dynCmdDelim3、dynCmdSubst3 文は、同じ規則に属します。</p> <p>パターンは正規表現を使って指定することができます。デリミタは、パターンの直後からデリミタまでのテキストを区切る文字列です。このテキストは、代替処理の指定に応じて、変数またはパス要素を置き換えるために使われます。代替処理の引き数は、以下のとおりです。</p> <p>QUERY <クエリー文字列内の変数></p> <p>PATH <位置インデックス></p> <p>POST <POST データ内の変数></p> <p>HOST <URL のホスト部分></p> <p>位置インデックスでは負の数を使って、要素を終わりから数えて指定することができます。たとえば、-1 は最後のパス要素を指します。</p> <p>例：</p> <pre> dynCmdPattern0=pageGenTime' value=' dynCmdDelim0='> dynCmdSubst0=QUERY pageGenTime dynCmdPattern1=AUTHMETHOD' value=' dynCmdDelim1='>dynCmdSubst1=POST AUTHMETHOD dynCmdPattern2=pageGenTime' value=' dynCmdDelim2='>dynCmdSubst2=PATH 1 </pre> <p>標準の動的な代替規則では置き換えができない場合には、別の形式オプションを使うことができます。</p>

表 3 IE モードのスク립ト文 (続き)

文	引き数	説明
		<p>例 :</p> <pre>DynCmdPattern0=test&arg=login DynCmdDelim0=" DynCmdSubst0=PATH -1=test&arg=login%s</pre> <p>これは、最後の位置のパターン “test&arg=login” を検索し、見つかった場合には、その後続く文字列を置き換えることを意味します。DynCmdSubst0 パラメータは、IEStep で指定されるパスの最後のパス要素を、等号の後に続く文字列 (test&arg=login%s) で置き換えます。文字列を実際に置き換える前に、%s をパターンとデリミタによって指定される文字列で置き換えます。</p> <p>同様に、POST でも検索文字列を使うことができます。</p> <p>例 :</p> <pre>DynCmdPattern0>Password=test DynCmdDelim0=; DynCmdSubst0=POST Password=test%s</pre>

表3 IEモードのスク립ト文(続き)

文	引き数	説明
dynCmdScript<idx> dynCmdSubst<idx>	代替規則(スク リプト、代替処 理)	<p>動的 URL に対する代替規則を指定します。文は、<idx>によってグループ化されます。たとえば、dynCmdScript1 文と dynCmdSubst1 文は、同じ規則に属します。</p> <p>スク립トには、代替部分の値を返すスク립トを指定します。代替処理の引き数は、以下のとおりです。</p> <p>QUERY <クエリー文字列内の変数></p> <p>PATH <位置インデックス></p> <p>POST <POST データ内の変数></p> <p>位置インデックスでは負の数を使って、要素を終わりから数えて指定することができます。たとえば、-1 は最後のパス要素を指します。</p> <p>例:</p> <pre>DynCmdScript1=getid.shDynCmdSubst1=POST sessionid</pre>
waitTime	ミリ秒	<p>ステップ間の待機時間をミリ秒単位で指定します。デフォルトは、1500 ミリ秒です。リダイレクション時に再生が正常に行われる前に終了してしまう場合には、この値を増加させます。</p> <p>例:</p> <pre>waitTime=10000</pre>

表 3 IE モードのスク립ト文 (続き)

文	引き数	説明
waitTimeReset	数値	<p>トランザクションステップで、指定された回数だけリダイレクションを実行した後に、waitTime をグローバル値に戻します。</p> <p>例： waitTime=5000 waitTimeReset=2</p> <p>この例では、waitTime は、1 回目のリダイレクション後は 5000 ミリ秒になり、2 回目のリダイレクション後は 1500 ミリ秒になります。</p> <p>リダイレクションの回数を確認するには、記録時のトレースレベルを 1 に設定します。</p>
continueClick	数値	<p>再生時にはブラウザからのイベントを待たないように指示します。たとえば、JavaScript プログラムが、サーバーから URL を読み込むことなく、ページを更新することがあります。このような場合に、continueClick=1 文を使用して、再生を次のトランザクションステップに進めます。</p> <p>例： continueClick=1</p>
noWindowClose	数値	<p>Web Transaction Recorder に対してどのウィンドウも閉じないように指示します。複数のウィンドウを使う Web サイトでは、ユーザーがログアウトしたときにウィンドウを閉じる場合があります。このようなイベント時にトランザクションステップが停止せず、プログレスインジケータが回転し続けている場合には、noWindowClose 文を追加することにより、この問題を回避させることができます。</p> <p>例： noWindowClose=1</p>

表3 IEモードのスク립ト文(続き)

文	引き数	説明
forceWindowClose	数値	IEによってブラウザウィンドウが閉じられないようにします。ウィンドウは Web Transaction Recorder とプローブによって閉じられます。このフラグに1を設定すると、一時的に進行状況を示すウィンドウなど、複数のウィンドウを開いたり閉じたりするサイトが処理できるようになります。 例： forceWindowClose=1
completeCheck	数値	再生時に、トランザクションステップの終了を確定するアルゴリズムを変更するように指示します。再生時にトランザクションステップの終了を検出しない場合に、トランザクションステップにこのフラグを追加します。 例： completeCheck=1 トレースの最後のイベントシーケンスが BeforeNavigate2、DownloadComplete である場合には、completeCheck=2 を使用してください。
noAbout	数値	再生時に、新しいウィンドウを開いた場合、空白ページ (about:blank) が読み込まれないように指示します。再生時に認証を使用している場合には、空白ページを読み込む必要があります。 例： noAbout=1
checkAvailability	スクリプト	Web ページの HTML テキストの可用性チェックを行うスクリプトの名前です。 例： checkAvailability=check.bat

表 3 IE モードのスク립ト文 (続き)

文	引き数	説明
docCompletes	数値	<p>再生時に、指定した回数のドキュメントを読み込んだ後に、トランザクションステップの終了を確定するように指示します。正確な回数は、トレースで調べることができます (DocCompletes: < 数値 >)。</p> <p>例:</p> <pre>docCompletes=31</pre> <p>上記の例では、31 回 DocCompletes を実行してから、トランザクションステップの終了を確定します。</p>
window	ウィンドウ番号	<p>ステップを適用するウィンドウの番号を指定します。複数のウィンドウが開かれる Web アプリケーションを再生する場合には、プローブに対して、ステップでどのウィンドウが使われるのか指示する必要があります。その場合に、この文を使います。</p> <p>ウィンドウには最初に開いたウィンドウから順番に 1 から始まる番号が割り振られます。常に最上位であるウィンドウには、window=0 を指定します。</p> <p>例:</p> <p>最初のページを読み込んだ後に、Web アプリケーションがそのウィンドウを閉じずに別のウィンドウを開きます。(注記: ウィンドウの最大化ボタンをクリックすると、ウィンドウの数が表示されます。)</p> <p>次のステップで 2 番目のウィンドウに対して操作を行う必要がある場合には、そのステップで "window=2" を指定します。</p>

表3 IEモードのSCRIPT文(続き)

文	引き数	説明
enableIEDialog	数値	再生時に、JavaScript の alert() 関数で生成されるポップアップダイアログボックスの表示を自動的に抑止しないように指示します。たとえば、JavaScript 関数がポップアップダイアログボックスで警告メッセージを出力する可能性がある場合には、enableIEDialog=1 文を指定しておけば、そのステップの再生時にダイアログボックスが表示されます。 例： enableIEDialog=1

表 3 IE モードのスク립ト文 (続き)

文	引き数	説明
WaitForDlgCaption	次のステップに進むためのダイアログのキャプション	<p>複雑な Web ページでは、Web Transaction Recorder が次のステップに進むタイミングを判断できない場合があります。このような状況では、通常、ステップに長い待機時間を指定して、先に進む前にすべてのページを確実に読み込ませることが必要になります。この方法は、可用性に影響を与えることはありませんが、そのステップの応答時間を正確に把握することはできません。ステップを終わらせることができたときにダイアログを表示すれば、Web Transaction Recorder は、それを先に進んでいいことの合図として使うことができます。ダイアログが表示されるかどうかは (多く場合、ダイアログは隠れて見えません)、記録時のトレースを有効にして、トレースを調べます。長い待機時間が必要なステップで次のようにダイアログが表示されるとします。</p> <p>Handle Dialogs class=ThunderRT6FormDC and wintext=Filters</p> <p>上記のステップに次のようなコマンドを追加します。</p> <p>WaitForDlgCaption="Filters"</p> <p>Web Transaction Recorder は、指定したダイアログが表示されたことを検出すると、内部タイマーを停止して次のステップに移ることができます。つまり、実際のステップの応答時間を知るための正確な手段として使うことができます。次のステップへの移動が誤って早く行われないように、最後のダイアログに固有のキャプションを表示することをお勧めします。それでも移動が早く行われる場合には、次のステップに delay コマンドを挿入して、ステップの実行に入る前に、ページの読み込みを完了させる時間を設けます。</p>

表3 IEモードのスク립ト文(続き)

文	引き数	説明
Delay	ステップの実行を遅らせるミリ秒単位の数値	ページの実行を少しの時間遅らせることが必要になることがあります。たとえば、前のステップで WaitForDlgCaption コマンドを指定して、ダイアログの表示を新しいステップへ進む契機としたにもかかわらず、そのページで次のステップに進む準備ができていない場合には、このコマンドを使って、ステップの実行を指定したミリ秒間遅らせることができます。この時間はステップまたはトランザクションの応答時間としてはカウントされません。
maximize	数値	再生時にブラウザウィンドウを最大化するかどうかを指定します。デフォルトでは、新しいブラウザウィンドウに maximize が設定されます。Web アプリケーションがページサイズの変更によって問題を起こす場合には、このフラグを使って、再生時にブラウザウィンドウを最大化しないように指示します。 例 : maximize=0 これにより、指定されたステップではブラウザウィンドウを最大化しません。
formInputMode	数値	このパラメータに "2"(formInputMode=2) を設定すると、 Web Transaction Recorder は、ウィンドウイベントを使って、フォーム入力フィールドへ記入します。それによって、フォーム入力要素に関連付けられるキーボード関連のイベントが生成されます。 デフォルトでは、ウィンドウイベントは使用されません。

表 3 IE モードのスク립ト文 (続き)

文	引き数	説明
clickMode	数値	<p>このパラメータに "2"(clickMode=2) を設定すると、Web Transaction Recorder は、ウィンドウイベントを使って、ボタン、イメージなどの HTML 要素のクリックをシミュレートします。</p> <p>デフォルトでは、ウィンドウイベントは使用されません。</p> <p>注記: ウィンドウ内に表示される要素だけが、ウィンドウイベントを受け取ることができます。要素がページ内の非表示領域にある場合には、clickMode=2 を指定しても適用されません。</p>
noContinue	数値	<p>MCLICK 文または KEY 文によって Web サーバーとの相互作用が行われると、Web サーバーがページを更新しないうちに次のステップへ移ってしまう場合があります。ステップの早期終了を防ぐため、MCLICK ステップに次の文を追加します。</p> <pre>noContinue=1 script</pre> <p>noContinue 文は、continueClick 文よりも高い優先順位で実行されます。</p>

表3 IEモードのスク립ト文(続き)

文	引き数	説明
savesession loadsession	Cookie ファイル名	<p>savesession 文があると、IE Cookie キャッシュ内のすべての Cookie が指定したファイルに書き込まれます。このファイルがすでにある場合には、上書きされます。このアクションはステップの最後に実行されます。</p> <p>savesession=cookie C:¥COOKIES.TXT</p> <p>loadsession 文は、ステップを実行する前に、指定したファイル内のすべての Cookie を IE Cookie キャッシュに読み込みます。IE Cookie キャッシュに同じ名前と URL の Cookie が存在している場合には、上書きされません。</p> <p>loadsession=cookie C:¥COOKIES.TXT</p> <p>Cookie のファイル形式は、交換可能なので、たとえば、Cookie を HTTP プローブを使って保存した後、HTTP_TRANS プローブを使って読み込むことができます。</p>

URL/ナビゲーションポイント モード

表 4 URL/ナビゲーションポイント モードのスク립ト文

文	引き数	説明
step	URL	指定された URL に移動します。 例： step=http://www.hp.com/go/openview
dstep	ナビゲーションポイント	プローブによって読み込まれた現在のドキュメント内のナビゲーションポイントの位置を見つけ、URL に変換し、その URL を読み込みます。= と : の間にある識別子は、ナビゲーションポイントのタイプを指定します。 A: アンカー FORM: フォーム (HTTP POST メソッド) FORM_GET: フォーム (HTTP GET メソッド) 例： dstep=A:Desktops and Monitors dstep= FORM:BV_SessionID=550&BV_EngineID=fmg.0 &product_oid=427
pos	インデックス	ナビゲーションポイントのインデックスを指定します。複数の同一のナビゲーションポイントがある場合に、解決する必要があるナビゲーションポイントをインデックスで指定します。
POST	HTTP POST データ	HTTP POST メソッドで使用されるデータを指定します。 例： post=product_oid=427&arg=search

表4 URL/ナビゲーションポイントモードのスク립ト文(続き)

文	引き数	説明
pattern	パターン文字列	<p>パターンを指定します。パターンが見つからない場合は、ステップは利用不可とみなされます。</p> <p>HTML 内の一部の文字は、コード化されています。たとえば、「&」は「&amp;」、「"」は「&quot;」でコード化されます。パターンを設定する場合には、HTML ソースを見て、ブラウザに表示されている文字列にこのようなコード化された文字がないかどうかを調べます。コード化された文字が文字列に含まれている場合には、それに従ってパターンを調整します。</p> <p>例</p> <pre>pattern="+HP OpenView"</pre>
patternconfig	パターンオプション	<p>単語間のデフォルトの演算子は、「AND」です。この文を使って、「OR」に変更することができます。単語の比較は、デフォルトでは、大文字と小文字を区別します。「ICASE」と入力することによって、大文字と小文字を区別しないようにすることができます。</p>
username	ユーザー名	<p>HTTP 認証で使用するユーザー名を指定します。</p> <p>例:</p> <pre>username=ovros</pre>
password	パスワード	<p>HTTP 認証で使用するパスワードを指定します。</p> <p>例:</p> <pre>password=artros</pre>
proxyUsername	ユーザー名	<p>HTTP プロキシ認証で使用するユーザー名を指定します。</p> <p>例:</p> <pre>proxyUsername=secureros</pre>

表 4 URL/ナビゲーションポイント モードのスクリプト文 (続き)

文	引き数	説明
proxyPassword	パスワード	HTTP プロキシ認証で使用するパスワードを指定します。 例： proxyPassword=artros
label	ラベル	トランザクションステップのシンボリック名です。 例： label=Check-out

表4 URL/ナビゲーションポイントモードのスキプト文(続き)

文	引き数	説明
dynCmdPattern<idx> dynCmdDelim<idx> dynCmdSubst<idx>	代替規則(パターン、デリミタ、代替処理)	<p>動的 URL に対する代替規則を指定します。文は <idx> によってグループ化されます。たとえば、dynCmdPattern3、dynCmdDelim3、dynCmdSubst3 文は同一の規則に属します。パターンは正規表現を使って指定することができます。デリミタは、パターンの直後からデリミタまでのテキストを区切る文字列です。このテキストは、代替処理の指定に応じて、変数またはパス要素を置き換えるために使われます。代替処理の引き数は、以下のとおりです。</p> <p>QUERY <クエリー文字列内の変数></p> <p>PATH <位置インデックス></p> <p>POST <POST データ内の変数></p> <p>HOST <URL のホスト部分></p> <p>位置インデックスでは負の数を使って、要素を後から数えて指定することができます。たとえば、-1 は最後のパス要素を指します。</p> <p>例:</p> <pre> dynCmdPattern0=pageGenTime' value=' dynCmdDelim0='> dynCmdSubst0=QUERY pageGenTime dynCmdPattern1=AUTHMETHOD' value=' dynCmdDelim1='>dynCmdSubst1=POST AUTHMETHOD dynCmdPattern2=pageGenTime' value=' dynCmdDelim2='>dynCmdSubst2=PATH 1 </pre> <p>標準の動的な代替規則では置き換えができない場合には、別の形式オプションを使うことができます。</p>

表 4 URL/ナビゲーションポイント モードのスク립ト文 (続き)

文	引き数	説明
		<p>例:</p> <pre>DynCmdPattern0=test&arg=login DynCmdDelim0=" DynCmdSubst0=PATH -1=test&arg=login%s</pre> <p>これは、最後の位置のパターン “test&arg=login” を検索し、見つかった場合には、その後続く文字列を置き換えることを意味します。DynCmdSubst0 パラメータは、IEStep で指定されるパスの最後のパス要素を、等号の後に続く文字列 (test&arg=login%s) で置き換えます。文字列を実際に置き換える前に、%s をパターンとデリミタによって指定される文字列で置き換えます。</p> <p>同様に、POST でも検索文字列を使うことができます。</p> <p>例:</p> <pre>DynCmdPattern0>Password=test DynCmdDelim0=; DynCmdSubst0=POST Password=test%s</pre>

表4 URL/ナビゲーションポイントモードのスク립ト文(続き)

文	引き数	説明
dynCmdScript<idx> dynCmdSubst<idx>	代替規則(スク リプト、代替処 理)	<p>動的 URL に対する代替規則を指定します。文は <idx> によってグループ化されます。たとえば、dynCmdScript1 文、dynCmdSubst1 文は同一の規則に属します。</p> <p>スク립トには、代替部分の値を返すスク립トを指定します。代替処理の引き数は、以下のとおりです。</p> <p>QUERY <クエリー文字列内の変数></p> <p>PATH <位置インデックス></p> <p>POST <POST データ内の変数></p> <p>位置インデックスでは負の数を使って、要素を後から数えて指定することができます。たとえば、-1 は最後のパス要素を指します。</p> <p>例:</p> <pre>DynCmdScript1=getid.shDynCmdSubst1=POST sessionId</pre>

Web Transaction Recorder についてのヒント

この項では、トランザクションの記録で問題が生じた場合に、役に立つヒントを紹介します。

パターン文字列の選択

Web Transaction Recorder でトレースを 1 に設定して、パターンマッチさせるステップに移ります (このトレースレベルは、Web Transaction Recorder が動作している場合のみ有効となります)。ウィンドウの最下部に、取得したタイトルの文字列を表示します。[**ステップのプロパティ**] ダイアログボックスの [**パターン**] フィールドで、この文字列を二重引用符 (") で囲み、先頭にプラス記号を付けます。

パターンマッチでは英数字だけを使う

HTML 内の一部の文字は、コード化されています。たとえば、「&」は「&」、"」は「"」でコード化されます。パターンを設定する場合には、HTML ソースを見て、ブラウザに表示されている文字列にこのようなコード化された文字がないかどうかを調べます。コード化された文字が文字列に含まれている場合には、それに従ってパターンを調整します。

Cookie を無効にする

システム内に存在する Cookie は、記録、再生、プローブの実行に影響を与えません。

HTTP_TRANS プローブを実行する前に Cookie を削除するには、Web Transaction Recorder のメインウィンドウで、[**ファイル**] メニューの [**設定**] をクリックします。[**Transaction Properties**] ダイアログボックスを使って、グローバルのトランザクションレベルで [**Cookie を無効にする (IE)**] を設定します。このフラグは、プローブを実行する OVIS スケジューラのみ有効です。Web Transaction Recorder で記録を再生するときには効果がありません。

トランザクションの記録を開始する前に Cookie を削除するには、Web Transaction Recorder のメインウィンドウで、[**ツール**] メニューの [**キャッシュビューアー**] をクリックします。[**すべて削除**] ボタンをクリックして、[**OK**] をクリックします。これはオプションの設定であり、Cookie を削除せず記録することも可能です。

Web Transaction Recorder で再生する前に Cookie を削除するには、記録を再生する前に、[ツール] メニューの [キャッシュビューアー] をクリックして、Cookie を削除します。

IE モードでボタンやドロップダウンが機能しない

再生時にボタンやドロップダウンが機能していない場合は、そのステップの [ステップのプロパティ] ダイアログボックスを使って [タイプ] を "IE Url" に設定します。次に、ページで [表示] メニューの [ソースの表示 (HTML)] をクリックして、該当するボタンまたはドロップダウンに対する URL を探し、その URL を [ステップのプロパティ] ダイアログボックスの [ターゲット] フィールドに設定します。URL に POST データがある場合には、[ターゲット] フィールドからその部分を切り取って、それを [ポストデータ] フィールドに移します。パターンマッチも指定します。

待機時間

トランザクションの個々のステップに、waitTime の値を定義することができます。多くのリダイレクションを行うステップがある場合、waitTime スクリプト文を指定することができます。これは、次のステップを確認するパターンマッチが行われる前に、ページが表示されるよう、十分な時間を確保するために設定します。[ステップのプロパティ] ダイアログボックスの [詳細] タブで waitTime 文を追加します (87 ページの表の waitTime を参照)。

ForceWindowClose

サイトで Web ページの前面に表示されるポップアップステータスウィンドウが使用されている場合には、ForceWindowClose スクリプト文を使う必要があります。この文がない場合には、再生時に最前面の空のウィンドウが削除されず、失敗します (パターンマッチは最前面のウィンドウを対象にするため)。詳細は、89 ページの表の ForceWindowClose を参照してください。

動的ホスト名

トランザクションが再生されるたびに、負荷分散に起因してホスト名 (ナビゲーションポイントの URL 部分) が変化する場合、「最も近い一致が見つかりません。」または「ステップを解決できません。」のエラーにより再生が失敗します。

```
A "_top" "35" "" "https://server1/servlet/a-b"
```

この状況を処理するには、ステップを編集して、URL のホスト名の部分に * を指定します。これによりプローブはチェックの際にホスト名を無視します。

```
A "_top" "35" "" https://*/servlet/a-b
```

トラブルシューティング

HTTP_TRANS プローブのトラブルシューティングでは、記録中の問題点や、記録の再生中の問題点を対象にします。

この項では、記録と再生についての問題点について説明します。また、トレース出力の説明を注釈として付けた HTTP_TRANS のトレースファイルを例として示してあります。

記録と再生に関する問題点

記録に関する問題点

記録に関する問題点では、以下の点について説明します。

- IE モードでトランザクションステップが記録されない (メニュー、プラグイン)
- IE モードでのポップアップウィンドウ
- フォームの値が記録されない



Microsoft スクリプトデバッガがインストールされているプラットフォームで、Web Transaction Recorder を使用する場合には、Internet Explorer でのスクリプトのデバッグを無効にすることをお勧めします。IE では、[ツール] > [インターネットオプション] > [詳細設定] を選択して、[スクリプトのデバッグを使用しない] にチェックを付けます。スクリプトのデバッグを有効にしていると、スクリプトエラーが発生するページでは、Web Transaction Recorder の再生と記録に支障をきたします。

IE モードでトランザクションステップが記録されない (メニュー、プラグイン)

複雑な JavaScript やプラグイン (Macromedia FLASH など) を基にしたメニューなどの Web アプリケーションの要素は、マウスイベントを内部的に処理して、Web Transaction Recorder には転送しません。そのため、新らたなページがブラウザウィンドウに読み込まれても、トランザクションステップが左ペインに記録されません。

この問題を解決するには、[設定] ダイアログボックスで [URL の記録を許可 (IE)] を選択します (46 ページの「基本的なトランザクションステップのプロパティ」を参照)。この指定により、Web Transaction Recorder は、URL をキャプチャし、トランザクションステップのリストに追加します。

ブラウザに読み込まれる URL ごとに、[確認] ダイアログボックスが表示されます。通常は、最初の URL のみを記録することが必要で、残りの URL は、ページが完全に読み込まれるまで無視してかまいません。

URL(または POST データ)に動的要素が含まれている場合には、代替規則を適用します(53 ページの「動的な URL の代替処理」を参照)。

上記の対応方法によってもトランザクションステップが記録されない場合には、手作業で URL を追加します。そのためには、トレースを有効にして、"Loading:" で始まるメッセージを探します。トレースリストで見つかった URL と POST 情報を、Web Transaction Recorder にステップとして貼り付けます。

URL を読み込むフレームを指定する必要がある場合があります。iestep コマンドで frame 文を使用します(詳細は、78 ページの表の iestep と frame 文を参照してください)。フレーム名は、BeforeNavigate2 メッセージの末尾に表示されます。

例:

```
BeforeNavigate2: 'http://www.hp.com/' ('mainFrame')
```

IE モードでのポップアップウィンドウ

Web アプリケーションが複数のウィンドウを開いていると、メインウィンドウがポップアップウィンドウの陰に隠れて見えなくなることがあります。メインウィンドウでの記録を継続するには、ブラウザビューウィンドウの[重ねて表示]ボタンを選択して、メインウィンドウを選択します。

または、グローバルの[プロパティ]ダイアログボックスの[ポップアップウィンドウを無視(IE)]または[エラーダイアログを無視(IE)]のダイアログチェックボックスにチェックを付けて、広告などのポップアップウィンドウを無視するように Web Transaction Recorder を設定することができます(39 ページの「グローバルなトランザクションプロパティの設定」を参照)。

また、トランザクションステップで自動的に生成されたパターンも必ず確認してください。これが、メイン Web ページのパターンではなく、ポップアップウィンドウのパターンの場合があります。

フォームの値が記録されない

状況によっては、記録時に入力したデータが記録されないことがあります。これは、フォーム要素を空にする OnClick ハンドラーをフォームに設定すると発生する場合があります。この指定により、Web Transaction Recorder が入力された情報を取得することができなくなります。唯一の回避策は、手作業で、次のような formdata 文を指定します。

トラブルシューティング

構文: `formdata=<フレーム名>.<フォーム>.<変数>=値`

例: `formdata=_top.form1.index=blended`

再生に関する問題点

- トランザクションがタイムアウトする
- HTTP_TRANS プローブがリソース使用率に与える影響
- 動的なフレーム名 (エラー: フレームが見つかりません。)
- 初期化エラーダイアログボックス
- 応答時間とシステム負荷の増加
- Java アプレットが異常終了または不正な動作を引き起こす
- 再生時にトランザクションステップの終了が検出できない
- 再生 / ステップ時の空のメッセージボックス
- 新しいウィンドウが正しく読み込まれない
- 選択ボックスの内容が再生のたびに変化する
- IE モードのレジストリ設定
- 動的な ID フィールド
- Active-X コントロール / Java アプレット対応の Web サイトにおけるスクリプト
- Windows 2000 での再生時の制約

トランザクションがタイムアウトする

プローブのタイムアウトは、以下の原因が考えられます (IE モード)。

- トランザクションが実行されるプローブロケーションにプロキシが設定されていない。
設定マネージャの [**プローブロケーションの情報**] ダイアログボックスに、トランザクション用のプロキシ設定を追加し、[**プローブ設定の保存**] をクリックします。
- プローブの実行に使用されるデフォルトのシステムアカウントに、トランザクションを記録したユーザーアカウントと同じ IE 設定が設定されていない。

コントロールパネルを使って、[HP Internet Services] サービスをトランザクションを記録したアカウント下で実行するように変更します (62 ページの「[プローブのログオンアカウントの変更](#)」を参照)。または、レジストリのシステムユーザーの IE 設定を変更します (66 ページの「[プローブの結果に影響を与えるレジストリの設定 \(IE モード\)](#)」を参照)。また、Web Transaction Recorder を起動したときに、最初のダイアログの [**実行ユーザー情報**] フィールドでユーザーを指定することもできます。

- 一部の IE ダイアログボックスでユーザー対話処理を待ち、すべての処理が終了していない。

IE ダイアログボックスを処理するようにスクリプトを変更します (50 ページの「[Internet Explorer のセキュリティダイアログボックス](#)」を参照)。

- IE モードのトランザクションステップが、ページを更新するときに、新しい URL を読み込まない。

タイムアウトの原因になっているトランザクションステップに、continueClick=1 コマンドを追加します (88 ページの表の continueClick を参照)。

- プローブのタイムアウト状況は、[**プローブの再試行回数**] の設定と [**プローブロケーションの情報**] ダイアログボックスの [**リクエストタイムアウト値**] の設定値によって変動します。

HTTP_TRANS プローブがリソース使用率に与える影響

IE モードの HTTP_TRANS プローブ (IE モードで記録、httptrans2.exe を実行) は IE エンジンを起動するので、複数の HTTP_TRANS プローブを実行するとリソース使用率が高くなります。たとえば、5 つの IE モードの HTTP_TRANS プローブを同時に実行すると、5 つの IE エンジンを動作させることとなります。これは同時に 5 つの IE ブラウザを開いて、すべてのブラウザで同時に Web ページを開いてナビゲートすることに相当します。このようなオーバーヘッドが発生するため、HTTP_TRANS プローブでは同時実行数にできるだけ小さな値 (たとえば、1 ~ 10) を設定することをお勧めします。また、HTTP_TRANS プローブは、専用のリモートプローブサーバーで実行することも検討してください。

実際のユーザーの応答時間を反映できる最適な同時実行数を調べるために、一連のテストを行うことができます。HTTP_TRANS プローブを設定して、それを concurrency=1 が設定されたネットワーク接続に割り当てます (設定マネージャの [**プローブロケーションの情報**] ダイアログを使用します)。プローブをプ

ローブシステムで実行して、応答時間を測定します。同時実行数を大きくしてプローブを実行しても、応答時間が変わらないかどうかを調べます。応答時間が近い値であれば、プローブはその同時実行数で、実際のユーザー操作を反映する応答時間で実行できることとなります。応答時間の値が長くなれば、複数のプローブを実行する際のオーバーヘッドが大きくなります。これは応答時間に影響を与えるため、ユーザー操作が正しく表現できないこととなります。この場合には、同時実行数により小さな値を割り当てる必要があります。ただし、可用性だけを調べる場合には、応答時間の遅延は無視できます。

リソースの使用率を下げるその他のオプションは、次のようなものがあります。

- プローブの同時実行数を1にして実行します。このようなプローブが数多くある場合には、プローブを複数のリモートプローブシステムに分散します。
- 異なるプローブタイプ (URL モードの HTTP_TRANS) を使ってトランザクションを監視し、オーバーヘッドを下げるができます。測定しているトランザクションが1つのステップしか持たない単純なトランザクションであれば、HTTP プローブを使うことができます。トランザクションが Java やプラグインを実行していなければ、URL モードのプローブを使うことができます。このプローブは、IE ブラウザを起動するオーバーヘッドなしに実行することができます。
- 可用性だけを測定するのであれば、同時実行数を大きくすることができます。ユーザー操作をリアルにシミュレートするのが目的ではなく、可用性だけを測定するのであれば、URL モードの HTTP_TRANS プローブを使うことができます。応答時間を測定する必要がある場合には、ネットワーク接続の同時実行数を小さな値に設定します。さらに、同時実行数を1に設定する必要がある場合があります。通常はIE ブラウザのウィンドウは1つしか開かないので、1以外の値を設定すると、典型的なユーザー操作の応答時間を反映できなくなります。

動的なフレーム名 (エラー: フレームが見つかりません。)

一部のフレームウィンドウは JavaScript によって作成されるため、フレームが無作為に生成される可能性があります。フォームデータがこのフレームに記録されている場合には、再生時に問題が発生することがあります。次のようなエラーメッセージが表示されます。

エラー : フレーム "1234567890" が見つかりません。

動的なフレーム名を持つトランザクションステップの例は、以下のとおりです。

```
IENAVPNT=A "1234567890" "252" "button"  
"javascript:addToCartUsingToggle();"  
  
FORMDATA=1234567890.items.item=Submit=True
```

上記の IENAVPNT と FORMDATA 文では、items フォームは 1234567890 フレーム上にあります。トランザクション再生時にフォーム名は変化するため、プローブはフォームデータの位置を見つけることができなくなります。

この問題を回避するには、フレーム名を削除 (未指定のままにしておく) して、最上位のフレームを使用するようにするか、フレームのインデックスを指定するようにします。

例 A:

```
IENAVPNT=A "" "252" "button"  
"javascript:addToCartUsingToggle();"  
  
FORMDATA=.items.item=Submit=True
```

例 B:

```
IENAVPNT=A "[2]" "252" "button"  
"javascript:addToCartUsingToggle();"  
  
FORMDATA=[2].items.item=Submit=True
```

例 A では、IENAVPNT と FORMDATA のフレーム名は空です。これはプローブに最上位のフレームを使うことを指示しています。例 B では、フレームコレクションオブジェクトのフレーム 2 を指定しています。フレーム番号を調べるには、Web Transaction Recorder のメインウィンドウの [表示] メニューの [ソースの表示 (DOM)] を選択して、すべての HTML 要素のインデックスとフレーム名を表示します。

初期化エラーダイアログボックス

初期化エラーによってプロセス (特に、プローブプロセスやスケジューラ) が異常終了したことを示すポップアップダイアログボックスが表示される場合には、[アカウント] を選択してトランザクションを記録したアカウント (ユーザー名) とパスワードを入力して、HP Internet Services のシステムアカウントの設定を変更します。これは、管理サーバーとリモートプローブシステム (トランザクションを実行するすべてのプローブセッション) の両方で変更を行います。この設

定情報には、Windows 2000 の場合、[**コンピュータの管理**]>[**サービスとアプリケーション**]>[**サービス**]>[**プロパティ**]>[**ログオン**]の順に選択して、アクセスできます。

Web Transaction Recorder を起動したときに、最初のダイアログの [**実行ユーザー情報**] フィールドに、ユーザーを指定しておくこともできます。

応答時間とシステム負荷の増加

Web Transaction Recorder は、IE モードでは、プローブを実行するために IE エンジンを開始します。

プローブは、デフォルトでは、同時実行数が 32 のデフォルトネットワーク接続を使います。IE モードのプローブを使う場合は、同時実行数が 1 ~ 10 のネットワーク接続にする必要があります。同時実行数はネットワーク接続のグローバル設定のため、システムに負荷を掛け過ぎずに、しかもプローブを指定したプローブ間隔で完了させることができる最大の同時実行数を選択する必要があります。また、プローブが通常より長く実行され、割り当てられているタイムアウト時間を超えるような例外状況に備えて、多少のゆとりも考慮しておく必要があります。同時実行数は設定マネージャの [**プローブケーションの編集**]>[**接続を編集**] ダイアログボックスの [**同時リクエスト数**] フィールドで設定できます。

Java アプレットが異常終了または不正な動作を引き起こす

Java アプレットを含むページの再生で問題が発生する場合には、Java アプレットをダウンロードしないようにします。Web Transaction Recorder で、[**ファイル**]>[**設定**]>[**プロパティ**] ダイアログボックスの [**Java を有効にする (IE)**] チェックボックスのチェックを外します。

再生時にトランザクションステップの終了が検出できない

非常に稀なケースとして、再生時にトランザクションステップの終了を検出することができずに、トランザクションがタイムアウトになることがあります。このような場合には、問題を発生させたトランザクションステップに、continueClick=1 フラグを追加します。

このフラグは、ブラウザがサーバーからドキュメントを取り込まない場合に、必要になります。たとえば、Web アプリケーションがメニューを開いたり、JavaScript で同様のアクションを実行する場合です。トレース結果には、DownloadBegin/DownloadComplete シーケンスしか表示されません。そのため、トランザクションステップの終了を検出することができません。

最後の3つのトレースメッセージが BeforeNavigate2、DownloadBegin、DownloadComplete シーケンスで、その後に DocumentComplete メッセージが続いていなかった場合には、completeCheck フラグには "2"

(completeCheck=2) を設定します。この設定によって、Web Transaction Recorder はトランザクションステップ終了の検出方法を変更します。フラグに "2" が設定されると、Web Transaction Recorder は上記のイベントシーケンスを検出すると、トランザクションステップは終了したと判断します。

Web アプリケーションが多くのリダイレクションを行って、バックグラウンドで (たとえば、JavaScript を使って) 追加ページを読み込んでいる場合には、そのステップでサーバーからダウンロードするドキュメントの正確な数を指定して、トランザクションステップの終了を判断するようにプロンプトに指示することもできます。トレース結果には、ダウンロードする各ドキュメントの docCompletes カウンター、たとえば、docCompletes=31 が含まれます。これらの docCompletes の正確な数を使って、プロンプトにトランザクションステップの完了を判断させるように指示することができます。[**ステップのプロパティ**] ダイアログボックスの [**詳細**] タブで docCompletes=< トレース結果からの完了とするドキュメント数 > を入力します。90 ページの表の docCompletes を参照してください。

再生 / ステップ時の空のメッセージボックス

再生 / ステップ時に空のメッセージボックスが表示されたり、トランザクションで複数のウィンドウが開かれる場合には、ステップに forceWindowClose=1 フラグを追加して (89 ページの表の forceWindowClose を参照)、これらのウィンドウを閉じます。

新しいウィンドウが正しく読み込まれない

トランザクションステップで新しいウィンドウが読み込まれても、内容が正しく表示されない場合 (たとえば、空白ページ) には、このトランザクションステップに noAbout=1 フラグを設定します (89 ページの表の noAbout を参照)。

このフラグは **Web Transaction Recorder** に対して新しいウィンドウを空白ページ (`about:blank`) で初期化しないように指示します。トランザクションで認証が使用されるときには初期化が必要です。

また、[**プロパティ**] ダイアログボックスで [**ブラウザを初期化する (IE)**] フラグのチェックを外して、トランザクション全体で初期化が行われないようにすることもできます (39 ページの「[グローバルなトランザクションプロパティの設定](#)」を参照)。

選択ボックスの内容が再生のたびに変わる

トランザクションステップで、トランザクションを再生するたびに変わる選択ボックス内の要素を参照している場合には、**Web Transaction Recorder** は元の要素が検出できずに異常終了します。たとえば、銀行トランザクションで、口座のプルダウンメニューに、口座と現在の残高が表示されることがあります。残高は絶えず変化しているので、**Web Transaction Recorder** は正しいエントリーを検出することができません。この問題を回避するには、対応する `formData` 文を指定して、選択ボックス内の要素の絶対位置を示します。たとえば、`formData=form.sel=@1` は、1 番目の要素を指定します (位置は通常 0 から始まります)。

IE モードのレジストリ設定

レジストリキーを設定して、IE モードのプローブのトラブルシューティングを容易にすることができます。次のように **DWORD** キーを以下のレジストリキーに追加します。

```
HKEY_LOCAL_MACHINE¥SOFTWARE¥Hewlett-Packard¥Internet  
Services¥CurrentVersion
```



リモートプローブシステムには、このキーでなく、次のキーが存在します。
HKEY_LOCAL_MACHINE¥SOFTWARE¥Hewlett-Packard¥Internet Services
Remote Probes¥CurrentVersion

このキーに、**DWORD** キーを追加しないでください。

リモートプローブシステムでは、
HKEY_LOCAL_MACHINE¥SOFTWARE¥Hewlett-Packard¥の下に、Internet
Services¥CurrentVersion キーを作成します。

(最初に Internet Services キーを作成し、Internet Services の下に CurrentVersion
キーを作成します。)

以下の **DWORD** キーを追加し、値を "1" に設定して、有効にします。これらの
キーを追加しても、再起動は不要です。

- ContinuousCapture

[**エラー時画面をキャプチャ**] が有効にされていると、プローブは連続的に
キャプチャを実行します。各ファイル名には、イメージまたはトレースが作
成された日時と時刻が示されます。このオプションは、問題の動作を長期間
にわたって観察する場合に、役立ちます。

 十分なディスクスペースを用意してください。

- CaptureTraceOnAvail

可用性が "1" の場合でも、trace.log の追加情報 (現在のトレース情報)
が、別のファイルに書き込まれます
(<data dir>¥data¥tmp¥probe¥capture¥に保存)。
これは応答時間に関連する問題のトラブルシューティングに役立ちます。

- ShowProgress

プローブと Web Transaction Recorder に対して、通常は IE の左下のステータスバーに表示される情報を追加するように指示します。この情報にはどのファイルが現在ダウンロードされているかが表示されます。

動的な ID フィールド

表データ (TD) や分割機能 (DIV) のような HTML 要素の ID フィールドが、Web サーバーによって動的に生成された場合には、Web Transaction Recorder は、再生時にステップを解釈することができなくなります。この回避策として、要素に関連付けられている内部の文字列を指定して、ステップのターゲットに追加します。

この問題は、Ariba サーバーで発生します。Ariba サーバーでは再起動するたびにフィールド ID が変化するので、Web Transaction Recorder は記録済みのトランザクションを再生することができません。1つの回避策としては、Ariba のパラメータを NamePrefix=True と設定することです。Ariba のバージョンによっては、これによって、再起動のたびに ID を動的に生成することを停止させることができます。これが有効ではなかった場合には、以下の例に示したステップを使用して、問題点を解決します。

例

元の HTML に次のように書かれているとします。

```
<td id="47A">example</td>
```

生成されるナビゲーションステップは、次のようになります。

```
TD "_top" "25" "47A"
```

“47A” の ID が Web サーバーによって変更されると、このステップの再生に失敗します。ステップに手作業で内部のテキスト “example” を ID として追加して、問題点を解決します。ステップは、次のようになります。

```
TD "_top" "25" "example"
```

Active-X コントロール /Java アプレット対応の Web サイトにおけるスクリプト

Active-X コントロールにおけるスクリプトを記録する場合には、スケジューラをローカルシステムアカウントで実行すると、正しく再生できないことに注意してください。ローカルシステムアカウントは、Active-X コントロールを実行するための権限を持っていません。スケジューラのログオンアカウントを、適切な権限

を持ったアカウントに変更する必要があります。また、IE には、サイトを含むゾーンに基づいて種々のセキュリティ設定が行われています。そのため、サイトを含むゾーンとセキュリティ設定が正しいことを確認する必要があります。または、Web Transaction Recorder を起動したときに、最初のダイアログの [**実行ユーザー情報**] フィールドでユーザーを指定します。

Windows 2000 での再生時の制約

IE モードの HTTP_TRANS プローブ (probehttptrans2) を、-print オプションを指定して実行すると、Windows 2000 では結果を生成することができません。このプラットフォームでは、Windows GUI アプリケーションからコンソールに書き込む機能がサポートされていません。設定マネージャの [**管理サーバーでテスト**] 機能は、Windows 2000 上の IE モードの HTTP_TRANS プローブでは、結果を返しません (XP 以降では正しく動作します)。また、Windows 2000 上のこのプローブに対する Probe Re-Execution TIP も、"No results returned for command" というメッセージを返します。

HTTP_TRANS プローブの TIP の設定

HTTP_TRANS の TIP を実行したときにダッシュボードの TIP ビューアーに "Directory/file cannot be found" というエラーが表示される場合には、Web Transaction Recorder の [**ファイル**] > [**設定**] > [**プロパティ**] ダイアログボックスで [**エラー時画面をキャプチャ**] オプションにチェックを付ける必要があります。このオプションにチェックを付けないと、プローブは、TIP での検索対象のエラーログやエラー画面イメージを生成しません。

HTTP_TRANS の注釈付きトレースファイル

HTTP_TRANS プローブの注釈付きのトレースファイルを以下に示します。

```
Setting proxies: 'http=web-proxy.corp.hp.com:8088
https=web-proxy.corp.hp.com:8088'
Proxies Configured: 'https=web-proxy.corp.hp.com:8088
http=web-proxy.corp.hp.com:8088'
Using proxies: 'web-proxy.corp.hp.com:8088', 'web-proxy.corp.hp.com:8088'
^^^ コマンド行から渡されたプロキシでIE と実際のプロキシを構成

Load 'about:blank'. This is a workaround for an IE bug where NTLM
authentication will fail if this page is not loaded. This can be turned off
(Initialize Browser (IE) or noAbout=1).

BeforeNavigate2: 'about:blank' (')
^^^ BeforeNavigate2 イベント。最初のパラメータはURL、2番目のパラメータはフレーム名
(')
BeforeNavigate2: KillTimer
^^^ URL の読み込みのため、ウォッチドッグタイマをリセット
DownloadBegin
DownloadComplete
Clearing select cache
DownloadBegin
NavigateComplete2
DownloadComplete
IE Version: '4.0 (compatible; MSIE 6.0; Windows NT 5.0; H010818;
Hewlett-Packard IE5.5-SP2; .NET CLR 1.0.3705; .NET CLR 1.1.4322) '
^^^ IE バージョンの表示。Win2K のセキュリティパッチをインストールした IE 6.0。
DocumentComplete: b=0:rs=4 about:blank
^^^ ドキュメント "about:blank" の読み込み (DocumentComplete イベント)
DocumentComplete: Document is done downloading
^^^ このメッセージはすべてのフレームをダウンロードしたことを示します (about:blank は1フ
レームしかありません)
DocCompletes: 1
^^^ 現在のステップの終了を確定するドキュメント数
Redirects: 1
^^^ 現在のステップのリダイレクト数
ElapsedTimer Proc 180
```

最初のステップ iestep=http://www.hp.com/ の開始

```
BeforeNavigate2: 'http://www.hp.com/' (')
BeforeNavigate2: KillTimer
DownloadBegin
```

トラブルシューティング

```
DownloadComplete
DownloadBegin
NavigateComplete2
DownloadComplete
DocumentComplete: b=0:rs=4 http://www.hp.com/
DocumentComplete: Document is done downloading
DocCompletes: 2
Redirects: 2
SelectFrame: '_top'
^^^ _top フレームを選択し、ドキュメントのMDI ウィンドウの前面に移動
FrameFinder: '_top'
^^^ 指定したフレーム名にドキュメントを配置する基本レベルの関数からのトレース
SelectFrame: found
Size:
_top=36843,s.gif=43,hpweb_1-2_arrw_sbmt.gif=79,b1_smb_text_oct_09.gif=2582,b1
_smb_photo_oct_09.jpg=16772,b2_b3_smb_valuepromo_oct_16.gif=4371,&s=1280x960&
c=32&j=1.3&v=Y&k=Y&bw=752&bh=362&ct=lan&hp=N&[AQE]=43,hpweb_utilities.js=7478
,hpweb_country_pulldown.js=10992,setcookie.js=1082,metrics.js=16044
^^^ ドキュメントサイズINFO Size=96329 を判断するために使用されたオブジェクトの
サイズ情報を表示 (http://www.hp.com/)
Set timer: 1500
^^^ ウォッチドッグタイマーに "1500 ms" を設定
OnTimer
^^^ ウォッチドッグタイマーのタイムアウト検出 (1500ms 後)、可用性を判定して、測定値を取得
Kill timer
^^^ ウォッチドッグを無効にする関数から低レベルトレース
```

次のセクション

```
Title: Welcome to HP
^^^ 現在のドキュメントのTITLE タグ
Matched: +"Welcome to HP"
^^^ "pattern" パラメータの値とパターンマッチの結果表示
IPAUtil:: Cookies retrieved='cc=us; lang=en-us; s_sq; s_cc=true'
^^^ OVTA 統合固有のメッセージ。現在のCookie を表示
10/21 09:52:33 (1) host='www.hp.com', target='Step 00: http://www.hp.com',
avail=1, setup=0, sresp=10, trans=1414, totRespTime=1424, bytes=96329,
tput=66.06, reqs=1 , stepUrl=http://www.hp.com/
^^^ キューファイルに書き込まれるステップ0 のすべての測定値
IPAUtil:: corr=, nodeId=97513558-e711-473e-841d-9fe4c9326903
^^^ OVTA 統合固有のメッセージ。相関関係とnodeId Cookie を表示
```

2 番目のステップ 'INPUT:image "_top" "475" "submit" "" "http://
welcome.hp-ww.com/img/hpweb_1-2_arrw_sbmt.gif" の開始

```
SelectFrame: '_top'
FrameFinder: '_top'
SelectFrame: found
```

```

^^^ HTML 要素の指示に指定されているフレームを検出
Clearing select cache
^^^ 内部メッセージのトレースメッセージ
Find: element='_top.searchForm.qt', value='Internet Services'
^^^ ステップのフォームデータ。"_top" フレームのフォーム "searchForm" で INPUT 要素 "qt"
を検出
Set form data: form='searchForm', element='qt', value='Internet Services'
^^^ 検出後データ値 ("Internet Services") を設定
Playing back: 'INPUT:image "_top" "475" "submit" "" "http://
welcome.hp-ww.com/img/hpweb_1-2_arrw_sbmt.gif"'
^^^ [送信] ボタンのクリックを準備
10/21 09:52:33 (9) FindFrame: find: '_top' current: '_top'
^^^ フレームファインダーからのトレースメッセージ
^^^ エラーメッセージがないことは、HTML 要素が検出されて、クリックされたことを意味する
10/21 09:52:33 (8) BeforeNavigate2: 'http://www.hp.com/search/
?cc=us&lang=en&qt=Internet+Services&submit.x=0&submit.y=0' (')
^^^ [送信] ボタンによって実行された URL を通知する BeforeNavigate2 イベント。クリックは
正常に動作

```

次のセクション

```

BeforeNavigate2: KillTimer
DownloadBegin
Security dialog (problem=12168)
^^^ セキュリティ問題を示す (12168 == ERROR_HTTP_REDIRECT_NEEDS_CONFIRMATION)
NavigateComplete2
DownloadComplete
DocumentComplete: b=0:rs=4 http://search.hp.com/gwuseng/
query.html?col=hpcom+ccen+ccenfor&la=en&qt=Internet+Services&.
DocumentComplete: Document is done downloading
DocCompletes: 1
Redirects: 1
Size:
_top=37569,s.gif=43,hpweb_1-2_topnav_home.gif=274,hpweb_1-2_topnav_prdsrv.gif
=483,.
INFO Size=75653 (http://search.hp.com/gwuseng/
query.html?col=hpcom+ccen+ccenfor&la=en&qt=Internet+Services.
Set timer: 1500
OnTimer
Kill timer
Title: Search HP US - Search results for 'Internet Services'
Matched: +"Search HP US - Search results for 'Internet Services'"
host='www.hp.com', target='Step 01: ', avail=1, setup=30, sresp=0,
trans=2256, totRespTime=2286, bytes=75653, tput=32.32, reqs=1 ,
stepUrl=http://www.hp.com/search/
?cc=us&lang=en&qt=Internet+Services&submit.x=0&submit.y=0
^^^ キューファイルに書き込まれるステップ1のすべての測定値
Kill timer
INFO: Stepping/Playback for '/' done

```

トラブルシューティング

^^^ すべてのステップが終了

```
host='Trans: www.hp.com', target='Trans: :', avail=1, setup=30, sresp=10,  
trans=3670, respTime=3710, bytes=171982, tput=45.27, reqs=2
```

^^^ キューファイルに書き込まれるトランザクションサマリー

```
ElapsedTimer stopped
```

索引

C

checkAvailability **89**

clickMode **94**

codebase パラメータ **14**

completeCheck **89**

continueClick **88**

Cookie

固定 **18**

セッション **18**

セッションの保存 **95**

設定 **63**

定義 **17**

D

delay **93**

dlgCmd **83**

DlgCmd で実行されるユーザーアクション **51**

docCompletes **90**

DOM ドキュメントオブジェクトモデル **26**

dstep **96**

dynCmdPattern **85, 99**

dynCmdPattern、dynCmdDelim、dynCmdSubst
例 **54**

dynCmdScript **87, 101**

E

enableIEDialog **91**

F

forceWindowClose **89**

ヒント **103**

formdata **80**

formInputMode **93**

G

GUI

例 **36**

H

HTML

定義 **11**

HTMLDIALOG の例 **72**

HTML ダイアログ **68**

HTML 要素の動的 ID **117**

HTTP **15**

HTTP_TRANS プロープ **34**

HTTP_TRANS プローブでの TIP エラー **118**
HTTPS **15**

I

ienavpnt **79**
iestep **78**
IE の設定 **66**
IE モード **20**
 記録 **24**
 再生 **29**
 使用するモードを判断するためのヒント
 22
IE モード用に 1 に設定された同時実行数 **21**

J

JavaScript **12**
 プラグインの例 **13**

L

label **82, 98**
loadsession **95**

M

MCLICK **94**

N

noAbout **89**
noContinue **94**
noWindowClose **88**

P

param パラメータ **14**
password **81, 97**

pattern **81, 97**
patternconfig **81, 97**
pos **96**
POST **80, 96**
proxyPassword **82, 98**
proxyUsername **81, 97**

S

savesession **95**
src 属性 **14**
SSL **15**
step **96**

U

URL
 代替規則 **53**
 定義 **17**
 動的 **53**
 表示する手順 **17**
URL モード **20**
 記録 **24**
 再生 **29**
username **81, 97**

W

waitForDlgCaption **92**
waitTime **87**
 ヒント **103**
waitTimeReset **88**
Web Recorder
 動作 **24**

Web Transaction Recorder

GUI の例 **36**

起動 **34**

使用する手順 **36**

Web Transaction Recorder の起動 **34**

window **90**

か

外部スクリプトを使った可用性チェック **59**

カスケーディング スタイルシート (CSS) **12**

管理サーバーでのテスト **38, 118**

キャッシュ **16**

記録

例 **25**

記録オプション

設定 **41**

クリックのシミュレートで使われるウィンドウイベント **94**

高負荷記録モード **20**

固定 Cookie **18**

さ

サービスレベル目標値 **38**

再生のオプション

設定 **43**

証明書 **15**

ルート **16**

スクリプトオプション **49**

スクリプト文 **78**

ステップアラーム機能 **38**

ステップのプロパティ

基本的な **46**

ステップのプロパティダイアログ
詳細 **49**

セキュリティのダイアログ
処理のヒント **51**

セッション Cookie **18**

設定

同時実行数 **21**

た

タグ **13, 14**

重複する入力変数 **65**

手作業で追加するブラウザプラグインのナビゲーション **63**

同時実行数の設定 **21**

動的 URL **53**

動的 URL の代替規則 **53**

動的に値が取り込まれるメニュー **73**

動的に値を取り込むプルダウンメニュー **73**

動的に値を取り込むメニュー **73**

動的ホスト名

処理のヒント **103**

トランザクション

記録方法 **37**

トランザクションスクリプトのリファレンス
78

トランザクションの記録方法 **37**

トランザクションのプロパティ **39**

トレースファイルの例 **119**

トレースレベル

設定 **45**

な

ナビゲーションポイント モード **20**

名前のないフォーム **65**

入力変数

外部スクリプトを使って設定 **58**

認証 **15**

認証局 **16**

認証ダイアログ **63**

認証用のユーザー名とパスワード **63**

は

パターンマッチのヒント **102**

パラメータ

codebase **14**

param **14**

ハンドラー **13**

フォームへの記入で使われるウィンドウイベント **93**

負荷分散 **103**

ブラウザウィンドウを最大化する **93**

プルダウンメニュー **41**

フレーム **14, 78**

フレーム名

動的 **61**

プローブ

HTTP_TRANS **34**

ロケーション **38**

プローブの結果に影響を与えるレジストリの設定 **66**

プロキシサーバー **16**

プロキシの設定 **45**

プロパティのダイアログ **39**

ページリダイレクション **64**

別のユーザーとして実行 **34**

ボタンやドロップダウンが使えない **103**

ポップアップウィンドウ

処理のヒント **103**

ま

モーダルダイアログ **68**

モード

IE **20**

URL **20**

高負荷記録 **20**

ナビゲーションポイント **20**

や

ユーザーなりすまし **34**

ら

リダイレクション **17**

waitTime の使用例 **64**

ページ **64**

リダイレクションで使われる docCompletes の例 **64**

リダイレクションで使われる waitTime の例 **64**

ルート証明書 **16**

ログオンアカウント

異なるアカウントで実行するように変更

62