



**MERCURY
LOADRUNNER™**

VERSION 8.1

コントローラ・ユーザーズ・ガイド

MERCURY™

Mercury LoadRunner™

コントローラ・ユーザーズ・ガイド
Version 8.1

Mercury LoadRunner コントローラ・ユーザーズ・ガイド, Version 8.1

本マニュアル, 付属するソフトウェアおよびその他の文書の著作権は, 米国および国際著作権法によって保護されており, それらに付随する使用契約書の内容に則する範囲内で使用できます。Mercury Interactive Corporation のソフトウェア, その他の製品およびサービスの機能は次の 1 つまたはそれ以上の特許に記述があります。米国特許番号 5,511,185; 5,657,438; 5,701,139; 5,870,559; 5,958,008; 5,974,572; 6,137,782; 6,138,157; 6,144,962; 6,205,122; 6,237,006; 6,341,310; 6,360,332; 6,449,739; 6,470,383; 6,477,483; 6,549,944; 6,560,564; 6,564,342; 6,587,969; 6,631,408; 6,631,411; 6,633,912; 6,694,288; 6,738,813; 6,738,933; 6,754,701; 6,792,460 および 6,810,494。オーストラリア特許番号 763468 および 762554。その他の特許は米国およびその他の国で申請中です。権利はすべて弊社に帰属します。

Mercury, Mercury Interactive, Mercury のロゴ, Mercury Interactive のロゴ, LoadRunner, WinRunner, SiteScope および TestDirector は, Mercury Interactive Corporation の商標であり, 特定の司法管轄内において登録されている場合があります。上記の一覧に含まれていない商標についても, Mercury が当該商標の知的所有権を放棄するものではありません。

その他の企業名, ブランド名, 製品名の商標および登録商標は, 各所有者に帰属します。Mercury は, どの商標がどの企業または組織の所有に属するかを明記する責任を負いません。

Mercury Interactive Corporation
379 North Whisman Road
Mountain View, CA 94043
Tel: (650) 603-5200
Toll Free: (800) TEST-911
Customer Support: (877) TEST-HLP
Fax: (650) 603-5300

© 1994 - 2005 Mercury Interactive Corporation, All rights reserved

本書に関するご意見, ご要望は documentation@mercury.com まで電子メールにてお送りください。

目次

本書について	xi
本書の構成	xii
LoadRunner 印刷マニュアル	xiii
LoadRunner オンライン・マニュアル	xiv
その他のオンライン・リソース	xv
表記規則	xvi

第 1 部 : LoadRunner について

第 1 章 はじめに	3
アプリケーションの負荷テスト	3
LoadRunner のソリューション	4
LoadRunner の使用	5
LoadRunner を使った作業	6
LoadRunner 仮想ユーザ技術	7
LoadRunner 仮想ユーザの種類	7
LoadRunner ライセンス情報	13
第 2 章 LoadRunner テスト工程	17
ステップ I : テストの計画	18
ステップ II : 仮想ユーザ・スクリプトの作成	18
ステップ III : シナリオの作成	18
ステップ IV : シナリオの実行	19
ステップ V : シナリオの監視	19
ステップ VI : テスト結果の分析	20
第 3 章 負荷テストの計画	21
負荷テストの計画について	21
アプリケーションの分析	22
テスト目標の定義	25
LoadRunner の導入計画	26
負荷テスト目標の調査	32

第4章 LoadRunner コントローラの概要	37
コントローラの起動	37
LoadRunner コントローラの紹介	40
シナリオ・ファイルの管理	43
シナリオの実行	46

第2部：シナリオの設計

第5章 マニュアル・シナリオの作成	51
シナリオの作成について	52
仮想ユーザ・グループの作成	54
仮想ユーザ・グループの仮想ユーザの設定	63
仮想ユーザ実行環境の設定	71
ロード・ジェネレータの設定	74
ロード・ジェネレータの追加設定	79
端末サービスの設定	89
WAN エミュレーションの設定	95
スクリプトの設定	103
スクリプトに対する相対パスの使用	108
第6章 パーセント・モードでのマニュアル・シナリオの作成	111
パーセント・モードでマニュアル・シナリオを作成する方法に ついて	111
仮想ユーザの総数の定義	114
スクリプトへのプロパティの割り当て	115
スクリプトの設定	118
シナリオの仮想ユーザ・グループ・モードへの切り替え	124
第7章 ゴール指向シナリオの作成	127
ゴール指向シナリオの計画について	127
ゴール指向シナリオの [デザイン] タブについて	129
シナリオのゴールの定義	131
スクリプトへのプロパティの割り当て	138
スクリプトの設定	141
第8章 シナリオのスケジュール	147
シナリオのスケジュールについて	147
シナリオの開始の遅延	148
スケジュールの選択	150
シナリオのスケジュール	153
仮想ユーザ・グループのスケジュール	156
スケジュールされたシナリオへの仮想ユーザの追加	159

第 9 章 ランデブー・ポイントの使用	161
ランデブー・ポイントの使用について	161
ランデブー属性の設定	163
ランデブー情報の表示	167
第 10 章 シナリオの設定	171
シナリオの設定について	171
タイムアウト時間の設定	172
シナリオ実行環境の設定	174
実行ファイルの格納場所の設定	176
パス変換の指定	180
第 11 章 シナリオの実行準備	181
シナリオの実行準備について	181
結果の格納場所の指定	182
結果ディレクトリのファイル構造	184
結果の照合	186
シナリオのサマリ情報の設定	189
第 12 章 Quality Center を使ったシナリオ管理	191
Quality Center を使ったシナリオ管理について	191
Quality Center の接続と切断	192
Quality Center プロジェクトからシナリオを開く	195
Quality Center プロジェクトへのシナリオの保存	197
Quality Center プロジェクトへの結果の保存	199
Quality Center プロジェクトからの仮想ユーザ・スクリプトの追加	200
第 13 章 LoadRunner での機能テスト・スクリプトの使用	203
LoadRunner での機能テスト・スクリプトの使用について	203
GUI 仮想ユーザ・スクリプトについて	204
QuickTest を使用した LoadRunner の GUI 仮想ユーザ・スクリプト の使用	206
WinRunner を使用した LoadRunner の GUI 仮想ユーザ・スクリプト の使用	209
LoadRunner シナリオでの仮想ユーザ・スクリプトの実行	214
第 3 部 : シナリオの実行	
第 14 章 シナリオの実行	219
シナリオの実行について	219
シナリオ全体の実行	220
仮想ユーザ・グループの制御	222
個々の仮想ユーザの制御	228
ランデブーからの手作業による仮想ユーザの解放	231
実行中のシナリオへの手作業による仮想ユーザの追加	232

第 15 章 実行中の仮想ユーザの表示	239
実行中の仮想ユーザの表示について	239
仮想ユーザのステータスの監視	240
[出力メッセージ] ウィンドウの表示	242
仮想ユーザ・スクリプト・ログの表示.....	248
実行メモの記録	251
エージェントのサマリ表示.....	251

第 4 部 : ファイアウォールを使った作業

第 16 章 LoadRunner でのファイアウォールの使用	255
LoadRunner でのファイアウォールの使用について	255
システムの設定	258
ファイアウォール越しの仮想ユーザの実行または監視	259
接続の確認.....	260
第 17 章 ファイアウォール越しの実行.....	263
ファイアウォール越しの仮想ユーザの実行について	263
ファイアウォール越しの仮想ユーザ実行の概要	264
ファイアウォールの内側での LoadRunner エージェントの インストール	264
ファイアウォールの内側での LoadRunner エージェントの設定.....	265
エージェントによるアクセスを許可するためのファイアウォール の設定.....	273
ファイアウォールの外側の MI リスナーのインストールと設定	274
ファイアウォール越しの仮想ユーザの実行または監視のための コントローラの設定	276
第 18 章 ファイアウォール越しの監視.....	281
ファイアウォール越しの監視について.....	281
ファイアウォール越しの監視機能のインストール.....	282
データ収集の準備.....	282
サーバ・モニタ・プロパティの設定	283
ファイアウォール越しのネットワーク遅延モニタの設定.....	288

第 5 部 : 診断を使った作業

第 19 章 LoadRunner 診断モジュール	291
LoadRunner 診断モジュールについて	291
LoadRunner ERP/CRM 診断の種類	292
ERP/CRM 診断モジュールのアーキテクチャ	293
LoadRunner ERP/CRM 診断を使った作業	295
リモート・サーバへの接続	297
診断の有効化	300
診断結果の表示	304
第 20 章 Siebel 診断モジュールの設定	305
アプリケーション・サーバおよび Web サーバ上での Siebel 診断 モジュールの設定	305
Web サーバが DMZ 内にある場合の診断モジュールの設定	307
Siebel アプリケーション・サーバからメディアータへのファイルの コピー	308
Siebel 診断モジュールの設定	309
[Siebel 設定] ダイアログ・ボックスについて	312
[Siebel サーバ設定] ダイアログ・ボックスについて	313
第 21 章 Siebel DB 診断モジュールの設定	315
スクリプトの準備	315
クロックの同期	316
Siebel サーバでのサーバ・ログの有効化	316
Siebel DB 診断モジュールの設定	318
第 22 章 Oracle 11i 診断モジュールの設定	323
Oracle サーバでのサーバ・ログの有効化	323
Oracle NCA アプリケーションのバージョンの選択	325
Oracle 11i 診断モジュールの設定	326
[Oracle 11i の設定] ダイアログ・ボックスについて	329
[Oracle 11i サーバの設定] ダイアログ・ボックスについて	330
第 23 章 SAP 診断モジュールの設定	331
SAP 診断がサポートしている環境	331
SAP 診断モジュールの概要	332
SAP 診断モジュールの設定	332

第 6 部 : シナリオの監視

第 24 章 オンライン監視	339
オンライン監視について	339
監視環境の設定	340
モニタの種類	341
コントローラでのモニタおよび測定値の選択	344
コントローラでのモニタの起動	358
コントローラでオンライン・モニタ・グラフを開く	360
オンライン・モニタ表示ビューのカスタマイズ	361
モニタ・オプションの設定	362
第 25 章 オンライン・グラフの設定	367
オンライン・モニタ・グラフについて	367
グラフのプロパティの設定	369
グラフの測定値の構成	373
グラフの結合	378
オンライン・モニタ・グラフのエクスポート	379
オフラインでのデータ表示	379
利用可能なグラフ・ツリー	379

第 7 部 : 付録

付録 A LoadRunner オンライン・グラフの解析	383
オンライン・モニタ・グラフ	383
付録 B パス変換の実行	387
パス変換について	387
パス変換テーブルへのエントリの追加	389
パス変換テーブルの編集	390
パス変換の例	391
付録 C エキスパート・モードでの作業	393
エキスパート・モードへの切り替え方	393
オプション — 一般設定	394
オプション — デバッグ情報の設定	395
オプション — 出力の設定	396
オプション — モニタの設定	398
ロード・ジェネレータ情報 — UNIX 環境の設定	398
ロード・ジェネレータ情報 — 接続ログの設定	399

付録 D コントローラのトラブルシューティング	401
トラブルシューティングについて	401
LoadRunner 通信	402
ロード・ジェネレータとの通信の失敗	403
AUT データベースへの接続の失敗	409
ファイルへのアクセスの失敗	409
仮想ユーザまたはトランザクションの失敗	411
Windows マシンの仮想ユーザ数の引き上げ	415
ファイアウォールのトラブルシューティング	416
LoadRunner エージェントを使った作業	423
付録 E マルチ IP アドレスの設定	427
マルチ IP アドレスについて	427
ロード・ジェネレータへの IP アドレスの追加	428
IP ウィザードの使用	429
UNIX での複数 IP アドレスの設定	433
ルーティング・テーブルの更新	435
コントローラでの複数 IP アドレス機能の有効化	436
付録 F コントローラのコマンド・ライン引数	437
コントローラのコマンド・ライン引数について	437
コマンド・ラインからのコントローラの起動	437
Quality Center の引数	438
実行時環境引数	439
付録 G デジタル証明書の使い方	441
ファイアウォールでのデジタル証明書の使用	441
デジタル証明書の作成と使用	442
索引	447

本書について

『Mercury LoadRunner コントローラ・ユーザズ・ガイド』へようこそ。本書では、LoadRunner のテスト工程の概要を説明し、Windows 環境での LoadRunner コントローラを使用した LoadRunner シナリオの作成および実行方法について説明します。

LoadRunner は、Mercury が提供するアプリケーションのパフォーマンスをテストするツールです。LoadRunner は、アプリケーション全体に負荷をかけて、クライアント、ネットワーク、サーバの潜在的なボトルネックを検出、特定します。

LoadRunner では、制御された負荷およびピーク時の負荷のもとでのシステムの動作をテストできます。LoadRunner は、ネットワーク上に分散している多数の仮想的なユーザ、つまり**仮想ユーザ**を実行することによって負荷を生成します。仮想ユーザは、UNIX および Windows のプラットフォームで動作します。こうした仮想ユーザは、最小限のハードウェア・リソースしか使わずに、一貫性を維持し、再現性があり、測定可能な負荷を生み出して、お使いのアプリケーションをあたかも実際のユーザが使っているかのように動作させます。LoadRunner の詳細なレポートとグラフは、アプリケーションのパフォーマンスを評価するために必要な情報を提供します。

本書の構成

本書は次の部で構成されています。

第 1 部 LoadRunner について

この部では、LoadRunner コントローラを紹介し、負荷テストでの使用方法について説明します。

第 2 部 シナリオの設計

この部では、負荷テスト用のシナリオの作成方法について説明します。

第 3 部 シナリオの実行

この部では、シナリオの実行方法と、シナリオ実行中にシステムのパフォーマンスを表示する方法について説明します。

第 4 部 ファイアウォールを使った作業

この部では、ファイアウォールをサービス環境における LoadRunner コントローラの使用方法について説明します。

第 5 部 診断を使った作業

この部では、LoadRunner 診断モジュールを使用して、Siebel、Oracle、SAP、J2EE および .NET 環境でパフォーマンスの問題を特定し切り分ける方法について説明します。

第 6 部 シナリオの監視

この部では、LoadRunner オンライン・モニタを使用してシナリオの実行を監視する方法について説明します。

第 7 部 付録

この部では、LoadRunner の使用に関するその他の情報を取り上げます。

LoadRunner 印刷マニュアル

LoadRunner には、次の印刷マニュアルが付属しています。

『**Mercury LoadRunner コントローラ・ユーザーズ・ガイド**』は、Windows 環境で LoadRunner コントローラを使って LoadRunner シナリオを作成し実行する方法について説明します。

『**Mercury LoadRunner インストール・ガイド**』は、次のインストール方法について説明します。

- ▶ LoadRunner コントローラ – Windows ベースのマシンへのインストール
- ▶ 仮想ユーザ・コンポーネント – Windows マシンおよび UNIX プラットフォーム用
- ▶ その他の LoadRunner コンポーネント

『**Mercury LoadRunner モニタ・リファレンス**』は、サーバ・モニタ環境のセットアップ方法と、LoadRunner モニタを設定してシナリオ実行中に生成されたデータの監視を行う方法について説明します。

『**Mercury LoadRunner アナリシス・ユーザーズ・ガイド**』は、シナリオの実行後に LoadRunner アナリシスのグラフとレポートを使用してシステムのパフォーマンスを分析する方法について説明します。

『**Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド**』は、VuGen を使ったスクリプトの作成方法を説明します。GUI スクリプトの作成については、必要に応じ、このマニュアルと併せてオンラインの『**Mercury LoadRunner 関数リファレンス**』と『**WinRunner ユーザーズ・ガイド**』をお読みください。

『**Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド**』のオンライン版は1つのボリュームですが、印刷版は「第1巻 - VuGen の使用」および「第2巻 - プロトコル」の2冊から成ります。

『**Mercury Diagnostics for J2EE & .NET 3.6 Supporting LoadRunner and Performance Center Version 8.1 Installation and User's Guide**』(英語版)は、Mercury Diagnostics for J2EE & .NET を使用して、J2EE および .NET プラットフォームでアプリケーションのパフォーマンスを監視する方法について説明します。

『Mercury LoadRunner チュートリアル』は、自分のペースで学習できる、印刷可能なガイドです。負荷テストの全工程について説明しており、LoadRunner テスト環境の理解を深めることができます。

LoadRunner オンライン・マニュアル

LoadRunner には、次のオンライン・マニュアルがあります。

最初にお読みください：LoadRunner に関する最新のお知らせと情報を提供します。「最初にお読みください」には、[スタート] メニューからアクセスします。

Mercury LoadRunner クイック・スタート：LoadRunner の概要を簡潔に順を追って説明し、その使用法を紹介します。「クイック・スタート」には、[スタート] メニューからアクセスします。

オンライン文書：PDF 版のマニュアルが含まれます。[ヘルプ] ボタンをクリックし、[オンライン文書] を選択します。

Mercury LoadRunner オンライン・ヘルプ：次のヘルプが含まれます。

- ▶ **Error Codes Troubleshooting** には、コントローラ接続および Web プロトコル・エラーの分かりやすい説明とトラブルシューティング、および Winsock, SAPGUI, Citrix プロトコルに関する一般的なトラブルシューティングのヒントが含まれます。
- ▶ **LoadRunner Agent Configuration Tool Online Help**：Agent Configuration Tool に関するヘルプです。このヘルプには、[スタート] メニューから [Agent Configuration] ダイアログ・ボックスにアクセスし、[ヘルプ] ボタンをクリックしてアクセスします。
- ▶ **LoadRunner Controller and Monitor Automation Reference**：LoadRunner コントローラを実行し、コントローラのユーザ・インタフェースで使用できるほとんどのアクションを実行するためのプログラムを書くことのできるインタフェースです。LoadRunner のオンライン文書からアクセスできます。
- ▶ **LoadRunner 関数リファレンス**：仮想ユーザ・スクリプトの作成時に使用する LoadRunner の関数をすべて、その使用例と共に参照できます。オンライン版の「Mercury LoadRunner 関数リファレンス」のアップデートについては、Mercury のカスタマー・サポート Web サイトをご覧ください。

任意の LoadRunner コントローラ・ウィンドウで、ウィンドウをクリックし、**F1** または [**ヘルプ**] ボタンを押してオンライン・ヘルプを使用することもできます。

その他のオンライン・リソース

Mercury Tours サンプル Web サイト：本書で説明する多くの例で使用されています。**Mercury Tours** にアクセスするには、次のようにします。**Mercury Tours** Web サイトの URL は、<http://newtours.mercuryinteractive.com> です。

オンライン・カスタマー・サポート：普段お使いの Web ブラウザで、Mercury のカスタマー・サポート Web サイトを開きます。このサイトで、Mercury カスタマー・サポートのナレッジ・ベースを参照し、独自の項目を追加できます。また、ユーザ・ディスカッション・フォーラムへの書き込みや検索、サポート要求の送信、パッチや更新された文書のダウンロードなどを行うこともできます。Web サイトの URL は <http://www.mercury.com/jp/> です。または、[**ヘルプ**] ボタンをクリックして、[**サポート情報**] を選択します。

Send Feedback を使用すると、製品チームに<製品>に関するフィードバックをオンラインで送信できます。[**ヘルプ**] > [**バージョン情報**] をクリックします。製品にこのボタンが含まれている場合はこれを含めます。

Mercury Interactive の Web サイト：普段お使いの Web ブラウザで Mercury の Web サイトを開きます。このサイトでは、Mercury の最新情報や製品に関する情報をご覧になれます。新しいソフトウェアのリリース、セミナー、展示会、カスタマー・サポート、教育サービスなどに関する情報をご覧いただけます。[**ヘルプ**] ボタンをクリックして、[**Mercury Interactive の Web サイト**] を選択します。Mercury の Web サイトの URL は、<http://www.mercury.com/jp> です。

表記規則

本書では次の表記規則に従います。

UI 要素	この形式で、アクションを実行するインタフェース要素の名前、ファイル名、パス、および注目すべきその他の項目を示します。 例：[保存] ボタンをクリックします。
引数	この形式で、メソッド、関数の引数、書名を示します。 例：『 Mercury ユーザーズ・ガイド 』を参照してください。
< 置換する値 >	実際の値と置換するファイル・パスや URL アドレスの一部は大括弧で囲みます。 例：< 製品のインストール・フォルダ > %bin
使用例	使用例やユーザがそのまま入力しなければならない文字列は、この形式で示します。例：編集ボックスに「 Hello 」と入力します。
関数名	メソッド名や関数名はこの形式で示します。 例： wait_window ステートメントには次のパラメータがあります。
CTRL	キーボードのキーはこの形式で示します。
[]	半角の大括弧は、省略可能な引数を囲みます。
{ }	引数に割り当てる値の候補は、中括弧で囲んで示します。
...	構文内の省略記号は、同じ形式で項目をさらに組み入れることができることを意味しますプログラム例での3つの点は、プログラム行が意図的に削除されていることを示します。
	2つの値のうちの1つを選択しなければならない場合、これらの値を垂直バーで区切ります。

第 1 部

LoadRunner について

第 1 章

はじめに

LoadRunner は、複数のユーザが同時に作業している環境をエミュレートすることによってアプリケーションの負荷テストを行います。アプリケーションに負荷をかけている間、LoadRunner は、システムのパフォーマンスと機能の正確な測定、監視、分析を行います。

アプリケーションの負荷テスト

現代のシステム・アーキテクチャは非常に複雑です。かつてないパワーと柔軟性を提供しますが、こうしたシステムをテストするのは困難です。シングルユーザ・テストの主な対象は、1つのシステム・コンポーネントの機能とユーザ・インタフェースですが、アプリケーション・テストの主な対象は、システム全体のパフォーマンスと信頼性です。

例えば、アプリケーション・テストのシナリオの典型例に、月曜の朝、1000人のユーザがシステムに一斉にログインする様子をエミュレートするというのがあります。システムの応答時間はどうか。システムはクラッシュしなかったか。こうした疑問に答え、さらに多くの役割を果たすために、完成されたアプリケーション・パフォーマンス・テスト・ソリューションは、次のことを行う必要があります。

- ▶ 様々なソフトウェア・アプリケーションとハードウェア・プラットフォームを組み合わせたシステムのテスト
- ▶ 特定のアプリケーションに対するサーバの適合性の調査
- ▶ 必要なクライアント・ソフトウェアの開発完了に先立つサーバのテスト
- ▶ 複数クライアントと単独サーバ・アプリケーションとの対話環境のエミュレーション
- ▶ 数十、数百、あるいは数千の潜在ユーザの負荷をかけた状態でのアプリケーションのテスト

手作業によるテストの限界

従来の手作業によるテストでは、負荷テストに対するソリューションの一部しか提供されません。例えば、システムで多数のユーザが同時に作業する環境を構築すれば、システム全体を手作業でテストすることができます。各ユーザは、1台のマシンで作業し、システムに入力を送ります。しかし、この手作業によるテストには、次のような問題点があります。

- ▶ コストがかかる。人員と機材がどちらも大量に必要。
- ▶ 複雑である。特に、多数のテスト要員をまとめ、一斉に作業を行う点において。
- ▶ 高度な体系化が必要。特に、意味のある結果の記録と分析を行うために。
- ▶ 手作業によるテストの反復の限界。

LoadRunner のソリューション

LoadRunner は、上記の手作業によるパフォーマンス・テストの問題点を解決します。

- ▶ LoadRunner は、実際のユーザを**仮想ユーザ (Vuser)** に置き換え、テストに必要な人員を減らします。仮想ユーザは、実ユーザの振る舞い、つまり実際のアプリケーションの操作をエミュレートします。
- ▶ 1台のコンピュータで多数の仮想ユーザを実行できるので、LoadRunner はテストに必要なハードウェア数を減らします。
- ▶ LoadRunner コントローラを使って、仮想ユーザを1か所から簡単に効率よく制御できます。
- ▶ LoadRunner はアプリケーションのパフォーマンスをオンラインで監視するので、テスト実行中にシステムを適切に調節できます。
- ▶ LoadRunner はテスト中に、アプリケーションのパフォーマンスを自動的に記録します。様々なグラフやレポートから、表示するパフォーマンス・データを選択できます。
- ▶ LoadRunner は、パフォーマンス遅延がどこで発生したかを調べます。ネットワークまたはクライアントの遅延、CPUのパフォーマンス、I/Oの遅延、データベースのロック、あるいはデータベース・サーバにおけるその他の問題などです。LoadRunner は、ネットワークとサーバのリソースを監視して、パフォーマンスの向上を支援します。

- ▶ LoadRunner のテストは完全に自動化されているので、必要なだけ何回でも簡単にテストを繰り返せます。

LoadRunner の使用

シナリオ	LoadRunner を使って、アプリケーション・パフォーマンス・テストの要件を、いくつかの シナリオ に分割できます。シナリオで、各テスト・セッション中に発生するイベントを定義します。このようにシナリオでは、例えば、エミュレートするユーザ数、ユーザが実行するアクション、エミュレーションを実行するマシンなどを定義および制御します。
仮想ユーザ	シナリオの中で、LoadRunner は実際のユーザを 仮想ユーザ (Vuser) で置き換えます。シナリオを実行すると、仮想ユーザは実際のユーザのアクションをエミュレートし、アプリケーションを操作します。1 台のワークステーションを使えるのは 1 度に 1 人だけですが、1 台のワークステーションで同時に多数の仮想ユーザを実行できます。実際、シナリオには、数十、数百、あるいは数千もの仮想ユーザを含めることができます。
仮想ユーザ・スクリプト	仮想ユーザ・スクリプトには、シナリオの中で仮想ユーザが実行するアクションが記述されます。シナリオを実行すると、各仮想ユーザが 仮想ユーザ・スクリプト を実行します。仮想ユーザ・スクリプトは、アプリケーションのコンポーネントのパフォーマンスを測定および記録する関数を含んでいます。
トランザクション	サーバのパフォーマンスを測定するには、 トランザクション を定義します。トランザクションは、測定対象のアクションまたはアクションの集合を表します。仮想ユーザ・スクリプト内でトランザクションを定義するには、スクリプトの適切なセクションをトランザクション・ステートメントの 開始 と 終了 で囲みます。例えば、サーバが口座の残高表示要求を処理して、ATM に情報が表示されるのにかかる時間を測定するトランザクションを定義できます。
ランデブー・ポイント	仮想ユーザ・スクリプトに ランデブー・ポイント を挿入して、サーバにかかる大きなユーザ負荷をエミュレートします。 ランデブー・ポイント はテスト実行中に仮想ユーザを待機させ、複数の仮想ユーザが特定のポイントに到着したときに、それらが同時にタスクを実行するようにします。例えば、銀行のサーバのピーク時の負荷をエミュレートするためのランデブー・ポイントを挿入して、100 仮想ユーザに同時に、それぞれの口座に預金させることができます。
コントローラ	LoadRunner コントローラ を使って、シナリオの管理と保守を行います。コントローラを使って、シナリオのすべての仮想ユーザを 1 台のワークステーションから制御します。
ロード・ジェネレータ	シナリオを実行するときには、LoadRunner コントローラによってシナリオの各仮想ユーザが ロード・ジェネレータ に分配されます。ロード・ジェネレータは

仮想ユーザ・スクリプトを実行するマシンで、仮想ユーザが実際のアクションをエミュレートできるようにします。

パフォーマンス
の分析

仮想ユーザ・スクリプトには、負荷テスト・セッション中に、システム・パフォーマンスの測定と記録を行う関数が含まれています。シナリオ実行中に、ネットワークとサーバのリソースを監視できます。シナリオ実行後、レポートとグラフを使ってパフォーマンスの分析データを表示できます。

LoadRunner を使った作業

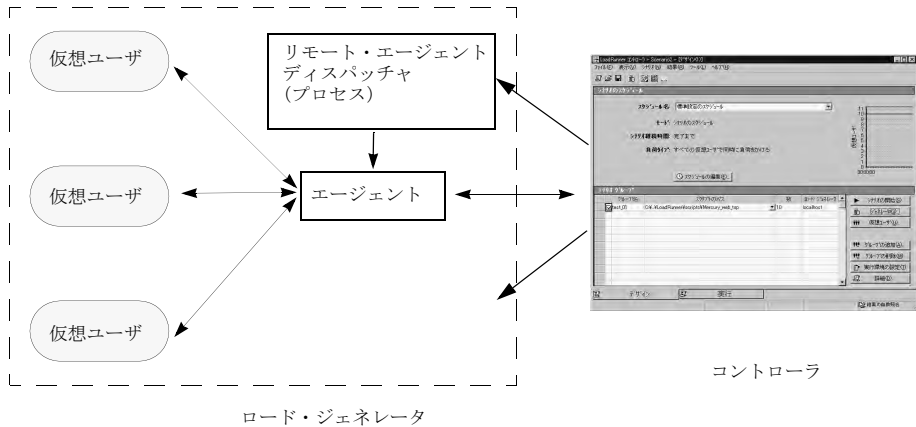
多数のインターネット・ユーザがアクセスするオンライン・バンキングの Web サーバをテストするものとします。この Web サイトは顧客に、振り込みや残高照会などの、各種バンキング・サービスを提供します。このサーバをテストするために、シナリオを作成します。シナリオでは、負荷テスト中にサーバで行われるアクションを定義します。

銀行のサーバに負荷をかけ、監視するシナリオの中では、次のことを実行します。

- ▶ サーバにおける制御された負荷条件のエミュレート
- ▶ サーバにおける最大負荷条件のエミュレート
- ▶ 負荷の下でのサーバ・パフォーマンスの測定
- ▶ パフォーマンス低下の原因箇所の特定：ネットワークまたはクライアントの遅延、CPU のパフォーマンス、I/O の遅延、データベースのロック、あるいはその他のサーバ上の問題
- ▶ 負荷のかかった状態でのネットワークとサーバ・リソースの監視

LoadRunner 仮想ユーザ技術

各 Windows ロード・ジェネレータに、**リモート・エージェント・ディスパッチャ (プロセス)** および **LoadRunner エージェント** をインストールします。



リモート・
エージェント・
ディスパッチャ
(プロセス)

リモート・エージェント・ディスパッチャ (プロセス) を使って、コントローラはロード・ジェネレータ・マシン上でアプリケーションを起動できます。

エージェント

LoadRunner エージェントを使って、コントローラとロード・ジェネレータは相互に通信できます。シナリオを実行すると、コントローラがリモート・エージェント・ディスパッチャ (プロセス) に LoadRunner エージェントを起動させます。エージェントは、コントローラから、仮想ユーザの初期化、実行、一時停止、停止に関する指示を受け取ります。同時にエージェントは、仮想ユーザのステータスのデータをコントローラに返します。

LoadRunner 仮想ユーザの種類

LoadRunner では、様々な種類の仮想ユーザを使用できます。それぞれの種類の仮想ユーザは、今日のシステム・アーキテクチャの様々な面を処理できるように設計されています。シナリオ内で、任意の種類の仮想ユーザを組み合わせて使用し、総合的なアプリケーション・テストを作成できます。次の種類の仮想ユーザを使用できます。

▶ Application Deployment ソリューション

Citrix プロトコルの場合。

▶ クライアント/サーバ

MSSQL, ODBC, Oracle Web Applications 11i, DB2 CLI, Sybase Ctlib, Sybase Dblib, Windows Sockets, および DNS プロトコルの場合。

▶ **カスタム**

C テンプレート, Visual Basic テンプレート, Java テンプレート, Javascript および VBScript タイプのスクリプトの場合。

▶ **分散コンポーネント**

COM/DCOM, Corba-Java, および Rmi-Java プロトコルの場合。

▶ **E ビジネス**

FTP, LDAP, Palm, Web (HTTP/HTML), Web サービス, およびデュアル Web/Winsocket プロトコルの場合。

▶ **エンタープライズ Java ビーン**

EJB Testing および Rmi-Java プロトコルの場合。

▶ **ERP/CRM**

Baan, Oracle NCA, Peoplesoft 8, Peoplesoft-Tuxedo, SAP-Web, SAPGUI, SAPGUI/SAP-Web dual, および Siebel (Siebel-DB2 CLI, Siebel-MSSQL, Siebel-Web, および Siebel-Oracle) プロトコルの場合。

▶ **レガシ**

端末エミュレータ (RTE) の場合。

▶ **メールング・サービス**

インターネット・メッセージング (IMAP), MS Exchange (MAPI), POP3, および SMTP。

▶ **ミドルウェア**

Jacada および Tuxedo (6, 7) プロトコルの場合。

▶ **ストリーミング**

MediaPlayer および RealPlayer プロトコルの場合。

▶ **ワイヤレス**

i モード, VoiceXML, WAP プロトコルの場合。

GUI 仮想ユーザ

GUI 仮想ユーザ

GUI 仮想ユーザは、グラフィカル・ユーザ・インタフェース (GUI) アプリケーションを操作します。こうしたアプリケーションを、Microsoft Windows 環境で実行できます。作成された各 GUI 仮想ユーザは、GUI アプリケーションに入力を送り、その出力を受け取ることで、実際のユーザをエミュレートします。例えば、GUI 仮想ユーザで、Microsoft ペイントを次のように操作できます。

1. [ファイル] メニューで [開く] を選択する。
2. test.bmp という名前の画像ファイルを選択する。
3. [開く] ボタンをクリックする。
4. [変形] メニューで [反転と回転] を選択する。
5. [水平方向] ラジオ・ボタンをクリックする。
6. [OK] ボタンをクリックする。
7. [ファイル] メニューで [名前を付けて保存] を選択する。

次の画像は結果を示しています。



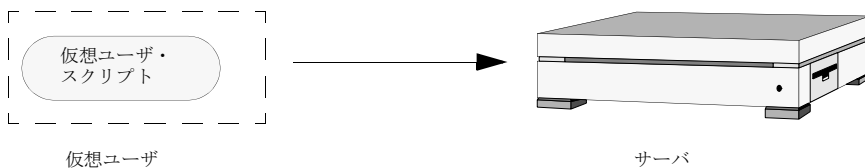
GUI 仮想ユーザがアプリケーションに対して実行する操作は、GUI 仮想ユーザ・スクリプトに定義されます。Mercury の GUI テスト・ツールを使って、GUI 仮想ユーザ・スクリプトを作成します。WinRunner (Microsoft Windows のアプリケーションの場合) および Astra QuickTest (Web アプリケーションの場合) を使用します。

Windows マシンのロード・ジェネレータでは、GUI 仮想ユーザを 1 つだけ実行できます。複数の GUI 仮想ユーザを実行するには Citrix を使用します。Citrix を使用するロード・ジェネレータの設定の詳細については、Readme ファイルを参照してください。Windows マシンの GUI 仮想ユーザの詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド』を参照してください。

注：リモート・ロード・ジェネレータ上で GUI 仮想ユーザを実行できるのは、リモート・エージェント・ディスパッチャをプロセスとしてインストールしている場合に限られます。リモート・エージェント・ディスパッチャをサービスとしてインストールしている場合には、GUI 仮想ユーザをリモート・ロード・ジェネレータ上で実行できません。

仮想ユーザ技術

仮想ユーザ (GUI 仮想ユーザと RTE 仮想ユーザを除く) は、サーバに入力を直接送信してサーバの負荷を生成します。仮想ユーザはクライアント・アプリケーションを操作しません。LoadRunner API 関数を使用してサーバにアクセスします。この API 関数が、実際のアプリケーションからの入力をエミュレートします。



仮想ユーザはクライアント・ソフトウェアに依存しないので、クライアント・ソフトウェアの開発が終わる前でも、仮想ユーザを使ってサーバのパフォーマンスをテストできます。さらに、仮想ユーザはユーザ・インタフェースを持たないので、最小限のシステム・リソースしか必要としません。このため、1 台のワークステーションで多数の仮想ユーザを実行できます。

次の例で、仮想ユーザの使い方を説明します。顧客情報を保持する Web ベースのデータベース・サーバがあるものとします。全国各地の多数の顧客サービス担当者が、この情報にアクセスします。サーバはクエリを受け取り、要求を処理し、顧客サービス担当者に Web 経由で応答を返します。

多数のサービス担当者が同時にサーバにアクセスしたときの、システム全体の応答時間をテストするものとします。LoadRunner を使って、サーバ・データベースにアクセスする数百の仮想ユーザを作成できます。これらの仮想ユーザにより、多数のユーザの負荷がかかった状態でのデータベースおよび Web サーバのパフォーマンスのエミュレートと測定ができます。

仮想ユーザのアクションを定義する仮想ユーザ・スクリプトを開発します。仮想ユーザ・スクリプトには、スクリプトの実行の制御、仮想ユーザがサーバに送る入力指定、およびサーバ・パフォーマンスの測定を行う関数が含まれています。

仮想ユーザ・スクリプトは、Mercury の仮想ユーザ・ジェネレータ (VuGen) で記録するか、LoadRunner の仮想ユーザ・スクリプト・テンプレートを使用することによって開発します。

上のデータベース・サーバの例で、次のアクションを実行する仮想ユーザ・スクリプトを作成できます。

- ▶ Web アプリケーションへのログイン
- ▶ データベース・サーバへの接続
- ▶ SQL クエリの発行
- ▶ サーバの応答の受け取りと処理
- ▶ サーバおよび Web との接続の切断

仮想ユーザ・スクリプトは、Windows ベースのプラットフォームで作成することも、UNIX プラットフォームでプログラムすることもできます。サポートされている UNIX プラットフォームのリストについては、LoadRunner の Readme ファイルを参照してください。仮想ユーザの詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド』を参照してください。

RTE 仮想ユーザ

RTE 仮想ユーザ

RTE 仮想ユーザは文字ベースのアプリケーションを操作します。開発する各 RTE 仮想ユーザは、文字ベースのアプリケーションに入力を送り、その出力を受け取ることで、実際のユーザをエミュレートします。

```
View - Vuser RTE/0
=====
operations:
  1) Withdraw Cash.
  2) Deposit Cash.
  3) Balance Report.
  4) Exit ATM.

Please select (1-4): 2
Enter amount of money to deposit: 168
Depositing $168 in process, Please wait...
Operation has been successfully completed.

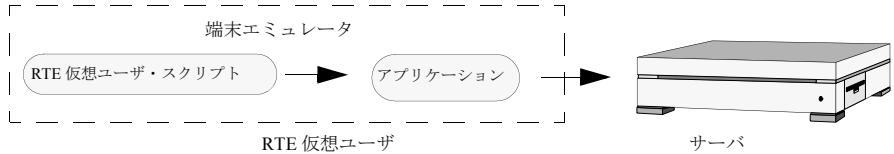
ATM Services
=====
operations:
  1) Withdraw Cash.
  2) Deposit Cash.
  3) Balance Report.
  4) Exit ATM.

Please select (1-4): 
```

次の例で、RTE 仮想ユーザの使い方を説明します。顧客情報を保持するデータベース・サーバがあるものとします。全国各地の多数のフィールド・サービス担当者が、この情報にアクセスします。フィールド・サービス担当者は修理のたびに、モデムでサーバ・データベースにアクセスします。サービス担当者は文字ベースのアプリケーションを使って顧客の不満を記録し、顧客の詳細な情報にアクセスします。

多数のサービス担当者が同時にサーバにアクセスしたときの、サーバの応答時間をテストするものとします。LoadRunner を使って、文字ベースのアプリケーションでサーバ・データベースにアクセスする数百の RTE 仮想ユーザを作成できます。これらの RTE 仮想ユーザにより、多数のユーザの負荷がかかった状態でのサーバのパフォーマンスのエミュレートと測定ができます。

RTE 仮想ユーザがアプリケーションに対して実行する操作は、RTE 仮想ユーザ・スクリプトに定義されます。VuGen を使用して、RTE 仮想ユーザを作成します。ジェネレータを使用して、文字ベースのアプリケーションに対するアクションを記録できます。



仮想ユーザの詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

LoadRunner ライセンス情報

ライセンス・キー情報を確認するには、[スタート] > [プログラム] > [Mercury LoadRunner] > [LoadRunner] をクリックします。Mercury LoadRunner のランチャ画面が開きます。[設定] メニューから、[LoadRunner ライセンス] を選択します。[LoadRunner ライセンス情報] ダイアログ・ボックスが開きます。



[LoadRunner ライセンス情報] ダイアログ・ボックスに次の情報が表示されます。

[ライセンス キー]: 利用可能なライセンス・キーと、利用可能なすべてのライセンス・キーのサマリを表示します。

[ライセンス キー情報]

[タイプ]：選択したライセンス・キーに有効なライセンスの種類を表示します。利用できるライセンスの種類は次のとおりです。

- ▶ **Parmanent**：ライセンスは無期限です。
- ▶ **Time Limited**：ライセンスは開始日から期限までに限定されます。
- ▶ **Temporary**：製品をインストールしてから、既定の日数だけ使用できるライセンスです。
- ▶ **VUD-based**：仮想ユーザ日（VUD）数で制限されたライセンスです。VUD ライセンスを使用すると、24 時間、製品を何回でも使用できます。

注：ユーザが 1000 仮想ユーザ用の VUD ベースのライセンスを持っており、24 時間で同時に実行する仮想ユーザの最大数が 300 の場合は、残りの 700 仮想ユーザは翌日実行できます。

- ▶ **Plugged**：このライセンスには dongle が必要です。

[ライセンスの有効性]：選択されたライセンス・キーがいつまで有効かを表示します。

[仮想ユーザのタイプ]：選択されたライセンス・キーに使用できる仮想ユーザ・プロトコルの一覧を表示します。

- ▶ **Group < n >**：組み合わせで使われることの多いプロトコル・グループ。
- ▶ **Global**：Global ライセンスは、仮想ユーザの総数内であれば、どのタイプの仮想ユーザでも動かすことができます。Global の後の数字は、仮想ユーザの総数です。

[モニタ]：選択されたライセンス・キーで利用できるサーバ・モニタが表示されます。

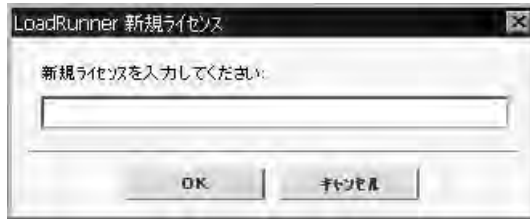
[ホスト ID]：特定のマシンの ID が表示されます。お使いのマシンのライセンス・キーを取得するには、Mercury のカスタマー・サポートまでお問い合わせください。

現在のライセンス情報を変更するには、次の手順を実行します。

- 1 [スタート] > [プログラム] > [Mercury LoadRunner] > [LoadRunner] をクリックして Mercury LoadRunner ランチャ・ウィンドウを開きます。[設定]

メニューから、[LoadRunner ライセンス] を選択します。[LoadRunner ライセンス] ボタンをクリックします。[LoadRunner ライセンス情報] ダイアログ・ボックスが開きます。

- 2 [新規ライセンス] ボタンをクリックします。[LoadRunner 新規ライセンス] ダイアログ・ボックスが開きます。



- 3 新しいライセンス番号を正確に入力し、[OK] をクリックします。ライセンスに有効期限が設けられている場合は、適宜メッセージが表示されます。
- 4 [OK] をクリックして、[LoadRunner 新規ライセンス] ダイアログ・ボックスを閉じます。

第 2 章

LoadRunner テスト工程

本章で説明する LoadRunner のテスト工程で、負荷テストのシナリオの作成と実行が簡単にできます。次の図に、テスト工程の概要を示します。



本章では、LoadRunner で Web ベースのアプリケーションの負荷テストを行う際のおおまかな手順を 6 つのステップに分けて説明します。

ステップ I : テストの計画

負荷テストを成功させるには、綿密なテスト計画を作成する必要があります。テスト計画を明確に定義しておけば、負荷テストの目標を確実に達成する LoadRunner のシナリオを作成できます。詳細については、第3章「負荷テストの計画」を参照してください。

ステップ II : 仮想ユーザ・スクリプトの作成

仮想ユーザは、Web ベースのアプリケーションと対話する実際のユーザをエミュレートします。仮想ユーザ・スクリプトには、シナリオ実行中に各仮想ユーザが実行するアクションが含まれています。

各仮想ユーザ・スクリプトに、以下のタスクを明記します。

- ▶ 各仮想ユーザが実行するタスク
- ▶ 複数の仮想ユーザが同時に実行するタスク
- ▶ トランザクションとして測定されるタスク

仮想ユーザ・スクリプトの作成の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

ステップ III : シナリオの作成

シナリオには、テスト・セッション中に発生するイベントが記述されます。シナリオには、仮想ユーザを実行するマシンのリスト、仮想ユーザが実行するスクリプトのリストが含まれ、シナリオ実行時に活動する仮想ユーザ数または仮想ユーザ・グループ数が指定されています。LoadRunner コントローラを使ってシナリオを作成します。コントローラの紹介は、第4章「LoadRunner コントローラの概要」を参照してください。

マニュアル・シナリオの作成

スクリプトを実行するには、仮想ユーザ・グループに割り当てる仮想ユーザ数、仮想ユーザ・スクリプト数、ロード・ジェネレータ数をそれぞれ定義し、シナリオを作成します。マニュアル・シナリオの作成の手順は、第5章「マニュアル・シナリオの作成」を参照してください。

また、パーセント・モードを使ってシナリオを作成できます。パーセント・モードでは、シナリオ内で使う仮想ユーザの合計数、ロード・ジェネレータ・

マシンの合計数、そして仮想ユーザの合計数のうち各仮想ユーザ スクリプトに割り当てる割合を定義します。パーセント・モードでマニュアル・シナリオを作成する手順は、第 6 章「パーセント・モードでのマニュアル・シナリオの作成」を参照してください。

ゴール指向シナリオの作成

Web テストのためにゴール指向シナリオを作成し、テストで達成すべきゴールを定義します。LoadRunner はこのゴールに基づいて自動的にシナリオを作成します。ゴール指向シナリオを作成する手順は、第 7 章「ゴール指向シナリオの作成」を参照してください。

ステップ IV : シナリオの実行

複数の仮想ユーザがタスクを同時に実行するよう指定することにより、サーバ上でユーザの負荷をエミュレートします。タスクを同時に実行する仮想ユーザ数を増減することによって、負荷のレベルを設定できます。詳細については、第 9 章「ランデブー・ポイントの使用」を参照してください。

シナリオを実行する前に、シナリオの構成とスケジュールを設定します。これによって、シナリオ実行時にすべてのロード・ジェネレータと仮想ユーザがどのように動作するかを決定します。詳細については、第 10 章「シナリオの設定」および第 8 章「シナリオのスケジュール」を参照してください。

シナリオ全体を実行することはもちろん、仮想ユーザのグループ（仮想ユーザ・グループ）ごと、あるいは個々の仮想ユーザごとにシナリオを実行できます。シナリオ実行中、LoadRunner は各仮想ユーザ・スクリプトで定義されているトランザクションを測定し、記録します。システム・パフォーマンスをオンラインで監視することもできます。詳細については、第 3 部「シナリオの実行」を参照してください。

ステップ V : シナリオの監視

LoadRunner の実行環境、トランザクション、システム・リソース、Web リソース、Web サーバ・リソース、Web アプリケーション・サーバ・リソース、データベース・サーバ・リソース、ネットワーク遅延、ストリーム・メディア・リソース、ファイアウォール・サーバ・リソース、ERP サーバ・リソース、Java パフォーマンス、J2EE & .NET 診断、Application Deployment、ミドルウェア・パフォーマンス、アプリケーション・コンポーネント・モニタ、およびインフラ

ストラクチャ・リソース・モニタを使って、シナリオの実行をオンラインで監視できます。詳細については、第 6 部「シナリオの監視」を参照してください。

ステップ VI : テスト結果の分析

シナリオ実行中、LoadRunner は、様々な負荷の下でアプリケーション・パフォーマンスを記録します。LoadRunner のグラフやレポートを使って、アプリケーションのパフォーマンスを分析します。LoadRunner のレポートとグラフの詳細については、『**Mercury LoadRunner アナリシス・ユーザーズ・ガイド**』を参照してください。

第 3 章

負荷テストの計画

包括的なテスト計画の作成は、負荷テストを成功させるための重要なポイントです。テスト計画を明確に定義しておけば、負荷テストの目標を確実に達成する LoadRunner シナリオを開発できます。本章では、負荷テスト計画の工程について説明します。

本章では、次の項目について説明します。

- ▶ 負荷テストの計画について
- ▶ アプリケーションの分析
- ▶ テスト目標の定義
- ▶ LoadRunner の導入計画
- ▶ 負荷テスト目標の調査

負荷テストの計画について

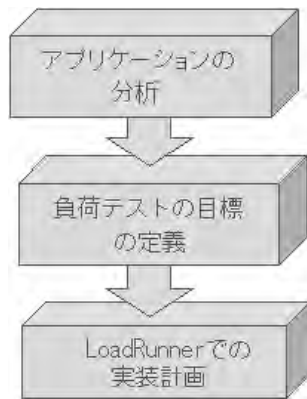
あらゆる種類のシステム・テストと同様に、綿密なテスト計画を定義することが、テストを成功させる重要な第一歩です。負荷テストの計画は、次のことをするのに役立ちます。

- ▶ 作業環境を正確にエミュレートするテスト・シナリオの作成
負荷テストとは、一般的な作業条件の下でアプリケーションをテストし、システム・パフォーマンス、信頼性、能力などを検査することです。
- ▶ テストに必要なリソースについて
アプリケーション・テストには、ハードウェア、ソフトウェア、および人員のリソースが必要です。テストを始める前に、利用可能なリソースを把握し、効果的な使い方を考えておく必要があります。

▶ 測定可能な合格基準の定義

テストの目標と基準を明確にして、有益なテストを実施できるようにします。例えば、「重い負荷の下におけるサーバの応答時間の検査」のような漠然とした目的の定義では不十分です。具体的で有益な基準は、「50人の顧客が同時に口座の残高を照会でき、サーバの応答時間が1分を超えないこと」のようなものになります。

負荷テストの計画は、次の3つのステップから成る作業です。



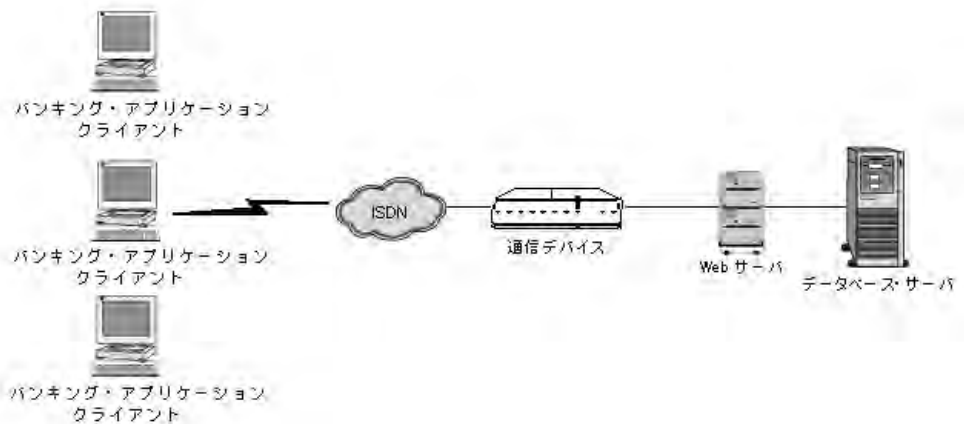
アプリケーションの分析

負荷テストの計画における最初のステップは、アプリケーションの分析です。ハードウェア、ソフトウェア・コンポーネント、システム構成、一般的な使用方を徹底的に理解していなければなりません。この分析により、LoadRunnerを使って作成するテスト環境が、テスト対象アプリケーションの環境と構成を正確に反映したものになります。

システム・コンポーネントの特定

アプリケーションの構造を示す構成図を作成します。可能ならば、既存の文書から構成図を抜き出します。テスト対象アプリケーションが大規模なネットワーク・システムの一部である場合は、テストするシステムのコンポーネントを特定します。構成図には、必ずすべてのシステム・コンポーネント（クライアント・マシン、ネットワーク、ミドルウェア、サーバなど）を含めるようにします。

次の構成図は、多数の Web ユーザによってアクセスされるオンライン・バンキング・システムを示しています。Web ユーザはそれぞれが個別に同じデータベースに接続して、送金や残高照会をします。顧客は複数のブラウザを使い、Web 経由でデータベース・サーバに接続しています。



システム構成の記述

構成図を詳細化します。各システム・コンポーネントの構成を記述します。次の点を明確にする必要があります。

- ▶ システムに接続する予想ユーザ数。
- ▶ アプリケーションのクライアント・マシンの構成（ハードウェア、メモリ、オペレーティング・システム、ソフトウェア、開発ツールなど）。
- ▶ 使用されているデータベースおよび Web サーバのタイプ（ハードウェア、データベースのタイプ、オペレーティング・システム、ファイル・サーバなど）。
- ▶ サーバとアプリケーション・クライアントの通信方法。
- ▶ フロントエンド・クライアントとバックエンド・サーバの間の中継ウェア設定とアプリケーション・サーバ。
- ▶ 応答時間に影響する可能性があるその他のネットワーク・コンポーネント（モデムなど）。

- ▶ 通信デバイスの処理能力。各デバイスが処理できる同時実行ユーザの数。

例えば、上の構成図はシステムに複数のアプリケーション・クライアントがアクセスしていることを示しています。

フロントエンド・クライアントの構成	
予想アプリケーション・クライアント数	50 の同時アプリケーション・クライアント
ハードウェア / メモリ	586 / 32MB
OS とそのバージョン	Windows NT 4.0
クライアント・ブラウザ	Internet Explorer 4.0

使用モデルの分析

システムが一般にどのように使用されているかを定義し、どの機能をテストするか決めます。誰がシステムを使うのか、各タイプのユーザ数、各ユーザの一般的なタスクを検討します。さらに、システムの応答時間に影響する可能性のあるあらゆるバックグラウンド負荷を検討する必要があります。

例えば、200 人の従業員が毎朝会計システムにログオンするものとします。そして、そのオフィス・ネットワークには、様々な文書処理や印字出力を行う 50 人のユーザのバックグラウンド負荷が常にかかっているものとします。このような場合には、会計データベースにログオンする 200 の仮想ユーザを含む LoadRunner シナリオを作成し、サーバの応答時間を検査します。

バックグラウンド負荷が応答時間に及ぼす影響を検査するには、文書処理や印字出力を実行する従業員の負荷もシミュレートするネットワークでシナリオを実行します

タスクの分散

共通のユーザ・タスクを定義するのに加え、これらのタスクの分布を調べます。例えば、銀行が時差のある様々な州の顧客に、1 つの集中化されたデータベースで対応するものとします。時差のある 2 つの地域に配置された 250 のアプリケーション・クライアントが、すべて同じ Web サーバに接続されています。シカゴに 150、デトロイトに 100 とします。どちらも始業は午前 9 時ですが、時差があるため 150 より多くのユーザが一斉にログインすることはありません。タスクの分布を分析して、データベースの動作状況がピークになる時間帯と、**負荷のピーク時**における典型的な動作を特定します。

テスト目標の定義

テスト開始前に、達成する目標を正確に定義しておく必要があります。

次に、Robert W. Buchanan, Jr の『**The Art of Testing Network Systems**』（John Wiley & Sons, Inc., 1996）で説明されているアプリケーション・テストの一般的な目標を示します。LoadRunner はこれらのテストを行うのに役立ちます。

目標	解決すべき疑問
エンド・ユーザの応答時間の測定	ビジネス・プロセスの完了に要する時間は？
最適なハードウェア構成の定義	最高のパフォーマンスをもたらすハードウェア構成は？
信頼性の検査	エラーも障害もなくシステムが動作する負荷の大きさと時間の長さは？
ハードウェアまたはソフトウェア・アップグレードの検査	アップグレードによるパフォーマンスと信頼性への影響は？
新製品の評価	サーバのハードウェアとソフトウェアにどれを選ぶべきか？
システムの許容量の測定	パフォーマンスを大きく低下させることなくシステムがどれだけの負荷を処理できるか？
ボトルネックの特定	どの要素が応答時間を遅らせているか？

各目標の詳細については、この章の最後で説明します。

測定可能な目標の記述

一般的な負荷テストの目標が決まったら、目標を測定可能な形で記述することによって、目標を明確なものにできます。評価のベースラインを定めるには、許容できるテスト結果と許容できないテスト結果を正確に規定します。

次に例を示します。

一般的な目標—製品の評価：Web サーバ用のハードウェアを選択。

明確化された目標—製品の評価：2つの異なるサーバ（HP と NEC）で300の仮想ユーザからなるグループを実行します。300 ユーザがすべて同時に Web アプリケーションにアクセスする。

リケーションのページを表示したときに、どちらのハードウェアの応答時間が短いかを判定します。

いつテストを行うか

負荷テストは、製品のライフ・サイクルを通じて必要です。次の表に、製品のライフ・サイクルの各段階に対応するテストのタイプを示します。

計画と設計	開発	配備	運用	進化
新製品の評価	応答時間の測定	信頼性の検査	応答時間の測定	ハードウェアまたはソフトウェア・アップグレードの検査
応答時間の測定	最適なハードウェア構成の検査	応答時間の測定	ボトルネックの特定	システム能力の測定
	ハードウェアまたはソフトウェア・アップグレードの検査	システム能力の測定		
	信頼性の検査			

LoadRunner の導入計画

次のステップでは、テスト目標を達成するために、LoadRunner をどのように使うか決定します。

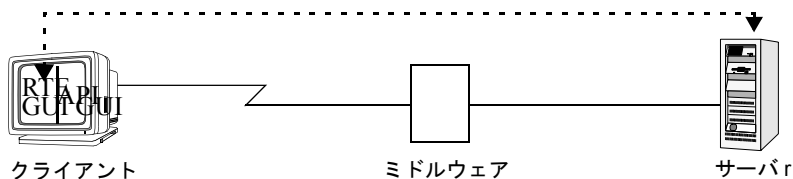
パフォーマンス測定の範囲の定義

LoadRunner を使って、アプリケーションの様々な場所での応答時間を測定できます。テスト目標に従って、どこでどのタイプの仮想ユーザを実行するかを決めます。

- ▶ エンドツーエンドの応答時間の測定：

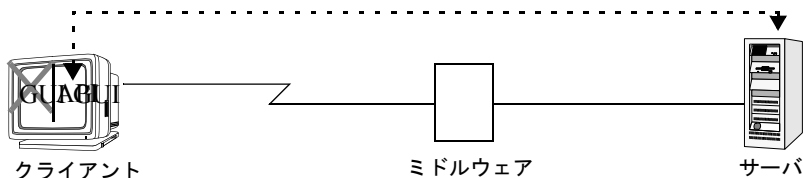
フロントエンドで GUI 仮想ユーザと RTE 仮想ユーザを実行することにより、一般的なユーザが経験する応答時間を測定できます。GUI 仮想ユーザは、クライアント・アプリケーションに入力を送り、その出力を受け取ることによって、実際のユーザをエミュレートします。RTE 仮想ユーザは、文字ベースのアプリケーションに入力を送り、その出力を受け取ることによって、実際のユーザをエミュレートします。

フロント・エンドで GUI 仮想ユーザや RTE 仮想ユーザを実行することにより、端末エミュレータまたは GUI フロント・エンド、ネットワーク、およびサーバを含む、ネットワーク全体の応答時間を測定できます。



▶ ネットワークとサーバの応答時間の測定：

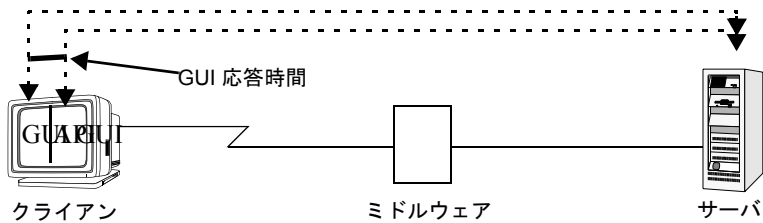
クライアント・マシンで仮想ユーザ（GUI と RTE を除く）を実行することにより、GUI フロント・エンドの応答時間を除いた、ネットワークとサーバの応答時間を測定できます。仮想ユーザは、ユーザ・インタフェースを除外したクライアントからサーバへの呼び出しをエミュレートします。クライアント・マシンから多数の仮想ユーザを実行している場合、負荷がネットワークやサーバの応答時間にどのように影響するか測定できます。



▶ GUI 応答時間の測定：

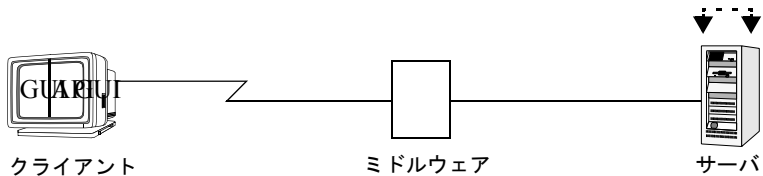
前の2つの測定値の差を計算すれば、クライアント・アプリケーション・インタフェースが応答時間に与える影響がわかります。

GUI 応答時間=エンドツーエンドの応答時間-ネットワークとサーバの応答時間



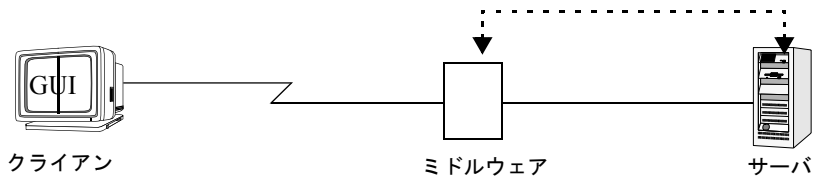
▶ サーバ応答時間の測定：

ネットワークを通さずに、サーバが要求に回答するのにかかる時間を測定できます。サーバに直接接続されているマシンで仮想ユーザを実行すると、サーバのパフォーマンスを測定できます。



▶ ミドルウェアとサーバの応答時間の測定：

ミドルウェアとその API にアクセスできる場合、サーバからミドルウェアまでの応答時間を測定できます。ミドルウェア API で仮想ユーザを作成し、ミドルウェアとサーバ間のパフォーマンスを測定できます。



仮想ユーザの動作状況の定義

仮想ユーザのタイプ、仮想ユーザの一般的なタスク、テスト目標の分析を基にして、仮想ユーザ・スクリプトを作成します。仮想ユーザは一般的なエンド・ユーザのアクションをエミュレートするので、仮想ユーザ・スクリプトは、一般的なエンド・ユーザのタスクを含んでいなければなりません。例えば、オン

ライン・バンキングのクライアントをエミュレートするには、一般的な銀行業務を行う仮想ユーザ・スクリプトを作成します。送金や残高照会をするために通常接続するページが表示されます。

テスト目標に基づいて、測定するタスクを決め、そのタスクのトランザクションを定義します。トランザクションは、仮想ユーザによって送信されるタスクに対するサーバの応答時間（エンドツーエンド時間）を測定します。例えば、銀行の Web サーバが口座の残高を提示するのにかかる応答時間を検査するには、仮想ユーザ・スクリプトでこのタスクのトランザクションを定義します。

さらに、スクリプトでランデブー・ポイントを使うことにより、ピーク時の動作状況をエミュレートできます。ランデブー・ポイントを使って、複数の仮想ユーザにまったく同時にタスクを実行させます。例えば、1つのランデブーを定義して、口座情報を同時に更新する70人のユーザをエミュレートできます。

仮想ユーザの選択

テストに使うハードウェア構成を決める前に、必要な仮想ユーザの数と種類を決めます。実行する仮想ユーザの種類と数を決めるには、テスト目標とともに一般的な使用モデルを検討します。そのための一般的なガイドラインを以下に示します。

- ▶ 1つまたは少数の GUI ユーザを使って、各種類の一般的なユーザ接続をエミュレートする。
- ▶ RTE 仮想ユーザを使って、エンド・ユーザをエミュレートする。
- ▶ 複数の非 GUI または非 RTE 仮想ユーザを実行して、ユーザの種類のそれぞれに残りの負荷を生成する。

例えば、5つの種類のユーザがあって、それぞれ別のビジネス・プロセスを実行するとします。

使用モデル	GUI	RTE	その他
ニューヨークの100人の顧客サービス (LAN 接続)	2	—	98
ヨーロッパの30人の顧客 (ダイヤルイン ISDN 接続)	2	—	28
5つのバックグラウンド・バッチ処理	—	—	5

使用モデル	GUI	RTE	その他
150 人の顧客（端末接続）	–	150	–
6 人の管理者（2 人は 486PC, 4 人は 586PC を使用）	1 (486 PC) 1 (586 PC)	–	4

テストするハードウェア / ソフトウェアの選択

ハードウェアとソフトウェアは、必要な数の仮想ユーザをエミュレートするのに十分に強力かつ高速でなければなりません。

マシンの数と適切な構成を決めるには、次のことを検討します。

- ▶ LoadRunner コントローラは別のマシンで実行することをお勧めします。
- ▶ 各 GUI 仮想ユーザには、個別に 1 台の Windows マシンが必要ですが、UNIX マシンの場合は 1 台で複数の GUI 仮想ユーザを実行できます。
- ▶ GUI 仮想ユーザのテスト・マシンの構成は、できる限り実際のユーザのマシンに合わせます。

次の表を参考にして、各 LoadRunner テスト・コンポーネントに必要なハードウェアを見積ります。この要件は、最適なパフォーマンスを得るためのものです。

注：多数のトランザクションを伴う大きなシナリオの結果ファイルには、数 MB のディスク領域が必要です。また、NFS がない場合、ロード・ジェネレータも一時ファイルを格納するために数 MB 必要とします。実行ファイルの保存の詳細については、第 10 章「シナリオの設定」を参照してください。

注：最新のインストール要件の詳細については、<http://www.mercury.com/us/products/performance-center/loadrunner/requirements.html> を参照してください。

Windows 構成要件

要件	オンライン・ モニタ付きの コントローラ	仮想ユーザ・ ジェネレータ	仮想ユーザ	アナリシス・ モジュール
コンピュータ / プロセッサ	Pentium 350 MHz 以上	Pentium 350 MHz 以上	Pentium 1 GHz 以上	Pentium 350 MHz 以上
オペレーティング・ システム	Windows NT 4.0 サービス・パッ ク 6a 以上 Windows 2000 Windows XP	Windows NT 4.0 サービス・パッ ク 6a 以上 Windows 2000 Windows XP	Windows NT 4.0 サービス・パッ ク 6a 以上 Windows 2000 Windows XP HP UX 11.x 以上, Solaris 2.6 以上, AIX 4.3.3 以上, Linux Red Hat 6.0 以上	Windows NT 4.0 サービス・パッ ク 6a 以上 Windows 2000 Windows XP
メモリ	128 MB 以上	128 MB 以上	非マルチスレ ッドの仮想ユー ザの場合は少な くとも 1 MB 以 上, マルチスレ ッドの仮想ユー ザの場合は少な くとも 512 KB 以上	128 MB 以上
スワップ領域	合計物理メモリ の 2 倍	合計物理メモリ の 2 倍	合計物理メモリ の 2 倍	合計物理メモリ の 2 倍
ハード・ディスク の空き領域	200 MB	200 MB	500 MB 以上	500 MB 以上
ブラウザ	IE 5.x 以上, Netscape Navigator 4.x, 6.x	IE 5.x 以上, Netscape Navigator 4.x, 6.x	なし	IE 5.x 以上, Netscape Navigator 4.x, 6.x

UNIX 構成要件

要件	GUI 仮想ユーザ (ユーザごと)	仮想ユーザ (ユーザ ごと)	Web 仮想ユーザ (ユーザごと)
メモリ	クライアント・ アプリケーションで 必要なメモリ + 4 ~ 5 MB	少なくとも 1.5 MB (アプリケーションの 種類による)	~ 0.5 MB
スワップ領域	合計物理メモリの 4 倍	合計物理メモリの 4 倍	合計物理メモリの 2 倍
ディスク領域	なし	なし	なし
プロセス数	4	1	1
pty 数	なし	なし	なし
1 CPU がサ ポートする ユーザ数	30 ~ 50 以上	200 ~ 300 以上	300 ~ 400 以上

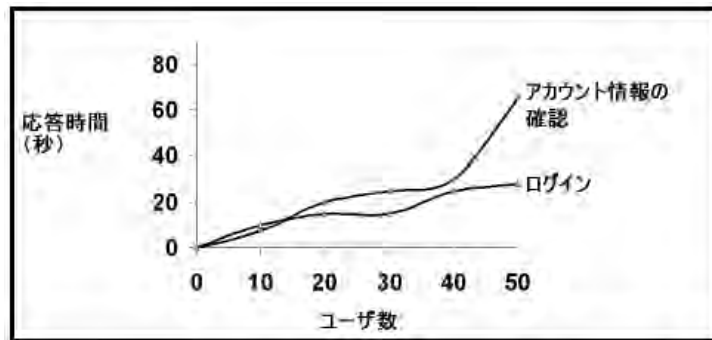
負荷テスト目標の調査

テスト計画は、明確に定義されたテスト目標に基づいていなければなりません。本項では、一般的なテスト目標の概要を説明します。

- ▶ エンド・ユーザの応答時間の測定
- ▶ 最適なハードウェア構成の定義
- ▶ 信頼性の検査
- ▶ ハードウェアまたはソフトウェア・アップグレードの検査
- ▶ 新製品の評価
- ▶ ボトルネックの特定
- ▶ システムの許容量の測定

エンド・ユーザの応答時間の測定

ユーザがビジネス・プロセスを実行し、サーバから応答を得るまでの時間を検査します。例えば、システムが通常の負荷条件で動作しているときに、エンド・ユーザは 20 秒以内にすべての要求に対する応答を受け取ることを確認するとします。次のグラフは、銀行アプリケーションの負荷と応答時間を示す例です。



最適なハードウェア構成の定義

様々なシステム構成（メモリ、CPU 速度、キャッシュ、アダプタ、モデム）がパフォーマンスにどのように影響するかを検査します。システム・アーキテクチャを理解し、アプリケーション応答時間をテストしたら、様々なシステム構成でのアプリケーション応答を測定して、どの設定が望ましいパフォーマンス・レベルを提供するかを判断できます。

例えば、3 種類のサーバを設定し、各構成で同じテストを実施して、パフォーマンスの違いを測定できます。

- ▶ 設定 1 : 200 MHz, 64 MB RAM
- ▶ 設定 2 : 200 MHz, 128 MB RAM
- ▶ 設定 3 : 266 MHz, 128 MB RAM

信頼性の検査

高い負荷または連続的な作業負荷のもとでシステムの安定度を判断します。LoadRunner を使って、システムに負荷をかけられます。システムに短い時間で広範な動作を実行させることにより、システムが通常は数週間または数か月の期間に経験するような動作状況をシミュレートします。

ハードウェアまたはソフトウェア・アップグレードの検査

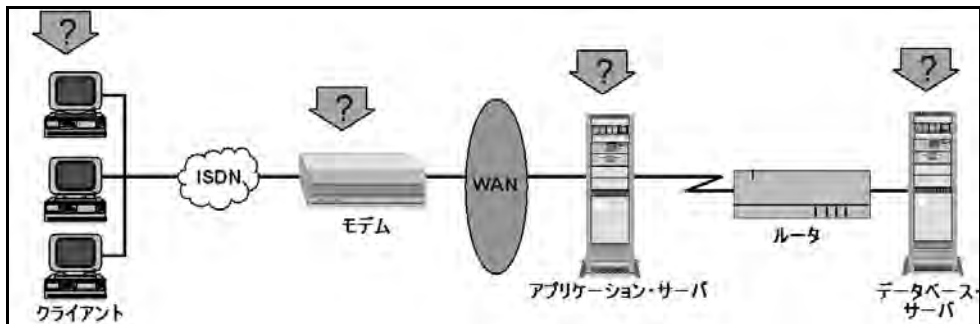
ハードウェアまたはソフトウェアの新しいリリースと以前のリリースを比較するには、回帰テストを実行します。アップグレードが応答時間にどのように影響しているか（ベンチマーク）、また信頼性にどのように影響しているか検査できます。アプリケーション回帰テストは、アップグレードの新しい機能を検査しません。新しいリリースが以前のリリースと同じ程度の効率性と信頼性を持っているかどうか検査します。

新製品の評価

製品のライフ・サイクルの企画および設計段階で個々の製品やサブシステムを評価するテストを実行できます。例えば、評価テストを基にして、サーバ・マシンのハードウェアを選択したり、データベース・パッケージを選択できます。

ボトルネックの特定

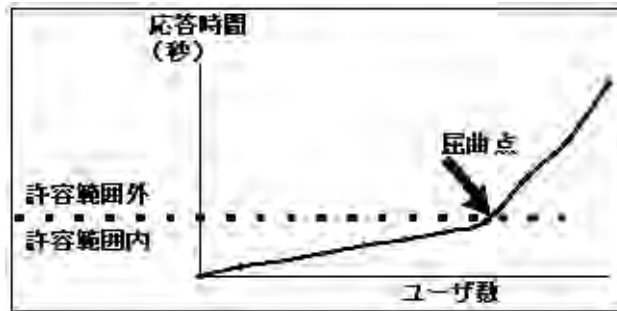
システムのボトルネックを特定し、どの構成要素がファイル・ロックング、リソース競合、ネットワーク過負荷などのパフォーマンスの低下を引き起こしているか判断するテストを実行できます。新しいネットワークおよびマシンの監視ツールと共に LoadRunner を使って、システムの様々な場所で負荷を生成し、パフォーマンスを測定します。詳細については、第6部「シナリオの監視」を参照してください。



システムの許容量の測定

システムの許容量を測定し、パフォーマンスが低下するまでの、システムの許容量を特定します。許容量を検査するには、既存のシステムでパフォーマンス

対負荷を比較し、応答時間の大幅な低下が始まる所を見つけます。これは、応答時間曲線の「屈曲点」と呼ばれています。



現在の許容量がわかれば、サポートするユーザを増やす際に、リソースを追加する必要があるかどうかわかります。

第4章

LoadRunner コントローラの概要

本章では、[LoadRunner コントローラ] ウィンドウを紹介し、基本的なシナリオ操作の方法を説明します。

本章では、次の項目について説明します。

- ▶ コントローラの起動
- ▶ LoadRunner コントローラの紹介
- ▶ シナリオ・ファイルの管理
- ▶ シナリオの実行

コントローラの起動

『Mercury LoadRunner インストール・ガイド』の手順に従って LoadRunner の環境を設定します。

コントローラを起動するには、次の手順を実行します。

コントローラは次のどちらかの方法で起動できます。

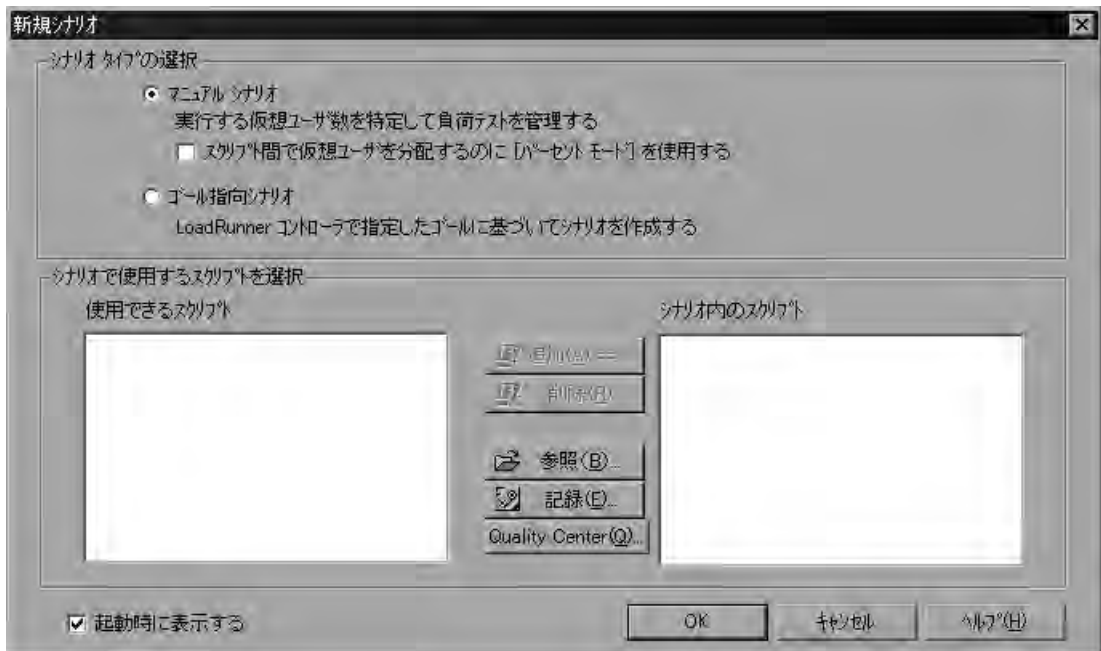


- ▶ [スタート] > [プログラム] > [Mercury LoadRunner] > [Applications] > [Controller] を選択します。
- ▶ [スタート] > [プログラム] > [Mercury LoadRunner] > [LoadRunner] を選択します。Mercury LoadRunner ランチャ画面が開きます。[負荷テスト] を選択し、[負荷テストの実行] をクリックします。

標準設定では、[Mercury LoadRunner コントローラ] ウィンドウが表示され、[新規シナリオ] ダイアログ・ボックスが表示されます。

注：[起動時に表示する] チェック・ボックスをクリアしている場合や、[表示] > ([新規シナリオ ダイアログを表示] を選択して、チェックマークをクリアした場合は、起動時に [新規シナリオ] ダイアログ・ボックスは開きません。起動時に [新規シナリオ] ダイアログ・ボックスを表示するには、[起動時に表示する] または [表示] > [新規シナリオ ダイアログを表示] を選択します (チェック・マークが表示されます)。

[新規シナリオ] ダイアログ・ボックスが起動時に開かない場合は、[ファイル] > [新規作成] を選択するか、コントローラのツールバーで [シナリオを新規作成] ボタンをクリックして開くことができます。



[**マニュアル シナリオ**] または [**ゴール指向シナリオ**] のどちらかを選択します。マニュアル・シナリオでは、実行する仮想ユーザ・グループ数を定義し、これらのグループを実行する LoadRunner のスケジュールを作成することによって、自分でシナリオを作成します。また、シナリオ内で使う仮想ユーザの総数を定義するとともに、仮想ユーザの総数に対する割合を各スクリプトに割り当てることによって、マニュアル・シナリオを作成できます。パーセント・モードを使ってシナリオを作成する場合は、[**スクリプト間で仮想ユーザを分配するのに [パーセント モード] を使用する**] を選択します。

ゴール指向シナリオでは、テストで達成するゴールを定義すると、このゴールに基づいて、LoadRunner が自動的にシナリオを作成します。

マニュアル・シナリオを作成する手順については第5章「マニュアル・シナリオの作成」を、パーセント・モードを使ってマニュアル・シナリオを作成する手順については第6章「パーセント・モードでのマニュアル・シナリオの作成」を参照してください。

ゴール指向シナリオを作成する手順については、第7章「ゴール指向シナリオの作成」を参照してください。

シナリオで使うスクリプトの選択は、次の手順で行います。

- 1 [使用できるスクリプト] リストからスクリプトを選択します。標準設定では、最近使った 50 のスクリプトがリストに表示されます。

注： 次のレジストリ・キーを変更して、[使用できるスクリプト] リストに表示するスクリプトの最大数を変更できます。

HKEY_CURRENT_USER¥Software¥Mercury Interactive¥RecentScripts¥
max_num_of_scripts

また、[**参照**] ボタンをクリックして、使用するスクリプトを探すこともできます。[使用できるスクリプト] リストに提示されたスクリプトのディレクトリ・パスを表示するには、スクリプトを右クリックして [**パスを表示**] を選択します。

Quality Center データベースに保存されているスクリプトを選択するには、[**Quality Center**] ボタンをクリックします。VuGen を使って新しいスクリプトを記録するには、[**記録**] をクリックします。

注：VB 仮想ユーザ・スクリプトを選択するには、**.usr** ファイルを探します。



2 スクリプトを選択し、**[追加]** ボタンをクリックして、[シナリオ内のスクリプト] リストにコピーします。



3 [シナリオ内のスクリプト] リストからスクリプトを削除するには、**[削除]** ボタンをクリックします。

4 次回新規シナリオを作成する際に、このダイアログ・ボックスが表示されないようにするには、**[起動時に表示する]** チェック・ボックスをクリアします。シナリオを作成中も、後からスクリプトを追加できます。

5 **[OK]** をクリックして、ダイアログ・ボックスを閉じます。

LoadRunner コントローラの紹介

[Mercury LoadRunner コントローラ] ウィンドウには、次の要素が含まれています。

- | | |
|----------|----------------------------------------------------------------------------------------------------------------------------|
| タイトル・バー | 現在作業中のシナリオの名前が表示されます。 |
| メニュー・バー | メニューが表示され、そこからコマンドを選択できます。 |
| ツールバー | コマンド選択のためのショートカットを提供します。ショートカット・ボタンをクリックすると、コマンドが実行されます。 |
| ステータス・バー | コントローラのメニュー項目に関するツール・ヒントが表示されるほか、Quality Center への接続、IP スプーファ、結果の自動照合、アナリシスの自動起動、WAN エミュレータが有効になっている場合には、それぞれのアイコンが表示されます。 |



[デザイン] タブ

[実行] タブ

[シナリオ グループ] 表示枠
(マニュアル・シナリオ)

[シナリオのスケジュール] 表示枠
(マニュアル・シナリオ)

[Mercury LoadRunner コントローラ] ウィンドウには、次の2つのビューに対応する2つのタブがあります。

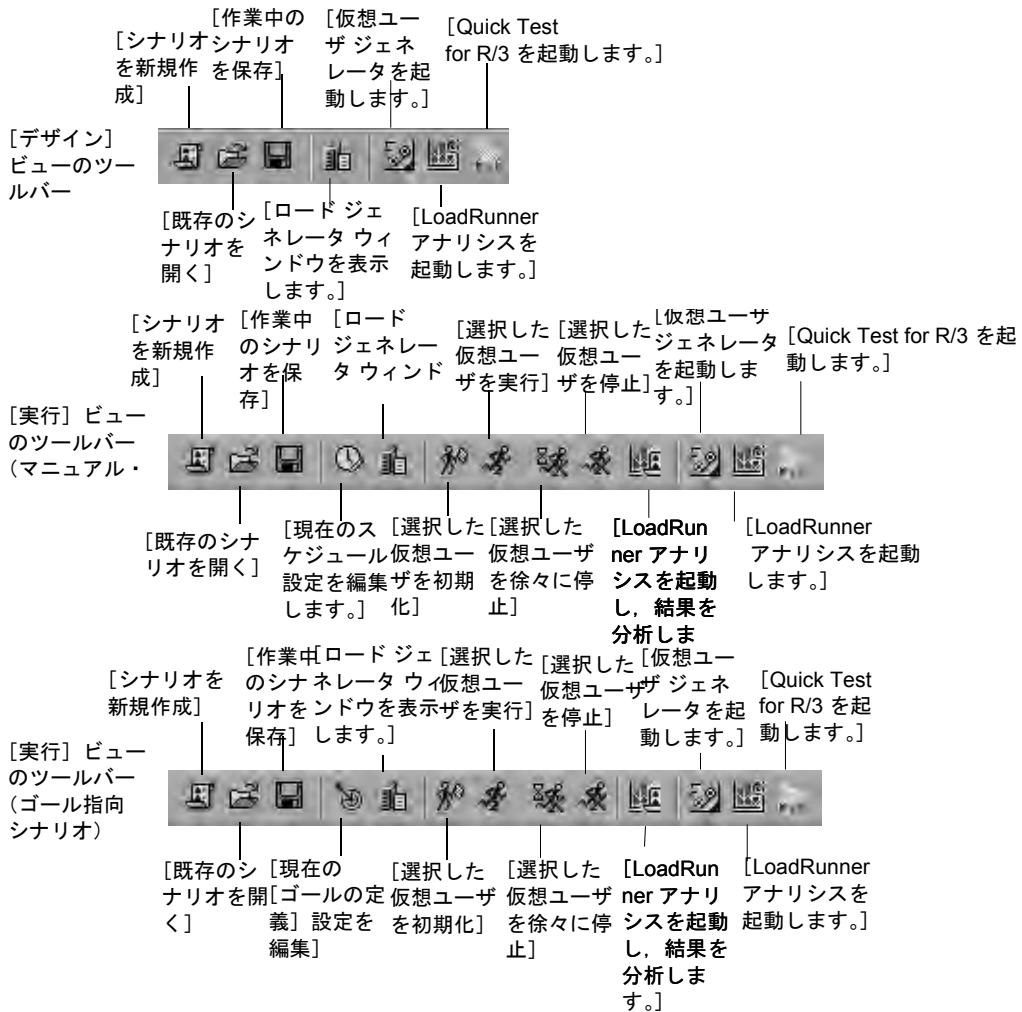
[デザイン] ビュー このビューには、シナリオ内のすべての仮想ユーザ・グループ/スクリプトのリスト、ロード・ジェネレータ・マシン、各グループ/スクリプトに割り当てられた仮想ユーザ数が表示されます。シナリオのスケジュール（マニュアル・シナリオ）またはゴール（ゴール指向シナリオ）についての基本的な情報も表示されます。

[実行] ビュー 実行中の仮想ユーザと仮想ユーザ・グループについての情報、およびオンライン・モニタ・グラフが表示されます。

さらに、**[表示] > [出力メッセージを表示]** を選択すると、シナリオ実行中に生成されたエラー、警告、通知、デバッグ、バッチ・メッセージを表示する**[出力メッセージ]** ウィンドウが表示されます。

ツールバーからのコマンドの選択

LoadRunner コントローラでツールバーのボタンをクリックして、多くの LoadRunner コマンドを実行できます。[デザイン] ビューまたは [実行] ビューのどちらを表示しているか、およびマニュアル・シナリオまたはゴール指向シナリオのどちらを作成しているかによって、ツールバーのボタンは変わります。



シナリオ・ファイルの管理

シナリオには、各負荷テスト・セッション中に発生するイベントを記述します。LoadRunner コントローラの [デザイン] ビューを使って、シナリオを作成します。

シナリオ作成後、LoadRunner は、情報をシナリオ・ファイル (.lrs) に保存します。[ファイル] メニューのコマンドを使って、シナリオ・ファイルの「作成」、「開く」、「保存」、「閉じる」ができます。これらのコマンドのいくつかは、ツールバーからも利用できます。

シナリオの新規作成

[新規作成] コマンドを使って、シナリオを新規作成します。[新規作成] コマンドを選択すると、[Mercury LoadRunner コントローラ] ウィンドウに表示されているすべての情報が消去されます。シナリオを新規作成するには、[ファイル] > [新規作成] を選択するか、コントローラのツールバーの [シナリオを新規作成] ボタンをクリックします。



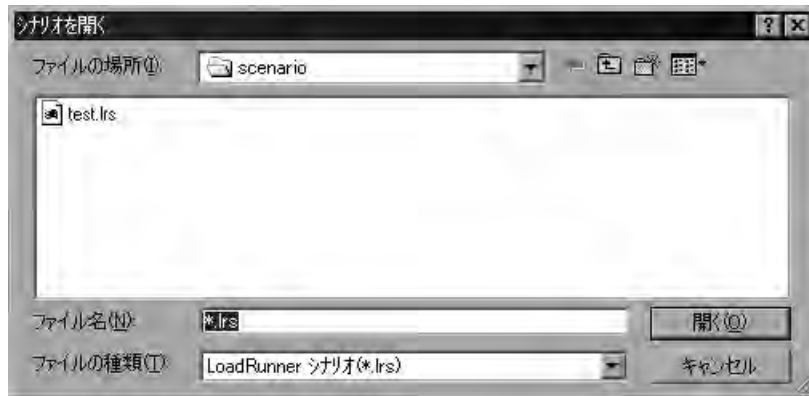
既存のシナリオを開く

[開く] コマンドで、既存のシナリオを開きます。

既存のシナリオを開くには、次の手順で行います。



- 1 [ファイル] > [開く] を選択するか、[既存のシナリオを開く] ボタンをクリックします。[シナリオを開く] ダイアログ・ボックスが表示されます。



- 2 [ファイル名] リストでファイルをクリックするか、[ファイル名] ボックスにファイル名を入力します。
- 3 [開く] をクリックします。[シナリオを開く] ダイアログ・ボックスが閉じ、シナリオが LoadRunner コントローラに表示されます。

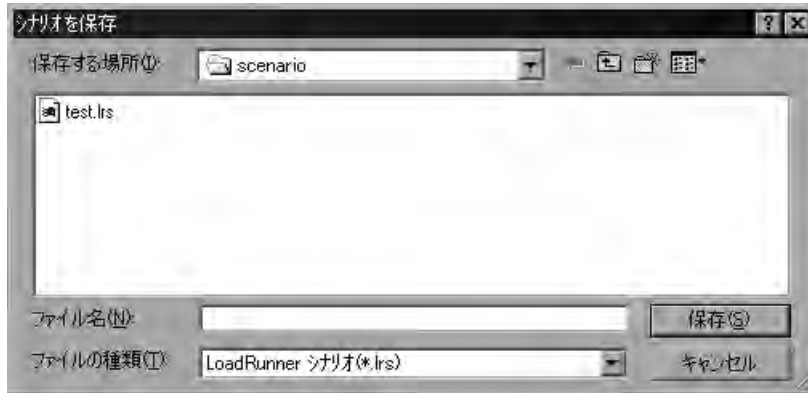
シナリオの保存

[保存] コマンドにより、現在のシナリオを保存します。

シナリオの保存は、次の手順で行います。



- 1 [ファイル] > [保存] を選択するか、[作業中のシナリオを保存] ボタンをクリックします。初めてシナリオを保存する場合は、[シナリオを保存] ダイアログ・ボックスが表示されます。



- 2 [ファイル名] テキスト・ボックスにシナリオ名を入力します。標準では、シナリオ・ファイルには拡張子 **.lrs** が付きます。
- 3 [保存] をクリックします。指定した場所にシナリオが保存されます。

シナリオを閉じる

シナリオを閉じると、コントローラのすべてのウィンドウも閉じます。シナリオを閉じるには、[ファイル] > [閉じる] を選択します。シナリオを変更すると、[<シナリオ名>への変更を保存しますか?] というメッセージが表示されます。[はい] をクリックして、変更を保存します。コントローラで開いているすべてのウィンドウとアイコンが閉じます。

シナリオの実行

シナリオのデザインが完了したら、それをすぐに実行できます。LoadRunner コントローラの [実行] ビューを使って、仮想ユーザと仮想ユーザ・グループを制御し、そのパフォーマンスをオンラインで監視できます。

The screenshot displays the LoadRunner Controller interface in the 'Execution' view. Key components include:

- シナリオマップ (Scenario Map):** A table showing the sequence of test cases and their durations.
- シナリオステータス (Scenario Status):** A window displaying real-time execution metrics.
- 実行中の仮想ユーザ - シナリオ全体 (Executing Virtual Users - Scenario Overall):** A line graph showing the number of active virtual users over time.
- トランザクション応答時間 - シナリオ全体 (Transaction Response Time - Scenario Overall):** A line graph showing the response time for transactions over time.
- Summary Table:** A table summarizing the execution results for different user groups.

項目	値
実行中の仮想ユーザ	0
経過時間	00:00:30 (時間:分:秒)
秒ごとのヒット数	85.29 (最終 60 秒)
成功したトランザクション	18
失敗したトランザクション	46
エラー	50

色	倍率	ステータス	最高	最低	平均	標準	最終
黒	1	実行中	5.000	0.000	4.077	1.730	5.000
黒	1	エラー	5.000	0.000	2.231	2.423	5.000
黒	1	実行完了	0.000	0.000	0.000	0.000	0.000

[デザイン]
タブ

[実行] タブ

[シナリオ グループ] 表
示枠

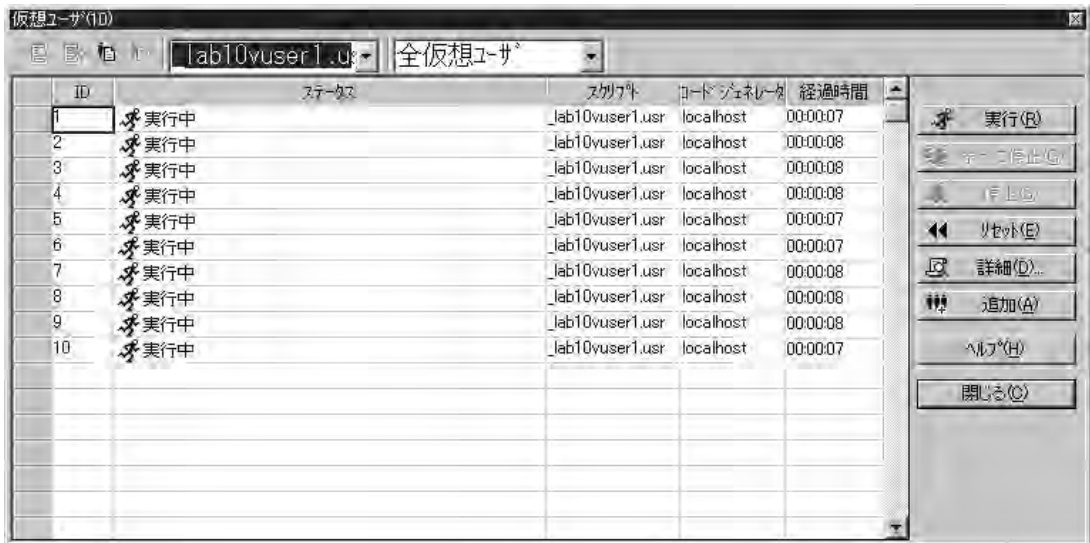
オンライン・
モニタ・グラフ

[シナリオ ステータス]
ウィンドウ

シナリオの実行中、[実行] ビューの [シナリオ グループ] 表示枠 (マニュアル・シナリオの場合) を使って、シナリオのすべての仮想ユーザおよび仮想ユーザ・グループのアクションを監視します。各仮想ユーザの [ステータス] フィールドに、仮想ユーザの現在のステータスが表示されます。

仮想ユーザ(10)

仮想ユーザ・グループを選択し、[仮想ユーザ] ボタンをクリックすると、そのグループ内の定義された個々の仮想ユーザを操作できます。グループ内の各仮想ユーザの [ID], [ステータス], [スクリプト], [ロード ジェネレータ], [経過時間] (シナリオ開始以降) を示す [仮想ユーザ] ダイアログ・ボックスが表示されます。



さらに、[実行] ビューの右上角のボックスに、実行中のシナリオの概要が表示されます。

シナリオ ステータス	
実行中の仮想ユーザ	10
経過時間	00:01:12 (時間:分:秒)
秒ごとのヒット数	4.87 (最終:60秒)
成功したトランザクション	10
失敗したトランザクション	0
エラー	0

[実行] ビューから [シナリオ ステータス] ウィンドウを切り離して、[シナリオ グループ] 表示枠を拡大できます。

シナリオの実行中に、仮想ユーザとロード・ジェネレータは、エラー、通知、警告、デバッグ、バッチ・メッセージをコントローラに送信します。これらのメッセージは [出力メッセージ] ウィンドウ ([表示] > [出力メッセージを表示]) に表示されます。

出力メッセージ (メッセージの総数: 40, エラー: 0)						
メッセージの種類	全メッセージ		詳細	閉	固定	
Type	メッセージ ID (ID)	サブアル・メッセージテキスト	メッセージ	仮想ユーザ	実行時間	エラー
警告	-26548	Action.c(35) 警告 -26548: HTML 11^...	30	10	1	1
成功	-17999	Action.c(24) The flight was booked	10	10	1	1

[出力メッセージ] ウィンドウの詳細については、242 ページ「[出力メッセージ] ウィンドウの表示」を参照してください。

オンライン・モニタおよびオンライン・モニタのグラフを使って、シナリオ実行中の仮想ユーザ・ステータス、トランザクション、システム・リソース、データベース・サーバ・リソース、Web サーバ・リソース、Web アプリケーション・サーバ・リソース、ネットワーク遅延、ストリーミング・メディア・リソース、ファイアウォール・サーバ・リソース、ERP/CRM サーバ・リソース、Java パフォーマンス、J2EE & .NET 診断、アプリケーション配備リソース、ミドルウェア・パフォーマンス、およびアプリケーション・コンポーネント・リソース、およびインフラストラクチャ・リソースを監視できます。オンライン・モニタの詳細については、第24章「オンライン監視」を参照してください。

第2部

シナリオの設計

第 5 章

マニュアル・シナリオの作成

グループを作成し、各グループに含まれるスクリプト、ロード・ジェネレータ、仮想ユーザ数を指定してマニュアル・シナリオを作成します。また、パーセント・モードを使ってマニュアル・シナリオを作成できます。パーセント・モードでは、シナリオで使う仮想ユーザの総数を定義し、各スクリプトにロード・ジェネレータと（総数に対する割合で指定して）仮想ユーザを割り当てます。

本章では、仮想ユーザ・グループ・モードを使ってマニュアル・シナリオを作成する方法を説明します。パーセント・モードでのマニュアル・シナリオの作成については、第 6 章「パーセント・モードでのマニュアル・シナリオの作成」を参照してください。

本章では、次の項目について説明します。

- ▶ シナリオの作成について
- ▶ 仮想ユーザ・グループの作成
- ▶ 仮想ユーザ・グループの仮想ユーザの設定
- ▶ 仮想ユーザ実行環境の設定
- ▶ ロード・ジェネレータの設定
- ▶ ロード・ジェネレータの追加設定
- ▶ 端末サービスの設定
- ▶ WAN エミュレーションの設定
- ▶ スクリプトの設定
- ▶ スクリプトに対する相対パスの使用

シナリオの作成について

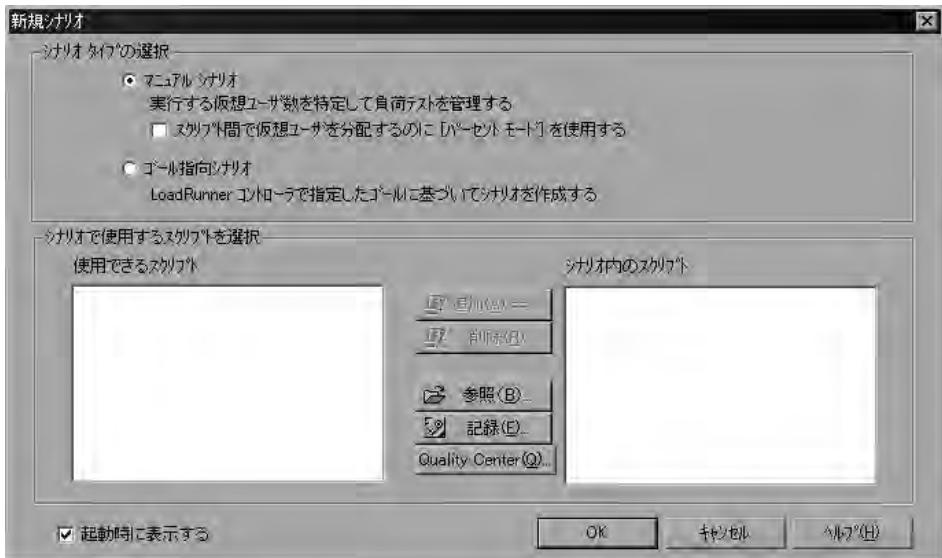
LoadRunner でシステムをテストするには、シナリオを作成する必要があります。シナリオは、テスト・セッションに関する情報が含まれているファイルです。シナリオは、現実のユーザをエミュレートするための手段です。シナリオには、実ユーザをエミュレートする方法についての情報（仮想ユーザ・グループ、仮想ユーザが実行するテスト・スクリプトおよびスクリプトを実行するロード・ジェネレータ・マシン）が含まれます。

通常のマニュアル・シナリオを作成する場合、[新規シナリオ] ダイアログ・ボックスで選択した各スクリプトが個別の仮想ユーザ・グループに割り当てられます。次に、各仮想ユーザ・グループに任意の数の仮想ユーザを割り当てます。同じグループの仮想ユーザがすべて同じロード・ジェネレータ・マシンで同じスクリプトを実行するようにも、仮想ユーザによって異なるスクリプトとロード・ジェネレータが割り当てられるようにもできます。

仮想ユーザ・グループを作成したら、シナリオのスケジュールを選択または作成します。シナリオのスケジュール作成の詳細については、第8章「シナリオのスケジュール」を参照してください。

[新規シナリオ] ダイアログ・ボックスについて

[新規シナリオ] ダイアログ・ボックスでは、新規シナリオ用のスクリプトを選択できます。



シナリオ・タイプの選択

次のシナリオ・オプションのどちらかを選択します。

- ▶ **[マニュアル シナリオ]**：マニュアル・シナリオを作成する場合はこの方式を選択します。グループを作成し、各グループに含まれるスクリプト、ロード・ジェネレータ、仮想ユーザ数を指定してマニュアル・シナリオを作成します。
- ▶ **[スクリプト間で仮想ユーザを分配するのに [パーセント モード] を使用する]**：選択した仮想ユーザ・スクリプト間で分配する仮想ユーザの数を指定してマニュアル・シナリオを作成する場合は、このオプションを選択します。
- ▶ **[ゴール指向シナリオ]**：LoadRunner が自動的にシナリオを作成するように指定する場合はこの方式を選択します。ゴール指向シナリオでは、テストで達成するゴールを定義すると、このゴールに基づいて、LoadRunner が自動的にシナリオを作成します。

スクリプトの選択

[使用できるスクリプト] リストからスクリプトを選択します。選択されたスクリプトは、[シナリオ内のスクリプト] 表示枠に表示されます。

[使用できるスクリプト]：標準設定では、最近使用された 50 のスクリプトがリストに表示されます。

注：次のレジストリ・キーを変更して、[使用できるスクリプト] リストに表示するスクリプトの最大数を変更できます。

**HKEY_CURRENT_USER\Software\Mercury
Interactive\RecentScripts\max_num_of_scripts**

[追加]：スクリプトをシナリオに追加します。

[削除]：スクリプトをシナリオから削除します。

[参照]：別のディレクトリからスクリプトを選択できます。VB 仮想ユーザ・スクリプトを選択するには、**.usr** ファイルを探します。

[記録]：スクリプトの記録を開始できるように、仮想ユーザ・ジェネレータを開きます。スクリプトの記録の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

[**Quality Center**] : [Quality Center への接続] ダイアログ・ボックスでは、Quality Center プロジェクトへの接続を開くことができます。

[**シナリオ内のスクリプト**] : シナリオで使用されるスクリプトが表示されます。

[**起動時に表示する**] : これを選択すると、LoadRunner がコントローラの起動時に [新規シナリオ] ダイアログ・ボックスを表示します。

仮想ユーザ・グループの作成

シナリオは、アプリケーションとやり取りをする実ユーザをエミュレートする仮想ユーザのグループで構成されています。シナリオを実行すると、仮想ユーザはサーバに負荷を生成し、LoadRunner はサーバとトランザクションのパフォーマンスを監視します。

仮想ユーザ・グループは、シナリオ内の仮想ユーザを管理しやすいグループにまとめるために使用されます。共通または類似の特性を備えた仮想ユーザをまとめて、仮想ユーザ・グループを作成します。例えば、同じ仮想ユーザ・スクリプトを実行する仮想ユーザをまとめた仮想ユーザ・グループなどを作成できます。

マニュアル・シナリオ・モードの [デザイン] タブについて

マニュアル・シナリオを作成するときには、[デザイン] タブに [シナリオのスケジュール] 表示枠と [シナリオグループ] 表示枠が表示されます。



[シナリオのスケジュール] 表示枠には、スケジュール・プロファイルの名前、スケジュール・モード、シナリオの継続時間、負荷の動作など、スケジュール・プロファイルに関する情報が表示されます。[負荷のプレビュー] には、定義したシナリオ・スケジュールのグラフが表示されます。スケジュールの設定の詳細については、153 ページ「シナリオのスケジュール」を参照してください。

[シナリオグループ] 表示枠には、すべての有効および無効な仮想ユーザ・グループ、各グループのパス、各グループに割り当てられた仮想ユーザの数、ロード・ジェネレータ・マシンが表示されます。

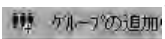
仮想ユーザ・グループまたはシナリオに対して、以下の操作が可能です。

- ▶ 仮想ユーザ・グループのグループ名、仮想ユーザの数、ロード・ジェネレータ・マシン、スクリプトの定義
- ▶ 仮想ユーザ・グループへの1つまたは複数のロード・ジェネレータ・マシンの追加、および追加したロード・ジェネレータ・マシンの設定
- ▶ 仮想ユーザ・グループへの1つまたは複数のスクリプトの追加、および設定
- ▶ シナリオの仮想ユーザ・グループの有効化または無効化
- ▶ シナリオからの仮想ユーザ・グループの削除
- ▶ 仮想ユーザ・グループまたはシナリオのスケジュール設定
- ▶ シナリオの実行
- ▶ シナリオの停止
- ▶ シナリオのリセット
- ▶ シナリオの実行結果に関する設定

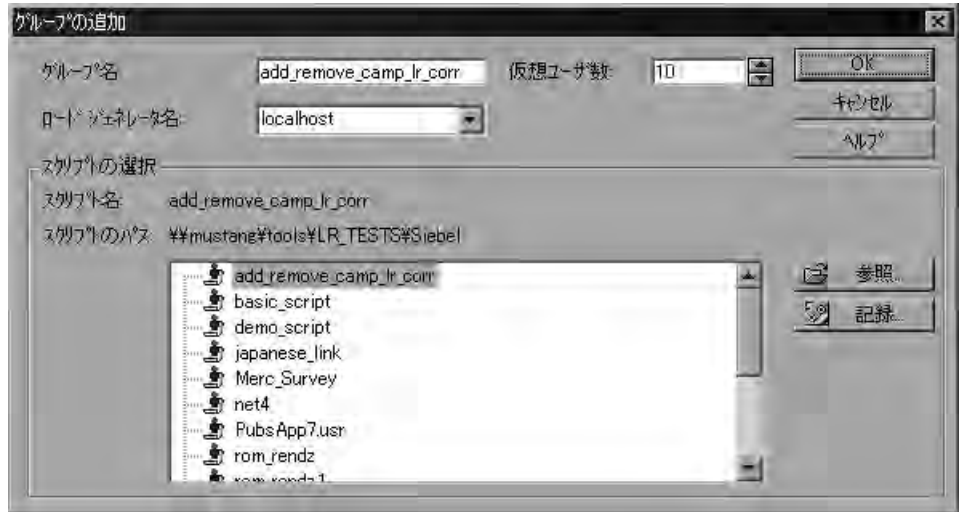
仮想ユーザ・グループの追加

[グループの追加] ダイアログ・ボックスを使用して、仮想ユーザ・グループを作成してシナリオに追加できます。

仮想ユーザ・グループを作成するには、次の手順を実行します。



- 1 [シナリオ グループ] 表示枠の右の **[グループの追加]** ボタンをクリックします。[グループの追加] ダイアログ・ボックスが表示されます。



- 2 [グループ名] ボックスに、仮想ユーザ・グループの名前を入力します。
- 3 [仮想ユーザ数] ボックスで、グループに作成する仮想ユーザ数を選択します。
- 4 [ロードジェネレータ名] リストからロード・ジェネレータを選択します。リストに表示されていないロード・ジェネレータを使用するには、[ロードジェネレータ名] リストから [**<追加 ... >**] を選択します。[新規ロードジェネレータの追加] ダイアログ・ボックスが表示されます。



[名前] ボックスにロード・ジェネレータの名前を入力します。[プラットフォーム] ボックスでは、ロード・ジェネレータを実行するプラットフォームのタイプを選択します。

標準設定では、LoadRunner はシナリオ実行中、ロード・ジェネレータの環境変数 TEMP または TMP で指定された一時ディレクトリに一時ファイルを格納し

ます。特定のロード・ジェネレータの標準設定を変更するには、[一時ディレクトリ] ボックスに格納場所を入力します。

ロード・ジェネレータをシナリオに参加できるようにするには、[ロードジェネレータのシナリオ参加を有効にする] を選択します。

[**詳細表示**] をクリックして、ダイアログ・ボックスを拡張すると [新規ロードジェネレータの追加] ダイアログ・ボックスにいくつかのタブが表示されます。各ロード・ジェネレータの設定の詳細については、79 ページ「ロード・ジェネレータの追加設定」を参照してください。

[**OK**] をクリックして、[新規ロードジェネレータの追加] ダイアログ・ボックスを閉じます。

- 5 スクリプト・リストからスクリプトを選択します。

リストに表示されていないスクリプトを使用するには、[参照] をクリックします。パスとファイル名を参照して新しいスクリプトを選択します。

- 6 [**OK**] をクリックして [グループの追加] ダイアログ・ボックスを閉じます。新しいグループのプロパティが [シナリオグループ] 表示枠に表示されます。

[グループの追加] ダイアログ・ボックスについて

[グループの追加] ダイアログ・ボックスを使用して、新しいグループをシナリオに挿入できます。

[**グループ名**] : 追加する新しいグループの名前を入力します。グループ名の長さは最大 55 文字までです。

[**仮想ユーザ数**] : グループに追加する仮想ユーザの数を選択します。

[**ロードジェネレータ名**] : 新しいグループのロード・ジェネレータ・マシンの名前を選択します。リストから既存のロード・ジェネレータを選択するか、[<追加 ... >] を選択して新しいロード・ジェネレータを作成します。[新規ロード・ジェネレータの追加] ダイアログ・ボックスが開きます。

[**スクリプトの選択**] : 現在のディレクトリ内にある使用可能なスクリプトが表示されます。このリストには、シナリオにすでに加えられているすべてのスクリプトが含まれています。

- ▶ [**スクリプト名**] : 追加する仮想ユーザ・グループで使用するスクリプトを選択します。選択したスクリプトが [スクリプト名] カラムに表示されます。
- ▶ [**スクリプトのパス**] : スクリプト・ディレクトリのパスが表示されます。

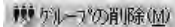
- ▶ **[参照]**：別のディレクトリにあるスクリプトのパスとファイル名を選択できます。VB 仮想ユーザ・スクリプトを使用するには、**.usr** ファイルを選択します。

注：スクリプトの場所を指定するとき、現在のシナリオ・ディレクトリを基準とする相対的な場所を指定できます。詳細については、108 ページ「スクリプトに対する相対パスの使用」を参照してください。

- ▶ **[記録]**：スクリプトの記録を開始できるように、仮想ユーザ・ジェネレータを開きます。スクリプトの記録の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド』を参照してください。

注：シナリオ実行中に、シナリオに仮想ユーザ・グループを追加して実行できます。ただし、シナリオ内のすべての仮想ユーザがランプ・アップされた後で仮想ユーザ・グループを追加した場合、その新しいグループは実行されません。

仮想ユーザ・グループの削除



仮想ユーザ・グループを削除するには、[シナリオ グループ] 表示枠の右にある **[グループの削除]** ボタンをクリックするか、削除する仮想ユーザ・グループを右クリックして **[グループの削除]** を選択します。

仮想ユーザ・グループの無効化

標準設定では、[シナリオ グループ] 表示枠に表示されるすべての仮想ユーザ・グループが有効になっており、シナリオ内で実行されます。仮想ユーザ・グループを無効にするには、仮想ユーザ・グループ名の左側にあるボックスをクリックします。これによって、グループの色がグレーに変化し、そのグループがシナリオに参加しないことを示します。仮想ユーザ・グループを再度有効にするには、同じボックスをもう一度選択します。

仮想ユーザ・グループの変更

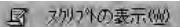
仮想ユーザ・グループのスクリプト、仮想ユーザ数、ロード・ジェネレータは、コントローラの [シナリオ グループ] 表示枠から直接、または [グループ情報] ダイアログ・ボックスを使って変更できます。

[シナリオ グループ] 表示枠から仮想ユーザ・グループを直接変更するには、次の手順を実行します。

- 1 変更する [グループ名], [スクリプトのパス], [数], または [ロード ジェネレータ] を選択します。
- 2 選択したプロパティの別の名前または数値を入力または選択します。



3 仮想ユーザ・グループのスクリプトに関する実行環境の設定を変更するには、[シナリオ グループ] 表示枠の右にある [実行環境の設定] ボタンをクリックします。実行環境の設定の詳細については、103 ページ「スクリプトの設定」を参照してください。



4 仮想ユーザ・グループのスクリプトを編集するには、[シナリオ グループ] 表示枠の右にある [スクリプトの表示] ボタンをクリックします。LoadRunner のスクリプト作成ツールである VuGen が起動します。スクリプトの編集の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド』を参照してください。

[グループ情報] ダイアログ・ボックスを使って仮想ユーザ・グループを変更するには、次の手順を実行します。



- 1 [シナリオ グループ] 表示枠の右にある [詳細] ボタンをクリックするか、変更対象のプロパティを右クリックし、[詳細] を選択します。[グループ情報] ダイアログ・ボックスが表示されます。



- 2 [グループ名] ボックスに、仮想ユーザ・グループ名を入力します。
- 3 [仮想ユーザ数] ボックスで、グループで実行する仮想ユーザ数を選択します。
- 4 [ロードジェネレータ名] リストからロード・ジェネレータを選択します。

リストに表示されていないロード・ジェネレータを使用するには、[ロードジェネレータ名] リストで [<追加 ... >] を選択し、[新規ロードジェネレー

タの追加] ダイアログ・ボックスを使って新しいロード・ジェネレータを追加します。

- 5 指定した実行環境の設定の変更を、VuGen を使ったスクリプトの記録中に行うには、[**実行環境の設定**] をクリックします。実行環境の設定の詳細については、103 ページ「スクリプトの設定」を参照してください。
- 6 仮想ユーザ・グループのスクリプトを編集するには、[**スクリプトを表示**] をクリックします。LoadRunner のスクリプト作成ツールである VuGen が起動します。スクリプトの編集についての詳細は、103 ページ「スクリプトの設定」を参照してください。
- 7 [**OK**] をクリックして、[グループ情報] ダイアログ・ボックスを閉じます。

[グループ情報] ダイアログ・ボックスについて

[グループ情報] ダイアログ・ボックスを使用して、仮想ユーザ・グループに関する詳細を表示し、グループの設定を変更できます。

[**グループ名**]：現在のグループが表示されます。この名前を変更するには、[グループ名] ボックスに新しい名前を入力します。

[**ロード ジェネレータ名**]：選択した仮想ユーザのロード・ジェネレータの名前が表示されます。別のロード・ジェネレータを指定するには、[ロード ジェネレータ名] リストからロード・ジェネレータを1つ選択します。新しいロード・ジェネレータを指定するには、[ロード ジェネレータ名] リストから [**<追加 ... >**] を選択します。

[**仮想ユーザ数**]：グループ内の仮想ユーザの数が表示されます。

[**スクリプト**]：選択したスクリプトの詳細が表示されます。

- ▶ [**名前**]：スクリプトの名前が表示されます。
- ▶ [**パス**]：スクリプト・ディレクトリのパスが表示されます。
- ▶ [**種類**]：スクリプトの種類が表示されます。
- ▶ [**スクリプトを表示**]：スクリプトを編集できるように、仮想ユーザ・ジェネレータを開きます。スクリプトの編集の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。
- ▶ [**実行環境の設定**]：[実行環境設定] ダイアログ・ボックスが表示され、以前に VuGen を使って設定した実行環境の設定を編集できます。VuGen でスクリプトの実行環境を設定していない場合、[ログ] タブと [思考遅延時間] タブを除くすべてのタブに VuGen の標準の設定が表示されます。[ログ] タブと

[思考遅延時間] タブには、コントローラの標準の設定が表示されます。実行環境の設定の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。

[更新] : コントローラの実行中にスクリプトに変更を加える場合は、このボタンをクリックして [スクリプト] を選択し、シナリオ内のスクリプトの詳細を更新します。コントローラから実行環境の設定を変更する場合、実行環境の設定を初期設定に戻すには、[実行環境の設定] を選択します。

[詳細表示 / 詳細非表示] : 次の情報を表示または非表示にします。

- ▶ [コマンドライン] : スクリプトの実行時に使用するコマンド・ライン・オプションを入力します。次に例を示します。

-x value -y value

コマンド・ライン引数の値をスクリプトに渡す方法の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。

- ▶ [ランデブー] : 選択したスクリプトについて定義されたランデブー・ポイントが表示されます。
- ▶ [仮想ユーザ] : 選択したスクリプトに関連付けられた仮想ユーザがすべて表示されます。
- ▶ [ファイル] : 選択したスクリプトによって使用されるファイルがすべて表示されます。このリストからファイルを除外するには、そのファイルの隣にあるチェック・ボックスをクリアします。

[追加] : このボタンをクリックして、[ファイルの追加] または [ディレクトリの追加] を選択します。リストに追加するファイルまたはディレクトリを探し、[開く] をクリックします。選択したファイルまたはディレクトリが、ファイルリストに追加されます。

注 : リモート・ロード・ジェネレータ・マシンで Visual C++ 仮想ユーザを実行するには、スクリプト・リストで使用されているファイルに仮想ユーザの .dll を追加する必要があります。

〔シナリオ グループ〕 表示枠の仮想ユーザ・グループの並べ替え

仮想ユーザ・グループを作成したら、グループ名、スクリプト名、グループに割り当てられている仮想ユーザ数、またはロード・ジェネレータ名でグループの並べ替えができます。

仮想ユーザ・グループを並べ替えるには、次の手順を実行します。

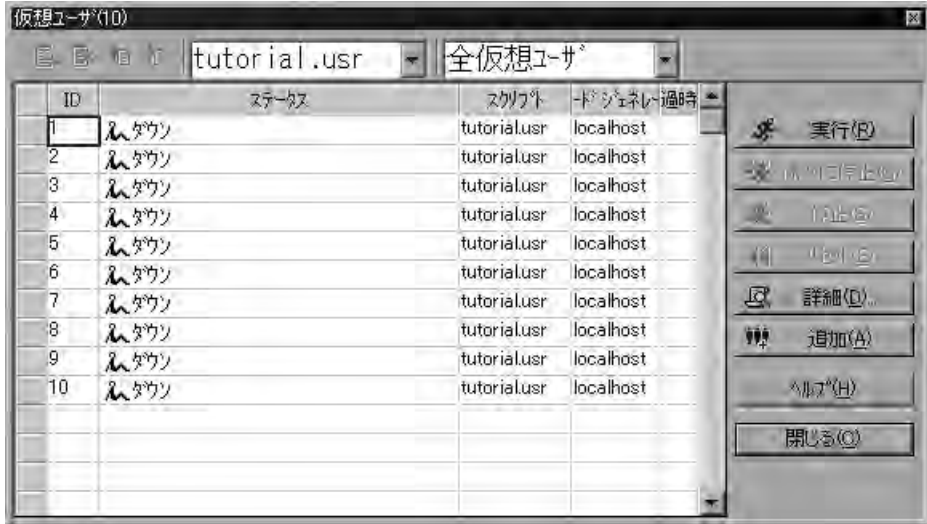
- ▶ グループの並べ替えの基準にするカラムを選択します。カラム見出しをクリックします。
- ▶ または、並べ替えの基準にするカラムの任意の場所を右クリックして、**〔グループの並べ替え〕**を選択します。〔名前で並べ替え〕、〔パスで並べ替え〕、〔数で並べ替え〕、〔ジェネレータで並べ替え〕のどれかを選択します。
- ▶ 新しい仮想ユーザ・グループのエントリをコントローラが自動的に並べ替えるようにするには、そのエントリを右クリックして、**〔自動並べ替え〕**を選択します。

仮想ユーザ・グループの仮想ユーザの設定

〔仮想ユーザ〕 ダイアログ・ボックスを使って、仮想ユーザ・グループで定義されている各仮想ユーザのプロパティを定義できます。各仮想ユーザに別々のスクリプトとロード・ジェネレータ・マシンを割り当てることができます。

各仮想ユーザのプロパティを定義するには、次の手順を実行します。

- 1 変更対象の仮想ユーザが含まれている仮想ユーザ・グループを選択し、〔シナリオ グループ〕 表示枠の右にある **〔仮想ユーザ〕** ボタンをクリックします。〔仮想ユーザ〕 ダイアログ・ボックスが表示されます。



- 2 各仮想ユーザのスクリプトを変更するには、[スクリプト] カラムで現在のものとは異なるスクリプトを選択します。または、[詳細] ボタンをクリックして、[仮想ユーザ情報] ダイアログ・ボックスのスクリプト・リストから現在のものとは異なるスクリプトを選択します。
- 3 仮想ユーザを実行するロード・ジェネレータを変更するには、[ロードジェネレータ] カラムで現在のものとは異なるロード・ジェネレータを選択します。または、[詳細] ボタンをクリックし、[仮想ユーザ情報] ダイアログ・ボックスの [ロードジェネレータ名] リストから現在のものとは異なるロード・ジェネレータを選択します。

リストに表示されていないロード・ジェネレータを使用するには、[ロードジェネレータ名] リストで [<追加 ... >] を選択し、[新規ロードジェネレータの追加] ダイアログ・ボックスを使って新しいロード・ジェネレータを追加します。

【仮想ユーザ】ダイアログ・ボックスについて

[仮想ユーザ] ダイアログ・ボックスには、グループの各仮想ユーザのステータスが表示されます。

ダイアログ・ボックスの最上部にあるリストから、シナリオ・グループを選択します。



[選択した仮想ユーザを表示]：選択した各仮想ユーザについて実行時ビューアを開きます。



[選択した仮想ユーザを非表示]：開いている実行時ビューアを閉じます。



[仮想ユーザ ログを表示]：仮想ユーザについての実行時の情報を表示します。この情報は、標準設定では 1000 ミリ秒ごとに更新されます。



[仮想ユーザ ログを非表示]：仮想ユーザ・ログを閉じます。

[ID]：仮想ユーザの ID 番号が表示されます。

[ステータス]：仮想ユーザのステータスが表示されます。次のようなステータスがあります。

ステータス	説明
ダウン	仮想ユーザはダウンしている。
保留中	仮想ユーザは、初期化できる状態にあり、ロード・ジェネレータが利用可能になるのを待機中か、ロード・ジェネレータにファイルを転送中。仮想ユーザは、スケジュール属性で設定されている条件が満たされると実行される。
初期化	仮想ユーザは、リモート・マシン上で初期化中。
準備完了	仮想ユーザはスクリプトの init セクションを実行済みで実行可能な状態。
実行中	仮想ユーザは実行中。仮想ユーザ・スクリプトがロード・ジェネレータで実行されている。
ランデブー	仮想ユーザはランデブーに到着し、LoadRunner によって解放されるのを待っている。
完了－成功	仮想ユーザは実行を終了した。スクリプトの実行は成功した。
完了－失敗	仮想ユーザは実行を終了した。スクリプトの実行は失敗した。
エラー	仮想ユーザにエラーが発生した。エラーの詳細については、[仮想ユーザ] ダイアログ・ボックスの [ステータス] フィールドか、エラーの詳細が表示される [出力メッセージ] ウィンドウを参照してください。
徐々に終了	仮想ユーザは、実行中の反復またはアクションを終了してから停止する（[ツール] > [オプション] > [実行環境の設定] で定義）。

ステータス	説明
終了中	仮想ユーザは実行を終えたか、停止されて終了中。
中止	仮想ユーザは [停止] コマンドによって実行を中止された。

[**スクリプト**] : 仮想ユーザによって実行されるスクリプトが表示されます。

[**ロード ジェネレータ**] : 仮想ユーザを実行するロード・ジェネレータ・マシンが表示されます。

[**経過時間**] : 仮想ユーザの実行が開始されてから経過した時間が表示されます。

[**実行**] : コントローラに対して、仮想ユーザの実行を開始するように指示します。

[**徐々に停止**] : コントローラに対して、現在の反復またはアクションを完了してから仮想ユーザの実行を停止するように指示します。このオプションを使用できるのは、[オプション] ダイアログ・ボックスの [実行環境の設定] タブで [停止前に現在の反復終了まで待機する] または [停止前に現在のアクション終了まで待機する] オプションを選択しており、仮想ユーザが「実行」ステータスになっているときのみです。

[**停止**] : コントローラに対して、仮想ユーザの実行を停止するように指示します。

[**リセット**] : 仮想ユーザのステータスを [ダウン] にリセットします。

[**詳細**] : [仮想ユーザ情報] ダイアログ・ボックスを開きます。

[**追加**] : 1 つまたは複数の仮想ユーザを追加できるよう [仮想ユーザの追加] ダイアログ・ボックスを開きます。

このほかに、右クリック・メニューで次の項目を選択できます。

- ▶ [**番号の再割り当て**] : グループ内の仮想ユーザの番号を再割り当てし、各仮想ユーザの ID 番号を変更します。
- ▶ [**実行環境の設定**] : [実行環境設定] ダイアログ・ボックスが表示され、以前に VuGen を使って設定した実行環境の設定を編集できます。VuGen でスクリプトの実行環境を設定していない場合、[ログ] タブと [思考遅延時間] タブを除くすべてのタブに VuGen の標準の設定が表示されます。[ログ] タブと [思考遅延時間] タブには、コントローラの標準の設定が表示されます。実行環境の設定の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。

注：1つの仮想ユーザの実行環境の設定を変更すると、グループ内のすべての仮想ユーザの実行環境の設定が変わります。

- ▶ **[スクリプトの表示]**：スクリプトを編集できるように、仮想ユーザ・ジェネレータを開きます。スクリプトの編集の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。
 - ▶ **[仮想ユーザの初期化]**：仮想ユーザを指定されたロード・ジェネレータに割り振り、スクリプトを実行できるようにします。仮想ユーザが初期化に失敗すると、その仮想ユーザのステータスは**[エラー]**に変わります。
 - ▶ **[一時停止]**：仮想ユーザのスクリプトの実行を一時的に停止します。
-

注：仮想ユーザ・グループを一時停止すると、そのトランザクション応答時間に影響します。

- ▶ **[仮想ユーザを表示]**：実行時ビューアを開き、スクリプトを実行している仮想ユーザを表示します。
- ▶ **[仮想ユーザを非表示]**：割り当てられたスクリプトを実行している仮想ユーザを表示している実行時ビューアを閉じます。
- ▶ **[仮想ユーザ ログを表示]**：仮想ユーザについての実行時の情報を表示します。この情報は、標準設定では1000ミリ秒ごとに更新されます。
- ▶ **[仮想ユーザ ログを非表示]**：仮想ユーザ・スクリプトのログを閉じます。
- ▶ **[仮想ユーザのフィルタ]**：[仮想ユーザ] ダイアログ・ボックスに表示される仮想ユーザを、ステータスに基づいてフィルタします。[仮想ユーザ] ダイアログ・ボックス上部のプルダウン・リストでフィルタ・オプションを選択することもできます。
- ▶ **[仮想ユーザの並べ替え]**：グループ内の仮想ユーザを、ID、ステータス、スクリプト、ロード・ジェネレータ、または経過時間の順に並べ替えます。

【仮想ユーザ情報】 ダイアログ・ボックスについて

【仮想ユーザ情報】 ダイアログ・ボックスには、グループ内の特定の仮想ユーザに関する詳細が表示され、その仮想ユーザのロード・ジェネレータとスクリプトの設定を変更できます。

【**グループ名**】：選択した仮想ユーザが属しているグループの名前が表示されます。

【**仮想ユーザ名**】：選択した仮想ユーザの名前が表示されます。

【**ロード ジェネレータ名**】：選択した仮想ユーザのロード・ジェネレータの名前が表示されます。別のロード・ジェネレータを指定するには、【ロード ジェネレータ名】 リストからロード・ジェネレータを1つ選択します。新しいロード・ジェネレータを指定するには、【<追加 ... >】を選択します。

【**スクリプトの選択**】：現在のディレクトリ内にある使用可能なスクリプトが表示されます。

- ▶ 【**スクリプト名**】：この仮想ユーザで使用するスクリプトを選択します。選択したスクリプトが【スクリプト名】 カラムに表示されます。
- ▶ 【**スクリプトのパス**】：スクリプト・ディレクトリのパスが表示されます。
- ▶ 【**参照**】：別のディレクトリからスクリプトを選択できます。VB 仮想ユーザ・スクリプトを選択するには、**.usr** ファイルを探します。
- ▶ 【**記録**】：VuGen を開いて、スクリプトの記録を開始できるようにします。スクリプトの記録の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。
- ▶ 【**実行環境の設定**】：【実行環境設定】 ダイアログ・ボックスが表示され、以前に VuGen を使って設定した実行環境の設定を編集できます。VuGen でスクリプトの実行環境を設定していない場合、【ログ】 タブと【思考遅延時間】 タブを除くすべてのタブに VuGen の標準の設定が表示されます。【ログ】 タブと【思考遅延時間】 タブには、コントローラの標準の設定が表示されます。実行環境の設定の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。
- ▶ 【**パラメータ リスト**】：VuGen でパラメータ・リストを開きます。パラメータ・リストの詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

仮想ユーザ・グループへの仮想ユーザの追加

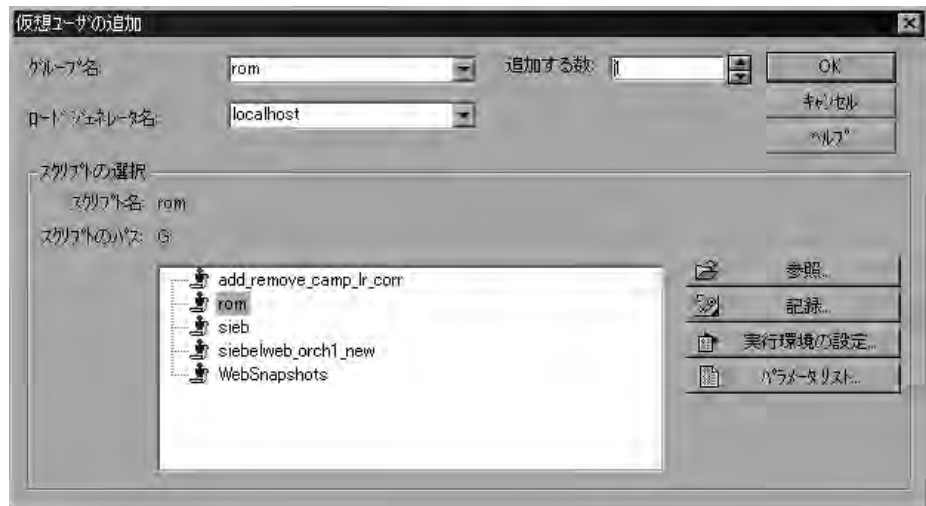
仮想ユーザ・グループに仮想ユーザを追加し、そのプロパティを定義するには、[仮想ユーザの追加] ダイアログ・ボックスを使います。

注：[仮想ユーザの実行/停止] ダイアログ・ボックスを使用して、シナリオ実行中に仮想ユーザを追加して実行できます。詳細については、232 ページ「実行中のシナリオへの手作業による仮想ユーザの追加」を参照してください。

仮想ユーザ・グループへ仮想ユーザを追加するには、次の手順を実行します。



- 1 [仮想ユーザ] ダイアログ・ボックスで、[追加] ボタンをクリックします。
[仮想ユーザの追加] ダイアログ・ボックスが表示されます。



- 2 [グループ名] ボックスで、仮想ユーザ・グループ名を選択します。
- 3 [追加する数] ボックスで、グループに追加する仮想ユーザ数を選択します。
- 4 [ロード ジェネレータ名] リストからロード・ジェネレータを選択します。

リストに表示されていないロード・ジェネレータを使用するには、[ロード ジェネレータ名] リストで [<追加 ... >] を選択し、[新規ロード ジェネレータの追加] ダイアログ・ボックスを使って新しいロード・ジェネレータを追加します。

- 5 スクリプト・リストからスクリプトを選択します。

リストに表示されていないスクリプトを使用するには、[参照] ボタンをクリックします。パスとファイル名を参照して新しいスクリプトを選択します。

- 6 [OK] をクリックして、[仮想ユーザの追加] ダイアログ・ボックスを閉じます。新しい仮想ユーザのプロパティが [仮想ユーザ] ダイアログ・ボックスに表示されます。

[仮想ユーザの追加] ダイアログ・ボックスについて

このダイアログ・ボックスでは、新しい仮想ユーザを仮想ユーザ・グループに追加します。

[グループ名]：仮想ユーザを追加する対象となるグループの名前を入力します。

[ロード ジェネレータ名]：新しい仮想ユーザのロード・ジェネレータ・マシンの名前を選択します。リストから既存のロード・ジェネレータを選択するか、[<追加 ... >] を選択して新しいロード・ジェネレータを作成します。[新規ロード ジェネレータの追加] ダイアログ・ボックスが開きます。

[追加する数]：グループに追加する仮想ユーザの数を選択します。

[スクリプトの選択]：現在のディレクトリ内にある使用可能なスクリプトが表示されます。

- ▶ [スクリプト名]：この仮想ユーザで使用するスクリプトを選択します。選択したスクリプトが [スクリプト名] カラムに表示されます。
- ▶ [スクリプトのパス]：スクリプト・ディレクトリのパスが表示されます。
- ▶ [参照]：別のディレクトリからスクリプトを選択できます。VB 仮想ユーザ・スクリプトを使用するには、**.usr** ファイルを選択します。

注：スクリプトの場所を指定するとき、現在のシナリオ・ディレクトリを基準とする相対的な場所を指定できます。詳細については、108 ページ「スクリプトに対する相対パスの使用」を参照してください。

- ▶ [記録]：VuGen を開いて、スクリプトの記録を開始できるようにします。スクリプトの記録の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド』を参照してください。

- ▶ **[実行環境の設定]** : [実行環境設定] ダイアログ・ボックスが表示され、以前に VuGen を使って設定した実行環境の設定を編集できます。VuGen でスクリプトの実行環境を設定していない場合、[ログ] タブと [思考遅延時間] タブを除くすべてのタブに VuGen の標準の設定が表示されます。[ログ] タブと [思考遅延時間] タブには、コントローラの標準の設定が表示されます。実行環境の設定の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。

注：新しい仮想ユーザの実行環境の設定を変更すると、グループ内のすべての仮想ユーザの実行環境の設定が変わります。実行環境の設定の詳細については、103 ページ「スクリプトの設定」を参照してください。

- ▶ **[パラメータ リスト]** : VuGen でパラメータ・リストを開きます。パラメータ・リストの詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。

仮想ユーザ実行環境の設定

スクリプトの実行環境を設定することによって、コントローラによる仮想ユーザ・スクリプトの実行方法をカスタマイズできます。

コントローラで実行環境の設定を表示するには、次の手順を実行します。

コントローラの [シナリオ グループ] 表示枠で、1 つまたは複数のグループを選択して強調表示し、**[実行環境の設定]** をクリックします。

[実行環境設定] ダイアログ・ボックスに、以前に VuGen を使って設定した設定値が表示されます。VuGen でスクリプトの実行環境が設定されていない場合、[ログ] タブおよび [思考遅延時間] タブを除くすべてのタブに VuGen の標準の設定が表示されます。[ログ] タブと [思考遅延時間] タブには、コントローラの標準の設定が表示されます。

注：Web および Java などのいくつかのプロトコルには、固有の設定があります。

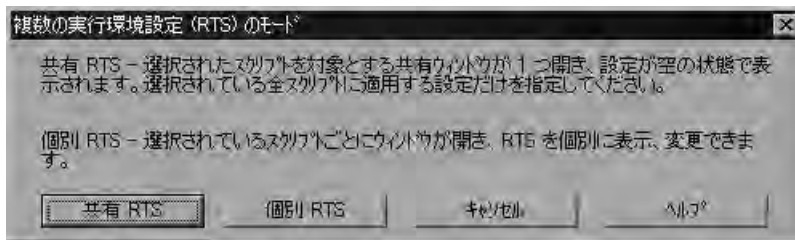
各実行環境の設定の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド』を参照してください。

新しい仮想ユーザの実行環境の設定を変更すると、グループ内のすべての仮想ユーザの実行環境の設定が変わります。1つのグループに複数の仮想ユーザのタイプが含まれる場合には、次の72ページ「複数のスクリプトの実行環境設定の変更」の記述に従って、共有実行環境の設定を変更できます。

注：コントローラで実行環境の設定を変更すると、LoadRunner は変更された設定を使ってスクリプトを実行します。初期設定に戻すには、[更新] ボタンをクリックし、[実行環境の設定] を選択します。

複数のスクリプトの実行環境設定の変更

スクリプトの実行環境の設定を変更するために、複数のスクリプトまたは複数のスクリプトを含むグループを選択すると、コントローラに共有実行環境設定を変更するオプションが表示されます。



注：選択したスクリプトの1つが共有実行環境の設定をサポートしない場合、各スクリプトの実行環境設定を個別に変更するオプションのみを使用できます。共有 RTS（実行環境設定）モードは、GUI または Astra LoadTest 仮想ユーザには使用できません。

複数のスクリプトの実行環境設定を変更する方法を選択します。

[共有 RTS]：すべての実行環境の設定を未設定の状態ですべてのウィンドウが表示されます。このモードでは、選択したすべてのスクリプトで変更するオプションだけを設定します。設定しなかった実行環境の設定は変更されません。

[個別 RTS]：選択したスクリプトごとに個別のウィンドウが表示されます。このモードでは、各スクリプトの設定を個別に変更します。

共有実行環境設定の変更

共有モードで変更した設定は、選択したすべてのスクリプトに適用されます。変更しなかった設定は変わりません。例えば、ダイアログ・ボックスにチェック・ボックスがあり、これらが選択できない状態になっている場合は、これらは選択もクリアもされていないことを意味します。チェック・ボックスを選択またはクリアすると、その変更が選択したすべてのスクリプトに適用されます。

実行環境の設定によっては共有モードで変更できないものがあります。これらの設定は表示されません。これらを変更するには、対応する個々のスクリプトの [実行環境設定] ダイアログ・ボックスを表示します。

[実行環境設定] ダイアログ・ボックスのボタン（例えば [ブラウザのエミュレーション] ノードの **[変更]** および **[詳細]** ボタン）はすべて無効になります。

次のノードは共有モードでは表示されません。

- ▶ **[Java Environment Settings : Classpath]** ノード
- ▶ **[インターネットプロトコル : 内容チェック]** ノード
- ▶ **[実行論理]** ノード – **[実行論理]** ノードをサポートするプロトコルでは、**[ペースの設定]** ノードに **[新規反復を開始するタイミング]** ボックスが表示されます。
- ▶ Citrix ICA, Oracle NCA, および WAP プロトコルで、「**Property**」カラムと「**Value**」カラムで構成されるテーブルが含まれるノード（例えば、**[Oracle NCA:Client Emulation]** ノード）。

ロード・ジェネレータの設定

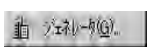
ロード・ジェネレータの属性の設定は、[ロード ジェネレータ] ダイアログ・ボックスを使って、ロード・ジェネレータ・リストにロード・ジェネレータを追加しているときに設定できます。また既存のロード・ジェネレータの属性は随時変更できます。

シナリオに参加しているすべてのロード・ジェネレータのグローバル設定を指定するには、LoadRunner の [オプション] ダイアログ・ボックスを使います。詳細については、第10章「シナリオの設定」を参照してください。各ロード・ジェネレータの特定のプロパティを設定するには、次に説明する [ロード ジェネレータ] ダイアログ・ボックスを使います。

また、シナリオの仮想ユーザを実行するロード・ジェネレータの指定もできます。例えば、あるロード・ジェネレータが特定のシナリオの実行に使えない場合、そのロード・ジェネレータをロード・ジェネレータ・リストから完全に削除するのではなく、一時的に除外することができます。

シナリオに参加させるロード・ジェネレータを選択するには、[有効にする] および [無効にする] コマンドを使用します。ロード・ジェネレータを無効にすると、そのロード・ジェネレータは一時的にリストから除外されます。ロード・ジェネレータを有効にすると再びリストに加えられます。ロード・ジェネレータの無効化は、特定のマシンを隔離してそのパフォーマンスをテストするのに特に便利です。

ロード・ジェネレータを設定するには、次の手順を実行します。

-  1 [ジェネレータ] ボタンをクリックするか、[シナリオ] > [ロード ジェネレータ] を選択します。[ロード ジェネレータ] ダイアログ・ボックスが開きます。ロード・ジェネレータの [名前] と、その [ステータス]、[プラットフォーム] および [詳細] が表示されます。



- 2 [接続] をクリックすると、ロード・ジェネレータの [ステータス] が「**ダウン**」から「**準備完了**」に変わります。ロード・ジェネレータが接続されると、このボタンは自動的に [切断] に変わります。ロード・ジェネレータの [ステータス] を「**準備完了**」から「**ダウン**」に変えるには、[切断] をクリックします。
- 3 ロード・ジェネレータを無効にするには、無効にするロード・ジェネレータを選択し、[無効にする] をクリックします。ロード・ジェネレータの名前が青からグレーに変わり、そのロード・ジェネレータが無効になります。ロード・ジェネレータを有効にするには、有効にするロード・ジェネレータを選択し、[有効にする] をクリックします。ロード・ジェネレータの名前がグレーから青に変わり、そのロード・ジェネレータが有効になります。
- 4 ロード・ジェネレータの詳細を表示するには、対象のロード・ジェネレータを選択し、[詳細] をクリックします。[ロードジェネレータの情報] ダイアログ・ボックスが開き、選択したロード・ジェネレータの情報が表示されます。
- 5 ロード・ジェネレータを削除するには、リストから選択し、[削除] をクリックします。

[ロード ジェネレータ] ダイアログ・ボックスについて

[ロード ジェネレータ] ダイアログ・ボックスには、シナリオに接続しているロード・ジェネレータの情報が表示されます。

[名前]：ロード・ジェネレータの名前を一覧表示します。

[ステータス]：ロード・ジェネレータのステータスが表示されます。次の表に、ロード・ジェネレータのステータスの説明を示します。

ステータス	説明
準備完了	ロード・ジェネレータは接続されている
接続中	ロード・ジェネレータは接続を行っている最中
アクティブ	ロード・ジェネレータは仮想ユーザを実行中
ダウン	ロード・ジェネレータは接続されていない
失敗	ロード・ジェネレータとの接続が確立できなかった

[プラットフォーム]：ロード・ジェネレータが実行されているプラットフォームのタイプが表示されます。

[詳細]：接続が失敗した場合に、失敗の詳しい原因が表示されます。

[接続]：コントローラに対して、シナリオを実行するロード・ジェネレータに接続するように指示します。ロード・ジェネレータが接続されると、このボタンは自動的に[切断]に変わります。

[追加]：[新規ロード ジェネレータの追加] ダイアログ・ボックスを開きます。

[削除]：ロード・ジェネレータを削除します。削除できるのは、切断されているロード・ジェネレータだけです。

[リセット]：失敗した接続のリセットを試みます。

[詳細]：[ロード ジェネレータの情報] ダイアログ・ボックスを開きます。

[無効にする / 有効にする]：コントローラに対して、ロード・ジェネレータを有効または無効にするように指示します。ロード・ジェネレータを無効にすると、その名前、ステータス、プラットフォーム、および詳細はグレーで表示されます。

注：コントローラは、Windows ロード・ジェネレータ・マシンの CPU 使用率を監視し、ロード・ジェネレータが過負荷になった場合には、仮想ユーザのロードを自動的に停止します。[ロード ジェネレータ] ダイアログ・ボックス内のアイコンを使用して、マシンの CPU 使用率のステータスを監視できます。ロード・ジェネレータの CPU 使用率に問題が生じると、ロード・ジェネレータ名の左側にあるアイコンに黄色の線が表示されます。マシンが過負荷になると、アイコンに赤色の線が表示されます。

ロード・ジェネレータの追加

シナリオにロード・ジェネレータ・マシンを追加したり、既存のロード・ジェネレータの情報を変更することができます。

ロード・ジェネレータを追加またはロード・ジェネレータ情報を変更するには、次の手順を実行します。

- 1 [ロード ジェネレータ] ダイアログ・ボックスで、**[追加]** をクリックします。
[新規ロード ジェネレータの追加] ダイアログ・ボックスが表示されます。



- 2 **[名前]** ボックスにロード・ジェネレータの名前を入力します。
- 3 **[プラットフォーム]** ボックスで、ロード・ジェネレータを実行するプラットフォームのタイプを選択します。
- 4 **[一時ディレクトリ]** ボックスに、一時ファイルを保存する場所としてロード・ジェネレータ上の場所を入力するか、標準設定の場所を使用するためにボックスを空白のままにします。標準設定では、LoadRunner はシナリオ実行中、ロード・ジェネレータの環境変数 TEMP または TMP で指定された一時ディレクトリに一時ファイルを格納します。

- 5 ロード・ジェネレータをシナリオに参加できるようにするには、**[ロード ジェネレータのシナリオ参加を有効にする]** を選択します。
- 6 **[詳細表示]** をクリックして、ダイアログ・ボックスを拡張すると**[新規ロード ジェネレータの追加]** ダイアログ・ボックスにいくつかのタブが表示されます。これらの設定の詳細については、79 ページ「ロード・ジェネレータの追加設定」を参照してください。
- 7 **[OK]** をクリックして、**[新規ロード ジェネレータの追加]** ダイアログ・ボックスを閉じます。入力したロード・ジェネレータ名がロード・ジェネレータ・リストに表示されます。ステータスは「**ダウン**」に設定されています。

[新規ロード ジェネレータの追加] ダイアログ・ボックスについて

[新規ロード ジェネレータの追加] ダイアログ・ボックスを使って、シナリオにロード・ジェネレータ・マシンを追加できます。

[名前] : **[名前]** ボックスには、追加するロード・ジェネレータの名前を入力します。

[プラットフォーム] : ロード・ジェネレータを実行するプラットフォームのタイプを選択します。

[一時ディレクトリ] : 一時ファイルを保存する場所として、ロード・ジェネレータ上の場所を入力します。標準設定では、LoadRunner はシナリオ実行中、ロード・ジェネレータの環境変数 TEMP または TMP で指定された一時ディレクトリに一時ファイルを格納します。

[ロード ジェネレータのシナリオ参加を有効にする] : このロード・ジェネレータをシナリオに含める場合に選択します。

[詳細表示 / 詳細非表示] : **[ステータス]**, **[実行ファイルの保存場所]**, **[UNIX 環境]**, **[実行時クォータ]**, **[仮想ユーザ ステータス]**, **[仮想ユーザの制限]**, **[WAN エミュレーション]**, **[端末サービス]**, **[ファイアウォール]**, そして **[接続ログ]** (エキスパート・モード) のタブを表示 / 非表示にします。

ロード・ジェネレータの追加設定

[新規ロード ジェネレータの追加] または [ロード ジェネレータの情報] ダイアログ・ボックスのタブを使って、各ロード・ジェネレータを追加設定できます。設定できるのは、ステータス、実行ファイルの保存場所、UNIX 環境、実行時クォータ、仮想ユーザの制限、仮想ユーザのステータス、端末サービス、ファイアウォール、そして WAN エミュレーションです。エキスパート・モードでは、接続ログの設定もできます。[接続ログ] タブの詳細については、付録 C「エキスパート・モードでの作業」を参照してください。

シナリオに参加しているすべてのロード・ジェネレータを対象とするグローバルな設定を行うには、[オプション] ダイアログ・ボックスを使います。詳細については、第 10 章「シナリオの設定」を参照してください。

ロード・ジェネレータを設定するには、次の手順を実行します。

- 1 [ジェネレータ] ボタンをクリックするか、[シナリオ] > [ロード ジェネレータ] を選択して [ロード ジェネレータ] ダイアログ・ボックスを開き、[詳細] ボタンをクリックします。[ロード ジェネレータの情報] ダイアログ・ボックスが開きます。

[新規ロード ジェネレータの追加] ダイアログ・ボックスを使って設定することもできます。[詳細表示] をクリックして、設定タブを表示します。

- 2 タブを選択し、ロード・ジェネレータの設定を行います。

これらのタブで行った設定は、[名前] ボックスで指定したロード・ジェネレータだけに適用されます。それ以外のロード・ジェネレータを設定するには、そのロード・ジェネレータの名前とプラットフォームをそれぞれのボックスで指定するか、[ロード ジェネレータ] ダイアログ・ボックスでそのロード・ジェネレータを選択します。

- 3 [OK] をクリックして、[ロード ジェネレータの情報] または [新規ロード ジェネレータの追加] ダイアログ・ボックスを閉じると、設定が保存されます。

[ロード ジェネレータの情報] ダイアログ・ボックスについて

[ロード ジェネレータの情報] ダイアログ・ボックスでは、ロード・ジェネレータ・マシンをシナリオに追加できます。

[名前] : [名前] ボックスには、追加するロード・ジェネレータの名前を入力します。

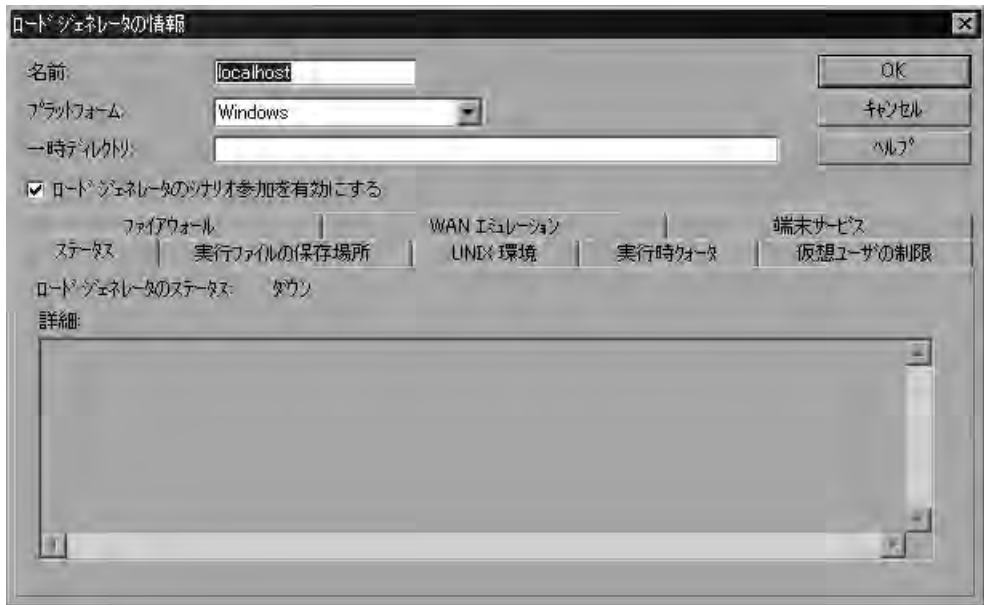
[プラットフォーム]：ロード・ジェネレータを実行するプラットフォームのタイプを選択します。

[一時ディレクトリ]：一時ファイルを保存する場所として、ロード・ジェネレータ上の場所を入力します。

[ロード ジェネレータのシナリオ参加を有効にする]：このロード・ジェネレータをシナリオに含める場合に選択します。

[ステータス] タブ

[ロード ジェネレータの情報] ダイアログ・ボックスで、[ステータス] タブを選択して、ロード ジェネレータのステータスの詳細を表示します。

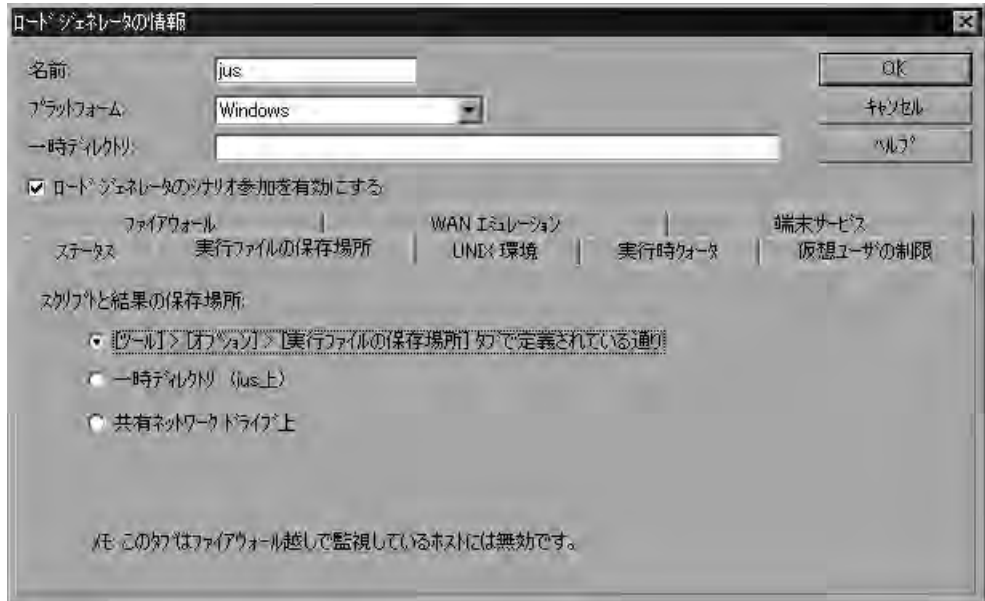


[ロード ジェネレータのステータス]：ロード・ジェネレータのステータスが表示されます。

[詳細]：選択したロード・ジェネレータに関するエラーと他の実行時の情報が表示されます。

【実行ファイルの保存場所】タブ

【ロード ジェネレータの情報】ダイアログ・ボックスで、シナリオ実行中に LoadRunner が各ロード・ジェネレータから集めるパフォーマンス・データを保存する結果ディレクトリを指定するには、【**実行ファイルの保存場所**】タブを選択します。



【スクリプトと結果の保存場所】：指定した数の仮想ユーザを実行します。

- ▶ [[ツール] > [オプション] > [実行ファイルの保存場所] タブで定義されている通り]：グローバル設定で指定された場所に結果を保存します。
- ▶ [一時ディレクトリ (現在のロード ジェネレータ上)]：コントローラに対して、ロード・ジェネレータ・コンピュータのハード・ディスク上に実行ファイル (シナリオの実行結果と仮想ユーザ・スクリプト) を保存するように指示します。
- ▶ [共有ネットワーク ドライブ上]：コントローラに対して、シナリオの実行結果、仮想ユーザ・スクリプト、またはその両方を、共有ネットワーク・ドライブに保存するように指示します。共有ネットワーク・ドライブとは、コントローラと、シナリオ内のすべてのロード・ジェネレータが読み取りおよび書き込みの権限を持っているドライブです。

注：ロード・ジェネレータが **localhost** の場合、LoadRunner はスクリプトと結果を共有ネットワーク・ドライブに保存し、場所を設定するためのチェック・ボックスとラジオ・ボタンはすべて無効になります。ファイアウォール越しに監視している場合には、[実行ファイルの保存場所] の設定は関係しません。

結果を格納するためのネットワークの場所の設定については、第 11 章「シナリオの実行準備」を参照してください。

[UNIX 環境] タブ

[ロードジェネレータの情報] ダイアログ・ボックスで、[UNIX 環境] タブを選択して、各 UNIX ロード・ジェネレータにログイン・パラメータとシェルのタイプを設定します。



[ログイン] :

- ▶ [名前] : ロード・ジェネレータが UNIX 環境の場合に、ロード・ジェネレータのログイン情報を設定します。標準設定では、LoadRunner は NT ユーザ名を UNIX ログイン用に使用します。つまり、NT ログイン名が **lrunner** の場

合、コントローラは **lrunner** としてロード・ジェネレータにログインします。別のログイン名を使って UNIX 環境のロード・ジェネレータにログインするには、**[名前]** チェック・ボックスを選択して、使用する UNIX ログイン名を指定します。このオプションを使用すれば、例えば **bill** として NT コントローラにログオンし、**mike** として UNIX ロード・ジェネレータに接続することもできます。ただし、**mike** が **bill** に対し、**mike** のログイン名を使ってログオンすることを許可している必要があります。そのためには、**mike** の **.rhosts** ファイルの先頭に「+bill」という行を追加する必要があります。

- ▶ **[ログイン名に小文字を使用する]** : UNIX オペレーティング・システムとの間で大文字と小文字の区別が原因で問題が起きるのを回避するために、LoadRunner に対して、ログイン時に小文字の名前を使用するように指示します。

注 : エキスパート・モードの [ローカルユーザ] 設定の詳細については、付録 C 「エキスパート・モードでの作業」を参照してください。

[シェル環境設定] : リモート UNIX ロード・ジェネレータの UNIX シェル設定を指定します。

- ▶ **[標準設定のシェル]** : UNIX ロード・ジェネレータで標準設定のシェルを選択します。**csch** (標準設定の C シェル)、**bsh** (Bourne シェル)、または **ksh** (Korn シェル)。

注 : LoadRunner が Korn シェル上でアプリケーションを実行できるようにするためには、まず **.profile** ファイルに LoadRunner のすべての環境設定 (例えば、**M_LROOT** 定義や **LicenseManager** 変数など) を含める必要があります。UNIX **\$M_LROOT/templates** ディレクトリには、**.profile** ファイルのテンプレートである **dot profile** があります。このテンプレートを参考に、LoadRunner 環境設定に合わせて **.profile** ファイルを変更します。さらに、Korn シェル (**ksh**) を使用する場合は、シナリオ実行前に、**.cshrc** ファイルから LoadRunner の設定 (**M_LROOT** など) をすべて削除しておく必要があります。

- ▶ **[初期化コマンド]** : UNIX システムにログインするときに LoadRunner が使用するコマンド・ライン・オプションを入力します。この初期化コマンドは、シェルの起動時に実行されます。例えば、**ksh** を選択して、次の初期化コマンドを使用することができます。

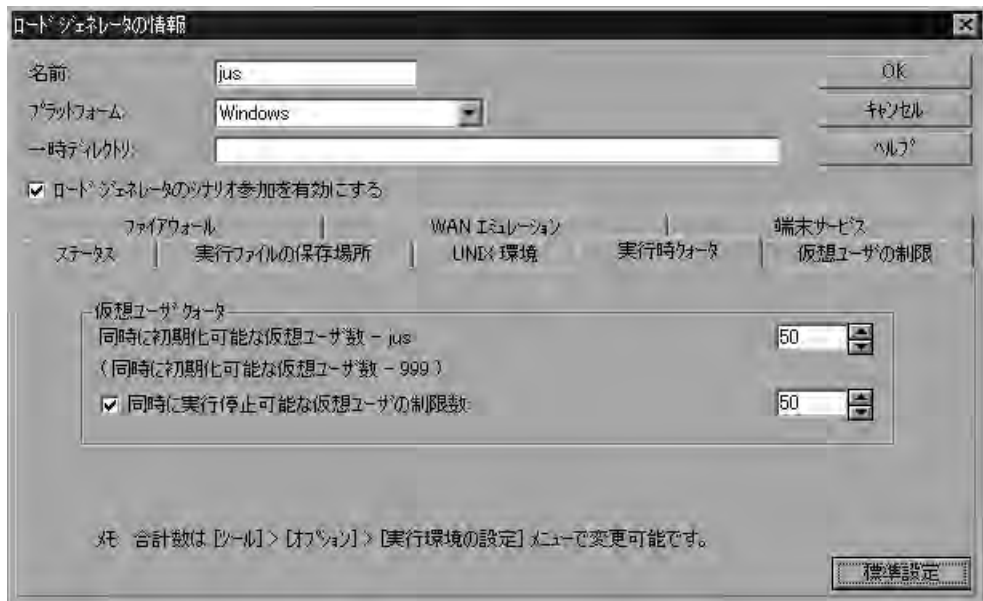
.profile;

注：ファイアウォール越しに仮想ユーザを監視または実行する場合には、UNIX 環境の設定は関係しません。

【実行時クォータ】 タブ

同時に多数の仮想ユーザを初期化または停止すると、ロード・ジェネレータに大きな負荷がかかります。ロード・ジェネレータへの負荷を軽減するために、仮想ユーザを少しずつ初期化または終了できます。

[ロードジェネレータの情報] ダイアログ・ボックスで、ロード・ジェネレータが同時に初期化または停止できる仮想ユーザ・タイプの最大数を指定するには、**【実行時クォータ】** タブを選択します。



【仮想ユーザ クォータ】：

- ▶ **【同時に初期化可能な仮想ユーザ数 - 現在のロード ジェネレータ】**：現在のロード・ジェネレータで一度に初期化できる仮想ユーザの最大数を選択します。

▶ **[同時に実行停止可能な仮想ユーザの制限数]**：現在のロード・ジェネレータで一度に停止できる仮想ユーザの最大数を選択します。

[標準設定]：一度に初期化または停止できる仮想ユーザの数を 50 に設定します。

[オプション] ダイアログ・ボックスの **[実行環境の設定]** タブを使って、シナリオ全体の実行時クォータを設定できます。シナリオ全体のクォータのグローバルな設定の詳細については、第 10 章「シナリオの設定」を参照してください。

【仮想ユーザの制限】タブ

[ロードジェネレータ情報] ダイアログ・ボックスで、ロード・ジェネレータが実行できる GUI、RTE、その他の仮想ユーザの最大数を変更するには、**[仮想ユーザの制限]** タブを選択します。



[利用可能タイプ]：ロード・ジェネレータが実行する仮想ユーザのタイプを選択します。

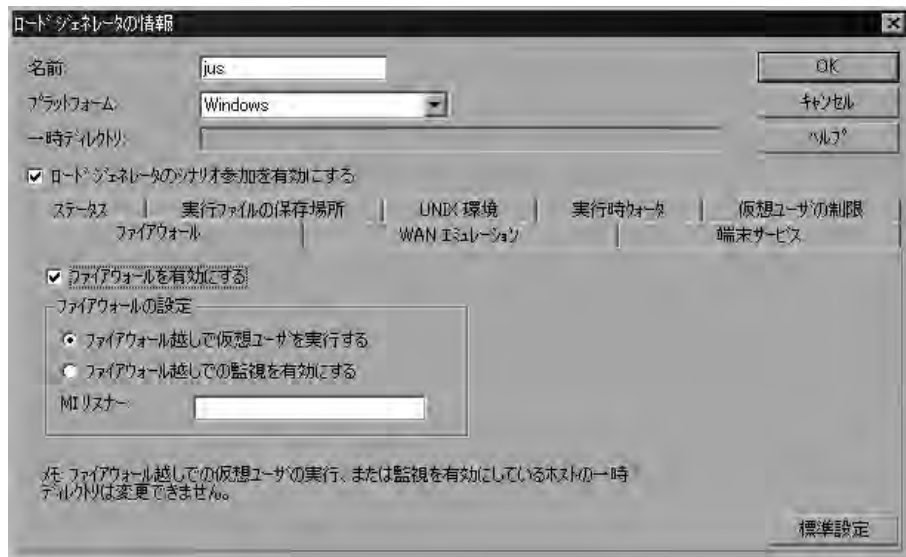
[実行制限数]：ロード・ジェネレータが実行する各タイプの仮想ユーザの最大数を選択します。

[標準設定]：[GUI/WinRunner] を「1」に、[RTE] を「1000」に、[その他の仮想ユーザ] を「5000」に設定します。

注：実行する仮想ユーザの最大数は、ライセンスされている仮想ユーザ数を超えてはなりません。仮想ユーザのライセンス数を確認するには、Mercury LoadRunner ランチャ画面で（[スタート] > [プログラム] > [Mercury LoadRunner] > [LoadRunner]）[Configuration] > [LoadRunner ライセンス] を選択します。

【ファイアウォール】タブ

[ロードジェネレータの情報] ダイアログ・ボックスで、【ファイアウォール】タブを選択して、仮想ユーザをファイアウォール越しに監視または実行できるようにします。



【ファイアウォールを有効にする】: LoadRunner がファイアウォール越しに仮想ユーザの実行または監視を行えるようにします。

注： [ファイアウォールを有効にする] オプションを選択した場合、一時ファイルを保存するための [一時ディレクトリ] オプションは無効になります。 [一時ディレクトリ] ボックス内の場所の指定はすべて消去されます。

[ファイアウォールの設定]：

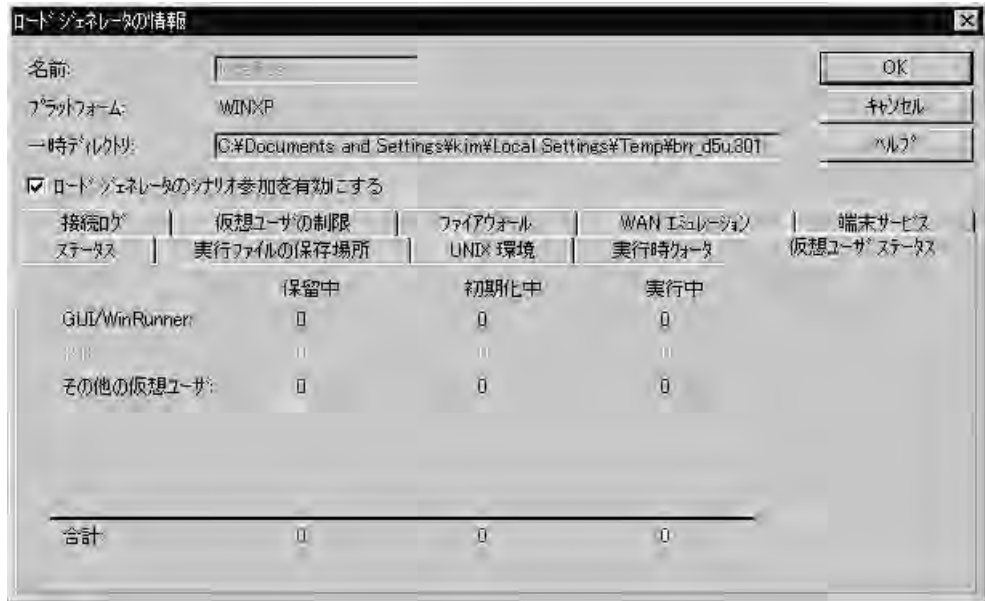
- ▶ **[ファイアウォール越しでの監視を有効にする]：** LoadRunner で、ファイアウォール越しにロード・ジェネレータ・マシンを監視できるようにします。
- ▶ **[ファイアウォール越しで仮想ユーザを実行する]：** LoadRunner がファイアウォールの外にあるロード・ジェネレータで仮想ユーザを実行できるようにします。

[MI リスナー]： ロード・ジェネレータが使用している MI リスナーの名前を入力します。

注： ロード・ジェネレータが接続されている場合には、**[ファイアウォール]** タブ内の値は変更できません。ロード・ジェネレータを切断するには、**[ロードジェネレータ]** ダイアログ・ボックスで切断するロード・ジェネレータを選択し、**[切断]** をクリックします。ロード・ジェネレータのステータスが「**ダウン**」に変わり、設定を変更できるようになります。ロード・ジェネレータが **localhost** の場合は、**[ファイアウォール]** タブは無効です。

【仮想ユーザステータス】タブ

【ロードジェネレータの情報】ダイアログ・ボックスで、【仮想ユーザステータス】タブを選択して、選択されたロード・ジェネレータ・マシンに接続しているすべての仮想ユーザのステータスを表示できます。



注：このタブは、ロード・ジェネレータ・マシンが接続されているときにだけ表示されます。

【GUI/WinRunner】：「保留中」、「初期化中」、および「実行中」のGUI/WinRunner 仮想ユーザの数が表示されます。

【RTE】：「保留中」、「初期化中」、および「実行中」のRTE 仮想ユーザの数が表示されます。

【その他の仮想ユーザ】：「保留中」、「初期化中」、および「実行中」の、GUI/WinRunner および RTE 仮想ユーザ以外の仮想ユーザの数が表示されます。

【合計】：「保留中」、「初期化中」、および「実行中」の仮想ユーザの総数が表示されます。

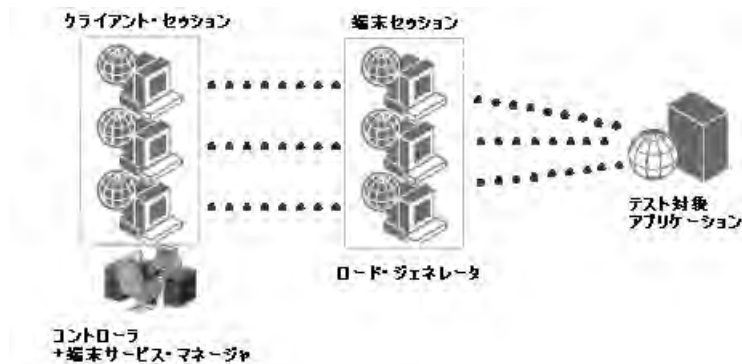
端末サービスの設定

LoadRunner の 端末サービス・マネージャを使用すると、端末サーバ上の負荷テスト・シナリオで実行されている複数のロード・ジェネレータをリモート管理できます。また、端末サーバを使用すれば、Windows ベースのロード・ジェネレータで実行できる GUI 仮想ユーザが1つのみという制限を回避できます。各 GUI 仮想ユーザごとに端末サーバ・セッションを開くことにより、同じアプリケーションに対して複数の GUI 仮想ユーザを実行できます。

端末サービスについて

端末サービスは、サーバに接続された各クライアントについてコンピューティング・リソースの集中管理を可能にし、各ユーザに個別の作業環境を提供します。端末サーバ・クライアントを使用すれば、リモート・マシンからサーバ・ベースのコンピューティング環境で作業できます。端末サーバはネットワークを通じてアプリケーションを送信し、端末エミュレーション・ソフトウェアを介してそのアプリケーションを表示します。各ユーザはログオンすると、それぞれの個別のセッションが表示されます。それらのセッションは、互いのクライアント・セッションとは無関係に、サーバのオペレーティング・システムによって透過的に管理されます。

次の図は、端末セッション中に LoadRunner コンポーネントがどのように連携して働くかを示しています。



端末サービス・マネージャ

端末サーバ・クライアントでは、複数の端末セッションを同時に実行できます。LoadRunner の Terminal Services Manager を使用すれば、シナリオで使用する端末の数と（十分な数の端末セッションが動作している場合）、各端末で実行できる仮想ユーザの最大数を選択できます。Terminal Services Manager は、その指定に基づいて、仮想ユーザをクライアント・セッション間で均等に分配します。

LoadRunner の Terminal Services Manager を使用するには、次の手順を実行します。

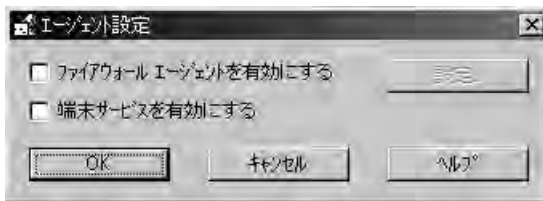
- ▶ ロード・ジェネレータ・マシンで端末サーバ・エージェントを設定します。
- ▶ コントローラ・マシンで端末クライアント・セッションを開始します。
- ▶ LoadRunner 端末サービス・マネージャを使って、端末サーバで仮想ユーザを分配します。

端末サーバ・エージェントの設定

ロード・ジェネレータ・マシンの端末サーバ・エージェントを設定する前に、端末サーバ・マシンにロード・ジェネレータがインストールされていることを確認してください。詳細については、『Mercury LoadRunner インストール・ガイド』を参照してください。

ロード・ジェネレータ・マシンで端末サーバ・エージェントを設定するには、次の手順を実行します。

- 1 システム・トレイでアイコンを右クリックして [閉じる] を選択し、LoadRunner エージェントを停止します。
- 2 [スタート] > [プログラム] > [Mercury LoadRunner] > [Advanced Settings] から [Agent Configuration] を実行するか、< LoadRunner のインストール先フォルダ > %launch_service%bin%AgentConfig.exe を実行します。[エージェント設定] ダイアログ・ボックスが開きます。



- 3 ファイアウォール越しに仮想ユーザの実行または監視を行う場合は、[ファイアウォール エージェントを有効にする] チェック・ボックスを選択し、[設定]

をクリックします。エージェントの設定については、271 ページ「エージェント設定」を参照してください。

- 4 **[端末サービスを有効にする]** チェック・ボックスを選択します。
[OK] をクリックします。
- 5 デスクトップのショートカットをダブルクリックするか、[スタート] > [プログラム] > [Mercury LoadRunner] > [LoadRunner Agent Process] を選択して LoadRunner エージェントをプロセスとして再起動します。

注：LoadRunner エージェントの詳細については、423 ページ「LoadRunner エージェントを使った作業」を参照してください。

端末クライアント・セッションの開始

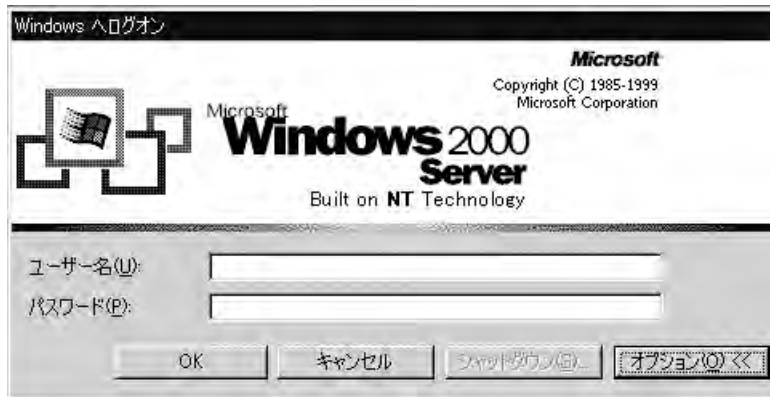
端末クライアント・セッションを開始するには、コントローラ・マシンに端末（ターミナル）サービス・クライアントがインストールされている必要があります。端末（ターミナル）サービス・クライアントのインストールの詳細については、端末サーバのインストール・マニュアルを参照してください。

コントローラ・マシンで端末クライアント・セッションを開くには、次の手順を実行します。

- 1 [スタート] > [プログラム] > [Terminal Services Client] > [Terminal Services Client] を選択します。[ターミナル サービス クライアント] ダイアログ・ボックスが開きます。



- 2 [サーバー] ボックスで、端末（ターミナル）サーバの名前または IP アドレスを入力するか、利用可能なサーバのリストから端末（ターミナル）サーバを選択します。
- 3 [画面領域] ボックスで、端末（ターミナル）クライアントのウィンドウ・サイズを選択します。
- 4 [接続] をクリックします。[Windows ヘログオン] ダイアログ・ボックスが開きます。

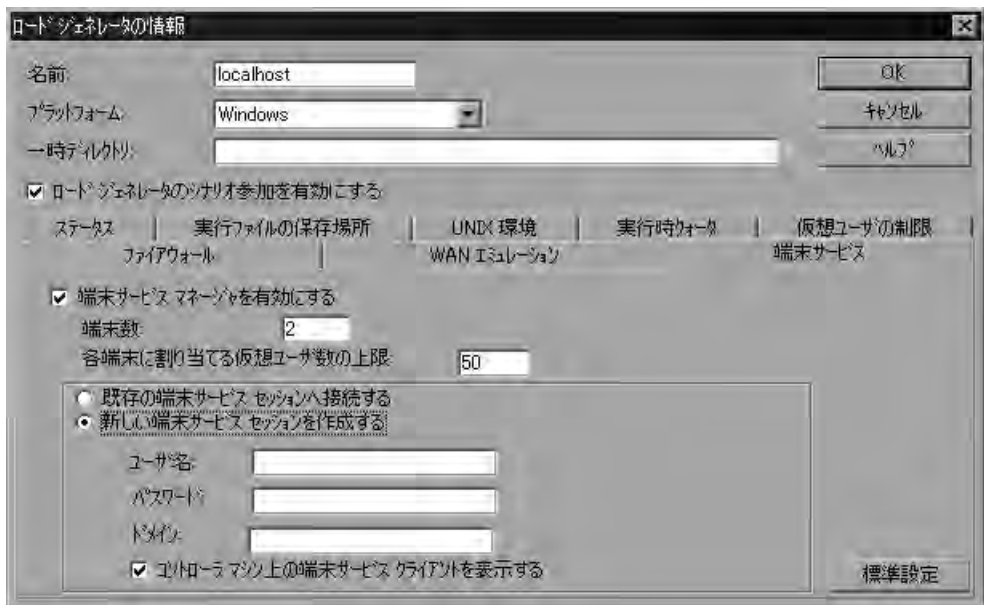


- 5 端末（ターミナル）サーバ用のユーザ名、パスワード、ドメイン名（必要な場合）を入力し、[OK] をクリックします。端末（ターミナル）クライアント・ウィンドウが開きます。
- 6 手順 1～2 を繰り返して、必要な数のセッションを開きます。

注：シナリオ実行中に仮想ユーザを実行する端末ごとに、端末サーバ・クライアント・セッションを開く必要があります。

端末サーバでの仮想ユーザの分配

端末サーバ上の負荷テスト・シナリオで実行される仮想ユーザを分配するには、[**端末サービス**] タブを選択します。



[**端末サービス マネージャを有効にする**]：コントローラでロード・ジェネレータ・マシン上の端末セッションを使用して負荷を自動的に管理できるようにします。

- ▶ [**端末数**]：負荷テストで使用する端末の数を入力します。

- ▶ **[各端末に割り当てる仮想ユーザ数の上限]**：各端末で実行する仮想ユーザの最大数を入力します。端末あたりの最大仮想ユーザ数は、スクリプトで使用されている仮想ユーザのタイプによって異なります。

この機能を使用すれば、余分な参照を加えずに、ロード・ジェネレータ・マシン名だけでの参照が可能になります。

例えば、`my_machine`、`my_machine:1`、`my_machine:2` となるような3つのロード・ジェネレータ・セッションを使用する場合には、ロード・ジェネレータ・リストにロード・ジェネレータとして「`my_machine`」だけ挿入し、3つの端末について端末サービス・マネージャを有効にします。

注：この機能を使用しなければ、`my_machine`、`my_machine:1`、`my_machine:2` の3つの端末をそれぞれ個々のロード・ジェネレータとして挿入しなければなりません。

[既存の端末サービス セッションへ接続する]：既存の（開いている）端末セッションに接続できます。

注：このオプションを選択すると、シナリオ実行中に仮想ユーザを実行する端末ごとに、端末サーバ・クライアント・セッションを手作業で開く必要があります。

[新しい端末サービス セッションを作成する]：コントローラで、ユーザ名とパスワードを使用して端末セッションを自動的に開いたり閉じたりできます。

- ▶ **[ユーザ名]**：端末サービス・セッションのユーザ名を入力します。
- ▶ **[パスワード]**：パスワードを入力します。
- ▶ **[ドメイン]**：端末サービス・セッションのドメインを指定します。

注：この機能は、コントローラとロード・ジェネレータがファイアウォール越しに接続されている場合にはサポートされません。

[コントローラ マシン上の端末サービス クライアントを表示する]：RDP クライアントを使用する新規の端末サービス・セッションとやりとりできるようにするには、このオプションを選択します。

[標準設定]：端末の数を 2 に、端末あたりの最大仮想ユーザ数を 50 に設定します。

トラブルシューティング

コントローラと端末サーバ上のロード・ジェネレータとの間の接続を確認します。コントローラで、対象ロード・ジェネレータを [ロードジェネレータ] ダイアログ・ボックスで選択し、**[接続]** をクリックします。ロード・ジェネレータが接続されると、ステータスが「**ダウン**」から「**準備完了**」に変わります。

接続がない場合は、端末サーバのシステム・トレイに LoadRunner エージェントのアイコンが表示されているかどうかチェックします。このアイコンは、エージェントが実行中であることを示します。必要なら、**[スタート] > [プログラム] > [Mercury LoadRunner] > [LoadRunner Agent Process]** から LoadRunner エージェントを再起動します。

WAN エミュレーションの設定

負荷テストのシナリオで、Shunra WAN エミュレータを使い、様々なネットワーク・インフラストラクチャの動作をエミュレートできます。WAN エミュレーションにより、エンド・ユーザの応答時間とパフォーマンスに対する WAN (Wide Area Network) の影響を、配備に先がけてシミュレートおよびテストできます。

WAN エミュレーションについて

WAN エミュレーションにより、WAN 配備される製品の実際のネットワークでのポイント・ツー・ポイント・パフォーマンスを、テスト環境で正確にテストできます。遅延時間、パケットの紛失、リンク障害、LAN 上の動的ルーティングの影響など可能性が高い WAN の影響を取り入れることによって、WAN 網を多様な面から特徴付け、単一のネットワーク環境でエミュレーションを効率的

に制御できます。WAN エミュレーション監視レポートで、ネットワーク・パフォーマンスに対するエミュレーション設定の影響を観察できます。



注：WAN エミュレーションは、Windows プラットフォームで実行しているロード・ジェネレータのみ使用できます。UNIX プラットフォームで実行しているロード・ジェネレータに対しては、[WAN エミュレーション] タブは無効です。

WAN エミュレータのセットアップ

Shunra WAN エミュレータを使用するには、LoadRunner インストール CD を使って、ロード・ジェネレータ・マシンに WAN エミュレータ・ドライバをインストールしておく必要があります。詳細については、『Mercury LoadRunner インストール・ガイド』を参照してください。

注：WAN エミュレーションには別途ライセンスが必要です。ライセンスについては、Mercury のカスタマー・サポート (http://www.mercury.co.jp/service_support/support/) にお問い合わせください。

WAN エミュレータの設定

負荷テストについて、コントローラ・マシンから WAN エミュレーションの設定を行うには、[WAN エミュレーション] タブを選択します。



[ロードジェネレータでWANエミュレーションを行う]：このチェック・ボックスを選択すると、WANエミュレーションがシナリオ実行時に自動的に起動するようになります。

[定義済みのプロファイルの設定]：遅延時間とパケットの紛失率が設定された定義済みプロファイルを選択します。次のプロファイル設定を使用できます。

- ▶ **[プロファイルなし]**：これは標準設定です。プロファイルが選択されていないか、定義済みプロファイルが手作業で変更されています。遅延時間：0ミリ秒。パケット紛失率：1%
- ▶ **[メトロポリタンエリアネットワーク (MAN) リンク]**：メトロポリタン・エリア・ネットワーク・リンクをエミュレートします。遅延時間：20ミリ秒。パケット紛失率：1%
- ▶ **[本土低輻輳回線 (地上回線)]**：ネットワーク・トラフィックの輻輳が少ない本土地上波リンクをエミュレートします。遅延時間：40ミリ秒。パケット紛失率：1%

- ▶ **[本土輻輳回線（地上回線）]**：ネットワーク・トラフィックの輻輳が多い本土地上波リンクをエミュレートします。遅延時間：100 ミリ秒。パケット紛失率：3%
- ▶ **[大西洋横断低輻輳回線（地上回線）]**：ネットワーク・トラフィックの輻輳が少ない海外向け地上波リンクをエミュレートします。遅延時間：60 ミリ秒。パケット紛失率：1%
- ▶ **[大西洋横断輻輳回線（地上回線）]**：ネットワーク・トラフィックの輻輳が多い海外向け地上波リンクをエミュレートします。遅延時間：120 ミリ秒。パケット紛失率：3%
- ▶ **[大西洋横断低輻輳回線（衛星回線）]**：ネットワーク・トラフィックの輻輳が少ない衛星リンクをエミュレートします。遅延時間：280 ミリ秒。パケット紛失率：1%
- ▶ **[大西洋横断輻輳回線（衛星回線）]**：ネットワーク・トラフィックの輻輳が多い衛星リンクをエミュレートします。遅延時間：400 ミリ秒。パケット紛失率：3%

[遅延時間]：IP パケットが WAN を通過するのにかかる時間を表す値（ミリ秒）が表示されます。これに影響するのは通常、地理的な距離、使用する帯域幅、2 地点間の経路のネットワーク負荷、および地上波リンクかどうかです。標準設定は 0 ミリ秒です。遅延時間を手動で設定するには、**[遅延時間]** スライダーを希望の位置に合わせるか、**[遅延時間]** ボックスに値を入力します。標準設定の範囲は 0-400 ミリ秒です。

注：WAN エミュレーションの遅延時間の範囲は最大 8000 ミリ秒まで拡大できます。テキスト・エディタで **wlrun7.ini** ファイルを開き、次の形式で希望の値を入力します。

例：**[WAN_Emulation]MaxLatency=1000**

[パケットの紛失]：データが WAN を通過する際に IP パケットが紛失する確率を表す値が表示されます。リンク障害や極端に高いネットワーク負荷が原因で、パケットが紛失することがあります。標準設定は 1% です。パケットの紛失率を手動で設定するには、**[パケットの紛失]** スライダーを希望の位置に合わせます。

[**すべてに適用**] : WAN エミュレーション設定を, [ロード ジェネレータ] ダイアログ・ボックスのリストに表示されているすべてのロード・ジェネレータに適用します。

[**ホストの除外**] : [ホストの除外] ダイアログ・ボックスを開きます。このダイアログ・ボックスで, エミュレートする WAN から特定のホスト名または IP アドレスを除外できます。エミュレートされた WAN からホストを除外する必要がある状況については, 101 ページ「WAN エミュレーションからのホストの除外」を参照してください。

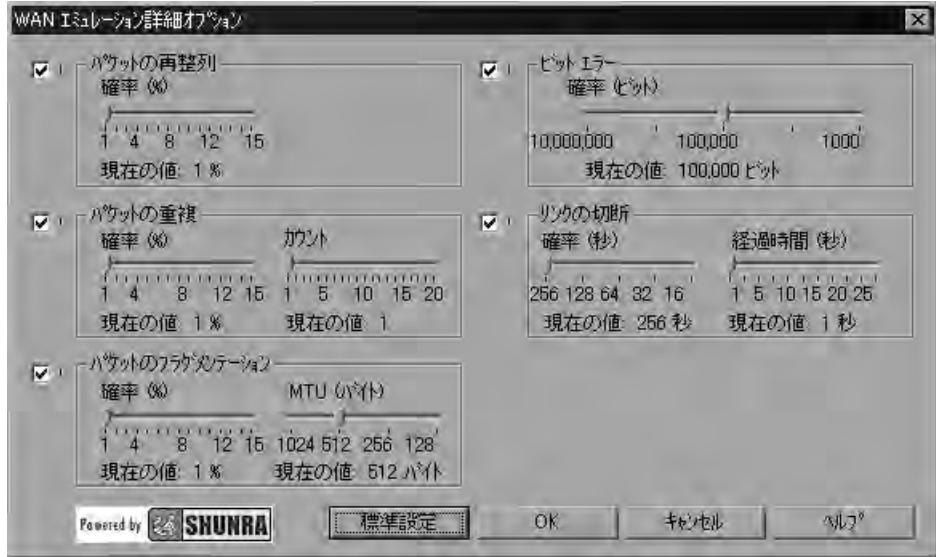
[**詳細設定**] : [WAN エミュレーション詳細オプション] ダイアログ・ボックスを開きます。詳細オプションについては, 99 ページ「WAN エミュレーション詳細オプションの設定」を参照してください。

[**標準設定**] : 標準設定に戻します。

注 : ロード・ジェネレータに接続中は, WAN エミュレーションの設定を変更できません。ロード・ジェネレータを切断するには, [ロード ジェネレータ] ダイアログ・ボックスで切断するロード・ジェネレータを選択し, [**切断**] をクリックします。ロード・ジェネレータのステータスが「**ダウン**」に変わり, 設定を変更できるようになります。ロード・ジェネレータが **Localhost** の場合, WAN エミュレーションは無効です。

WAN エミュレーション詳細オプションの設定

WAN エミュレーション詳細オプションでは, パケットの再整列, パケットの重複, パケットのフラグメンテーション, ビット・エラー, リンクの切断を設定できます。詳細オプションを設定するには, [WAN エミュレーション] タブで [**詳細設定**] ボタンをクリックします。



注：標準設定では、すべてのオプションが有効になっています。オプション設定を変えるには、スライダを動かして希望の値にします。プロファイルの値は設定範囲の下に表示されます。

[パケットの再整列]：WAN 網通過の際にパケットの順序が入れ替わる確率。標準設定は 1% です。

[パケットの重複]：WAN 網通過の際にパケットの重複が生じる確率。重複発生時に作成されるパケットごとのコピー数を指定します。標準発生率は 1% です。標準設定は数量は 1 です。

[パケットのフラグメンテーション]：WAN 網通過の際にパケット・フラグメンテーションが発生（最大伝送単位が短い）する確率。MTU は最大サイズのパケットすなわちフレーム（バイトで指定）であり、インターネットなどのパケット・ベースまたはフレーム・ベースのネットワークで送信できます。標準発生率は 1% です。MTU の標準設定は 512 バイトです。

[**ビットエラー**]：エミュレータが1ビットを反転させる頻度。指定された数のビットがWAN網を通過するたびにビットの反転が発生するようにします。確率の標準設定は100,000ビットです。

[**リンクの切断**]：パケットのWAN網通過の際に、ネットワーク切断が発生する確率（平均頻度）と切断時間。標準の切断頻度は256秒に1回です。標準の切断時間は1秒です。

[**標準設定**]：標準設定に戻します。

WAN エミュレーションからのホストの除外

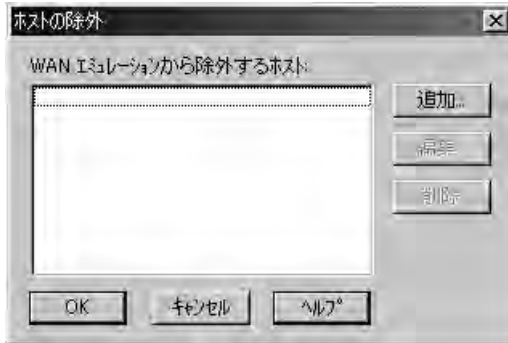
状況に応じて、WAN エミュレーションから特定のホストを除外できます。つまり、指定されたホストに対するトラフィックに影響を与えないように、WAN エミュレータを設定できます。エミュレーションの影響を受けないネットワーク・トラフィックは、WAN の影響を一切受けず、WAN エミュレーション監視レポートにも含まれません。

エミュレートされる WAN からホストを除外する必要があるのは、次のような場合です。

- ▶ Web サーバとデータベース・サーバを含むマルチプロトコル・シナリオで、データベース・サーバからの情報が負荷テストの一部として要求されない場合。
- ▶ ユーザが共有ネットワーク・ドライブでスクリプトを実行および格納する場合。
- ▶ コントローラが TCP 設定を使用してファイアウォール越しに仮想ユーザを実行および監視している場合。MI リスナーがコントローラとは別のマシンにある場合は、MI リスナー・マシンを除外する必要があります。
- ▶ コントローラが HTTPS 設定を使用してファイアウォール越しに仮想ユーザを実行または監視している場合。プロキシ・サーバの IP アドレスは除外します。

【ホストの除外】 ダイアログ・ボックスについて

特定のホストを WAN エミュレーションから除外するには、[ロードジェネレータの情報] ダイアログ・ボックスの [WAN エミュレーション] タブで [ホストの除外] を選択します。



[追加] : [ホストの追加] ダイアログ・ボックスを開きます。WAN エミュレーションから除外するマシンの名前または IP アドレスを入力します。

注 : コントローラ・マシンとネットワーク・ファイル・サーバ（ネットワーク・インストール構成の場合）を除外する必要はありません。これらはエミュレートされた WAN から自動的に除外されます。

[編集] : 変更するホスト名または IP アドレスを [WAN エミュレーションから除外するホスト] リストから選択し、[ホストの編集] ダイアログ・ボックスでそのホストに変更を加えることができます。

[削除] : [WAN エミュレーションから除外するホスト] リストからホスト名または IP アドレスを削除します。

注 : マシンの名前を入力した場合は、LoadRunner によって名前が解決され、[WAN エミュレーションから除外するホスト] リストの名前がマシンの IP アドレスに置き換えられます。

WAN エミュレーションの停止と開始

シナリオ実行中、いつでも WAN エミュレーションの停止と開始ができます。

WAN エミュレーションを停止または開始するには、次の手順を実行します。

- 1 WAN エミュレーションを停止するには、[シナリオ] > [WAN エミュレーションの停止] を選択します。
- 2 WAN エミュレーションを開始するには、[シナリオ] > [WAN エミュレーションの開始] を選択します。

スクリプトの設定

仮想ユーザまたは仮想ユーザ・グループのスクリプトを選択したら、そのスクリプトの編集や、[仮想ユーザ情報] または [グループ情報] ダイアログ・ボックスからの詳細の表示ができます。

仮想ユーザ・グループで使用するスクリプトの編集および詳細を表示するには、次の手順を実行します。

- 1 変更対象のスクリプトが含まれている仮想ユーザ・グループを選択し、[シナリオ グループ] 表示枠の右にある [詳細] ボタンをクリックするか、変更対象の仮想ユーザ・グループを右クリックして [詳細] を選択します。[グループ情報] ダイアログ・ボックスが表示され、スクリプトの現在の名前、パス、およびタイプが表示されます。

 詳細(D)...



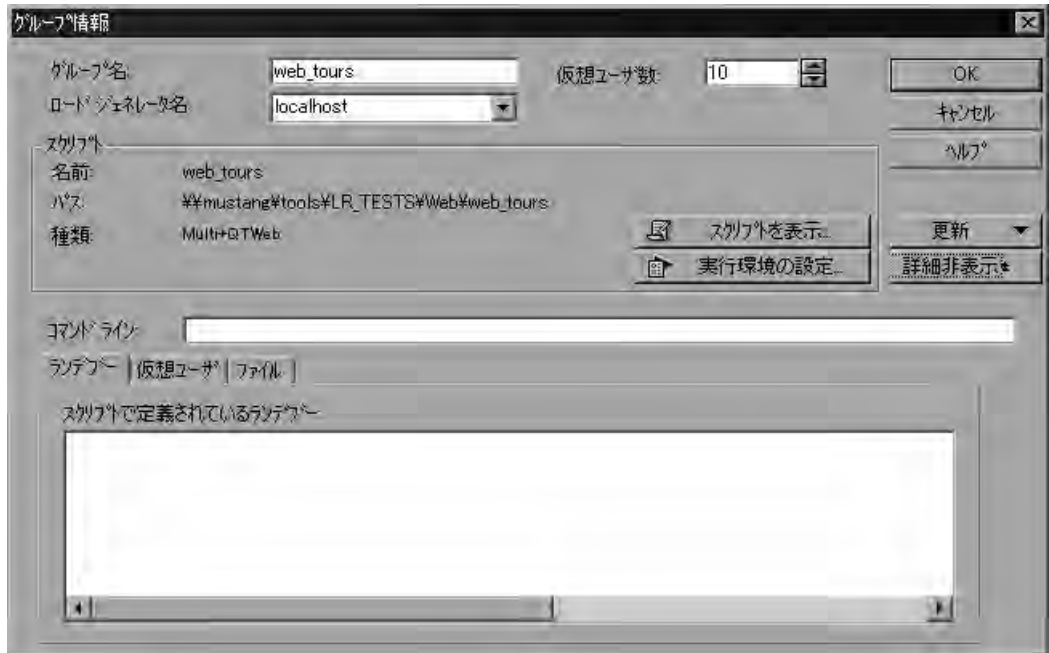
- 2 [実行環境の設定] をクリックして、スクリプトの実行環境を設定します（任意）。詳細については、71 ページ「仮想ユーザ実行環境の設定」を参照してください。

注：コントローラで実行環境の設定を変更すると、LoadRunner は変更された設定を使ってスクリプトを実行します。初期設定に戻すには、[更新] ボタンをクリックし、[実行環境の設定] を選択します。

- 3 スクリプトを編集するには、[スクリプトを表示] をクリックします。スクリプト作成ツール、VuGen が起動します。スクリプトの編集の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド』を参照してください。

注：コントローラの実行中に VuGen を使ってスクリプトに変更を加える場合、[更新] ボタンをクリックして [スクリプト] を選択し、シナリオ内のスクリプトの詳細を更新します。

- 4 [詳細表示] をクリックして [グループ情報] ダイアログ・ボックスを拡張すると、さらに多くのスクリプト情報が表示されます。



- 5 [コマンドライン] ボックスに、スクリプトの実行時に使用する任意のコマンド・ライン・オプションを入力します。例を次に示します。

-x value -y value

コマンド・ライン引数の値をスクリプトに渡す方法の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

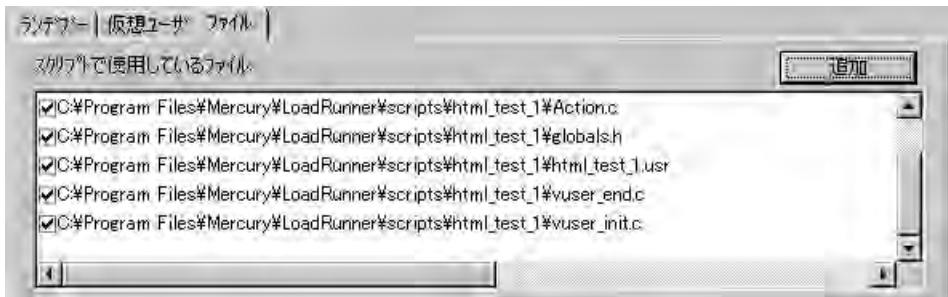
- 6 選択されたスクリプト内に含まれるランデブー・ポイントを参照するには、[ランデブー] タブをクリックします。

- 7 選択されたスクリプトに関連している仮想ユーザのリストを参照するには、**[仮想ユーザ]** タブをクリックします。



まだ仮想ユーザを作成していない場合には、空のボックスが表示されます。

- 8 スクリプトによって使用されるファイルのリストを参照するには、**[ファイル]** タブをクリックします。




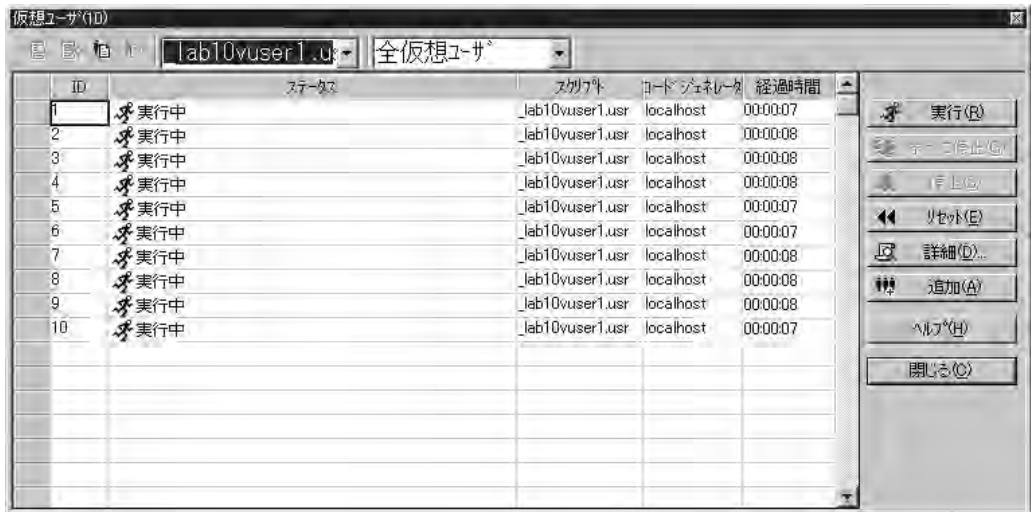
標準設定では、このリストにはスクリプトのディレクトリ内のファイルがすべて表示されます（スクリプトがスクリプト・リストに追加された後のみ）。これらのファイルには、構成設定ファイル、スクリプトの `init`、`run`、および `end` 部分、パラメータ化定義ファイル、`.usr` ファイルが含まれます。このリストにファイルを追加するには、**[追加]** をクリックし、ファイル名を追加します。追加したファイルは削除できますが、それ以外の表示されているファイルは削除できません。

注：リモート・ロード・ジェネレータ・マシンで Visual C++ 仮想ユーザを実行するには、スクリプト・リストで使用されているファイルに仮想ユーザの **.dll** を追加する必要があります。

- 9 **[OK]** をクリックして、**[グループ情報]** ダイアログ・ボックスを閉じます。

各仮想ユーザで使用されるスクリプトの編集と詳細の表示は、次の手順で行います。

-  **[仮想ユーザ]** 1 **[シナリオ グループ]** 表示枠の右の **[仮想ユーザ]** ボタンをクリックします。
[仮想ユーザ] ダイアログ・ボックスが表示されます。



- 2 スクリプトの詳細を表示するには、**[詳細]** をクリックします。**[仮想ユーザ情報]** ダイアログ・ボックスにスクリプトの名前とパスが表示されます。別のスクリプトを選択するには、**[参照]** ボタンをクリックし、パスとファイル名を選択します。VB 仮想ユーザ・スクリプトを選択するには、**.usr** ファイルを探します。

注：スクリプトの場所を指定するとき、現在のシナリオ・ディレクトリを基準とする相対的な場所を指定できます。詳細については、108 ページ「スクリプトに対する相対パスの使用」を参照してください。

- 3 スクリプトを編集するには、[仮想ユーザ] ダイアログ・ボックスでスクリプトを右クリックし、[スクリプトの表示] を選択します。スクリプト作成ツール、VuGen が起動します。スクリプトの編集の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。
- 4 指定した実行環境の設定の変更を、VuGen を使ったスクリプトの記録中に行うには、[仮想ユーザ] ダイアログ・ボックスでスクリプトを右クリックして [実行環境の設定] を選択します。

注：1 つの仮想ユーザの実行環境の設定を変更すると、グループ内で同じスクリプトを使用しているすべての仮想ユーザの実行環境の設定が変更されます。

複数のスクリプトを強調表示すれば、72 ページ「複数のスクリプトの実行環境設定の変更」の説明に従って、共有モードの実行環境の設定を変更できます。

個別の実行環境の設定の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。

スクリプトに対する相対パスの使用

スクリプトの場所を指定する際に、その場所を相対的に指定できます。現在のシナリオ・ディレクトリまたは LoadRunner のインストール・ディレクトリからの相対パスとして場所を指定できます。

スクリプト・パスの先頭を次のどちらかの形式で入力することにより、現在のシナリオ・ディレクトリからの相対パスを指定できます。

- .\$ シナリオ・ディレクトリからの相対パスであることを示します。
- ..\$ シナリオ・ディレクトリの親ディレクトリからの相対パスであることを示します。

例えば、現在のシナリオが **F:\$scenarios** にある場合、スクリプト **F:\$scenarios\$scripts\$user1.usr** を指定するには、次のように入力します。

```
.$scripts$user1.usr
```

スクリプト・パスの先頭にパーセント記号 (%) をつけることで、LoadRunner のインストール・ディレクトリからの相対パスを指定できます。例えば、LoadRunner のインストール・ディレクトリが **F:\$LoadRunner** である場合、スクリプト **F:\$LoadRunner\$scripts\$user1.usr** を指定するには、次のように入力します。

```
%. $scripts$user1
```

注：相対パスを指定するとき、**M:\$LR\$my_tests\$.\$. \$test.usr** のように、パスに標準の DOS 表記 (\$ と .\$) を含めることができます。

シナリオを実行すると、標準設定ではスクリプトは仮想ユーザ・グループ・マシン上の一時ディレクトリにコピーされます。これにより、仮想ユーザ・グループのロード・ジェネレータは、ネットワークを通さず、ローカルでスクリプトにアクセスできます。

スクリプトが共有ネットワーク・ドライブに保存されるように、コントローラを設定できます (第 10 章「シナリオの設定」を参照してください)。スクリプトがネットワーク・ドライブに保存されるようにコントローラを設定した場合、そのドライブが仮想ユーザのロード・ジェネレータによって確実に認識されるようにする必要があります。[スクリプト] ウィンドウには、すべての仮想ユーザ・スクリプトとそのパスが含まれます。スクリプトのパスは、コントローラのロード・ジェネレータによるその場所のマッピングに基づきます。仮想ユーザのロード・ジェネレータで、これと異なるスクリプトのパスが設定さ

れている場合には、パスの変換が必要になります。パスの変換によって、コントローラのロード・ジェネレータのマッピングが、仮想ユーザのロード・ジェネレータのマッピングに変換されます。詳細については、付録B「パス変換の実行」を参照してください。

第 6 章

パーセント・モードでのマニュアル・シナリオの作成

シナリオ内で使う仮想ユーザの総数を定義し、さらに仮想ユーザの総数に対する割合とロード・ジェネレータを各スクリプトに割り当てることによって、パーセント・モードでマニュアル・シナリオを作成できます。本章では、パーセント・モードでマニュアル・シナリオを作成する方法について説明します。

本章では、次の項目について説明します。

- ▶ パーセント・モードでマニュアル・シナリオを作成する方法について
- ▶ 仮想ユーザの総数の定義
- ▶ スクリプトへのプロパティの割り当て
- ▶ スクリプトの設定
- ▶ シナリオの仮想ユーザ・グループ・モードへの切り替え

パーセント・モードでマニュアル・シナリオを作成する方法について

通常のマニュアル・シナリオを設計する場合、仮想ユーザ・グループを作成して、それにスクリプト、ロード・ジェネレータ・マシン、および仮想ユーザを割り当てます。パーセント・モードでは、シナリオで使う仮想ユーザの総数を定義し、各スクリプトに仮想ユーザの総数に対する割合とロード・ジェネレータを割り当てます。

新しいシナリオを作成するときに [新規シナリオ] ダイアログ・ボックスで [スクリプト間で仮想ユーザを分配するのに [パーセントモード] を使用する] を選択することによって、パーセント・モードに直接入ることができます。また、[シナリオ] > [シナリオを次のモードに変換: パーセントモード] を選

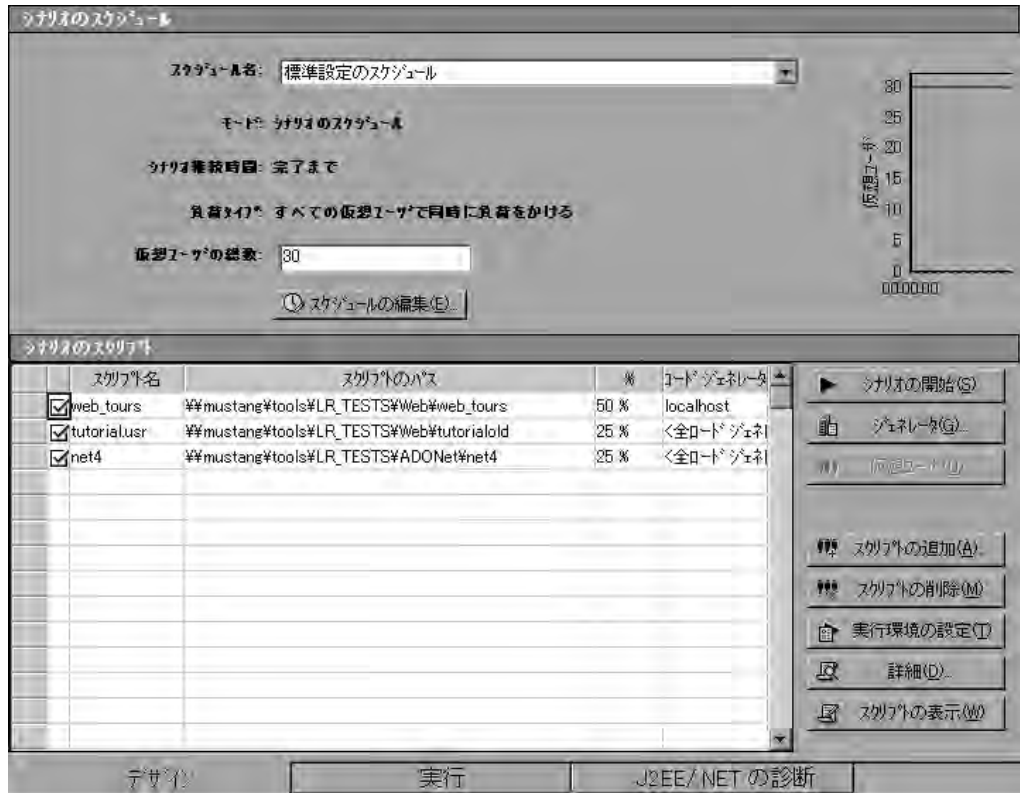
択することによって、仮想ユーザ・グループ・モードで作成したシナリオをパーセント・モードに切り替えることができます。

シナリオを仮想ユーザ・グループ・モードからパーセント・モードに切り替える場合、以下の点に注意します。

- ▶ 仮想ユーザ・グループに複数のスクリプトを定義した場合、パーセント・モードで作成される仮想ユーザ・スクリプトの数と仮想ユーザ・グループで定義されたスクリプトの数が一致します。
- ▶ <全ロード・ジェネレータ>に、パーセント・モードで作成されるすべての仮想ユーザ・スクリプトが割り当てられます。仮想ユーザ・グループに複数のロード・ジェネレータを定義した場合、仮想ユーザ・グループに割り当てたロード・ジェネレータに、パーセント・モードでスクリプトに割り当てた仮想ユーザが均等に分配されます。
- ▶ 仮想ユーザ・グループのスケジュール設定はすべて失われます。すべてのプロファイルにはシナリオのスケジュール設定だけが含まれます。

パーセント・モードの [デザイン] タブについて

パーセント・モードを使ってマニュアル・シナリオを作成するときには、[デザイン] タブに [シナリオのスケジュール] 表示枠と [シナリオのスクリプト] 表示枠が表示されます。



【シナリオのスケジュール】表示枠には、スケジュール・プロファイルの名前、スケジュール・モード、シナリオの継続時間、負荷の動作、シナリオ内で使用される仮想ユーザの総数など、スケジュール・プロファイルに関する情報が表示されます。【負荷のプレビュー】には、定義したシナリオ・スケジュールのグラフが表示されます。スケジュールの設定の詳細については、147ページ「シナリオのスケジュール」を参照してください。

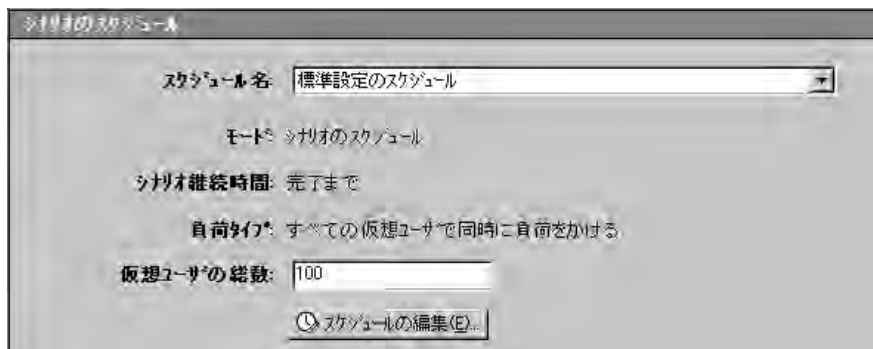
【シナリオのスクリプト】表示枠には、すべての有効および無効な仮想ユーザ・スクリプト、各スクリプトのパス、ロード・ジェネレータ・マシン、各スクリプトに割り当てられた仮想ユーザの総数の割合のリストが表示されます。

仮想ユーザ・スクリプトまたはシナリオに対して、次の操作が可能です。

- ▶ シナリオ内で使用する仮想ユーザの総数の定義
- ▶ スクリプト名、スクリプトのパス、ロード・ジェネレータ・マシン、その仮想ユーザ・スクリプトの仮想ユーザの総数の割合の定義
- ▶ 仮想ユーザ・スクリプトへの1つまたは複数のロード・ジェネレータ・マシンの追加、および追加したロード・ジェネレータの設定
- ▶ シナリオへの新規のスクリプトの追加とその設定
- ▶ シナリオの仮想ユーザ・スクリプトの有効化または無効化
- ▶ シナリオからの仮想ユーザ・スクリプトの削除
- ▶ シナリオのスケジュール設定
- ▶ シナリオの実行
- ▶ シナリオの停止
- ▶ シナリオのリセット
- ▶ シナリオの実行結果に関する設定

仮想ユーザの総数の定義

パーセント・モードでシナリオを作成する場合、スクリプトごとに仮想ユーザの数を定義するのではなく、シナリオ内で使う仮想ユーザの総数を定義します。[シナリオのスケジュール] ウィンドウに仮想ユーザの総数を入力します。

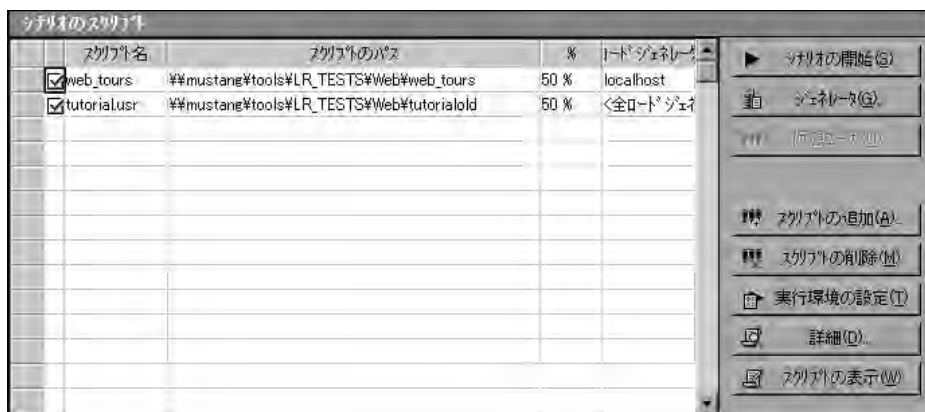


シナリオのスケジュールを作成する方法の詳細については、第8章「シナリオのスケジュール」を参照してください。

注：パーセント・モードでは仮想ユーザ・グループの設定を使用できません。

スクリプトへのプロパティの割り当て

[シナリオのスクリプト] 表示枠のリストには、[新規シナリオ] ダイアログ・ボックスで選択したスクリプト、または仮想ユーザ・グループ・モードで定義したスクリプトが表示されます。



[%] カラムには、各仮想ユーザ・スクリプトに自動的に分配される仮想ユーザの総数に対する割合が表示されます。シナリオの実行中に、各スクリプトは割り当てられた割合の仮想ユーザを実行します。[ロードジェネレータ] カラムには、各仮想ユーザ・スクリプトに対して自動的に<全ロードジェネレータ>が表示されます。

注：仮想ユーザ・グループに複数のロード・ジェネレータを定義した場合、仮想ユーザ・グループに割り当てたロード・ジェネレータに、パーセント・モードでスクリプトに割り当てた仮想ユーザが均等に分配されます。

各スクリプトで次の要素を変更できます。

- ▶ 仮想ユーザの総数のうち、スクリプトに割り当てる割合。
- ▶ 仮想ユーザがスクリプトを実行するロード・ジェネレータ。

スクリプトに割り当てる仮想ユーザの割合の変更は、次の手順で行います。

[シナリオのスケジュール] 表示枠で定義した仮想ユーザの総数に対する割合を、スクリプトの [%] カラムに入力します。これによって、仮想ユーザ・スクリプトの全割合が 100 パーセントになるように他のスクリプトの割合が変化します。

スクリプトのロード・ジェネレータを追加または変更するには、次の手順で行います。

- 1 スクリプトの [ロード ジェネレータ] カラムで、ロード・ジェネレータ名のリストから1つ以上のマシンを選択し、[OK] をクリックします。複数のマシンを選択した場合、スクリプトに割り当てた仮想ユーザがロード・ジェネレータ間で均等に分配されます。
- 2 また、[追加] を選択して、リストにロード・ジェネレータを追加することもできます。[新規ロード ジェネレータの追加] ダイアログ・ボックスが表示されます。



- 3 [名前] ボックスにロード・ジェネレータの名前を入力します。

- 4 [プラットフォーム] ボックスでは、ロード・ジェネレータを実行するプラットフォームのタイプを選択します。
- 5 [一時ディレクトリ] ボックスに、一時ファイルを保存する場所としてロード・ジェネレータ上の場所を入力するか、標準設定の場所を使用するためにボックスを空白のままにします。標準設定では、LoadRunner はシナリオ実行中、ロード・ジェネレータの環境変数 TEMP または TMP で指定された一時ディレクトリに一時ファイルを格納します。
- 6 ロード・ジェネレータをシナリオに参加できるようにするには、[**ロード ジェネレータのシナリオ参加を有効にする**] を選択します。
- 7 [詳細表示] をクリックして、ダイアログ・ボックスを拡張すると [新規ロード ジェネレータの追加] ダイアログ・ボックスにいくつかのタブが表示されます。各ロード・ジェネレータの設定の詳細については、79 ページ「ロード・ジェネレータの追加設定」を参照してください。
- 8 [OK] をクリックして、[新規ロード ジェネレータの追加] ダイアログ・ボックスを閉じます。LoadRunner は、ロード・ジェネレータ名のリストに新しいロード・ジェネレータを追加します。シナリオに新しいロード・ジェネレータを含めるには、ロード・ジェネレータ名のリストから対象のロード・ジェネレータを選択して、[OK] をクリックします。

シナリオに追加する各ロード・ジェネレータに対して上記の手順を繰り返します。

注：コントローラは、Windows ロード・ジェネレータ・マシンの CPU 使用率を監視し、過負荷のロード・ジェネレータへの仮想ユーザの負荷を自動的に停止し、シナリオに含まれるその他のロード・ジェネレータに分散します。詳細については、140 ページ「負荷分散」を参照してください。[ロード ジェネレータ] ダイアログ・ボックス内のアイコンを使用して、マシンの CPU 使用率のステータスを監視できます。ロード・ジェネレータの CPU 使用率に問題が生じると、ロード・ジェネレータ名の左側にあるアイコンに黄色の線が表示されます。マシンが過負荷になると、アイコンに赤色の線が表示されます。

ロード・ジェネレータの設定

ロード・ジェネレータの属性の設定は、[**ロード ジェネレータ**] ダイアログ・ボックスを使って、ロード・ジェネレータ・リストにロード・ジェネレータを追加しているときに設定できます。また既存のロード・ジェネレータの属性は

随時変更できます。また、[ロードジェネレータ] ダイアログ・ボックスを使って、シナリオで仮想ユーザを実行するロード・ジェネレータを指定できます。例えば、あるシナリオの実行にロード・ジェネレータが使えない場合は、[ロードジェネレータ] ダイアログ・ボックスを使って、そのロード・ジェネレータをロード・ジェネレータのリストから完全に削除してしまうのではなく、一時的に除外できます。[ロードジェネレータ] ダイアログ・ボックスを使用する手順については、76 ページ「[ロードジェネレータ] ダイアログ・ボックスについて」を参照してください。

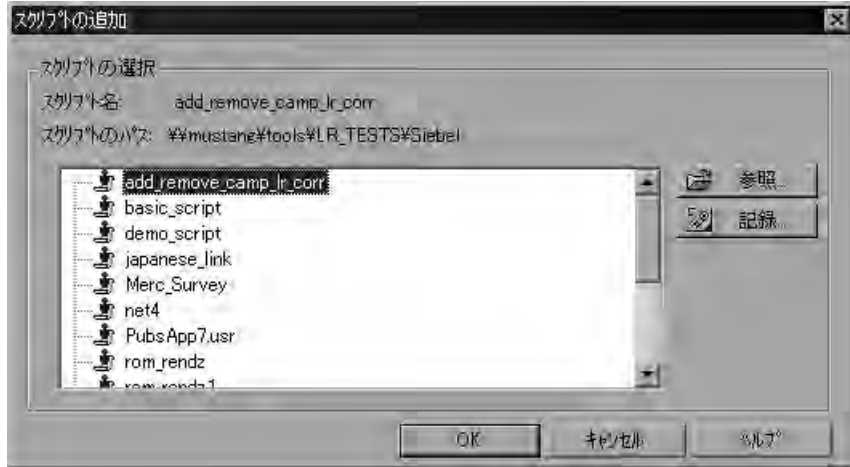
シナリオに参加しているすべてのロード・ジェネレータのグローバル設定を構成するには、LoadRunner の [オプション] ダイアログ・ボックスを使います。詳細については、第 10 章「シナリオの設定」を参照してください。

スクリプトの設定

[スクリプトの追加] ダイアログ・ボックスを使って、[シナリオのスクリプト] リストにスクリプトを追加できます。リストに追加したスクリプトは、選択して詳細表示したり、編集したり、有効化または無効化したり、実行環境の設定を変更したりできます。

スクリプトの追加は、次の手順で行います。

- 1 [シナリオのスクリプト] ウィンドウの右にある [スクリプトの追加] ボタンをクリックするか、カラム内で右クリックし、[スクリプトの追加] を選択します。[スクリプトの追加] ダイアログ・ボックスが表示されます。



- 2 [参照] ボタンをクリックします。[テストを開く] ダイアログ・ボックスが表示されます。

新しいスクリプトのパスとファイル名を選択します。

注： スクリプトの場所を指定する場合、現在のシナリオ・ディレクトリに相対する場所を指定できます。詳細については、108 ページ「スクリプトに対する相対パスの使用」を参照してください。

- 3 [開く] をクリックして、ファイルを選択します。[テストを開く] ダイアログ・ボックスが閉じ、[スクリプトの追加] ダイアログ・ボックスに新しいスクリプト名が表示されます。
- 4 [OK] をクリックすると、[スクリプトの追加] ダイアログ・ボックスが閉じ、[シナリオのスクリプト] リストに新しいスクリプト情報が追加されます。

[スクリプトの追加] ダイアログ・ボックスについて

[スクリプトの追加] ダイアログ・ボックスを使用して、スクリプトをシナリオに追加できます。

[**スクリプトの選択**]：現在のディレクトリ内にある使用可能なスクリプトが表示されます。

- ▶ [**スクリプト名**]：シナリオに追加するスクリプトをクリックします。スクリプトが [スクリプト名] カラムに表示されます。
- ▶ [**スクリプトのパス**]：スクリプト・ディレクトリのパスが表示されます。
- ▶ [**参照**]：別のディレクトリからスクリプトを選択できます。VB 仮想ユーザ・スクリプトを選択するには、.usr ファイルを探します。
- ▶ [**記録**]：スクリプトの記録を開始できるように、仮想ユーザ・ジェネレータを開きます。スクリプトの記録の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

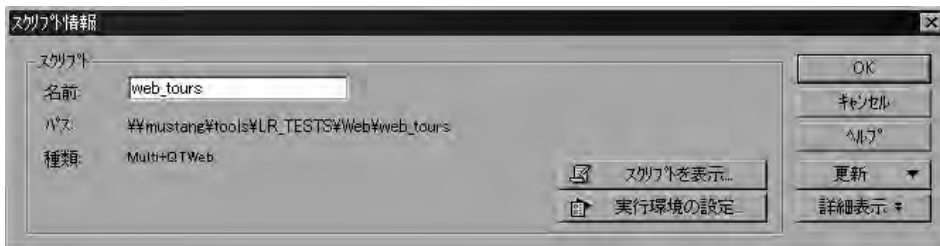
注：シナリオ実行中に、シナリオに仮想ユーザ・スクリプトを追加して実行できます。ただし、シナリオ内のすべての仮想ユーザがランプ・アップされた後でスクリプトを追加した場合、その新しいスクリプトは実行されません。

スクリプト情報の表示

リストに追加したスクリプトは、選択して詳細を表示したり、編集したり、有効化または無効化したり、実行環境の設定を変更したりできます。

スクリプトの詳細表示は、次の手順で行います。

- 1 スクリプトを選択して [シナリオのスクリプト] ウィンドウの右にある [**詳細**] ボタンをクリックするか、スクリプトを右クリックして [**詳細**] を選択します。[スクリプト情報] ダイアログ・ボックスが表示され、選択したスクリプトの「パス」、「名前」、「種類」が表示されます。



- 2 **〔実行環境の設定〕** をクリックして、スクリプトの実行環境の設定（オプション）を指定できます。これによって、コントローラで仮想ユーザ・スクリプトを実行する方法をカスタマイズできます。**〔実行環境設定〕** ダイアログ・ボックスが表示され、以前に **VuGen** を使って設定した設定値が表示されます。

注：コントローラで実行環境の設定を変更すると、LoadRunner は変更された設定を使ってスクリプトを実行します。初期設定に戻すには、**〔更新〕** ボタンをクリックし、**〔実行環境の設定〕** を選択します。

- 3 スクリプトを編集するには、**〔スクリプトを表示〕** をクリックします。スクリプト作成ツール、**VuGen** が起動します。スクリプトの編集の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

注：コントローラの実行中に **VuGen** を使ってスクリプトに変更を加える場合、**〔更新〕** ボタンをクリックして **〔スクリプト〕** を選択し、シナリオ内のスクリプトの詳細を更新します。

- 4 [詳細表示] をクリックして、[スクリプト情報] ダイアログ・ボックスを拡張すると、さらに多くのスクリプト情報が表示されます。



- 5 [コマンドライン] ボックスに、スクリプトの実行時に使用するコマンド・ライン・オプションをすべて入力します。例を次に示します。

-x value -y value

コマンド・ライン引数の値をスクリプトに渡す方法の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。

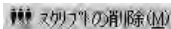
- 6 選択されたスクリプト内に含まれるランデブー・ポイントを参照するには、[ランデブー] タブをクリックします。
- 7 選択されたスクリプトに関連している仮想ユーザのリストを参照するには、[仮想ユーザ] タブをクリックします。
- 8 スクリプトによって使用されるファイルのリストを参照するには、[ファイル] タブをクリックします。標準設定では、このリストにはスクリプトのディレクトリ内のファイルがすべて表示されます（スクリプトがスクリプト・リストに追加された後のみ）。これらのファイルには、構成設定ファイル、スクリプトの init, run, および end 部分、パラメータ化定義ファイル、.usr ファイルが含

まれます。このリストにファイルを追加するには、**[追加]** をクリックし、ファイル名を追加します。

注：追加したファイルは削除できますが、それ以外の表示されているファイルは削除できません。

9 **[OK]** をクリックして、**[スクリプト情報]** ダイアログ・ボックスを閉じます。

スクリプトの削除は、次の手順で行います。



スクリプトを選択して **[シナリオのスクリプト]** ウィンドウの右にある **[スクリプトの削除]** ボタンをクリックするか、スクリプトを右クリックして **[スクリプトを削除]** を選択します。

スクリプトの無効化は、次の手順で行います。

仮想ユーザ・スクリプトの名前の左側にあるボックスをクリックします。これによって、スクリプト・エントリの色がグレーに変化します。これは、スクリプトがシナリオに参加しないことを意味します。仮想ユーザ・スクリプトを再度有効にするには、同じボックスをもう一度クリックします。

[スクリプト情報] ダイアログ・ボックスについて

選択したスクリプトの詳細を表示し、その設定を変更できます。

[スクリプト]：選択したスクリプトの詳細が表示されます。

- ▶ **[名前]**：選択したスクリプトの名前が表示されます。この名前を変更するには、**[名前]** ボックスに新しい名前を入力します。
- ▶ **[パス]**：スクリプト・ディレクトリのパスが表示されます。
- ▶ **[種類]**：選択したスクリプトのタイプが表示されます。
- ▶ **[スクリプトを表示]**：スクリプトを編集できるように、仮想ユーザ・ジェネレータを開きます。スクリプトの編集の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

- ▶ **[実行環境の設定]** : [実行環境設定] ダイアログ・ボックスが表示され、以前に VuGen を使って設定した実行環境の設定を編集できます。実行環境の設定の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。

[更新] : コントローラの実行中に**スクリプト**に変更を加える場合は、このボタンをクリックして [スクリプト] を選択し、シナリオ内のスクリプトの詳細を更新します。コントローラから実行環境の設定を変更する場合、実行環境の設定を初期設定に戻すには、**[実行環境の設定]** を選択します。

[詳細表示 / 詳細非表示] : 以下の情報を表示または非表示にします。

- ▶ **[コマンドライン]** : スクリプトの実行時に使用するコマンド・ライン・オプションを入力します。例えば、**-x value -y value** のように入力します。コマンド・ライン引数の値をスクリプトに渡す方法の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。
- ▶ **[ランデブー]** : 選択したスクリプトについて定義されたランデブー・ポイントが表示されます。
- ▶ **[仮想ユーザ]** : 選択したスクリプトに関連付けられた仮想ユーザがすべて表示されます。
- ▶ **[ファイル]** : 選択したスクリプトによって使用されるファイルがすべて表示されます。このリストからファイルを除外するには、そのファイルの隣にあるチェック・ボックスを選択します。このリストにファイルを追加するには、**[追加]** をクリックします。

シナリオの仮想ユーザ・グループ・モードへの切り替え

[シナリオ] > [シナリオを次のモードに変換 : 仮想ユーザ グループ モード]
を選択することによって、パーセント・モードで作成したシナリオを仮想ユーザ・グループ・モードに切り替えることができます。

注：仮想ユーザ・グループ・モードのシナリオをパーセント・モードに切り替えることもできます。詳細については、111 ページ「パーセント・モードでマニュアル・シナリオを作成する方法について」を参照してください。

LoadRunner は、マニュアル・シナリオが仮想ユーザ・グループ・モードからパーセント・モードへ、またはその逆に切り替えられようとしていることを知らせる警告メッセージを表示します。シナリオのモードを切り替える場合は、**[はい]** をクリックします。現在のモードのままにする場合は、**[いいえ]** をクリックします。

[シナリオ変換の前に、このダイアログをいつも表示する]：現在の警告が表示されないようにするには、このボックスをクリアします。現在の警告を元に戻すには、**[シナリオ]** > **[シナリオ モードの変換時に警告を表示]** を選択します。

シナリオをパーセント・モードから仮想ユーザ・グループ・モードに切り替える場合、以下の点に注意します。

- ▶ 各スクリプトが仮想ユーザ・グループに切り替えられます。
- ▶ 仮想ユーザ・スクリプトに複数のロード・ジェネレータを定義した場合、シナリオ切り替え時に作成される仮想ユーザ・グループにも複数のロード・ジェネレータが含まれます。
- ▶ すべてのスケジュール設定が保持されます。

第7章

ゴール指向シナリオの作成

テストで達成するゴールを定義することによって、アプリケーションのゴール指向シナリオを作成します。本章では、ゴール指向シナリオの作成方法について説明します。

本章では、次の項目について説明します。

- ▶ ゴール指向シナリオの計画について
- ▶ ゴール指向シナリオの [デザイン] タブについて
- ▶ シナリオのゴールの定義
- ▶ スクリプトへのプロパティの割り当て
- ▶ スクリプトの設定

ゴール指向シナリオの計画について

ゴール指向シナリオでは、テストで達成するゴールを定義すると、このゴールに基づいて、LoadRunner が自動的にシナリオを作成します。ゴール指向のシナリオでは次の5種類のゴール・タイプを指定できます。仮想ユーザ数、秒ごとのヒット数 (Web 仮想ユーザのみ)、秒ごとのトランザクション数、分ごとのページ数 (Web 仮想ユーザのみ)、シナリオが達するトランザクション応答時間。[シナリオ ゴールの編集] ダイアログ・ボックスを使って、いずれかのタイプのシナリオ・ゴールを定義します。このダイアログ・ボックスの詳細については 131 ページ「シナリオのゴールの定義」を参照してください。

注：秒ごとのトランザクションまたはトランザクション応答時間ゴール・タイプを実行するには、スクリプトにトランザクションが含まれている必要があります。これらのゴール・タイプそれぞれに、テストするスクリプトのトランザクションを定義します。

仮想ユーザのゴール・タイプ

アプリケーションが同時に実行できる仮想ユーザ数をテストする場合は、「仮想ユーザ」ゴール・タイプを指定することをお勧めします。このタイプのゴール指向シナリオの実行は、マニュアル・シナリオの実行とほぼ同じです。このゴール・タイプの定義の詳細については、131 ページ「シナリオのゴールの定義」を参照してください。

分ごとのヒット数および秒ごとのヒット数 / トランザクション数のゴール・タイプ

サーバの強度をテストする場合、秒ごとのヒット数、分ごとのページ数または秒ごとのトランザクション数のゴール・タイプを指定することをお勧めします。実行する LoadRunner に、仮想ユーザ数の最小と最大の範囲と、秒ごとのトランザクション数のゴール・タイプのトランザクション名を指定します。

コントローラは最小限の仮想ユーザを使ってゴールを達成しようとします。最小限の仮想ユーザではゴールを達成できない場合、コントローラは指定された最大仮想ユーザ数に達するまで、仮想ユーザ数を増やしていきます。指定された最大仮想ユーザ数でゴールが達成されない場合は、最大仮想ユーザ数を増やしてから再度シナリオを実行します。

分ごとのページ数および秒ごとのヒット数またはトランザクション数のゴール・タイプを実行する際にコントローラで使用される数式の詳細については、136 ページ「秒ごとのヒット数 / トランザクション数ゴール・タイプと分ごとのページ数ゴール・タイプについて」を参照してください。

トランザクション応答時間のゴール・タイプ

指定のトランザクション応答時間を超えずに、同時に実行できる仮想ユーザ数をテストする場合は、トランザクション応答時間のゴール・タイプを指定することをお勧めします。テストするスクリプトのトランザクション名と、実行する LoadRunner の最小および最大仮想ユーザ数の範囲を指定します。トランザ

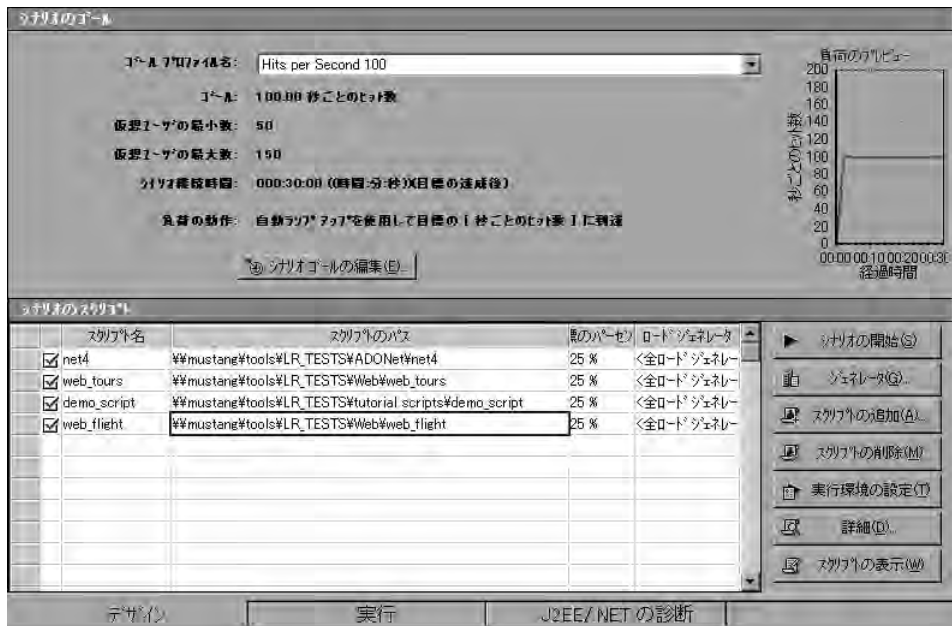
クシヨソ応答時間は、あらかじめ定義されているしきい値に合わせて指定します。例えば、電子商取引サイトへのログインで顧客を5秒以上待たせないようにしたい場合には、最大許容トランザクシヨソ応答時間を5秒に指定します。最小および最大仮想ユーザ数を、同時に処理できるようにしたい最小および最大顧客数に設定します。

シナリオが、定義した最大トランザクシヨソ応答時間に達しなければ、想定している同時サービス顧客数にサーバは妥当な時間で応答できます。仮想ユーザの一部を実行しただけで、定義した応答時間に達した場合、またはコントローラが定義した最大数の仮想ユーザを使えば定義された応答時間を超過するというメッセージを受信した場合は、アプリケーションの改良や、サーバのソフトウェアとハードウェアのアップグレードを検討します。

注：トランザクシヨソ応答時間のゴール指向シナリオを効果的なものにするには、サーバに効果的にヒットするトランザクシヨソを注意深く選択します。

ゴール指向シナリオの [デザイン] タブについて

ゴール指向シナリオを作成するときには、[デザイン] タブに [シナリオのゴール] 表示枠と [シナリオのスクリプト] 表示枠が表示されます。



【シナリオのゴール】表示枠には、ゴール・プロファイルの名前、定義されたゴール、仮想ユーザの最小数と最大数、シナリオの継続時間、負荷の動作など、ゴール・プロファイルに関する情報が表示されます。

ゴール指向のシナリオでは次の5種類のゴール・タイプを指定できます。シナリオに達成させたい仮想ユーザの数、秒ごとのヒット数（Web 仮想ユーザのみ）、秒ごとのトランザクション数、分ごとのページ数（Web 仮想ユーザのみ）、またはトランザクションの応答時間です。ゴール・タイプの定義の詳細については、131 ページの「シナリオゴールの編集」ダイアログ・ボックスを参照してください。

【シナリオのスクリプト】表示枠には、すべての有効および無効な仮想ユーザ・スクリプト、各スクリプトのパス、各スクリプトに割り当てられた総目標の割合のリストが表示されます。「シナリオのスクリプト」表示枠の詳細については、138 ページ「スクリプトへのプロパティの割り当て」を参照してください。

ゴール・プロファイルまたはシナリオに対して、以下のアクションを実行できます。

- ▶ ゴール・プロファイル名とゴール・タイプを定義する

- ▶ 新規のスクリプトをシナリオに追加して設定する
- ▶ 1つまたは複数のロード・ジェネレータ・マシンをスクリプトに追加し、そのロード・ジェネレータを設定する
- ▶ シナリオのスクリプトを有効または無効にする
- ▶ シナリオの継続時間とランプ・アップ動作を定義する
- ▶ シナリオを実行する
- ▶ シナリオを停止する
- ▶ シナリオをリセットする
- ▶ シナリオの実行結果に関する設定を行う

シナリオのゴールの定義

ゴール指向シナリオのシナリオ・ゴールの設定は、[シナリオ ゴールの編集] ダイアログ・ボックスから定義します。

シナリオのゴールの定義は、次の手順で行います。

- 1 [シナリオのゴール] 表示枠で [シナリオ ゴールの編集] ボタンをクリックするか、[シナリオ] > [ゴールの定義] を選択します。[シナリオ ゴールの編集] ダイアログ・ボックスが表示されます。



- 2 [ゴール プロファイル名] を選択します。新しい名前を入力するには、[新規作成] をクリックし、[新規ゴール プロファイル] ダイアログ・ボックスに新しいゴール・プロファイル名を入力した後、[OK] をクリックします。新しいゴール・プロファイル名がセレクトアに表示されます。
- 3 [シナリオ ゴールの編集] ボックスで、[ゴールのタイプ] を選択します。詳細については 133 ページ「[シナリオ ゴールの編集] ダイアログ・ボックスについて」を参照してください。

注： VuGen は、**Init**、**Action**、**End** の各ユニットを、自動的にトランザクションとして定義します。さらに、トランザクション開始関数とトランザクション終了関数を使用して、スクリプトに静的なトランザクションを挿入できます。

- 4 [シナリオの設定] タブを選択して、目標達成後にシナリオの実行を継続する時間の長さ、目標を達成できなかった場合にシナリオの実行を続けるかを指定します。詳細については 135 ページ「[シナリオの設定] タブについて」を参照してください。

- 5 **[負荷の動作]** タブを選択して、コントローラにいつどのように目標を達成させるかを指定します。詳細については136ページ「[負荷の動作] タブについて」を参照してください。
- 6 LoadRunner によるシナリオ実行時にスクリプトに記録された思考遅延時間を使用するには、**[記録された思考遅延時間を変更しない]** を選択します。
- 7 **[OK]** をクリックして、**[シナリオ ゴールの編集]** ダイアログ・ボックスを閉じます。入力したシナリオ・ゴールの情報は、**[シナリオのゴール]** ウィンドウに表示されます。

注：ゴール指向シナリオを実行すると、定義したゴールがシナリオの結果とともに適切なグラフに表示されます。これによって、定義したゴールとシナリオの結果を比較できます。

[シナリオ ゴールの編集] ダイアログ・ボックスについて

[シナリオ ゴールの編集] ダイアログ・ボックスでは、ゴール指向シナリオに関するシナリオ情報を定義できます。

[ゴール プロファイル名]：ゴール・プロファイル名を選択します。

[名前の変更]：[新規ゴール プロファイル] ダイアログ・ボックスを使ってゴール・プロファイル名を変更します。

[削除]：ゴール・プロファイル名のリストから、ゴール・プロファイルを削除します。

[新規作成]：[新規ゴール プロファイル] ダイアログ・ボックスを使って新規のゴール・プロファイル名を入力します。

[シナリオ開始時間]：[シナリオ開始] ダイアログ・ボックスを開きます。ゴール指向シナリオの実行を後で開始できます。

[シナリオ ゴールの定義]

[ゴールのタイプ]：シナリオのゴールのタイプを選択します。

- ▶ 「**分ごとのページ数**」(Web 仮想ユーザのみ) : シナリオの実行で達成する分ごとの目標ダウンロードページ数を入力し、そのシナリオの仮想ユーザの最小数および最大数を選択します。
- ▶ 「**仮想ユーザ**」 : シナリオの実行で達成するゴール仮想ユーザ数を入力します。
- ▶ 「**秒ごとのヒット数**」(Web 仮想ユーザのみ) : シナリオの実行で達成する秒ごとの目標ヒット数(秒ごとの HTTP 要求数)を入力し、そのシナリオの仮想ユーザの最小数および最大数を選択します。
- ▶ 「**秒ごとのトランザクション**」 : シナリオの実行で達成する秒ごとの目標トランザクション数を入力し、そのシナリオの仮想ユーザの最小数および最大数を選択します。さらに、テストするシナリオの静的なスクリプト・トランザクションを選択するか、記録済みの自動スクリプト・トランザクション名を [トランザクション名] ボックスに入力します。
- ▶ 「**トランザクション応答時間**」 : シナリオの実行で達成する目標トランザクション応答時間を入力し、そのシナリオの仮想ユーザの最小数および最大数を選択します。さらに、テストするシナリオの静的なスクリプト・トランザクションを選択するか、記録済みの動的なスクリプト・トランザクション名を [トランザクション名] ボックスに入力します。

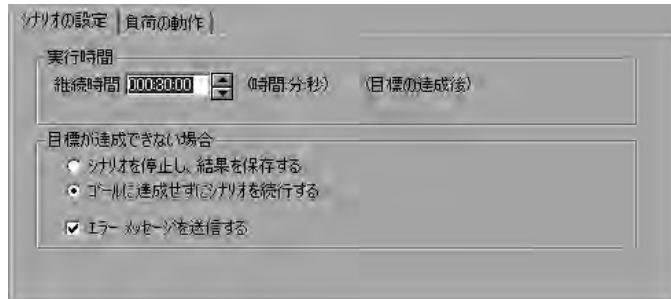
[記録された思考遅延時間を変更しない] : LoadRunner に対して、スクリプトに記録された思考遅延時間を使用してスクリプトを実行するように指示します。

注 : このオプションを選択した場合、目標達成のためにシナリオ中の仮想ユーザ数を増やさなければならないことがあります。

[負荷のプレビュー] : 定義されたゴールと負荷の動作のグラフを表示します。

【シナリオの設定】 タブについて

【シナリオの設定】 タブでは、目標達成後にシナリオの実行を継続する時間の長さ、目標を達成できなかった場合にシナリオの実行を続けるかどうかを指定できます。



【実行時間】

【**継続時間（時間：分：秒）（目標の達成後）**】：目標達成後にシナリオの実行を継続する時間の長さを選択します。

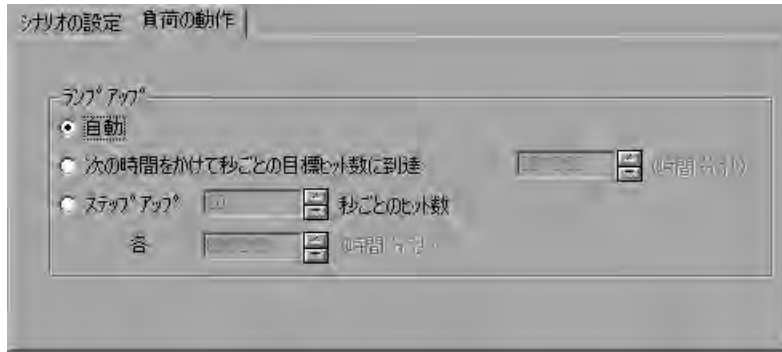
【**目標が達成できない場合**】：次の2つのオプションのどちらかを選択します。

- ▶ 【**シナリオを停止し、結果を保存する**】：コントローラに対して、定義された目標を達成できなかった場合に、シナリオを停止して実行結果を保存するように指示します。
- ▶ 【**ゴールを達成せずにシナリオを続行する**】：コントローラに対して、定義された目標を達成できない場合でも、シナリオの実行を継続するように指示します。

【**エラーメッセージを送信する**】：コントローラに対して、定義された目標を達成できなかった場合に、エラー・メッセージを送信するように指示します。

[負荷の動作] タブについて

[負荷の動作] タブでは、コントローラにいつどのように目標を達成させるかを指定できます。



[ランプアップ]：次のオプションのいずれかを選択します。

- ▶ [自動]：コントローラに対して、1つのバッチで標準設定の数の仮想ユーザを実行するように指示します（2分ごとに、50の仮想ユーザ、または定義された最大仮想ユーザ数が50より少ない場合はすべての仮想ユーザ）。
- ▶ [次の時間をかけて秒ごとの目標ヒット数に到達]：コントローラが目標を達成するまでに、シナリオの実行を継続する時間の長さを選択します。
- ▶ [ステップアップ]：（秒ごとのトランザクション数ゴール・タイプとトランザクション応答時間ゴール・タイプでは使用できません。）目標を達成するまで徐々に増加させる量（時間 x ごとに仮想ユーザ数 / ページ数 / ヒット数を x ずつ）を選択します。

秒ごとのヒット数 / トランザクション数ゴール・タイプと分ごとのページ数ゴール・タイプについて

ゴール・タイプとして「分ごとのページ数」または「秒ごとのヒット数 / トランザクション」を指定した場合、コントローラは指定された目標を指定された最小仮想ユーザ数で割って、各仮想ユーザが達成するべき「分ごとのページ数」または「秒ごとのヒット / トランザクション数」を決定します。[負荷の動作] の設定にしたがって次のように仮想ユーザの実行を開始します。

- ▶ 仮想ユーザの自動実行を選択した場合、LoadRunner は最初のバッチで 50 の仮想ユーザの実行を開始します。定義した仮想ユーザの最大数が 50 未満の場合は、LoadRunner はすべての仮想ユーザを同時に実行開始します。

- ▶ 一定の時間でシナリオに目標を達成させることを選択した場合は、LoadRunnerはその時間内に定義した目標を達成しようとします。そして、指定された制限時間と、仮想ユーザごとの目標ヒット数、トランザクション数、またはページ数の計算結果に基づいて、最初の実行する仮想ユーザのバッチ・サイズを決定します。
- ▶ 徐々に目標に達するように選択した場合（時間 x ごとに x 個のページ数 / ヒット数）、LoadRunnerは仮想ユーザごとの目標ヒット / ページ数を計算し、それに基づいて最初のバッチの仮想ユーザ数を決定します。

注：最後の負荷の動作オプションは、秒ごとのトランザクション数のゴール・タイプでは使用できません。

仮想ユーザのバッチを実行するたびに、LoadRunnerはバッチの目標が達成されたかどうか評価します。バッチ目標が達成されていない場合は、LoadRunnerは定義されたゴールを達成できるように、仮想ユーザごとの目標ヒット数、トランザクション数、またはページ数を再計算し、次のバッチの仮想ユーザ数を再調整します。標準設定では、新しい仮想ユーザのバッチは2分ごとに開始されます。

コントローラが定義した最大数の仮想ユーザを起動してもゴールを達成できない場合、LoadRunnerは仮想ユーザごとの目標ヒット数、トランザクション数、またはページ数を再計算し、最大数の仮想ユーザを同時に実行して、もう一度定義された目標の達成を試みます。

次に該当する場合には、分ごとのページ数または秒ごとのヒット / トランザクション数のゴール指向シナリオには「失敗」ステータスが割り当てられます。

- ▶ コントローラが、指定した最大数の仮想ユーザを使用して目標を達成しようと2回試みて、2回とも目標が達成されなかった場合。
- ▶ 仮想ユーザの最初のバッチが実行された後、分ごとのページ数または秒ごとのヒット / トランザクション数が記録されなかった場合。
- ▶ コントローラが仮想ユーザのバッチをいくつか実行した後、分ごとのページ数または秒ごとのヒット / トランザクション数が増加しなかった場合。
- ▶ 実行した仮想ユーザがすべて失敗した場合。
- ▶ 実行しようとした仮想ユーザ・タイプで利用可能なロード・ジェネレータがなかった場合。

スクリプトへのプロパティの割り当て

[シナリオのスクリプト] 表示枠には、そのシナリオに対して選択されたスクリプトのリストが表示されます。

スクリプト名	スクリプトのパス	のパーセント	ジェネレータ
<input checked="" type="checkbox"/> net4	##mustang%tools%LR_TESTS%ADONet%net4	25 %	<全ロードジェネレータ>
<input checked="" type="checkbox"/> web_tours	##mustang%tools%LR_TESTS%Web%web_tours	25 %	<全ロードジェネレータ>
<input checked="" type="checkbox"/> demo_script	##mustang%tools%LR_TESTS%demo_script	25 %	<全ロードジェネレータ>
<input checked="" type="checkbox"/> web_flight	##mustang%tools%LR_TESTS%Web%web_flight	25 %	<全ロードジェネレータ>

[**目標のパーセント率**] カラムには、各仮想ユーザ・スクリプトに自動的に割り振られた仮想ユーザ数、分ごとのページ数、秒ごとのヒット数、秒ごとのトランザクション数、またはトランザクション応答時間の、全目標に対する割合が表示されます。[ロードジェネレータ] カラムには、各仮想ユーザ・スクリプトに対して自動的に<全ロードジェネレータ>が表示されます。

スクリプトに割り当てる仮想ユーザの割合の変更は、次の手順で行います。

スクリプトの [**目標のパーセント率**] カラムに、LoadRunner に対してシナリオ実行中に到達するように指示する仮想ユーザ数、分ごとのページ数、秒ごとのヒット数、秒ごとのトランザクション数、またはトランザクション応答時間の、全目標数に対する割合を入力します。シナリオ実行中に、LoadRunner は、シナリオ内の各スクリプトで指定された割合に到達することを試みます。

スクリプトのロード・ジェネレータの追加または変更は、次の手順で行います。

- 1 スクリプトの [ロードジェネレータ] カラムで、ロードジェネレータ名のリストから1つ以上のマシンを選択し、[OK] をクリックします。複数のマシンを選択した場合、スクリプトに割り当てた仮想ユーザがロード・ジェネレータ間で均等に分配されます。
- 2 また、[追加] を選択して、リストにロード・ジェネレータを追加することもできます。[新規ロードジェネレータの追加] ダイアログ・ボックスが表示さ

れます。



- 3 [名前] ボックスにロード・ジェネレータの名前を入力します。
- 4 [プラットフォーム] ボックスで、ロード・ジェネレータを実行するプラットフォームのタイプを選択します。
- 5 [一時ディレクトリ] ボックスに、一時ファイルを保存する場所としてロード・ジェネレータ上の場所を入力するか、標準設定の場所を使用するためにボックスを空白のままにします。標準設定では、LoadRunner はシナリオ実行中、ロード・ジェネレータの環境変数 TEMP または TMP で指定された一時ディレクトリに一時ファイルを格納します。
- 6 ロード・ジェネレータをシナリオに参加できるようにするには、**[ロードジェネレータのシナリオ参加を有効にする]** を選択します。
- 7 **[詳細表示]** をクリックして、ダイアログ・ボックスを拡張すると [新規ロードジェネレータの追加] ダイアログ・ボックスにいくつかのタブが表示されます。各ロード・ジェネレータの設定の詳細については、79 ページ「ロード・ジェネレータの追加設定」を参照してください。
- 8 **[OK]** をクリックして、[新規ロードジェネレータの追加] ダイアログ・ボックスを閉じます。LoadRunner は、ロード・ジェネレータ名のリストに新しいロード・ジェネレータを追加します。シナリオに新しいロード・ジェネレータを含めるには、ロード・ジェネレータ名のリストから対象のロード・ジェネレータを選択して、**[OK]** をクリックします。

注：複数のロード・ジェネレータを選択できます。

シナリオに追加する各ロード・ジェネレータに対して上記の手順を繰り返します。

ロード・ジェネレータの設定

ロード・ジェネレータの属性の設定は、[ロード ジェネレータ] ダイアログ・ボックスを使って、ロード・ジェネレータ・リストにロード・ジェネレータを追加しているときに設定できます。また既存のロード・ジェネレータの属性は随時変更できます。また、[ロード ジェネレータ] ダイアログ・ボックスを使って、シナリオで仮想ユーザを実行するロード・ジェネレータを指定できます。例えば、あるシナリオの実行にロード・ジェネレータが使えない場合は、[ロード ジェネレータ] ダイアログ・ボックスを使って、そのロード・ジェネレータをロード・ジェネレータのリストから完全に削除してしまうのではなく、一時的に除外できます。[ロード ジェネレータ] ダイアログ・ボックスを使用する手順については、74 ページ「ロード・ジェネレータの設定」を参照してください。ロード・ジェネレータの設定の追加については、79 ページ「ロード・ジェネレータの追加設定」を参照してください。

シナリオに参加しているすべてのロード・ジェネレータのグローバル設定を構成するには、LoadRunner の [オプション] ダイアログ・ボックスを使います。詳細については、第 10 章「シナリオの設定」を参照してください。

負荷分散

負荷分散により、仮想ユーザにより生成された負荷が、要求ロード・ジェネレータ・マシンに均等に分散され、負荷テストが正確になります。

Windows ロード・ジェネレータ・マシンの CPU の使用が過負荷状態になると、コントローラは、そのロード・ジェネレータの仮想ユーザのロードを停止し、シナリオで指定された他のロード・ジェネレータに仮想ユーザを自動的に分散します。シナリオ内に他のロード・ジェネレータがない場合だけ、コントローラは仮想ユーザのロードを停止します。

[ロード ジェネレータ] ダイアログ・ボックス内のアイコンを使用して、マシンの CPU 使用率のステータスを監視できます。ロード・ジェネレータの CPU 使用率に問題が生じると、ロード・ジェネレータ名の左側にあるアイコンに黄色の線が表示されます。マシンが過負荷になると、アイコンに赤色の線が表示されます。

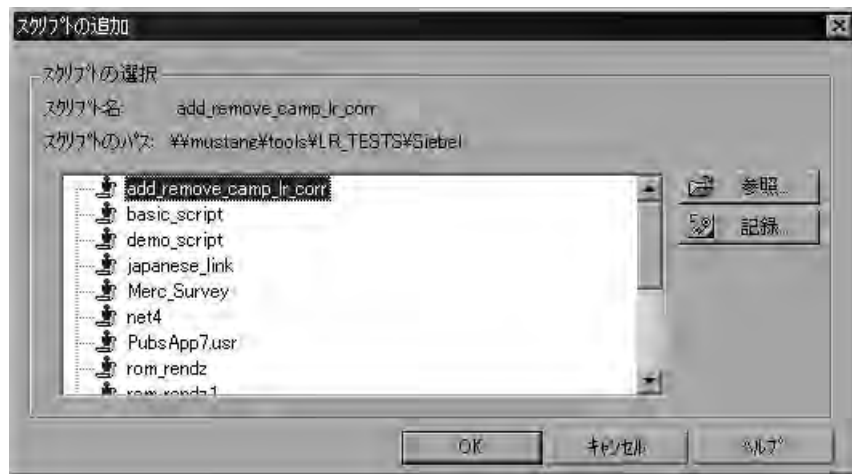
注：負荷分散が行えるのは、ゴール指向シナリオとパーセント・モードのマニュアル・シナリオだけです。

スクリプトの設定

[スクリプトの追加] ダイアログ・ボックスを使って、[シナリオのスクリプト] リストにスクリプトを追加できます。リストに追加したスクリプトは、選択して詳細表示したり、編集したり、実行環境の設定を変更したりできます。

スクリプトの追加は、次の手順で行います。

- 1 [シナリオのスクリプト] 表示枠の右にある [スクリプトの追加] ボタンをクリックするか、カラム内で右クリックし、[スクリプトの追加] を選択します。[スクリプトの追加] ダイアログ・ボックスが表示されます。



- 2 [参照] ボタンをクリックします。[テストを開く] ダイアログ・ボックスが表示されます。

新しいスクリプトのパスとファイル名を選択します。VB 仮想ユーザ・スクリプトを選択するには、**.usr** ファイルを探します。

注：スクリプトの場所を指定する場合、現在のシナリオ・ディレクトリに相対する場所を指定できます。詳細については、108 ページ「スクリプトに対する相対パスの使用」を参照してください。


- 3 [開く] をクリックして、ファイルを選択します。[テストを開く] ダイアログ・ボックスが閉じ、[スクリプトの追加] ダイアログ・ボックスに新しいスクリプト名が表示されます。
- 4 [OK] をクリックして、[スクリプトの追加] ダイアログ・ボックスを閉じた後、[シナリオのスクリプト] 表示枠に新しいスクリプト情報を入力します。

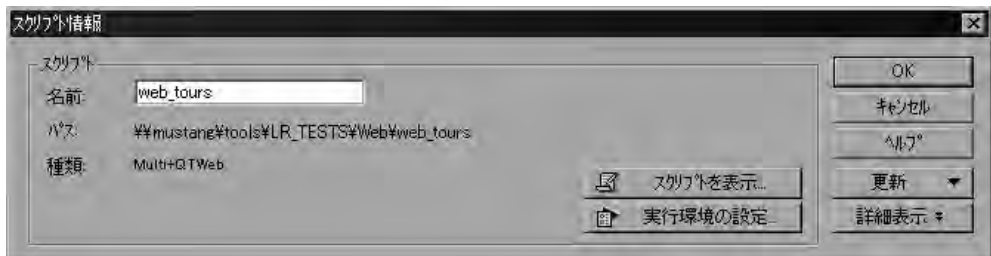
注：スクリプトのランデブー・ポイントはゴール指向シナリオでは無効です。

スクリプト情報の表示

リストに追加したスクリプトは、選択して詳細を表示したり、編集したり、有効化または無効化したり、実行環境の設定を変更したりできます。

スクリプトの詳細表示は、次の手順で行います。

-  1 [シナリオのスクリプト] 表示枠の右にある [詳細] ボタンをクリックするか、スクリプトを右クリックして [詳細] を選択します。[スクリプト情報] ダイアログ・ボックスが表示され、選択したスクリプトの「パス」、「名前」、「種類」が表示されます。



- 2 [実行環境の設定] をクリックして、スクリプトの実行環境の設定（オプション）を指定できます。これによって、コントローラで仮想ユーザ・スクリプト

を実行する方法をカスタマイズできます。[実行環境の設定] ダイアログ・ボックスが表示され、以前に **VuGen** を使って設定した設定値が表示されます。**VuGen** でスクリプトの実行環境を設定していない場合、[ログ] タブと [思考遅延時間] タブを除いて、すべてのタブにデフォルトの **VuGen** 設定が表示されます。[ログ] タブと [思考遅延時間] タブには、標準のコントローラ設定が表示されます。

注： Web および Java などのいくつかのプロトコルには、固有の設定があります。

実行環境の設定の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド**』を参照してください。

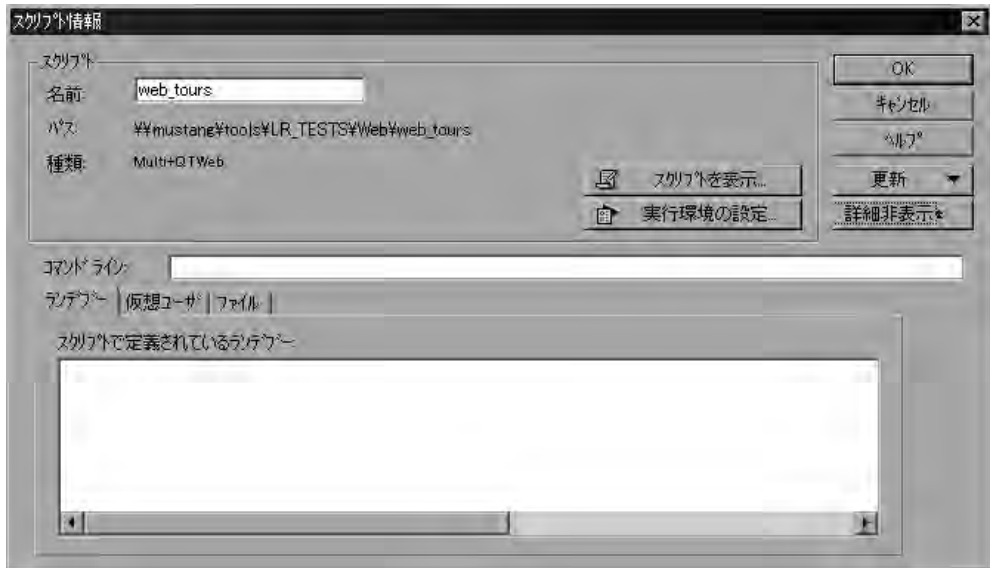
注： コントローラで実行環境の設定を変更すると、LoadRunner は変更された設定を使ってスクリプトを実行します。初期設定に戻すには、[更新] ボタンをクリックし、[実行環境の設定] を選択します。

- 3 スクリプトを編集するには、[スクリプトを表示] をクリックします。スクリプト作成ツール、**VuGen** が起動します。スクリプトの編集の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド**』を参照してください。

注： コントローラの実行中に **VuGen** を使ってスクリプトに変更を加える場合、[更新] ボタンをクリックして [スクリプト] を選択し、シナリオ内のスクリプトの詳細を更新します。

- 4 [詳細表示] をクリックして、[スクリプト情報] ダイアログ・ボックスを拡張

すると、さらに多くのスクリプト情報が表示されます。



- 5 [コマンドライン] ボックスに、スクリプトの実行時に使用するコマンド・ライン・オプションをすべて入力します。例を次に示します。

-x value -y value

コマンド・ライン引数の値をスクリプトに渡す方法の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザーズ・ガイド』を参照してください。

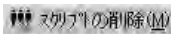
- 6 選択されたスクリプト内に含まれるランデブー・ポイントを参照するには、[ランデブー] タブをクリックします。
- 7 選択されたスクリプトに関連している仮想ユーザのリストを参照するには、[仮想ユーザ] タブをクリックします。まだ仮想ユーザを作成していない場合には、空のボックスが表示されます。
- 8 スクリプトによって使用されるファイルのリストを参照するには、[ファイル] タブをクリックします。標準設定では、このリストにはスクリプトのディレクトリ内のファイルがすべて表示されます（スクリプトがスクリプト・リストに追加された後のみ）。これらのファイルには、構成設定ファイル、スクリプトの `init`、`run`、および `end` 部分、パラメータ化定義ファイル、`.usr` ファイルが含まれます。このリストにファイルを追加するには、[追加] をクリックし、

ファイル名を追加します。

注：追加したファイルは削除できますが、それ以外の表示されているファイルは削除できません。

9 **[OK]** をクリックして、**[スクリプト情報]** ダイアログ・ボックスを閉じます。

スクリプトの削除は、次の手順で行います。



[シナリオのスクリプト] 表示枠の右にある **[スクリプトの削除]** ボタンをクリックするか、削除するスクリプトを右クリックして **[スクリプトの削除]** を選択します。

スクリプトの無効化は、次の手順で行います。

仮想ユーザ・スクリプトの名前の左側にあるボックスをクリックします。これによって、スクリプトエントリの色がグレーに変化します。これは、スクリプトがシナリオに参加しないことを意味します。仮想ユーザ・スクリプトを再度有効にするには、同じボックスをもう一度クリックします。

第 8 章

シナリオのスケジュール

シナリオ作成後、シナリオの実行を開始する時間を設定できます。また、マニュアル・シナリオについては、シナリオまたはシナリオ内の仮想ユーザ・グループの継続時間を設定できます。さらに、シナリオまたは仮想ユーザ・グループの仮想ユーザを徐々に実行または停止できます。

注：パーセント・モードでは仮想ユーザ・グループの設定は適用されません。

本章では、次の項目について説明します。

- ▶ シナリオのスケジュールについて
- ▶ シナリオの開始の遅延
- ▶ スケジュールの選択
- ▶ シナリオのスケジュール
- ▶ 仮想ユーザ・グループのスケジュール
- ▶ スケジュールされたシナリオへの仮想ユーザの追加

シナリオのスケジュールについて

シナリオ作成の重要な要素は、ユーザの振る舞い（仮想ユーザ・スクリプトで表されるアクションの種類とそのタイミング）を正確に演じるテストを開発することです。

[シナリオ開始] ダイアログ・ボックスを使って、LoadRunner にシナリオの開始を遅延させることができます。**実行**コマンドを発行してから LoadRunner を待機させる時間 (分)、またはシナリオを開始する時刻を指定できます。

スケジュール・ビルダを使って、マニュアル・シナリオの時間にかかわる要素を設定し、シナリオまたはシナリオの仮想ユーザ・グループの実行継続時間を制限できます。シナリオまたは仮想ユーザ・グループを [実行中] ステータスに留めておく時間 (分) を指定することにより、実行継続時間を制限します。シナリオまたはグループが制限時間に達すると、実行を終了します。

注：パーセント・モードでは仮想ユーザ・グループの設定は適用されません。

マニュアル・シナリオの場合、LoadRunner がある時間範囲内に開始および停止する仮想ユーザの数を指定できます。LoadRunner が、シナリオまたは仮想ユーザ・グループのすべての仮想ユーザを同時に開始または停止するか、指定した時間内に特定の数の仮想ユーザだけを開始または停止するかを指定します。

定義したスケジュールは視覚化され、[負荷のプレビュー] グラフに表示されます。

注：仮想ユーザ・スクリプトのランデブー・ポイントは、スケジュールが設定されたシナリオに干渉します。スクリプトにランデブー・ポイントが含まれている場合、シナリオはスケジュールどおりに実行されません。

シナリオの開始の遅延

マニュアル・シナリオでもゴール指向シナリオでも、LoadRunner にシナリオの実行開始を遅延させることができます。**実行**コマンド発行してから LoadRunner を待機させる時間 (分)、またはシナリオを開始する時刻を指定できます。

シナリオの開始を遅らせる設定は、次の手順で行います。

- 1 **[シナリオ]** > **[開始時間]** を選択します。**[シナリオ開始]** ダイアログ・ボックスが表示されます。標準設定では**[遅延なし]** が選択されています。



- 2 **[遅延間隔 X (時間:分:秒)]** を選択して、シナリオ開始を遅らせる時間の長さ（時間:分:秒の形式）を入力します。
あるいは、**[X (時:分:秒) 日付]** を選択し、シナリオを開始する時刻（時間:分:秒の形式）と日付を指定することもできます。
- 3 **[OK]** をクリックして、ダイアログ・ボックスを閉じると、設定が保存されます。

[シナリオ開始] ダイアログ・ボックスについて

[シナリオ開始] ダイアログ・ボックスを使って、シナリオの開始時間を遅らせることができます。

[シナリオ開始] : 次のオプションのいずれかを選択します。

- ▶ **[遅延なし]** : **[シナリオ開始]** ボタンをクリックすると、すぐにシナリオが開始されます。
- ▶ **[遅延間隔 X (時間:分:秒)]** : 指定した時間が経過するとシナリオが開始されます。
- ▶ **[X (時:分:秒) 日付]** : 指定した日時にシナリオが開始されます。

注：スケジュール・ビルダを使用して、ランプ・アップ・スケジュールと、シナリオまたは仮想ユーザ・グループの継続時間を設定します。

スケジュールの選択

[シナリオのスケジュール] 表示枠の [スケジュール名] ボックスで、マニュアル・シナリオに使用するスケジュールを選択します。既存のスケジュール（[徐々にランプアップ] または [ランプアップ]）を選択することも、[<新規スケジュール>] を選択し、スケジュール・ビルダで新しいプロパティを持つスケジュールを作成することもできます。

スケジュール・ビルダを使って、3つの既存のスケジュールのプロパティを変更することもできます。

新規スケジュールの作成は、次の手順で行います。

- 1 [シナリオのスケジュール] 表示枠の [スケジュール名] ボックスから [**<新規スケジュール>**] を選択します。[新規スケジュール] ダイアログ・ボックスが表示されます。

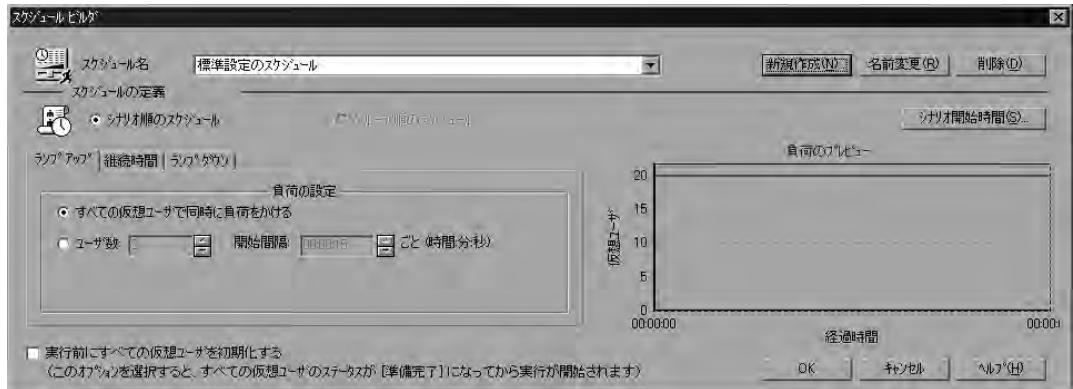


- 2 [名前] テキスト・ボックスに、新規スケジュールの名前を入力し、[OK] をクリックします。[スケジュールビルダ] ダイアログ・ボックスが表示されます。

既存のスケジュールのプロパティの変更は、次の手順で行います。

- 1 [デザイン] タブの [シナリオのスケジュール] 表示枠にある [スケジュール名] ボックスから、[徐々にランプアップ] または [ランプアップ] のいずれかを選択します。

- 2 [シナリオ] > [スケジュールビルダ] を選択するか、[スケジュールの編集] ボタンをクリックします。[スケジュールビルダ] ダイアログ・ボックスが表示されます。



スケジュール名を変更するには、[名前変更] をクリックします。表示されたダイアログ・ボックスに、新しい名前を入力します。スケジュールを削除するには、[削除] をクリックします。

[スケジュールビルダ] ダイアログ・ボックスについて

[スケジュールビルダ] ダイアログ・ボックスを使用して、シナリオのスケジュール設定が行えます。

[スケジュール名]：シナリオに使用するスケジュール名を選択します。「標準設定のスケジュール」、「ランプアップ」、「徐々にランプアップ」の3つの標準設定の名前が表示されます。「ランプアップ」は仮想ユーザを一定の割合で段階的に開始します。「徐々にランプアップ」は、仮想ユーザをより遅いペースで段階的に開始します。

[新規作成]：[新規スケジュール] ダイアログ・ボックスを開きます。新しいスケジュール名を入力できます。

[名前変更]：スケジュールの名前を変更します。

[削除]：スケジュール名を削除します。

[シナリオ開始時間] : [シナリオの開始] ダイアログ・ボックスを開きます。このダイアログ・ボックスを使用して、後からマニュアル・シナリオおよびゴール指向シナリオを実行できます。

[スケジュールの定義]

- ▶ **[シナリオ順のスケジュール]** : シナリオ全体の設定を定義します。
 - ▶ **[ランプアップ] タブ** : シナリオの開始方法を定義します。
 - ▶ **[継続時間] タブ** : シナリオの継続時間を設定します。
 - ▶ **[ランプダウン] タブ** : シナリオの停止方法を定義します。
- ▶ **[グループ順のスケジュール]** : 個々のグループの設定を定義します。左側のボックスから、スケジューリングする仮想ユーザ・グループを選択します。
 - ▶ **[開始時間] タブ** : 仮想ユーザ・グループの開始時間の設定方法を定義します。
 - ▶ **[ランプアップ] タブ** : 仮想ユーザ・グループの開始方法を定義します。
 - ▶ **[継続時間] タブ** : 仮想ユーザ・グループの継続時間を設定します。
 - ▶ **[ランプダウン] タブ** : 仮想ユーザ・グループの停止方法を定義します。

注 : パーセント・モードでは仮想ユーザ・グループの設定は適用されません。

[実行前にすべての仮想ユーザを初期化する] : 負荷をかけ始める前に仮想ユーザを初期化するように LoadRunner に指示します。仮想ユーザの実行は、仮想ユーザが「準備完了」ステータスになった後にだけ開始されます。

[負荷のプレビュー] : 定義したシナリオのスケジュールのグラフを表示します。

注 : 仮想ユーザ・スクリプトのランデブー・ポイントは、スケジュールが設定されたシナリオに干渉します。スクリプトにランデブー・ポイントが含まれている場合、シナリオはスケジュールどおりに実行されません。

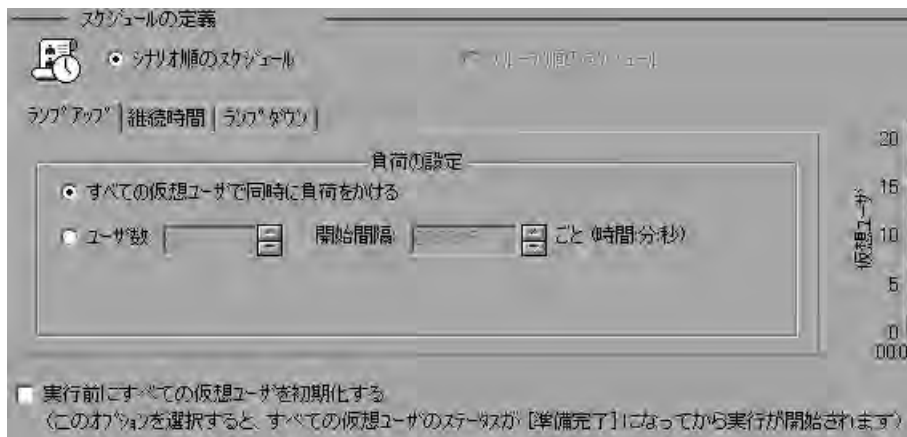
シナリオのスケジュール

スケジュール・ビルダを使って、シナリオの実行の次のような制御ができます。

- ▶ シナリオの継続時間の制限
- ▶ シナリオの仮想ユーザのランプ・アップ（徐々に実行）
- ▶ シナリオの仮想ユーザのランプ・ダウン（徐々に停止）

シナリオのスケジュール・オプションの設定は、次の手順で行います。

- 1 [シナリオ順のスケジュール] オプションを選択します。



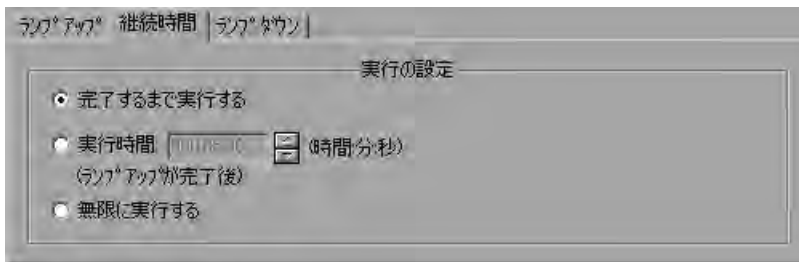
- 2 シナリオを開始する方法を指定するには、[ランプアップ] タブをクリックします。次のオプションのいずれかを選択します。
 - ▶ [すべての仮想ユーザで同時に負荷をかける]：シナリオ内のすべての仮想ユーザを同時に開始します。
 - ▶ [ユーザ数：X 開始間隔：X ごと（時間：分：秒）]：指定された数の仮想ユーザを同時に開始し、仮想ユーザのランプ・アップの開始間隔として指定された時間待機します。

注：シナリオ実行中に、仮想ユーザ・グループまたは仮想ユーザ・スクリプトをシナリオに追加し、それらを有効にできます。仮想ユーザを徐々に開始するランプ・アップ・モードでも、シナリオ内のすべての仮想ユーザがランプ・アップされた後に仮想ユーザ・グループまたはスクリプトを追加した場合には、新しいグループまたはスクリプトの実行が一斉に開始されます。

- 3 LoadRunner に、仮想ユーザのロード前にそれらを初期化させるには、**[実行前にすべての仮想ユーザを初期化する]** を選択します。

注：LoadRunner が仮想ユーザの実行を開始するのは、すべての仮想ユーザが**[準備完了]** ステータスに達した後になります。

- 4 シナリオの継続時間を設定するには、**[継続時間]** タブを選択します。

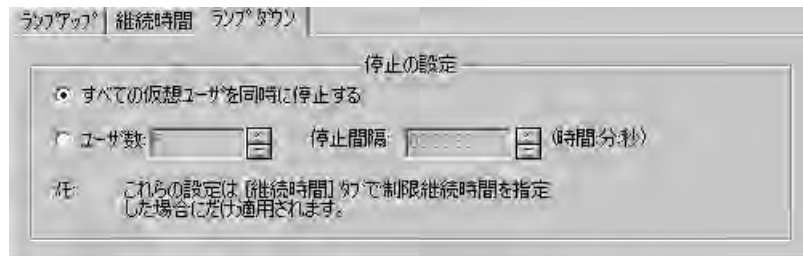


次のオプションのいずれかを選択します。

- ▶ **[完了するまで実行する]**
- ▶ **[実行時間 : X (時間 : 分 : 秒) (ランプアップが完了後)]** : すべての仮想ユーザがランプ・アップされた後、指定された時間だけシナリオを実行します。
- ▶ **[無限に実行する]**

注：継続時間の設定は、仮想ユーザの反復設定に優先します。つまり、継続時間が5分に設定されている場合、実行環境の設定で1回の反復しか指定されていなくても、仮想ユーザは5分に達するまで、実行を反復し続けます。継続時間が制限されているシナリオでは、すべての仮想ユーザが「初期化」ステータスに到達してから継続時間が始まります。初期化に時間のかかる仮想ユーザの場合、ステータスが「実行」にならないままシナリオが終了することがあります。シナリオ内のすべての仮想ユーザが実行されるようにするには、**[実行前にすべての仮想ユーザを初期化する]** チェック・ボックスを選択します。継続時間は、すべての仮想ユーザが「実行」ステータスに到達した場合にのみ始まります。

- 5 シナリオを停止する方法を指定するには、**[ランプダウン]** タブをクリックします。



次のオプションのいずれかを選択します。

- ▶ **[すべての仮想ユーザを同時に停止する]**：シナリオ内のすべての仮想ユーザを同時に停止します。
- ▶ **[ユーザ数：X 停止間隔：X（時間：分：秒）]**：指定した時間ごとに指定した数の仮想ユーザを停止します。

注：[ランプダウン] タブの設定は、[継続時間] タブで2番目のオプションを選択した場合にだけ適用されます。

- 6 **[OK]** をクリックして、スケジュール・ビルダを閉じると、設定が保存されます。

仮想ユーザ・グループのスケジュール

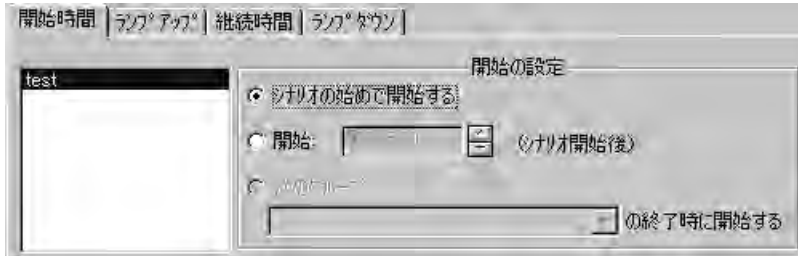
仮想ユーザ・グループの作成後、次の項目を設定して、グループのスクリプト実行スケジュールを設定できます。

- ▶ シナリオ開始後、グループが実行開始を待つ時間の長さ
- ▶ 指定された時間内に実行する仮想ユーザ数
- ▶ 指定された時間内に停止する仮想ユーザ数
- ▶ グループが実行を続ける時間の長さ

注：パーセント・モードでは仮想ユーザ・グループの設定は適用されません。

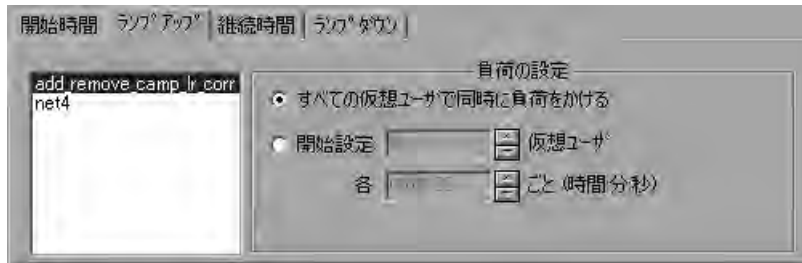
仮想ユーザ・グループのスケジュールの作成は、次の手順で行います。

- 1 [グループ順のスケジュール] オプションを選択します。標準設定では、[開始時間] タブが表示されます。



- 2 左のボックスから対象グループを選択します。
- 3 グループの開始時刻を設定するには、次の3つのオプションからいずれかを選択します。
 - ▶ [シナリオの始めで開始する]
 - ▶ [開始 : X (シナリオ開始後)] : グループ実行前に、指定された時間待機します。
 - ▶ [次のグループ X の終了時に開始する] : 指定したグループの実行終了後にグループの実行を開始します。

- 4 グループのランプ・アップを設定するには、[ランプアップ] タブをクリックします。



次のオプションのいずれかを選択します。

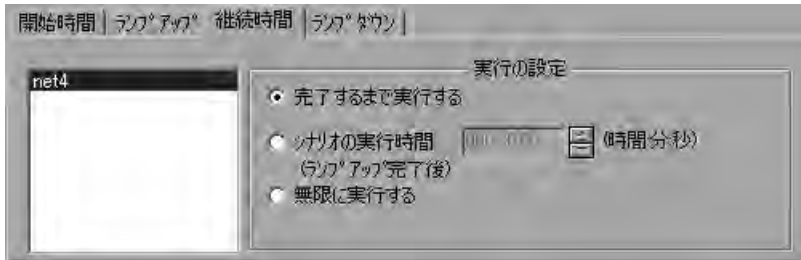
- ▶ **[すべての仮想ユーザで同時に負荷をかける]**：グループ内のすべての仮想ユーザを同時に開始します。
- ▶ **[開始設定：Xの仮想ユーザ/各Xごと(時間:分:秒)]**：指定された数の仮想ユーザを同時に開始し、仮想ユーザのランプ・アップの開始間隔として指定された時間待機します。

注：シナリオ実行中に、シナリオに仮想ユーザ・グループを追加して実行できます。仮想ユーザを徐々に開始するランプ・アップ・モードでも、シナリオ内のすべての仮想ユーザがランプ・アップされた後に仮想ユーザ・グループまたはスクリプトを追加した場合には、新しいグループまたはスクリプトの実行が一斉に開始されます。

- 5 LoadRunner に、仮想ユーザのロード前にそれらを初期化させるには、**[実行前にすべての仮想ユーザを初期化する]** を選択します。

注：LoadRunner が仮想ユーザの実行を開始するのは、すべての仮想ユーザが [準備完了] ステータスに達した後になります。

6 グループの継続時間を設定するには、**「継続時間」** タブをクリックします。



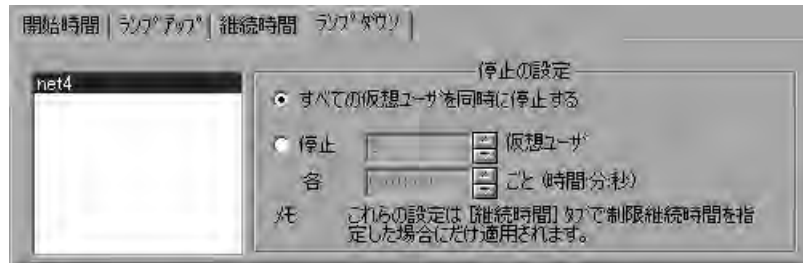
次のオプションのいずれかを選択します。

▶ **「完了するまで実行する」**

▶ **「シナリオの実行時間 X (時間 : 分 : 秒) (ランプアップ完了後)」** : すべての仮想ユーザがランプ・アップされた後、指定された時間だけグループを実行します。

注 : 継続時間の設定は、仮想ユーザの反復設定に優先します。つまり、継続時間が5分に設定されている場合、実行環境の設定で1回の反復しか指定されていなくても、仮想ユーザは5分に達するまで、実行を反復し続けます。継続時間が制限されているシナリオでは、すべての仮想ユーザが「初期化」ステータスに到達してから継続時間が始まります。初期化に時間のかかる仮想ユーザの場合、ステータスが「実行」にならないままシナリオが終了することがあります。シナリオ内のすべての仮想ユーザが実行されるようにするには、**「実行前にすべての仮想ユーザを初期化する」** チェック・ボックスを選択します。継続時間は、すべての仮想ユーザが「実行」ステータスに到達した場合にのみ始まります。

- 7 仮想ユーザ・グループを停止する方法を指定するには、[ランプダウン] タブをクリックします。



次のオプションのいずれかを選択します。

- ▶ **[すべての仮想ユーザを同時に停止する]**：グループ内のすべての仮想ユーザを同時に停止します。
- ▶ **[停止：X 仮想ユーザ：各 X ごと (時間：分：秒)]**：指定した時間ごとに指定した数の仮想ユーザを停止します。

注：[ランプダウン] タブの設定は、[継続時間] タブで2番目のオプションを選択した場合にだけ適用されます。

- 8 [OK] をクリックして、スケジュール・ビルダを閉じると、設定が保存されます。

スケジュールされたシナリオへの仮想ユーザの追加

スケジュール・ビルダの設定を使って、シナリオまたは仮想ユーザ・グループを実行すると、これらの設定は、シナリオ実行中にシナリオまたは仮想ユーザ・グループに手作業で追加されたすべての仮想ユーザに適用されます。例えば、実行中のシナリオまたは仮想ユーザ・グループに設定された継続時間が5分ある場合、それ以降にシナリオまたは仮想ユーザ・グループに追加されたすべての仮想ユーザは、残り時間の間だけ実行されます。

スケジュールされたシナリオまたは仮想ユーザ・グループに追加された仮想ユーザは、実行を完了しても、スケジュール・ビルダの設定には影響されず、シナリオの実行時間設定に従って実行されます。

第2部・シナリオの設計

手作業で制御される仮想ユーザの詳細については、232 ページ「実行中のシナリオへの手作業による仮想ユーザの追加」を参照してください。

第 9 章

ランデブー・ポイントの使用

LoadRunner では、指定した負荷をかけた状態でのシステムの応答を検査できます。そのために、**ランデブー・ポイント**を使って複数の仮想ユーザにまったく同時にタスクを実行させることにより、サーバ上に大きなユーザ負荷を生成できます。

本章では、次の項目について説明します。

- ▶ ランデブー・ポイントの使用について
- ▶ ランデブー属性の設定
- ▶ ランデブー・ポリシーの設定
- ▶ ランデブー・ポイントを有効 / 無効にする
- ▶ ランデブー・ポイントの仮想ユーザを有効 / 無効にする
- ▶ ランデブー情報の表示

ランデブー・ポイントの使用について

シナリオの実行中、ランデブー・ポイントを使って複数の仮想ユーザにタスクを同時に実行させることができます。ランデブー・ポイントでサーバに大きなユーザ負荷を生成し、LoadRunner で負荷がかかった状態のサーバのパフォーマンスを測定できます。

10 人の仮想ユーザが同時に口座情報を照会するときの、Web ベースのバンキング・システムのパフォーマンスを測定するものとします。サーバ上で必要なユーザ負荷をエミュレートするために、すべての仮想ユーザに、まったく同じタイミングで口座情報を照会させます。

「**ランデブー・ポイント**」を作成して、複数の仮想ユーザが確実に同時に実行するようにします。仮想ユーザがランデブー・ポイントに到着すると、コント

ローラはその仮想ユーザをランデブー・ポイントで待機させます。コントローラは、必要な数の仮想ユーザが到着するか、指定の時間が経過すると、ランデブー・ポイントから仮想ユーザを解放します。解放の基準については、164 ページ「ランデブー・ポリシーの設定」を参照してください。

ランデブー・ポイントは仮想ユーザ・スクリプトで定義します。仮想ユーザ・スクリプトにランデブー・ポイントを挿入する方法については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

コントローラを使い、次の選択を行って、サーバ負荷のレベルを変更できます。

- ▶ シナリオ実行中にアクティブにするランデブー・ポイント
- ▶ 各ランデブー・ポイントに参加する仮想ユーザの数

例えば、銀行のサーバをテストするために、2つのランデブー・ポイントを含むシナリオを作成します。最初のランデブー・ポイントでは、1,000 仮想ユーザが同時に現金を預金するようにします。2番目のランデブー・ポイントでは、別の1,000 仮想ユーザが同時に現金を引き出すようにします。500 仮想ユーザが現金を預金するときの、サーバのパフォーマンスを測定するには、「引き出し」ランデブーは非活性化（無効に）し、「預金」ランデブーには500 仮想ユーザだけが参加するようにします。

次に、サーバの負荷のピークを制御する手順の概要を説明します。

- 1 仮想ユーザ・スクリプトを作成し、必要なランデブー・ポイントを挿入します。
- 2 シナリオを作成します。

シナリオに仮想ユーザ・グループを追加すると、LoadRunner は、そのグループに関連付けられているスクリプトでランデブー・ポイントの名前を探し、見つけた名前を [ランデブー情報] ダイアログ・ボックスのリスト（**[シナリオ]** > **[ランデブー]**）に追加します。同じスクリプトを実行する別の仮想ユーザ・グループを作成した場合、コントローラは、新しい仮想ユーザをランデブーに追加し、リストを更新します。

- 3 エミュレートするユーザ負荷のレベルを設定します。

シナリオで実行するランデブー・ポイントを選択し、各ランデブーに参加する仮想ユーザの数を指定して、負荷の正確なレベルを決めます。

4 ランデブーの属性を設定します（任意）。

各ランデブーに、「**ポリシー**」属性を設定できます。詳細については、164 ページ「ランデブー・ポリシーの設定」を参照してください。

5 シナリオを実行します。

ランデブー属性の設定

[ランデブー情報] ダイアログ・ボックス（[シナリオ] > [ランデブー]）から、次のランデブー属性を設定できます。

- ▶ ランデブーのポリシー
- ▶ ランデブー・ポイントを有効 / 無効にする
- ▶ 仮想ユーザを有効 / 無効にする

また、ダイアログ・ボックスには、ランデブー・ポイントについての一般的な情報が表示されます。例えば、ランデブーに関連付けられているスクリプトや、解放の履歴などです。



[解放] コマンドを使ってシナリオ実行中に仮想ユーザを操作する方法については、第14章「シナリオの実行」を参照してください。

ランデブー・ポリシーの設定

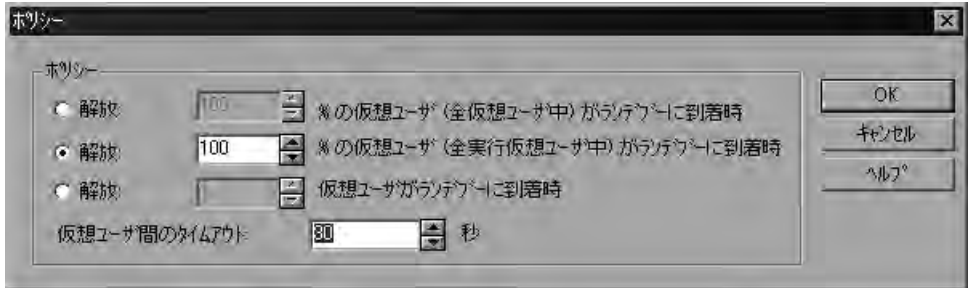
ランデブー・ポリシーを設定して、仮想ユーザでランデブー・ポイントをどのように扱うかを決めます。各ランデブーに、次のポリシー属性を設定します。

開放ポリシー：ランデブーから一度に解放される仮想ユーザの数。

タイムアウト：コントローラが、ランデブーから仮想ユーザを解放する前に待機する時間。

ランデブー・ポリシーの属性の設定は、次の手順で行います。

- 1 [シナリオ] > [ランデブー] を選択します。[ランデブー情報] ダイアログ・ボックスが表示されます。
- 2 [ランデブー] ボックスでランデブーを選択し、[ポリシー] ボタンをクリックします。[ポリシー] ダイアログ・ボックスが表示されます。



- 3 [ポリシー] セクションで、3つのオプションのいずれか1つを選択します。
 - ▶ **[解放: X%の仮想ユーザ(全仮想ユーザ中)がランデブーに到着時]** : すべての仮想ユーザのうち、指定した割合の仮想ユーザがランデブー・ポイントに到着した場合にだけ仮想ユーザを解放します。

注: このオプションは、シナリオのスケジュールに干渉します。このオプションを選択した場合には、シナリオはスケジュールどおりに実行されません。

- ▶ **[解放: X%の仮想ユーザ(全実行仮想ユーザ中)がランデブーに到着時]** : シナリオで実行中のすべての仮想ユーザのうち、指定した割合の仮想ユーザがランデブー・ポイントに到達した場合にだけ仮想ユーザを解放します。
 - ▶ **[解放: X仮想ユーザがランデブーに到着時]** : ランデブー・ポイントに指定した数の仮想ユーザが到達した場合にだけ仮想ユーザを解放します。
- 4 [仮想ユーザ間のタイムアウト] ボックスにタイムアウト値を入力します。各仮想ユーザがランデブー・ポイントに到着した後、LoadRunner は指定した最大**タイムアウト**時間だけ、次の仮想ユーザが到着するのを待機します。次の仮想ユーザが**タイムアウト**時間内に到着しない場合、コントローラはランデブーからすべての仮想ユーザを解放します。

新しい仮想ユーザが到着するたびに、タイマがゼロにリセットされます。**タイムアウト**の標準設定は30秒です。

- 5 [OK] をクリックして設定を保存し、[ポリシー] ダイアログ・ボックスを閉じます。

ランデブー・ポイントを有効 / 無効にする

ランデブーを一時的に無効にして、シナリオから除外できます。ランデブーを有効または無効にすることによって、サーバ負荷のレベルを変えられます。

[ランデブー情報] ダイアログ・ボックスの [ランデブーを無効にする / ランデブーを有効にする] ボタンを使って、ランデブーのステータスを変更します。

ランデブーを無効にするには、次の手順で行います。

- 1 [ランデブー] ボックスで、無効にするランデブーを選択します。
- 2 [ランデブーを無効にする] ボタンをクリックします。ボタンが [ランデブーを有効にする] に変わり、ランデブーは無効になります。

ランデブーを有効にするには、次の手順で行います。

- 1 [ランデブー] ボックスの無効になっているランデブーから、有効にするランデブーを選択します。
- 2 [ランデブーを有効にする] ボタンをクリックします。ボタンが [ランデブーを無効にする] に変わり、ランデブーは有効になります。

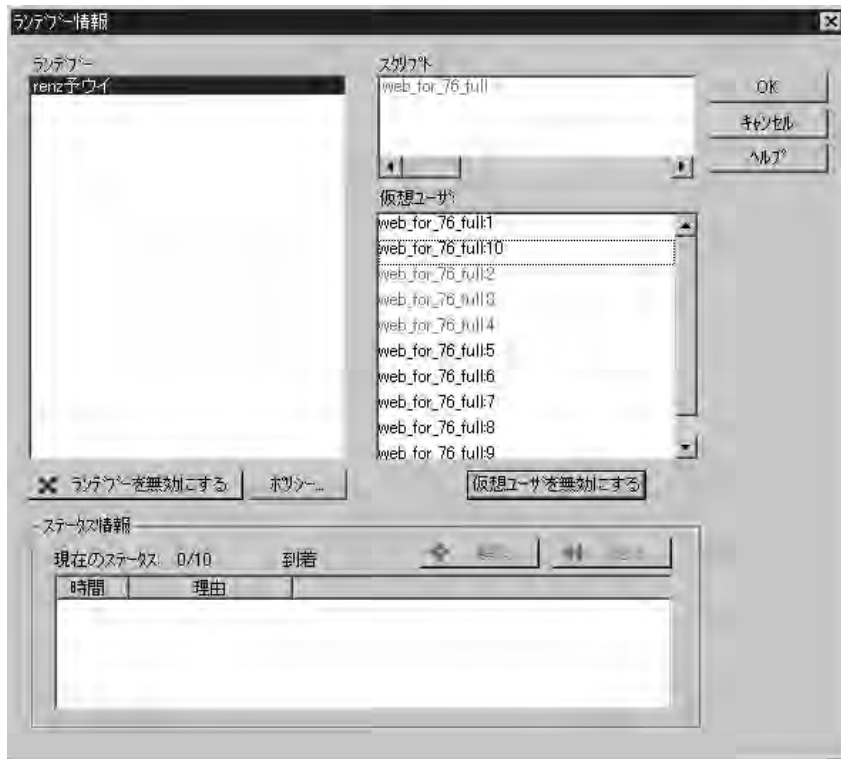
ランデブー・ポイントの仮想ユーザを有効 / 無効にする

LoadRunner は、シナリオのすべての仮想ユーザではなく、特定の仮想ユーザでだけランデブー・ポイントを無効にすることもできます。ランデブーで仮想ユーザを無効にすることによって、その仮想ユーザをランデブーから一時的に除外できます。無効にした仮想ユーザを再び有効にすれば、その仮想ユーザはランデブーに戻ります。[有効にする] コマンドと [無効にする] コマンドを使って、ランデブーに参加する仮想ユーザを指定します。

ランデブーで仮想ユーザを無効にするには、次の手順で行います。

- 1 [ランデブー] ボックスで、仮想ユーザを無効にするランデブーを選択します。

- 2 [仮想ユーザ] ボックスで、ランデブーから除外する仮想ユーザを選択します。複数の仮想ユーザを選択するには、Ctrl キーを使用します。



- 3 [仮想ユーザ] ボックスの下にある [仮想ユーザを無効にする] ボタンをクリックします。無効になった仮想ユーザが黒からグレーに変わり、ランデブーに参加しません。

仮想ユーザを有効にするには、仮想ユーザを選択して [仮想ユーザを有効にする] ボタンをクリックします。

ランデブー情報の表示

シナリオの実行中と実行後に、[ランデブー情報] ダイアログ・ボックスにランデブーのステータスを表示できます。次の情報が表示されます。

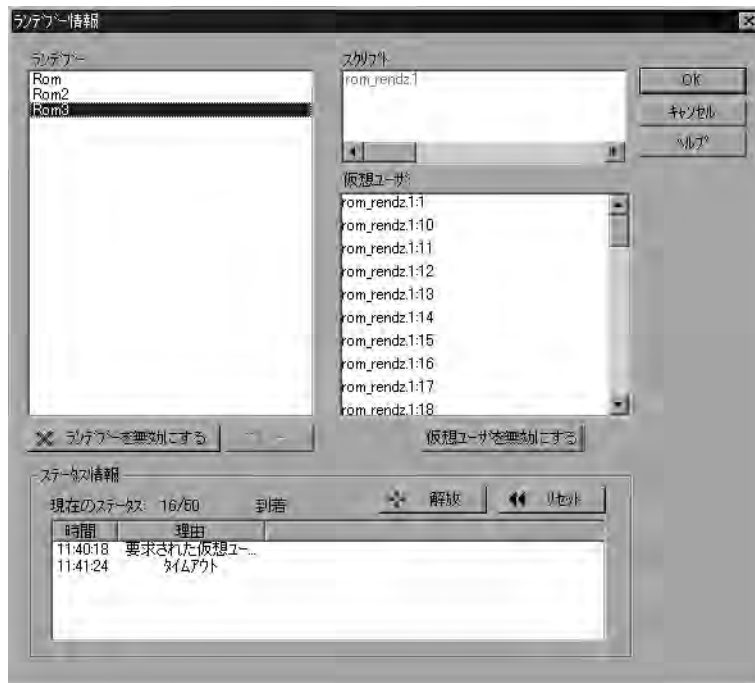
[現在のステータス]：ランデブー・ポイントに到着した仮想ユーザの数とランデブーに割り当てられた仮想ユーザの総数。

[時間]：ランデブー・ポイントで仮想ユーザが解放された時間。

[理由]：ランデブー・ポイントで仮想ユーザが解放された理由。理由は、「**タイムアウト**」または「**到着**」のいずれかです。

ランデブー情報の表示は、次の手順で行います。

情報を表示するランデブーを選択します。ランデブーのステータスが [ステータス情報] セクションに表示されます。



[ランデブー情報] ダイアログ・ボックスについて

[ランデブー情報] ダイアログ・ボックスを使用して、シナリオ内の各ランデブー・ポイントの属性の表示および変更ができます。

[ランデブー]：シナリオのランデブー・ポイントの名前が表示されます。

▶ **[ランデブーを有効にする / 無効にする]** : 選択したランデブー・ポイントのシナリオへの参加を有効 / 無効にします。

[スクリプト] : ランデブー・ポイントと関連付けられている仮想ユーザ・スクリプトを一覧表示します。

[仮想ユーザ] : ランデブー・ポイントと関連付けられている仮想ユーザを一覧表示します。

▶ **[仮想ユーザを有効にする / 無効にする]** : 仮想ユーザのランデブーへの参加を有効 / 無効にします。

[ポリシー] : [ポリシー] ダイアログ・ボックスを開きます。このダイアログ・ボックスでは、一度にランデブーから開放する仮想ユーザの数や、ランデブーから仮想ユーザを開放するまでにコントローラが待機する時間を設定できます。

▶ **[仮想ユーザ間のタイムアウト]** : タイムアウトの値を秒単位で入力します。各仮想ユーザがランデブー・ポイントに到着した後、LoadRunner はタイムアウトに指定した秒数だけ、次の仮想ユーザが到着するのを待機します。次の仮想ユーザがタイムアウト時間内に到着しない場合、コントローラはランデブーからすべての仮想ユーザを解放します。新しい仮想ユーザが到着するたびに、タイマがゼロにリセットされます。タイムアウトの標準設定は 30 秒です。ランデブー・ポイントごとにタイムアウトを設定します。

[ステータス情報]

▶ **[現在のステータス]** : ランデブー・ポイントに到着した仮想ユーザの数とランデブーに割り当てられた仮想ユーザの総数が表示されます。

▶ **[時間]** : ランデブーが開放された時刻が表示されます。

▶ **[理由]** : ランデブー・ポイントから仮想ユーザが開放された理由が表示されます。理由は、**タイムアウト**または**到着**のいずれかです。

▶ **[解放]** : 選択されたランデブー・ポイントで現在待機しているすべての仮想ユーザを開放します。すべての仮想ユーザがランデブーに到着しなかった場合にシナリオを続行するには、このボタンをクリックします。

▶ **[リセット]** : 現在表示されている情報を削除して、ステータス情報をリセットします。

第 10 章

シナリオの設定

シナリオ実行時のロード・ジェネレータや仮想ユーザの動作を設定して、シナリオが正確に作業環境をエミュレートするようにできます。

本章では、次の項目について説明します。

- ▶ シナリオの設定について
- ▶ タイムアウト時間の設定
- ▶ シナリオ実行環境の設定
- ▶ 実行ファイルの格納場所の設定
- ▶ パス変換の指定

シナリオの設定について

シナリオを実行する前に、シナリオに対するロード・ジェネレータと仮想ユーザの振る舞いを設定します。標準設定はほとんどの環境に対応しますが、LoadRunner では、設定を変更してシナリオの振る舞いをカスタマイズできます。この設定はその後のすべてのシナリオ実行に適用されるので、通常は一度だけ設定すれば済みます。

本章で説明されている設定は、シナリオのすべてのロード・ジェネレータに適用されます。各ロード・ジェネレータ・マシンの設定変更については、第 5 章「マニュアル・シナリオの作成」を参照してください。グローバルなシナリオ設定が各ロード・ジェネレータのシナリオ設定と異なる場合は、ロード・ジェネレータの設定が適用されます。

本章で説明されている設定は、仮想ユーザの実行環境の設定とは関係ありません。仮想ユーザの実行環境の設定は、各仮想ユーザまたはスクリプトに適用されるもので、ログ、思考遅延時間、ネットワーク、反復回数、ブラウザに関する

る情報が含まれます。実行環境の設定方法の詳細については、『Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド』を参照してください。

オンライン・モニタの設定の詳細については、第24章「オンライン監視」を参照してください。

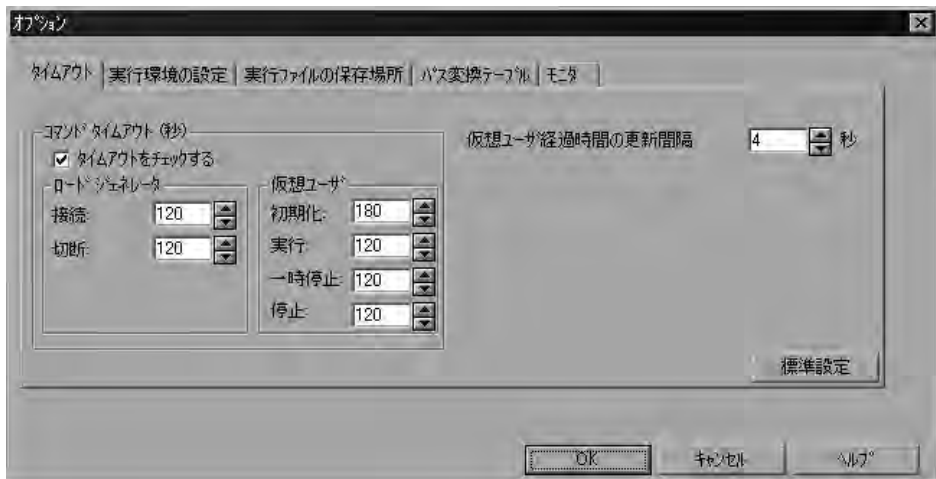
LoadRunner のエキスパート・モードでは、LoadRunner エージェントとその他の LoadRunner コンポーネントに対するさらに多くの設定ができます。詳細については、付録 C 「エキスパート・モードでの作業」を参照してください。

タイムアウト時間の設定

[タイムアウト] タブでは、ロード・ジェネレータに関連するコマンドのタイムアウト値を指定できます。コマンドの実行がタイムアウト時間内に成功しなかった場合、ロード・ジェネレータのステータスは「エラー」に変わります。

タイムアウト時間を設定するには、次の手順で行います。

- 1 [ツール] > [オプション] を選択します。[オプション] ダイアログ・ボックスが表示されます。[タイムアウト] タブを選択します。



- 2 コマンドのタイムアウト時間を指定するには、[タイムアウトをチェックする] チェック・ボックスを選択し、適切なタイムアウト時間を指定します。タイムアウトの検査を無効にするには、[タイムアウトをチェックする] チェック・ボックスをクリアします。

- 3 [仮想ユーザ経過時間の更新間隔] ボックスで、LoadRunner が経過時間を更新する頻度を指定します。

[オプション] ダイアログ・ボックスの [タイムアウト] タブについて

LoadRunner では、コマンドと仮想ユーザの経過時間についてタイムアウト時間を設定できます。

コマンドのタイムアウトは、各種の LoadRunner コマンドに対する時間制限です。コマンドがコントローラにより発行される時に、ロード・ジェネレータまたは仮想ユーザがコマンドを実行するときの最長時間を設定します。タイムアウト時間内にコマンドが完了しないと、コントローラがエラー・メッセージを発行します。

[コマンドタイムアウト (秒)]

[タイムアウトをチェックする] : LoadRunner に対して、コマンドがコントローラから発行された後で、ロード・ジェネレータと仮想ユーザのステータスを監視するように指示します。ロード・ジェネレータまたは仮想ユーザが指定されたタイムアウト時間内にコマンドを完了しなかった場合、コントローラがエラー・メッセージを発行します。タイムアウトの制限を無効にした場合、LoadRunner は、ロード・ジェネレータが接続および切断するまで、および [初期化], [実行], [一時停止], [停止] コマンドが実行されるまで無期限に待機します。

▶ [ロード ジェネレータ]

- ▶ [接続] : ロード ジェネレータに接続するまで LoadRunner が待機する制限時間を入力します。接続がこの制限時間内に成功しなかった場合、ロード・ジェネレータのステータスは [失敗] に変わります。標準設定の接続タイムアウトの値は 120 秒です。
- ▶ [切断] : ロード ジェネレータから切断するまで LoadRunner が待機する制限時間を入力します。切断がこの制限時間内に成功しなかった場合、ロード・ジェネレータのステータスは [失敗] に変わります。標準設定の切断タイムアウトの値は 120 秒です。

注 : LoadRunner は、アクティブな仮想ユーザの数がタイムアウト値に影響するという事実を認識します。例えば、1000 仮想ユーザの初期化は、10 仮想ユーザの初期化よりかなり時間がかかります。LoadRunner は、アクティブな仮想ユーザの数に基づき、指定したタイムアウト値に内部的な値を加算します。

▶ **[仮想ユーザ]**

- ▶ **[初期化]** : [初期化] コマンドのタイムアウト値を入力します。標準設定のタイムアウト値は 180 秒です。
- ▶ **[実行]** : [実行] コマンドのタイムアウト値を入力します。標準設定のタイムアウト値は 120 秒です。
- ▶ **[一時停止]** : [一時停止] コマンドのタイムアウト値を入力します。標準設定のタイムアウト値は 120 秒です。
- ▶ **[停止]** : [停止] コマンドのタイムアウト値を入力します。標準設定のタイムアウト値は 120 秒です。

[仮想ユーザ経過時間の更新間隔] : [仮想ユーザ] ダイアログ・ボックスの [経過時間] カラムに表示される値が LoadRunner によって更新される頻度を指定します。標準設定は 4 秒です。

例を示します。

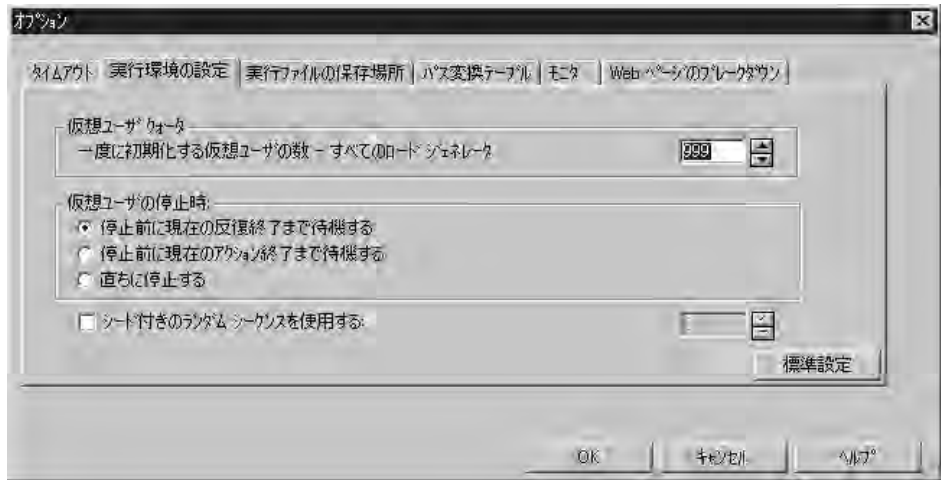
仮想ユーザを選択して **[選択した仮想ユーザを初期化]** ボタンをクリックすると、LoadRunner は、仮想ユーザのステータスが 180 秒（標準設定の **[初期化]** タイムアウト時間）以内に「準備完了」になったかどうかを検査します。このステータスになっていない場合、コントローラは **[初期化]** コマンドがタイムアウトしたことを示すメッセージを発行します。

シナリオ実行環境の設定

[実行環境の設定] タブでは、仮想ユーザ・クォータ、仮想ユーザの停止、およびランダム・シーケンスのシードに関連するシナリオ実行環境の設定を指定できます。

シナリオの実行環境の設定は、次の手順で行います。

- 1 [ツール] > [オプション] を選択します。[オプション] ダイアログ・ボックスが表示されます。[実行環境の設定] タブをクリックします。



- 2 仮想ユーザ・クォータを設定するには、必要な値を指定します。
- 3 LoadRunner にどのように仮想ユーザの実行を停止させるか選択します。
- 4 ランダム・シーケンスのシード値を指定するには、[シード付きのランダムシーケンスを使用する] チェック・ボックスを選択し、必要なシード値を入力します。

[オプション] ダイアログ・ボックスの [実行環境の設定] タブについて

[実行環境の設定] タブでは、仮想ユーザ・クォータ、仮想ユーザの停止方法、ランダム・シーケンスのシードなどの実行環境設定の値を指定できます。

[**仮想ユーザ・クォータ**]: システムの過負荷を防ぐために、仮想ユーザの活動のクォータを設定できます。仮想ユーザ・クォータはすべてのロード・ジェネレータ上の仮想ユーザに適用されます。

- ▶ [**一度に初期化する仮想ユーザの数 - すべてのロード ジェネレータ**]: [初期化] コマンドの送信時にロード・ジェネレータが一度に初期化できる仮想ユーザの最大数を設定できます。

[仮想ユーザの停止時]：[停止] ボタンをクリックしたときの、仮想ユーザの実行停止の方法を制御します。

次のオプションのいずれかを選択します。

- ▶ **[停止前に現在の反復終了まで待機する]**：LoadRunner に対して、実行中の反復が完了してから仮想ユーザを停止するように指示します。仮想ユーザのステータスが [徐々に終了] に変わり、シナリオは徐々に停止します。
- ▶ **[停止前に現在のアクション終了まで待機する]**：LoadRunner に対して、実行中のアクションが完了してから仮想ユーザを停止するように指示します。仮想ユーザのステータスが [徐々に終了] に変わり、シナリオは徐々に停止します。
- ▶ **[直ちに停止する]**：LoadRunner に対して、仮想ユーザを直ちに停止するように指示します。仮想ユーザのステータスが [終了中] に変わり、シナリオが直ちに停止します。

[シード付きのランダム シーケンスを使用する]：LoadRunner がランダム・シーケンスにシード値を使用できるようにします。各シード値は、テスト実行に使用されるランダム値のシーケンスを表します。同じシード値を使用している場合はいつも同じ値のシーケンスがシナリオ内の仮想ユーザに割り当てられます。この設定は、データ・ファイルの値を割り当てるためにランダム方式を使って、パラメータ化された仮想ユーザ・スクリプトに適用されます。この設定は、記録される思考遅延時間のランダム・パーセンテージにも影響します（『Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド』で [実行環境設定] ダイアログ・ボックスに関する情報を参照してください）。

テスト実行で問題を見つけ、同じランダム値シーケンスを使ってテストを繰り返したい場合は、このオプションを有効にします。

実行ファイルの格納場所の設定

シナリオを実行すると、標準では実行ファイルは各仮想ユーザのロード・ジェネレータ（仮想ユーザ・スクリプトを実行するマシン）に格納されます。ファイルの標準の格納先は、ロード・ジェネレータの環境変数（Windows の場合は TEMP または TMP、UNIX の場合は \$TMPDIR または \$TMP）で指定された一時

ディレクトリです。環境変数が定義されていない場合は、ファイルは¥tmpディレクトリに格納されます。

注：本章で説明する実行ファイルの格納場所は、シナリオのすべてのロード・ジェネレータに適用されます。74 ページ「ロード・ジェネレータの設定」で説明しているとおりに、個々のロード・ジェネレータの設定を変更できます。

主な実行ファイルは仮想ユーザ・スクリプト・ファイルと結果ファイルです。

スクリプト・ファイル

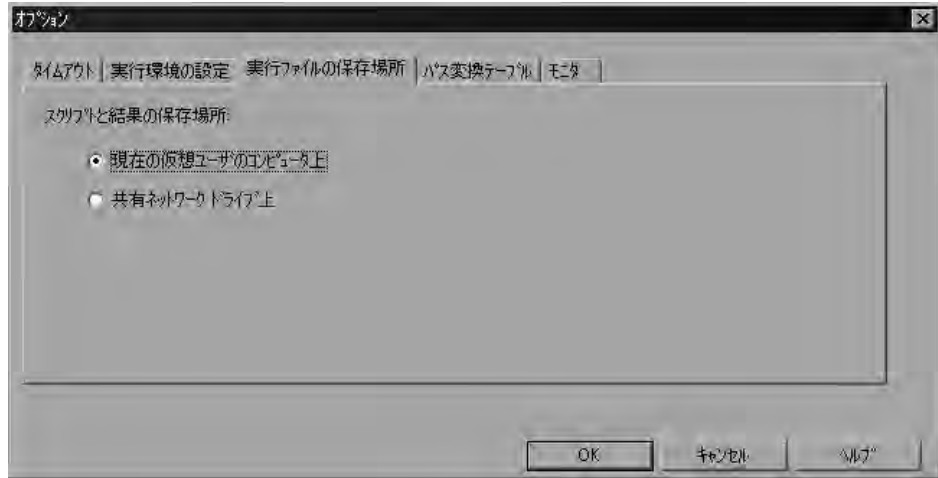
仮想ユーザを実行すると、コントローラは関係する仮想ユーザ・スクリプトのコピーを仮想ユーザのロード・ジェネレータに送信します。このスクリプトは、ロード・ジェネレータの実行ファイルの一時ディレクトリに保存されます。

結果ファイル

シナリオ実行中、参加している仮想ユーザは結果を実行ファイルの一時ディレクトリに書き込みます。シナリオ実行後、これらの結果ファイルは、照合または統一され、すべてのロード・ジェネレータの結果は、結果ディレクトリに転送されます。第14章「シナリオの実行」で説明したとおりに結果ディレクトリの場所を設定します。結果照合後、実行ファイルの一時ディレクトリは削除されます。

LoadRunner が実行ファイルを格納する場所の指定は、次の手順で行います。

- 1 [ツール] > [オプション] を選択します。[オプション] ダイアログ・ボックスが表示されます。[実行ファイルの保存場所] タブを選択します。



標準では、[現在の仮想ユーザのコンピュータ上] オプションが選択されています。このオプションを選択すると、結果ファイルとスクリプト・ファイルを含むすべての実行ファイルが仮想ユーザのロード・ジェネレータに格納されます。ローカルのロード・ジェネレータ（コントローラ・マシン）上で実行している仮想ユーザは例外で、共有ドライブ・オプションを使用する必要があります。

- 2 共有ネットワーク上に結果ファイルを格納するには、[共有ネットワーク ドライブ上] をクリックします。ネットワーク上で場所を正確に指定する方法については、第 11 章「シナリオの実行準備」を参照してください。
- 3 [OK] をクリックして、ダイアログ・ボックスを閉じます。

[オプション] ダイアログ・ボックスの [実行ファイルの保存場所] タブについて

[実行ファイルの保存場所] タブでは、実行ファイルを LoadRunner に保存する場所を指定できます。

[スクリプトと結果の保存場所]：次のオプションのいずれかを選択します。

- ▶ [現在の仮想ユーザのコンピュータ上]：コントローラに対して、仮想ユーザ・スクリプトを実行しているコンピュータに実行ファイルを保存するように指示

します。NT ベースのコンピュータでは、TEMP または TMP 環境変数によって定義されるディレクトリに結果が保存されます。UNIX マシンでは、TMPDIR 環境変数によって定義されるディレクトリに結果が保存されます。TMPDIR 環境変数が定義されていない場合、結果は /tmp ディレクトリに保存されます。

注：結果ファイルを仮想ユーザのロード・ジェネレータに保存する場合は、分析を実行する前に結果を照合しなくてはなりません。アナリシス・ツールを起動したときには LoadRunner が結果を照合するのを待機できます。また、**[結果]** > **[結果の照合]** を選択して結果を照合することもできます。別の方法として、**[結果]** > **[結果の自動照合]** を選択して、自動的に各シナリオ実行の最後に結果を照合するようにできます。

- ▶ **[共有ネットワーク ドライブ上]**：コントローラに対して、シナリオの実行結果および仮想ユーザ・スクリプトの一方または両方を、共有ネットワーク・ドライブに保存するように指示します。共有ネットワーク・ドライブは、コントローラと、シナリオ内のすべてのロード・ジェネレータが読み取りおよび書き込み権限を持っているドライブです。結果を共有ネットワーク・ドライブに保存する場合は、パス変換が必要になることもあります。パス変換により、リモートのロード・ジェネレータは指定した結果ディレクトリを確実に識別できるようになります。パス変換の詳細については、付録 B 「パス変換の実行」を参照してください。

すべての仮想ユーザが特定の場所にある仮想ユーザ・スクリプトに直接アクセスするように指定すると、実行時にスクリプト・ファイルの転送は発生しません。この方法は、次のような場合に役に立ちます。

- ▶ ファイル転送が機能していない。
- ▶ 仮想ユーザ・スクリプトが大きく転送に時間がかかる。仮想ユーザ・スクリプト・ファイルはシナリオ実行中で一度しか転送されません。

この方法を選択すると、多くの場合はパス変換が必要となります。詳細については、付録 B 「パス変換の実行」を参照してください。

パス変換の指定

実行ファイルの格納場所として共有ネットワーク・ドライブを指定した場合、(176 ページ「実行ファイルの格納場所の設定」を参照)、「**パス変換**」を実行する必要がある場合もあります。パス変換は、LoadRunner がリモート・パス名を変換する際に使用するメカニズムです。一般的なシナリオには、共有ネットワーク・ドライブを個別にマップするいくつかのロード・ジェネレータ・マシンが含まれます。詳細については、付録 B「パス変換の実行」を参照してください。

第 11 章

シナリオの実行準備

シナリオを実行する前に、シナリオの結果の格納場所と実行環境に関連するその他の設定を行います。

本章では、次の項目について説明します。

- ▶ シナリオの実行準備について
- ▶ 結果の格納場所の指定
- ▶ 結果ディレクトリのファイル構造
- ▶ 結果の照合
- ▶ シナリオのサマリ情報の設定

シナリオの実行準備について

シナリオを実行する前に、結果の場所を指定し（必須）、結果に名前を割り当て、シナリオのスケジュールを設定し、シナリオのサマリ情報を提供する必要があります。さらに、シナリオ起動時に呼び出すアプリケーションを指定することもできます。

シナリオの準備設定の大部分は省略可能ですが、それらを使うことによってテスト・プロセスを強化できます。これらの値はシナリオに固有のもので、各 LoadRunner シナリオに別々の値を設定できます。

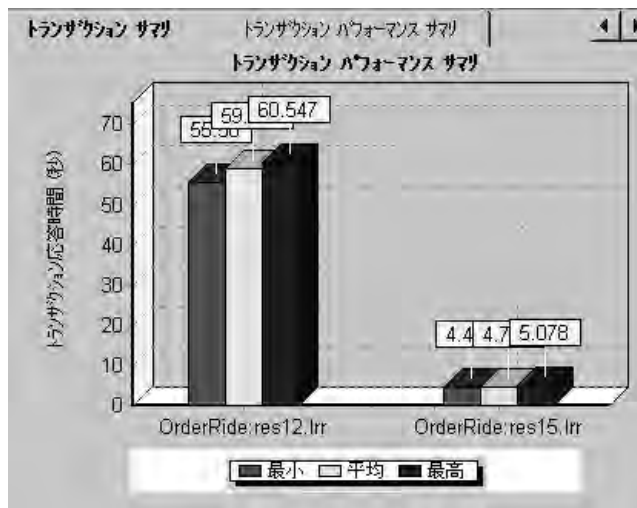
タイムアウト、出力、クォータなどの 1 回限りの構成設定については、第 10 章「シナリオの設定」を参照してください。

結果の格納場所の指定

シナリオを実行すると、標準設定では実行ファイルは各ロード・ジェネレータにローカルに保存されます。シナリオ実行後、結果はコントローラ・マシンに集められて処理されます。また、LoadRunnerが結果を共有ネットワーク・ドライブに保存するようにもできます。ファイルの格納方法の指定については、176ページの「実行ファイルの格納場所の設定」を参照してください。

LoadRunnerでは、各結果セットにわかりやすい名前を付けることができます。これは、LoadRunnerでいくつもの実行結果を1つのグラフに重ね合わせて複数のシナリオ実行を比較するクロス結果分析で、特に役立ちます。わかりやすいグラフの名前で、複数の実行結果を区別できます。

下の例では、2つのシナリオ実行結果が重ね合わされています。結果セットは、**res12**と**res15**です。

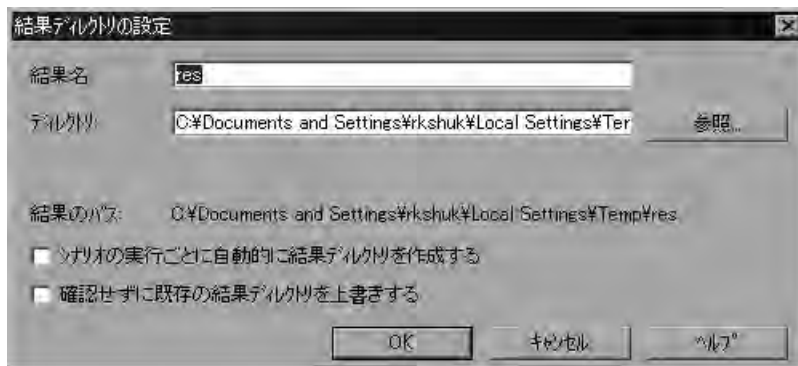


クロス結果グラフの詳細については、『Mercury LoadRunner アナリシス・ユーザーズ・ガイド』を参照してください。

注：また、Mercury の Web ベース・テスト管理プログラム Quality Center を使って、結果をプロジェクトに保存することもできます。詳細については、第 12 章「Quality Center を使ったシナリオ管理」を参照してください。

結果を格納する場所の指定は、次の手順で行います。

- 1 [結果] > [結果の設定] を選択します。[結果ディレクトリの設定] ダイアログ・ボックスが表示されます。



- 2 [結果名] ボックスに、結果の名前を入力します。異なるパスを持つ同じ名前を使わないようにします。パスが違っていてもグラフ上では同じに見えてしまうからです。
- 3 [ディレクトリ] ボックスに、結果ディレクトリの完全パスを入力します。標準のファイル格納場所の設定（ローカル・マシン）を使っている場合には、シナリオ実行後に集められたすべての結果を格納するディレクトリを指定します。ファイルの格納先に共有ネットワーク・ドライブを指定している場合には、シナリオ実行中に仮想ユーザが書き込みを行うディレクトリを指定します。
- 4 以降の実行に適切であれば、[シナリオの実行ごとに自動的に結果ディレクトリを作成する] または [確認せずに既存の結果ディレクトリを上書きする] チェック・ボックスを選択します。
- 5 [OK] をクリックして、結果ディレクトリの設定を保存します。

[結果ディレクトリの設定] ダイアログ・ボックスについて

[結果ディレクトリの設定] ダイアログ・ボックスを使用して、コントローラがシナリオを保存する場所を指定できます。

注：Quality Center プロジェクトと接続中ならば、コントローラはテスト・セットに結果を保存します。標準ファイル・システムを使って、結果をディスクに直接保存することもできます。

[**結果名**]：結果の名前を指定します。コントローラは、その名前を使用して結果を保存します。

[**ディレクトリ**]：コントローラが結果を保存するファイル・システムの場所を指定します。[参照] をクリックして、希望の場所を見つけます。コントローラは結果ディレクトリ内にサブディレクトリを作成します。すべての結果は、このサブディレクトリに保存されます。

[**結果のパス**]：結果に指定した場所を表示します。

[**シナリオの実行ごとに自動的に結果ディレクトリを作成する**]：LoadRunner が、シナリオの実行ごとに固有の結果ディレクトリを作成するようにします。標準設定では、**res1**、**res2**、**res3** などの結果名になります。

[**確認せずに既存の結果ディレクトリを上書きする**]：ユーザに確認せずに既存の結果セットを自動的に上書きします。

[**Quality Center**] (Quality Center に接続中のみ)：Quality Center テスト・セットに結果を保存できます。

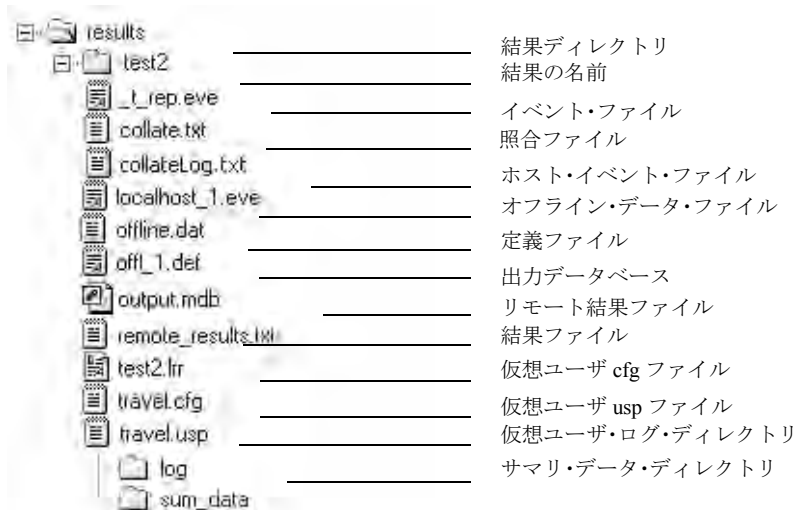
[**ファイル システム**] (Quality Center に接続中のみ)：標準設定の LoadRunner ディレクトリ・パスを表示します。

結果ディレクトリのファイル構造

結果ディレクトリを設定する場合は、結果の名前も指定します。LoadRunner は、結果の名前を使ってサブディレクトリを作成し、そのディレクトリに収集したすべてのデータをまとめて格納します。どの結果にも、シナリオに関する

一般的な情報が含まれている結果ファイル（.lrr）とイベント・ファイル（.eve）があります。

シナリオ実行中，LoadRunner によって，シナリオのグループごとのディレクトリと，仮想ユーザごとのサブディレクトリが作成されます。典型的な結果ディレクトリの構造は次のようになります。



- ▶ メインの結果ディレクトリにある **t_rep.eve** には，仮想ユーザとランデブー情報が含まれます。
- ▶ **collate.txt** には，結果ファイルのファイル・パスとアナリシスの照合情報が含まれます。
- ▶ **collateLog.txt** には，各ロード・ジェネレータからの結果，診断，およびログ・ファイルの照合のステータス（成功 / 失敗）が含まれます。
- ▶ **local_host.eve** には，各エージェント・ホストからの情報が含まれます。
- ▶ **offline.dat** には，サンプルのモニタ情報が含まれます。
- ▶ ***.def** は，オンライン・モニタとその他のカスタム・モニタを記述するグラフの定義ファイルです。

- ▶ **output.mdb** は、アナリシスによって（結果ファイルから）作成されるデータベースで、出力情報が格納されます。
- ▶ **remote_results.txt** には、ホスト・イベント・ファイルのファイル・パスが格納されます。
- ▶ **results_name.lrr** は、LoadRunner アナリシスのドキュメント・ファイルです。
- ▶ ***.cfg** ファイルには、仮想ユーザ・アプリケーションで定義されたスクリプトの実行環境の設定（思考遅延時間、反復、ログ、Web）のリストが含まれます。
- ▶ ***.usp** ファイルには、アクション・セクションをどのように実行するかを含めた、スクリプトの実行論理が含まれます。
- ▶ **Log** ディレクトリには、各仮想ユーザの再生中に生成された出力情報が含まれます。シナリオで実行される各仮想ユーザ・グループごとに、独立したディレクトリが存在します。各グループ・ディレクトリは仮想ユーザのサブディレクトリからなります。
- ▶ **Sum data** ディレクトリ。グラフ・サマリ・データ (.dat) ・ファイルを含むディレクトリ。

アナリシス・グラフとレポートを生成するとき、LoadRunner アナリシス・エンジンはすべてのシナリオ結果ファイル (.eve と .lrr) をデータベースにコピーします。データベースが作成されると、アナリシスは直接データベースを処理し、結果ファイルは使用しません。


LoadRunner アナリシスについては、『Mercury LoadRunner アナリシス・ユーザーズ・ガイド』を参照してください。

結果の照合

シナリオを実行すると、標準設定ではすべての仮想ユーザの情報が各ロード・ジェネレータでローカルに保存されます。シナリオ実行後、結果は自動的に照合または統合されます。つまり、すべてのロード・ジェネレータの実行結果は結果ディレクトリに転送されます。182 ページ「結果の格納場所の指定」の説明にしたがって、結果ディレクトリの場所を設定します。

注：すべてのシナリオの実行結果を直接共有ネットワーク・ドライブに格納する場合は、結果の照合は必要ありません。実行結果の格納方法の変更については、171 ページ「シナリオの設定について」を参照してください。

自動照合を有効 / 無効にするには、次の手順を実行します。

 結果の自動照合

自動照合を有効にするには、**[結果]** > **[結果の自動照合]** を選択します。この機能が有効なときには、**[結果の自動照合]** アイコンがステータス・バーに表示されます。自動照合を無効にするには、このオプションの横のチェック・ボックスをクリアにします。

注：診断モジュールを使用している場合は、ロード・ジェネレータが **localhost** の場合も結果を照合する必要があります。

結果を手動で照合するには、次の手順を実行します。

[結果] > **[結果の照合]** > **[結果を照合]** を選択します。**[結果の照合]** ダイアログ・ボックスが開き、各ロード・ジェネレータからの結果とログ・ファイルの照合の進行状況が表示されます。



[**一般ステータス**]：照合結果ファイルに、イベント、診断、ログ・ファイルのステータスとファイル・サイズが表示されます。

注：表示されるファイル・サイズは、圧縮前のサイズです。圧縮後、結果ファイルのサイズは小さくなります。

[**進行の詳細**]：各ロード・ジェネレータからの結果、診断、ログ・ファイルの照合のステータス（成功 / 失敗）が表示されます。この情報は **collateLog.txt** ファイルに保存されます。

[**自動的に閉じる**]：照合の終了後、[結果の照合] ダイアログ・ボックスを自動的に閉じます。

結果の照合を停止するには、次の手順を実行します。

[**停止**] をクリックし、[**閉じる**] をクリックします。結果の照合を再開するには、[**結果**] > [**結果の照合**] > [**停止した照合を継続**] を選択します。

注： ログ・ファイルの照合を無効にすることもできます。詳細については、394 ページ「オプション — 一般設定」を参照してください。

ログ・ディレクトリと結果ディレクトリがロード・ジェネレータから削除されるのは、LoadRunner がマシンからの結果の照合に成功した後です。したがって、シナリオを保存した後でコントローラを閉じて、コントローラでシナリオを再び開けば結果を照合できます。

ディスク領域の不足によって照合が失敗した場合は、[**結果**] > [**結果の照合**] > [**結果の再照合**] を選択します。これによって LoadRunner は、.eve ファイルを圧縮することなく、再び結果の照合を試みます。

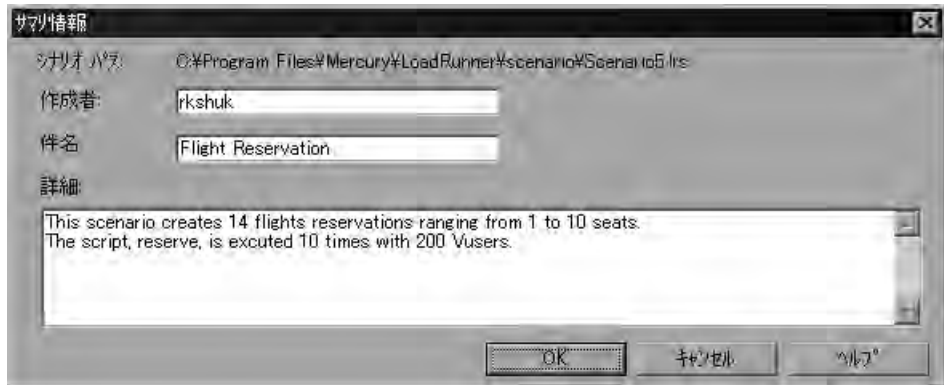
LoadRunner は、分析データを生成する前に、以前に照合が行われていなければ自動的に結果を照合します。

注： [結果] メニューで [**アナリシスの自動起動**] オプションを有効にすると、時間のかかる照合プロセスの間にアナリシスが起動し、アナリシス・サマリ・データが表示されるようになります。

シナリオのサマリ情報の設定

コントローラで、シナリオの詳細を記述できます。また、作成者の名前とシナリオの件名も記載できます。シナリオを開くときにはいつでも、このサマリ情報を利用できます。

[サマリ情報] ボックスは, [シナリオ] > [サマリ情報] を選択して開きます。



[シナリオパス] : シナリオの定義ファイル (.lrs) の名前と場所を表示します。

[作成者] : シナリオの作成者の名前を入力します。

[件名] : 件名やシナリオの簡単なタイトルを入力します。

[詳細] : シナリオの詳細を入力します。

第 12 章

Quality Center を使ったシナリオ管理

LoadRunner と Quality Center の統合により、Quality Center を使って LoadRunner のシナリオを管理できます。Quality Center によって、スクリプト、シナリオおよび実行結果の整理と管理が容易になります。

本章では、次の項目について説明します。

- ▶ Quality Center を使ったシナリオ管理について
- ▶ Quality Center の接続と切断
- ▶ Quality Center プロジェクトからシナリオを開く
- ▶ Quality Center プロジェクトへのシナリオの保存
- ▶ Quality Center プロジェクトへの結果の保存
- ▶ Quality Center プロジェクトからの仮想ユーザ・スクリプトの追加

Quality Center を使ったシナリオ管理について

LoadRunner は、Mercury の Web ベースのテスト管理ツール Quality Center と組み合わせて使えます。Quality Center はシナリオの格納と取得、および結果の収集を効率よく行う手段を提供します。シナリオと実行結果を Quality Center プロジェクトに保存し、それらを一意のグループにまとめます。

LoadRunner で Quality Center プロジェクトにアクセスするには、Quality Center がインストールされている Web サーバに接続する必要があります。ローカル Web サーバとリモート Web サーバのどちらにも接続できます。

Quality Center を使った作業の詳細については、『**Mercury Quality Center ユーザーズ・ガイド**』を参照してください。

Quality Center の接続と切断

LoadRunner と Quality Center の両方を使って作業している場合には、LoadRunner から Quality Center プロジェクトとやり取りできます。テスト・プロセスにおいて、LoadRunner と Quality Center プロジェクトはいつでも接続または切断できます。

Quality Center への LoadRunner の接続

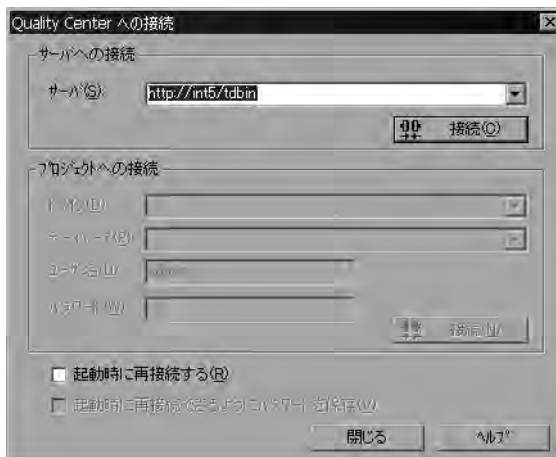
接続プロセスには次の2つの段階があります。最初に、LoadRunner をローカル Quality Center Web サーバまたはリモート Quality Center Web サーバに接続します。このサーバによって、LoadRunner と Quality Center プロジェクトの間の接続が処理されます。

次に、LoadRunner からアクセスするプロジェクトを選択します。プロジェクトには、テスト対象アプリケーションに関するシナリオと実行結果が保存されます。

注：Quality Center プロジェクトはパスワードで保護されているため、ユーザ名とパスワードを指定する必要があります。

LoadRunner から Quality Center へ接続するには、次の手順を実行します。

- 1 コントローラで、[ツール] > [Quality Center への接続] を選択します。
[Quality Center への接続] ダイアログ・ボックスが表示されます。



- 2 [サーバ] ボックスに、Quality Center がインストールされている Web サーバの URL アドレスを入力します。

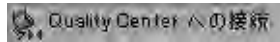
注： ローカル・エリア・ネットワーク (LAN) または広域ネットワーク (WAN) を介してアクセスできる Web サーバを選択します。

- 3 **[接続]** をクリックします。サーバへの接続が確立されると、[サーバ] ボックスにサーバの名前が読み取り専用形式で表示されます。
- 4 [プロジェクトへの接続] セクションに関連する情報を入力して **[接続]** をクリックし、LoadRunner を選択したプロジェクトへ接続します。選択したプロジェクトへの接続が確立されると、[データベース] ボックスにプロジェクトの名前が読み取り専用形式で表示されます。
- 5 起動時に Quality Center サーバと選択したプロジェクトに自動的に再接続するには、**[起動時に再接続する]** チェック・ボックスを選択します。
- 6 **[起動時に再接続する]** チェック・ボックスを選択した場合は、起動時に再接続するために、指定したパスワードを保存できます。**[起動時に再接続できるようにパスワードを保存]** チェック・ボックスを選択します。

パスワードを保存しなければ、起動時に LoadRunner を Quality Center に接続するときに、パスワードの入力が必要になります。

- 7 **[閉じる]** をクリックして、[Quality Center への接続] ダイアログ・ボックスを閉じます。

ステータス・バーに、LoadRunner が現在 Quality Center プロジェクトに接続していることが示され通知されます。

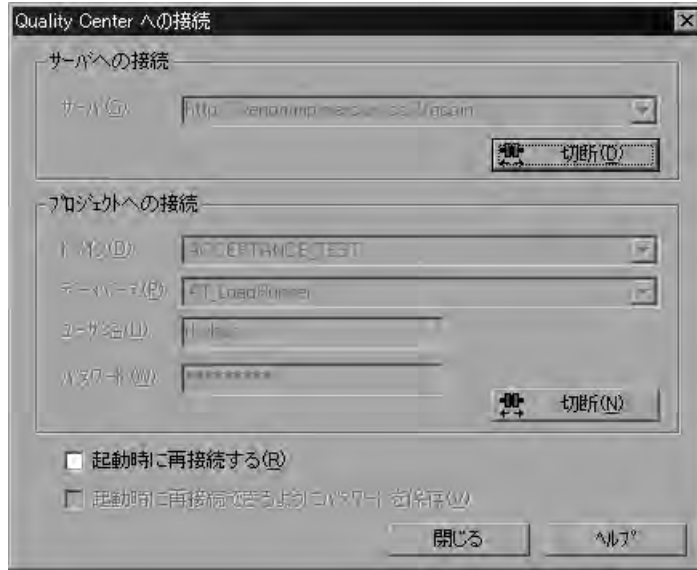


Quality Center からの LoadRunner の切断

選択した Quality Center プロジェクトと Web サーバから LoadRunner を切断できます。

Quality Center から LoadRunner を切断するには、次の手順を実行します。

- 1 コントローラで、[ツール] > [Quality Center への接続] を選択します。
[Quality Center への接続] ダイアログ・ボックスが表示されます。



- 2 LoadRunner を選択したプロジェクトから切断するには、[プロジェクトへの接続] セクションの [切断] をクリックします。
- 3 LoadRunner を選択したサーバから切断するには、[サーバへの接続] セクションの [切断] をクリックします。
- 4 [閉じる] をクリックして、[Quality Center への接続] ダイアログ・ボックスを閉じます。

[Quality Center への接続] ダイアログ・ボックスについて

[Quality Center への接続] ダイアログ・ボックスでは、Quality Center プロジェクトへの接続を開くことができます。Quality Center では、シナリオ、シナリオの実行結果、仮想ユーザ・スクリプトの整理と管理のために、プロジェクト・リポジトリを使用します。

[サーバへの接続] : Quality Center プロジェクトを使って作業する前に、プロジェクトをホストするサーバへの接続を確立しておく必要があります。

- ▶ **[サーバ]** : Quality Center プロジェクトをホストするサーバの名前を入力します。
- ▶ **[接続]** : 指定したサーバに接続します。
[プロジェクトへの接続] : コントローラを Quality Center データベース・サーバに接続したら、ドメイン名を入力して、プロジェクトを選択し、そのプロジェクトのユーザ名とパスワードを入力します。プロジェクトには、シナリオ実行に関する情報が保存されます。
- ▶ **[ドメイン]** : ドメイン名を入力します。
- ▶ **[データベース]** : 接続先のデータベースを選択します。一覧には、選択したサーバに含まれているすべてのプロジェクトが表示されます。
- ▶ **[ユーザ名]** : ユーザ名を入力します。
- ▶ **[パスワード]** : パスワードを入力します。
- ▶ **[接続]** : 選択したプロジェクトに接続します。
[起動時に再接続する] : これが選択されている場合は、LoadRunner は、コントローラの起動時に Quality Center サーバと、指定されたプロジェクトに自動的に接続します。
[起動時に再接続できるようにパスワードを保存] : これが選択されている場合は、ログイン・プロセスを自動化するために、指定したパスワードがレジストリに保存されます。

Quality Center プロジェクトからシナリオを開く

LoadRunner を Quality Center プロジェクトに接続する場合、Quality Center からシナリオを開くことができます。テストは、ファイル・システムの実際の位置からではなく、テスト計画ツリーでの位置から見つけます。

Quality Center プロジェクトからシナリオを開くには、次の手順を実行します。

- 1 Quality Center サーバに接続します (192 ページ「Quality Center への LoadRunner の接続」を参照してください)。

- 2 コントローラで、[ファイル] > [開く] を選択するか、[ファイルを開く] をクリックします。[Quality Center プロジェクトからシナリオを開く] ダイアログ・ボックスが表示され、テスト計画ツリーが表示されます。



ファイル・システムから直接シナリオを開くには、[File System] をクリックします。[Open Scenario] ダイアログ・ボックスが表示されます ([シナリオを開く] ダイアログ・ボックスから [Quality Center プロジェクトからシナリオを開く] ダイアログ・ボックスに戻るには、[Quality Center] をクリックします)。

- 3 テスト計画ツリーの関連するサブジェクトをクリックします。ツリーを展開してサブレベルを表示するには、閉じているフォルダをダブルクリックします。ツリーを折りたたむには、開いているフォルダをダブルクリックします。
サブジェクトを選択すると、そのサブジェクトに属しているシナリオが [Test Name] リストに表示されます。
- 4 [Test Name] リストからシナリオを選択します。読み取り専用の [TestName] ボックスにシナリオが表示されます。
- 5 [OK] をクリックして、シナリオを開きます。LoadRunner によってシナリオがロードされます。コントローラのタイトル・バーにシナリオ名が表示されま

す。[デザイン] タブに、シナリオ内のスクリプト、ロード・ジェネレータ、仮想ユーザ、仮想ユーザ・グループが表示されます。

注：シナリオは [ファイル] メニューの最近開いたシナリオのリストから開くこともできます。Quality Center プロジェクト内のシナリオを選択したときに、LoadRunner がそのプロジェクトに接続されていない場合は、[Quality Center への接続] ダイアログ・ボックスが表示されます。プロジェクトにログインするためにユーザ名とパスワードを入力し、[OK] をクリックします。

Quality Center プロジェクトへのシナリオの保存

LoadRunner が Quality Center プロジェクトに接続されている場合は、LoadRunner での新規シナリオの作成や、作成したシナリオのプロジェクトへの直接保存ができます。シナリオを保存するには、そのシナリオにわかりやすい名前を付け、テスト計画ツリーの対応するサブジェクトに関連付けます。これによって、各サブジェクトに対して作成されたシナリオを追跡したり、テストの計画と作成の進行状況を直ちに表示したりできます。

シナリオを Quality Center プロジェクトに保存するには、次の手順を実行します。

- 1 Quality Center サーバに接続します (192 ページ「Quality Center への LoadRunner の接続」を参照してください)。
- 2 コントローラで、[ファイル] > [名前を付けて保存] を選択します。[Quality Center プロジェクトへシナリオを保存] ダイアログ・ボックスが表示され、テスト計画ツリーが表示されます。



ファイル・システムに直接シナリオを保存するには、**[File System]** をクリックします。**[Save Scenario]** ダイアログ・ボックスが表示されます。(**[Save Scenario]** ダイアログ・ボックスから **[Quality Center プロジェクトにシナリオを保存]** ダイアログ・ボックスに戻るには、**[Quality Center]** をクリックします)。

- 3 テスト計画ツリーの関連するサブジェクトを選択します。ツリーを展開してサブレベルを表示するには、閉じているフォルダをダブルクリックします。サブレベルの表示を折りたたむには、開いたフォルダをダブルクリックします。
- 4 **[Test Name]** ボックスに、シナリオの名前を入力します。シナリオを容易に識別できるわかりやすい名前を使います。
- 5 **[OK]** をクリックしてシナリオを保存し、ダイアログ・ボックスを閉じます。

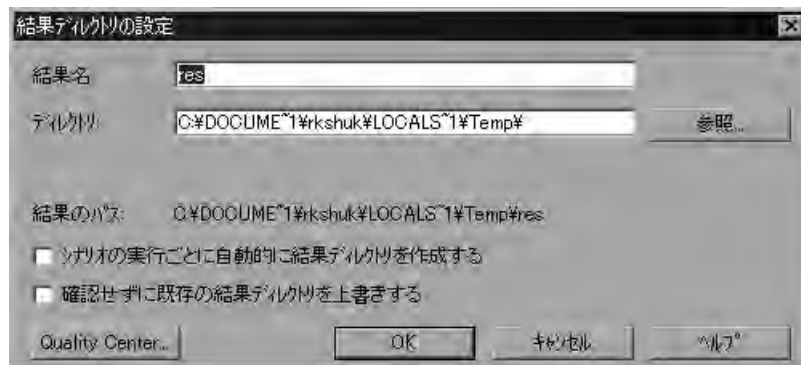
次に Quality Center を起動したときに、Quality Center のテスト計画ツリーに新しいシナリオが表示されます。

Quality Center プロジェクトへの結果の保存

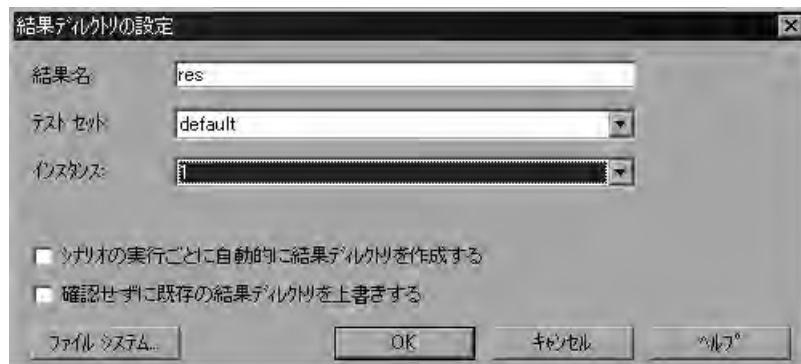
シナリオを実行する前に、結果の格納場所を設定します。LoadRunner が Quality Center プロジェクトに接続されている場合、実行結果はテスト・セットに保存されます。標準ファイル・システムを使って、結果をディスクに保存することもできます。

実行結果を Quality Center プロジェクトに保存するには、次の手順を実行します。

- 1 Quality Center サーバに接続します（192 ページ「Quality Center への LoadRunner の接続」を参照してください）。
- 2 コントローラで、**[結果]** > **[結果の設定]** を選択します。[結果ディレクトリの設定] ダイアログ・ボックスが表示されます。



- 3 **[Quality Center]** をクリックします。[ディレクトリ] ボックスが [テストセット] に変わります。



- 4 [結果名] ボックスに、結果の名前を入力します。
- 5 [テストセット] リストで、標準のテスト・セット名をそのまま受け入れるか、異なる名前を選択します。
- 6 次の適切なチェック・ボックスを選択します。
 - ▶ [シナリオの実行ごとに自動的に結果ディレクトリを作成する] : LoadRunner が、シナリオの実行ごとに固有の結果ディレクトリを作成するようにします。標準設定では、res1, res2, res3 などの結果名になります。
 - ▶ [確認せずに既存の結果ディレクトリを上書きする] : LoadRunner が、ユーザに確認せずに既存の結果セットを自動的に上書きするようにします。
- 7 [OK] をクリックして、結果ディレクトリの設定を保存します。

Quality Center プロジェクトからの仮想ユーザ・スクリプトの追加

Quality Center プロジェクトからコントローラのスクリプト・リストに仮想ユーザ・スクリプトを追加できます。このスクリプトは、マニュアル・シナリオまたはゴール指向シナリオのどちらにも追加できます。

マニュアル・シナリオへの仮想ユーザ・スクリプトの追加

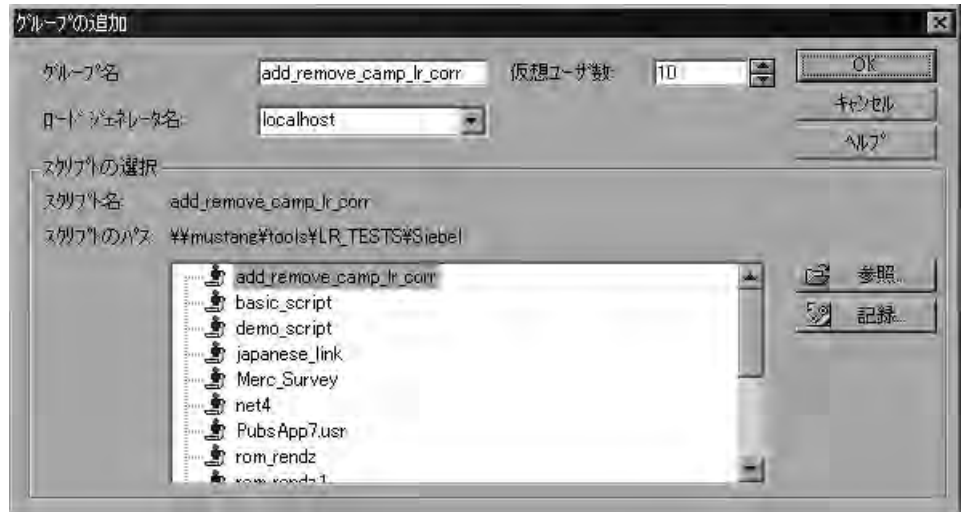
マニュアル・シナリオを作成している場合、[グループの追加] ダイアログ・ボックスを使って仮想ユーザ・スクリプトを追加できます。

マニュアル・シナリオに仮想ユーザ・スクリプトを追加するには、次の手順を実行します。

- 1 Quality Center サーバに接続します (192 ページ「Quality Center への LoadRunner の接続」を参照してください)。

- 2 [シナリオ グループ] 表示枠で、[**グループの追加**] ボタンをクリックします。
[グループの追加] ダイアログ・ボックスが表示されます。

*** グループの追加 (A)



- 3 [**参照**] をクリックします。[Open Test from Quality Center Project] ダイアログ・ボックスが表示され、テスト計画ツリーが表示されます。
- 4 スクリプトを選択し、[**OK**] をクリックします。[スクリプトのパス] フィールドに、[TD]、完全なサブジェクト・パス、およびスクリプト名が表示されます。
次に例を示します。

[TD]#Subject#System#test_qc

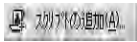
- 5 [**OK**] をクリックして [グループの追加] ダイアログ・ボックスを閉じます。
スクリプトが [シナリオのグループ] 表示枠に表示されます。

ゴール指向シナリオへの仮想ユーザ・スクリプトの追加

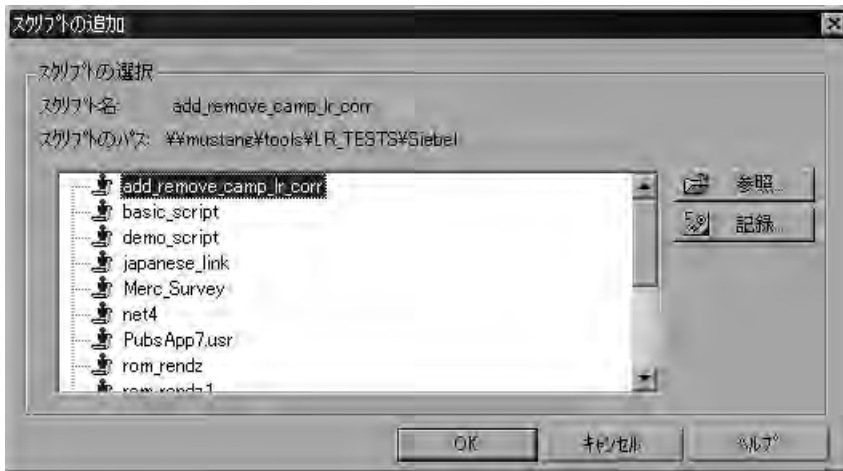
ゴール指向シナリオを作成している場合、[スクリプトの追加] ダイアログ・ボックスを使ってスクリプトを追加できます。

ゴール指向シナリオに仮想ユーザ・スクリプトを追加するには、次の手順を実行します。

- 1 Quality Center サーバに接続します (192 ページ「Quality Center への LoadRunner の接続」を参照してください)。



- 2 [シナリオ スクリプト] 表示枠で、[スクリプトの追加] ボタンをクリックします。[スクリプトの追加] ダイアログ・ボックスが開きます。



- 3 [参照] をクリックします。[Open Test from Quality Center Project] ダイアログ・ボックスが表示され、テスト計画ツリーが表示されます。
- 4 スクリプトを選択し、[OK] をクリックします。[スクリプトのパス] フィールドに、[TD]、完全なサブジェクト・パス、およびスクリプト名が表示されます。
次に例を示します。

[TD]¥Subject¥System¥test_qc

- 5 [OK] をクリックして [スクリプトの追加] ダイアログ・ボックスを閉じます。スクリプトが [シナリオのスクリプト] 表示枠の [スクリプトのパス] カラムに表示されます。

第 13 章

LoadRunner での機能テスト・スクリプトの使用

LoadRunner を使用して、大きな負荷がアプリケーションの機能にどのような影響を与えるかをテストし監視できます。LoadRunner は、機能テスト・スクリプトを GUI 仮想ユーザ・スクリプトの形式で負荷テスト・シナリオに統合できます。これらの GUI 仮想ユーザ・スクリプトは、Mercury の機能テスト製品である QuickTest Professional または WinRunner を使用して作成されます。

本章では、次の内容について説明します。

- ▶ LoadRunner での機能テスト・スクリプトの使用について
- ▶ GUI 仮想ユーザ・スクリプトについて
- ▶ QuickTest を使用した LoadRunner の GUI 仮想ユーザ・スクリプトの使用
- ▶ WinRunner を使用した LoadRunner の GUI 仮想ユーザ・スクリプトの使用
- ▶ LoadRunner シナリオでの仮想ユーザ・スクリプトの実行

LoadRunner での機能テスト・スクリプトの使用について

Mercury の機能テスト製品（QuickTest および WinRunner）を使用して、アプリケーションの機能を包括的に検査する複雑なテストを作成できます。

LoadRunner は、これらの機能テスト・スクリプトを GUI 仮想ユーザ・スクリプトの形式で負荷テスト・シナリオに統合できます。すでに QuickTest または WinRunner で設計されデバッグされているこうしたスクリプトは、負荷テストの基本として使用できます。

LoadRunner における負荷テスト・スクリプトの主な利点は次のとおりです。

- ▶ アプリケーションの機能が大きな負荷によってどのような影響を受けるか確認するため。

- ▶ アプリケーションに負荷が掛かっている状態で、クライアント側で一般的なユーザが体験する応答時間を測定する（エンド・ツー・エンドの応答時間）。

例えば、QuickTest または WinRunner テスト・スクリプトを LoadRunner シナリオの特定のポイントに追加し、それらのポイントでの追加の負荷によってアプリケーションの機能が影響を受けていないことを確認できます。

LoadRunner シナリオの一部として GUI 仮想ユーザ・スクリプトを使用することによる別の利点は、シナリオの実行中にスクリーン上で GUI 仮想ユーザ・スクリプトが実行されるので、リアルタイムで仮想ユーザによって実行されている実際のステップを監視できることです。

GUI 仮想ユーザ・スクリプトについて

GUI 仮想ユーザを使えば、クライアント/サーバ・システムに負荷をかけたときのエンド・ツー・エンドのユーザ側の応答時間の測定および監視が行えます。GUI 仮想ユーザは、実際のユーザの操作環境を完全にエミュレートします。

例えば、実際のユーザはマシンの前に座り、キーボードとマウスを使用してアプリケーションを操作し、モニタ画面の情報を読みます。これと同様に、GUI 仮想ユーザもそれぞれのマシンで実行され、アプリケーションを操作します。GUI 仮想ユーザをプログラミングし、モニタ画面に表示される情報を読み込んだり、操作したりできます。

現金自動預払い機（ATM）を管理している銀行のサーバを考えてみます。次のことを行う GUI 仮想ユーザ・スクリプトを作成できます。

- ▶ ATM アプリケーションを開く
- ▶ 口座番号を入力する
- ▶ 引き出す現金の金額を入力する
- ▶ 口座から現金を引き出す
- ▶ 口座の残高を確認する
- ▶ ATM アプリケーションを閉じる
- ▶ 処理を繰り返す

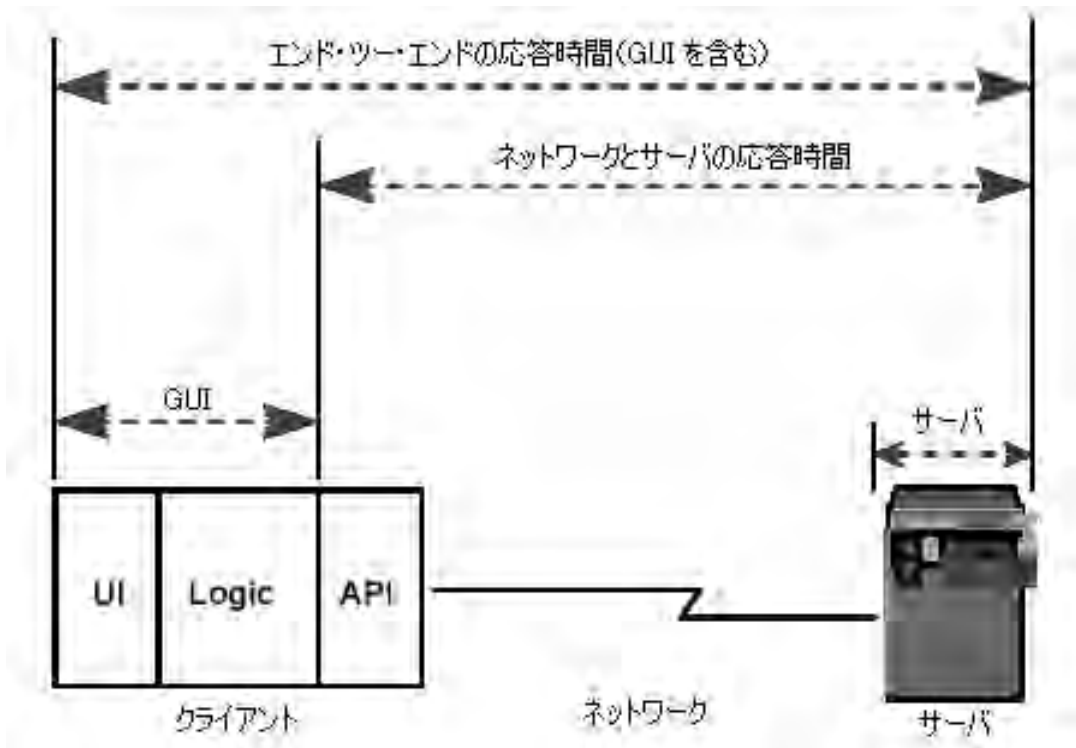
各 GUI 仮想ユーザのアクションは、「GUI 仮想ユーザ・スクリプト」に記述されます。QuickTest と WinRunner を使って、GUI 仮想ユーザ・スクリプトを作成します。

GUI 仮想ユーザの監視と管理は、LoadRunner コントローラを使って行います。例えば、コントローラを使って、仮想ユーザの実行、一時停止、表示およびシナリオのステータスの監視ができます。

注： VuGen では、GUI 仮想ユーザ・スクリプトを実行できません。GUI 仮想ユーザ・スクリプトをシナリオの一部として実行するには、コントローラを使います。スタンドアロン・モードで GUI 仮想ユーザ・スクリプトを実行するには、WinRunner または QuickTest を使います。

GUI 仮想ユーザ・テクノロジーについて

GUI 仮想ユーザは、エンド・ツー・エンドの実際の応答時間を測定します。エンド・ツー・エンドの応答時間は、ユーザが要求を出してから応答を得るまでの合計待ち時間を表します。エンド・ツー・エンドの応答時間には、GUI、ネットワーク、サーバそれぞれの応答時間が含まれます。



QuickTest を使用した LoadRunner の GUI 仮想ユーザ・スクリプトの使用

LoadRunner のテスト・シナリオで GUI 仮想ユーザ・スクリプトとして使用されるテスト・スクリプトを QuickTest で作成するには、特定のガイドラインにしたがって、スクリプトの統合を円滑に行う必要があります。QuickTest でのテストの作成方法の詳細については、『**QuickTest Professional ユーザーズ・ガイド**』を参照してください。

本項では、次の一般的なガイドラインについて説明します。

- ▶ トランザクションのインクルード
- ▶ テストへの統合データの保存

- ▶ ステートメントの追加
- ▶ LoadRunner 向けのテストの設計

制限事項

QuickTest には、LoadRunner と統合するために特別に設計されたいくつかの機能が提供されています。しかし、QuickTest の機能には、LoadRunner と統合すると使用できないものもあります。特定の制限事項の詳細については、「**QuickTest Professional 最初にお読みください**」を参照してください。

トランザクションのインクルード

サーバのパフォーマンスを測定するには、**トランザクション**を定義します。トランザクションは、測定対象のアクションまたはアクションの集合を表します。仮想ユーザ・スクリプト内でトランザクションを定義するには、スクリプトの適切なセクションをトランザクション・ステートメントの **start** と **end** で囲みます。

例えば、サーバが口座の残高表示要求を処理して、ATM に情報が表示されるのにかかる時間を測定するトランザクションを定義できます。

注：LoadRunner は、トランザクションに含まれるデータのパフォーマンス情報を提供するだけです。したがって、QuickTest テストは、LoadRunner によって使用されたトランザクションを含めなければなりません。



[挿入] > [トランザクション開始] および [挿入] > [トランザクション終了] メニュー・オプションまたはツールバー・ボタンを使用して「トランザクション開始」および「トランザクション終了」ステートメントを挿入できます。これらのオプションの詳細については、QuickTest Professional のマニュアルを参照してください。

テストへの統合データの保存

LoadRunner の仮想ユーザ技術との統合を可能にするために、QuickTest は特別な統合ファイルを生成する必要があります。標準設定では、このデータを生成するためのオプションは有効になっています。しかし、自分あるいは QuickTest テストで作業をしている別のユーザがディスク領域を節約するためにこのオプションを無効にしている可能性があります。LoadRunner で使用するテストの作成を開始する前に、次の手順を実行してこのオプションを有効にします。



- 1 QuickTest において、[ツール] > [オプション] を選択するか、[オプション] ツールバー・ボタンをクリックします。[オプション] ダイアログ・ボックスの [一般] タブが開きます。
- 2 [一般] タブで、[Mercury パフォーマンス テスティングおよびアプリケーション管理製品との統合のためにデータを保存する] が選択されていることを確認します。
- 3 このオプションが無効なときに保存されたテストを LoadRunner と統合するには、このオプションを有効にしてから対象のテストを開いて保存します。

ヒント：テストに統合データが保存されているかどうかを確認するには、テストの**更新日**が同じ**<テスト名>.usr** ファイルがテスト・フォルダにあることを確認します。

ステートメントの追加

Services オブジェクトと関連メソッドを使用して、パフォーマンス・テストに特に関連のあるステートメントを挿入できます。これらには、**Abort**、**GetEnvironmentAttribute**、**LogMessage**、**SetTransactionStatus**、**ThinkTime**、**UserDataPoint**、**StartTransaction** および **EndTransaction** があります。これらのメソッドの詳細については、『**QuickTest Professional オブジェクト・モデル・リファレンス**』の「**Utility**」の項を参照してください。

LoadRunner 向けのテストの設計

LoadRunner で使用するテストを設計するときは、次の設計ガイドラインに従ってください。

- ▶ LoadRunner で使用する QuickTest テストは、特定の操作に的を絞った簡単なテストでなければなりません。
- ▶ LoadRunner はネストされたアクションの反復を実行できません。
- ▶ 外部アクションや、外部データ・テーブル・ファイル、環境変数ファイル、共有オブジェクト・リポジトリなどの外部リソースへの参照を含めないでください。
- ▶ LoadRunner はトランザクションに含まれるデータのパフォーマンス情報のみを提供するので、QuickTest テストにはトランザクションを含めます。

WinRunner を使用した LoadRunner の GUI 仮想ユーザ・スクリプトの使用

本項では、WinRunner を使用して、LoadRunner テスト・シナリオで使用する GUI 仮想ユーザ・スクリプトを作成し、拡張する方法の概略を説明します。WinRunner でのテストの作成の詳細については、WinRunner のマニュアルを参照してください。

注： WinRunner の機能には、LoadRunner と統合すると使用できないものもあります。

GUI 仮想ユーザ・スクリプトについて

WinRunner は、Windows ベースの GUI 仮想ユーザ・スクリプトの作成、編集、デバッグを行うための完全な開発環境です。WinRunner を使用して、アプリケーションに対する実際のユーザのアクションを記録します。

例えば、ユーザが ATM に口座番号を入力し、50 ドル引き出す操作を記録できます。これらのアクションは、Mercury のテスト・スクリプト言語 (TSL) を使って、スクリプトに自動的に記録されます。

GUI 仮想ユーザ・スクリプトは、Mercury インタラクティブのテスト・スクリプト言語 (TSL) を使用して作成します。TSL は、C 言語に似た高度なプログラミング言語です。TSL は、強力かつ柔軟な従来のプログラミング言語の特性と、テスト用に設計された関数を組み合わせます。TSL の詳細については、「[TSL Online Reference](#)」を参照してください。

本項では、WinRunner で作成した基本的な仮想ユーザ・スクリプトを紹介します。このスクリプトは ATM アプリケーション (`mratm.exe`) を起動し、口座番号を入力し、50 ドル預金して、ATM アプリケーションを終了します。

注： このスクリプトは、UNIX マシンでは動作しません。

スクリプトの最初のセクションで、アプリケーションを起動し、そのアプリケーションを画面上の新しい場所に移動します。**system** 関数で、ATM アプリ

セッションを起動します。**win_move** 関数で、ATM アプリケーションを画面上の指定の場所に動かします。

```
# ATM クライアント・アプリケーションを初期化して起動する
invoke_application("c:\%mratm.exe", "", "", SW_SHOWMINIMIZED);
win_move ("Mercury ATM", 325, 0);
```

次に、仮想ユーザが口座番号を ATM アプリケーションに入力します。**set_window** 関数で ATM ウィンドウをアクティブにします。**edit_set** 関数で口座番号を ATM アプリケーションの口座番号のフィールドに入力するように仮想ユーザに指示します。

```
# Account フィールドに口座番号を入力する
account = 100;
set_window ("Mercury ATM");
edit_set ("Account", account);
```

口座番号を入力した後、仮想ユーザは預金する金額を入力し、[Deposit] ボタンを押します。**edit_set** 関数で、預金額を金額フィールドに入力します。**button_press** 関数は、仮想ユーザに ATM の [Deposit] ボタンを押すよう指示します。

```
# Amount フィールドに預金額を入力する
amount = 50;
set_window ("Mercury ATM");
edit_set ("Amount", amount);
# [Deposit] ボタンを押す。
button_press ("Deposit");
```

このテストの最後のセクションで、仮想ユーザに ATM アプリケーションを終了するよう指示します。**menu_select_item** 関数で [File] メニューから [Exit] コマンドを選択します。

```
# クライアント・アプリケーションを閉じる
menu_select_item ("File; Exit");
```

トランザクションとランデブー・ポイントの挿入

サーバのパフォーマンスを測定するには、**トランザクション**を定義します。トランザクションは、測定対象のアクションまたはアクションの集合を表しま

す。仮想ユーザ・スクリプト内でトランザクションを定義するには、スクリプトの適切なセクションをトランザクション・ステートメントの **start** と **end** で囲みます。例えば、サーバが口座の残高表示要求を処理して、ATM に情報が表示されるのにかかる時間を測定するトランザクションを定義できます。

注：LoadRunner は、トランザクションに含まれるデータのパフォーマンス情報のみを提供します。したがって、WinRunner テストには LoadRunner によって使用されるトランザクションを含める必要があります。

仮想ユーザ・スクリプトに **ランデブー・ポイント** を挿入して、サーバにかかる大きなユーザ負荷をエミュレートします。**ランデブー・ポイント** はテスト実行中に仮想ユーザを待機させ、複数の仮想ユーザが特定のポイントに到着したときに、それらが同時にタスクを実行するようにします。例えば、銀行のサーバのピーク時の負荷をエミュレートするためのランデブー・ポイントを挿入して、100 仮想ユーザに同時に、それぞれの口座に預金させることができます。

WinRunner を使って基本的な仮想ユーザ・スクリプトを作成した後、以下を手作業で挿入します。

- ▶ スクリプトにトランザクション・ステートメントを挿入し、サーバのパフォーマンスを測定します。
- ▶ ステートメントとスクリプトを同期化し、特定のユーザ負荷をエミュレートします。

トランザクションの開始を示すには、次の手順を実行します。

start_transaction ステートメントを仮想ユーザ・スクリプトに挿入します。

トランザクションの終了を示すには、次の手順を実行します。

end_transaction ステートメントを仮想ユーザ・スクリプトに挿入します。

start_transaction と **end_transaction** 関数の構文については、**TSL Online Reference** (WinRunner の [ヘルプ] メニューから表示できます) を参照してください。

ランデブー・ポイント を挿入するには、次の手順を実行します。

rendezvous ステートメントを仮想ユーザ・スクリプトに挿入します。

rendezvous 関数の構文の詳細については、「**TSL Online Reference**」(WinRunner の [ヘルプ] メニューから表示できます) を参照してください。

GUI 仮想ユーザ・スクリプトでの仮想ユーザ関数の使用

本項では、GUI 仮想ユーザ・スクリプトの拡張に使用できる仮想ユーザ関数のいくつかを示します。関数の構文と使用例については、本書中の関連する項、または「**TSL Online Reference**」(WinRunner の [ヘルプ] メニューから表示できます) を参照してください。

declare_rendezvous	ランデブーを宣言します。
declare_transaction	トランザクションを宣言します。
end_transaction	パフォーマンス分析を実行するためのトランザクションの終了位置を示します。
error_message	エラー・メッセージをコントローラに送信します。
get_host_name	ロード・ジェネレータの名前を返します。
get_master_host_name	コントローラ・ロード・ジェネレータの名前を返します。
lr_whoami	スクリプトを実行する仮想ユーザに関する情報を返します。
output_message	コントローラにメッセージを送信します。
rendezvous	仮想ユーザ・スクリプトにランデブー・ポイントを設定します。
start_transaction	パフォーマンス分析を実行するためのトランザクションの開始位置を示します。
user_data_point	ユーザ定義データのサンプルを記録します。

コントローラへのメッセージの送信

シナリオを実行すると、コントローラの [出力] ウィンドウに、スクリプトの実行に関するメッセージが表示されます。WinRunner によって自動的に送信されるメッセージに加え、エラー・メッセージや通知メッセージをコントローラに送信するステートメントを各スクリプトに挿入できます。例えば、アプリケーションの現在の状態を表示するメッセージを挿入できます。これらのメッセージはシナリオの実行後にファイルに保存できます。

error_message 関数は、エラー・メッセージをコントローラの [出力] ウィンドウに送信します。この関数の構文は次のとおりです。

error_message (message);

message にはテキスト文字列を指定します。

次の例では、スクリプトの実行中に致命的なエラーが発生したときに、仮想ユーザ・スクリプトがメッセージを送信します。

```
if (fatal_error < 0){
    mess="fatal error - Exiting.";
    error_message (mess);
    textit (1);
}
```

output_message 関数を使って、エラー・メッセージ以外の特別な通知を送ります。この関数の構文は次のとおりです。

output_message (message);

message にはテキスト文字列を指定します。

error_message 関数と **output_message** 関数の詳細については、**TSL Online Reference** (WinRunner の [ヘルプ] メニューから表示できます) を参照してください。

仮想ユーザとロード・ジェネレータに関する情報の取得

シナリオの実行中、次の ID を取得できます。

- ▶ シナリオ内のある特定の時点でタスクを実行している仮想ユーザ
- ▶ スクリプトを実行しているロード・ジェネレータ
- ▶ コントローラが稼働しているマシン

例えば、仮想ユーザ・スクリプト内にステートメントを記述し、現在アプリケーションを使用しているアクティブな仮想ユーザの ID を取得し、その情報をファイルに出力できます。

次の関数で仮想ユーザとロード・ジェネレータに関する情報を取得します。

lr_whoami	仮想ユーザ名と、その仮想ユーザが属する仮想ユーザ・グループを返します。
get_host_name	スクリプトを実行しているマシンの名前を返します。
get_master_host_name	コントローラを実行しているマシンの名前を返します。

次の例では、**get_host_name** 関数を使用して、スクリプトを実行しているロード・ジェネレータの名前を取得しています。その後、**print** ステートメントで、その情報をファイルに保存しています。

```
my_host_name = get_host_name();
print("my local load generator name is:" & my_host_name) > vuser_file;
```

これらの関数の詳細については、「[TSL Online Reference](#)」(WinRunner の [ヘルプ] メニューから表示できます) を参照してください。

基本的な仮想ユーザ・スクリプトを記録した後で、サーバのパフォーマンスを測定したり (トランザクション)、同期化されたユーザ負荷 (ランデブー・ポイント) を作成したりするためのステートメントをスクリプトに挿入します。GUI 仮想ユーザの詳細については、「[LoadRunner オンライン関数リファレンス](#)」を参照してください。

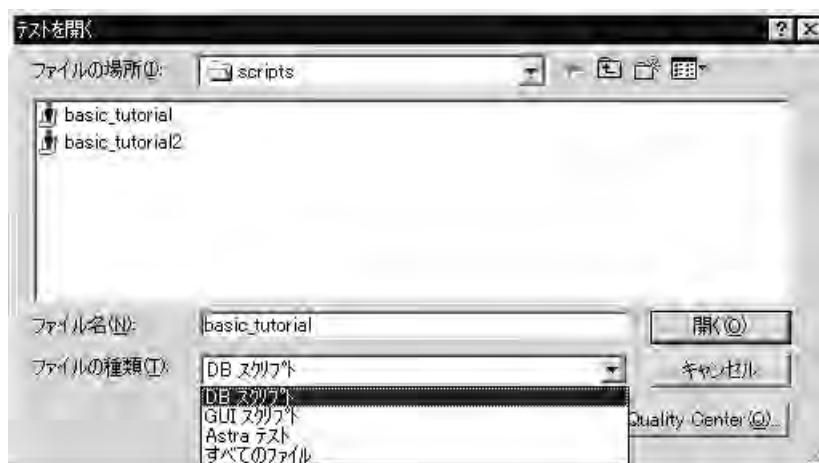
LoadRunner シナリオでの仮想ユーザ・スクリプトの実行

QuickTest または WinRunner で GUI 仮想ユーザ・スクリプトを作成したら、このスクリプトを LoadRunner シナリオに統合できます。LoadRunner シナリオを実行する前に、次のガイドラインについて検討してください。

- ▶ 1 台のコンピュータで同時に実行できる GUI 仮想ユーザは 1 つまでです。
- ▶ シナリオの実行前に QuickTest および WinRunner を必ず閉じておきます。
- ▶ [実行環境設定 スクリプト : <スクリプト名>] ダイアログ・ボックスの中で、[一般] カテゴリとそのサブカテゴリのみ ([General], [Iterations], [その他], [思考遅延時間]) が QuickTest および WinRunner テストに関係します。[Replay] オプションは関係ありません。

LoadRunner シナリオへの GUI 仮想ユーザ・スクリプトの追加は、次の手順を実行します。

- 1 テスト・スクリプトの含まれるフォルダへ移動します。
 - ▶ 新しいシナリオの場合は、[新規シナリオ] ダイアログ・ボックスで [参照] をクリックします。
 - ▶ 既存のシナリオにスクリプトを追加する場合は、[グループの追加] ダイアログ・ボックスで [参照] をクリックします。[テストを開く] ダイアログ・ボックスが表示されます。



- 2 [ファイルの種類] ボックスで、適切なファイルの種類を選択します。
 - ▶ WinRunner の場合は、「GUI スクリプト」を選択します。
 - ▶ QuickTest の場合は、「Astra テスト」を選択します。
- 3 適切なスクリプトに移動して、シナリオに追加します。

第 3 部

シナリオの実行

第 14 章

シナリオの実行

シナリオを実行すると、LoadRunner はテスト中のアプリケーションに対する負荷を生成し、システムのパフォーマンスを測定します。

本章では、次の項目について説明します。

- ▶ シナリオの実行について
- ▶ シナリオ全体の実行
- ▶ 仮想ユーザ・グループの制御
- ▶ 個々の仮想ユーザの制御
- ▶ ランデブーからの手作業による仮想ユーザの解放
- ▶ 実行中のシナリオへの手作業による仮想ユーザの追加

シナリオの実行について

シナリオを実行すると、仮想ユーザ・グループがロード・ジェネレータに割り当てられ、仮想ユーザ・スクリプトを実行します。シナリオ実行中には、LoadRunner では次のことが行われます。

- ▶ 仮想ユーザ・スクリプトで定義されているトランザクションの継続時間の記録
- ▶ 仮想ユーザ・スクリプトに含まれるランデブーの実行
- ▶ 仮想ユーザが生成するエラー、警告、および通知メッセージの収集

シナリオ全体を無人で実行することも、実行する仮想ユーザや仮想ユーザ・グループを選んで対話的に実行することもできます。シナリオの実行が開始すると、コントローラは最初にシナリオの構成情報を検査します。次に、シナリオで実行することが指定されているアプリケーションを起動します。それから、

各仮想ユーザ・スクリプトを指定のロード・ジェネレータに割り振ります。準備が完了すると、仮想ユーザ・グループはスクリプトの実行を開始します。

シナリオ実行中に、各仮想ユーザの監視、仮想ユーザが生成するエラー、警告、通知メッセージの表示、および仮想ユーザ・グループまたは個々の仮想ユーザの停止ができます。LoadRunner が、個々の仮想ユーザやグループ内の仮想ユーザを、それらが実行中の反復またはアクションが完了してから、あるいは直ちに停止するように指定できます。詳細については、174 ページ「シナリオ実行環境の設定」を参照してください。

注：ゴール指向シナリオで自動的に仮想ユーザを停止する場合、LoadRunner は仮想ユーザの実行を直ちに停止します。

また、シナリオの実行中に、[仮想ユーザの実行/停止] ダイアログ・ボックスを使用して、別の仮想ユーザを起動することもできます。詳細については、232 ページ「実行中のシナリオへの手作業による仮想ユーザの追加」を参照してください。

シナリオは、すべての仮想ユーザがスクリプトを完了するか、継続時間が終わるか、ユーザが停止することによって終了します。

次の手順は、シナリオ実行方法の概略です。

- 1 既存のシナリオを開くか、新しいシナリオを作成します。
- 2 シナリオの構成とスケジュールを設定します。
- 3 結果ディレクトリを設定します。
- 4 シナリオを実行し、監視します。

シナリオ全体の実行

シナリオのすべての仮想ユーザと仮想ユーザ・グループを実行できます。あるいは、実行する特定の仮想ユーザ・グループと仮想ユーザを選択できます。シナリオ全体を実行する場合、すべての仮想ユーザのステータスが「**準備完了**」になるまで、LoadRunner は仮想ユーザの実行を開始しません。ただし、個々の仮想グループまたは仮想ユーザを実行する場合は、仮想ユーザのステータスが「**準備完了**」になるとすぐに、LoadRunner は仮想ユーザを実行します。

次の項では、シナリオ全体の実行方法について説明します。222 ページ「仮想ユーザ・グループの制御」と 228 ページ「個々の仮想ユーザの制御」では、仮想ユーザ・グループと個々の仮想ユーザの操作方法について説明します。

シナリオ全体は、次の手順で実行します。

- 1 既存のシナリオを開くか、新しいシナリオを作成します。[実行] タブを選択します。画面の左上角に [シナリオ グループ] 表示枠が表示されます。



- 2 [シナリオ] > [開始] を選択するか、[シナリオの開始] をクリックします。コントロールは、仮想ユーザの初期化と指定されたロード・ジェネレータ（そこで仮想ユーザ・スクリプトの実行を開始します）への割り振りを開始します。

注： コントローラは、[シナリオ開始] ダイアログ・ボックスで設定された時間に従って、シナリオの実行を開始します。

シナリオの結果ディレクトリを指定していない場合、[結果ディレクトリの設定] ダイアログ・ボックスが開きます。

シナリオの実行中に、個々の仮想ユーザおよび仮想ユーザ・グループを操作できます。これについては、次の「仮想ユーザ・グループの制御」と 228 ページ「個々の仮想ユーザの制御」で説明します。

- 3 ランプ・アップ処理を停止するには、[シナリオ] > [ランプアップの停止 / 再開] を選択します。仮想ユーザのランプ・アップを再開するには、上記の操作を繰り返します。

- 4 ランプ・ダウン処理を停止するには、[シナリオ] > [ランプ ダウンの停止 / 再開] を選択します。仮想ユーザのランプ・ダウンを再開するには、上記の操作を繰り返します。

■ (停止)

- 5 [シナリオ] > [停止] を選択するか、[停止] をクリックするとシナリオが終了します。[オプション] ダイアログ・ボックスの [実行環境の設定] タブで [直ちに停止する] オプションが選択されている場合には、シナリオのすべての仮想ユーザのステータスが [終了中] に変わります。

[オプション] ダイアログ・ボックスの [実行環境の設定] タブで [停止前に現在の反復終了まで待機する] または [停止前に現在のアクション終了まで待機する] オプションが選択されている場合には、シナリオの仮想ユーザのステータスは [徐々に終了] に変わり、シナリオは徐々に停止します。仮想ユーザを直ちに停止するには、[直ちに停止する] をクリックします。

- 6 [シナリオ] > [リセット] を選択するか、[リセット] をクリックするとすべての仮想ユーザが、シナリオ実行前の状態（「ダウン」）に戻されます。

◀ (リセット)

仮想ユーザ・グループの制御

上で説明したようにシナリオ全体を実行できます。また、シナリオの個々の仮想ユーザの操作もできます。この項では、仮想ユーザ・グループの初期化、実行、一時停止、停止、およびリセットの方法を説明します。

仮想ユーザ・グループの初期化

仮想ユーザ・グループの初期化によって、グループの仮想ユーザは指定されたロード・ジェネレータに割り振られ、スクリプトを実行できるようになります。すべての仮想ユーザを実行前に初期化することによって、すべての仮想ユーザが同時にシナリオの実行を開始するようになります。

仮想ユーザ・グループを初期化するには、次の手順を実行します。

- 1 初期化する1つまたは複数の仮想ユーザ・グループを選択します。
- 2 ツールバーの [選択した仮想ユーザを初期化] をクリックするか、初期化する仮想ユーザ・グループ（1つまたは複数）を右クリックして [グループの初期化] を選択します。仮想ユーザ・グループのステータスが、[ダウン] から [保留中]、[初期化]、[準備完了] に変わります。仮想ユーザ・グループが初期化に失敗すると、仮想ユーザ・グループのステータスは [エラー] に変わります。



仮想ユーザ・グループの実行

仮想ユーザ・グループを実行すると、その仮想ユーザ・グループはスクリプトを実行します。

仮想ユーザ・グループは、次の手順で実行します。

- 1 実行する 1 つまたは複数の仮想ユーザ・グループを選択します。
- 2 ツールバーの **[選択した仮想ユーザを実行]** をクリックするか、実行する 1 つまたは複数の仮想ユーザ・グループを右クリックして **[グループでシナリオ開始]** を選択します。仮想ユーザ・グループがスクリプトを実行します。ステータスが **[ダウン]** または **[エラー]** の仮想ユーザ・グループを実行した場合には、LoadRunner は仮想ユーザ・グループを初期化してから実行します。



注：仮想ユーザ・グループを右クリックして **[仮想ユーザ 1 名でスクリプトを開始]** を選択することにより、LoadRunner がグループ内のランダムな 1 仮想ユーザだけを実行することができます。仮想ユーザ・スクリプト・ログが開き、その仮想ユーザの実行時の情報が表示されます。仮想ユーザ・ログの詳細については、248 ページ「仮想ユーザ・スクリプト・ログの表示」を参照してください。

仮想ユーザ・グループの一時停止

仮想ユーザ・グループを一時停止すると、スクリプトの実行も一時的に停止します。**[一時停止]** コマンドにより、仮想ユーザ・グループのステータスは **[実行中]** から **[一時停止]** に変わります。

注：仮想ユーザ・グループを一時停止すると、そのトランザクション応答時間に影響します。

仮想ユーザを一時停止するには、次の手順を実行します。

- 1 一時停止する 1 つまたは複数の仮想ユーザ・グループを選択します。
- 2 右クリックで表示されるメニューから **[一時停止]** を選択します。仮想ユーザ・グループは一時的にスクリプトの実行を停止します。

仮想ユーザ・グループの停止

仮想ユーザ・グループを停止すると、スクリプトの実行も停止します。仮想ユーザ・グループを停止しても、そのグループは仮想ユーザ・グループ・リストに表示されたままです。

仮想ユーザ・グループの停止は、次の手順で行います。

- 1 停止する1つまたは複数の仮想ユーザ・グループを選択します。



- 2 ツールバーの「**選択したユーザを停止**」をクリックするか、1つまたは複数の仮想ユーザ・グループを右クリックして「**停止**」を選択します。仮想ユーザ・グループは、直ちにスクリプトの実行を停止します。

[オプション] ダイアログ・ボックスの[実行環境の設定]タブで「**停止前に現在の反復終了まで待機する**」または「**停止前に現在のアクション終了まで待機する**」オプションが選択されているときに、[実行]ステータスの仮想ユーザ・グループを徐々に停止する場合には、ツールバーの「**選択した仮想ユーザを徐々に停止**」ボタンをクリックするか、仮想ユーザ・グループを右クリックして「**徐々に停止**」を選択します。グループ内の仮想ユーザのステータスは「**徐々に終了**」に変わり、シナリオは徐々に停止します。



注：仮想ユーザが「**実行**」ステータスでない場合、「徐々に停止」の選択肢は無効になっています。

仮想ユーザ・グループのリセット

仮想ユーザ・グループをリセットすると、グループ内の仮想ユーザがすべてシナリオ実行前の状態（「**ダウン**」ステータス）に戻されます。

仮想ユーザ・グループをリセットするには、次の手順を実行します。

- 1 停止する1つまたは複数の仮想ユーザ・グループを選択します。
- 2 停止する1つまたは複数の仮想ユーザ・グループを右クリックし、「**グループのリセット**」を選択します。仮想ユーザ・グループはシナリオ実行以前の状態、つまり「**ダウン**」ステータスに戻ります。

【実行】ビューのタブについて

【実行】タブには、【シナリオグループ】ウィンドウ、シナリオ・ステータスのサマリ、スクリプトの実行中に生成されたオンライン情報を示すグラフが表示されます。

【シナリオグループ】：仮想ユーザ・グループと、それぞれの現在のステータスが表示されます。

- ▶ 【シナリオの開始】：コントローラに対して、シナリオ内の仮想ユーザの初期化と指定されたロード・ジェネレータ（そこで仮想ユーザ・スクリプトの実行を開始します）への割り振りを開始するように指示します。

注：コントローラは、【シナリオ開始】ダイアログ・ボックスで指定された時間に、シナリオの実行を開始します。負荷テスト実行中は、コントローラ・マシンとロード・ジェネレータ・マシンで時刻 / 日付および時差の設定を変更しないことをお勧めします。

- ▶ 【停止】：コントローラに対して、シナリオを停止するように指示します。
【オプション】ダイアログ・ボックスの【実行環境の設定】タブで【直ちに停止する】オプションが選択されている場合には、シナリオのすべての仮想ユーザのステータスが【終了中】に変わります。【オプション】ダイアログ・ボックスの【実行環境の設定】タブで【停止前に現在の反復終了まで待機する】または【停止前に現在のアクション終了まで待機する】オプションが選択されている場合には、シナリオの仮想ユーザのステータスは【徐々に終了】に変わり、シナリオは徐々に停止します。
- ▶ 【停止】：コントローラに対して、シナリオを直ちに停止するよう指示します。
- ▶ 【リセット】：シナリオ内のすべての仮想ユーザ・グループをシナリオ実行以前の状態、つまり【ダウン】ステータスに戻します。
- ▶ 【仮想ユーザ】：【仮想ユーザ】ダイアログ・ボックスを開きます。このダイアログ・ボックスには、仮想ユーザ・グループ内の各仮想ユーザのステータスが表示されます。
- ▶ 【実行 / 停止】：【仮想ユーザの実行 / 停止】ダイアログ・ボックスを開きます。このダイアログ・ボックスで、仮想ユーザを追加して実行できます。

[シナリオ グループ] ウィンドウ内でグループを右クリックすると、シナリオ内の個々の仮想ユーザ・グループを対象に以下の操作を実行できます。

- ▶ **[番号の再割り当て]**：グループ内の各仮想ユーザの番号を再割り当てし、それぞれの ID 番号を変更します。
- ▶ **[グループの初期化]**：グループの仮想ユーザを指定されたロード・ジェネレータに割り振り、スクリプトを実行できるようにします。仮想ユーザ・グループのステータスが、「**ダウン**」から「**保留中**」、「**初期化**」、「**準備完了**」に変わります。仮想ユーザ・グループが初期化に失敗すると、仮想ユーザ・グループのステータスは「**エラー**」に変わります。すべての仮想ユーザを実行前に初期化することによって、すべての仮想ユーザが同時にシナリオの実行を開始するようにできます。
- ▶ **[グループでシナリオ開始]**：仮想ユーザ・グループがスクリプトを実行します。ステータスが「**ダウン**」または「**エラー**」の仮想ユーザ・グループを実行した場合には、LoadRunner は仮想ユーザ・グループを初期化してから実行します。
- ▶ **[仮想ユーザ 1 名でスクリプトを開始]**：コントローラに対して、仮想ユーザ・グループ内のランダムな 1 仮想ユーザを実行するように指示します。仮想ユーザ・ログが開き、その仮想ユーザの実行時の情報が表示されます。
- ▶ **[一時停止]**：シナリオの実行を一時的に停止します。仮想ユーザ・グループのステータスが、「**実行中**」から「**一時停止**」に変わります。


注：仮想ユーザ・グループを一時停止すると、そのトランザクション応答時間に影響します。

- ▶ **[徐々に停止]**：コントローラに対して、現在の反復またはアクションを完了してから仮想ユーザ・グループの実行を停止するように指示します。

注：[オプション] ダイアログ・ボックスの [実行環境の設定] タブで [停止前に現在の反復終了まで待機する] または [停止前に現在のアクション終了まで待機する] オプションを選択した場合、仮想ユーザ・グループが「**実行**」ステータスになっているときにだけ、このオプションを使用できます。

- ▶ **[停止]**：コントローラに対して、仮想ユーザ・グループの実行を直ちに停止するように指示します。

- ▶ **[グループのリセット]**：グループ内のすべての仮想ユーザをシナリオ実行以前の状態、つまり **[ダウン]** ステータスに戻します。
- ▶ **[イベント認識]**：仮想ユーザ・グループを有効にし、シナリオに参加できるようにします。
- ▶ **[無効]**：仮想ユーザ・グループを無効にし、シナリオから除外します。
- ▶ **[仮想ユーザを表示]**：グループ内の各仮想ユーザについて実行時ビューアを開きます。
- ▶ **[仮想ユーザを非表示]**：開いている実行時ビューアを閉じます。
- ▶ **[仮想ユーザ ログを表示]**：グループ内の各仮想ユーザの実行時の情報を表示するスクリプト・ログを開きます。標準設定では、仮想ユーザ・スクリプト・ログは 1000 ミリ秒ごとに更新されます。
- ▶ **[仮想ユーザ ログを非表示]**：仮想ユーザ・スクリプト・ログを閉じます。
- ▶ **[名前で並べ替え]**：グループを名前のアルファベット順に並べ替えます。

 **[シナリオ ステータス]**：実行中のシナリオの概要が表示されます。拡大鏡のアイコンをクリックして、個々のトランザクションおよびエラーの詳細を表示できます。

注： 右上角のボタンをクリックして、**[シナリオ ステータス]** ウィンドウを **[実行]** ビューから切り離せます。これにより、**[シナリオ グループ]** ウィンドウを拡大表示できます。

[利用可能なグラフ]：利用可能なグラフのリストを表示するには、**[表示]** > **[利用可能なグラフの表示 / 非表示]** を選択します。グラフ・ツリー・ビューを非表示にするには、**[表示]** > **[利用可能なグラフの表示 / 非表示]** を選択するか、**[利用可能なグラフ]** リストの右上隅にある **[×]** ボタンをクリックします。

グラフを表示するには、左側の表示枠内のグラフをクリックして、右側の表示枠までドラッグします。標準設定では、4 つのグラフが表示されます。オンライン・グラフ表示をカスタマイズするには、**[表示]** > **[グラフを表示]** をクリックし、表示するグラフの数を選択します。最高で 16 のグラフを同時に表示できます。グラフを 1 つだけ表示するには、右側の表示枠内で、そのグラフをダブルクリックします。前の表示に戻るには、グラフを再度ダブルクリックします。グラフの下には、選択したグラフの統計値が表示される凡例がありま

す。この凡例に表示されるのは、色、目盛り、測定値 / ステータス、マシン、最大値、最小値、平均値、標準偏差、最新の値です。

[モニタ] メニューを使用するか、グラフを右クリックすることによって、グラフに対して次のアクションを実行できます。

- ▶ グラフの設定
- ▶ グラフの複製
- ▶ グラフの固定 / 解放
- ▶ グラフの HTML へのエクスポート
- ▶ グラフの合成

測定値を右クリックして、測定値を対象に次の操作を実行できます。

- ▶ 測定値の追加 / 削除
- ▶ 測定値の表示 / 非表示
- ▶ 測定値の設定
- ▶ 測定値の説明

個々の仮想ユーザの制御

定義した仮想ユーザ・グループ内の個々の仮想ユーザを操作できます。この項では、個々の仮想ユーザの初期化、実行、停止の方法を説明します。

個々の仮想ユーザを制御するには、次の手順を実行します。

- 1 仮想ユーザ・グループを選択して、**[仮想ユーザ]** ボタンをクリックします。
[仮想ユーザ] ダイアログ・ボックスが開き、グループの各仮想ユーザの [ID], [ステータス], [スクリプト], [ロード ジェネレータ], [経過時間] (シナリオ開始以降) が表示されます。

仮想ユーザ(1)



次のユーティリティを使って、個々の仮想ユーザを制御します。

- ▶ 仮想ユーザを選択して **[実行]** をクリックすると、仮想ユーザが実行されます。
- ▶ 仮想ユーザを選択して **[停止]** をクリックすると、仮想ユーザの実行が直ちに停止されます。
- ▶ [オプション] ダイアログ・ボックスの [実行環境の設定] タブで **[停止前に現在の反復終了まで待機する]** または **[停止前に現在のアクション終了まで待機する]** オプションを選択されているときに、[実行] ステータスの仮想ユーザ・グループを徐々に停止するには、**[徐々に停止]** ボタンをクリックします。仮想ユーザのステータスが **[徐々に終了]** に変わり、シナリオは徐々に停止します。
- ▶ 仮想ユーザを一時停止するには、その仮想ユーザを右クリックをして **[一時停止]** を選択します。仮想ユーザを一時停止すると、そのトランザクション応答時間に影響します。
- ▶ 仮想ユーザのステータスを「ダウン」に戻すには、その仮想ユーザを選択して **[リセット]** をクリックします。

- ▶ 仮想ユーザを初期化するには、右クリックしてから **[仮想ユーザの初期化]** を選択します。
- ▶ グループの仮想ユーザの番号を再割り当てするには、対象の仮想ユーザを右クリックして **[番号の再割り当て]** を選択します。
- ▶ リスト内の仮想ユーザをフィルタするには、カラムの1つを右クリックして **[仮想ユーザのフィルタ]** を選択します。仮想ユーザのフィルタ方法を選択します。**[仮想ユーザ]** ダイアログ・ボックスの右上角のプルダウン・リストでフィルタ・オプションを選択することもできます。
- ▶ リスト内の仮想ユーザを並べ替えるには、カラムの1つを右クリックして **[仮想ユーザの並べ替え]** を選択します。仮想ユーザの並べ替え方法を選択します。
- ▶ 割り当てられたスクリプトを実行している仮想ユーザを表示するには、対象の仮想ユーザを選択して、**[選択した仮想ユーザを表示]** ボタンをクリックします。実行時ビューアが開いて仮想ユーザに返されたページのスナップショットが表示され、そのスクリプトを実行している仮想ユーザの状況を確認できます。実行時ビューアは、ブラウザとしては機能しません。したがって、表示される画像はスナップショットであり、再生のすべての面を示すものではありません。
- ▶ **[オプション]** メニューの項目を使用すると、表示するコントロールの種類を選択できます。
- ▶ **[表示]** メニューの項目を使用すると、さまざまなツールバーとビューを開くことができます。
- ▶ 実行時ビューアを閉じるには、**[選択した仮想ユーザを非表示]** ボタンをクリックします。



- ▶ 仮想ユーザ・スクリプト・ログを表示するには、**[仮想ユーザ ログを表示]** ボタンをクリックします。スクリプト・ログは、次のように表示されます。



- ▶ 仮想ユーザ・スクリプト・ログを閉じるには、**[閉じる]** ボタンをクリックします。仮想ユーザ・スクリプト・ログの詳細については、248 ページを参照してください。

2 **[閉じる]** をクリックして、**[仮想ユーザ]** ダイアログ・ボックスを閉じます。

ランデブーからの手作業による仮想ユーザの解放

シナリオの実行中に、コントローラが解放する前にランデブーから仮想ユーザを手作業で解放できます。

ランデブーから手作業で仮想ユーザを解放するには、次の手順を実行します。

- 1 **[シナリオ]** > **[ランデブー]** を選択します。**[ランデブー情報]** ダイアログ・ボックスが表示されます。
- 2 **[ランデブー]** リストからランデブーを選択します。
- 3 **[解放]** をクリックします。ランデブーの仮想ユーザが解放されます。



実行中のシナリオへの手作業による仮想ユーザの追加

シナリオの実行中に、[仮想ユーザの実行/停止] ダイアログ・ボックスを使用して、新規仮想ユーザの追加を手作業で制御できます。このダイアログ・ボックスは、実行しているシナリオ・モードにより異なります。

- ▶ 仮想ユーザ・グループ・モードで実行している場合、各仮想グループに追加できる新規仮想ユーザの数、および追加する仮想ユーザが実行するロード・ジェネレータを制御できます。
- ▶ パーセント・モードで実行している場合、定義するパーセントに基いて、仮想ユーザ・スクリプトに分散できる新規仮想ユーザの数を制御できます。また、追加する仮想ユーザが実行するロード・ジェネレータも制御できます。

注：スケジュール・ビルダの設定を使って、シナリオまたは仮想ユーザ・グループを実行すると、これらの設定は、シナリオ実行中にシナリオまたは仮想ユーザ・グループに手作業で追加されたすべての仮想ユーザに適用されます。詳細については、159 ページ「スケジュールされたシナリオへの仮想ユーザの追加」を参照してください。

実行中のシナリオに仮想ユーザを追加するには、次の手順を実行します。

- 1 [シナリオ] > [仮想ユーザの実行/停止] を選択するか、[実行] ビューの [シナリオ グループ] 表示枠にある [実行/停止] をクリックします。[仮想ユーザの実行/停止] ダイアログ・ボックスが表示されます。仮想ユーザ・グループ・モードの場合、ダイアログ・ボックスにはシナリオに含まれる仮想ユーザ・グループが表示されます。

実行/停止 (E)



パーセント・モードの場合、[仮想ユーザの実行/停止] ダイアログ・ボックスにはシナリオに含まれる仮想ユーザ・スクリプトが表示されます。



- 仮想ユーザ・グループ・モードの場合、[数] カラムの各グループに対して実行する仮想ユーザの数を入力します。

パーセント・モードの場合、検査済み仮想ユーザ・スクリプト間で配分する仮想ユーザの数および割合を入力します。LoadRunner が入力された仮想ユーザの数を自動的に配分します。

- 仮想ユーザ・グループまたはスクリプトを無効にするには、グループまたはスクリプト名の左側にあるチェック・ボックスをクリアします。グループまたはスクリプトは [デザイン] ビューで無効になっていれば、自動的に無効と表示されます。

注：仮想ユーザ・グループ・モードで仮想ユーザ・グループを無効にすると、仮想ユーザは配分されません。パーセント・モードで仮想ユーザ・スクリプトを無効にすると、そのスクリプトには仮想ユーザが配分されず、無効にしたスクリプトの割合を 0 に定義しなければ、未使用の仮想ユーザの割合は、残りのスクリプトに配分されません。

- 仮想ユーザ・グループまたはスクリプトを実行するロード・ジェネレータを変更するには、[ロード ジェネレータ] カラムで現在のものとは異なるロード・ジェネレータを選択します。

リストに表示されていないロード・ジェネレータを使用するには、[ロードジェネレータ名] リストから [**<追加 ... >**] を選択します。[新規ロードジェネレータの追加] ダイアログ・ボックスが表示されます。新しいロード・ジェネレータの名前、プラットフォーム、一時ディレクトリを追加し、[OK] をクリックします。



パーセント・モードの場合、複数のロード・ジェネレータを選択して、仮想ユーザ・スクリプトを実行できます。ロード・ジェネレータ名のリストからロード・ジェネレータを選択し、[OK] をクリックします。このリストのすべてのロード・ジェネレータを使用するには、[**全ジェネレータ**] ボタンをクリックします。

注：複数のロード・ジェネレータがスクリプト用に定義されている場合、追加された仮想ユーザは定義されているロード・ジェネレータ間で均等に配分されます。

- 5 [初期化] をクリックして、追加した仮想ユーザの数を初期化します。
- 6 [実行] をクリックして、実行オプションを選択します。
- 7 [停止] をクリックして、[仮想ユーザの実行 / 停止] ダイアログ・ボックスに定義されているロード・ジェネレータで実行中の仮想ユーザを停止します。
- 8 [閉じる] をクリックして、[仮想ユーザの実行 / 停止] ダイアログ・ボックスを閉じます。

手作業によるシナリオ制御の例

次の例では、パーセント・モードで実行しているシナリオの [仮想ユーザの実行 / 停止] ダイアログ・ボックスを示します。



検査済みスクリプトで配分される仮想ユーザの数は 15 です。[%] カラムを見ると、これらの仮想ユーザの 60% がスクリプト **flights2002** に配分され、20% が **travel** および **test1** の両方に配分されるように指定されていることがわかります。この割合に従い、[数] カラムには、9 仮想ユーザが **flights2002** に配分され、3 仮想ユーザが **travel** と **test1** に配分されることが示されています。

注：無効にされたスクリプト **test1** に割り当てられている未使用の仮想ユーザは、このスクリプトに割り当てられる割合が定義されているため、残りのスクリプトには配分されません。

アクション ([**初期化**], [**実行**], [**停止**]) が [仮想ユーザの実行 / 停止] ダイアログ・ボックスから選択されている場合、コントローラでは [数] カラムで指定されている数の仮想ユーザが実行されます。この例では、9 人の仮想ユーザが **flights2002** スクリプトで、また 3 人の仮想ユーザが **travel** スクリプトで、初期化、実行または停止されます。

flights2002 スクリプトに配分されるすべての仮想ユーザは、**localhost** ロード・ジェネレータで実行されます。**travel** スクリプトの場合、仮想ユーザは、定義されたすべてのロード・ジェネレータ間で均等に配分されます。

注：仮想ユーザ・スクリプトに割り当てられたロード・ジェネレータがほかにある場合、ロード・ジェネレータの分散は、手作業で制御されているパーセント・モードのシナリオに適用されます。詳細については、140 ページ「負荷分散」を参照してください。

[仮想ユーザの実行 / 停止] ダイアログ・ボックスについて

[仮想ユーザの実行 / 停止] ダイアログ・ボックスでは、追加の仮想ユーザを手作業で起動できます。このダイアログ・ボックスは、シナリオが仮想ユーザ・グループ・モードとパーセント・モードのどちらで実行されているかによって異なります。

[グループごとに仮想ユーザ数を指定してください。]：各グループで実行する仮想ユーザの数を [数] カラムに入力します (仮想ユーザ・グループ・モード)。

[分配 X すべてのスクリプト内の仮想ユーザ]：印を付けた仮想ユーザ・スクリプトの間でパーセンテージに応じて配分する仮想ユーザの数を入力します。LoadRunner は、入力された数の仮想ユーザを自動的に配分します (パーセント・モード)。

仮想ユーザの実行 / 停止テーブル

- ▶ **[グループ名 / スクリプト名]**：シナリオで実行されている仮想ユーザ・グループまたはスクリプトの名前が表示されます。
- ▶ **[%]**：各仮想ユーザ・スクリプトに配分される仮想ユーザの割合を示します (パーセント・モードのみ)。
- ▶ **[数]**：各仮想ユーザ・スクリプトに配分される仮想ユーザの数を示します。
- ▶ **[ロード ジェネレータ]**：仮想ユーザが実行される場となるロード・ジェネレータを示します。1つのスクリプト用に複数のロード・ジェネレータが定義されている場合、追加された仮想ユーザは定義されているロード・ジェネレータ間で均等に配分されます。

注：仮想ユーザ・グループまたはスクリプトを無効にするには、グループまたはスクリプト名の左側にあるチェック・ボックスをクリアします。グループまたはスクリプトは [デザイン] ビューで無効になっていれば、自動的に無効と表示されます。仮想ユーザ・スクリプトを無効にすると、そのスクリプトに対して仮想ユーザは配分されません。ただし、無効にしたスクリプトに対する割合として 0 パーセントを定義していない限り、残りのスクリプトの間で 100 パーセントの仮想ユーザが配分されることはありません。

[初期化]：追加された仮想ユーザを指定されたロード・ジェネレータに配分し、スクリプトを実行できるようにします。コントローラは最初に、シナリオ内の実行されていない仮想ユーザを、現在のダイアログ・ボックスで定義されているロード・ジェネレータで初期化します。次に、現在のダイアログ・ボックスで定義されている仮想ユーザ数に達するまで、必要に応じて仮想ユーザが追加されます。

[実行]：次のいずれかのオプションを選択します。

- ▶ **[初期化して実行]**：現在のダイアログ・ボックスで定義されているロード・ジェネレータですでに初期化された、シナリオ内の仮想ユーザを実行します。コントローラは、仮想ユーザの数に関係なく、すでに初期化されている仮想ユーザのみ実行します。
- ▶ **[新規実行]**：指定された数の仮想ユーザを実行します。コントローラは最初に、シナリオ内の実行されていない仮想ユーザを、現在のダイアログ・ボックスで定義されているロード・ジェネレータで実行します。次に、現在のダイアログ・ボックスで定義されている仮想ユーザ数に達するまで、必要に応じて仮想ユーザが追加されます。

[停止]：現在のダイアログ・ボックスで定義されているロード・ジェネレータで実行中の仮想ユーザを停止します。コントローラは、174 ページ「シナリオ実行環境の設定」で定義された設定に従って、仮想ユーザを停止します。

第 15 章

実行中の仮想ユーザの表示

シナリオの実行中に、仮想ユーザが実行するアクションを表示できます。

本章では、次の項目について説明します。

- ▶ 実行中の仮想ユーザの表示について
- ▶ 仮想ユーザのステータスの監視
- ▶ [出力メッセージ] ウィンドウの表示
- ▶ 仮想ユーザ・スクリプト・ログの表示
- ▶ 実行メモの記録
- ▶ エージェントのサマリ表示

実行中の仮想ユーザの表示について

LoadRunner では、シナリオ実行中に仮想ユーザの動作状況を表示できます。

- ▶ コントローラのロード・ジェネレータ・マシンで、[出力メッセージ] ウィンドウの表示、オンラインでの仮想ユーザのパフォーマンスの監視、シナリオを実行している仮想ユーザのステータスの検査が可能です。
- ▶ リモート・マシンで、アクティブな仮想ユーザについての情報を含むエージェントのサマリを表示できます。

仮想ユーザのステータスの監視

シナリオ実行中、[実行] ビューの [シナリオ グループ] 表示枠を使って、シナリオのすべての仮想ユーザおよび仮想ユーザ・グループのアクションを監視できます。

各仮想ユーザの [ステータス] フィールドに、仮想ユーザの現在のステータスが表示されます。次の表で、シナリオ実行中の仮想ユーザのステータスについて説明します。

ステータス	説明
ダウン	仮想ユーザはダウンしている。
保留中	仮想ユーザは、初期化できる状態にあり、使用可能なロード・ジェネレータを待機中か、ロード・ジェネレータにファイルを転送中。仮想ユーザは、スケジュール属性で設定されている条件が満たされると実行される。
初期化	仮想ユーザは、リモート・マシン上で初期化中。
準備完了	仮想ユーザはスクリプトの INIT セクションを実行済みで実行可能な状態。
実行	仮想ユーザは実行中。仮想ユーザ・スクリプトがロード・ジェネレータで実行されている。
ランデブー	仮想ユーザはランデブーに到着し、LOADRUNNER によって解放されるのを待っている。
成功	仮想ユーザは実行を終了し、スクリプトの実行は成功した。
失敗	仮想ユーザは実行を終了し、スクリプトの実行は失敗した。
エラー	仮想ユーザにエラーが発生した。エラーの詳細については、[仮想ユーザ] ダイアログ・ボックスの [ステータス] フィールドか、エラーの詳細が表示される [出力メッセージ] ウィンドウを参照してください。
徐々に終了	仮想ユーザは、実行中の反復またはアクションを終了してから停止する ([ツール] > [オプション] > [実行環境の設定] で定義)。
終了中	仮想ユーザは実行を終えたか、停止されて終了中。
中止	仮想ユーザは [停止] コマンドによって実行を中止された。

さらに、[実行] ビューの右上角のボックスに、実行中のシナリオの概要が表示されます。

シナリオ ステータス	
実行中の仮想ユーザ	10
経過時間	00:01:12 (時間-分-秒)
秒ごとのヒット数	4.37 (最終 60 秒)
成功したトランザクション	10
失敗したトランザクション	0
エラー	0

注： 右上角のボタンをクリックして、[シナリオ ステータス] ウィンドウを [実行] ビューから切り離せます。これにより、[シナリオ グループ] 表示枠を拡大表示できます。

次の表で、シナリオの実行中に表示される実行時間情報について説明します。

ステータス・サマリ	説明
シナリオ・ステータス	シナリオが「実行中」か「ダウン」かを示します。
実行中の仮想ユーザ	ロード・ジェネレータのマシンで実行されている仮想ユーザ数を示します。
経過時間	シナリオ開始からの経過時間を示します。
秒ごとのヒット数	各仮想ユーザが実行しているテスト対象 WEB サイトへのヒット数 / 秒 (HTTP 要求) を示します。
成功したトランザクション	実行が成功したトランザクション数を示します。
失敗したトランザクション	実行に失敗したトランザクション数を表示します。
エラー	仮想ユーザで発生した問題の数を示します。

トランザクション

個々のトランザクションの詳細は、[トランザクション] ダイアログ・ボックスに表示できます。[トランザクション] ダイアログ・ボックスを開くには、

[シナリオステータス] ウィンドウの [成功したトランザクション] または [失敗したトランザクション] の右側にある [スナップショットを表示] ボタンをクリックします。



名前	TPS	成功	失敗	中止
BookFlight_Transaction	0.4	18	0	0
ShowItinerary_Transaction	1.3	54	0	0
vuser_end_Transaction	0.2	8	0	0
vuser_init_Transaction	0.2	10	0	0

[名前] : スクリプト内の個々のトランザクションの名前が表示されます。

[TPS] : 秒ごとのトランザクション数が表示されます。

[成功] : 成功したトランザクションの数が表示されます。

[失敗] : 失敗したトランザクションの数が表示されます。

[中止] : 中止されたトランザクションの数が表示されます。

[出力メッセージ] ウィンドウの表示

シナリオの実行中に、仮想ユーザとロード・ジェネレータは、エラー、通知、警告、デバッグ、バッチ・メッセージをコントローラに送信します。これらのメッセージは [出力メッセージ] ウィンドウに表示できます。

LoadRunner は、各シナリオの実行を開始するときに、[出力メッセージ] ウィンドウのメッセージを消去します。シナリオをリセットする場合、リセット時に [出力メッセージ] ウィンドウのメッセージを削除するよう LoadRunner に指示しない限り、[出力メッセージ] ウィンドウにメッセージが残ります。詳細については、付録 C 「オプション - 出力の設定」を参照してください。

[出力メッセージ] ウィンドウにメッセージを表示するには、次の手順で行います。

- 1 **[表示]** > **[出力メッセージを表示]** を選択するか、**[エラー]** リストの右側にある **[スナップショットを表示]** ボタンをクリックします。**[出力メッセージ]** ウィンドウが開き、エラー情報が表示されます。



- 2 **[メッセージの種類]** ボックスで、フィルタに付けるメッセージの種類を選択します。
- 3 メッセージの詳細を表示するには、そのメッセージを選択して、**[詳細]** ボタンをクリックします。**[出力メッセージ]** ウィンドウのボックス **[詳細メッセージテキスト]** ボックスに、完全なメッセージ・テキストが表示されます。



- 4 メッセージ、仮想ユーザ、スクリプト、またはロード・ジェネレータごとのログ情報の詳細を表示するには、それぞれのカラムの青いリンクをクリックします。詳細については、244 ページ「ログ情報の詳細表示」を参照してください。

ログ情報の詳細表示

各メッセージ、仮想ユーザ・スクリプト、およびエラー・コードに関連しているロード・ジェネレータの詳細は、それぞれのカラムの青いリンクをクリックして表示できます。[出力メッセージ] ウィンドウの [フィルタ済み] タブには、ドリル・ダウンされたビューが仮想ユーザのメッセージ、スクリプト、またはロード・ジェネレータごとに表示されます。

例えば、[Vuser] カラムをドリル・ダウンすると、全メッセージが、選択したコードとともに、メッセージを送信した仮想ユーザによってグループ分けされて [出力メッセージ] ウィンドウに表示されます。

Message	Script	Action	Line No.	Time	Iterati.	User	Generator
vuser_initc(12):	test1	vuser_in	12	2005/07/28 10:4	0	test1:4	localhost
vuser_initc(12):	test1	vuser_in	12	2005/07/28 10:4	0	test1:1	localhost
vuser_initc(12):	test1	vuser_in	12	2005/07/28 10:4	0	test1:2	localhost
vuser_initc(12):	flights2c	vuser_in	12	2005/07/28 10:4	0	flights2002:9	localhost
vuser_initc(12):	flights2c	vuser_in	12	2005/07/28 10:4	0	flights2002:2	localhost
vuser_initc(12):	flights2c	vuser_in	12	2005/07/28 10:4	0	flights2002:10	localhost
vuser_initc(12):	flights2c	vuser_in	12	2005/07/28 10:4	0	flights2002:11	localhost
vuser_initc(12):	flights2c	vuser_in	12	2005/07/28 10:4	0	flights2002:12	localhost
vuser_initc(12):	flights2c	vuser_in	12	2005/07/28 10:4	0	flights2002:1	localhost
vuser_initc(12):	flights2c	vuser_in	12	2005/07/28 10:4	0	flights2002:3	localhost

メッセージの種類、メッセージ・コード、およびドリル・ダウンを選択したカラムがグリッドの上に表示されます。

青で表示されているエントリは、さらにドリル・ダウンすることができます。仮想ユーザをドリル・ダウンすると、仮想ユーザのログが表示されます。ロード・ジェネレータをドリル・ダウンすると、選択したロード・ジェネレータが表示されている [ロード ジェネレータ] ダイアログ・ボックスが開きます。スクリプト (または [Action] か [Line Number]) をドリル・ダウンすると、VuGen が起動し、選択したスクリプトが表示されます。

注：ドリル・ダウンする際に表示される行の数を制限するには、任意のテキスト・エディタで **wlrun7.ini** ファイルを開き、次の行を探します。

MaxOutputUIRowsToShow=0

0（制限なし）を表示する行数に変更します。

[出力メッセージ] ウィンドウに新しいメッセージが到着すると、[更新] ボタンが有効になります。[更新] をクリックして、新しいログ情報を [フィルタ済み] タブのビューに追加します。



いろいろなドリル・ダウン階層間を移動するには、[出力メッセージ] ウィンドウの左上隅にある [前のビュー] ボタンと [次のビュー] ボタンをクリックします。

[出力メッセージ] ウィンドウについて






シナリオの実行中に仮想ユーザとロード・ジェネレータがコントローラに送信したエラー、通知、警告、デバッグ、およびバッチ・メッセージが表示されます。送信されたメッセージの総数はタイトル・バーに表示されます。

注：また、コントローラ・マシンで同時に表示する仮想ユーザ・ログの最大数を指定できます。詳細については、付録 C「エキスパート・モードでの作業」を参照してください。

[サマリ] タブ

[サマリ] タブには、シナリオ実行中に送信されたメッセージに関するサマリ情報が表示されます。エラー・コードに関連付けられた各メッセージ、仮想ユーザ・スクリプト、およびロード・ジェネレータの詳細を表示するには、それぞれのカラムの青いリンクをクリックします。

[**メッセージの種類**]：特定の種類のメッセージだけを表示するために、出力メッセージにフィルタを適用します。次のフィルタのいずれかを選択します。

アイコン	メッセージの種類	説明
	全メッセージ	すべての種類のメッセージを表示します。
	通知	例えば <code>LR_OUTPUT_MESSAGE</code> を使って送信されたメッセージなど、実行時の情報を提供します。
	エラー	通常は、スクリプトの実行が失敗したことを示します。
	警告	仮想ユーザが問題に遭遇したが、テストの実行は継続されたことを示します。
	デバッグ	デバッグ・メッセージは、[ツール] > [オプション] > [デバッグ情報] (エキスパート・モード) を選択してデバッグ機能を有効にした場合にだけ送信されます。詳細については、「オプション - [デバッグ情報] タブ (エキスパート・モードのみ)」を参照してください。
	バッチ	自動化機能を使用している場合に、コントローラでのメッセージ・ボックス表示の代わりに送信されます。

[**詳細**]：選択された出力メッセージの完全なテキストを [出力メッセージ] ウィンドウに表示します。



[**ビューをエクスポート**]：出力ビューを指定されたファイルに保存します。



[**すべてのメッセージを削除**]：すべてのログ情報を [出力メッセージ] ウィンドウから消去します。

[**固定 / 再開**]：[出力メッセージ] ウィンドウのメッセージの更新を停止します。LoadRunner に、[出力メッセージ] ウィンドウの更新を再開させるには、[再開] ボタンをクリックします。新しく更新されたログ情報は赤い枠で囲んで表示されます。

[**メッセージコード**]：類似したすべてのメッセージに割り当てられているコードが表示されます。括弧内の数は、[出力メッセージ] ウィンドウに表示される異なるコードの数を示します。

[**サンプル メッセージ テキスト**]：指定されたコードが割り当てられているメッセージのテキストの例が表示されます。

[**メッセージ合計**]：指定されたコードが割り当てられているメッセージが送信された総数が表示されます。

[**仮想ユーザ**]：指定されたコードが割り当てられているメッセージを生成した仮想ユーザの数が表示されます。

[**スクリプト**]：実行によって、指定されたコードのメッセージが生成されたスクリプトの数が表示されます。

[**ジェネレータ**]：指定されたコードが割り当てられているメッセージが生成されたロード・ジェネレータの数を表示します。

注：ログ情報を並べ替えるには、該当するカラム見出しをクリックします。メッセージは降順または昇順に並べ替えられます。

[フィルタ済み] タブ

[フィルタ済み] タブには、ドリル・ダウンされたビューが、仮想ユーザのメッセージ、スクリプト、またはロード・ジェネレータごとに表示されます。例えば、[仮想ユーザ] カラムをドリル・ダウンすると、全メッセージが、選択したコードとともに、メッセージを送信した仮想ユーザによってグループ分けされて [フィルタ済み] タブに表示されます。



[**前のビュー / 次のビュー**]：さまざまなドリル・ダウン階層間を移動できます。

[**メッセージの種類**] **アイコン**：現在の出力ビューに対するフィルタとして使用されているメッセージの種類を示すアイコンが表示されます。

[**使用中のフィルタ**]：現在の出力ビューに対するフィルタとして使用されているカテゴリが表示されます。

[**表示対象**]：ドリル・ダウンの対象として選択されたカラムの名前が表示されます。



[**ビューをエクスポート**]：出力ビューを指定されたファイルに保存します。

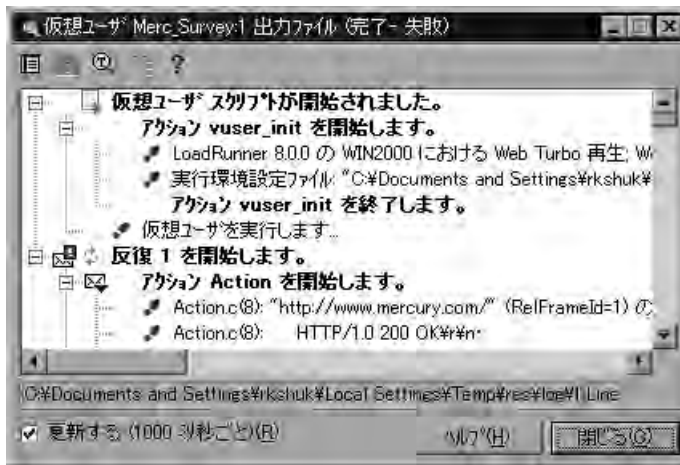
[**更新**]：[出力メッセージ] ウィンドウで受信された新規のログ情報を [フィルタ済み] タブのビューに追加します。

仮想ユーザ・スクリプト・ログの表示

シナリオの実行中、実行中の各仮想ユーザに関する実行時の情報が格納されているログを表示できます。

特定の仮想ユーザの仮想ユーザ・スクリプト・ログの表示は、次の手順で行います。

- 1 [仮想ユーザ] ダイアログ・ボックスで、ログを表示する仮想ユーザを選択し、**[仮想ユーザ ログを表示]** ボタンをクリックするか、仮想ユーザを右クリックして **[仮想ユーザ ログを表示]** を選択します。仮想ユーザ・スクリプトのログが開き、仮想ユーザについての実行時の情報が表示されます。この情報は、標準設定では 1000 ミリ秒ごとに更新されます。



更新間隔の設定については、396 ページ「オプション - 出力の設定」を参照してください。

- 2 **[閉じる]** をクリックして、仮想ユーザ・スクリプト・ログを閉じます。

仮想ユーザ・スクリプト・ログについて

仮想ユーザ・スクリプトのログには、仮想ユーザについての実行時の情報が表示されます。この情報は、標準設定では 1000 ミリ秒ごとに更新されます。

注： [実行環境の設定] ダイアログ・ボックスの [ログ] タブでログ機能が無効に設定されている場合、スクリプトに `lr_output_message` または `lr_message` 関数が含まれている場合にのみ、仮想ユーザ・スクリプト・ログに出力が含まれます。 [ログ] タブの [エラー発生時のみメッセージを送信する] オプションを選択した場合、仮想ユーザ・スクリプト・ログには、スクリプト・エラーが発生した場合のみ出力が格納されます。



[**テキストビューを表示**]：実行時の情報をテキスト形式で表示します。ツリー・ビューに戻すには、このボタンを再度クリックします。



[**ツリービューを表示**]：実行時の情報をツリー形式で表示します。テキスト・ビューに戻すには、このボタンを再度クリックします。



[**表示**]：仮想ユーザ・ログで強調表示したエラーが発生した Web ページのスナップショットを表示します。

注： エラーが発生した Web ページのスナップショットを表示するには、シナリオを実行する前に [実行環境の設定] ダイアログ・ボックスの [一般] タブで [エラー時にスナップショットを生成する] オプションを選択する必要があります。



[**テキスト検索**]：仮想ユーザ・ログ内で検索するテキストを入力します。



[**ノードの展開**]：仮想ユーザに関する詳細な実行時の情報を表示できるように、ノードを展開します。折りたたまれた状態のツリー・ビューに戻すには、このボタンを再度クリックします。



[**ノードの折りたたみ**]：ノードを折りたたみます。ツリー・ビューを元の大きさに戻すには、同じボタンを再度クリックします。

[**更新する (1000 ミリ秒ごと)**]：LoadRunner に対して、表示された実行時の情報を 1000 ミリ秒ごとに更新するように指示します。ログの更新を無効にするには、[更新する (1000 ミリ秒ごと)] チェック・ボックスをクリアします。

注：更新間隔の設定については、396 ページ「オプション - 出力の設定」を参照してください。

[コピー]：仮想ユーザ・ログからテキストをコピーできます。仮想ユーザ・ログ内で選択したテキストを右クリックし、[コピー]を選択します。

[Copy path from status bar]：仮想ユーザ・ログのパスをコピーできます。ステータス・バー内のパスを右クリックし、[Copy path from status bar]を選択します。

仮想ユーザのログ・メッセージのアイコン：



[仮想ユーザ スクリプトが開始されました]：仮想ユーザ・スクリプトの開始を示します。



[反復 X を開始します。]：反復の開始を示します。



[反復 X を終了します。]：反復の終了を示します。



[トランザクション X を開始しました。 / 終了しました。]：トランザクションの開始または終了を示します。



[アクション]：アクションの名前と説明が表示されます。



[通知]：アクション情報を提供します。



[エラー]：問題に遭遇したが、テスト実行を継続した仮想ユーザを示します。エラー・コードとエラーの詳細が表示されます。

実行メモの記録

コントローラは、シナリオ実行中にコメントを記録するためのダイアログ・ボックスを提供します。



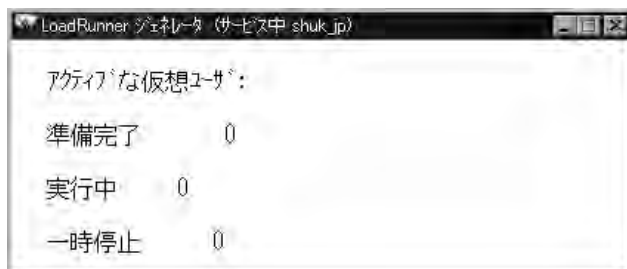
実行メモの記録は、次の手順で行います。

- 1 [シナリオ] > [実行メモ] を選択します。[実行メモ] ダイアログ・ボックスが表示されます。
- 2 記録するメモを入力します。
- 3 [閉じる] をクリックして、ダイアログ・ボックスを閉じます。LoadRunner は、記録したメモを保存します。

エージェントのサマリ表示

GUI 仮想ユーザ以外を使ってシナリオを実行すると、仮想ユーザを実行しているマシンが、そのロード・ジェネレータ上で仮想ユーザの実行を制御するエージェントを起動します。シナリオの実行中、エージェントは [準備完了], [実行中], [一時停止] のサマリを表示します。

エージェント・ウィンドウがシナリオの開始時に開きます。このウィンドウは、いつでも最小化または復元できます。



第4部

ファイアウォールを使った作業

第 16 章

LoadRunner でのファイアウォールの使用

コントローラがファイアウォールの外側にある場合でも、ファイアウォール内での仮想ユーザの実行やサーバの監視が可能です。

本章では、次の項目について説明します。

- ▶ LoadRunner でのファイアウォールの使用について
- ▶ システムの設定
- ▶ ファイアウォール越しの仮想ユーザの実行または監視
- ▶ 接続の確認

LoadRunner でのファイアウォールの使用について

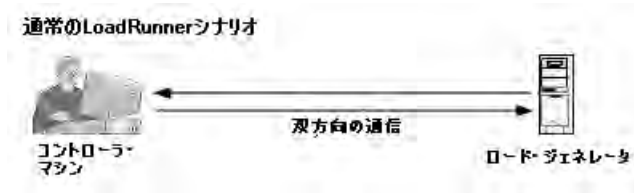
ファイアウォールを使用すると、プライベート・ネットワークへの不正アクセスとプライベート・ネットワークからの不正アクセスを、ポート番号ごとに防ぐことができます。

例えば、メール・ポート（23）を除いたすべてのポートに外部からアクセスできないように指定したり、メール・ポートと Web ポート（80）を除いたすべてのポートから外部への接続を許可しないように指定したりできます。ポートの設定は、システム管理者によって指定されています。

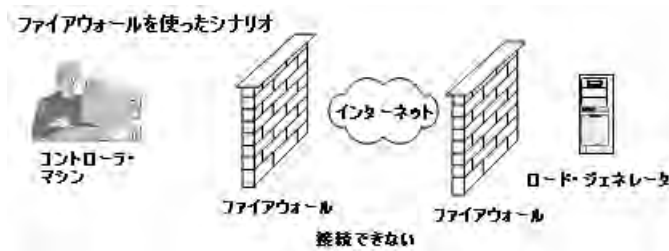
通常の LoadRunner シナリオ（ファイアウォール越しの処理を行わないシナリオ）では、コントローラは、リモート・マシンで実行中の LoadRunner エー

第4部・ファイアウォールを使った作業

ジェントに直接アクセスできます。これにより、コントローラはリモート・マシンに直接接続できます。



ファイアウォール越しに仮想ユーザを実行したり、サーバを監視したりする場合、直接の接続はファイアウォールによって阻止されます。コントローラにはファイアウォールに通過ポートを開ける権限がないため、接続を確立できません。

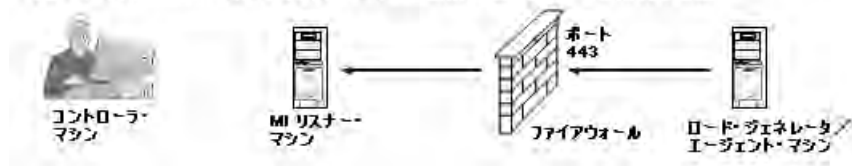


LoadRunner は、ファイアウォール上の標準 SSL ポート（ポート 443）を使用する HTTPS またはセキュア TCP/IP に基づいた通信メカニズムを使用することで、この問題を解決します。HTTPS および TCP/IP システム設定の詳細については、258 ページ「システムの設定」を参照してください。

LoadRunner エージェントは、ファイアウォールの内側にある、仮想ユーザを実行するロード・ジェネレータ・マシンか、監視対象サーバへの中継マシンとして動作するエージェント・マシン（「メディアータ」と呼ばれます）にインストールされます。エージェントは、ファイアウォールのポート 443 経由で、リスナー・マシンである MI リスナーと通信します。

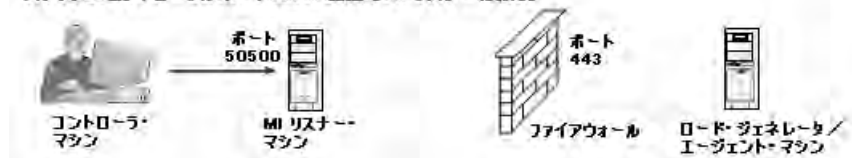
MI リスナーは、コントローラと LoadRunner エージェントの間でルータの役割を果たすコンポーネントです。

ステップ1: LoadRunner エージェントがポート443経由でMI リスナーへの接続を確立



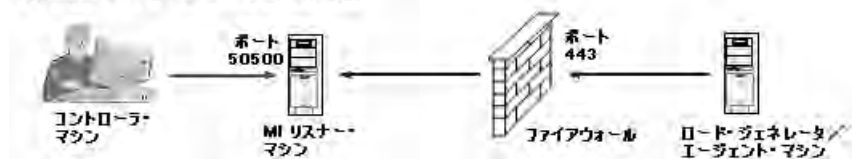
LoadRunner エージェントが MI リスナーへの接続を確立すると、MI リスナーはエージェントから渡されたシンボル名を使って、エージェントへの接続のリストを維持します。コントローラは、MI リスナーに接続するときに、ポート 50500 を経由して MI リスナーと通信します。

ステップ2: コントローラがポート50500経由でMI リスナーに接続



コントローラは、エージェントのシンボル名を使用し、MI リスナー・マシンの名前を渡します。同じシンボル名を持つエージェントからその MI リスナーへの接続が過去にあった場合には、接続が確立されます。エージェントに接続されたら、ファイアウォール越しに仮想ユーザの実行や監視ができるようになります。

ステップ3: コントローラとエージェントが接続

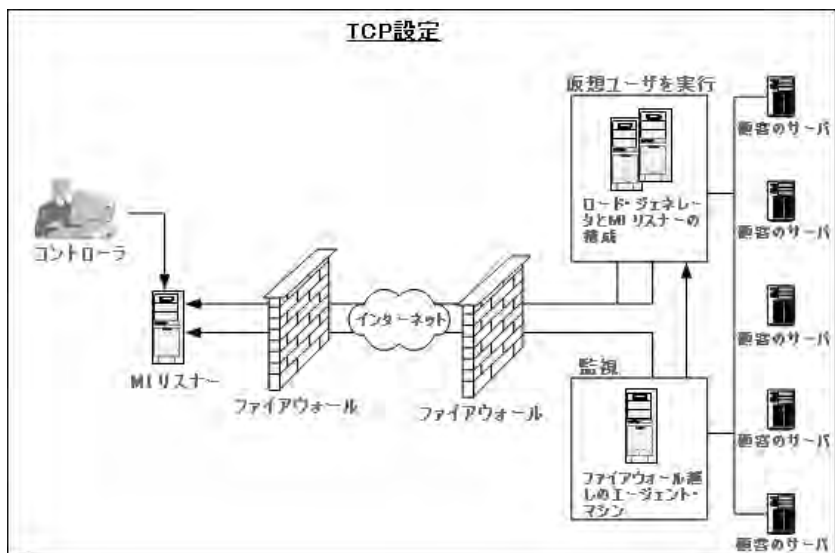


システムの設定

ファイアウォール越しに仮想ユーザの実行やサーバの監視を行うには、HTTPS またはセキュア TCP/IP 構成に従って、システムを設定します。これらの構成では、各 LAN のファイアウォールがあります。LAN1 にだけファイアウォールが存在する構成も考えられます。

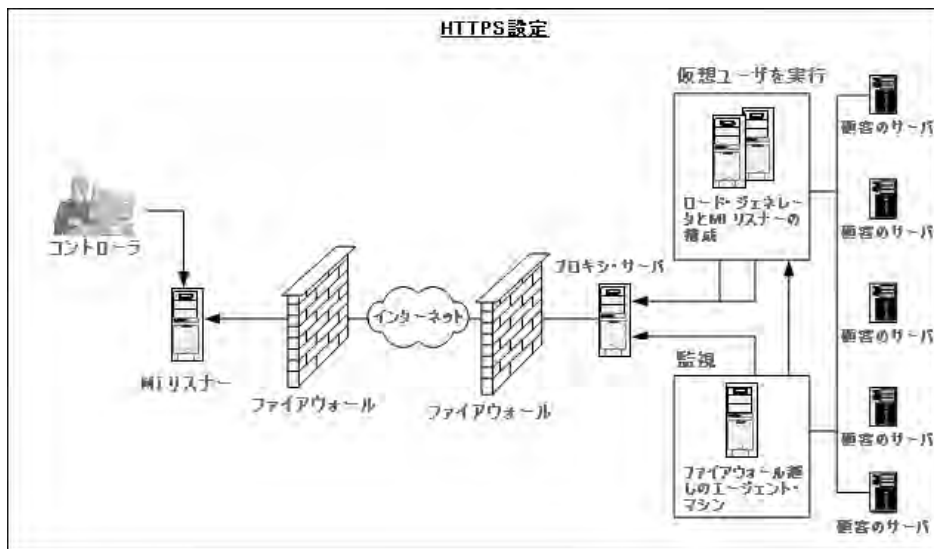
TCP 設定

TCP 設定では、ファイアウォールの内側にあるすべての LoadRunne エージェント・マシンが、ファイアウォールに送信用のポートを開く許可を持っている必要があります。



HTTPS 設定

HTTPS 設定では、1 台のマシン（プロキシ・サーバ）だけが、ファイアウォールのポートを開くことができます。したがって、すべての送信をプロキシ・サーバ経由でルーティングする必要があります。



注：MI リスナーと LoadRunner エージェントを使用せずに、ファイアウォール越しにコントローラをロード・ジェネレータに接続できます。これを行うには、ロード・ジェネレータの LAN とコントローラの LAN でファイアウォールのポート 54345 をオープンにし、データを送受信できるようにします。

ファイアウォール越しの仮想ユーザの実行または監視

負荷テストを成功させるには、ファイアウォール越しに仮想ユーザの実行とサーバの監視を行う能力が不可欠です。ファイアウォール越しに仮想ユーザの実行やサーバの監視ができるように LoadRunner の準備を整えるには、次の手順に従ってインストール、設定、接続を行う必要があります。

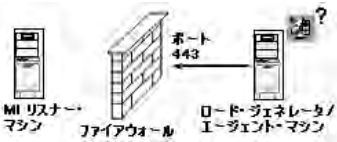
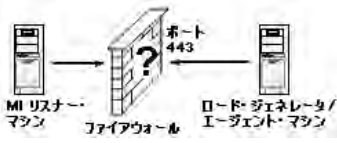
LoadRunner でファイアウォール越しに仮想ユーザを実行するためには、第17章「ファイアウォール越しの実行」を参照してください。

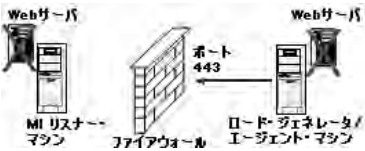
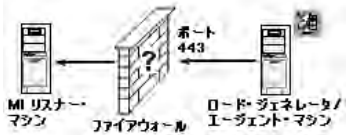
LoadRunner でファイアウォール越しにサーバを監視するには、第18章「ファイアウォール越しの監視」を参照してください。

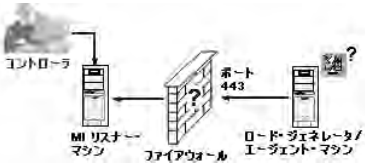
接続の確認

ファイアウォール越しに仮想ユーザの実行やサーバの監視を行うには、LoadRunner エージェント、MI リスナー、およびコントローラ・マシンの間に接続を確立できる必要があります。

インストール後やすべての必要なコンポーネントの設定後に接続の問題が発生した場合は、次の表のトラブルシューティングに関するヒントを参照してください。

確認	解決策
<p>エージェント・マシンでファイアウォール・サービスが起動されていることを確認するには</p>  <p>MI リスナーマシン ファイアウォール ポート 443 ロード・ジェネレータ/エージェント・マシン</p>	<p>ファイアウォール越しに仮想ユーザを実行または監視しているマシンの LoadRunner エージェントのアイコンの右側には、交通信号が表示されているはずですが、交通信号がない場合は、[エージェント設定] の [FireWall] セクションで、「FirewallServiceActive=1」が設定されていません。265 ページ「Windows での LoadRunner エージェントの設定と実行」を参照してください。</p>
<p>ポート 443 が開いていることを確認するには</p>  <p>MI リスナーマシン ファイアウォール ポート 443 ロード・ジェネレータ/エージェント・マシン</p>	<p>エージェント・マシンで [コマンドプロンプト] ウィンドウを開き、次のように入力します。telnet <MI リスナーの IP アドレス> 443。次に例を示します。 telnet 111.111.111.1111 443。</p> <p>ポート 443 が開いていれば、新規の Telnet ウィンドウが開きます。ポート 443 が開いていない場合は、ネットワーク管理者に問い合わせてください。</p>

確認	解決策
<p>ポート 443 が使用可能かどうか確認するには</p> 	<p>MI リスナーまたは「ファイアウォール越しの監視」マシンで Web サーバが動作している場合、リスンと監視のプロセスに必要なアクセスをポート 443 を通じて行うことはできません。ネットワーク管理者に連絡して、Web サーバ用のポートを変更してください。</p>
<p>LoadRunner エージェントをサービスとして実行している場合に、エージェントと MI リスナーの間の接続を確認するには</p> 	<p>LoadRunner エージェントをサービスとして実行しているときに、LoadRunner エージェントのアイコンの右側に赤信号が表示された場合は、次の対策を実行します。</p> <ul style="list-style-type: none"> ● ポート 443 が開いているかどうか確認します。上記のヒントを参照してください。 ● [エージェント設定] と [Agent Configuration] が正しく設定されているかどうか確認します。265 ページ「Windows での LoadRunner エージェントの設定と実行」を参照してください。 ● エージェントをプロセスとして実行します。 < LoadRunner のインストール先フォルダ > %Launch_service%bin%agentproc.exe を起動します。これでうまくいった場合は、LoadRunner エージェント・サービスに認証の問題があることを意味します。[Service] > [LoadRunner Agent Service] までブラウズし、このサービスのプロパティを System User Account に変更するか、当該マシンに対する管理者権限を持つ人のユーザ名とパスワードを指定します。

確認	解決策
<p>ファイアウォール越しに監視を行う場合に、エージェントとコントローラとの接続を確認するには</p> 	<ul style="list-style-type: none"> • [モニタ設定] ダイアログ・ボックスで監視対象のサーバを指定してあるかどうか確認します (283 ページ「サーバ・モニタ・プロパティの設定」を参照してください)。 • メディエータで LoadRunner エージェント・プロセスを起動します (265 ページ「ファイアウォールの内側での LoadRunner エージェントの設定」を参照してください)。 • コントローラで、メディエータの名前を [ロードジェネレータ] ダイアログ・ボックスに入力し、[接続] をクリックします。およそ 1 分後、データがメディエータから MI リスナーを通じてコントローラに流れ込み始めます (263 ページ「ファイアウォール越しの仮想ユーザの実行について」を参照してください)。 • データがコントローラに到着しない場合は、MI リスナーをロード・ジェネレータとして使用するかのように、コントローラを MI リスナーに接続してみてください。これにより、問題の原因を特定しやすくなります。LoadRunner エージェントのアイコンを右クリックして、メディエータ上のログ・ファイルを調べます。ログ・ファイルの中にエラー・メッセージはないはずです。 • MI リスナーを起動した後、メディエータで < LoadRunner のインストール先フォルダ > %launch_service%bin%magnetproc.exe を実行して LoadRunner エージェント・プロセスを手作業で起動します。メディエータが MI リスナーに接続するまで待つから、コントローラをメディエータに接続します。LoadRunner エージェント・プロセスがクラッシュした場合は、エージェントを再起動するか、メディエータを再起動します。

その他のファイアウォールのトラブルシューティングについては、416 ページ「ファイアウォールのトラブルシューティング」を参照してください。

第 17 章

ファイアウォール越しの実行

LoadRunner では、コントローラがファイアウォールの外側にある場合に、ファイアウォール越しに仮想ユーザを実行できます。

本章では、次の項目について説明します。

- ▶ ファイアウォール越しの仮想ユーザの実行について
- ▶ ファイアウォール越しの仮想ユーザ実行の概要
- ▶ ファイアウォールの内側での LoadRunner エージェントのインストール
- ▶ ファイアウォールの内側での LoadRunner エージェントの設定
- ▶ エージェントによるアクセスを許可するためのファイアウォールの設定
- ▶ ファイアウォールの外側の MI リスナーのインストールと設定
- ▶ ファイアウォール越しの仮想ユーザの実行または監視のためのコントローラの設定

ファイアウォール越しの仮想ユーザの実行について

LoadRunner でファイアウォール越しに仮想ユーザを実行するためには、まず 264 ページ「ファイアウォール越しの仮想ユーザ実行の概要」に示すすべての手順を行う必要があります。ファイアウォールの外側からサーバの監視が行えるようにするには、第 18 章「ファイアウォール越しの監視」に示す手順も完了しておく必要があります。

インストール、ファイアウォールを使用する際の設定および接続の概要については、255 ページ「LoadRunner でのファイアウォールの使用」を参照してください。

ファイアウォール越しの仮想ユーザ実行の概要

- 1 ファイアウォールの内側で仮想ユーザを実行するマシンに **LoadRunner** エージェントがインストールされていることを確認します。
264 ページ「ファイアウォールの内側での LoadRunner エージェントのインストール」を参照してください。
- 2 ファイアウォール越しに動作するように **LoadRunner** エージェントを設定します。
265 ページ「ファイアウォールの内側での LoadRunner エージェントの設定」を参照してください。
- 3 ファイアウォールを設定して、ファイアウォールの内側のエージェントとファイアウォールの外側のマシンの間で通信できるようにします。
273 ページ「エージェントによるアクセスを許可するためのファイアウォールの設定」を参照してください。
- 4 **MI** リスナーをファイアウォールの外側のマシンにインストールします。
詳細については、『**Mercury LoadRunner インストール・ガイド**』を参照してください。
- 5 各 **MI** リスナー・マシンのセキュリティ属性を設定します。
274 ページ「ファイアウォールの外側の MI リスナーのインストールと設定」を参照してください。
- 6 エージェントと **MI** リスナー・マシンを認識するようにコントローラ・マシンを設定します。
276 ページ「ファイアウォール越しの仮想ユーザの実行または監視のためのコントローラの設定」を参照してください。

ファイアウォールの内側での LoadRunner エージェントのインストール

ファイアウォール越しに仮想ユーザを実行するには、LoadRunner エージェントが仮想ユーザを実行するロード・ジェネレータ・マシンにインストールされていなければなりません。エージェントは Windows サービス、またはスタートアップ・フォルダから実行する実行可能ファイルとして追加されます。

セットアップから **ロード・ジェネレータ** をインストールした場合は、ファイアウォールの外側のロード・ジェネレータ・マシンに LoadRunner エージェントがすでにインストールされている可能性があります。

インストールされているかどうか確認するには、**[スタート]** > **[プログラム]** > **[Mercury LoadRunner]** > **[LoadRunner Agent Service/Process]** を選択します。

「**Agent Service**」または「**Agent Process**」が LoadRunner オプションの一覧に表示された場合は、エージェントがすでにインストールされています。

エージェントがインストールされていない場合は、LoadRunner のインストール CD から、ファイアウォールの内側にある仮想ユーザを実行するマシンにロード・ジェネレータ・コンポーネントをインストールします。ロード・ジェネレータ・コンポーネントをインストールする場所の詳細については、258 ページ「システムの設定」の図を参照してください。

ファイアウォールの内側での LoadRunner エージェントの設定

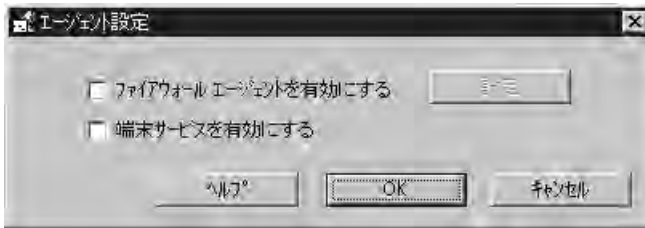
ファイアウォールの内側にあるマシンには、仮想ユーザを実行するロード・ジェネレータ・マシンと、コントローラから監視する対象となるサーバに接続されているメディアータ・マシンがあります。ファイアウォールの内側にある LoadRunner エージェントは、ファイアウォール越しに動作するように設定できます。コントローラ・マシンは、ファイアウォールの外側にあります。

Windows での LoadRunner エージェントの設定と実行

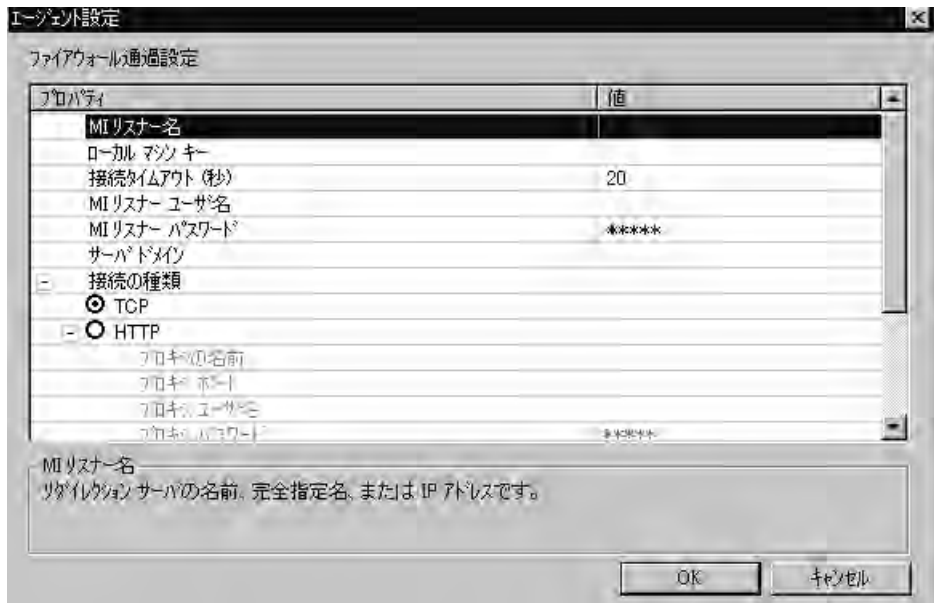
LoadRunner エージェントを Windows マシンで設定するには、次の手順で行います。

- 1 システム・トレイでアイコンを右クリックして **[Close]** を選んで、LoadRunner エージェントを停止します。
- 2 **[スタート]** > **[プログラム]** > **[Mercury LoadRunner]** > **[Advanced Settings]** から **[Agent Configuration]** を実行するか、**< LoadRunner のインストール先フォルダ > %launch_service%bin%AgentConfig.exe** を実行します。

- 3 [ファイアウォール エージェントを有効にする] チェック・ボックスを選択してから、[設定] をクリックします。



[エージェント設定] ダイアログ・ボックスが開きます。



- 4 各オプションを 271 ページ「エージェント設定」の説明どおりに設定します。
- 5 変更を保存する場合は [OK] を、変更を取り消す場合は [キャンセル] をクリックします。
- 6 デスクトップのショートカットをダブルクリックするか、[スタート] > [プログラム] > [Mercury LoadRunner] > [LoadRunner Agent Service/Process] を選択して LoadRunner エージェントを再起動します。

- 7 LoadRunner エージェントと MI リスナーとの間の接続のステータスを確認します。



システム・トレイにある LoadRunner エージェントのアイコンの隣にある信号が緑なら、LoadRunner エージェントと MI リスナーとの接続は成功しています。この信号が赤の場合、LoadRunner エージェントと MI リスナーの間に接続はありません。

UNIX LoadRunner エージェントの設定と実行

LoadRunner エージェントを UNIX マシンで設定するには、次の手順で行います。

- 1 テキスト・エディタで **< LoadRunner のインストール先フォルダ > /dat/br_inch_server.cfg** を開きます。
- 2 [Firewall] セクションで、FireWallServiceActive を 1 に設定し、変更を保存します。
- 3 **< LoadRunner のインストール先フォルダ > /bin** ディレクトリから **agent_config** を実行し、次のメニューを表示します。

```
Menu:
1. Show current settings.
2. Change a setting.
3. Save changes and exit.
4. Exit without saving.
5. Use default values.
```

- 4 「1」を入力して、現在の設定を表示します。

```
Settings:
-----
1. MI Listener Name =
2. Local Machine Key =
3. Connection Timeout (seconds) = 20
4. Connection Type = TCP
5. Use Secure Connection (SSL) = False
6. Check Server Certificates = False
7. Client Certificate Owner = False
8. Private Key User Name =
9. Private Key Password =
10. Proxy Name =
11. Proxy Port =
12. Proxy User Name =
13. Proxy Password =
14. Proxy Domain =

Menu:
1. Show current settings.
2. Change a setting.
3. Save changes and exit.
4. Exit without saving.
5. Use default values.
```

- 5 設定を変更するには、「2」を入力して設定メニューを表示します。

```
Settings:
-----
1. MI Listener Name =
2. Local Machine Key =
3. Connection Timeout (seconds) = 20
4. Connection Type = TCP
5. Use Secure Connection (SSL) = False
6. Check Server Certificates = False
7. Client Certificate Owner = False
8. Private Key User Name =
9. Private Key Password =
10. Proxy Name =
11. Proxy Port =
12. Proxy User Name =
13. Proxy Password =
14. Proxy Domain =

Enter number of setting to change or 0 to go back to menu.
```

設定を入力し、メニューの指示に従って先に進みます。各オプションを 271 ページ「エージェント設定」に従って設定します。

UNIX でのエージェント設定の変更例

「MI Listener Name」(MI リスナー名) を変更するには、次の手順で行います。

- 1 設定メニューで「1」を入力し、次の画面を表示します。

```
MI Listener Name - The name, full name or IP address of the redirection server
Old value = ranch
Enter new MI Listener Name.
```

1 行目は設定の説明です。2 行目は、設定の現在値を示します。

- 2 新しい値 (例: 「bunji」) を入力すると、次のように表示されます。

```
MI Listener Name - The name, full name or IP address of the redirection server
Old value =
Enter new MI Listener Name.
bunji
Change MI Listener Name from "" to "bunji"? 1.OK 2.CANCEL 3.FIX
```

- 3 新しい値を保存してメニューに戻るには、「1」を入力します。
新しい値を破棄してメニューに戻るには、「2」を入力します。
新しい値を廃棄して設定をもう一度変更するには、「3」を入力します。

「Connection Type」(接続の種類) を変更するには、次の手順で行います。

- 1 設定メニューで「4」を入力し、次の画面を表示します。

```
xterm
Connection Type - The connection type: TCP or HTTP.
Old value = TCP
Enter number for new Connection Type: 1.TCP 2.HTTP 3.CANCEL
```

1 行目は設定の説明です。2 行目は、設定の現在値を示します。

- 2 接続の種類を「TCP」に設定するには「1」を入力します。「HTTP」に設定するには「2」を入力します。次のように表示されます。

```
· xterm
Connection Type - The connection type: TCP or HTTP.
Old value = TCP
Enter number for new Connection Type: 1.TCP 2.HTTP 3.CANCEL
2
Change Connection Type from "TCP" to "HTTP"? 1.OK 2.CANCEL
```

- 3 新しい値を保存してメニューに戻るには、「1」を入力します。
新しい値を破棄してメニューに戻るには、「2」を入力します。

設定の表示とエージェントの再起動

現在の設定を表示するには、次の手順で行います。

- 1 「1」を入力してメイン・メニューに戻ります。
- 2 「1」を入力すると現在の設定が表示されます。次の例では、「MI Listener Name」と「Connection Type」の新しい設定が反映されています。

```
· xterm

Settings:
-----
1. MI Listener Name = bunji
2. Local Machine Key = gumbi
3. Connection Timeout (seconds) = 20
4. Connection Type = HTTP
5. Use Secure Connection (SSL) = False
6. Check Server Certificates = False
7. Client Certificate Owner = False
8. Private Key User Name =
9. Private Key Password =
10. Proxy Name =
11. Proxy Port =
12. Proxy User Name =
13. Proxy Password =
14. Proxy Domain =

Menu:
1. Show current settings.
2. Change a setting.
3. Save changes and exit.
4. Exit without saving.
5. Use default values.
```

3 変更を保存するには、メイン・メニューで「3」を入力します。

変更を取り消すには「4」を入力します。

LoadRunner によって提示される標準設定の値（271 ページ「エージェント設定」で参照）を使用するには、「5」を入力します。

LoadRunner エージェントを開始または削除するには、次の手順で行います。

- 1 LoadRunner エージェントを開始するには、**< LoadRunner のインストール先フォルダ > /bin** ディレクトリからコマンド「**m_daemon_setup -install**」を実行します。
- 2 LoadRunner エージェントを削除するには、**< LoadRunner のインストール先フォルダ > /bin** ディレクトリからコマンド「**m_daemon_setup -remove**」を実行します。

注： LoadRunner エージェントをファイアウォール越しに実行するように設定し、そのエージェントを MI リスナーに接続すると、LoadRunner エージェント・マシンの一時ディレクトリの中に、**< ローカル・マシン・キー > _connected_to_MI_Listener** ファイルが作成されます。このファイルは、LoadRunner エージェントが MI リスナーから切断されると削除されます。

LoadRunner エージェントの実行の詳細については、406 ページ「UNIX シェル」を参照してください。

エージェント設定

オプション	標準設定値	説明
MI リスナー名	なし	Mercury のリスナー用のマシンである MI リスナーのフルネームまたは IP アドレス。
ローカル マシン キー	なし	MI リスナー・マシンを経由でコントローラ・ホストとエージェント・マシンの間で一意の接続を確立するために使用される文字列識別子。

オプション	標準設定値	説明
接続タイムアウト (秒)	20 秒	MI リスナー・マシンとの接続を再試行するまでに、エージェントが待機する時間の長さ。ゼロの場合、エージェントは実行が開始されたときから接続されたままになります。
MI リスナー ユーザ名	なし	MI リスナー・マシンに接続するために必要なユーザ名。
MI リスナー パスワード	なし	MI リスナー・マシンに接続するために必要なパスワード。
サーバドメイン	なし	MI リスナー・マシンに接続するために必要なドメイン名。このフィールドは、NTLM が使用されている場合のみ必要です。
接続の種類 - TCP/HTTP	TCP	使用している設定に応じて、 TCP か HTTP のどちらかを選択します。
接続の種類 - HTTP プロキシの名前	なし	プロキシ・サーバの名前。このオプションは、[接続の種類] オプションが「 HTTP 」に設定されている場合には必須です。
接続の種類 - HTTP プロキシ・ポート	なし	プロキシ・サーバの接続ポート。このオプションは、[接続の種類] オプションが「 HTTP 」に設定されている場合には必須です。
接続の種類 - HTTP プロキシ・ユーザ名	なし	プロキシ・サーバへの接続権限を持つユーザのユーザ名。
接続の種類 e - HTTP プロキシ・パスワード	なし	プロキシ・サーバへの接続権限を持つユーザのパスワード。
接続の種類 - HTTP プロキシ・ドメイン	なし	ユーザのドメイン (プロキシ・サーバの設定で定義されている場合)。このオプションは、NTLM が使用されている場合のみ必要です。
セキュア接続を使用 (SSL)	無効	Secure Sockets Layer プロトコルを使用して接続できます。

オプション	標準設定値	説明
セキュア接続を使用 (SSL) -サーバ証明書を確認	なし	サーバから送信されてくる SSL 証明書を認証します。サーバの証明書が信頼できる認証機関によって署名されているかどうか確認する場合は、「中」を選択します。送信者の IP が証明情報と一致するかどうかチェックする場合は、「高」を選択します。この設定を使用できるのは、「 セキュア接続を使用する (SSL) 」オプションが「True」に設定されている場合だけです。
セキュア接続を使用 (SSL) - 秘密キー・パスワード	なし	SSL 証明書認証処理に必要な場合があるパスワード。このオプションが意味を持つのは、「 クライアント証明書を使用 」オプションが有効になっている場合だけです。
セキュア接続を使用 (SSL) - クライアント証明書を使用	無効	SSL 証明書の読み込みを有効にします (接続の許可を得るために、サーバによって証明書が要求される場合)。このオプションが意味を持つのは、「 セキュア接続を使用 」オプションが有効になっている場合だけです。

エージェントによるアクセスを許可するためのファイアウォールの設定

ファイアウォールの設定を変更して、ファイアウォールの内側のマシンとファイアウォールの外側のマシンが相互に通信できるようにします。

TCP 設定

The LoadRunner エージェントは、ポート 443 を通じて MI リスナーとの接続を確立しようと試みます。この接続を可能にするには、ファイアウォールの 443 番ポート通じて HTTPS サービスへの送信ができるように設定します。これにより、エージェントは、エージェント設定の「接続タイムアウト」フィールド

で指定された間隔（単位は秒）で、MI リスナーへの接続を試み続けるようになります。この接続が成功すると、MI リスナーからエージェントへの接続も行われます。以降、エージェントはMI リスナーからのコマンドをリッスンします。

HTTPS 設定

LoadRunner エージェントは、「プロキシポート」フィールドで指定されたプロキシ・ポートを通じてMI リスナーとの接続を確立しようと試みます。この接続を可能にするには、ファイアウォールの443番ポートを通じてHTTPS サービスへの送信ができるように設定します。エージェントは、エージェント設定の「接続タイムアウト」フィールドで指定された間隔（単位は秒）で、MI リスナーへの接続を試み続けるようになります。接続が成功すると、プロキシ・サーバ上のエージェントがMI リスナーに接続し、MI リスナーもプロキシ・サーバを通じてエージェントに接続します。以降、エージェントはMI リスナーからのコマンドをリッスンします。

注：MI リスナーと LoadRunner エージェントを使用せずに、コントローラをファイアウォール越しにロード・ジェネレータに接続できます。これには、コントローラのLANとロード・ジェネレータのLANで、ファイアウォールのポート54345を開き、データの送受信が行えるようにします。ただし、RSHを使用してコントローラからUNIXロード・ジェネレータ・マシン上のエージェントを起動する場合は、エージェントは動的ポートともにロードされます。この場合は、ロードされた動的ポートを確認し、UNIXロード・ジェネレータ・マシンでそのポートを開きます。

ファイアウォールの外側のMI リスナーのインストールと設定

ファイアウォール越しの仮想ユーザの実行または監視を行うには、ファイアウォールの外側のコントローラ・マシンと同じLAN内の1台または複数台のマシンにMI リスナーをインストールする必要があります。インストールの詳細については、『**Mercury LoadRunner インストール・ガイド**』を参照してください。

注：

- ▶ コントローラをインストールすると MI リスナーが自動的にインストールされるため、コントローラを MI リスナー・マシンとして指定することができます。
- ▶ MI リスナーは、Windows マシンにのみインストールできます。

MI リスナーのセキュリティ属性を設定するには、次の手順で行います。

- 1 ポート 443 を HTTPS サービスの受信ポートとして開きます。ポートの設定は、システム管理者によって指定されています。
- 2 システム・トレイでアイコンを右クリックして **[Close]** を選んで、LoadRunner エージェントを停止します。
- 3 **[スタート]** > **[プログラム]** > **[Mercury LoadRunner]** > **[Advanced Settings]** から **[MI Listener Configuration]** を実行するか、**< LoadRunner のインストール先フォルダ > \¥launch_service¥bin¥MILsnConfig.exe** を実行します。



- 4 各オプションを 276 ページ「MI リスナーの設定」の説明どおりに設定します。
- 5 変更を保存する場合は **[OK]** を、変更を取り消す場合は **[キャンセル]** を、標準設定を使用する場合は **[標準設定を使用]** をクリックします。

- 6 デスクトップのショートカットをダブルクリックするか、[スタート] > [プログラム] > [Mercury LoadRunner] から選択して、LoadRunner エージェントを再起動します。
- 7 ポート 443 が MI リスナー・マシンで空いていることを確認します。

注：MI リスナーまたは「ファイアウォール越しの監視」マシンで動作している Web サーバがないことを確認します。Web サーバもポート 443 を使用するため、リッスンおよび監視を行うプロセスにとって必要なアクセスを防げます。

MI リスナーの設定

オプション	標準設定値	説明
クライアント認証のチェック	False	クライアントの接続時に SSL 証明書を送信するよう要求する場合、および証明書を認証する場合は、「True」を選択します。
秘密鍵のパスワード	なし	SSL 証明書認証処理に必要な場合があるパスワード。

ファイアウォール越しの仮想ユーザの実行または監視のためのコントローラの設定

ファイアウォールの内側で仮想ユーザの実行やサーバの監視を行うには、コントローラとエージェント・マシンの間に一意の接続を確立する必要があります。この接続は、コントローラと LoadRunner エージェント間でルータの役割を果たす MI リスナーを通じて確立します。この接続を確立するには、コントローラ・マシンを設定して、エージェント・マシンをロード・ジェネレータとして定義する必要があります。

ファイアウォール越しの仮想ユーザの実行と監視のためにコントローラを設定するには、次の手順で行います。

- 1 [スタート] > [プログラム] > [Mercury LoadRunner] からコントローラを実行し、新しいシナリオを作成するか、既存のシナリオをロードします。

- ロード・ジェネレータを選択し、**[詳細]** をクリックしてロード・ジェネレータの情報を表示します。



- [ファイアウォール] タブの [MI リスナー] フィールドに、MI リスナー・マシンの名前を入力します。この名前は、[エージェントの設定] ダイアログ・ボックスの [MI リスナー名] 設定で入力した名前と同じです。次の例では、MI リスナーは「bunji」です。
- [ファイアウォールの設定] セクションで、次のオプションのうち1つを選択します。
 - ▶ **[ファイアウォール越しで仮想ユーザを実行する]**: 仮想ユーザをファイアウォール越しに実行する場合。
 - ▶ **[ファイアウォール越しでの監視を有効にする]**: 仮想ユーザをファイアウォール越しに監視する場合。

注：WAN エミュレータを使用する場合は、MI リスナー・マシンの IP アドレスを [WAN エミュレーションから除外するホスト] リストに追加する必要があります。詳細については、101 ページ「WAN エミュレーションからのホストの除外」を参照してください。

- 6 **[OK]** をクリックして、[ロード ジェネレータ] ダイアログ・ボックスに戻ります。
 - 7 ロード・ジェネレータを選択し、**[接続]** をクリックします。
-

注：ファイアウォール越しに仮想ユーザの実行または監視を行っているホストの一時ディレクトリは変更できないので注意してください。

接続上の問題が生じた場合は、260 ページ「接続の確認」を参照してください。その他のファイアウォール関連のトラブルシューティングは、416 ページ「ファイアウォールのトラブルシューティング」を参照してください。

第4部・ファイアウォールを使った作業

第 18 章

ファイアウォール越しの監視

コントローラがファイアウォールの外側にある場合に、ファイアウォールの内側からサーバを監視できます。

本章では、次の項目について説明します。

- ▶ ファイアウォール越しの監視について
- ▶ ファイアウォール越しの監視機能のインストール
- ▶ データ収集の準備
- ▶ サーバ・モニタ・プロパティの設定
- ▶ ファイアウォール越しのネットワーク遅延モニタの設定

ファイアウォール越しの監視について

重要：ファイアウォールの外側からサーバを監視できるようにするには、まず 264 ページ「ファイアウォール越しの仮想ユーザ実行の概要」に示すすべてのステップを完了します。

ファイアウォール越しの監視を有効にするには、次の手順を実行します。

- ▶ 「ファイアウォール越しの監視」コンポーネントをインストールします。
- ▶ データ収集の準備をします。
- ▶ サーバ・モニタのプロパティを設定します。

ファイアウォール越しの監視コンポーネントは、ファイアウォールの内側の指定のマシンにインストールする必要があります。インストールを行うと、サーバ・モニタ設定ツールのほか、サーバ・モニタ・メディエータ（メディエータと呼びます）が設定されます。その後、監視対象サーバを設定し、LoadRunnerメディエータが各監視対象サーバで収集する測定値を定義します。

ファイアウォール越しの監視機能のインストール

次のいずれかの方法で、「ファイアウォール越しの監視」をメディエータにインストールします。

- ▶ LoadRunnerのCDからLoadRunnerのカスタム・インストールを実行し、[ファイアウォール越しの監視] オプションだけを選択します。

LoadRunnerのカスタム・インストールの方法については、『**Mercury LoadRunner インストール・ガイド**』を参照してください。

- ▶ 「**Monitors over Firewall**」ファイルは、Mercuryのカスタマー・サポート Web サイト（US サイト）（<http://support.mercury.com>）から入手できます。ファイアウォール越しの監視は、ダウンロード可能で、スタンドアロンでのインストールが可能です。このコンポーネントは、自己解凍式のインストーラ・ファイルの形で提供されています。

[ファイアウォール越しの監視] コンポーネントのインストール場所の詳細については、255 ページ「LoadRunnerでのファイアウォールの使用」の図を参照してください。

データ収集の準備

ファイアウォール越しの監視コンポーネントをインストールしたら、データ収集の準備を行う必要があります。先に進む前に、263 ページ「ファイアウォール越しの実行」に示すすべてのステップを完了したことを確かめてください。

データ収集用にコントローラを設定するには、次の手順で行います。

- 1 ファイアウォール越しの処理を行うには、276 ページ「ファイアウォール越しの仮想ユーザの実行または監視のためのコントローラの設定」で説明するように LoadRunner エージェントとコントローラが設定されている必要があります。

[ロード ジェネレータの情報] ダイアログ・ボックスの [ファイアウォール] タブで、[**ファイアウォール越しでの監視を有効にする**] をチェックして MI リスナー・マシンの IP アドレスを入力します。

- 2 モニタ・エージェントに接続します。[シナリオ] > [ロード ジェネレータ] を選択して、[ロード ジェネレータ] 画面を表示します。[名前] フィールドで、サーバのシンボル名を入力します。この名前は、エージェント設定の [ローカル マシン キー] 設定で入力した名前と同じ名前です (271 ページ「エージェント設定」を参照してください)。

ファイアウォールの内側に設定されているモニタからすべての情報を取得できることを確認します。

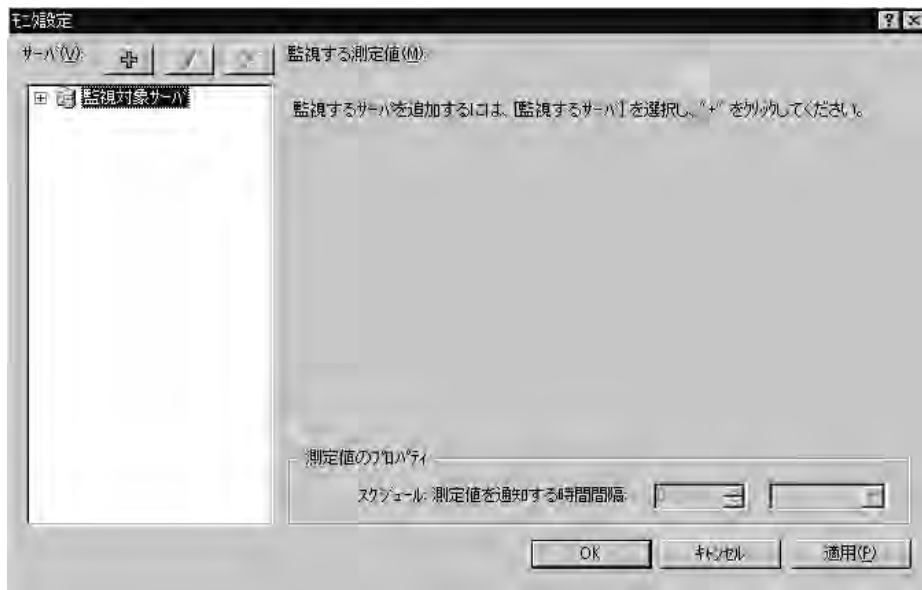
サーバ・モニタ・プロパティの設定

LoadRunner エージェント、ファイアウォール越しでの監視コンポーネント、MI リスナー、コントローラ・マシンのインストールと設定が終了したら、メディエータに監視させるサーバ測定値を選択します。

メディエータから、[モニタ設定] ダイアログ・ボックスを使用して、サーバ・モニタのプロファイルを設定します。実行するモニタのタイプと監視対象リソースが含まれるサーバを選択し、サーバごとに監視対象の測定値を追加したら、監視対象の測定値の報告を受け付ける頻度を指定します。

サーバ・モニタのプロパティを設定するには、次の手順で行います。

- 1 [スタート] > [プログラム] > [Mercury LoadRunner] > [Advanced Settings] > [Monitor Configuration] を選択します。LoadRunner が完全にインストールされていないマシンの場合は、[スタート] > [プログラム] > [Server Monitor] > [Monitor Configuration] を選択します。[モニタ設定] ダイアログ・ボックスが開きます。



- +** 2 [サーバの追加] ボタンをクリックします。[新規監視対象サーバプロパティ]ダイアログ・ボックスが開きます。



- 3 [監視対象サーバ] ボックスに、監視対象のリソースが含まれるサーバの名前または IP アドレスを入力します。

注：同時に複数のサーバを追加するには、サーバ名または IP アドレスをカンマで区切ります。次に例を示します。

255.255.255.0-255.255.255.5, server1, server2

- 4 [利用可能なモニタ] リストから、監視対象のサーバに適したモニタを選択します。

注：LoadRunner ライセンス・キーで利用できるモニタのデータだけを表示できます。ライセンス・キー情報を確認するには、[スタート] > [プログラム] > [Mercury LoadRunner] をクリックします。Mercury LoadRunner が起動します。[LoadRunner ライセンス] ボタンをクリックして、LoadRunner のライセンス情報を表示します。

- 5 [OK] をクリックして、[新規監視対象サーバプロパティ] ダイアログ・ボックスを閉じます。[監視対象サーバ] リストが [モニタ設定] ダイアログ・ボックスに表示されます。



いくつかのモニタでは、LoadRunner によって [監視する測定値] セクションに標準設定の測定値が表示されます。[測定値のプロパティ] セクションに LoadRunner からの測定値の報告を受け付ける頻度を指定できます。測定値の選択の詳細については、286 ページ「測定値の追加と削除」を参照してください。

- 6 監視対象サーバをリストにさらに追加するには、1～5 の手順を繰り返します。




- 7 サーバのモニタ設定のプロパティを編集するには、[編集] ボタンをクリックします。[監視対象サーバのプロパティ] ダイアログ・ボックスが開き、リソースを監視中のサーバのモニタを編集できます。

- 8 [適用] をクリックして、設定を保存します。

測定値の追加と削除


監視対象のサーバ・マシン（1台または複数台）を設定したら、各サーバで監視する測定値を追加します。LoadRunnerによって追加された標準設定の測定値は、必要に応じて編集できます。

監視対象の測定値を追加するには、次の手順で行います。

- 1 [監視対象サーバ] リストからサーバを選択します。
- 2  [測定値の追加] ボタンをクリックします。対象モニタを選択します。選択したモニタの測定値を選択するためのダイアログ・ボックスが開きます。
- 3 監視する測定値を選択して、[OK] をクリックします。
- 4 [適用] をクリックして、設定を保存します。

各サーバ・モニタの測定値の設定については、『Mercury LoadRunner モニタ・リファレンス』の対応する章を参照してください。

測定値のリストから測定値を削除するには、次の手順で行います。

- 1  測定値を選択して、[削除] ボタンをクリックします。
- 2 [適用] をクリックして、設定を保存します。

測定頻度の設定

モニタの測定値を設定したら、測定頻度を設定します。

[測定値のプロパティ] セクションで、各測定値を通知する測定スケジュールを設定します。



測定値の測定スケジュールを設定するには、次の手順で行います。

- 1 スケジュールを設定する対象となるサーバの測定値を選択します。
- 2 LoadRunner に測定値を通知させる頻度を指定します。
- 3 [適用] をクリックして、設定を保存します。

監視対象サーバ・プロパティのクローニング

異なるサーバ・マシンで同じプロパティを監視する場合は、[監視対象サーバプロパティの複製] ダイアログ・ボックスを使用して、選択したサーバのクローンを作成します。

監視対象サーバのプロパティのクローンの作成は、次の手順で行います。

- 1 [モニタ設定] ダイアログ・ボックスで、クローンを作成したいサーバを右クリックし、[複製] を選択します。[監視対象サーバプロパティの複製] ダイアログ・ボックスが開きます。



- 2 [監視対象サーバ] ボックスに、作成するクローン・サーバの名前または IP アドレスを入力します。

注：同時に複数のサーバを作成するには、サーバ名または IP アドレスをカンマで区切ります。次に例を示します。

255.255.255.0-255.255.255.5, server1, server2

- 3 [利用可能なモニタ] リストに、クローニングされるサーバ用に選択されたモニタが表示されます。クローン・サーバに適したモニタを追加選択します。
- 4 [OK] をクリックして、[監視対象サーバプロパティの複製] ダイアログ・ボックスを閉じます。クローニングされたサーバが [監視対象サーバ] リストに表示されます。
- 5 [適用] をクリックして、設定を保存します。

ファイアウォール越しのネットワーク遅延モニタの設定

コントローラ・マシンとソース・マシンの間にファイアウォールがあるときにネットワーク遅延モニタを実行するには、ネットワーク遅延モニタを設定し (『Mercury LoadRunner モニタ・リファレンス』の「ネットワーク遅延時間モニタの設定」を参照)、次を手順3に追加します。

次の形式に従って、[次のマシンからネットワーク遅延を監視する] セクションに監視元マシンのサーバ名または IP アドレスを入力します。

< MI リスナー・マシン > : < 送信元マシンのローカル・キー >

< 送信元マシンのローカル・キー > は、送信元マシンで LoadRunner エージェントを設定するときに選択するローカル・マシン・キーです。詳細については、271 ページ「エージェント設定」を参照してください。次に例を示します。

12.12.12.3:vds

第 5 部

診断を使った作業

本部では、ERP/CRM 診断モジュールを使った作業について説明します。

Diagnostics for J2EE & .NET を使った作業については、『**Mercury Diagnostics for J2EE & .NET Installation and User's Guide**』（英語版）を参照してください。

第 19 章

LoadRunner 診断モジュール

LoadRunner の診断モジュールは、Siebel, Oracle, SAP, J2EE および .NET 環境においてパフォーマンスの問題をすばやく特定するのに役立つ詳細なパフォーマンス情報を提供します。

診断モジュールには、ユーザ定義のアナリシス・レポート一式が付属しています。このレポートを使えば、気付いたことを開発者や上司に伝えることができ、高価なハードウェアのアップグレードをせずに、問題を早期に解決できます。

本章では、次の項目について説明します。

- ▶ LoadRunner 診断モジュールについて
- ▶ LoadRunner ERP/CRM 診断の種類
- ▶ ERP/CRM 診断モジュールのアーキテクチャ
- ▶ LoadRunner ERP/CRM 診断を使った作業
- ▶ リモート・サーバへの接続
- ▶ 診断の有効化
- ▶ 診断結果の表示

LoadRunner 診断モジュールについて

LoadRunner の診断モジュールはパフォーマンス・テスト中に、Web サーバ、アプリケーション・サーバ、およびデータベース・サーバを通じて、個々のトランザクションのトレース、時間測定、トラブルシューティングを行います。遅いエンド・ユーザ・トランザクションから、ボトルネックになっているメソッドまたは SQL ステートメントまでドリル・ダウンできます。LoadRunner の診断モジュールを使用すれば、次のことが可能になります。

- ▶ ビジネス・プロセスによって実行されるアプリケーション・コンポーネントをトレースする。

第5部・診断を使った作業

- ▶ エンド・ユーザ体験に大きく影響しているアプリケーション・コンポーネントをすばやく特定する。
- ▶ パフォーマンスを向上させる方法に関する正確なデータを開発者に提供する。

このような正確なパフォーマンス問題の特定は、次のような大きなビジネス価値を直接生み出します。

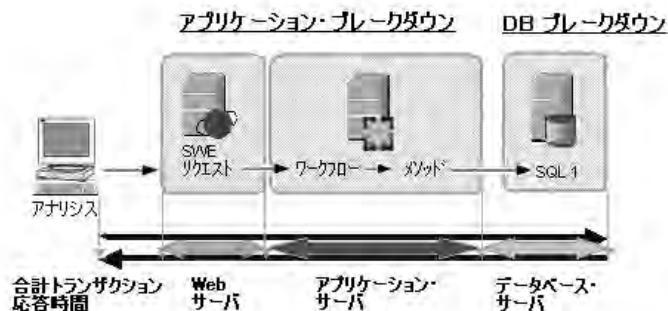
- ▶ より迅速で効率的なパフォーマンス・テスト・サイクル
- ▶ 開発部門による問題解決にかかる時間の短縮
- ▶ ビジネスのニーズに合わせて最適化され、パフォーマンスが向上したアプリケーション

LoadRunner ERP/CRM 診断の種類

LoadRunner は、次の ERP/CRM 診断ソリューションを提供しています。

▶ Siebel 診断

次の図は、Siebel CRM システムでのトランザクション・ブレイクダウンを示しています。



LoadRunner の Siebel 診断は、次の 2 つのモジュールに分かれています。

- ▶ **Siebel 診断モジュール**では、Siebel トランザクションをレイヤ、エリア、サブエリア、サーバ、およびスクリプトにブレイクダウンできます。また、トランザクションの呼び出しチェーンと呼び出しスタックの統計情報を表示して、トランザクションの各部分で費やされる時間の割合を知ることができま

す。Siebel Web 仮想ユーザは、Siebel 診断をサポートします。詳細については、第 20 章「Siebel 診断モジュールの設定」を参照してください。

- ▶ **Siebel DB 診断モジュール**は、データベースのパフォーマンスの問題をすばやく突き止めて解決するのに役立ちます。各トランザクションの SQL を表示し、各スクリプトの問題のある SQL クエリを特定して、どの時点で問題が発生したかを突き止めることができます。Siebel Web 仮想ユーザは、Siebel DB 診断をサポートします。詳細については、第 21 章「Siebel DB 診断モジュールの設定」を参照してください。

▶ Oracle 11i 診断

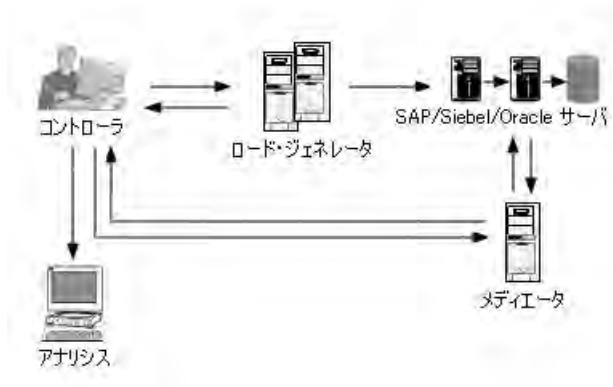
Oracle トランザクション・ブレイクダウンは、Oracle NCA システム上でのパフォーマンスの問題を特定するのに役立ちます。診断情報は、トランザクションから SQL ステートメントへ、そして各ステートメントの SQL ステージへとドリル・ダウンします。Oracle NCA 仮想ユーザは Oracle 11i 診断モジュールをサポートしています。詳細については、第 22 章「Oracle 11i 診断モジュールの設定」を参照してください。

▶ SAP 診断

SAP ブレイクダウンを使うと、問題の根本原因（DBA、ネットワーク、WAS、アプリケーション、OS/HW）をすばやく簡単に突き止めて、関連のある専門家だけで対応でき、チーム全員に問題を提示する必要がありません。詳細については、第 23 章「SAP 診断モジュールの設定」を参照してください。

ERP/CRM 診断モジュールのアーキテクチャ

ERP/CRM 診断モジュールのアーキテクチャは、次のコンポーネントで構成されています。



ERP/CRM メディエータ：ERP/CRM メディエータ（「メディエータ」）は、Web サーバ、データベース・サーバ、およびアプリケーション・サーバからオフライン・トランザクション・データを収集して相関させます。メディエータは、監視対象の ERP/CRM サーバと同じ LAN 内の、できれば専用のサーバにインストールする必要があります。負荷テストに関する Siebel サーバ、Oracle サーバ、または SAP サーバ上にメディエータをインストールすることはお勧めしません。メディエータのインストールの詳細については、『**Mercury LoadRunner インストール・ガイド**』を参照してください。

注：

標準設定では、メディエータ・エージェントはプロセスとして実行されるようにインストールされます。メディエータ・エージェントはサービスとして実行するように設定することをお勧めします。

エージェントをプロセスとして実行すると、Microsoft Windows のネットワーキング制限「Failed to establish connection. System error 1219」が表示される場合があります。次のいずれかを実行できます。

- ▶ エージェントをサービスとして実行します。詳細については、423 ページ「LoadRunner エージェントをサービスとして実行」を参照してください。
 - ▶ サーバへの以前のすべての接続を切断して、再度接続を試みます。
-

注： Siebel DB 診断の場合、ファイルをアプリケーション・サーバからメディアータに、そしてメディアータから結果ディレクトリにコピーするには、長い時間がかかる場合があります。コピーの第 1 段階では、[Summary Data Processing] ウィンドウが表示されます。コピーの第 2 段階にかかる時間を最小限に抑えるために（ファイアウォール越しの監視ではない場合）、メディアータ・マシンとして **localhost** を使用することをお勧めします。Siebel DB 診断ファイルがすでにコントローラ・マシン上に存在するため、これにより診断ファイルを結果ディレクトリにコピーするためにかかる時間が短縮されます。

コントローラ： シナリオ実行の前に、コントローラはすべてのサーバ情報をメディアータに転送し、その監視に関わるユーザの割合を分配します。シナリオの実行後、コントローラは集積されたトランザクション・データ・ファイルを各メディアータから収集して、結果を照合します。Siebel の結果は **¥sbl_bd** ディレクトリに転送され、Oracle 11i 診断は **¥ora_bd** ディレクトリに転送されます。SAP の結果は **¥sap_bd** ディレクトリに転送されます。診断ファイル照合のステータスを、[結果の照合] ダイアログ・ボックスに表示することができます。詳細については、186 ページ「結果の照合」を参照してください。

ロード・ジェネレータ： シナリオの実行時には、コントローラが各仮想ユーザをロード・ジェネレータに分配し、ロード・ジェネレータが仮想ユーザ・スクリプトを実行します。

アナリシス： 詳細な診断グラフと診断レポートを表示します。診断グラフの詳細については、『Mercury LoadRunner アナリシス・ユーザーズ・ガイド』を参照してください。

LoadRunner ERP/CRM 診断を使った作業

LoadRunner の ERP/CRM 診断を使用するには、次の手順を実行します。

1 診断データ生成の準備をします。

ERP/CRM メディアータがインストールされていることを確認します。メディアータは、診断データを収集して処理します。

ERP/CRM メディアータは、LoadRunner フル・セットアップの一部としてコントローラ・マシンにインストールされます。ERP/CRM メディアータを専用マ

シンにインストールする方法については、『**Mercury LoadRunner インストール・ガイド**』を参照してください。

2 サーバ・マシンを設定して診断機能を有効にします。

詳細については、305 ページ「アプリケーション・サーバおよび Web サーバ上での Siebel 診断モジュールの設定」、316 ページ「Siebel サーバでのサーバ・ログの有効化」、323 ページ「Oracle サーバでのサーバ・ログの有効化」、および 325 ページ「Oracle NCA アプリケーションのバージョンの選択」を参照してください。

3 診断データの生成とメディエータ・マシンとの通信のために、コントローラ・マシンの準備を整えます。

詳細については、以下を参照してください。

- ▶ 第 20 章「Siebel 診断モジュールの設定」
- ▶ 第 21 章「Siebel DB 診断モジュールの設定」
- ▶ 第 22 章「Oracle 11i 診断モジュールの設定」
- ▶ 第 23 章「SAP 診断モジュールの設定」
- ▶ 300 ページ「診断の有効化」

4 診断データを収集して準備します。

負荷テストの実行中に、メディエータはデータを収集して診断情報を処理します。

5 結果を作成します。

負荷テストの実行後、コントローラは集積されたデータを各メディエータ・マシンから収集して、結果を照合します。結果の照合の詳細については、186 ページ「結果の照合」を参照してください。

6 データを表示します。

アナリシス・グラフを使って診断データを表示し、問題のあるエリアまでドリル・ダウンします。ERP/CRM トランザクション・ブレイクダウン・グラフの詳細については、『**Mercury LoadRunner アナリシス・ユーザーズ・ガイド**』を参照してください。

リモート・サーバへの接続

- ▶ リモート Windows サーバへの接続
- ▶ リモート UNIX サーバへの接続

リモート Windows サーバへの接続

Windows プラットフォームの場合、このユーザは管理者権限を持っている必要があります。

リモートの WINDOWS サーバ上で動作する場合、メディエータはまず、ユーザが入力した詳細情報を使って、そのマシンへの接続を追加しようと試みます。この設定では、リモート・マシンに管理者権限が付与されます。

メディエータ・マシンがすでに異なる設定でサーバ・マシンに接続されている場合は、その既存の接続が使用されます。そのため、ユーザが管理者でない場合にはエラーが発生することがあります（トラブルシューティングの章のエラー 1219 の説明を参照してください）。

この問題を解決するには、次のいずれかの方法を実行します。

- ▶ LoadRunner エージェントがプロセスではなくサービスとして動作するように変更する。

詳細については、423 ページ「LoadRunner エージェントをサービスとして実行」を参照してください。

- ▶ メディエータからサーバへの既存の接続を閉じる。

これらの接続は、Windows エクスプローラ内の **¥¥servername¥sharename**、または [マイ コンピュータ] 内の指定されたマウントを使用していることがあります。

存在する接続をチェックするには、コマンド・プロンプトを開いて、コマンド・ラインから次のコマンドを実行します。

```
net use
```

接続を削除するには、マウントを削除して、コマンド・ラインから次のコマンドを実行します。

```
net use ¥¥servername¥sharename /DELETE
```

- ▶ ログ・ディレクトリへの UNC パスを指定して、OS を WINDOWS としてマークすることにより、ログ・ディレクトリへの独自に作成した接続を使用できま

す。メディアータは、実行時に接続の作成を試みることはなく、指定された UNC パスを使用します。

リモート UNIX サーバへの接続

UNIX プラットフォームで動作する場合、メディアータは次の2種類の接続をサポートします。

- ▶ リモート・シェル (RSH/RCP)
- ▶ セキュア・シェル (PLINK/PSCP)

リモート・シェル (RSH/RCP)

メディアータを通じてリモート・シェル (RSH) を使用する前に、次のことを確認してください。

- ▶ UNIX サーバ上で RSH および RCP デーモンが実行している必要があります。
- ▶ ユーザは、リモート・シェル・コマンドを実行する権限を持っている必要があります。これをチェックするには、DOS コマンド・プロンプトで次のように入力します。

```
rsh <サーバ・マシン名> -l <UNIX ユーザ・ログイン名> -n <コマンド>
```

次に例を示します。

```
rsh my_unix -l my_name -n "cd ~;pwd"
```

注：DOS コマンドのプロンプト・ウィンドウで実行する RSH コマンドだけが LoadRunner で使用できます。

「RSH コマンドの失敗」エラーが LoadRunner で発生したら、DOS コマンド・プロンプトから同じコマンドを実行して、この問題がコマンドによるものか LoadRunner によるものかを検証します。DOS コマンド・プロンプトで実行してコマンドが有効ならば、Mercury のカスタマー・サポートに問い合わせを仰いでください。

DOS コマンド・プロンプトからもこのコマンドを実行できなければ、コマンドに問題があります。RSH コマンドの詳細については、406 ページ「UNIX シェル」を参照してください。

- ▶ RSH コマンドの実行後に出力を生成してはなりません。RCP UNIX コマンドにバグがあるため、**.login**、**.profile**、および **.cshrc** ファイルから出力を生成してはなりません（例えば **echo** によって、または **biff** のように間接的に出力を生成するコマンドも含めたほかの手段によって）。既存のユーザが

RSH ステップで削除できない出力を生成する場合は、サーバ・マシン上で RSH および RCP コマンドを実行する権限を持った、出力を生成しない新規のユーザを作成する必要があります。

セキュア・シェル (PLINK/PSCP)

セキュア・シェルは SSH プロトコルに基づいています。このプロトコルは、RSA/DSA 鍵ペアを使った認証機構と暗号化された通信によって、リモート・マシンに接続するための安全な手段を提供します。

SSH プロトコルによるセキュリティには、次のようなレベルがあります。

- ▶ ユーザ名とパスワードを使用する。
- ▶ ユーザ名と鍵ペアをパスフレーズ保護なしで使用する。
- ▶ ユーザ名、鍵ペア、およびパスワード保護された秘密鍵を使用する。

メディアータは、無料で配布されている SSH の実装である PuTTY スイートを使用します。

次のような PuTTY ツールが、インストール先の bin ディレクトリにあります。

PUTTY.EXE, PAGEANT.EXE, PLINK.EXE, PSCP.EXE, PSFTP.EXE,
PUTTYGEN.EXE, PUTTY.HLP

PuTTYgen の [Conversions] メニューから Import コマンドを使用して、OpenSSH の形式と ssh.com の形式の秘密鍵をロードして、それを PuTTY 形式の鍵として保存できます。詳細については、インストール先の bin ディレクトリにある『**PuTTY User Manual**』を参照してください。

メディアータを通じてセキュア・シェルを使用する前に、次のことを確認してください。

- ▶ UNIX サーバ上で SSH デーモンが実行している必要があります。
- ▶ PuTTY コマンドの実行後に出力を生成してはなりません。**.login**、**.profile**、および **.cshrc** ファイルから出力を生成してはなりません（例えば echo によって、または biff のように間接的に出力を生成するコマンドも含めたほかの手段によって）。既存のユーザが削除できない出力を生成する場合は、出力を生成しない新規のユーザを作成する必要があります。

PuTTY コマンドがエラーなしで動作していることを確認するには、DOS コマンド・プロンプトで、インストール先の bin ディレクトリの下で次のコマンドを入力します。

PLINK.EXE <サーバ・マシン名> -ssh -l <UNIX ユーザ・ログイン名> -i <使用している場合は秘密鍵のフルネーム> -pw <パスワードまたはパスフレーズ> <コマンド>

次に例を示します。

セキュリティ・レベル A: PLINK.EXE my_unix -ssh -l my_name -pw "my_password" "ls"

セキュリティ・レベル B: PLINK.EXE my_unix -ssh -l my_name -i "my_private_key" -pw "" "ls"

セキュリティ・レベル C: PLINK.EXE my_unix -ssh -l my_name -i "my_private_key" -pw "my_passphrase" "ls"

同様に、**PSCP.EXE** コマンドをチェックします。

PSCP.EXE -scp -r -q <使用している場合は秘密鍵のフルネーム> -pw <パスワードまたはパスフレーズ> <ローカル・ファイル名> <UNIX ユーザ・ログイン名> @ <サーバ・マシン名> : <リモート・ファイル名>

「SSH コマンドの失敗」エラーが LoadRunner で発生したら、DOS コマンド・プロンプトから同じコマンドを実行して、この問題がコマンドによるものか LoadRunner によるものかを検証します。DOS コマンド・プロンプトで実行してコマンドが有効ならば、詳細については Mercury のカスタマー・サポート (<http://www.mercury.com/jp/services/support/>) にお問い合わせください。

DOS コマンド・プロンプトからもこのコマンドが実行できない場合は、Unix 管理者にお問い合わせください。

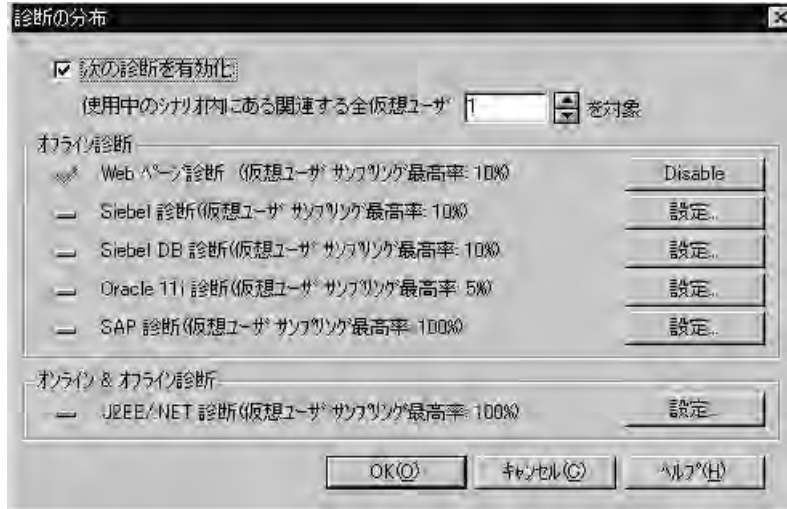
診断の有効化

診断モジュールの設定を完了し、コントローラとメディアータ・マシンの間の接続をテストし終わったら、診断モジュールを有効にして、診断グラフに含めるトランザクション・データのサンプリング・パーセンテージを指定します。

注： [診断の分布] ダイアログ・ボックスは、シナリオ実行中には無効になっています。シナリオを実行する前に診断モジュールを有効にして設定しておく必要があります。

トランザクション・ブレイクダウンを有効にするには、次の手順で行います。

- 1 コントローラのメニューで、**[診断]** > **[設定]** を選択します。**[診断の分布]** ダイアログ・ボックスが開きます。



- 2 トランザクション・ブレイクダウンの監視を有効にするには、**[次の診断を有効化]** を選択し、302 ページ「**[診断の分布]** ダイアログ・ボックスについて」の説明に従って、監視対象に加える仮想ユーザの割合を設定します。
- 3 標準設定では、Web ページ診断が有効になっています。Web ページ診断を無効にするには、**[Disable]** をクリックします。Web ページ診断が無効になっている場合は、**[Enable]** をクリックすると有効にできます。

オフライン診断を有効にして設定するには、トランザクション・ブレイクダウン実行の対象にする診断タイプの横にある **[設定]** ボタンをクリックします。

- ▶ Siebel 診断の設定については、第 20 章「Siebel 診断モジュールの設定」を参照してください。
- ▶ Siebel DB 診断の設定については、第 21 章「Siebel DB 診断モジュールの設定」を参照してください。
- ▶ Oracle 11i 診断の設定については、第 22 章「Oracle 11i 診断モジュールの設定」を参照してください。
- ▶ SAP 診断の設定については、第 23 章「SAP 診断モジュールの設定」を参照してください。

- 4 オンラインおよびオフライン診断を有効にして設定するには [J2EE/.NET Diagnostics (仮想ユーザ サンプリング最高率: 100%)] の横にある [設定] ボタンをクリックします。

J2EE/.NET 診断の設定については、『Mercury Diagnostics for J2EE & .NET Installation and User's Guide』（英語版）を参照してください。

【診断の分布】ダイアログ・ボックスについて

- ▶ **[次の診断を有効化]** : オフラインの Web ページ, Siebel, Siebel DB, Oracle 11i, および SAP ブレークダウン・グラフの生成と, オンラインとオフラインの J2EE および .NET ブレークダウン・グラフの生成を有効にします。
- ▶ **[使用中のシナリオ内にある関連する全仮想ユーザ X を対象]** : 実行するトランザクション・ブレークダウンに Web 仮想ユーザの割合 (パーセント) を指定します。この割合により, コントローラに報告される, アプリケーション・サーバ上のトランザクションの数が決定します。この数値を減らすと, Web ページ診断, Oracle 11i 診断, および J2EE/.NET 診断によるアプリケーション・サーバのオーバーヘッドが減少します。

例えば, サンプリング値として 25% を入力し, **script1** で 12 仮想ユーザ, **script2** で 8 仮想ユーザ, **script3** で 1 仮想ユーザを実行した場合には, **script1** で 3 仮想ユーザ, **script2** で 2 仮想ユーザ, **script3** で 1 仮想ユーザに対してトランザクション・ブレークダウンが実行されます。

注 : 最小トランザクション・ブレークダウンは, スクリプトあたり 1% または 1 仮想ユーザのいずれか多い方です。最大値は, 選択されたすべての診断タイプの最大パーセンテージのうちで最も小さい値です。

- ▶ **[オフライン診断]** : 以下のオフライン診断モジュールを有効にして設定するには, そのモジュールに対応する [設定] ボタンをクリックします。
- ▶ **[Web ページ診断]** : Web ページ・ブレークダウン・グラフを生成します。このグラフには, スクリプト内で定義された各トランザクションおよびサブトランザクションのパフォーマンス情報が表示されます。トランザクション・ブレークダウン実行の対象にできる仮想ユーザの最大数は 10% です。
- ▶ **[Siebel 診断]** : Siebel ブレークダウン・グラフを生成します。このグラフでは, Siebel トランザクションをレイヤ, エリア, サブエリア, サーバ, およびスクリプトにブレークダウンできます。これにより, 時間が費やされている箇所を正確に特定できます。また, トランザクションの呼び出しチェーンと呼び出しスタックの統計情報を表示して, トランザクションの各部分で費やされる時間の割合を知ることができます。トランザクション・ブレークダ

ウン実行の対象にできる仮想ユーザの最大数は、10% または 100 を超えない数です。

- ▶ **[Siebel DB 診断]** : Siebel DB ブレークダウン・グラフを生成します。このグラフでは、トランザクションをその SQL ステートメントおよびステージにブレークダウンできます。これにより、Siebel データベース呼び出しのあらゆるレイヤの問題エリアを特定できます。トランザクション・ブレークダウン実行の対象にできる仮想ユーザの最大数は 10% です。
- ▶ **[Oracle 11i 診断]** : Oracle 11i ブレークダウン・グラフを生成します。このグラフでは、トランザクションをその SQL ステートメントおよびステージにブレークダウンできます。これにより、Oracle データベース呼び出しのあらゆるレイヤの問題エリアを特定できます。トランザクション・ブレークダウン実行の対象にできる仮想ユーザの最大数は 5% です。
- ▶ **[SAP 診断]** : SAP トランザクション・ブレークダウン・グラフを生成します。このグラフでは、トランザクションを DB、アプリケーション、システム、およびインタフェースのパフォーマンス・レベルまでブレークダウンできます。これにより、SAP 環境のあらゆるコンポーネントの問題エリアを特定できます。トランザクション・ブレークダウン実行の対象にできる仮想ユーザの最大数は 100% です。
- ▶ **[オンライン & オフライン診断]** : 以下のオフラインおよびオフライン診断モジュールを有効にして設定するには、そのモジュールに対応する **[設定]** ボタンをクリックします。
- ▶ **[J2EE/.NET 診断]** : オンラインおよびオフラインの J2EE/.NET 診断グラフを提供します。オンライン診断では、システムのサーバ側での活動のチェーン全体を表示できます。J2EE/.NET レイヤをコンポーネントとメソッドにブレークダウンして、時間がかかっている箇所を正確に特定できます。また、トランザクションの呼び出しチェーンと呼び出しスタックの統計情報を表示して、トランザクションの各部分で費やされる時間の割合を知ることがもできます。トランザクション・ブレークダウン実行の対象にできる仮想ユーザの最大数は 100% です。

シナリオの実行後に J2EE/.NET 診断グラフを使用して、サーバのパフォーマンスを分析してレポートを生成できます。詳細については、『**Mercury Diagnostics for J2EE & .NET for LoadRunner Version 8.0 Installation and User's Guide**』（英語版）を参照してください。

診断結果の表示



トランザクション・ブレイクダウンの結果を表示するには、コントローラの [実行] タブで、**[結果]** > **[結果の分析]** を選択するか、**[LoadRunner アナリシスを起動し結果を分析します]** ボタンをクリックします。LoadRunner アナリシスが開きます。

アナリシスのトランザクション・ブレイクダウン・グラフとレポートを使用して、パフォーマンス・データを表示し、ドリル・ダウンによってアプリケーションのあらゆるレイヤの問題エリアを特定できます。

ERP/CRM トランザクション・ブレイクダウン診断グラフの詳細については、『**Mercury LoadRunner アナリシス・ユーザーズ・ガイド**』を参照してください。

第 20 章

Siebel 診断モジュールの設定

Siebel 診断モジュールでは、Siebel トランザクションをレイヤ、エリア、サブエリア、サーバ、およびスクリプトにブレイクダウンできます。また、トランザクションの呼び出しチェーンと呼び出しスタックの統計情報を表示して、トランザクションの各部分で費やされる時間の割合を知ることができます。Siebel Web 仮想ユーザは、Siebel 診断をサポートします。

本章では、次の項目について説明します。

- ▶ アプリケーション・サーバおよび Web サーバ上での Siebel 診断モジュールの設定
- ▶ Web サーバが DMZ 内にある場合の診断モジュールの設定
- ▶ Siebel アプリケーション・サーバからメディアータへのファイルのコピー
- ▶ Siebel 診断モジュールの設定

注：Siebel 診断モジュール（Siebel Application Response Measurements）は、バージョン 7.53 および 7.7 の Siebel アプリケーション・サーバをサポートしています。

アプリケーション・サーバおよび Web サーバ上での Siebel 診断モジュールの設定

Siebel アプリケーション・サーバおよび Web サーバで診断データの収集の設定を行うには、次の操作を実行します。

- ▶ 負荷テストに関わるすべての Siebel アプリケーション・サーバおよび Web サーバで Siebel 診断モジュールを有効にする
- ▶ サーバのパフォーマンス設定を最適化する

- ▶ Siebel サーバ ID のリストを生成する (Siebel アプリケーション・サーバの場合にのみ必要)

Siebel 診断を有効にするには、次の手順を実行します。

- 1 Siebel サーバの環境変数を次のように設定します。
`SIEBEL_SarmEnabled=true`
- 2 サーバを再起動します。

サーバのパフォーマンスを最適化するには、次の手順を実行します。

次の変数を使って、最大メモリ・キャッシュとファイル・サイズを変更できます。

`SIEBEL_SarmMaxMemory= <バイト数>`

`SIEBEL_SarmMaxFileSize = <バイト数>`

`SIEBEL_SarmMaxMemory` は、Siebel が情報を Siebel ログ・ファイルに書き込む前にメモリ内に保持するバッファのサイズを制御します。パラメータ値を大きくすることでサーバのパフォーマンスを向上させられます。ただし、実行終了時からの情報がアナリシス・グラフから欠落することになります。サーバの負荷が小さい場合は `SIEBEL_SarmMaxMemory= 50000`、サーバの負荷が大きい場合は `SIEBEL_SarmMaxMemory= 1000000` に設定することをお勧めします。サーバに対する負荷は、仮想ユーザ数が 20 以下の場合を小さい、仮想ユーザ数が 100 以上の場合を大きいと判断します。

`SIEBEL_SarmMaxFileSize` で設定する推奨ファイル・サイズは、サーバの負荷が小さい場合は `5000000`、負荷が大きい場合は `25000000` です。サーバ上で複数の Siebel ログ・ファイルが 10 秒ごとに生成される場合は、`SIEBEL_SarmMaxFileSize` の値を大きくする必要があります。

注： Siebel 7.7 では、パラメータがわずかに異なります。

注： 負荷テストを実行する前に、そのテストに関係するすべてのサーバから Siebel 診断ログ (*.sarm ファイル) を削除する必要があります。

Siebel サーバ ID を生成するには、次の手順を実行します。

Siebel アプリケーション・サーバで、コマンド・ウィンドウを開いて次のコマンドを実行します。

```
< Siebel bin ディレクトリ > %svrmgr /u <ユーザ名> /p <パスワード> /g <
ゲートウェイ・サーバ> /e <エンタープライズ・サーバ> /c "list servers show
SBLSRVR_NAME, SV_SRVRID"
```

引数の意味は次のとおりです。

/u <ユーザ名> は、サーバ管理者のユーザ名です。

/p <パスワード> は、サーバ管理者のパスワードです。

/g <ゲートウェイ・サーバ> は、ゲートウェイ・サーバのアドレスです。

/e <エンタープライズ・サーバ> は、エンタープライズ・サーバの名前です。

/c <コマンド> は、実行される単一のコマンドです。

このコマンドは、すべての Siebel アプリケーション・サーバとその ID のリストを生成します。サーバ ID の情報は [Siebel サーバ設定] ダイアログ・ボックスで必要になるので、サーバ ID の記録を保存してください。詳細については、309 ページ「Siebel 診断モジュールの設定」の手順 9 を参照してください。

Web サーバが DMZ 内にある場合の診断モジュールの設定

内部ネットワーク内のアプリケーション・サーバと、DMZ（会社の私的なデータへの外部からのアクセスを防ぐために使用される、内部ネットワークを公共ネットワークから隔てる「中立ゾーン」）内の Web（ファイル）サーバを使用している場合は、メディアータを内部（ファイアウォール越しの）LAN にインストールして、内部マシンから DMZ 内のファイル・サーバへの SMB/CIFS 通信を有効にする必要があります。SMB/CIFS は、トランスポート・プロトコルとして NBT（NetBIOS over TCP/IP）を使用するファイル共有サービスです。

クライアント（ファイアウォール・マシン越しの）とファイル・サーバの間で NBT プロトコルを有効にするには、次のポート設定を使用します。

ファイル共有サービス	ポート
SMB/CIFS over NBT	TCP 139 (SMB)
CIFS over TCP/IP (Direct SMB)	TCP 445

例えば、ファイアウォール設定は次のように指定します。

Service enabled: "nbsession" for TCP 139 connection.

Service enabled: "Microsoft-ds" for TCP 445 connection.

注：CIFS over TCP 445 (direct SMB over TCP/IP) は、Windows 2000 以降ではオプションです (Windows 2000 以降の方がファイル・サーバとの通信手段としてより安全なため)。CIFS over TCP/IP を有効にするには、オペレーティング・システム設定を使って NetBIOS over TCP/IP プロトコルを無効にする必要があります。

Siebel アプリケーション・サーバからメディアータへのファイルのコピー

アプリケーション・サーバの設定後、以下のファイルを Siebel アプリケーション・サーバの **¥bin** ディレクトリから、メディアータ・マシンの **< LR メディアータのインストール先フォルダ > ¥bin** ディレクトリ、**< Windows > ¥System32** ディレクトリ、または **PATH** 内のその他のディレクトリにコピーする必要があります。

Siebel 7.53 の場合は次のファイルをコピーします。

<ul style="list-style-type: none">• sarmanalyzer.exe• ssllever.dll• sslsym.dll	<ul style="list-style-type: none">• sslshar.dll• sslcosa.dll
----------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------

Siebel 7.7 の場合は次のファイルをコピーします。

• sarmanalyzer.exe	• sslcosa.dll
• libarm.dll	• sslcosd.dll
• msvcp70.dll	• sslcrsa.dll
• msucr70.dll	• sslcscr.dll
• sslcacln.dll	• sslcshar.dll
• sslccore.dll	• sslcsrd.dll
• sslcevt.dll	• sslcsym.dll
• sslcos.dll	• sslcver.dll

Siebel 診断モジュールの設定

診断データを生成するには、Siebel 診断モジュールをメディアエータと通信するように設定し、監視対象のサーバを定義する必要があります。その後、300 ページ「診断の有効化」で説明しているように、診断モジュールを有効にして、診断グラフに含めるトランザクション・データのサンプリング・パーセンテージを指定できます。

注：

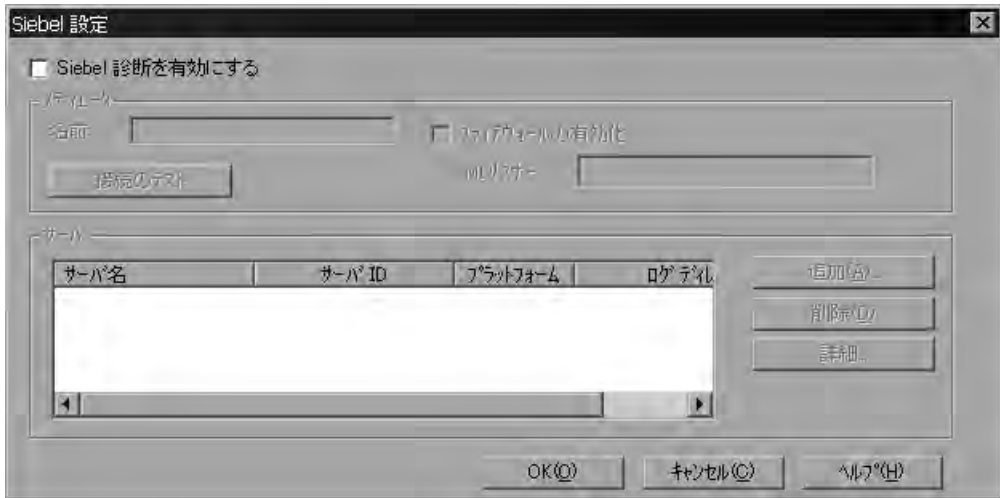
- ▶ ここで行う設定はシナリオごとです。シナリオ内のスクリプトはすべて、同じ診断設定で実行されます。
 - ▶ 有効な診断データが生成されたことを確認するには、自動トランザクションを使用するのではなく、仮想ユーザ・スクリプトでトランザクションを手動で定義します。実行環境の設定の [一般：その他] ノードで、[各アクションをトランザクションとして定義する] オプションと [各ステップをトランザクションとして定義する] を無効にすることを確認してください。
-

Siebel 診断モジュールを設定するには、次の手順を実行します。

- 1 コントローラのメニューで、[診断] > [設定] を選択します。[診断の分布] ダイアログ・ボックスが開きます。
- 2 [次の診断を有効化] が選択されていることを確認します。

- 3 [オフライン診断] セクションで, [Siebel 診断 (仮想ユーザ サンプリング最高率: 10%)] の隣にある [設定] ボタンをクリックします。

[Siebel 設定] ダイアログ・ボックスが開きます。



- 4 [Siebel 診断を有効にする] を選択します。これにより, ダイアログ・ボックス内のすべての他のフィールドが有効になります。
- 5 312 ページ「[Siebel 設定] ダイアログ・ボックスについて」の説明に従って, メディエータ情報を入力します。
- 6 ファイアウォール越しに監視を行う場合は, [ファイアウォールの有効化] を選択して, Mercury リスナー・マシンの名前または IP アドレスを入力します。詳細については, 第 16 章「LoadRunner でのファイアウォールの使用」を参照してください。
- 7 コントローラとメディエータとの間の接続をテストするには, [接続のテスト] をクリックします。Siebel 診断モジュールが, メディエータに接続しようと試みます。

接続が成功しない場合は, [出力] ウィンドウで詳細を確認します。[出力] ウィンドウは, ステータス・バーの [エラー] リンクをクリックすると表示されます。

- 8 **[追加]** をクリックして Siebel サーバを追加します。**[Siebel サーバ設定]** ダイアログ・ボックスが開きます。



- 9 313 ページ「**[Siebel サーバ設定]** ダイアログ・ボックスについて」の説明に従って Siebel サーバ情報を入力し、**[OK]** をクリックして **[Siebel サーバ設定]** ダイアログ・ボックスを閉じます。
- 10 **[Siebel 設定]** ダイアログ・ボックスで **[詳細]** をクリックして、選択した Siebel サーバのサーバ情報を表示します。
- 11 **[OK]** をクリックして **[Siebel 設定]** ダイアログ・ボックスを閉じます。**[診断の分布]** ダイアログ・ボックス内の **[Siebel 診断 (仮想ユーザ サンプリング 最高率: 10%)]** の横にチェック・マーク・アイコンが表示され、Siebel 診断が有効になったことを示します。
- 12 **[OK]** をクリックして、**[診断の分布]** ダイアログ・ボックスを閉じます。



[Siebel 設定] ダイアログ・ボックスについて

[Siebel 設定] ダイアログ・ボックスでは、メディエータを定義し、監視対象サーバの詳細を入力し、コントローラとメディエータ・マシンの間の接続をテストすることができます。

[**Siebel 診断を有効にする**]：このオプションを選択して、Siebel 診断を有効にして、Siebel 診断の設定を行います。

[メディエータ]

[**名前**]：Siebel 診断データの収集と処理に使用するメディエータの名前を入力します。各診断モジュールに対してサポートされるメディエータは1台だけです。

注：ファイアウォール越しにメディエータを使用する場合は、メディエータ・マシン名ではなく、メディエータのローカル・マシン・キーを入力してください。

[**ファイアウォールの有効化**]：メディエータがファイアウォール越しかどうかを選択します。

[**MI リスナー**]：ファイアウォール越しに監視する場合は、Mercury リスナー・マシンの名前、フルネーム、または IP アドレスを入力します。

[**接続のテスト**]：Siebel 診断モジュールとメディエータの間の接続をテストします。

注：テスト接続では、Siebel サーバへの接続は検証されません。

[サーバ]

[**サーバ名**]：Siebel サーバの名前。

[**サーバ ID**]：Siebel サーバ ID (Siebel アプリケーション・サーバの場合のみ)。

[**プラットフォーム**]：Siebel サーバのプラットフォーム。

[**ログ ディレクトリ**]：Siebel ログ・ファイル (*.SARM) が書き込まれる Siebel サーバ上のディレクトリ。

[**追加**] : [Siebel サーバ設定] ダイアログ・ボックスを開きます。Siebel サーバの情報を入力できます。

[**削除**] : サーバ・リストからサーバを削除します。

[**詳細**] : 選択されたサーバに関する情報を表示します。

[Siebel サーバ設定] ダイアログ・ボックスについて

[Siebel サーバ設定] ダイアログ・ボックスでは、Siebel サーバの情報を入力できます。

[**サーバ名**] : Siebel サーバの名前を入力します。

[**サーバタイプ**] : Siebel サーバの種類を選択します。

[**OS**] : Siebel サーバのプラットフォームを選択します。

[**セキュリティ保護されたシェルを使用する**] : セキュア・シェル接続を使用するかどうかを選択します。

[**アプリケーション サーバ ID**] : Siebel サーバ ID を入力します (Siebel アプリケーション・サーバの場合にのみ必要)。サーバ ID のリストを生成する方法については、305 ページ「アプリケーション・サーバおよび Web サーバ上での Siebel 診断モジュールの設定」を参照してください。

[**サーバ ログ ディレクトリ**] : Siebel アプリケーションがログ・ファイル (*.SARM) を保存する場所を入力します。ログ・ファイルは、Siebel サーバの共有ログ・ディレクトリに保存することも、別個のフォルダに保存することもできます。

[**ユーザ名**] : ログ・ファイルが保存されるマシンのユーザ名を入力します。

注 : Windows プラットフォームの場合、このユーザは管理者権限を持っている必要があります。詳細については、297 ページ「リモート Windows サーバへの接続」を参照してください。UNIX プラットフォームについては、298 ページ「リモート UNIX サーバへの接続」を参照してください。

[**パスワード/パスフレーズ**] : ユーザのパスワードまたはパスフレーズを入力します。

[**秘密鍵ファイル**] : 秘密鍵が保存されるファイルの名前を入力します。このファイルはメタデータ上にあります。パスを指定せずにファイル名だけを指定した場合は、自動的にメタデータの < **LoadRunner** > **¥bin** ディレクトリ内のファイルが探されます。

[**ドメイン**] : Siebel サーバのドメインを入力します。

第 21 章

Siebel DB 診断モジュールの設定

Siebel DB 診断モジュールは、データベースのパフォーマンスの問題をすばやく突き止めて解決するのに役立ちます。各トランザクションの SQL を表示し、各スクリプトの問題のある SQL クエリを特定して、どの時点で問題が発生したかを突き止めることができます。Siebel Web 仮想ユーザは、Siebel DB 診断モジュールをサポートします。

本章では、次の項目について説明します。

- ▶ スクリプトの準備
- ▶ クロックの同期
- ▶ Siebel サーバでのサーバ・ログの有効化
- ▶ Siebel DB 診断モジュールの設定

スクリプトの準備

診断データの収集のためにスクリプトを準備するときには、1 時間のテストにつき 1 秒の割合で、各トランザクションの最後に思考遅延時間を付け加えることをお勧めします。

セッション ID の競合を防ぐために、各セッションの終了時に仮想ユーザが必ず Siebel システムからログオフするようにしてください。

クロックの同期

診断データ収集に先立ってマシンのクロックを同期させることで、トランザクションに対する SQL の相関が正確になります。

Siebel システムのすべてのマシンのクロックが同期していることを確認します。

Windows Siebel サーバでは、ロード・ジェネレータ・マシンから次のコマンドを実行して、Siebel ゲートウェイとロード・ジェネレータ・マシンのクロックを合わせます。

```
net time ¥<ゲートウェイ名> /set /y
```

<ゲートウェイ名>の部分には、Siebel ゲートウェイの名前を指定します。

Unix Siebel サーバの場合は、次のいずれかの方法でクロックを合わせることができます。

- ▶ UNIX Siebel ゲートウェイ・サーバで **date** コマンドを使用して、ロード・ジェネレータのクロックと同期するよう手作業で時刻を変更する。
- ▶ ロード・ジェネレータ・マシンの時刻を UNIX Siebel ゲートウェイ・サーバの時刻と同期するよう変更する。
- ▶ アナリシスで時刻の差異を設定する。詳細については、『**Mercury LoadRunner アナリシス・ユーザズ・ガイド**』の Siebel DB 診断グラフに関する章を参照してください。

Siebel サーバでのサーバ・ログの有効化

コントローラで Siebel DB 診断モジュールを設定する前に、Siebel サーバをデータベース・ログ・ファイルを作成するように設定する必要があります。

Siebel サーバ上でログ機能を有効にするには、次の手順を実行します。

- 1 Siebel サーバで、コマンド・ウィンドウを開いて次のコマンドを実行します。

```
< Siebel bin ディレクトリ> ¥svrmgr /g <ゲートウェイ・サーバ> /s < Siebel サーバ> /e <エンタープライズ・サーバ名> /u <ユーザ名> /p <パスワード>
```

/u <ユーザ名> はサーバ管理者のユーザ名です。

/p <パスワード> はサーバ管理者のパスワードです。

/g <ゲートウェイ・サーバ> はゲートウェイ・サーバのアドレスです。

/e <エンタープライズ・サーバ> はエンタープライズ・サーバの名前です。

/c <コマンド> は 1 回のコマンドの実行です。

2 次のコマンドを入力します。

```
change evtloglvl ObjMgrsqllog=4 for comp <コンポーネント名>
```

```
evtloglvl EventContext=3 for comp <コンポーネント名>
```

```
evtloglvl ObjMgrSessionInfo =3 for comp <コンポーネント名>
```

例えば、コール・センター・コンポーネントの場合は、コンポーネント名として次のように入力します。

```
change evtloglvl ObjMgrsqllog=4 for comp sccobjmgr_enu
```

Siebel サーバ上でログ機能を無効にするには、次の手順を実行します。

Siebel サーバで、次のコマンドを入力します。

```
change evtloglvl ObjMgrsqllog=0 for comp <コンポーネント名>
```

```
change evtloglvl EventContext=0 for comp <コンポーネント名>
```

```
change evtloglvl ObjMgrSessionInfo =0 for comp <コンポーネント名>
```

注： 負荷テストを実行する前に、そのテストに関係するすべてのサーバからログ・ファイルを削除する必要があります。

Siebel DB 診断モジュールの設定

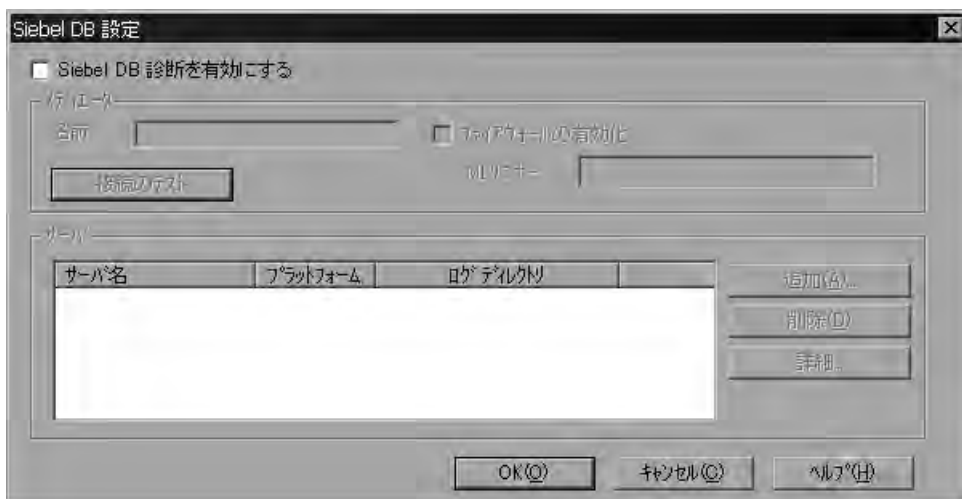
診断データを生成するには、Siebel DB 診断モジュールをメディアータと通信するように設定し、監視対象のサーバを定義する必要があります。その後、300 ページ「診断の有効化」で説明しているように、診断モジュールを有効にして、診断グラフに含めるトランザクション・データのサンプリング・パーセンテージを指定できます。

注：

- ▶ ここで行う設定はシナリオごとです。シナリオ内のスクリプトはすべて、同じ診断設定で実行されます。
- ▶ 有効な診断データが生成されたことを確認するには、自動トランザクションを使用するのではなく、仮想ユーザ・スクリプトでトランザクションを手動で定義します。実行環境の設定の [**一般：その他**] ノードで、[**各アクションをトランザクションとして定義する**] オプションと [**各アクションをステップとして定義する**] を無効にすることを確認してください。

Siebel DB 診断モジュールを設定するには、次の手順を実行します。

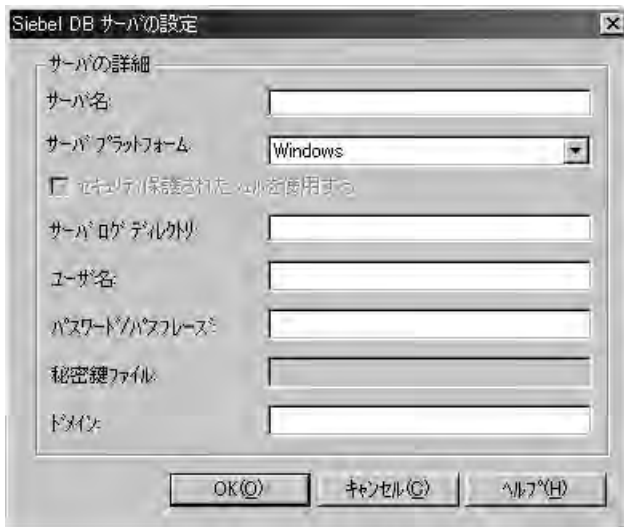
- 1 コントローラのメニューで、[**診断**] > [**設定**] を選択します。[診断の分布] ダイアログ・ボックスが開きます。
- 2 [**次の診断を有効化**] が選択されていることを確認します。
- 3 [**オフライン診断**] セクションで、[**Siebel DB 診断（仮想ユーザ サンプリング 最高率：10%）**] の隣にある [**設定**] ボタンをクリックします。[Siebel DB 設定] ダイアログ・ボックスが開きます。



- 4 **[Siebel DB 診断を有効にする]** を選択します。これにより、ダイアログ・ボックス内のすべての他のフィールドが有効になります。
- 5 320 ページ「[Siebel DB 設定] ダイアログ・ボックスについて」の説明に従って、メディエータ情報を入力します。
- 6 ファイアウォール越しに監視を行う場合は、**[ファイアウォールの有効化]** を選択して、Mercury リスナー・マシンの名前または IP アドレスを入力します。詳細については、第 16 章「LoadRunner でのファイアウォールの使用」を参照してください。
- 7 コントローラとメディエータとの間の接続をテストするには、**[接続のテスト]** をクリックします。Siebel DB 診断モジュールが、メディエータに接続しようと試みます。

接続が成功しない場合は、[出力] ウィンドウで詳細を確認します。[出力] ウィンドウは、ステータス・バーの **[エラー]** リンクをクリックすると表示されます。

- 8 [追加] をクリックして Siebel DB サーバを追加します。[Siebel DB サーバの設定] ダイアログ・ボックスが開きます。



- 9 322 ページ「[Siebel DB サーバの設定] ダイアログ・ボックスについて」の説明に従って Siebel サーバ情報を入力し、[OK] をクリックして [Siebel DB サーバの設定] ダイアログ・ボックスを閉じます。
- 10 [Siebel DB 設定] ダイアログ・ボックスで [詳細] をクリックして、選択した Siebel サーバのサーバ情報を表示します。
- 11 [OK] をクリックして [Siebel DB 設定] ダイアログ・ボックスを閉じます。
[診断の分布] ダイアログ・ボックス内の [Siebel DB 診断 (仮想ユーザ サンプリング最高率: 10%)] の横にチェック・マーク・アイコンが表示され、Siebel 診断が有効になったことを示します。
- 12 [OK] をクリックして、[診断の分布] ダイアログ・ボックスを閉じます。



[Siebel DB 設定] ダイアログ・ボックスについて

[Siebel DB 設定] ダイアログ・ボックスでは、メディアータを定義し、監視対象サーバの詳細を入力し、コントローラとメディアータ・マシンの間の接続をテストすることができます。

[Siebel DB 診断を有効にする]: このオプションを選択して、Siebel DB 診断を有効にし、Siebel DB 診断の設定を行います。

[メディアータ]

[名前] : Siebel 診断データの収集と処理に使用するメディアータの名前を入力します。各診断モジュールに対してサポートされるメディアータは 1 台だけです。

注 : Siebel DB 診断ファイルをアプリケーション・サーバからメディアータに、そしてメディアータから結果ディレクトリにコピーするには、長い時間がかかる場合があります。コピーの第 1 段階では、[Summary Data Processing] ウィンドウが表示されます。コピーの第 2 段階にかかる時間を最小限に抑えるために (ファイアウォール越しの監視ではない場合)、メディアータ・マシンとして **localhost** を使用することをお勧めします。Siebel DB 診断ファイルがすでにコントローラ・マシン上に存在するため、これにより診断ファイルを結果ディレクトリにコピーするためにかかる時間が短縮されます。ファイアウォール越しにメディアータを使用する場合は、メディアータ・マシン名ではなく、メディアータのローカル・マシン・キーを入力してください。

[ファイアウォールの有効化] : メディアータがファイアウォール越しかどうかを選択します。

[MI リスナー] : ファイアウォール越しに監視する場合は、Mercury リスナー・マシンの名前、フルネーム、または IP アドレスを入力します。

[接続のテスト] : Siebel DB 診断モジュールとメディアータ・マシンの間の接続をテストします。

注 : テスト接続では、Siebel サーバへの接続は検証されません。

[サーバ]

[サーバ名] : Siebel サーバの名前。

[プラットフォーム] : Siebel サーバのプラットフォーム。

[ログ ディレクトリ] : Siebel ログ・ファイルが書き込まれるディレクトリ。

[追加] : [Siebel DB サーバの設定] ダイアログ・ボックスを開きます。Siebel サーバの情報を入力できます。

[**削除**] : サーバ・リストからサーバを削除します。

[**詳細**] : 選択されたサーバに関する情報を表示します。

[Siebel DB サーバの設定] ダイアログ・ボックスについて

[Siebel DB サーバの設定] ダイアログ・ボックスでは、Siebel サーバの情報を入力できます。

[**サーバ名**] : Siebel サーバの名前を入力します。

[**サーバプラットフォーム**] : Siebel サーバのプラットフォームを選択します。

[**セキュリティ保護されたシェルを使用する**] : セキュア・シェル接続を使用するかどうかを選択します。

[**サーバログディレクトリ**] : Siebel アプリケーションがログ・ファイルを保存する場所を入力します。ログ・ファイルは、Siebel サーバの共有ログ・ディレクトリに保存することも、別個のフォルダに保存することもできます。

[**ユーザ名**] : ログ・ファイルが保存されるマシンのユーザ名を入力します。

注 : Windows プラットフォームの場合、このユーザは管理者権限を持っている必要があります。詳細については、297 ページ「リモート Windows サーバへの接続」を参照してください。UNIX プラットフォームについては、298 ページ「リモート UNIX サーバへの接続」を参照してください。

[**パスワード/パスフレーズ**] : ユーザのパスワードまたはパスフレーズを入力します。

[**秘密鍵ファイル**] : 秘密鍵が保存されるファイルの名前を入力します。このファイルはメディアータ・マシン上にあります。パスを指定せずにファイル名だけを指定した場合は、自動的にメディアータの < **LoadRunner** > %bin ディレクトリ内のファイルが探されます。

[**ドメイン**] : Siebel サーバのドメインを入力します。

第 22 章

Oracle 11i 診断モジュールの設定

Oracle 11i 診断は、Oracle NCA システム上でのパフォーマンスの問題を特定するのに役立ちます。診断情報は、トランザクションから SQL ステートメントへ、そして各ステートメントの SQL ステージへとドリル・ダウンします。Oracle NCA 仮想ユーザは Oracle 11i 診断モジュールをサポートしています。

本章では、次の項目について説明します。

- ▶ Oracle サーバでのサーバ・ログの有効化
- ▶ Oracle NCA アプリケーションのバージョンの選択
- ▶ Oracle 11i 診断モジュールの設定

Oracle サーバでのサーバ・ログの有効化

Oracle サーバ上でサーバ・ログ機能を有効にするには、トレース診断が有効になっていてトレース・ファイルのサイズが無制限に設定されていることを確認する必要があります。標準では、Oracle サーバ上のトレース診断機能はインストール時に有効にされています。また、LoadRunner が Oracle アプリケーション診断のパスワードを処理できるように、仮想ユーザ・スクリプト内で診断パスワードを設定するか、アプリケーション・サーバ上でのパスワード要求を無効にすることができます。

トレース診断が有効になっていることを確認するには、次の手順を実行します。

- 1 管理者権限を持つユーザとして Oracle アプリケーション・サーバにログオンし、[Oracle Application] でモジュールを選択します。[Responsibilities] ダイアログ・ボックスが開きます。
- 2 [System Administrator] を選択して [OK] をクリックします。
- 3 [Functions] タブで [Profile] > [System] を選択し、[Open] をクリックします。[System Profile Values] ダイアログ・ボックスが開きます。

- 4 [Display] セクションで, [Site] と [Profiles with No Values] を選択し, [Profiles] フィールドに %Diagnostics% と入力して [Find] をクリックします。
- 5 無効になっている診断プロファイルがあったら ([Site] カラムが「Yes」に設定されている), 設定を「No」に変更します。
- 6 設定を保存します。

トレース・ファイルのサイズを無制限に設定するには, 次の手順を実行します。

Oracle 9i の場合

Oracle サーバ上で, SQL エディタを使って次のコマンドを実行します。

```
Alter system set max_dump_file_size=UNLIMITED scope=both;
```

Oracle 8i の場合

- 1 Oracle サーバ上で, SQL エディタを使って次のコマンドを実行します。

```
Alter system set max_dump_file_size=2048000;
```
- 2 `$ORACLE_HOME\admin<sid>\pfile\init<sid>.ora` にある `init*.ora` ファイルを編集します。パラメータの行を見つけて値を変更し, ファイルを保存します。

注：トレース・ファイルは非常に大きくなる場合があるので, データベース・サーバに十分なディスク領域があることを確認してください。

仮想ユーザ・スクリプト内に診断パスワードを設定するには, 次の手順を実行します。

VuGen で, スクリプトに `nca_set_diagnostics_password(<password>)` 関数を追加し, パスワードを選択します。

注：`nca_set_diagnostics_password` 関数は, `nca_connect_server` 関数の後に追加しなければなりません。

アプリケーション・サーバ上での診断パスワード要求を無効にするには、次の手順を実行します。

- 1 管理者権限を持つユーザとして Oracle アプリケーション・サーバにログオンし、[Oracle Application] でモジュールを選択します。[Responsibilities] ダイアログ・ボックスが開きます。
- 2 [System Administrator] を選択して [OK] をクリックします。
- 3 [Functions] タブで [Profile] > [System] を選択し、[開く] をクリックします。[System Profile Values] ダイアログ・ボックスが開きます。
- 4 [Display] セクションで [User] を選択し、目的のユーザ名を入力します。[Profile] フィールドに %Utilities:Diagnostics% と入力し、[Find] をクリックします。[Utilities:Diagnostics] プロファイルの値が表示されます。
- 5 [Utilities:Diagnostics] プロファイルの [User] カラムで、値を「Yes」に設定します。
- 6 設定を保存します。

Oracle NCA アプリケーションのバージョンの選択

Oracle 11i 診断モジュールは、Oracle NCA のバージョン 11.5.0 以降をサポートしています。組み込みのトレース機構を有効にするには、VuGen の実行環境設定で Oracle アプリケーション・サーバのバージョンを入力します。Oracle サーバのバージョンをチェックするには、Oracle サーバにログインして、[Help] > [About Oracle] をクリックします。すると、Oracle サーバのバージョンが [Oracle Application] フィールドに表示されます。

Oracle アプリケーションのバージョンを入力するには、次の手順を実行します。

VuGen でスクリプトを開き、[仮想ユーザ] > [実行環境の設定] を選択します。[Oracle NCA: Client Emulation] ノードで、使用している Oracle NCA のバージョンを [Diagnostics] > [Application Version] フィールドで選択します。

注：組み込みのトレース機構を使って Oracle 11i のトレースを有効にできない場合は、仮想ユーザ・スクリプトの中で `nca_set_custom_dbtrace` 関数と `nca_set_dbtrace_file_index` 関数を使って手作業で有効にすることができます。標準 UI を備えていないカスタム・アプリケーションを使用している場合は、この作業が必要になる可能性があります。

注：負荷テストを実行する前に、そのテストに関係するすべてのサーバからトレース・ログ・ファイルを削除する必要があります。

診断モジュールの実行中に現実のユーザと他のユーザが Oracle サーバを使用すると、診断の結果に影響する可能性があるため、現実のユーザが Oracle サーバを使用しないようにする必要があります。

Oracle 11i 診断モジュールの設定

診断データを生成するには、Oracle 11i 診断モジュールをメディアータ・マシンと通信するように設定し、監視対象のサーバを定義する必要があります。その後、300 ページ「診断の有効化」で説明しているように、診断モジュールを有効にして、診断グラフに含めるトランザクション・データのサンプリング・パーセンテージを指定できます。

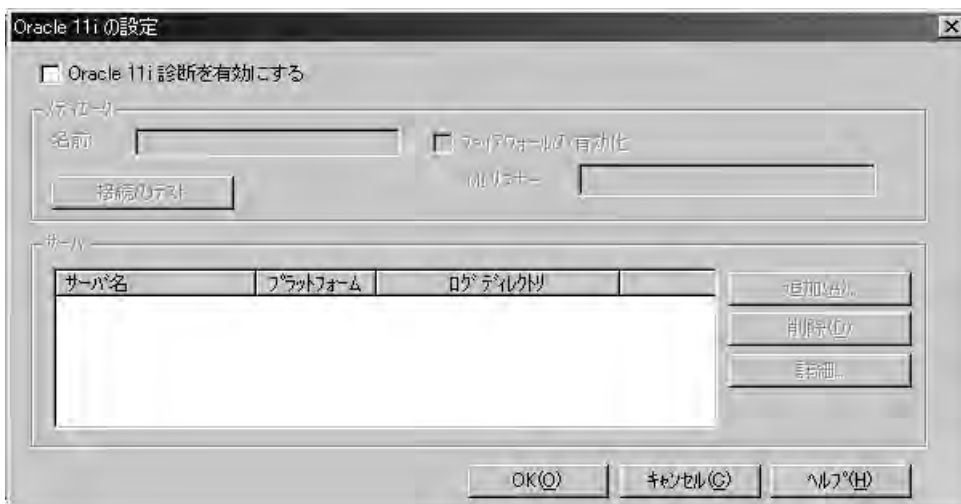
注：

- ▶ ここで行う設定はシナリオごとです。シナリオ内のスクリプトはすべて、同じ診断設定で実行されます。
- ▶ 有効な診断データが生成されたことを確認するには、自動トランザクションを使用するのではなく、仮想ユーザ・スクリプトでトランザクションを手動で定義します。実行環境の設定の [一般：その他] ノードで、[各アクションをトランザクションとして定義する] オプションと [各ステップをトラン

ザクシオンとして定義する] を無効にすることを確認してください。

Oracle 11i 診断モジュールを設定するには、次の手順を実行します。

- 1 コントローラのメニューで、[診断] > [設定] を選択します。[診断の分布] ダイアログ・ボックスが開きます。
- 2 [次の診断を有効化] が選択されていることを確認します。
- 3 [オフライン診断] セクションで、[Oracle 11i 診断 (仮想ユーザ サンプリング 最高率 : 5%)] の隣にある [設定] ボタンをクリックします。
[Oracle 11i の設定] ダイアログ・ボックスが開きます。

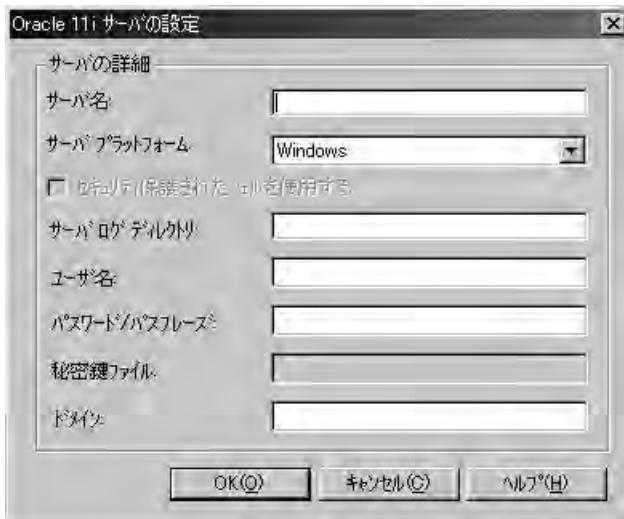


- 4 [Oracle 11i 診断を有効にする] を選択します。これにより、ダイアログ・ボックスの他のすべてのフィールドが有効になります。
- 5 329 ページ「[Oracle 11i の設定] ダイアログ・ボックスについて」の説明に従って、メタデータ情報を入力します。
- 6 ファイアウォール越しに監視を行う場合は、[ファイアウォールの有効化] を選択して、Mercury リスナー・マシンの名前または IP アドレスを入力します。詳細については、第 16 章「LoadRunner でのファイアウォールの使用」を参照してください。

- 7 コントローラとメディアータとの間の接続をテストするには、[接続のテスト] をクリックします。Oracle 11i 診断モジュールが、メディアータに接続しようと試みます。

接続が成功しない場合は、[出力] ウィンドウで詳細を確認します。[出力] ウィンドウは、ステータス・バーの [エラー] リンクをクリックすると表示されます。

- 8 [追加] をクリックして Oracle サーバを追加します。[Oracle 11i サーバの設定] ダイアログ・ボックスが開きます。



- 9 330 ページ「[Oracle 11i サーバの設定] ダイアログ・ボックスについて」の説明に従って Oracle サーバ情報を入力し、[OK] をクリックします。
- 10 [Oracle 11i の設定] ダイアログ・ボックスで [詳細] をクリックして、選択した Oracle サーバのサーバ情報を表示します。
- 11 [OK] をクリックして [Oracle 11i サーバの設定] ダイアログ・ボックスを閉じます。
[診断の分布] ダイアログ・ボックスで、[Oracle 11i 診断 (仮想ユーザ サンプリング最高率: 5%)] の横に、有効であることを示すチェック・マーク・アイコンが表示されます。
- 12 [OK] をクリックして [診断の分布] ダイアログ・ボックスを閉じます。

[Oracle 11i の設定] ダイアログ・ボックスについて

[Oracle 11i の設定] ダイアログ・ボックスでは、メディエータ・マシンを定義し、監視対象サーバの詳細を入力し、コントローラとメディエータ・マシンの間の接続をテストすることができます。

[**Oracle 11i 診断を有効にする**] : このオプションを選択して Oracle 11i 診断を有効にし、Oracle 11i 診断を設定します。

[メディエータ]

[**名前**] : Oracle 11i 診断データの収集と処理に使用するメディエータ・マシンの名前を入力します。各診断モジュールに対してサポートされるメディエータ・マシンは 1 台だけです。

注 : ファイアウォール越しにメディエータを使用する場合は、メディエータ・マシン名ではなく、メディエータのローカル・マシン・キーを入力してください。

[**ファイアウォールの有効化**] : メディエータがファイアウォール越しかどうかを選択します。

[**MI リスナー**] : ファイアウォール越しに監視する場合は、Mercury リスナー・マシンの名前、フルネーム、または IP アドレスを入力します。

[**接続のテスト**] : Oracle 11i 診断モジュールとメディエータ・マシンの間の接続をテストします。

注 : テスト接続では、Oracle サーバへの接続はチェックされません。

[サーバ]

[**サーバ**] : Oracle サーバの名前。

[**プラットフォーム**] : Oracle サーバのプラットフォーム。

[**ログディレクトリ**] : Oracle トレース・ファイル (*.trc) が書き込まれるディレクトリ。

[追加] : [Oracle 11i サーバの設定] ダイアログ・ボックスを開きます。Oracle サーバの情報を入力できます。

[削除] : サーバ・リストからサーバを削除します。

[詳細] : 選択されたサーバに関する情報を表示します。

[Oracle 11i サーバの設定] ダイアログ・ボックスについて

[Oracle 11i サーバの設定] ダイアログ・ボックスでは、Oracle 11i サーバの情報を入力できます。

[サーバ名] : Oracle サーバの名前を入力します。

[サーバプラットフォーム] : Oracle サーバのプラットフォームを選択します。

[セキュリティ保護されたシェルを使用する] : セキュア・シェル接続を使用するかどうかを選択します。

[サーバログディレクトリ] : Oracle アプリケーションがトレース・ファイルを保存する場所を入力します。トレース・ファイルは、Oracle サーバの共有ディレクトリに保存することも、別個のフォルダに保存することもできます。

[ユーザ名] : トレース・ファイルが保存されるマシンのユーザ名を入力します。

注 : Windows プラットフォームの場合、このユーザは管理者権限を持っている必要があります。詳細については、297 ページ「リモート Windows サーバへの接続」を参照してください。UNIX プラットフォームについては、298 ページ「リモート UNIX サーバへの接続」を参照してください。

[パスワード/パスフレーズ] : ユーザのパスワードまたはパスフレーズを入力します。

[秘密鍵ファイル] : 秘密鍵が保存されるファイルの名前を入力します。このファイルはメディアエータ・マシン上にあります。パスを指定せずにファイル名だけを指定した場合は、自動的にメディアエータの < LoadRunner > %bin ディレクトリ内のファイルが探されます。

[ドメイン] : Oracle サーバのドメインを入力します。

第 23 章

SAP 診断モジュールの設定

SAP 診断モジュールは、問題の根本原因（DBA、ネットワーク、WAS、アプリケーション、OS/HW など）をすばやく簡単に突き止めます。具体的な情報があれば、チームの全員に問題を提示する必要なく適切な専門家だけに相談できます。

本章では、次の項目について説明します。

- ▶ SAP 診断がサポートしている環境
- ▶ SAP 診断モジュールの概要
- ▶ SAP 診断モジュールの設定

SAP 診断がサポートしている環境

次の表に、SAP Application Server および SAPGUI Client のサポートされているバージョンと、必要なカーネル・パッチを示します。

	サポートされているバージョン	必要なカーネル・パッチ
SAP Application Server	4.6C, 4.6D	カーネル・パッチ 1984 (11/01/05 にリリース。SAP ノート 0451251)
	4.7 以降	パッチは不要
SAPGUI Client	SAPGUI for Windows 6.20	最小限のパッチ・レベル：48
	SAPGUI for Windows 6.40	最小限のパッチ・レベル：2

SAP 診断モジュールの概要

サーバ側でのデータ収集

SAP 診断は、SAP DSR (Distributed Statistical Record) に基づいています。アプリケーション・サーバで実行される各ダイアログ・ステップごとに統計レコードが生成され、そのレコードには応答時間コンポーネント、データベース統計、RFC 時間などの情報が含まれます。

シナリオ実行中にサーバから DSR データを抽出するのは、メディアータの役目です。

仮想ユーザの色分け

SAP 診断は、SAPGUI プロトコルの再生に対して適用されます。ユーザは [Configuration] ダイアログで、色分けされた各仮想ユーザのパーセンテージを設定します (100% まで指定できます)。SAPGUI ユーザの色分けによってアプリケーション・サーバでオーバーヘッドが生じることはありません。

SAP 診断モジュールの設定

診断データを生成するには、SAP 診断モジュールをメディアータ・マシンと通信するように設定し、監視対象のサーバを定義する必要があります。その後、300 ページ「診断の有効化」で説明しているように、診断モジュールを有効にして、診断グラフに含めるトランザクション・データのサンプリング・パーセンテージを指定できます。

注：

- ▶ ここで行う設定はシナリオごとです。シナリオ内のスクリプトはすべて、同じ診断設定で実行されます。
 - ▶ 有効な診断データが生成されたことを確認するには、自動トランザクションを使用するのではなく、仮想ユーザ・スクリプトでトランザクションを手動で定義します。実行環境の設定の [一般：その他] ノードで、[各アクションをトランザクションとして定義する] オプションと [各ステップをトランザクションとして定義する] を無効にすることを確認してください。
-

SAP 診断モジュールを設定するには、次の手順を実行します。

- 1 コントローラのメニューで、**[診断]** > **[設定]** を選択します。**[診断の分布]** ダイアログ・ボックスが開きます。
- 2 **[次の診断を有効化]** が選択されていることを確認します。
- 3 **[オフライン診断]** セクションで、**[SAP 診断 (仮想ユーザ サンプリング最高率 : 100%)]** の隣にある **[設定]** ボタンをクリックします。**[SAP 設定]** ダイアログ・ボックスが開きます。



- 4 **[SAP 診断を有効にする]** を選択します。これにより、ダイアログ・ボックス内のほかのフィールドがすべて有効になります。
- 5 334 ページ「**[SAP 設定]** ダイアログ・ボックスについて」の説明に従って、メタデータ情報を入力します。
- 6 ファイアウォール越しに監視を行う場合は、**[ファイアウォールを有効にする]** を選択して、Mercury リスナー・マシンの名前または IP アドレスを入力します。詳細については、第 16 章「LoadRunner でのファイアウォールの使用」を参照してください。
- 7 **[プロパティ]** セクションに、SAP サーバとクライアント・サーバのプロパティを入力します。
- 8 **[検証]** をクリックして、サーバが適切に接続されていることを確認します。
[検証] をクリックすると、コントローラによって、サーバ・ホストを通じた診断が可能なすべてのサーバに関するレポートが生成されます。

- 9 [OK] をクリックして [SAP 設定] ダイアログ・ボックスを閉じます。



[診断の分布] ダイアログ・ボックス内の [SAP 診断 (仮想ユーザ サンプリング最高率: 100%)] の横にチェック・マーク・アイコンが表示され、SAP 診断が有効になったことを示します。

- 10 [OK] をクリックして、[診断の分布] ダイアログ・ボックスを閉じます。

[SAP 設定] ダイアログ・ボックスについて

[SAP 設定] ダイアログ・ボックスでは、メディアータ・マシンを定義し、監視対象サーバの詳細を入力し、メディアータが診断データを収集できることが確認できます。

[SAP 診断を有効にする]: このオプションを選択すると SAP 診断が有効になり、SAP 診断を設定できます。

[メディアータ]

- ▶ [名前]: SAP 診断データの収集と処理に使用するメディアータ・マシンの名前を入力します。各診断モジュールに対してサポートされるメディアータ・マシンは1台だけです。

注: ファイアウォール越しにメディアータを使用する場合は、メディアータ・マシン名ではなく、メディアータのローカル・マシン・キーを入力してください。

- ▶ [ファイアウォールを有効にする]: メディアータがファイアウォール越しかどうかを選択します。
- ▶ [MI リスナー]: ファイアウォール越しに監視する場合は、Mercury リスナー・マシンの名前、フルネーム、または IP アドレスを入力します。

[プロパティ]

▶ SAP サーバのプロパティ

- ▶ [アプリケーション サーバ]: SAP サーバの名前。
- ▶ [ルータ文字列]: SAP サーバのシステム・ルータ文字列を入力します (オプション)。
- ▶ [システム番号]: SAP サーバのシステム番号を入力します。

- ▶ **クライアントのプロパティ**：メディエータはこれらの詳細を使用して、SAP サーバに接続し、診断データを収集します。
 - ▶ **[ユーザ名]**：ユーザが SAP サーバにログオンするための一意の名前。
 - ▶ **[パスワード]**：ユーザが SAP サーバにログオンするためのパスワード。
 - ▶ **[クライアント番号]**：選択されたユーザのクライアント番号。
- [検証]**：これをクリックすると、SAP サーバへの接続を確認できます。

第6部

シナリオの監視

第 24 章

オンライン監視

LoadRunner オンライン・モニタを使用して、シナリオの実行を監視できます。

本章では、オンライン・モニタのユーザ・インタフェースについて説明します。それぞれのモニタについては、『**Mercury LoadRunner モニタ・リファレンス**』で解説します。

本章では、次について説明します。

- ▶ オンライン監視について
- ▶ 監視環境の設定
- ▶ モニタの種類
- ▶ コントローラでのモニタおよび測定値の選択
- ▶ コントローラでのモニタの起動
- ▶ コントローラでオンライン・モニタ・グラフを開く
- ▶ オンライン・モニタ表示ビューのカスタマイズ
- ▶ モニタ・オプションの設定

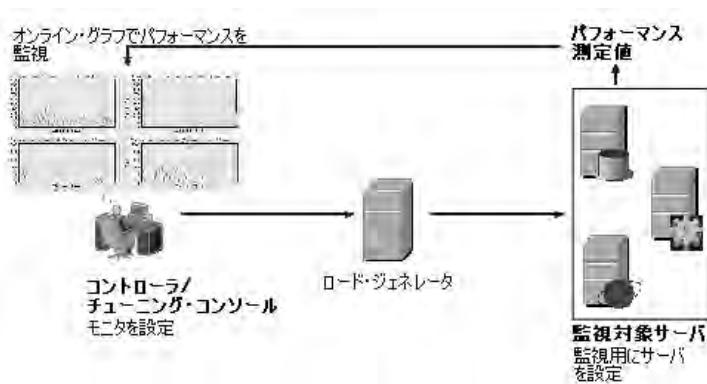
オンライン監視について

LoadRunner オンライン・モニタを使用して、シナリオの実行中に生成されたデータを表示できます。コントローラがシナリオの実行時に監視し、LoadRunner オンライン・グラフを使ってモニタが収集したデータを表示するマシンを指定します。

トランザクションの応答時間に影響を与える最大の要因は、リソースの使用状況です。シナリオ実行中にリソースを監視することにより、特定のマシンでボトルネックが発生した原因を突き止められます。LoadRunnerのサーバ・リソース・モニタでは、シナリオの実行時に使用されるリソースを追跡できます。LoadRunnerは、テストの実行時に、選択されたリソース・モニタをリアルタイムで表示します。監視の対象にするサーバ・リソース測定値は、シナリオ実行の前でも最中でも選択できます。

監視環境の設定

シナリオを監視するには、あらかじめLoadRunner監視コンポーネントを設定しておく必要があります。各モニタの構成要件は異なります。これについては、『Mercury LoadRunner モニタ・リファレンス』の各モニタの章で説明します。次の図は、LoadRunnerの監視のプロセスを示しています。



サーバを監視する前に、次のステップを実行します。

- ▶ サーバ・マシンでの監視環境の設定（必要な場合）
- ▶ コントローラまたはコンソール・マシンでのモニタの設定

サーバ・マシンでの監視環境の設定

次のモニタを使用するには、まずサーバ・マシン上に監視コンポーネントをインストールまたは設定する必要があります。

- COM+
- Citrix
- DB2
- IBM WebSphere MQ
- iPlanet (NAS)
- J2EE
- J2EE & .NET 診断
- ネットワーク遅延時間
- Oracle
- PeopleSoft (Tuxedo)
- SAP CCMS
- SAP Portal
- SAPGUI
- Siebel Server Manager
- Siebel Web サーバ
- SiteScope
- Sybase
- Tuxedo
- UNIX
- WebLogic (JMX)
- WebSphere (EPM)
- WebSphere
- WebSphere アプリケーション・サーバ

コントローラ・マシンでのモニタの設定

パフォーマンス・データを取得するために、(コントローラから) モニタを有効にして、監視対象の統計値と測定値を指定する必要があります。カウンタは、モニタの [測定値の追加] ダイアログ・ボックスで選択します。

SiteScope モニタを設定する場合は、リモート・マシンを設定する必要があります。詳細については、350 ページ「SiteScope モニタ用のリモート・マシンの設定」を参照してください。

監視環境およびモニタの設定の詳細については、『Mercury LoadRunner モニタ・リファレンス』の各モニタの章を参照してください。

モニタの種類

オンライン・モニタは、次のカテゴリに分類されます。

- ▶ 実行時間モニタは、シナリオに参加している仮想ユーザ数とステータスのほか、仮想ユーザが生成するエラーの数やタイプを示します。詳細について

は、『**Mercury LoadRunner モニタ・リファレンス**』の「実行環境とトランザクションの監視」を参照してください。

- ▶ トランザクション・モニタは、シナリオ実行時のトランザクションの速度と応答時間を示します。詳細については、『**Mercury LoadRunner モニタ・リファレンス**』の「実行環境とトランザクションの監視」を参照してください。
- ▶ Web リソース・モニタでは、シナリオ実行時の Web サーバにおける Web 接続数、スループット量、HTTP 応答数、サーバの再送回数、ダウンロードされたページ数についての情報が提供されます。詳細については、『**Mercury LoadRunner モニタ・リファレンス**』の「Web リソースの監視」の項を参照してください。
- ▶ システム・リソース・モニタは、シナリオ実行時に使用された Windows, UNIX, Tuxedo, SNMP, Antara FlameThrower, および SiteScope のリソースを測定します。詳細については、『**Mercury LoadRunner モニタ・リファレンス**』の「システム・リソースの監視」の項を参照してください。
- ▶ ネットワーク遅延時間モニタは、システムのネットワーク遅延に関する情報を示します。詳細については、『**Mercury LoadRunner モニタ・リファレンス**』の「ネットワークの監視」の項を参照してください。
- ▶ ファイアウォール・モニタは、シナリオ実行時にファイアウォール・サーバの統計値を測定します。詳細については、『**Mercury LoadRunner モニタ・リファレンス**』の「ファイアウォール・サーバ・パフォーマンスの監視」の項を参照してください。
- ▶ Web サーバ・リソース・モニタは、シナリオ実行時に、Apache, Microsoft IIS, iPlanet (SNMP), iPlanet/Netscape Web サーバの統計値を測定します。詳細については、『**Mercury LoadRunner モニタ・リファレンス**』の「Web リソースの監視」の項を参照してください。
- ▶ Web アプリケーション・サーバ・リソース・モニタは、シナリオの実行時に、Ariba, ATG Dynamo, BroadVision, ColdFusion, Fujitsu INTERSTAGE, iPlanet (NAS), Microsoft ASP, Oracle9iAS HTTP, SilverStream, WebLogic (SNMP), WebLogic (JMX), および WebSphere アプリケーション・サーバの統計値を測定します。詳細については、『**Mercury LoadRunner モニタ・リファレンス**』の「Web アプリケーション・サーバ・リソースの監視」の項を参照してください。
- ▶ データベース・サーバ・リソース・モニタは、シナリオの実行時に、SQL サーバ, Oracle, Sybase, DB2 データベースの統計値を測定します。詳細に

については、『Mercury LoadRunner モニタ・リファレンス』の「データベース・リソースの監視」の項を参照してください。

- ▶ ストリーミング・メディア・モニタは、シナリオの実行時に、Windows Media サーバと RealPlayer オーディオ・サーバまたはビデオ・サーバのほか、RealPlayer クライアントの統計値を測定します。詳細については、『Mercury LoadRunner モニタ・リファレンス』の「ストリーミング・メディアの監視」の項を参照してください。
- ▶ ERP/CRM サーバ・リソース・モニタは、シナリオの実行時に、SAP Portal, SAP CCMS, SAPGUI, Siebel Server Manager, Siebel Web Server, および PeopleSoft (Tuxedo) サーバの統計値を測定します。詳細については、『Mercury LoadRunner モニタ・リファレンス』の「ERP/CRM サーバ・リソースの監視」の項を参照してください。
- ▶ Java パフォーマンス・モニタは、J2EE マシンを使用している Java 2 Platform, Enterprise Edition (J2EE) オブジェクト、および Java ベースのアプリケーションの統計値を測定します。詳細については、『Mercury LoadRunner モニタ・リファレンス』の「Java パフォーマンスの監視」を参照してください。
- ▶ J2EE/.NET 診断モニタは、J2EE/.NET Web, アプリケーション、データベース・サーバを利用する個々のトランザクションを追跡し、時間を測定し、トラブルシューティングを行うための情報を提供します。詳細については、『Mercury Diagnostics for J2EE & .NET Installation and User's Guide』（英語版）を参照してください。
- ▶ アプリケーション・コンポーネント・モニタは、シナリオの実行時に、Microsoft COM+ サーバの統計値を測定します。詳細については、『Mercury LoadRunner モニタ・リファレンス』の「アプリケーション・コンポーネントの監視」の項を参照してください。
- ▶ Application Deployment ソリューション・モニタは、シナリオの実行時に、Citrix MetaFrame XP サーバおよび Citrix MetaFrame 1.8 サーバの統計値を測定します。詳細については、『Mercury LoadRunner モニタ・リファレンス』の「Application Deployment ソリューション」の項を参照してください。
- ▶ ミドルウェアのパフォーマンス・モニタは、シナリオの実行時に、Tuxedo および IBM WebSphere MQ サーバの統計値を測定します。詳細については、『Mercury LoadRunner モニタ・リファレンス』の「ミドルウェアのパフォーマンス監視」の項を参照してください。

- ▶ インフラストラクチャ・リソース・モニタは、シナリオの実行時に、ネットワーク・クライアントのデータ・ポイントの統計値を測定します。詳細については、『Mercury LoadRunner モニタ・リファレンス』の「インフラストラクチャ・リソースの監視」の項を参照してください。

グラフの設定と測定値の設定およびグラフ・データのエクスポートについては、第 25 章「オンライン・グラフの設定」を参照してください。

どのモニタでも、シナリオの実行後に、収集したデータのサマリを表示できます。LoadRunner アナリシスを使って、任意のモニタのグラフを生成できます。詳細については、『Mercury LoadRunner アナリシス・ユーザーズ・ガイド』を参照してください。

注：アプリケーション・コンポーネント・モニタは、LoadRunner コントローラでのみ使用可能です。アプリケーション・トラフィック・モニタおよびアプリケーション・セキュリティ・モニタは、Mercury チューニング・コンソールでのみ使用可能です。

LoadRunner モニタの詳細な一覧については、Mercury の Web サイト (<http://www.mercury.com/us/products/performance-center/loadrunner/monitors/>) を参照してください。

コントローラでのモニタおよび測定値の選択

コントローラがシナリオの実行時に監視対象サーバ・マシンのダイアログ・ボックスを使用して監視するマシンと測定値を指定します。

マシンのリソースを監視するには、次の手順を実行します。

- 1 監視するモニタをグラフ表示領域で開き、[モニタ] > [測定値の追加] を選択するか、グラフを右クリックして [測定値の追加] を選択します。監視対象サーバ・マシンのダイアログ・ボックスが開きます。



- 2 LoadRunner の標準装備モニタもありますが、SiteScope モニタ・エンジン経由で監視することもできます。

SiteScope モニタ・エンジンを経由してサーバを監視する場合は、**[詳細]** をクリックしてください。[モニタ エンジンの選択] ダイアログ・ボックスが開きます。

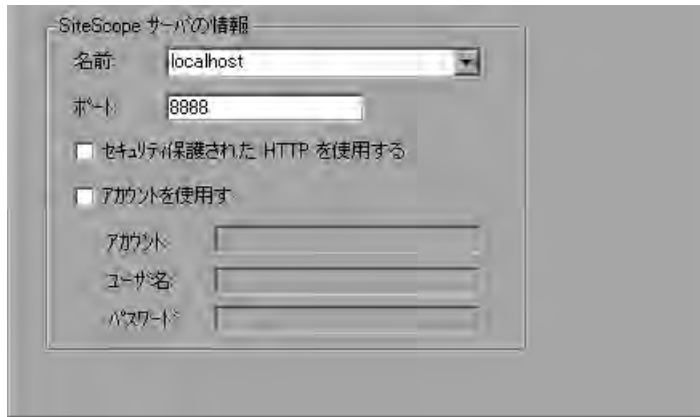
[SiteScope] を選択し、**[OK]** をクリックします。

詳細については、関連する監視の項を参照してください。

- 3 **[監視するサーバマシン]** セクションで **[追加]** をクリックします。[マシンの追加] ダイアログ・ボックスが開きます。



サーバの監視に SiteScope を使用するモニタについては、[マシンの追加] ダイアログ・ボックスに [SiteScope サーバの情報] セクションも表示されます。



- 4 監視対象マシンのサーバ名または IP アドレスを入力し、マシンを実行するプラットフォームを選択します。

SiteScope モニタの場合は、SiteScope サーバの名前とポート番号を入力して、Secure HTTP 接続を使用するかどうか指定します。アカウントを使用する場合は、該当するアカウント情報を入力します。詳細については、350 ページ「[マシンの追加] ダイアログ・ボックスについて」を参照してください。

[OK] をクリックします。

監視対象のサーバ・マシンのダイアログ・ボックスの [監視するサーバマシン] ボックスに、マシンが表示されます。



- 5 監視するサーバ・マシンを選択して，[リソース測定場所 : <マシン>] ボックスで [追加] をクリックします。

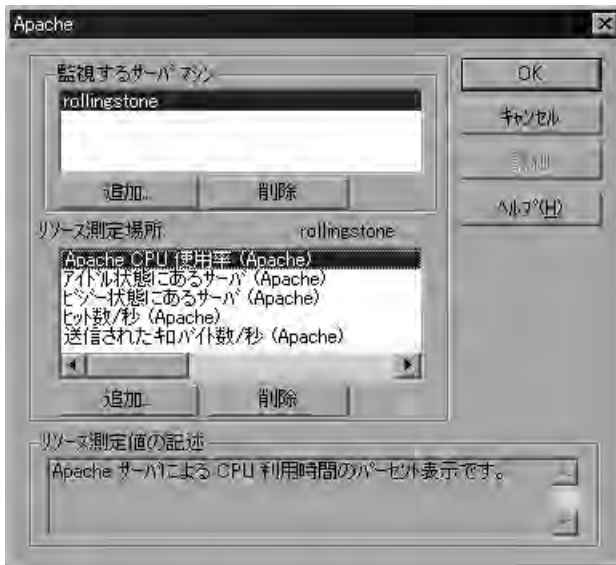
監視対象のサーバの [測定値の追加] ダイアログ・ボックスが開き，利用可能な測定値が表示されます。



注：

- ▶ [測定値の追加] ダイアログ・ボックスは、モニタごとに異なります。モニタに固有の測定値の追加方法については、関連するモニタの項を参照してください。
- ▶ SiteScope モニタを設定する場合は リモート・マシンを設定する必要があります。詳細については、350 ページ「SiteScope モニタ用のリモート・マシンの設定」を参照してください。
- ▶ モニタによっては、測定値を追加する前に設定が必要なものがあります。設定方法については、関連するモニタの項を参照してください。

- 6 必要な測定値を選択します。Ctrl キーを使って複数の測定値を選択できます。
- 7 [OK] をクリックします。[測定値の追加] ウィンドウが閉じ、選択された測定値が監視対象サーバ・マシンのダイアログ・ボックスの [リソース測定場所：<マシン>] ボックスに表示されます。



- 8 監視対象のサーバ・マシンのダイアログ・ボックスで [OK] をクリックすると、モニタが起動します。

監視するサーバマシンのダイアログ・ボックスについて

[監視するサーバマシン]：リソース監視の対象となるマシン。

- ▶ [追加]：[マシンの追加] ダイアログ・ボックスが表示され、監視対象にするマシンを既存のリストに追加できます。
- ▶ [削除]：選択したマシンをリストから削除します。

注：場合によっては、他の形式を使って監視対象のサーバを指定できます（または指定する必要があります）。詳細については、『**Mercury LoadRunner モニタ・リファレンス**』の関連する監視の項を参照してください。

[リソース測定場所：<マシン名>]：選択したマシン上で監視するリソース測定値が表示されます。

- ▶ [追加]：選択したマシン上の監視の対象リソースのリストを作成するためのダイアログ・ボックスを開きます。
- ▶ [削除]：選択したリソース測定値をリストから削除します。

[詳細]：[モニタ エンジンの選択] ダイアログ・ボックスが開き、ネイティブな LoadRunner または SiteScope での監視を選択できます。

- ▶ LoadRunner でサーバを監視するには、[**LoadRunner native monitors**] を選択します。
- ▶ SiteScope でサーバを監視するには、[**SiteScope**] を選択します。

注：このボタンは、モニタが SiteScope モニタまたは LoadRunner の標準装備モニタのいずれかの場合にのみ有効です。

[詳細]：選択したリソース測定値の説明が表示されます。

注：測定値を設定する前に特定のサーバ・モニタを設定する方法については、『**Mercury LoadRunner モニタ・リファレンス**』の対応する項を参照してください。

[マシンの追加] ダイアログ・ボックスについて

監視するマシンを [監視するサーバマシン] リストに追加します。

[監視するマシンの情報]：

- ▶ **[名前]**：監視するマシンの名前または IP アドレスを入力します。
- ▶ **[プラットフォーム]**：監視するマシンのプラットフォームを入力します。

[SiteScope サーバの情報]：

SiteScope を使用するモニタに、次の SiteScope サーバ情報を入力します。

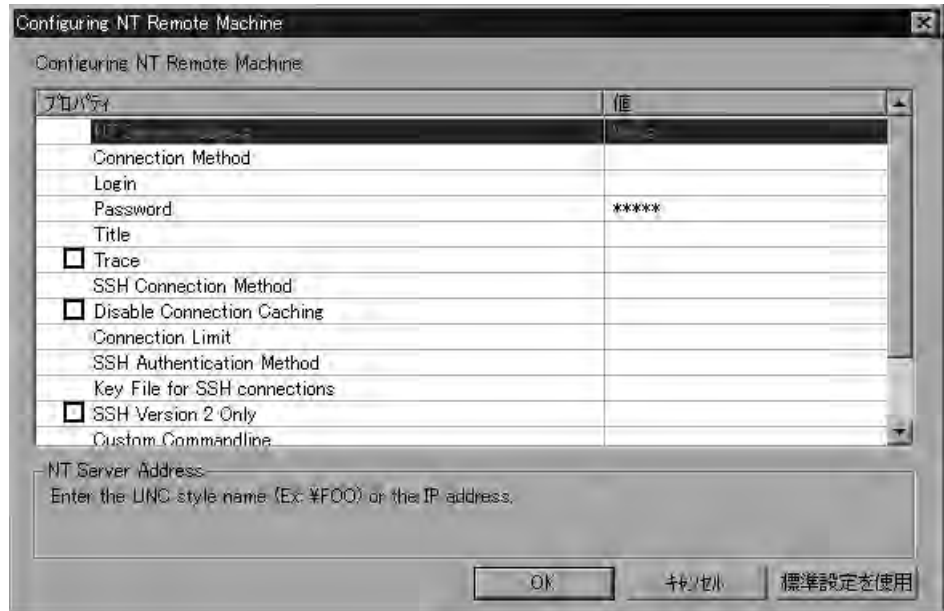
- ▶ **[名前]**：SiteScope サーバの名前を入力します。
- ▶ **[ポート]**：SiteScope ポートを入力します（標準設定：8888）。
- ▶ **[セキュリティ保護された HTTP を使用する]**：Secure HTTP 接続を使用する場合に選択します。
- ▶ **[アカウントを使用する]**：特定の SiteScope ユーザ・アカウントを使用する場合にこのオプションを選択します。アカウントの次の詳細を入力します。
 - **[アカウント]**：SiteScope アカウントの名前と番号。
 - **[ユーザ名]**：SiteScope アカウントへのログイン用に定義されているユーザ名。
 - **[パスワード]**：SiteScope アカウントへのログイン用に定義されているパスワード。

SiteScope モニタ用のリモート・マシンの設定

マシンを実行するプラットフォームに応じて、リモート・マシンを設定します。

NT リモート・マシンの設定

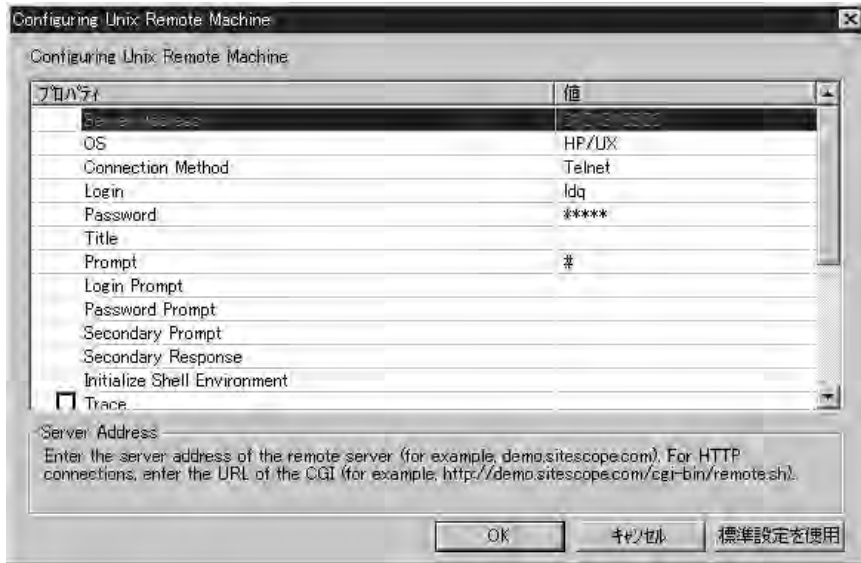
NT プラットフォーム・マシンの場合は、最初に監視対象の測定値を追加するときに [Configuring NT Remote Machine] ダイアログ・ボックスが開きます。



352 ページ 「[Configuring NT Remote Machine] ダイアログ・ボックスについて」の説明にしたがって、リモート・マシンの設定を選択します。

UNIX リモート・マシンの設定

UNIX プラットフォーム・マシンの場合は、最初に監視対象の測定値を追加するときに [Configuring Unix Remote Machine] ダイアログ・ボックスが開きます。



355 ページ「[Configuring Unix Remote Machine] ダイアログ・ボックスについて」の説明にしたがって、リモート・マシンの設定を選択します。

[Configuring NT Remote Machine] ダイアログ・ボックスについて

[Configuring NT Remote Machine] ダイアログ・ボックスからリモート NT マシンを設定します。

[NT Server Address] : 監視する NT サーバの IP アドレスか UNC スタイル名。SiteScope サーバがホスト名を IP アドレスに解決することができれば (例えば hosts ファイル, DNS および WINS/DNS 統合などによって), IP ホスト名も使用することができます。

[Connection Method] : SiteScope は, リモート NT サーバのリソース監視で 2 種類の接続タイプより 1 つを選択して使用することができます。要素クラスは, 次の 2 つです。

- ▶ **NetBIOS** : Windows NT および 2000 ネットワーク用の標準設定のサーバ間通信プロトコル。

- ▶ **SSH** : セキュアシェル。Windows NT/2000 ベースのネットワークにインストール可能な、よりセキュアなプロトコル。この接続方法を使用するには、通常接続したいサーバに SSH ライブラリをインストールすることが求められます。

[**Login**] : リモート・サーバのログイン名を挿入します。サーバが SiteScope マシンと同一ドメイン内にある場合は、ユーザ・ログイン名の前にドメイン名を入れます。次に例を示します。

domainname¥user

ドメイン内およびドメイン外のマシンに対してローカル・マシン・ログイン・アカウントを使用する場合は、ユーザ・ログイン名の前にマシン名を入れます。次に例を示します。

machinename¥user

[**Password**] : リモート・サーバのパスワードまたは SSH 鍵ファイルのパスフレーズ。

注 : SSH 認証で公開 / 秘密鍵ベースの認証を行う場合は、identity ファイルのパスフレーズをここに入力します。

[**Title**] : (オプション) リモート・マシンを識別する名前。この名前はドロップダウン・リストに表示されます。

[**Trace**] : このチェック・ボックスをチェックすると、対象のサーバとの間でやり取りされる追跡メッセージが、SiteScope の RunMonitor.log ファイルに記録されます。

[**SSH Connection Method**] : この接続に使用するメソッド。現在サポートしているメソッドは次のとおりです。

- ▶ **Java 内部ライブラリ** : SiteScope に統合された Java SSH クライアントによる接続。
- ▶ **Plink** : 外部 SSH クライアントによる接続。Windows NT では、Plink は SiteScope に付属しています。Unix または Linux SiteScope では、インストールされたクライアントを OpenSSH として使用します。

[**Disable Connection Caching**] : このオプションをチェックすると、このリモートの接続キャッシュが無効になります。標準設定では、SiteScope キャッシュは接続を開きます。

[**Connection Limit**] : SiteScope がこのリモートに対して許可する、接続を開く数を制御します。この接続を使用するように設定されたモニタが多数ある場合は、潜在的なボトルネックが避けられるように、この数値を高く設定します。

注 : この設定は、リモートに対するテストの実行には影響しません。テストでは、常に新しい接続が作成されます。

[**SSH Authentication Method**] : SSH 接続で使用する認証メソッド。現在サポートしているメソッドは次のとおりです。

- ▶ [**Password**] : パスワードによる認証。
- ▶ [**Key File**] : 公開 / 秘密鍵による認証。このオプションを選択すると、SiteScope は認証に **SiteScope/groups/identity/** ファイル内の秘密鍵を使用します。対応する公開鍵は、リモート・ホストの **authorized_keys** ファイルに書き込まれている必要があります。

[**Key File for SSH connections**] : この接続の秘密鍵が含まれるファイルを選択します。標準の鍵ファイルは **SiteScope%groups%identity** です。この設定は、認証方法が **Key File** の場合にのみ適用されます。

[**SSH Version 2 Only**] : このオプションをチェックすると、SiteScope で SSH プロトコルのバージョン 2 のみが使用されるようになります。このオプションは、SiteScope に統合された Java クライアントを使用する場合にのみ適用されます。

[**Custom Commandline**] : 外部クライアントを使用するリモートに対してカスタム・コマンド・ラインを入力します。このオプションは、実行している外部クライアントに特定のオプションを渡す必要がある場合に使用します。有効な代入変数は次のとおりです。

- ▶ **\$root\$** : SiteScope ディレクトリに変換されます。
- ▶ **\$user\$** : リモートに入力されるユーザ名に変換されます。
- ▶ **\$password\$** : リモートに入力されるパスワードに変換されます。
- ▶ **\$host\$** : リモートに入力されるホスト名に変換されます。

[**SSH Port Number**] : リモート SSH サーバがリッスンするポートを入力します。標準のポート番号は 22 です。

[Configuring Unix Remote Machine] ダイアログ・ボックスについて

[Configuring Unix Remote Machine] ダイアログ・ボックスから Unix リモート・マシンを設定します。

[**Server Address**] : [マシンの追加] ダイアログ・ボックスで入力したサーバの IP アドレスまたはホスト名が表示されます。

[**OS**] : リモート・サーバで実行するオペレーティング・システムを選択します。次のバージョンの UNIX がサポートされています。

AIX	OPENSERVER
FreeBSD	SCO
HP/UX	SGI Irix
HP/UX 64-bit	Sun Solaris
Linux	Tru64 5.x
MacOSX	Tru64 Pre 4.x (Digital)

[**Connection Method**] : サーバへの接続方法を選択します。サポートしている方法は次のとおりです。

- ▶ **Telnet** : Telnet を使ってリモート・サーバにログインします。
- ▶ **SSH** : SSH プロトコルを使用してリモート・サーバにログインします。UNIX のバージョンによっては、追加のソフトウェアおよび設定が必要となる場合もあります。
- ▶ **Rlogin** : Rlogin プロトコルを使用してリモート・サーバにログインします。
- ▶ **HTTP** : リモート・サーバ上の HTTP サーバに接続し、CGI を通してコマンドを実行します。この方法の場合、ログインおよびパスワードは任意であり、必要に応じてリモート・マシンへのログオン認証に使用されます。

[**Login**] : リモート・サーバのログイン名を挿入します。

[**Password**] : リモート・サーバのパスワードを入力します。

[**Title**] : リモート・マシンを識別する名前を入力します。この名前は、このサーバに接続可能なモニタのドロップダウン・リストに表示されます。

[**Prompt**] : システムがコマンドを処理する準備ができたときに表示するプロンプトを入力します。標準設定は「#」です。

[**Login Prompt**] : システムがログインの入力を待っているときに表示するプロンプトを入力します。標準設定は「ogin:」です。

[**Password Prompt**] : システムがパスワードの入力を待っているときに表示するプロンプトを入力します。標準設定は「assword:」です。

[**Secondary Prompt**] : リモート・サーバへの Telnet 接続によって、リモート・サーバが接続に関する詳細情報を求めてきた場合に表示されるプロンプトを入力します。複数のプロンプト文字列はコンマ (,) で区切ります。例えば、リモート・サーバへの Telnet 接続では、そのリモート・サーバが、その接続にどの端末タイプを模倣すべきかを聞いてくる場合があります。この場合は、二次プロンプトとして「Terminal type?」と入力することになるでしょう。2 番目のプロンプトへの応答は、次の [Secondary Response] フィールドに入力されます。

[**Secondary Response**] : このリモート・サーバとの接続を確立するために必要な 2 番目のプロンプトへの応答を入力します。複数の応答を入力する場合はカンマ (,) で区切ります。

[**Initialize Shell Environment**] : セッションの始めに実行するシェル・コマンドを入力します。複数のコマンドを入力する場合は、セミコロン (;) で区切ります。このオプションで、Telnet または SSH セッションの起動直後にリモート・マシンで実行するシェル・コマンドを指定できます。これらのコマンドは、各 SiteScope リモートのシェルのカスタマイズに使用することができます。

例 :

- ▶ リモートシェルは、SiteScope のスクリプトを実行するための正しいパスを持っていない可能性があります。次のコマンドは、リモート・マシンの現在のシェルの PATH に **/usr/local/bin** ディレクトリを追加します。

```
export PATH=$PATH:/usr/local/sbin
```

- ▶ リモートシェルは擬似端末を正しく初期化していない可能性があります。端末の幅を 1024 文字に広げるために次のコマンドを入力します。

```
stty cols 1024;${SHELL}
```

注 : シェル呼び出し後のコマンドは実行されません。

- ▶ リモートの Telnet サーバが、正しいコマンド・ラインのエコーを返さない場合があります。これは、このエコーの動作に依存するモニタでの不可解な動作を引き起こす可能性があります。

リモート端末にエコーを強要するために、次のコマンドを入力します。

```
stty echo
```

- ▶ UNIX シェルの中には、SiteScope に適さない動作を行うものがあります。bash, ksh, csh がそうです。次のコマンドを入力して、SiteScope の接続に使用するシェルを sh に変更します。

```
/bin/sh
```

[**Trace**] : このオプションを選択すると、リモート・サーバとやり取りするメッセージが **RunMonitor.log** ファイルに記録されます。

[**SSH Connection Method**] : リモート・サーバへの接続方法を選択します。

- ▶ [**Internal Java Libraries**] : SiteScope に統合された Java SSH クライアントによる接続。
- ▶ [**Plink**] : 外部 SSH クライアントによる接続。Windows NT では、Plink は SiteScope に付属しています。

[**Disable Connection Caching**] : このオプションを選択すると、SSH 接続キャッシュが無効になります。

[**Connection Limit**] : このリモート・マシンの最大接続数を入力します。

[**SSH Authentication Method**] : リモート・サーバの認証方法を選択します (SSH 接続の場合のみ)。

- ▶ [**Password**] : パスワードによる認証。
- ▶ [**Keyfile**] : 公開 / 秘密鍵による認証。このオプションが選択されている場合、SiteScope は **SiteScope/groups/identity** ファイルの秘密鍵を使って認証します。対応する公開鍵は、リモート・ホストの **authorized_keys** ファイルに書き込まれている必要があります。

[**Connection Limit**] : このリモート・マシンの最大接続数を入力します。

[**SSH Version 2 Only**] : このオプションを選択した場合、SSH は SSH プロトコル・バージョン 2 のみを使用します。このオプションは、内部 Java ライブラリの接続メソッドを使用している場合のみサポートされます。

[**Custom Commandline**] : 外部 SSH クライアントを実行するためのコマンドを入力します。上記のオプションの置換を行う場合、それぞれ \$host\$, \$user\$, \$password\$ を使用します。この設定は、外部プロセスを使う接続でのみサポートされます。

[**SSH Port Number**] : SSH サービスを実行中のポートを入力します。

SiteScope 用のサーバを定義した後、テスト・リンクをクリックすれば設定をテストできます。SiteScope は、リモート・マシンがアクセス可能で、コマンドを正しく実行できることを確認するためのテストとして、リモート・マシンの作業ディレクトリの表示を試みます (UNIX では「pwd」コマンド、Windows NT では「cd」コマンド)。

コントローラでのモニタの起動

オンライン・モニタを起動するには、次の手順を実行します。

- 1 シナリオを開始します。実行する仮想ユーザ・グループを選択して、**[シナリオの開始]** を選択するか、**[シナリオ]** > **[開始]** を選択します。
- 2 **[実行]** タブを選択します。**[シナリオ グループ]** ウィンドウの下に、標準設定のグラフが表示されます。





- 3 グラフを最大化するには、グラフをダブルクリックします。もう 1 度ダブルクリックすると、並べて表示された状態に戻ります。
- 4 グラフ・ツリーが表示されない場合は、[表示] > [利用可能なグラフの表示] を選択します。グラフ・ツリーを展開するには、左の表示枠で [+] をクリックします。グラフ・ツリー・ビューを非表示にするには、[表示] > [利用可能なグラフの非表示] を選択するか、[利用可能なグラフ] リストの右上隅にある [×] ボタンをクリックします。
- 5 ツリーからグラフを選択して、右の表示枠までドラッグします。表示枠の間でグラフをドラッグすることもできます。

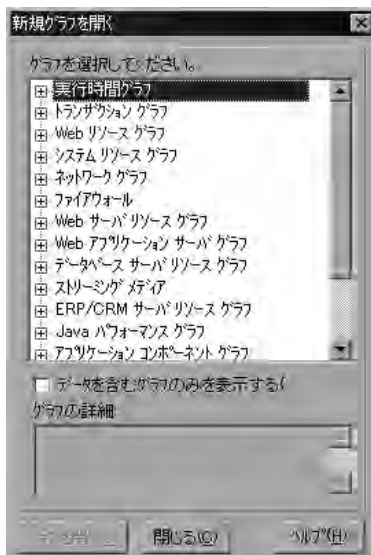
注：トランザクションが実行されていない場合、トランザクション・モニタ・グラフにはデータが表示されません。また、シナリオを実行する前に監視対象のリソースのリストを作成していないと、他のグラフにはデータが表示されません。

コントローラでオンライン・モニタ・グラフを開く

標準設定では、LoadRunnerによって[実行]ビューに次の4つのグラフが表示されます。実行中の仮想ユーザ、トランザクション応答時間、秒ごとのヒット数、Windowsリソース。他のグラフを表示するには、グラフ・ツリーからグラフ表示領域までグラフをクリックしてドラッグします。または、[新規グラフを開く]ダイアログ・ボックスを使って新しいグラフを開きます。

[新規グラフを開く]ダイアログ・ボックスを使って新しいグラフを開くには、次の手順を実行します。

- 1 [モニタ] > [オンライングラフ] > [新規グラフの開始] を選択するか、グラフを右クリックして [新規グラフを開く] を選択します。[新規グラフを開く]ダイアログ・ボックスが開きます。



- 2 **[+]** をクリックしてグラフ・ツリーを展開し、グラフを選択します。**[グラフの詳細]** ボックスにはグラフの説明が表示されます。
- 3 **[グラフを開く]** をクリックするか、セッション・ビューの右側の枠に選択したグラフをドラッグします。グラフ表示領域にグラフが表示されます。

[新規グラフを開く] ダイアログ・ボックスについて

[新規グラフを開く] ダイアログ・ボックスを使用して、新規グラフを開いて、その説明を表示できます。

[グラフを選択してください。]：ツリー・ビューを展開するには、各カテゴリの左側にある「+」をクリックします。グラフを選択します。

注：グラフは一度に1つだけ開くことができます。

[データを含むグラフのみを表示する]：このオプションを選択すると、データのみを含むグラフが表示されます。LoadRunner アナリシス・グラフのリスト全体を表示するには、このオプションをクリアします。

[グラフの詳細]：選択されたグラフの説明を表示します。

[グラフを開く]：選択されたグラフを開き、グラフ・ツリー・ビューに表示します。

オンライン・モニタ表示ビューのカスタマイズ

LoadRunner では、オンライン・モニタ・グラフを同時に 16 個まで表示できます。

オンライン・モニタの表示をカスタマイズするには、次の手順を実行します。

- 1 グラフを右クリックして、**[グラフを表示]** を選択するか、**[表示] > [グラフを表示]** を選択します。
- 2 表示するグラフの数を選択します。**[グラフを 1 つ表示]**、**[グラフを 2 つ表示]**、**[グラフを 4 つ表示]**、**[グラフを 8 つ表示]** または **[グラフを指定数表示]** のいずれかを選択します。

[**グラフを指定数表示**] を選択した場合は、表示するグラフの数を [グラフの表示] ダイアログ・ボックスに入力し、[OK] をクリックします。選択した数のグラフが、グラフ表示領域に表示されます。

グラフを1つだけ表示するには、グラフ表示枠をダブルクリックします。前の表示に戻るには、グラフを再度ダブルクリックします。

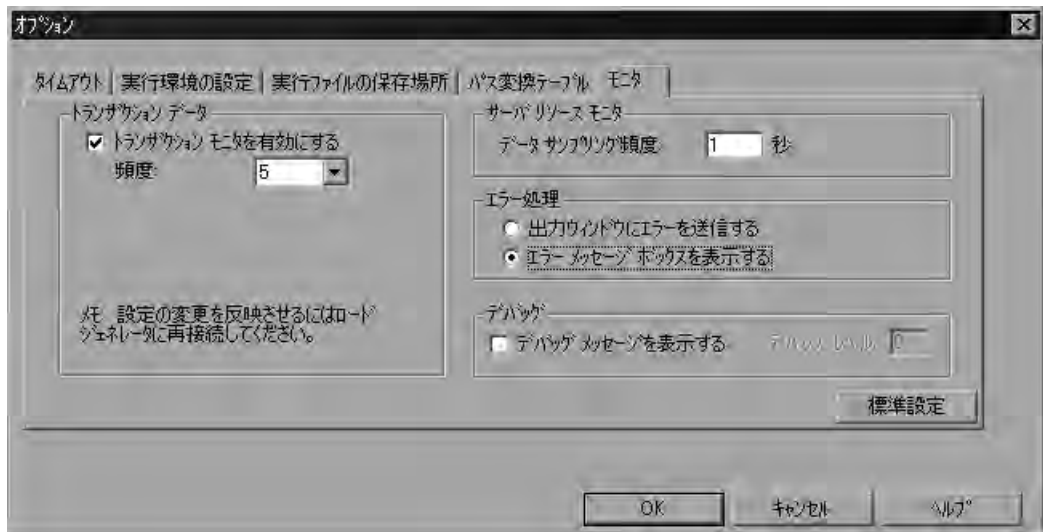
モニタ・オプションの設定

シナリオの実行前に、LoadRunner ではオンライン・モニタの設定ができます。オンライン・モニタについて、データ・サンプリング頻度、エラー処理、デバッグ、頻度設定を指定できます。

シナリオを保存すると、オンライン・モニタの設定も保存されます。

モニタ・オプションを設定するには、次の手順を実行します。

- 1 [ツール] > [オプション] を選択して、[モニタ] タブを選択します。



- 2 [**トランザクション モニタを有効にする**] が選択されていることを確認し (標準設定)、モニタからコントローラのトランザクション・グラフ、データ・ポイント・グラフ、Web リソース・グラフに更新情報を送信する頻度を設定します。

リソースを節約するために、[トランザクション モニタを有効にする] チェック・ボックスをクリアしてトランザクション・モニタを無効にすることもできます。

注：シナリオの実行中、これらの設定は変更できません。モニタを無効にしたり、その頻度を変更したりするには、まずシナリオの実行を停止する必要があります。

- 3 サンプルング頻度を入力します。
- 4 必要な [エラー処理] オプションを設定します。
- 5 [出力メッセージ] ウィンドウにデバッグ・メッセージを表示するには、[デバッグ メッセージを表示する] チェック・ボックスを選択します。ネットワーク・モニタの場合は、[デバッグ レベル] の値を1～9の間で指定します。
- 6 [OK] をクリックして設定を保存し、[オプション] ダイアログ・ボックスを閉じます。

エキスパート・モードでは、追加のモニタ設定が行えます。エキスパート・モードでの作業の詳細については、付録C「エキスパート・モードでの作業」を参照してください。

[オプション - モニタ] タブについて

[モニタ] タブでは、トランザクション・モニタを有効にし、トランザクション・データの振る舞いを設定して、オンライン・モニタについてデータ・サンプルング頻度、エラー処理、デバッグ、頻度設定を指定できます。

[トランザクション データ]：トランザクション、データ・ポイント、Web リソースの各オンライン・グラフについて、データの振る舞いを設定します。

- ▶ [トランザクション モニタを有効にする]：オンライン仮想ユーザ・トランザクション・モニタを有効にして、シナリオの開始時にトランザクションの監視が開始されるようにします。
- ▶ [頻度]：トランザクション、データ・ポイント、Web リソースの各オンライン・グラフを生成するために、オンライン・モニタがデータをサンプルングする頻度（単位は秒）を選択します。標準設定は5秒です。小さいシナリオの場合は、頻度を1秒に設定することを推奨します。大きいシナリオでは、3～5秒に設定することを推奨します。頻度の値を大きくすると、ネッ

トワーク通信量が少なくなります。指定された間隔の間に発生したデータの平均値が算出され、単独の値としてコントローラに送信されます。

トランザクション・モニタおよび Web ページ・ブレイクダウンの有効化と無効化の詳細については、『**Mercury LoadRunner モニタ・リファレンス**』の「実行環境とトランザクションの監視」を参照してください。

[**サーバリソース モニタ**]：サーバ・リソース・モニタの振る舞いを設定します。

- ▶ [**データ サンプリング頻度**]：サンプリング頻度とは、連続する 2 件のサンプリングの間隔（秒単位）です。監視データ用に LoadRunner がシナリオのサンプリングを行う頻度を入力します。標準設定では、オンライン・モニタは 3 秒の間隔でデータをサンプリングします。サンプリング頻度の値を大きくすると、データを監視する頻度が少なくなります。この設定は、すべてのグラフに適用されます。特定のグラフのサンプリング頻度を設定する方法については、369 ページ「グラフのプロパティの設定」を参照してください。

注：設定したデータ・サンプリング頻度は、その後起動されるすべてのサーバ・モニタに適用されます。ただし、すでに実行中のサーバ・モニタには適用されません。新しいデータ・サンプリング頻度を実行中のサーバ・モニタにも適用するには、シナリオを保存して再度開きます。各モニタの最小サンプリング頻度は異なります。標準のサンプリング頻度または [オプション] > [モニタ] タブで設定された頻度がモニタの最小サンプリング頻度よりも小さい場合、モニタは自身の最小サンプリング頻度でデータをサンプリングします。例えば、Oracle モニタの最小サンプリング頻度は 10 秒です。[オプション] > [モニタ] タブのサンプリング頻度を 10 秒未満に設定しても、Oracle モニタは 10 秒間隔でデータを監視し続けます。

[**エラー処理**]：LoadRunner のエラー・メッセージの発行方法を制御します。次のいずれかのオプションを選択します。

- ▶ [**出力ウィンドウにエラーを送信する**]：すべてのエラーを出力ウィンドウに送信します。
- ▶ [**エラーメッセージボックスを表示する**]：エラーをメッセージ・ボックスに送信します（標準設定）。メッセージ・ボックスを閉じるには、[OK] をクリックします。

[**デバッグ**] : シナリオをデバッグするために、次のオプションを設定できます。

[**デバッグ メッセージを表示する**] : デバッグに関連したメッセージを出力ログに送信します。1 から 9 までのデバッグ・レベルも指定できます。デバッグ・レベルを指定できるのは、ネットワーク・モニタの場合だけです。

第 25 章

オンライン・グラフの設定

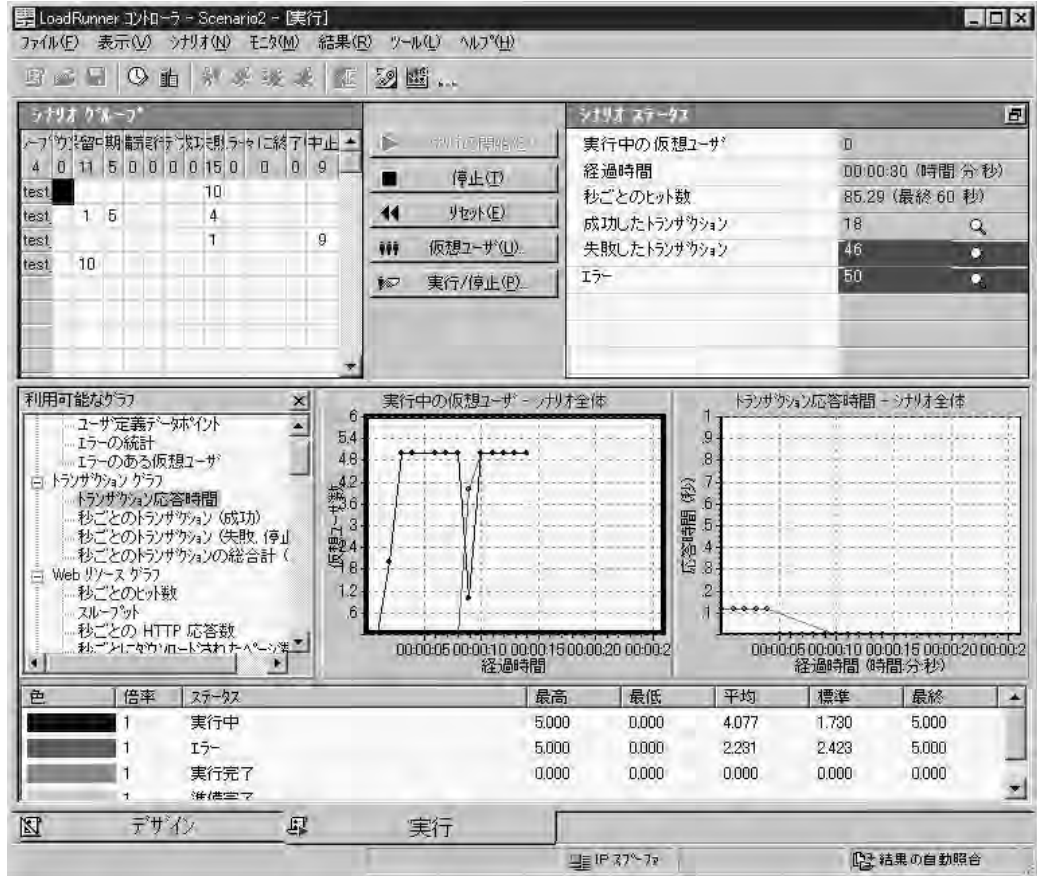
LoadRunner オンライン・モニタ・グラフを使用して、モニタによって収集されたデータを表示できます。

- ▶ オンライン・モニタ・グラフについて
- ▶ グラフのプロパティの設定
- ▶ グラフの測定値の構成
- ▶ グラフの結合
- ▶ オンライン・モニタ・グラフのエクスポート
- ▶ オフラインでのデータ表示
- ▶ 利用可能なグラフ・ツリー

オンライン・モニタ・グラフについて

オンライン・モニタ・グラフには、シナリオの実行中に監視されているリソースのパフォーマンス測定値が表示されます。各測定値は、色分けされてグラフに表示されます。測定値に関する情報は、グラフの下の凡例に表示されます。凡例には、選択したグラフの測定値だけが表示されます。

モニタ・グラフの開き方や表示のカスタマイズについては、360 ページ「コントローラでオンライン・モニタ・グラフを開く」と 362 ページ「モニタ・オプションの設定」を参照してください。



測定値についての追加情報を取得するには、その測定値を右クリックして「**詳細**」を選択します。

特定の線を選択して、次のことができます。

- ▶ **測定値の強調表示**：特定の測定値を強調表示するには、凡例でその測定値を選択します。対応するグラフの線が太線で表示されます。
- ▶ **測定値の非表示**：測定値を非表示にするには、その測定値を右クリックして「**非表示**」を選択します。選択した測定値以外のすべての測定値を非表示にするには、その測定値を右クリックして「**選択済みのみ表示**」を選択しま

す。非表示の測定値を表示するには、その測定値を右クリックして **[表示]** を選択します。

- ▶ **モニタの一時停止**：シナリオの実行中に特定のグラフを一時停止するには、グラフを選択して **[モニタ]** > **[オンライン グラフ]** > **[固定]** を選択するか、グラフを右クリックして **[固定]** を選択します。再開するには、上記の操作を繰り返します。再開すると、一時停止していた時間のデータがグラフに表示されます。

凡例が更新された後の並べ替え順を保持するには、グラフを右クリックして、**[凡例を順序別に表示]** を選択します。並べ替え順を削除するには再度クリックします。凡例は 5 分ごとに更新されます。

グラフのプロパティの設定

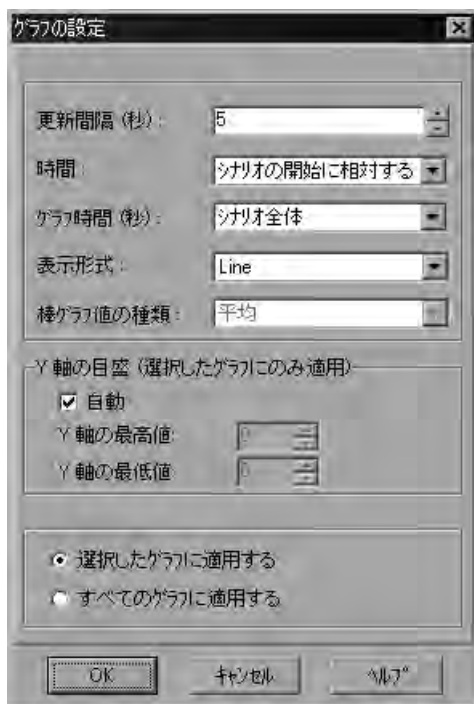
LoadRunner では、オンライン・モニタ・グラフの設定ができます。次の領域に関して、グラフをカスタマイズできます。

- ▶ 更新間隔
- ▶ 時間
- ▶ グラフ時間
- ▶ 表示形式
- ▶ 棒グラフ値の種類
- ▶ Y 軸の目盛
- ▶ ネットワーク遅延時間ビュー

注：これらの設定は、すべてのグラフに適用されるようにグローバルに設定することも、グラフごとに設定することもできます。

グラフをカスタマイズするには、次の手順を実行します。

- 1 設定するオンライン・グラフを（左右どちらかの表示枠で）選択し、**[モニタ]** > **[オンライングラフ]** > **[設定]** を選択します。または、グラフを右クリックし **[設定]** を選択します。[グラフの設定] ダイアログ・ボックスが開きます。



- 2 [更新間隔 (秒)] ボックスに、必要な更新頻度 (グラフの更新間隔) を入力します。
- 3 [時間] ボックスから X 軸のスタイルを選択します。
- 4 [グラフ時間 (秒)] ボックスから値を選択します。グラフ作成時間とは、X 軸に表示される時間 (秒単位) です。
- 5 [表示形式] ボックスからグラフのスタイルを選択します。
- 6 選択した表示形式が「Bar」(棒グラフ) の場合、[棒グラフ値の種類] ボックスから値を選択します。これによって、棒グラフに表示される値の種類が決まります。[平均]、[最終値]、[最低]、および[最高] のいずれかを選択できます。

- 7 Y 軸の最小値または最高値を指定します。または、標準設定の目盛りでグラフを表示する場合は、**[自動]** を選択します。
- 8 ダイアログ・ボックスの設定をすべてのグラフに適用するには、**[すべてのグラフに適用する]** を選択します。
- 9 ネットワーク遅延時間グラフの場合は、次のオプションを選択できます。
 - ▶ **[サブパス]** : 送信元マシンからネットワーク・パス上にある各ノードまでの遅延時間の測定値を表示します。
 - ▶ **[DNS 名]** : 凡例の中に測定値の DNS 名を表示します。
- 10 **[OK]** をクリックし、設定を保存して **[グラフの設定]** ダイアログ・ボックスを閉じます。

[グラフの設定] ダイアログ・ボックスについて

[グラフの設定] ダイアログ・ボックスでは、オンライン・グラフの設定をカスタマイズできます。

[更新間隔] : グラフを新しいデータで更新する間隔です。標準設定では、グラフは 5 秒ごとに更新されます。更新率の値を大きくすると、データの更新頻度が少なくなります。

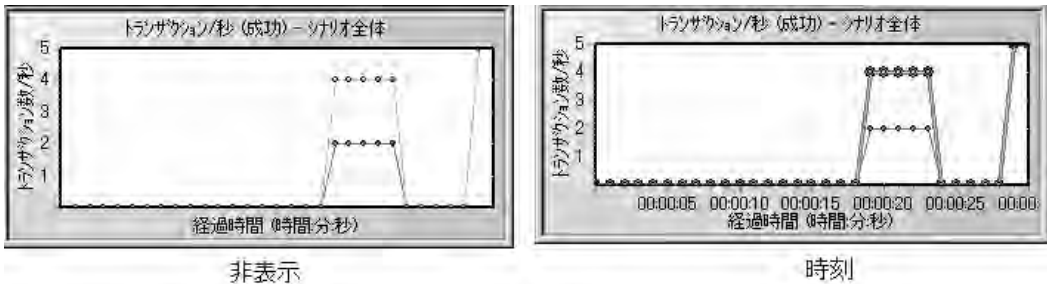
注 : 大きい負荷テストでは、更新率を 3 ~ 5 秒に設定することをお勧めします。これにより、CPU リソースの使用率の問題を防ぐことができます。

[時間] : X 軸の時間をグラフにどのように表示するか、次から選択できます。

- ▶ **[非表示]** : LoadRunner に対して、X 軸の値を表示しないように指示します。
- ▶ **[時刻]** : システム・クロックに基づいた絶対時間が表示されます。
- ▶ **[シナリオの開始に相対する]** : シナリオの開始からの相対時間が表示されます。

注 : ステップの実行中でない場合は、クロック時間が表示されます。

次の例では、[非表示] オプションと [時刻] オプションを選択してグラフを表示しています。



[**グラフ時間**]：グラフの X 軸が時間に基づく場合の X 軸のスケールを指定します。グラフに 60 秒あるいは 3600 秒の動作状況を表示できます。グラフの表示をより詳細にするには、[グラフ時間 (秒)] の値を小さくします。長時間にわたるパフォーマンスの推移を表示するには、[グラフ時間 (秒)] の値を大きくします。指定可能なグラフ作成時間は、次のとおりです。シナリオ全体、60, 180, 600, 3600 秒。

[**表示形式**]：LoadRunner がグラフを線グラフと棒グラフのどちらで表示するかを指示できます。標準設定では、各グラフは線グラフとして表示されます。

注：ネットワーク遅延グラフの場合、[セグメントを表示] を選択すると、グラフのネットワーク・セグメントを円グラフや面グラフとして表示することができます。

[**棒グラフ値の種類**]：[棒グラフ値の種類] ボックスから値を選択します（選択した表示タイプが「Bar」（棒グラフ）の場合）。これによって、棒グラフに表示される値の種類が決まります。[平均]、[最終値]、[最低]、および [最高] のいずれかを選択できます。

[**Y 軸の目盛**]：標準設定の Y 軸の目盛りを使ってグラフを表示するように LoadRunner に指示するか、標準設定とは異なる目盛りを指定できます。標準設定の Y 軸の目盛りを使用するように LoadRunner に指定する場合は、[自動] をクリックします。Y 軸の目盛りを変更するには、Y 軸の最小値または最高値を指定します。

[**ネットワーク遅延時間ビュー**]：このオプションは、ネットワーク遅延時間グラフを設定するときだけに表示されます。[サブパス] をクリックすると、送

信元マシンからネットワーク・パス上にある各ノードまでの遅延時間の測定値を表示できます。[DNS 名] をクリックすると、凡例内の測定値の DNS 名を表示できます。

グラフの測定値の構成

次のオンライン測定の設定ができます。

- ▶ 線の色の変更
- ▶ 測定値の倍率の設定
- ▶ トランザクションの表示と非表示

線の色の変更

測定値には、LoadRunner によってそれぞれ一意の色が割り当てられます。この色は、設定インターフェースを使って変更できます。

測定値の線の色の変更は、次の手順で行います。

- 1 グラフの下の凡例で、設定対象の測定値を選択します。右クリックして **[設定]** を選択します。[測定値の設定] ダイアログ・ボックスが開きます。



- 2 線の色を変更するには、[配色] リストから色を選択します。

- 3 [OK] をクリックして設定を承認し、ダイアログ・ボックスを閉じます。

指定した色の変更は、グラフとグラフの下の凡例に反映されます。色は、凡例の最初のコラムに表示されます。

色	拡大縮小	トランザクション	最高	最低	平均	標準
	10	DOGBER...	0.5	0	0.013158	0.080036
Hidden	10	Mercury_I...	0.5	0	0.013158	0.080036
	10	Mercury_I...	0.5	0	0.013158	0.080036
	10	Mercury_I...	0.5	0	0.045455	0.14374

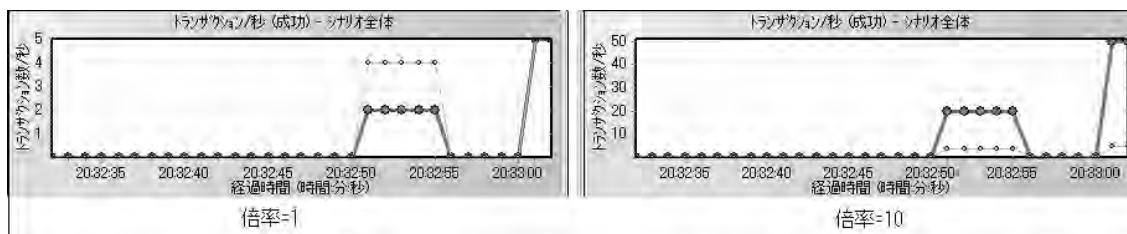
測定値の倍率の設定

測定値の倍率（Y 軸とグラフの実際の値との関係）を変更できます。例えば、倍率が 1 に設定されている場合、測定値は Y 軸の値と等しくなります。倍率として 10 を選択した場合、正確な測定値を得るには、Y 軸の値を 10 で割る必要があります。

測定値の倍率を設定するには、次の手順を実行します。

- 1 設定対象の測定値を選択します。右クリックして [設定] を選択します。[測定値の設定] ダイアログ・ボックスが開きます。
- 2 倍率を変更するには、[自動倍率] チェック・ボックスをクリアして、倍率のリストから倍率を選択します。
- 3 [OK] をクリックして設定を承認し、ダイアログ・ボックスを閉じます。

次の例では、倍率に 1 と 10 を指定して同一のグラフを表示しています。



左のグラフでは、グラフの実際の値が 0 から 1 の間で推移しています。右のグラフのように倍率大きくして表示すれば、その情報をより正確に表示できます。ただし、実際の値を得るには、表示されている値を倍率で割る必要があります。上の例では、グラフに表示されている最高値が 5 です。倍率は 10 なので、実際の値は 0.5 です。

グラフの下の凡例には、グラフの倍率が示されています。

色	拡大縮小	測定値	単位	最高	最低	平均	標準	末尾
■	10	Processor Queue Length (System)	zeus	3		1.823529..	0.705882..	1
■	1	File Data Operations/sec (System)	zeus	127,1463..	9,64241..	43,56983..	24,31799..	43,928041

↑
倍率

LoadRunner では [自動倍率] オプションが標準で選択されています。このオプションが選択されていると、グラフの表示に最適な比率が計算され、測定値の倍率が自動的に算出されます。

トランザクションの非表示と表示

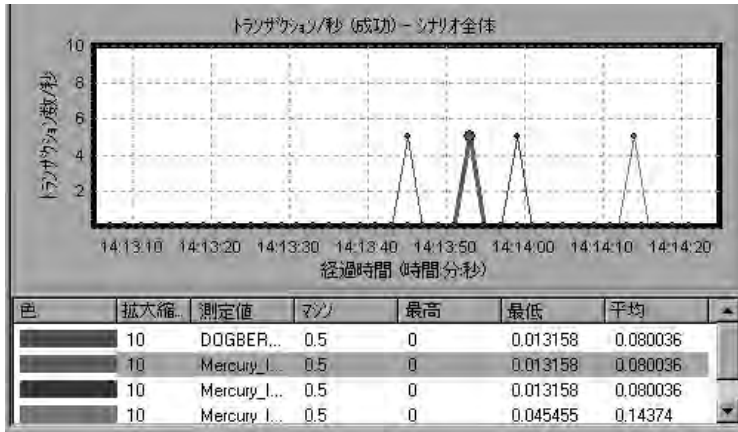
標準設定では、トランザクション・モニタにトランザクション・リストの各項目に対応する線が表示されます。特定のトランザクションに注目するために、監視中の任意のトランザクションの線を非表示にできます。

トランザクションを非表示または表示にするには、次の手順を実行します。

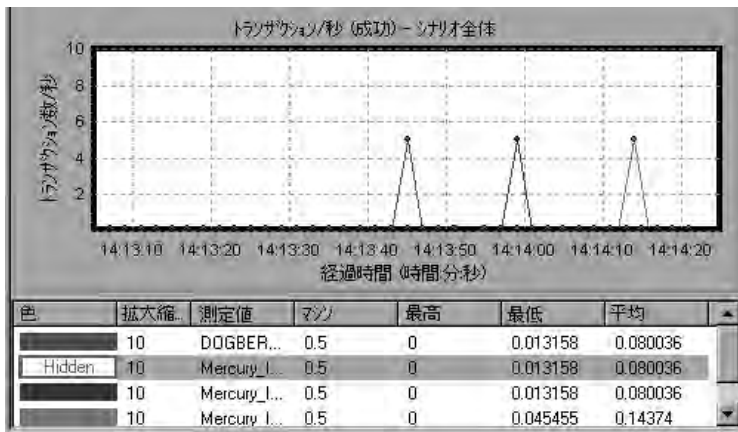
- 1 トランザクションを非表示にするには、[非表示] をクリックします。非表示のリソースを表示するには、[表示する] をクリックします。選択したトランザクション以外のすべてのトランザクションを非表示にするには、そのトランザクションを右クリックして [選択済みのみ表示] を選択します。
- 2 [OK] をクリックして設定を承認し、ダイアログ・ボックスを閉じます。

また、凡例のトランザクションを右クリックして [表示] または [非表示] を選択することで、[測定値の設定] ダイアログ・ボックスを開かずにトランザクションの表示 / 非表示を切り替えることができます。

次の例では、各トランザクションの線が表示されています。



次の例では、凡例の2つ目の項目が非表示になっています。



【測定値の設定】ダイアログ・ボックスの【設定】タブについて

【測定値の設定】ダイアログ・ボックスの【設定】タブでは、線の色の変更、測定値の倍率の設定、トランザクションの表示 / 非表示の切り替えができます。

【測定値】：監視対象リソースの種類が表示されます。

【マシン】：リソースが監視対象になっているマシンの名前が表示されます（マシンのリソースが監視されている場合にだけ表示されます）。

注：ネットワーク・パスを監視しているときには、ここに**マシン**の代わりに**ネットワーク・タイプ**が表示されます。

[配色]：選択した測定値に割り当てる色を選択します。

[倍率]：Y 軸とグラフの実際の値の関係が表示されます。例えば、倍率が 1 に設定されている場合、測定値は Y 軸の値と等しくなります。倍率として 10 を選択した場合、正確な測定値を得るには、Y 軸の値に 10 をかける必要があります。

[自動倍率]：LoadRunner に対して、グラフの表示に最適な比率を計算して測定値の倍率を自動的に設定するように指示します。一部のグラフでは、このオプションは使用できません。

[表示]：選択したリソースを表示します。選択したリソースの線がグラフに再表示されます。標準設定では、すべてのリソースの測定値がグラフに表示されます。

[非表示]：選択したリソースを非表示にします。選択したリソースの線がグラフから消えます。非表示にしたリソースは、凡例の中で塗りつぶされていないボックスとして表示されます。

[測定値の設定] ダイアログ・ボックスの [詳細] タブについて

[測定値の設定] ダイアログ・ボックスの [詳細] タブには、測定値に関する情報が表示されます。

[測定値]：監視対象リソースの種類が表示されます。

[マシン]：リソースが監視対象になっているマシンの名前が表示されます（マシンのリソースが監視されている場合にだけ表示されます）。

[詳細]：選択された測定値の説明を表示します。

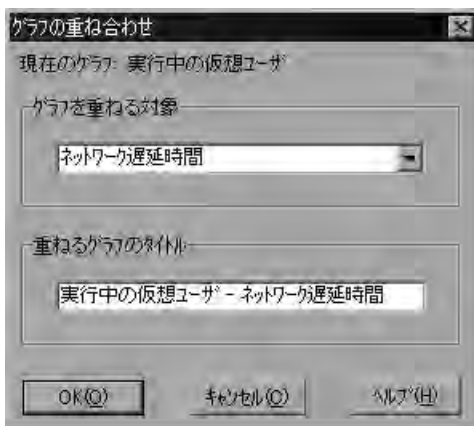
グラフの結合

LoadRunner を使って、1つのシナリオの2つの結果グラフを1つのグラフに結合できます。結合することで、いくつかの異なる測定項目を一度に比較できます。たとえば、Web スループットと秒ごとのヒット数をシナリオの経過時間の関数として表示するようなグラフを作成できます。グラフを結合するには、両方のグラフの X 軸の測定値が同じでなければなりません。

共通の X 軸を持つ2つのグラフの内容を合成すると、結合後のグラフの左の Y 軸には、現在のグラフの値が表示されます。右の Y 軸には、結合されたグラフの値が表示されます。

2つのグラフを合成するには、次の手順を実行します。

- 1 合成対象のグラフを右クリックして、[**グラフを重ねる**] を選択します。[グラフの重ね合わせ] ダイアログ・ボックスが開きます。



- 2 現在のグラフに合成するグラフを選択します。ドロップダウン・リストには、現在のグラフと共通の X 軸を持つアクティブなグラフだけが表示されます。
- 3 合成後のグラフのタイトルを入力します。
- 4 [OK] をクリックします。結合したグラフがグラフ表示領域に表示されます。

オンライン・モニタ・グラフのエクスポート

LoadRunner では、オンライン・グラフを HTML 形式でエクスポートして後で見ることができます。HTML 形式でエクスポートすると、グラフと一緒に凡例も表示されます。

オンライン・グラフを HTML 形式でエクスポートするには、次の手順を実行します。

- 1 オンライン・モニタ・ビューのすべてのグラフをエクスポートするには、[**モニタ**] > [**オンライン グラフを HTML にエクスポート**] を選択します。[ファイル名とパスを選択してください。] ダイアログ・ボックスが開きます。
- 2 ファイル名とパスを指定して [**保存**] をクリックします。

オフラインでのデータ表示

シナリオ実行中にリソースを監視した後に、LoadRunner アナリシスを使って、収集したデータのグラフを表示できます。アナリシス・ユーティリティを実行すると、データが処理され、監視の対象となった各測定値のグラフが作成されます。

グラフを表示するには、アナリシスのウィンドウで [**グラフ**] > [**グラフの追加**] を選択します。シナリオの実行後に LoadRunner アナリシスを使う作業の詳細については、『**Mercury LoadRunner アナリシス・ユーザーズ・ガイド**』を参照してください。

利用可能なグラフ・ツリー

利用可能なグラフ・ツリーには、LoadRunner グラフが表示されます。

グラフを開くには、グラフ・ツリーでグラフをクリックして、[実行] ウィンドウの右の表示枠までドラッグします。

グラフ上で監視する測定値を選択するには、『**Mercury LoadRunner モニタ・リファレンス**』の関連するモニタの設定方法を参照してください。

第7部

付録

付録 A

LoadRunner オンライン・グラフの解析

LoadRunner オンライン・モニタ・グラフは、シナリオのパフォーマンスに関する重要な情報を示します。この付録では、主要なオンライン・グラフを詳しく説明し、シナリオの実行時に、このグラフを使ってパフォーマンスのボトルネックを特定し、切り分ける方法を示します。

オンライン・モニタ・グラフ

オンライン・モニタ・グラフを使って、トランザクションが許容時間内に完了しているか、ダウンロード時間を最小限に保つのに帯域幅が十分かどうか、またハードウェアとオペレーティング・システムがピーク負荷に対応できるかどうかを判定できます。

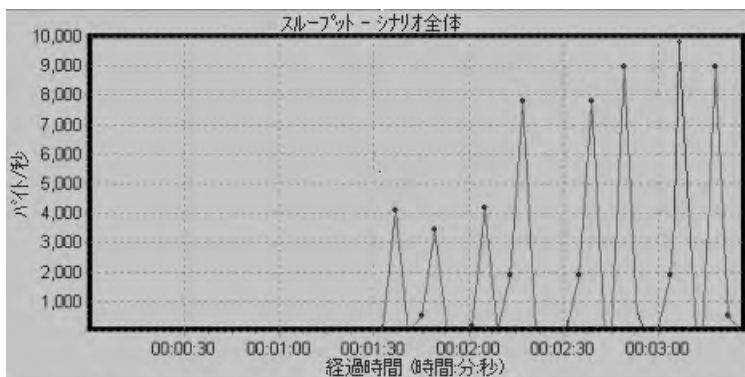
質問 1: シナリオのすべてのトランザクションが許容時間内に完了していますか？ どのトランザクションが長時間を要していますか？

回答: **[トランザクション応答時間]** グラフには、各トランザクションが完了するのに要した時間が表示されます。以下のグラフは、ログイン・トランザクション以外のトランザクション応答時間が速いことを示しています。最初のログインでは、長い時間かかっていませんが、次のログインから極端に遅くなっていることを示します。これは、データベースが同時に2つ以上のログインを処理できず、その原因はおそらくデータベースのクエリ機能が十分でないことを示しています。



質問2：帯域幅はダウンロード時間を最小限に抑えるのに十分ですか？

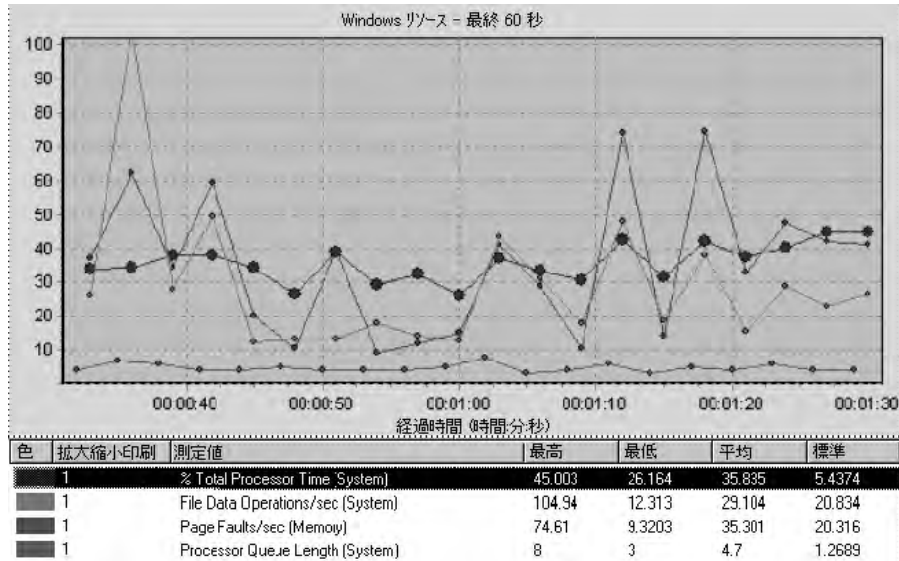
回答：[スループット] グラフは、シナリオ実行の経過秒ごとの Web サーバのスループットを示します。スループットは、任意の秒ごとにサーバから受け取るデータの量を示します。



上のグラフでは、スループットが時間の経過とユーザ数の増加に伴って増えており、帯域幅が十分なことを示しています。ユーザ数が増加してもグラフに変化があまり見られない場合は、要求しているデータの量に対して、帯域幅によって制限が課せられていると考えられます。

質問 3 : ハードウェアとオペレーティング ・ システムはピーク負荷に対応できますか？

回答 : [Windows リソース] グラフは、Windows サーバのリソースの使用量をリアルタイムで示します。このグラフを使って、シナリオ実行時に特定のマシンで使用されているリソースと、ボトルネックを特定できます。



上のグラフの [% Total Processor Time] は、サーバによって処理されるデータ量を示します。[File Data Operations/sec] には、サーバによってファイル ・ システム ・ デバイスに発行される読み込みと書き込みの割合が表示されます。[Page Faults/sec] は、仮想メモリとキャッシュを処理するアルゴリズムの実行頻度を示すプロセッサ内のページ ・ フォルトの数を示します。

一般的に、新しい高速なサーバを導入すると、ダウンロード時間に関する問題が解消されます。ただし上のグラフでは、サーバで処理されているデータが少量だということがわかります。このグラフから、サーバのプロセッサ能力は十分で、ハードウェアを追加してもパフォーマンス向上につながらないことがわかります。ただし、データ ・ ファイル ・ システムを最適化することによってパフォーマンスを向上できることもあります。

付録 B

パス変換の実行

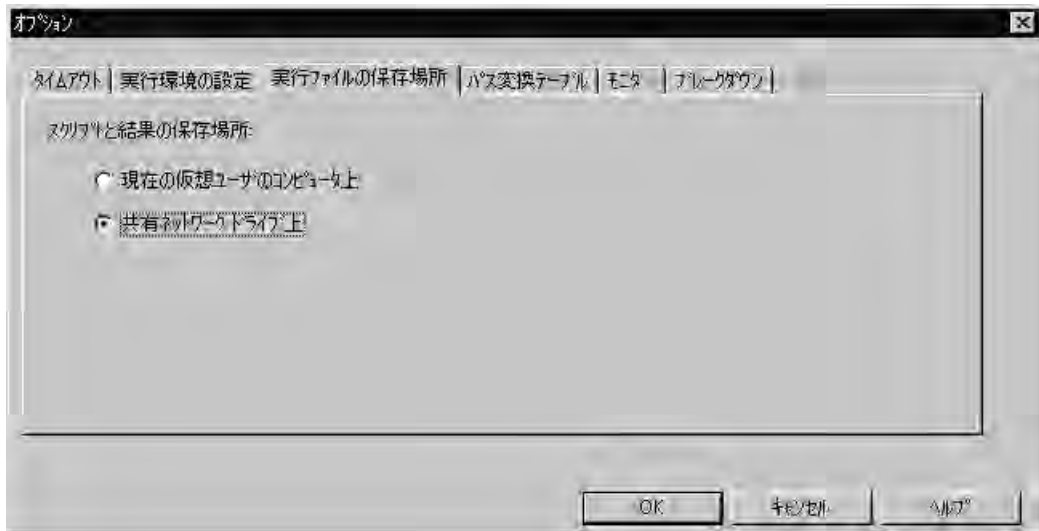
シナリオを実行すると、LoadRunner は参加している仮想ユーザから実行時のデータを収集します。標準では、LoadRunner は、各仮想ユーザ・マシンの一時ファイル内にデータを格納します。シナリオの実行後、データは総合的な結果ディレクトリで照合されます。

または、実行時のデータを LoadRunner からネットワーク・ドライブに直接書き込ませることも可能です。第 10 章「シナリオの設定」を参照してください。ただし、この方法ではネットワーク・トラフィックが増え、パス変換も必要となるので避けることをお勧めします。

パス変換について

パス変換は、LoadRunner がコントローラ用にリモート・パス名を変換するために使う仕組みです。典型的なシナリオの例として、LoadRunner コントローラが Windows マシンで実行され、複数の仮想ユーザが Windows ロード・ジェネレータと UNIX ロード・ジェネレータで実行されている場合を考えます。同一のネットワーク・ドライブを、あるリモート・ロード・ジェネレータでは F に割り当て、別のロード・ジェネレータでは H に割り当てたとします。このような複雑なシナリオでは、参加しているすべてのマシンが同じネットワーク・ドライブを認識できるようにしておく必要があります。

LoadRunner では、[オプション] ダイアログ・ボックスの [実行ファイルの保存場所] タブで、共有ネットワーク・ドライブにスクリプトと実行時の結果データを格納するように設定できます。



共有ネットワーク・ドライブに結果とスクリプト・ファイルを格納する場合にはパス変換を行う必要があります。

[デザイン] タブのシナリオ・グループ/シナリオ・スクリプト表示枠には、シナリオに関連するすべての仮想ユーザ・スクリプトとそれらの格納場所が表示されます。スクリプトの格納場所（パス）は、必ず、コントローラ・マシンによって割り当てられた場所に基づきます。仮想ユーザのロード・ジェネレータで、スクリプトのパスとして別のパス名が割り当てられている場合は、パス変換が必要です。

例えば、コントローラが **pc2** という Windows マシンで実行されており、仮想ユーザ・スクリプトがネットワーク・ドライブに格納されているとします。コントローラ・マシンは、ネットワーク・ドライブを **m:\lr_tests** として割り当てているとします。このときに、仮想ユーザをホストしているリモート・マシン（ロード・ジェネレータ）もパスを **m:\lr_tests** に割り当てていれば、変換は必要ありません。しかし、リモート・マシンが **r:\lr_tests** のようにこのパスをほかのドライブやパスとして割り当てている場合は、ロード・ジェネレータがスクリプトの格納場所を認識できるように、パスを変換しなければなりません。

同様に、コントローラとリモート・ロード・ジェネレータで違うパス名に割り当てられている共有ドライブに実行時の結果ファイルを保存する場合もパス変換が必要です。

パス変換は、異なるプラットフォーム（Windows と UNIX）間でも使用できます。パス変換を使って、Windows マシンのパス（コントローラによって認識されるパス）を UNIX 仮想ユーザ・ロード・ジェネレータで認識可能なパスに変換します。

パス変換テーブルへのエントリの追加

Windows マシン間、または Windows と UNIX マシンの間でパスを変換するには、パス変換テーブルにエントリを作成します。このテーブルには、個々のマシンが認識できる形式に変換されるパスのリストが含まれます。

パス変換テーブルの各行の形式は次のとおりです。

```
< controller_host > < controller_path > < remote_path > [ < remote_host > ]
```

controller_host : コントローラを実行しているホストの名前または種類。例えば、コントローラを Windows マシンで実行している場合、このホスト・フィールドに **win** と入力します。また、**LOADPC1** のようにコントローラを実行しているマシン名を指定することも可能です。

controller_host には次の値を指定できます。

- ▶ **hostname** : コントローラを実行しているマシンの名前。
- ▶ **win** : コントローラを Windows マシンで実行している。
- ▶ **unix** : コントローラを UNIX マシンで実行している。
- ▶ **all** : コントローラを Windows マシンまたは UNIX マシンで実行している。

controller_path : コントローラによって認識されている特定のディレクトリのパス。例えば、ディレクトリ **scripts** がネットワーク・ドライブ（コントローラで割り当てられているネットワーク・ドライブ名）に格納されている場合、**controller_path** フィールドに **r:\%scripts** というパスを指定します。

remote_path : リモート・マシンによって認識されている特定のディレクトリのパス。例えば、ディレクトリ **scripts** がネットワーク・ドライブ **n**（リモー

ト・ロード・ジェネレータで割り当てられているネットワーク・ドライブ名)に格納されている場合、**remote_path** フィールドに `n:¥scripts` というパスを指定します。

リモート UNIX ロード・ジェネレータで仮想ユーザが上記のパスを `/m/tests` として認識する場合、このパスを **remote_path** フィールドに入力します。

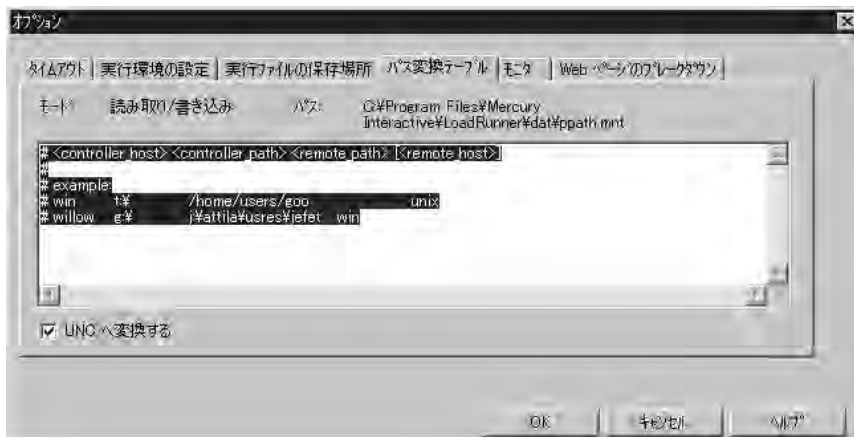
remote_host : リモート・ロード・ジェネレータの名前または種類。例えば、すべてのリモートマシンが UNIX ワークステーションの場合、**remote_host** フィールドに `unix` と指定します。**remote_host** フィールドのオプションは、上記の **controller_host** フィールドのオプションと同じです。**remote_host** パラメータは省略可能です。

パス変換テーブルの編集

パス変換テーブルの保守には、LoadRunner コントローラを使います。LoadRunner は、パス変換テーブルを **ppath.mnt** という名前の ASCII ファイルとして保存します。このファイルは **<LoadRunner のインストール先フォルダ> ¥dat** に格納されており、変換するネットワーク・パスごとに1行のエントリがあります。

パス変換テーブルを編集するには、次の手順で行います。

- 1 LoadRunner コントローラを起動します。
- 2 [ツール] > [オプション] を選択し、[パス変換テーブル] タブを選びます。[パス変換テーブル] ウィンドウが開きます。



- 3 パス変換情報を指定する前に、UNC (Universal Naming Convention) 方式を使うことを検討してください。ホストが Windows マシンの場合、コントローラに対して、すべてのパスを UNC に変換するよう指示できます。パスを UNC に変換すれば、どのマシンもパス変換を行わなくてもパスを認識できます。UNC 形式では `¥¥machine_a¥results` のようになります。

[UNC へ変換する] チェック・ボックスを選択して、パス変換テーブルを無視してすべてのパスを UNC に変換するように LoadRunner に指示します。

- 4 ホストが Windows マシンではなく、パス変換が必要な場合、テーブルにパスの情報を入力します。コメントを挿入するには、テーブルのコメント行の先頭に「#」記号を指定します。
- 5 [OK] をクリックしてテーブルを閉じ、情報を保存します。

パス変換の例

次に、パス変換テーブルのエントリの例を示します。

注： Windows マシンのパスを UNIX マシンのパスに変換する場合は、正しいスラッシュ (UNIX ではスラッシュ、Windows ではバックスラッシュもしくは円記号) を入力しなければなりません。

次の例は、Merlin という Windows マシンのコントローラにおけるパス変換テーブルを示します。

最初の例では、仮想ユーザが Oasis という Windows 2000 マシンで実行されています。Merlin はネットワーク・ドライブを f: に割り当て、Oasis は g:¥loadtest を割り当てています。

merlin	f:¥	g:¥loadtest¥	Oasis
--------	-----	--------------	-------

2 番目の例では、仮想ユーザが Ultra という UNIX マシンで実行されています。Ultra はネットワーク・ドライブを /u/tests/load に割り当てています。

merlin	f:¥	/u/tests/load/	Ultra
--------	-----	----------------	-------

3番目の例では、リモート・ロード・ジェネレータ **Jaguar** のネットワーク・ドライブの割り当てがコントローラの割り当てと同じなので、変換は必要ありません。この行はパス変換テーブルから削除しても問題ありません。

merlin	n:¥	n:¥	Jaguar
--------	-----	-----	--------

4番目の例では、すべての Windows マシンの仮想ユーザ・ロード・ジェネレータでネットワーク・ドライブを m:¥loadtest に割り当てています。

merlin	l:¥mnt¥	m:¥loadtest¥	win
--------	---------	--------------	-----

付録 C

エキスパート・モードでの作業

上級ユーザは、「**エキスパート・モード**」で作業をするときに LoadRunner の設定を調整できます。エキスパート・モードでは、[オプション] ダイアログ・ボックスと [ロード ジェネレータの情報] ダイアログ・ボックスに追加オプションが表示されます。この付録では、エキスパート・モードで利用できる設定について説明します。

- ▶ エキスパート・モードへの切り替え方
- ▶ オプション — 一般設定
- ▶ オプション — デバッグ情報の設定
- ▶ オプション — 出力の設定
- ▶ オプション — モニタの設定
- ▶ ロード・ジェネレータ情報 — UNIX 環境の設定
- ▶ ロード・ジェネレータ情報 — 接続ログの設定

エキスパート・モードへの切り替え方

LoadRunner コントローラのエキスパート・モードは、サポート担当者がシステム情報にアクセスできるようにするためのものです。エキスパート・モードを使用すると、コントローラの操作をより細かく調整するためのオプションがコントローラのダイアログ・ボックスに表示されます。

エキスパート・モードをアクティブにするには、[ツール] > [**エキスパートモード**] を選択します。エキスパート・モードが有効になっているときは、メニュー項目にチェック・マークが付いています。

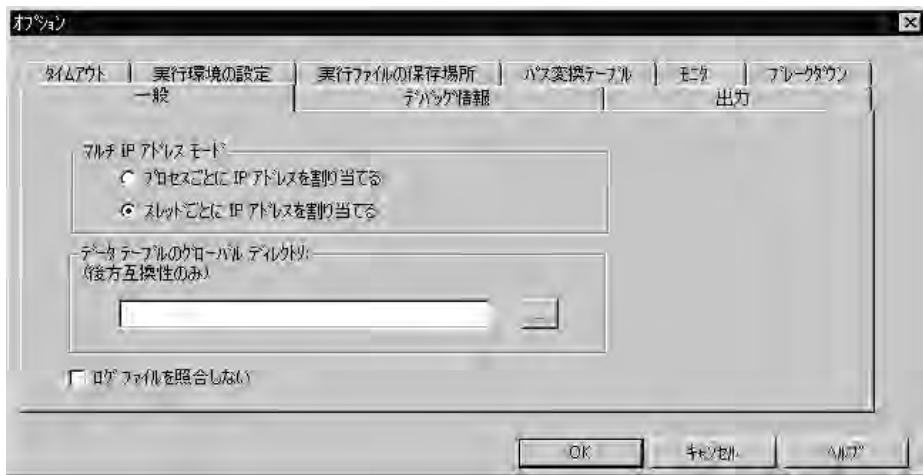
エキスパート・モードを終了する場合も、同じ手順を繰り返します。

オプション — 一般設定

[オプション] ダイアログ・ボックスの [一般] タブでは、データ・テーブル格納領域のグローバルな設定やマルチ IP アドレスの割り当てを指定し、LoadRunner がログ・ファイルを照合しないようにできます。このタブは、コントローラをエキスパート・モードで使用している場合にのみ表示されます。

エキスパート・モードの一般設定は、次の手順で行います。

- 1 [ツール] > [オプション] を選択します。[オプション] ダイアログ・ボックスが開きます。[一般] タブを選択します。



- 2 [マルチ IP アドレス モード] を選択します。
- 3 [データ テーブルのグローバルディレクトリ] を入力します。
- 4 LoadRunner でログ・ファイルを照合せずに結果ファイルだけを照合する場合は、[ログ ファイルを照合しない] を選択します。
- 5 [OK] をクリックして設定を適用し、ダイアログ・ボックスを閉じます。

[オプション — 一般] タブについて

[一般] タブでは、データテーブル格納領域のグローバルな設定や、ログファイルの照合、マルチ IP アドレスの割り当てを指定することができます。

[マルチ IP アドレス モード]: マルチ IP アドレス・オプション機能を有効にしたとき ([シナリオ] > [IP スプーファを有効にする]) に IP アドレスの割り

当てに使用されるモード。コントローラは、プロセスごとまたはスレッドごとに IP アドレスを割り当てることができます。スレッドごとに割り当てを行うと、1つのシナリオにおける IP アドレスの分散の範囲が広がります。

[データ テーブルのグローバル ディレクトリ]：パラメータ値のソースとして使用されるデータ・テーブルのネットワーク上の場所。この設定は、旧バージョンの LoadRunner で作成されたスクリプトに対してのみ必要です。

[ログ ファイルを照合しない]：LoadRunner がログ・ファイルは照合せずに結果ファイルだけを照合するようにします。

オプション — デバッグ情報の設定

[オプション] ダイアログ・ボックスの [デバッグ情報] タブでは、シナリオの実行時にどの程度までのトレースを行うかを指定できます。デバッグ情報は、[出力メッセージ] ウィンドウに書き込まれます。このタブは、コントローラをエキスパート・モードで使用している場合にのみ表示されます。

[デバッグ情報] を設定するには、次の手順で行います。

- 1 [ツール] > [オプション] を選択します。[オプション] ダイアログ・ボックスが開きます。[デバッグ情報] タブを選択します。



- 2 使用するトレース・フラグのチェック・ボックスを選択します。
- 3 一時実行時ファイルを保存するには、[一時ファイルを保持する] を選択します。

- 4 [OK] をクリックして設定を適用し、ダイアログ・ボックスを閉じます。

[オプション - デバッグ情報] タブについて

LoadRunner のデバッグの設定を定義できます。

[**フラグのトレース**] : デバッグのために、テストの実行中に LoadRunner によって実行されるトレースの種類を設定できます。詳細なトレースを行うには、チェック・ボックスを選択します。指定したエージェント・ログ・ディレクトリに含まれるログ・ファイルにトレース情報が表示されます。使用可能なトレース・フラグは、「**一般**」、「**ファイル転送**」、「**受信**」、「**送信**」です。この中から問題に関連したフラグを選択します。例えば、ファイル転送で問題が生じた場合は、「**ファイル転送**」フラグを選択します。

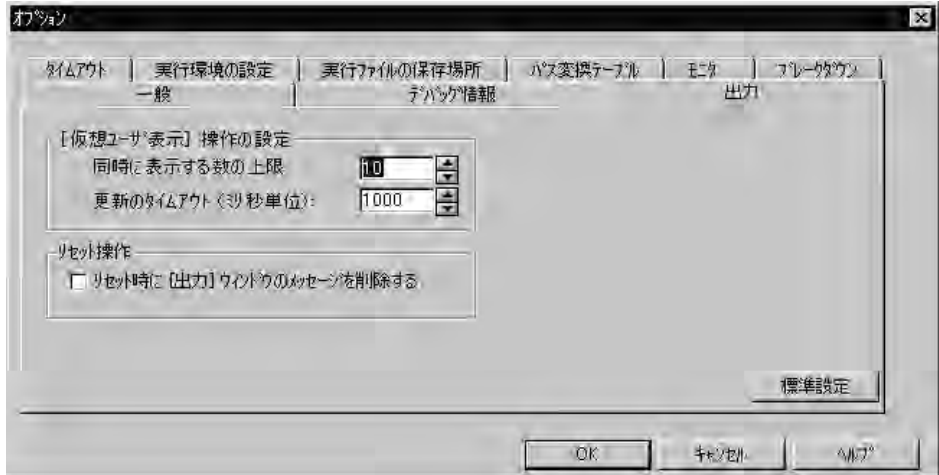
[**一時ファイルを保持する**] : エージェントとコントローラは複数の一時ファイルを作成します。一時ファイルには、仮想ユーザに送信されるパラメータ・ファイル、出力コンパイル・ファイル、構成設定ファイルなどの情報が含まれます。エージェント・ファイルは、エージェント・マシンの TMP ディレクトリまたは TEMP ディレクトリ内にある **brr** フォルダに格納されています。コントローラ・ファイルは、コントローラ・マシンの TMP ディレクトリまたは TEMP ディレクトリ内にある **lrr** フォルダに格納されています。シナリオの終了時に、これらのファイルはすべて自動的に削除されます。[**一時ファイルを保持する**] を設定することで、デバッグに必要なファイルを削除しないようエージェントとコントローラに指示できます。

オプション - 出力の設定

[オプション] ダイアログ・ボックスの [出力] タブを使用して、仮想ユーザを実行時にコントローラ・マシンにどのように表示するかを設定できます。

[出力] を設定するには、次の手順で行います。

- 1 [ツール] > [オプション] を選択します。[オプション] ダイアログ・ボックスが開きます。[出力] タブを選択します。



- 2 [同時に表示する数の上限] ボックスに、同時に表示する仮想ユーザ・ログの最大数を指定します。
- 3 [更新のタイムアウト (ミリ秒単位)] ボックスに、LoadRunner がユーザ・ログを更新する頻度を指定します。
- 4 シナリオのリセット時に [出力メッセージ] ウィンドウのメッセージをクリアするには、[リセット時に [出力] ウィンドウのメッセージを削除する] チェック・ボックスを選択します。
- 5 [OK] をクリックして設定を適用し、ダイアログ・ボックスを閉じます。

[オプション - 出力] タブについて

[出力] タブを使用して、コントローラ・マシンで実行中の仮想ユーザをどのように表示するかを設定できます。

[[仮想ユーザ表示] 操作の設定] :

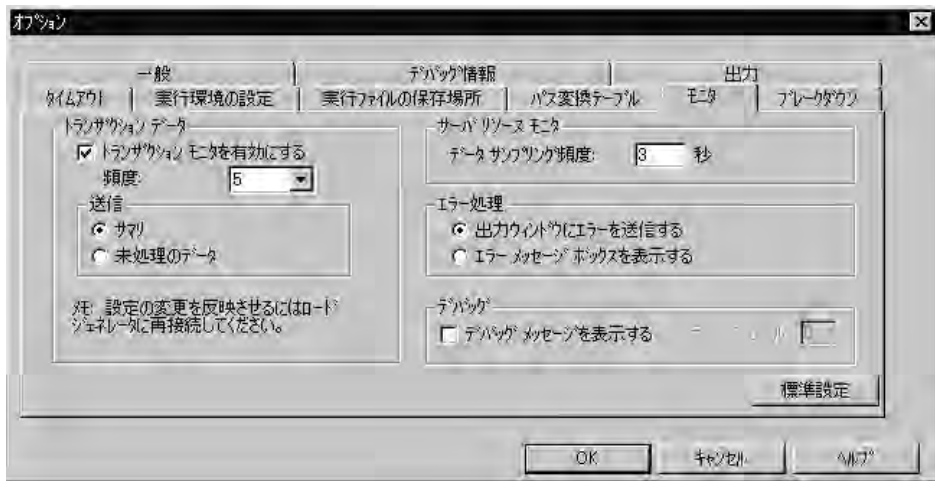
- ▶ [同時に表示する数の上限] : 同時に表示できる仮想ユーザ・ログの最大数と、マシンで実行時ビューアを開いてコントローラが表示するアクティブな UNIX, GUI, RTE または Web の仮想ユーザの最大数を指定します。標準設定の値は 10 です。
- ▶ [更新のタイムアウト (ミリ秒単位)] : 仮想ユーザ・ログを更新する頻度を指定します。標準設定は 1000 ミリ秒ごとです。

[リセット時に [出力] ウィンドウのメッセージを削除する] : シナリオのリセット時に, [出力メッセージ] ウィンドウ内のすべてのメッセージをクリアにするよう, LoadRunner に命じます。

オプション - モニタの設定

エキスパート・モードには, 次の追加モニタ設定があります。

[送信] セクションの [サマリ] および [未処理のデータ] : 収集されたデータのサマリ, またはすべてのデータを未処理のままコントローラに送信します。データを未処理のまま送信すると, データを処理する必要がないので時間を節約できます。ただし, すべてのデータがコントローラに転送されるので, ネットワーク・トラフィックが増える可能性があります。転送速度を重視する場合は, [サマリ] を選択します。



ロード・ジェネレータ情報 - UNIX 環境の設定

エキスパート・モードでは, UNIX 環境向けに次の項目を設定できます。

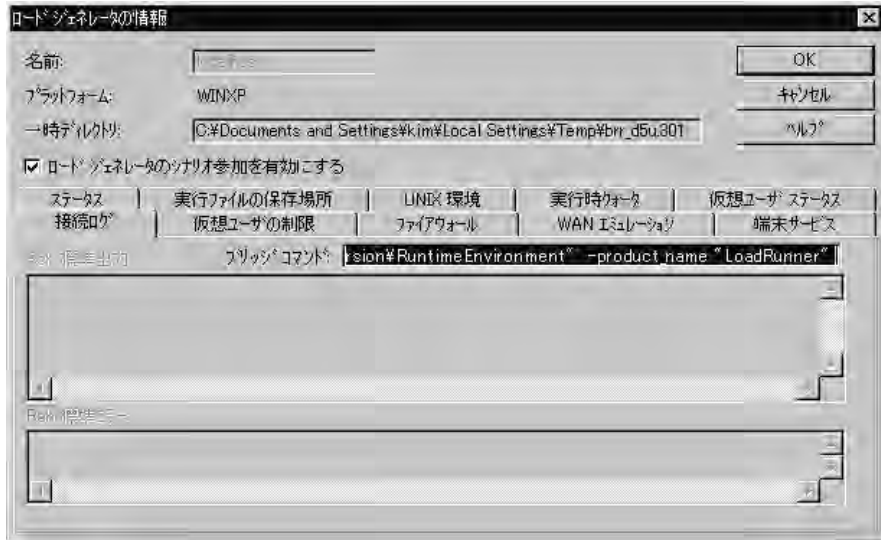
[ローカル ユーザ] : **rsh** シェルを使用する UNIX ロード・ジェネレータは, セキュリティ上の理由から現在の NT ユーザとして接続を確立します。rsh を「ごまかし」て, 現在の NT ユーザとは別のユーザとしてログインするためには, [ローカル ユーザ] チェック・ボックスを選択し, UNIX ログイン名を入力します。ローカル・ユーザ名の変更は, **rsh** のセキュリティを侵害します。した

がって、このオプションは、リモート・マシンへの接続に問題が発生した場合に限って使用するべきものです。



ロード・ジェネレータ情報 — 接続ログの設定

[ロードジェネレータの情報] ダイアログ・ボックスの [接続ログ] タブでは、コントローラが指定の UNIX ロード・ジェネレータに接続する際に生成される、標準出力および標準エラーを表示することができます。コントローラがロード・ジェネレータに接続するためにリモート・ブリッジに送信するコマンドを変更することもできます。

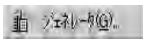


[Rsh 標準出力] : コントローラが選択された UNIX ロード・ジェネレータに接続したときの rsh の標準出力を表示します。

[ブリッジコマンド] : コントローラが UNIX ロード・ジェネレータに接続するためにリモート・ブリッジに送信する標準ブリッジ・コマンドを変更する場合は、新しいコマンドを入力します。

[Rsh 標準エラー] : コントローラが選択された UNIX ロード・ジェネレータに接続したときの rsh の標準エラーを表示します。

[接続ログ] を設定するには、次の手順で行います。



- 1 [ジェネレータ] ボタンをクリックするか、[シナリオ] > [ロードジェネレータ] を選択します。[ロードジェネレータ] ダイアログ・ボックスが表示されます。
- 2 [接続] をクリックして、ロード・ジェネレータの [ステータス] を「ダウン」から「準備完了」に変更します。
- 3 [詳細] ボタンをクリックします。[ロードジェネレータの情報] ダイアログ・ボックスで、[接続ログ] タブを選択します。
- 4 rsh 標準出力および標準エラーを表示するか、[ブリッジコマンド] ボックスに新しいコマンドを入力して、標準のブリッジ・コマンドを変更します。

付録 D

コントローラのトラブルシューティング

LoadRunner では、アプリケーション全体をテストできます。アプリケーションのコンポーネントが 1 つでも適切に設定されていないと、LoadRunner のシナリオは実行されません。

この付録では、LoadRunner で起こる可能性のある問題を取り上げます。

- ▶ トラブルシューティングについて
- ▶ LoadRunner 通信
- ▶ ロード・ジェネレータとの通信の失敗
- ▶ AUT データベースへの接続の失敗
- ▶ ファイルへのアクセスの失敗
- ▶ 仮想ユーザまたはトランザクションの失敗
- ▶ Windows マシンの仮想ユーザ数の引き上げ
- ▶ ファイアウォールのトラブルシューティング
- ▶ LoadRunner エージェントを使った作業

トラブルシューティングについて

LoadRunner はネットワーク上のマシン間の通信に大きく依存しています。通信が正しく確立されないと、コントローラはリモート・ロード・ジェネレータにコマンドを送ることができず、シナリオは失敗します。失敗の理由を理解し、いつ失敗が起こったのか突き止めることによって、ほとんどのネットワーク通信関連の問題を解決できます。

問題が仮想ユーザ・スクリプトにではなく、シナリオにあることを確認するには、スクリプトがスタンドアロン・モードですべてのリモート・ロード・ジェネレータで正常に実行できることを検証しなければなりません。

- ▶ Windows プラットフォームで GUI 仮想ユーザ・スクリプトをテストするには、WinRunner を使います。
- ▶ UNIX プラットフォームで仮想ユーザ・スクリプトをテストするには、コマンド・ラインから実行します。
- ▶ Windows プラットフォーム上で、他のすべての種類の仮想ユーザ・スクリプトをテストするには、VuGen から仮想ユーザ・スクリプトを実行するか、コントローラから1つの仮想ユーザを実行します。

注：VuGen でテストを実行すると、完全なブラウザが使用されます。これは、基本ブラウザが使用されるコントローラでのテスト実行とは異なります。そのため、VuGen でテストの実行が成功しても、コントローラでテストの実行が失敗する可能性があります。コントローラで複数の仮想ユーザのシナリオを実行する前に、仮想ユーザを1つだけ実行して不具合がないことを確認します。

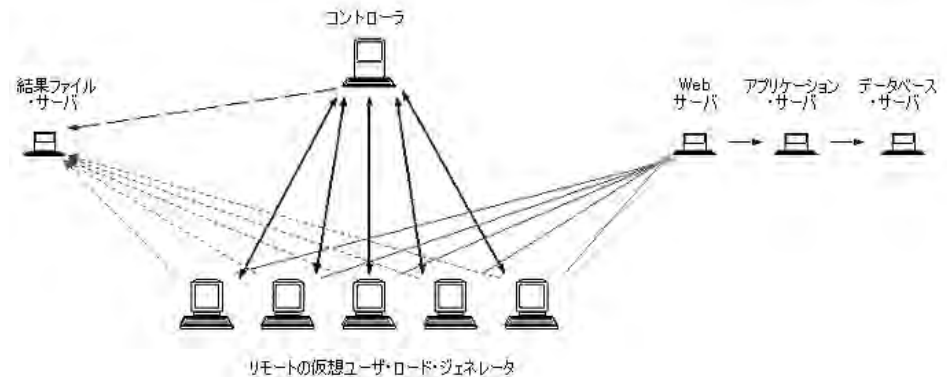
スタンドアロン・モードで仮想ユーザを実行する方法の詳細については、仮想ユーザ・スクリプトの作成について書かれたマニュアルを参照してください。

LoadRunner 通信

通信に関する問題のほとんどは、LoadRunner の設定を理解していれば解決できます。設定に関する理解は、問題の原因を突き止め、必要な処置を行うのに役立ちます。

次の図に、LoadRunner を実行しているネットワークの例を示します。次の5つのサーバがあります。LoadRunner コントローラ、Web サーバ、アプリケーション・サーバ、データベース・サーバ、シナリオ結果を格納するファイル・サーバ（結果ファイルは、専用サーバでないサーバにも格納できます）。また、5つのリモート・ロード・ジェネレータがあり、それぞれ複数の仮想ユーザを実行しています。

矢印は、ネットワークの要素間で必要な通信のタイプを示します。仮想ユーザとコントローラとの通信は双方向（送信と受信）ですが、仮想ユーザとファイル・サーバとの通信は片方向（送信）です。コントローラは、ファイル・サーバにアクセスできなければなりません。シナリオに参加する仮想ユーザはすべて双方向（送信と受信）で Web サーバと通信できなければなりません。クライアント・マシンがサーバ・マシンに接続するには、クライアント・マシンでサーバのマシン名を解決できなければなりません。



接続がどこかで切断されていると、シナリオは失敗します。

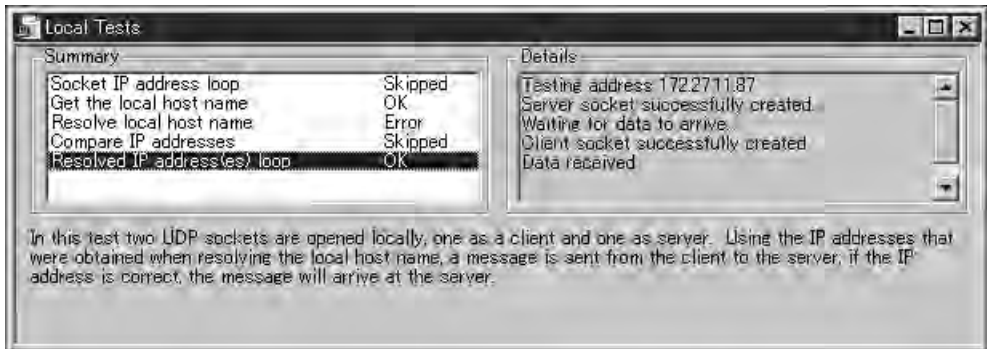
ロード・ジェネレータとの通信の失敗

通信エラーで最も多いのは、コントローラ・マシンからリモート・ロード・ジェネレータに接続できないというものです。次の項目を確認してください。

- ▶ TCP/IP の設定
- ▶ TCP/IP の接続
- ▶ ロード・ジェネレータの接続
- ▶ UNIX シェル

TCP/IP の設定の確認

まず使用しているマシンの TCP/IP の設定を確認します。LoadRunner には、LoadRunner の bin ディレクトリに Hostinfo というユーティリティ (hostinfo.exe) があります。このユーティリティは、ローカル・マシン名やローカル・アドレスなど、現在のマシンに関する情報を提供します。また、TCP/IP が現在のマシンに正しくインストールされているかどうかを確認します。



Hostinfo を起動すると、次のようにして、TCP スタックが自動的に検証されます。

- ▶ ローカル・マシン名が検索され、解決されます。
- ▶ IP アドレスが検索され、解決されます。

IP アドレス解決のために、Hostinfo によって、同じマシンで2つの UDP ソケットを使って通信が試みられます。マシン名の解決で得られた IP アドレスが、このマシンの実際の IP アドレスと同じであるかどうか確認されます。

[Details] ボックスにテストの結果を表示するには、テストの名前を強調表示します。

注： サポート担当者に情報を送信できるように、Hostinfo の [Edit] メニューで、すべてのマシン情報をクリップボードにコピーできます。

TCP/IP の接続の確認

コントローラ・マシンと仮想ユーザのマシンで TCP/IP 接続が機能していることを確認します。ping ユーティリティを使うか、DOS のコマンド・ラインで「ping <サーバ名>」と入力して、リモート・マシンとの通信を確かめます。リモート・ロード・ジェネレータとコントローラ・マシンが、IP アドレスとホスト名の両方を使って相互に ping できることを確認します。

ping に応答がないか、タイムアウト・エラーが発生した場合は、マシン名が認識されていません。この問題を解決するには、**WINNT¥system32¥drivers¥etc** ディレクトリにある hosts ファイルを編集し、IP アドレスと名前から成る行を追加します。

次に例を示します。

```
#      102.54.94.97      rhino.acme.com      # ソース・サーバ
#      38.25.63.10      x.acme.com          # x クライアント・ホスト
```



ロード・ジェネレータの接続

ロード・ジェネレータの接続を検証するには、コントローラの [ロードジェネレータ] ダイアログ・ボックスから個々のリモート・ロード・ジェネレータへ接続します。ロード・ジェネレータの [プラットフォーム] フィールドで、Windows または UNIX を選択します。ロード・ジェネレータを選択し、[接続] ボタンをクリックします。ステータスが、「接続中」に変わります。

接続に失敗すると、ステータスは「失敗」に変わり、[詳細] ボックスに詳細が表示されます。失敗の詳しい情報を見るには、[詳細] ボックスをダブルクリックします。

接続に成功すると、ステータスは「準備完了」に変わり、[プラットフォーム] ボックスに実際のプラットフォーム名 (WINNT, UNIX など) が表示されます。

名前	ステータス	プラットフォーム	詳細
ariel	準備完了	WIN2000	
dambo	失敗	Windows	Load Generator への接続に失敗しました。詳細は出力ウィンドウを参照してください。
localhost	接続中	Windows	
seven1	失敗	Windows	Load Generator への接続に失敗しました。詳細は出力ウィンドウを参照してください。

シナリオで複数のドメイン (コントローラとは異なるドメインの仮想ユーザなど) を使用している場合、コントローラによるロード・ジェネレータとの通信に問題が生じる場合があります。この問題は、コントローラが標準ではドメインを含まない短いロード・ジェネレータ名を使用するために発生します。この

問題を解決するには、コントローラでドメイン名を含むロード・ジェネレータの完全な名前を判別するように設定します。

コントローラのマシンの Windows ディレクトリ内の **miccomm.ini** ファイルを次のように修正します。

```
[tcpnet]
LocalHostNameType= 1
```

LocalHostNameType に設定できる値は次のとおりです。

- 0 – 完全マシン名を使う。
- 1 – 短縮マシン名を使う。この設定が標準です。

注：WINS などの特定の環境では、ロード・ジェネレータはマシン名を解決できません。

複数の IP アドレスを使用したコントローラへの接続

コントローラがロード・ジェネレータ・マシンによって短縮マシン名および完全マシン名のどちらでも認識されず、コントローラ・マシンに複数の IP アドレスが割り当てられている場合は、ロード・ジェネレータの `WINNT¥system32¥drivers¥etc` ディレクトリにある **hosts** ファイルでコントローラ・マシンの別名を定義できます。この別名に対して、ロード・ジェネレータに認識させる IP アドレスを指定します。次に例を示します。

255.0.0.1 delta

UNIX シェル

UNIX 仮想ユーザの場合は、Windows コントローラからリモート・シェル・コマンドを実行できることを確認します。DOS コマンド・プロンプトで次のように入力します。 `rsh -l <UNIX ユーザ・ログイン名> <ロード・ジェネレータ名> <コマンド>` パーミッションの問題を示すようなメッセージが表示された場合は、UNIX のホーム・ディレクトリの **.rhosts** ファイルに、該当するユーザ・ログイン名にコントローラ・マシンからアクセスできる権限が定義されていることを確認してください。場合によっては、文字「+」を **.rhosts** ファイルの最後に指定する必要があります。例えば、コントローラに **bill** としてログオンし、UNIX ロード・ジェネレータに **mike** として接続する場合は、**mike** の名前を使用して **bill** がログオンすることを **mike** が許可しなければなりません。このためには、**mike** の **.rhosts** ファイルの最後に「+bill」という行を指定する

必要があります。

ユーザのログイン名の設定の詳細については、79 ページ「ロード・ジェネレータの追加設定」を参照してください。

rsh を使用せずに **UNIX** を使用するには、次の手順を実行します。

- 1 UNIX ロード・ジェネレータ・マシンで、**< LoadRunner のインストール先フォルダ > /bin** から次のコマンドを実行してエージェント・デーモンを実行します。

```
m_daemon_setup -install
```

これにより、**m_agent_daemon** というデーモンが実行され、正常に実行された場合は、「**m_agent_daemon installed successfully**」というメッセージが表示されます。

エージェントは、ユーザがログオフしても実行を継続します。エージェントは、下の手順 3 で説明するコマンドを実行するか、エージェント・マシンを再起動した場合にだけ終了します。

- ▶ 「**ERROR:File m_agent_daemon doesn't exist**」というメッセージが表示された場合は、実行しようとしたファイルがカレント・ディレクトリにありません。つまり、カレント・ディレクトリが**< LoadRunner のインストール先フォルダ > /bin** ディレクトリでないか、このファイルが実際に存在しないことを意味します。
- ▶ この名前のデーモンが当該ユーザによってすでに実行されている場合は、次の警告が表示されます。

```
WARNING:Could not install m_agent_daemon, reason - user <user_name>
is already running m_agent_daemon on this machine.
```

- ▶ エラーが発生した場合は、次のエラー・メッセージが表示されます。

```
ERROR: Could not install m_agent_daemon. Check log file
m_agent_daemon[xxx].log in your temp directory.
```


- ▶ 一時ディレクトリにあるログ・ファイル `m_agent_daemon[xxx].log` には、インストールが正常に行われた場合でも、次のエラーが記録されます。

```

xterm
Mon Nov 26 16:34:32 2001 Error -10344 : Communication error: -59961 : Failed to
bind socket while calling bind function. [MsgId: MERR-59961] (sys error
message - Permission denied) [MsgId: MERR-10344]
Mon Nov 26 16:34:32 2001 Error -10344 : Communication error: -59927 : Failed to
create a TCP server for the HTTP channel's server. [MsgId: MERR-59927] (sy
s error message - Permission denied) [MsgId: MERR-10344]
Mon Nov 26 16:34:32 2001 Warning -29974 : Failed to create "router" server.
[MsgId: MERR-29974]

```

これらのメッセージが記録される理由は、LoadRunner エージェントが必ずポート番号 443 をリッスンするにもかかわらず（どのエージェントでも MI リスナーとして使用することができ、MI リスナーは必ずこのポートをリッスンするため）、UNIX マシンでは root 以外のユーザがこのポートを開くことができないからです。ただし、このメッセージが表示されても、ロード・ジェネレータ・マシンにこのエージェントを使用することに問題はありません。

- 2 コントローラで、**[ジェネレータ]** > **[詳細]** > **[UNIX 環境]** タブを選択し、**[RSH を使用しない]** オプションを選択します。この後、通常どおり接続を行います。
- 3 エージェント・デーモンを停止するには、**< LoadRunner のインストール先フォルダ > /bin** ディレクトリにある次のコマンドを実行します。

```
m_daemon_setup -remove
```

`m_agent_daemon` が停止します。正常に停止すると、次のメッセージが表示されます。

```
m_agent_daemon removed successfully.
```

- ▶ この名前のデーモンが当該ユーザによって実行されていない場合は、次の警告が表示されます。

```
WARNING:Could not remove m_agent_daemon, reason - user <user_name>
is not running m_agent_daemon on this machine.
```

- ▶ エラーが発生した場合は、次のエラー・メッセージが表示されます。

```
ERROR: Could not remove m_agent_daemon. Check log file
m_agent_daemon[xxx].log in your temp directory.
```

AUT データベースへの接続の失敗

データベース・アプリケーションを実行している場合は、すべてのリモート・クライアントがデータベース・サーバに接続できることを確認しなければなりません。クライアントがサーバにアクセスするとネットワーク・エラーまたは設定エラーが生じる場合は、シナリオを実行する前に問題を解決しておかなければなりません。クライアント・アプリケーションがデータベース・サーバに接続できることを確認するには、次のテストを実行します。

- ▶ Ping
- ▶ SQL ユーティリティ

Ping : TCP/IP を使ってクライアントがデータベース・サーバと通信できることを確認します。ping ユーティリティを使うか、DOS のコマンド・ラインに「ping <サーバ名>」と入力します。

SQL ユーティリティ : ISQL や SQLPLUS のような簡単なユーティリティを使って、データベース・サーバにログオンして、いくつかの基本操作を実行します。

ファイルへのアクセスの失敗

LoadRunner シナリオに参加している 1 台または複数台のマシンから結果パスまたは仮想ユーザ・スクリプトにアクセスできないと、シナリオは失敗します。次の項目を確認してください。

- ▶ パス変換
- ▶ 仮想ユーザ・スクリプト
- ▶ 結果のパス

パス変換 : スクリプトの格納場所 (パス) は、必ず、コントローラ・マシンによって割り当てられた場所に基づきます。仮想ユーザのロード・ジェネレータで、スクリプトのパスとして別のパス名が割り当てられている場合は、パス変換が必要です。パス変換では、コントローラにおける特定の場所の割り当てが、仮想ユーザ・ロード・ジェネレータの割り当てに変換されます。例えば、あるホストでスクリプト・ディレクトリが **g:¥test** に割り当てられていて、別のホストでこれが **h:¥test** に割り当てられている場合、このパスを変換する必要があります。

パス変換は、異なるプラットフォーム（Windows と UNIX）間でも使用できます。パス変換を使って、Windows コントローラのパスを UNIX プラットフォームで認識可能なパスに変換します。

注：パス変換は、すべてのスクリプトと結果を共有ネットワーク・ドライブに保存するようにした場合にだけ必要です。標準設定では、LoadRunner によってファイルがローカルに保存され、コントローラ・マシンと照合されます。この場合、パス変換は不要です。

スクリプトが `/usr/jon/lr_test1` ディレクトリにあり、**sunny** という UNIX マシンで実行されるとします。UNIX ディレクトリが **r** として割り当てられている Windows コントローラ・マシン、**pc1** でパスを変換するには、パス変換テーブルに次のような行を入力します。

pc1	r:¥	/usr/jon	sunny
-----	-----	----------	-------

コントローラの `f:¥qa` ディレクトリを UNIX プラットフォームの `/m/qa/lr_test2/lr_test2.usr` を実行しているすべてのロード・ジェネレータに変換するには、次のように入力します。

win	f:¥qa	/m/qa	UNIX
-----	-------	-------	------

パスが正しく変換されないと、シナリオは失敗します。パス変換の詳細については、付録 B 「パス変換の実行」を参照してください。

仮想ユーザ・スクリプト：パス変換とパーミッションにより、シナリオに参加しているすべてのロード・ジェネレータが仮想ユーザ・スクリプトにアクセスできることを確認します。参加している各ロード・ジェネレータで、仮想ユーザ・スクリプトをスタンドアロン・モードで表示し実行します。

結果のパス：パス変換とパーミッションにより、シナリオに参加しているすべてのロード・ジェネレータが結果のパスにアクセスできることを確認します。

仮想ユーザまたはトランザクションの失敗

LoadRunner 仮想ユーザまたはトランザクションは、ネットワーク、データベース、または実際のスクリプトに関係するさまざまな理由により失敗することがあります。シナリオ実行に関する情報は以下から得ることができます。

- ▶ [実行] ビュー
- ▶ [出力メッセージ] ウィンドウ
- ▶ 出力ファイル（GUI 仮想ユーザを除く）
- ▶ アナリシス・レポートとグラフ

【実行】ビュー

[実行] ビューは、LoadRunner コントローラの一部です。左上に表示される [シナリオグループ] 表示枠に、シナリオの実行中および実行後の仮想ユーザ・グループのステータスが表示されます。シナリオの実行中、カラムには「保留中」、「初期化」、「準備完了」、「実行中」および「ランデブー」のステータスが表示されます。[仮想ユーザ] ダイアログ・ボックスでも個々の仮想ユーザのステータスを表示できます。仮想ユーザが失敗し、スクリプトの実行が完了しないと、LoadRunner によって [エラー] ステータスが表示されます。仮想ユーザがスクリプトの実行を完了すると、LoadRunner によって、完了したスクリプト実行のトランザクション・ステータスを示す「完了-失敗」または「完了-成功」が表示されます。

仮想ユーザ・ステータスの詳細については、第14章「シナリオの実行」を参照してください。

ID	ステータス	スクリプト	ホスト	シナリオ	経過時間
1	完了-失敗 反復試行回数: 5 成功数: 0	test_10	localhost		00:00:07
2	実行中	test_10	localhost		00:00:11
3	完了-失敗 反復試行回数: 5 成功数: 0	test_10	localhost		00:00:07
4	実行中	test_10	localhost		00:00:08
5	実行中	test_10	localhost		00:00:08
6	実行中	test_10	localhost		00:00:08
7	完了-失敗 反復試行回数: 5 成功数: 0	test_10	localhost		00:00:09
8	実行中	test_10	localhost		00:00:09
9	実行中	test_10	localhost		00:00:09
10	完了-失敗 反復試行回数: 5 成功数: 0	test_10	localhost		00:00:08

[出力メッセージ] ウィンドウ

コントローラから [出力メッセージ] ウィンドウを表示します。[出力メッセージ] ウィンドウには、シナリオのデバッグに役立つ情報が表示されます。[出力メッセージ] ウィンドウには、次の5種類のメッセージが表示されます。エラー、警告、通知、デバッグ、バッチです。エラー・メッセージは通常、スクリプトの失敗を示します。警告メッセージは、仮想ユーザに問題が生じたが、テストの実行は継続されたことを示します。通知は、記録された思考遅延時間の値や実行時のその他の情報など役立つ情報を提供します。デバッグ・メッセージは、[ツール] > [オプション] > [デバッグ情報] (エキスパート・モード) を選択してデバッグ機能を有効にした場合に表示されます。バツ

チ・メッセージは、自動化機能を使用している場合に、コントローラでのメッセージ・ボックス表示の代わりに送信されます。



[出力メッセージ] ウィンドウの詳細については、第 15 章「実行中の仮想ユーザの表示」を参照してください。

出力ファイル

スクリプト実行に関する情報は、仮想ユーザ結果ディレクトリにある出力ファイルでも見ることができます。出力ファイル、**output.txt** には、以下が含まれています。

- ▶ シナリオ実行中に呼び出された主な関数のリスト
- ▶ データベース・サーバによって発行されたエラー・メッセージ
- ▶ トランザクションとランデブーの情報

出力ファイルに送信される情報の範囲は、出力ファイルの設定によって異なります。VuGen の実行環境の設定で、[標準ログ] または [拡張ログ] を指定します。拡張ログの場合には、詳細トレース、返されたデータ、または現在のパラメータの値を出力させることができます。拡張ログは、スクリプトのデバッグには役立ちますが、余分なオーバーヘッドの処理が発生するので、デバッグしていないのであれば、拡張ログは使用しないようにしてください。実行環境の設定の詳細については、『**Mercury 仮想ユーザ・ジェネレータ・ユーザズ・ガイド**』を参照してください。

アナリシス・レポートとグラフ

グラフとレポートを作成して、シナリオ実行に関する情報を表示できます。例えば、サマリ・レポートでは、シナリオの実行時データを含むテーブルが表示

され、次のグラフへのリンクが提供されます。実行中の仮想ユーザ、スループット (Web)、秒ごとのヒット数 (Web)、秒ごとの HTTP 応答数、トランザクション・サマリ、平均トランザクション応答時間。

アナリシス サマリ 期間: 12/12/2004 17:54:04 - 12/12/2004 18:...

ツリオ名: G:\Load_Runner_Files\LR80\SCENARIOS\J2ee_Scenario_on_korean_6_vusers.lrs
ツリオの結果: C:\Documents and Settings\rkshuk\デスクトップ
 \Load_Runner_Files\RESULTS\j2ee_pet_store_lm\j2ee_pet_store_lm.lrs
継続時間: 22 分、25 秒。

統計サマリ

最大実行仮想ユーザ数: 6
合計スループット [バイト]: 11,821,728
平均スループット [バイト/秒]: 8,783
合計対象: 4,368
秒ごとの平均対象: 3.245 [HTTP 応答サマリを表示する](#)

トランザクション サマリ

トランザクション: 合計成功数: 156 合計失敗数: 0 中止の合計: 0 [平均応答時間](#)

トランザクション名	最小	平均	最大	標準偏差値	90 パーセント	成功	失敗	停止
Action Transaction	160,703	165,423	182,259	4.3	167,834	48	0	0
bulldog and fish	4,591	5,534	8,344	0.865	6,965	48	0	0
start	75,929	77,909	86,649	2,505	78,752	48	0	0
user and Transaction	0	0,001	0,004	0,001	0,004	6	0	0
user init Transaction	0,001	0,001	0,001	0	0,001	6	0	0

HTTP 応答サマリ

HTTP 応答	合計	秒ごと
HTTP 200	4,368	3.245

使用可能なグラフとレポートの詳細については、『Mercury LoadRunner アナリシス・ユーザーズ・ガイド』を参照してください。

Windows マシンの仮想ユーザ数の引き上げ

Windows マシンの通常の設定では、仮想ユーザ数は数百に制限されています。この制限はオペレーティング・システムによるものであり、CPU またはメモリを強化しても変わりません。

Windows オペレーティング・システムの制限を回避するには、次の手順で Windows のカーネルを変更します。

- 1 変更によって問題が生じた場合に備えて、レジストリ・ファイルのコピーを保存します。
- 2 Regedit を実行します。
- 3 KEY_LOCAL_MACHINE の下にある次のキーを開きます。
System\CurrentControlSet\Control\Session Manager\SubSystems
- 4 Windows キーを選択します。NT 4.0 の標準の Windows キーは次のようになっています。

```
%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows  
SharedSection=1024,3072  
Windows=On SubSystemType=Windows ServerDll=basesrv,1  
ServerDll=winsrv:UserServerDllInitialization,3  
ServerDll=winsrv:ConServerDllInitialization,2  
ProfileControl=Off MaxRequestThreads=16
```

SharedSection=1024,3072 キーは「xxxx,yyyy」という形式です。

xxxx はシステム全体のヒープの最大サイズを定義します（単位はキロバイト）。

yyyy はデスクトップあたりのヒープのサイズを定義します。

- 5 yyyy の設定を 3072 から 8192（8 MB）に変更して、SharedSection パラメータの値を大きくします。

この設定により、1 GB の RAM と 200 MHz で動作する 2 つの Pentium PRO を搭載した Windows マシンで 1250 の Oracle 仮想ユーザを正常に実行した実績があります。

この設定では、各仮想ユーザが約 2 MB のメモリを使用しました。他の仮想ユーザは、メモリがもう少し必要なことがあります。

LoadRunner では Windows ターミナル（端末）サーバをオペレーティング・システムとして実行し、前述のようにレジストリ設定を変更した場合、2500 個以上の仮想ユーザをロードできます。

前述のレジストリ変更によって、実行できるスレッドの数が増えるのに伴って、マシンで実行できる仮想ユーザの数が増えます。つまり、ユーザ数が受ける制約は、Windows オペレーティング・システムではなく、ハードウェアと内部のスケラビリティによる制約だけだということです。

ファイアウォールのトラブルシューティング

ファイアウォールの動作に関する追加情報を提供するログ・ファイルは3つあります。

LoadRunner エージェント・ログ・ファイルには、LoadRunner エージェントと MI リスナーの間の通信に関する情報が含まれています。

- ▶ Windows マシンにあるファイルを開くには、LoadRunner エージェント・マシンのシステム・トレイにある LoadRunner エージェント・アイコンを右クリックし、[View Log] を選択します。または、最新の<一時ディレクトリ>
¥LoadRunner_agent_startup <一意の識別子> .log ファイルか（LoadRunner エージェントがプロセスである場合）、<一時ディレクトリ>
¥LoadRunner_agent_service <一意の識別子> .log ファイル（LoadRunner エージェントがサービスである場合）をテキスト・エディタで開きます。
- ▶ UNIX マシンの場合は、<一時ディレクトリ> /m_agent_daemon <一意の識別子> .log ファイルをテキスト・エディタで開きます。

MI リスナー・ログ・ファイルには、LoadRunner エージェントおよびコントローラと MI リスナーとの通信に関する情報が含まれています。

このファイルを開くには、MI リスナー・マシンのシステム・トレイにある MI リスナー・エージェント・アイコンを右クリックし、[View Log] を選択します。または、最新の<一時ディレクトリ> **¥LoadRunner_agent_startup <一意の識別子> .log** ファイルか（LoadRunner エージェントがプロセスである場合）、<一時ディレクトリ> **¥LoadRunner_agent_service <一意の識別子> .log** ファイル（LoadRunner エージェントがサービスである場合）をテキスト・エディタで開きます。

コントローラ・ログ・ファイルには、コントローラと MI リスナー間の通信処理に関する情報が含まれています。

Windows マシンの場合は、**<一時ディレクトリ> ¥drv_log.txt** ファイルをテキスト・エディタで開きます。

LoadRunner エージェントと MI リスナー間の接続の確認

LoadRunner エージェントと MI リスナーの間の接続が正常である場合は、次のことを確認できます。

- ▶ Windows プラットフォームの場合、システム・トレイにあるエージェント・アイコンの色が赤から緑に変わる。
- ▶ UNIX プラットフォームの場合、**< Local_machine_key > _connected_to_MI_Listener** というファイルが LoadRunner エージェント・マシンの一時ディレクトリに作成される。Local_machine_key は、第 17 章「ファイアウォール越しの実行」で説明したとおり、[エージェント設定] で設定された値です。このファイルは、LoadRunner エージェントが MI リスナーとの接続を解除したときに削除されます。
- ▶ UNIX と Windows のどちらのプラットフォームでも、「Notify Connected to MI Listener」というメッセージが LoadRunner エージェント・ログ・ファイルに記録されます。

注： LoadRunner エージェントは、タイムアウト秒（[エージェントの設定] で定義した値）ごとに MI リスナー・マシンに接続しようとします。接続が正常に確立された後、次のタイムアウト時間が経過するまでにこの MI リスナーを経由して LoadRunner エージェントに接続するコントローラがなかった場合は、LoadRunner によってコントローラとの接続が解除されます。Windows マシンの場合、システム・トレイにあるエージェント・アイコンの色が緑から赤に変わります。UNIX マシンでは、**< Local_machine_key > _connected_to_MI_Listener** ファイルが、LoadRunner エージェント・マシンの一時ディレクトリから削除されます。Windows と UNIX のどちらの場合でも、LoadRunner エージェントのログ・ファイルに「Disconnected from MI Listener」というメッセージが記録されます。

UNIX での接続エラー

第 17 章「ファイアウォール越しの実行」で説明したとおりに **m_agent_daemon** をインストールすると、「m_agent_daemon installed successfully」というメッセージが表示されます。

エージェント・デーモンのエラー

ERROR:File m_agent_daemon doesn't exist.

上記のメッセージが表示された場合は、実行しようとしたファイルがカレント・ディレクトリにありません。つまり、カレント・ディレクトリが**<LoadRunner のインストール先フォルダ> /bin** ディレクトリでないか、このファイルが実際に存在しないことを意味します。ファイルが存在しない場合は、インストールに問題があります。

WARNING:Could not install m_agent_daemon, reason - user < user_name > is already running m_agent_daemon on this machine.

上記の警告メッセージは、この名前のデーモンがすでに同じユーザによって実行されている場合に表示されます。

ERROR: Could not install m_agent_daemon. Check log file m_agent_daemon[xxx].log in your temp directory.

上記のエラーは、デーモンのロード時にエラーが発生したことを示します。ログ・ファイルを確認し、次に示すトラブルシューティングのヒントを参照してください。

LoadRunner エージェントのログ・ファイルのエラー

Error -10344 :Communication Error:-59961 : Failed to bind a socket while calling bind function.

Error -10344 :Communication Error:-59927 : Failed to create a TCP server for the HTTP channel's server.

Warning -29974 :Failed to create "router" server.

これらのメッセージが記録される理由は、LoadRunner エージェントが必ずポート番号 443 をリッスンするにもかかわらず（どのエージェントでも MI リスナーとして使用することができ、MI リスナーは必ずこのポートをリッスンするため）、UNIX マシンでは root 以外のユーザがこのポートを開くことができないからです。ただし、このメッセージが表示されても、ロード・ジェネレータ・マシンにこのエージェントを使用することに問題はありませぬ。

Error -10343 :Communication error:-59981 : Failed to connect to remote host - < MI リスナー名 > .

[エージェント設定] の [MI リスナー名] で設定されたマシンに接続しようとしたときに MI リスナーが実行されていません。

Error -10343 :Communication error:-59928 : Unresolved server name.

[エージェント設定] の [MI リスナー名] で指定された名前が、有効なマシンの名前、完全マシン名、または IP アドレスではないか、あるいは値が設定されていません。

Error -10343 :Communication error:-59928 : Unresolved server name .

[エージェント設定] の [プロキシの名前] で指定された名前が、有効なマシンの名前、完全マシン名、または IP アドレスではありません。

Error -10343 :Communication error:-59945 : Client failed to connect to a PROXY Server with the following settings:(-server_port= <プロキシ・サーバのポート>)(-server_fd_primary=2)(-server_type=8)(-allowed_msg_size=0)(-allowed_msgs_num=0)(-proxy_configuration_on)(-tcp_tunnel_configuration_on).

[プロキシの名前] フィールドに値が入力されていません。

Error -10343 :Communication error:-59982 : Failed to connect to remote host - <MI_Listener_Name>.The remote address is not a valid address.**Error -10343 :Communication error:-59945 : Client failed to connect to a PROXY Server with the following settings:(-server_name= <プロキシ・サーバ名>)(-server_port=<proxy_server_port>)(-server_fd_primary=2)(-server_type=8)(-allowed_msg_size=0)(-allowed_msgs_num=0)(-proxy_configuration_on)(-tcp_tunnel_configuration_on).**

[エージェント設定] で [プロキシポート] の値が間違っていて設定されています。

Error -10343 :Communication error:-59913 : NTLM authentication to proxy server error - connection to proxy refused.

プロキシ・サーバが NTLM 認証用に設定されていますが、[エージェント設定] で [プロキシユーザ名]、[プロキシパスワード] または [プロキシドメイン] が正しく設定されていません。

Error -10343 :Communication error:- 59880 : Basic authentication to proxy server error - connection to proxy refused.

プロキシ・サーバが Basic 認証用に設定されていますが、[エージェント設定] で [プロキシユーザ名] または [プロキシパスワード] が正しく設定されていません。

Error -10343 :Communication error:-59907 : SSL connect error :verify host failed :wrong DNS test.

このエラーは、[サーバ証明書を確認] 設定を [True] に設定し、MI リスナー・マシンに新しい証明書を発行していない場合に発生します（詳細については、付録 G「デジタル証明書の使い方」を参照してください）。

Error -10343 :Communication error:-59907 : SSL connect error :certificate verify failed.

Error -10343 :Communication error:-59907 : SSL connect error :ssl3 alert handshake failure.

Error -10343 :Communication error:-59907 : SSL connect error :ssl3 alert bad certificate.

Error -10343 :Communication error:-59907 : SSL connect error :ssl3 alert certificate expired.

これらのエラーは、[サーバ証明書を確認] 設定を [True] に設定すると発生します。有効な証明書の発行方法については、付録 G「デジタル証明書の使い方」を参照してください。

Error -10343 :Communication error:-59910 : SSL initialization error :Certificate not found.

Error -10343 :Communication error:-59910 : SSL initialization error :No such file or directory.

Error -10343 :Communication error:-59910 : SSL initialization error :system lib.

これらのエラーは、[エージェント設定] で [Client Certificate owner] の設定を [True] に設定し、LoadRunner エージェント・マシンに証明書がインストールされていない場合に発生します（詳細については、付録 G「デジタル証明書の使い方」を参照してください）。

MI リスナーのログ・ファイルのエラー

Error - 10344 :Communication Error:-59961 : Failed to bind a socket while calling bind function.

Error -10344 :Communication Error:-59927 : Failed to create a TCP server for the HTTP channel's server.

Warning -29974 :Failed to create "router" server.

これらのエラーは、MI リスナー・マシンで別のプロセス（IIS サービスなど）がポート 443 を占有していることを示します。

Error -10343 :Communication error:-59904 : SSL accept error :sslv3 alert certificate expired.

このエラーは、[サーバ証明書を確認] 設定を [True] に設定し、MI リスナーの証明書が期限切れである場合に発生します。

Error -10343 :Communication error:-59904 : SSL accept error :sslv3 alert bad certificate.

このエラーは、[サーバ証明書を確認] 設定を [True] に設定し、次のいずれかに該当する場合に発生します。

- ▶ MI リスナーの証明書に、LoadRunner エージェントの CA リストに含まれる署名がない。
- ▶ MI リスナーの証明書の有効期限の開始日に到達していない。

有効な証明書の発行方法、CA リストへの認証局の追加方法、新しい有効期限を持つ証明書の作成方法については、付録 G「デジタル証明書の使い方」を参照してください。

Error -10343 :Communication error:-59904 : SSL accept error :peer did not return a certificate.

このエラーは、[MI リスナーの設定] で [クライアント証明のチェック] 設定が [True] に設定されているのに対し、[エージェント設定] で [Client Certificate owner] 設定が [False] に設定されていることを示します。

Error -10343 :Communication error:-59904 : SSL accept error :no certificate returned.

このエラーは、[MI リスナーの設定] で [クライアント証明のチェック] 設定が [True] に設定されているのに対し、[エージェントの設定] で [Client Certificate owner] 設定が [True] に設定されていて、以下のいずれかに該当することを示します。

- ▶ LoadRunner エージェントの証明書に、MI リスナーの CA リストに含まれる署名がない。
- ▶ LoadRunner エージェントの証明書の有効期限の開始日に到達していない。

有効な証明書の発行方法、CA リストへの認証局の追加方法、新しい有効期限を持つ証明書の作成方法については、付録 G「デジタル証明書の使い方」を参照してください。

Error -10343 :Communication error:-59904 : SSL accept error :no certificate returned.

このエラーは、[MI リスナーの設定] で [クライアント証明のチェック] が [True] に設定されているのに対し、[エージェント設定] で [Client Certificate owner] が [True] に設定されているが、LoadRunner エージェントの証明書が期限切れであることを示します。

一般接続エラー

これらのエラーは、すべての設定を使用した場合に発生します。

LoadRunner エージェントのログと MI リスナーのログにエラーがないにもかかわらず、エージェントが MI リスナーに接続しない場合は、LoadRunner エージェント・マシンにある **<LoadRunner のインストール先フォルダ>** `%dat%br_Inch_server.cfg` ファイルの Firewall セクションで `FireWallServiceActive` 属性を 1 に設定します。

MI リスナーを経由するコントローラとエージェントの間の接続の確認

LoadRunner エージェントと MI リスナーの間の接続が正常に確立されたにもかかわらず、コントローラ・マシンからの接続に失敗した場合は、次を確認します。

- ▶ コントローラの [ロード ジェネレータ] ダイアログの [名前] フィールドの値が [エージェント設定] の [ローカル マシン キー] で設定された名前と一致する。
- ▶ 上記のホストの [ロード ジェネレータ] > [詳細] > [ファイアウォール] タブの [MI リスナー] フィールドの値が [エージェント設定] の [MI リスナー名] で設定した名前と一致する。
- ▶ ファイアウォールによって通信速度が低下することも考えられるため、コントローラの [ツール] メニューから、[オプション] > [タイムアウト] タブを選択し、[ロード ジェネレータ] の接続タイムアウトの値を大きくする必要があるかどうか確認する。
- ▶ コントローラ・マシンが LoadRunner エージェント・マシンを認識することを確認する (ping ユーティリティなどを使用)。コントローラ・マシンが LoadRunner エージェント・マシンを認識しない場合は、LoadRunner に関連しないシステムで設定上の問題があります。この問題は、接続を行う前に解決しておく必要があります。
- ▶ MI リスナー・マシンのポート 50500 を調べて、コントローラが MI リスナーに正常に接続されていることを確認する (MI リスナー・マシンで netstat ユーティリティを使用する)。

LoadRunner エージェントを使った作業

LoadRunner エージェントはロード・ジェネレータ・マシンで実行され、コントローラ、ロード・ジェネレータ、MI リスナー間で通信できるようにします（ファイアウォール設定を使用）。エージェントは、コントローラから、仮想ユーザの初期化、実行、一時停止、停止に関する指示を受け取ります。同時にエージェントは、仮想ユーザのステータスのデータをコントローラに返します。

エージェントの現在の設定を確認するには、タスク・バー領域でマウスをエージェントのアイコンの上に移動し、説明を読みます。説明には、「LoadRunner エージェント・プロセス」または「LoadRunner エージェント・サービス」のどちらかが書かれています。

LoadRunner エージェントをプロセスとして実行

SAPGUI の再生、リモート・マシンでの GUI 仮想ユーザの実行、端末セッションの実行など、場合によっては、LoadRunner エージェントをプロセスとして実行する必要があります。

LoadRunner エージェントをサービスからプロセスに変更するには、次の手順で行います。

- 1 サービスをアンインストールするには、< LR インストール・フォルダ > **¥launch_service¥bin¥magentservice.exe -remove** を実行します。
- 2 プロセスをインストールするには、< LR のインストール・フォルダ > **¥launch_service¥bin¥magentproc.exe** のショートカットを作成し、システムのスタートアップ・フォルダに配置します。
- 3 スタートアップ・フォルダまたは < LR のインストール・フォルダ > **¥launch_service¥bin** から **magentproc.exe** を実行するか、マシンを再起動して、プロセスを実行します。

LoadRunner エージェントをサービスとして実行

ほとんどの場合、LoadRunner エージェントはサービスとして実行されます。

LoadRunner エージェントをプロセスからサービスに変更するには、次の手順を実行します。

- 1 システム・トレイでエージェント・プロセスのアイコンを右クリックし、メニューで [**Close**] を選択して停止します。

- 2 システムのスタートアップ・フォルダからプロセスのショートカットを削除して、プロセスをアンインストールします。
- 3 < LR のインストール・フォルダ > %launch_service%bin%magentservice.exe -install <ユーザ・ドメイン> %<ユーザ名> <パスワード> を実行して、サービスをインストールします。

このコマンドを実行すると、エージェント・サービスも実行されます。

[Services] ダイアログ・ボックスで、LoadRunner エージェント・サービスが **ローカル・システム** という名前で実行され、デスクトップと対話できるようになります。ただし、引き続き提供されたユーザの証明書を使用します。エージェントのすべての子プロセス (**lr_bridge.exe** や **mdrv.exe** など) は、提供されたユーザの証明書を使用して実行されます。

- 4 サービスを停止した後でサービスを実行するには、[Services] ダイアログ・ボックスからサービスを起動します。

LoadRunner エージェントのパスワード変更

LoadRunner エージェント・サービスのパスワードを変更するには、まずサービスをアンインストールし、次にサービスを新しいユーザ名とパスワードで再インストールする必要があります。

- 1 サービスをアンインストールするには、< LR のインストール・フォルダ > %launch_service%bin%magentservice.exe -remove を実行します。
- 2 サービスを再インストールするには、< LR のインストール・フォルダ > %launch_service%bin%magentservice.exe -install <ユーザ・ドメイン> %<ユーザ名> <パスワード> を実行します。

このコマンドを実行すると、エージェント・サービスも実行されます。

ユーザ名またはパスワードは [Services] ダイアログ・ボックスで変更しないでください。

エージェントをサービスとして実行する場合のネットワーク・ドライブの割り当て

Windows プラットフォームでは、ユーザがログアウトしているとき、サービスはネットワーク・ドライブの割り当てを解決できません。Windows XP では、ユーザがログインしているときも、サービスはネットワーク・ドライブの割り当てを解決できません。

サービスがマップされたネットワーク・ドライブで動作できない場合、ディレクトリへのフル・パスを使用します。

例 : < ¥¥ <マシン名> ¥ <ディレクトリ> ¥ >

付録 E

マルチ IP アドレスの設定

シナリオ実行時には、各ロード・ジェネレータ・マシン上の仮想ユーザは、そのホストの IP アドレスを使います。1 台のロード・ジェネレータ・マシン上で複数の IP アドレスを定義して、ユーザが複数の異なるマシンから操作を行うという現時的な状況をエミュレートできます。

この付録では、次の項目について説明します。

- ▶ マルチ IP アドレスについて
- ▶ ロード・ジェネレータへの IP アドレスの追加
- ▶ IP ウィザードの使用
- ▶ UNIX での複数 IP アドレスの設定
- ▶ ルーティング・テーブルの更新
- ▶ コントローラでの複数 IP アドレス機能の有効化

マルチ IP アドレスについて

アプリケーション・サーバとネットワーク・デバイスは、クライアントの識別に IP アドレスを使います。多くの場合、アプリケーション・サーバは同じマシンから接続してくるクライアントに関する情報をキャッシュします。ネットワーク・ルータは、スループット向上のために発信側と受信側の情報をキャッシュしようとしています。このため多数のユーザが同じ IP アドレスを持っていると、サーバとルータの両方が最適化しようとしています。同じロード・ジェネレータ・マシン上の仮想ユーザはすべて同じ IP アドレスを持つことになるので、サーバとルータの最適化はどちらも現実の状況を正確に反映しているとは言えません。

LoadRunner のマルチ IP アドレス機能により、1 台のマシンで実行されている複数の仮想ユーザをその IP アドレスで特定できます。これによりサーバとルータは、複数の仮想ユーザが別々のマシンから接続してきていると認識するため、テスト環境はより現実に近いものとなります。

注：ネットワークカードごとにスプーフィングできる IP アドレス数の上限は、Windows NT SP3 では 35 個、Solaris（バージョン 2.5.1）では 255 個、Solaris（version 2.6 以上）では 8192 個までです。

適用可能なプロトコル

マルチ IP アドレス機能は、以下のプロトコルに対して適用できます。

- ▶ **クライアント / サーバ** : DNS, Windows Sockets
- ▶ **ユーザ定義** : Javascript Vuser, VB Vuser, VB Script Vuser
- ▶ **E ビジネス** : FTP, Palm, SOAP, Web (HTTP/HTML), Web サービス, WinSock/Web Dual プロトコル
- ▶ **ERP/CRM** : Oracle NCA, Oracle Web Applications 11i, PeopleSoft Enterprise, SAP-Web, Siebel-Web
- ▶ **Legacy**: RTE
- ▶ **メール・サービス** : インターネット・メッセージング (IMAP), POP3, および SMTP
- ▶ **ストリーミング・データ** : Real
- ▶ **ワイヤレス** : iモード, VoiceXML, WAP プロトコル

この機能は Windows および UNIX プラットフォームで実装できます。

ロード・ジェネレータへの IP アドレスの追加

LoadRunner には IP ウィザードというプログラムが含まれています。このプログラムを、複数の IP アドレスを作成する Windows NT または Windows 2000 ロード・ジェネレータ・マシンごとに実行します。1 台のマシンに新しい IP アドレスを一度追加して、すべてのシナリオに対してそのアドレスを使います。

UNIX マシンへの IP アドレスの追加については、433 ページ「UNIX での複数 IP アドレスの設定」を参照してください。

次に、ロード・ジェネレータに新しい IP アドレスを追加する手順を簡単にまとめます。

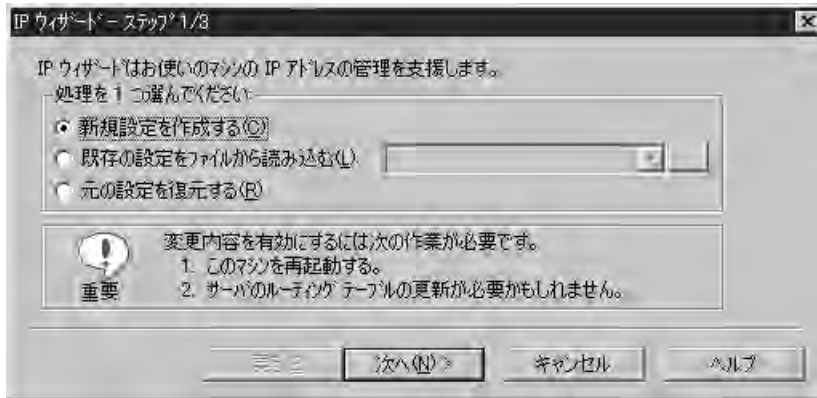
- 1 対象のロード・ジェネレータ・マシン上で IP ウィザードを実行して、指定した数の IP アドレスを追加します。UNIX ロード・ジェネレータ・マシンでは、手作業で新しい IP アドレスを設定します。
- 2 マシンを再起動します。
- 3 必要であれば、サーバのルーティング・テーブルを新しいアドレスで更新します。
- 4 コントローラからこの機能を有効にします。詳細については、436 ページ「コントローラでの複数 IP アドレス機能の有効化」を参照してください。

IP ウィザードの使用

IP ウィザードは各ロード・ジェネレータ・マシンにあります。この手続きを 1 回実行すれば、新しい IP アドレスが Windows マシンに作成、保存されます。新しいアドレスには、IANA (Internet Assignment Numbers Authority) が定めた範囲にあるアドレスを使用できます。これらのアドレスは内部での使用に限定され、インターネットへの接続には使用できません。アドレス範囲は、IP ウィザードによって標準で使用される範囲です。

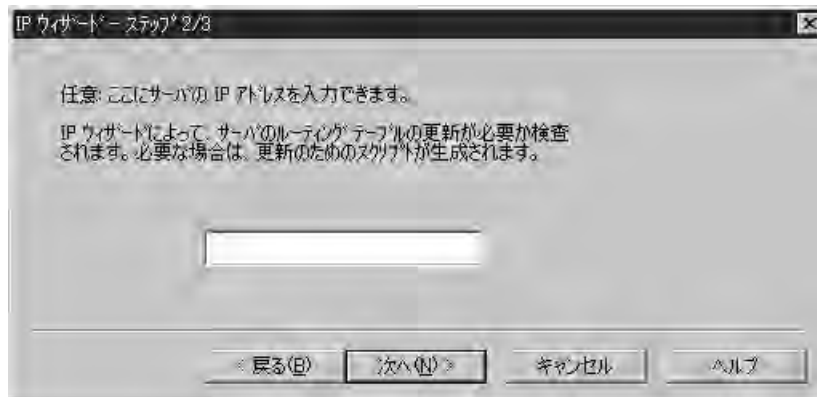
ロード・ジェネレータ・マシンに新しい IP アドレスを追加するには、次の手順で行います。

- 1 LoadRunner プログラム・グループから [IP ウィザード] を起動します。

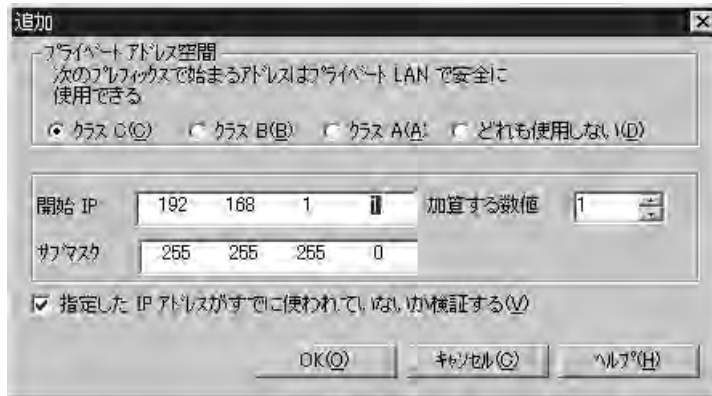


- 2 IP アドレスを設定した既存のファイルがある場合、**[既存の設定をファイルから読み込む]** を選択してからファイルを選びます。
- 3 新しい設定を定義する場合は、**[新規設定を作成する]** を選択します。
- 4 **[次へ]** をクリックして次のステップに進みます。複数のネットワーク・カードがある場合は、指定した IP アドレスを使うカードを選び、**[次へ]** をクリックします。

任意のステップである Web サーバ IP アドレスのステップでは、IP ウィザードでサーバのルーティング・テーブルを検査することによって、ロード・ジェネレータに新しい IP アドレスを追加した後にテーブルを更新する必要があるかどうかを調べることができます。



- 5 アドレスの追加直後にサーバのルーティング・テーブルを検査するには、サーバの IP アドレスを入力します。詳細については、435 ページ「ルーティング・テーブルの更新」を参照してください。
- 6 **[次へ]** をクリックするとマシンの IP アドレスのリストが表示されます。**[追加]** をクリックしてアドレスの範囲を定義します。



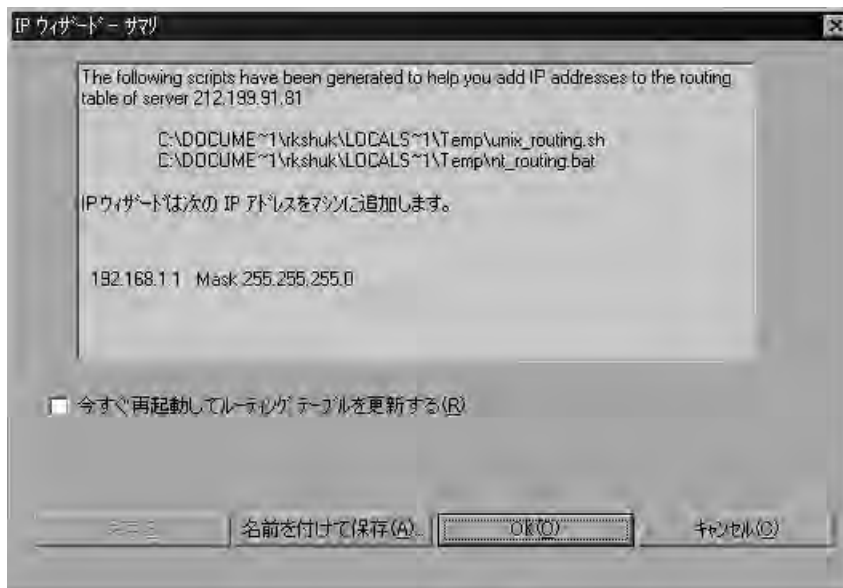
IP アドレスは、**netid** と **hostid** という 2 つの要素で構成されています。アドレスのうちどこまでが **netid** でどこからが **hostid** であるかは、サブマスクによって決まります。

- 7 マシンの IP アドレスに対して正しいサブマスクを表しているクラスを選択します。
- 8 作成するアドレスの数を指定します。**[指定した IP アドレスがすでに使われていないか検証する]** を選択すると、IP ウィザードに新しいアドレスを検査するように指示できます。IP ウィザードによって未使用のアドレスだけが追加されます。
[OK] をクリックして先に進みます。

- 9 IP ウィザードによって新しいアドレスが作成され、概要情報ダイアログ・ボックスが開いて、すべての IP アドレスのリストが表示されます。



- 10 [完了] をクリックして、[IP ウィザード] を終了します。[IP ウィザード - サマリ] ダイアログ・ボックスが表示されます。



- 11 **.bat** ファイルのアドレスに注目してください。ルーティング・テーブルを更新するためのバッチ・ファイルの用法については、435 ページ「ルーティング・テーブルの更新」を参照してください。
- 12 ルーティング・テーブルを更新したら、**[今すぐ再起動してルーティング テーブルを更新する]** をチェックして、新しいアドレスで NT のデバイス・ドライバを初期化します。
- 13 **[OK]** をクリックします。

UNIX での複数 IP アドレスの設定

UNIX で複数の IP アドレスを設定するには、手作業でロード・ジェネレータ・マシンのアドレスを設定します。

- ▶ Solaris 2.5, 2.6, 7.0, 8.0
- ▶ Linux
- ▶ HP 11.0 以上
- ▶ IBM AIX

Solaris 2.5, 2.6, 7.0, 8.0

hme0 デバイスを設定して複数の IP アドレスをサポートするには、次の手順で行います。

- 1 次に示すように、物理的なマシンの `/etc/hosts` に、ホスト名ごとにエントリを作成します。

```
128.195.10.31 myhost
128.195.10.46 myhost2
128.195.10.78 myhost3
```

仮想ホスト「n」のホスト名を含む `/etc/hostname.hme0:n` ファイルを作成します。

注 : `hostname.hme0:0` は、`hostname.hme0` と同じになるので注意してください。

/etc/hostname.hme0 (名前 myhost を含む)

/etc/hostname.hme0:1 (名前 myhost2 を含む)

/etc/hostname.hme0:2 (名前 myhost3 を含む)

上記の変更により、ブート時に仮想ホストが設定されます。

- 2 また、**hme0:n** 形式の命名規則を使って、論理ホストの1つで直接 **ifconfig** を実行することによって、論理ホストの設定を直接有効にしたり変更したりできます。

```
% ifconfig hme0:1 up
% ifconfig hme0:1 129.153.76.72
% ifconfig hme0:1 down
```

現在の設定を検証するには、**ifconfig -a** を使用します。

Linux

1 枚の Ethernet カードに対して複数の IP アドレスを定義するには、IP Aliasing を組み込んでカーネルをコンパイルしておく必要があります。このためには、**ifconfig** コマンドを使用します。

```
/sbin/ifconfig eth0:0 x.x.x.x netmask 255.255.x.x up
```

x.x.x.x に新しい IP アドレスを指定し、正しいサブネット・マスクを指定します。起動時に実行されるように、このコマンドを **rc.local** ファイルに入れます。

HP 11.0 以上

1 枚の Ethernet カードに対して複数の IP アドレスを定義するには、IP Aliasing を組み込んでカーネルをコンパイルしておく必要があります。このためには、**ifconfig** コマンドを使用します。

```
/sbin/ifconfig lan1:0 x.x.x.x netmask 255.255.x.x up
```

x.x.x.x に新しい IP アドレスを指定し、正しいサブネット・マスクを指定します。起動時に実行されるように、このコマンドを **rc.local** ファイルに入れます。

IBM AIX

1 枚の Ethernet カードに対して複数の IP アドレスを定義するには、IP Aliasing を組み込んでカーネルをコンパイルしておく必要があります。このためには、**ifconfig** コマンドを使用します。

```
/usr/sbin/ifconfig [int] [ip address] alias netmask [mask]
```

例えば、IP アドレス **10.0.0.1** をメイン・インタフェースに追加する場合は、**root** として下記を実行する必要があります。

```
/usr/sbin/ifconfig ne0 10.0.0.1 alias netmask 255.255.255.0
```

この行が起動時に実行されるようにするには、適切な実行レベル (**/etc/rc.d/rc#.d**) に標準スクリプトを作成します。

ルーティング・テーブルの更新

クライアント・マシンに新しい IP アドレスを追加したら、サーバ側でもクライアントまでのルートを確認できるように、サーバのルーティング・テーブルにそのアドレスを登録する必要があります。サーバとクライアントが同じネットワークマスク、IP クラス、ネットワークを使っている場合は、サーバのルーティング・テーブルを変更する必要はありません。

注：クライアントとサーバ・マシンの間にルータがある場合、サーバはルータを経由するパスを確認する必要があります。次の項目がサーバのルーティング・テーブルに追加してあることを確認してください：

Web サーバからルータへの経路、およびルータからロード・ジェネレータ・マシンのすべての IP アドレスへの経路。

Web サーバのルーティング・テーブルを更新するには、次の手順で行います。

- 1 [IP ウィザード-サマリ] 画面に表示されたバッチ・ファイルを編集します。
.bat ファイルのサンプルを次に示します。

```
REM This is a bat file to add IP addresses to the routing table of a
server
REM Replace [CLIENT_IP] with the IP of this machine that the server
already recognizes
REM This script should be executed on the server machine

route ADD 192.168.1.50 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
route ADD 192.168.1.51 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
route ADD 192.168.1.52 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
route ADD 192.168.1.53 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
route ADD 192.168.1.54 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
```

- 2 [CLIENT_IP] とある箇所に、IP アドレスを挿入します。
- 3 サーバ・マシンでバッチ・ファイルを実行します。

コントローラでの複数 IP アドレス機能の有効化

複数の IP アドレスを定義した後は、コントローラでこの機能を使うオプションを設定します。

コントローラからマルチ IP アドレス機能を有効にするには、次の手順で行います。

- 1 コントローラの [デザイン] ビュー・ウィンドウで [シナリオ] > [IP スプーファを有効にする] を選択します。

注：ロード・ジェネレータに接続する前にこのオプションを選択してください。

- 2 コントローラのエキスパート・モードで [一般] タブを使って、この機能の動作方法を指定します。

詳細については、付録 C 「エキスパート・モードでの作業」を参照してください。

付録 F

コントローラのコマンド・ライン引数

コマンド・ラインからコントローラを起動するときに、引数を渡してコントローラの動作を指定することができます。コマンド・ラインで引数を渡せば、コントローラ・シナリオの設定を手作業でコントローラの UI から行う必要がありません。

この付録では、次の項目について説明します。

- ▶ コントローラのコマンド・ライン引数について
- ▶ コマンド・ラインからのコントローラの起動
- ▶ Quality Center の引数
- ▶ 実行時環境引数

コントローラのコマンド・ライン引数について

コントローラを起動すると、コントローラに渡したすべての引数が確認され、引数に従って起動環境が設定されます。引数を渡さなかった場合は、コントローラの標準の設定が使用されます。

例えば、起動時に Quality Center に接続し、結果をシナリオで定義されているディレクトリとは別のディレクトリに保存し、シナリオ終了時にアナリシスを起動するようにコントローラに指示できます。

コマンド・ラインからのコントローラの起動

コントローラを起動するには、コマンド・ラインに `wlrun` と入力し、その後ろに引数を続けます。各引数の先頭にはダッシュ記号を指定します。

注：引数の大文字と小文字は区別されます。

次に例を示します。

```
wlrun -TestPath C:\LoadRunner\scenario\Scenario.lrs -Run
```

コマンド・ラインからコントロールを起動するとき、次の規則が適用されます。

- ▶ 引数を指定せずにコマンド・ラインからコントローラを起動した場合、コントローラは標準の設定を使用する。
- ▶ コントローラは、必ず結果を上書きする。
- ▶ コントローラはシナリオ終了時に自動的に終了し、結果が照合される。シナリオの終了時にコントローラが自動的に終了しないようにするには、コマンド・ラインに **-DontClose** というフラグを追加します。
- ▶ コマンド・ラインから起動したコントローラは、**-Run** オプションを指定した場合を除き、通常どおりに動作します。**-Run** オプションを指定してコマンド・ラインから起動すると、起動時に通常開くダイアログ・ボックスとメッセージ・ボックスは開かないので、閉じるための操作は必要なくなります。
- ▶ コントローラの設定は Windows ディレクトリに格納されている `wlrun5.ini` からロードされます。

Quality Center の引数

Quality Center 引数は、LoadRunner と Quality Center の統合を定義します。LoadRunner と Quality Center の統合に関する詳細については、第12章「Quality Center を使ったシナリオ管理」を参照してください。

ConnectToQC	コントローラを起動時に Quality Center に接続するかどうかを指定します (0/1 または ON/OFF)。
QCServer	Quality Center サーバ名。Quality Center がインストールされているマシンである必要があります。
QCDB	Quality Center データベース名。次の形式を使用します。 " <ドメイン名> . <プロジェクト名> "

UserName	Quality Center に接続するユーザ名。
Password	ユーザ名に対応するパスワード。
TestPath	Quality Center データベース内のシナリオのパス。次に例を示します。 "[TD]¥Subject¥LoadRunner¥Scenario1" パスにスペースが含まれる場合は、パスを引用符で囲みます。
TestId	テスト ID (Quality Center にだけ使用される)。
ResultCleanName	ResultCycle と併せて使用します。例えば、「Res1」のように指定します。
ResultCycle	Quality Center サイクル。例えば、「LR_60_SP1_247」のように指定します。 注： ResultCycle 引数と ResultCleanName 引数は、Quality Center データベースに結果を保存する場合に必要です。

実行時環境引数

実行時環境引数は、実行時の環境に関連するシナリオ設定を指定します。シナリオ設定の詳細については、第 11 章「シナリオの実行準備」を参照してください。

TestPath	シナリオのパス。 例： C:¥LoadRunner¥scenario¥Scenario.lrs この引数は、Quality Center データベースに格納されているシナリオにも使用できます。次に例を示します。 "[TD]¥Subject¥LoadRunner¥Scenario1" パスにスペースが含まれる場合は、パスを引用符で囲みます。
Run	シナリオを実行し、すべての出力メッセージを res_dir¥output.txt に出力し、コントローラを終了します。
InvokeAnalysis	シナリオ終了時にアナリシスを起動するよう LoadRunner に指示します。この引数を指定しなかった場合、シナリオの標準の設定が LoadRunner によって使用されます。
ResultName	結果の完全パス。例えば、「C:¥Temp¥Res_01」のように指定します。

ResultCleanName	結果の名前。例えば、「Res_01」のように指定します。
ResultLocation	結果ディレクトリ。例えば、「C:¥Temp」のように指定します。

注：シナリオに結果ディレクトリが指定されておらず，結果指定引数を渡さなかった場合，シナリオが実行されません。

付録 G

デジタル証明書の使い方

デジタル証明書は、Web 上での商取引やそれ以外のやり取りを行うときに信用を証明する電子的な「信用証明」です。デジタル証明書は、認証局（CA）によって発行されます。デジタル証明書には、証明書が発行されたマシンの IP アドレス、有効期限、認証局のデジタル署名が含まれます。

この付録では、次の項目について説明します。

- ▶ ファイアウォールでのデジタル証明書の使用
- ▶ デジタル証明書の作成と使用

ファイアウォールでのデジタル証明書の使用

MI リスナーによって公開鍵が LoadRunner エージェントに送られるとき、リスナーの証明書も必ず送られます（サーバ・サイド証明書）。LoadRunner エージェントは、第 17 章「ファイアウォール越しの実行」で説明したように、受け取った証明書を認証するように設定できます。エージェントが証明書を認証するように設定されている場合は、次のことを行って、送信元が本物であるかどうかを確認できます。

- ▶ 証明書の IP アドレスと送信元の IP アドレスを比較する。
- ▶ 有効期限を確認する。
- ▶ 証明書に含まれている認証局のリストの中でデジタル署名を探す。

MI リスナーでは、セッションの任意の時点で LoadRunner エージェントから証明書を送信する必要がある場合もあります。これは、第 17 章「ファイアウォール越しの実行」の MI リスナーの設定方法で示したように、クライアント・サイド証明書と呼ばれます。LoadRunner エージェントが証明書を所有している場合は、同じ認証プロセスで MI リスナーに証明書が送信されます。LoadRunner エージェントが証明書を所有しない場合は、通信を継続できない場合があります。

LoadRunner のすべてのインストールに、SSL CA リストと SSL 証明書が含まれています。この証明書は、すべての LoadRunner のインストールで同じです。したがって、第三者が同じ証明書を入手できます。したがって、安全性を高めたい場合は、認証局を独自に作成し、リストに含め、対象とするマシンに対応する証明書を発行します。

デジタル証明書の作成と使用

認証局の作成には、`gen_ca_cert.exe` (UNIX プラットフォームでは `gen_ca_cert`) ユーティリティを使用し、デジタル証明書の作成には、`gen_cert.exe` (UNIX プラットフォームでは `gen_cert`) を使用します。どちらのユーティリティも UNIX と Windows プラットフォームのコマンド・ライン・インタフェースから使用できます。

`gen_ca_cert` を使用して認証局を作成するには、次の手順で行います。

- 1 コマンド・ラインの形式と使用方法を表示するには、**< LoadRunner のインストール先フォルダ > %launch_service%bin** ディレクトリから `gen_ca_cert` ユーティリティを実行します。

```

C:\Program Files\Mercury Interactive\LoadRunner\launch_service\bin>gen_ca_cert.exe
Usage: gen_ca_cert [option] <value> ...

  -install           <File Name to install>
OR
  -install_add      <File Name to install>
OR
  -country_name     <Country Name>
  -organization_name <Organization Name>
  -common_name      <common name>
  -CA_cert_file_name <CA Certificate File Name, Default: cacert.cert>
  -CA_pk_file_name  <CA Private Key File Name, Default: capvk.cert>
  -nb_time          <Valid Not Before Date <day/month/year> >
  -na_time          <Valid Not After Date <day/month/year> >

```

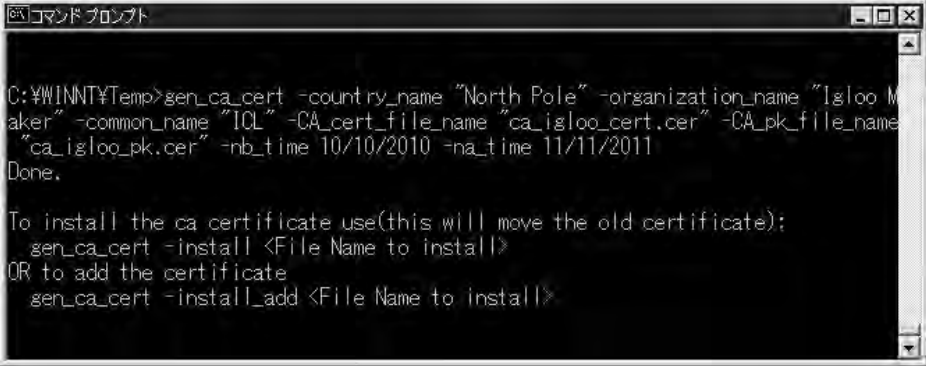
- 2 `gen_ca_cert` コマンドにオプションを少なくとも1つ指定して実行すると、新しい認証局が作成されます：

`-country_name` <国名> `-organization_name` <組織名> および `-common_name` <CA の名前>

これにより、ユーティリティが実行されたディレクトリに CA 証明書 (`cacert.cer`) と CA 秘密鍵 (`capvk.cer`) の2つのファイルが作成されます。別のファイル名にするには、`-CA_cert_file_name` オプションと `-CA_pk_file_name` オプションをそれぞれのファイルに対して使用してください。

標準設定では、CA は生成されてから3年間有効です。有効期限を変更するには、オプション `-nb_time` <dd/mm/yyyy 形式での有効期限の開始日> または `-na_time` <dd/mm/yyyy 形式での有効期限の終了日> を使用します。

次の例では、カレント・ディレクトリに `ca_igloo_cert.cer` と `ca_igloo_pk.cer` の2つのファイルが作成されます。



```

C:\WINNT\Temp>gen_ca_cert -country_name "North Pole" -organization_name "Igloo M
aker" -common_name "ICL" -CA_cert_file_name "ca_igloo_cert.cer" -CA_pk_file_name
"ca_igloo_pk.cer" -nb_time 10/10/2010 -na_time 11/11/2011
Done.

To install the ca certificate use(this will move the old certificate):
  gen_ca_cert -install <File Name to install>
OR to add the certificate
  gen_ca_cert -install_add <File Name to install>

```

- 3 この CA をインストールするには、`-install` <証明書ファイルの名前> オプションを使用します。このオプションを実行すると、既存のすべての CA リストが削除され、この CA だけを含む新しい CA リストが作成されます。

既存の CA リストに新しい CA を追加するには、`-install_add` <証明書ファイルの名前> を使用します。



```

C:\WINNT\Temp>gen_ca_cert -install "ca_igloo_cert.cer"
ca_igloo_cert.cer is installed.

```

- 4 `-install` と `-install_add` オプションを指定すると、証明書ファイルだけがインストールされます。秘密鍵ファイルは安全な場所に保管し、証明書を発行する場合にのみ使用してください。

`gen_cert` を使用してデジタル証明書を作成するには、次の手順で行います。

- 1 コマンド・ラインの形式と使用方法を表示するには、**< LoadRunner のインストール先フォルダ >** `%launch_service%bin` ディレクトリから `gen_cert` ユーティリティを実行します。

```

C:\WINNT\Temp>gen_cert
Usage: gen_cert [option] <value> ...

-install          <File Name to install>
OR
-common_name     <full_DNS_server_name>
-country_name    <Country Name>
-organization_name <Organization Name>
-organizationUnit_name <Organization Unit Name>
-eMail           <Email>
-cert_file_name  <output Certificate File Name, Default: cert.cert>
-CA_cert_file_name <CA Certificate File Name, Default: cacert.cert>
-CA_pk_file_name <CA Private Key File Name, Default: capvk.cert>
-nb_time         <Valid Not Before Date <day/month/year> >
-na_time         <Valid Not After Date <day/month/year> >

```

- 2 `gen_ca_cert` コマンドに、次のオプションの少なくとも1つを指定して実行し、新しいデジタル証明書を作成します：

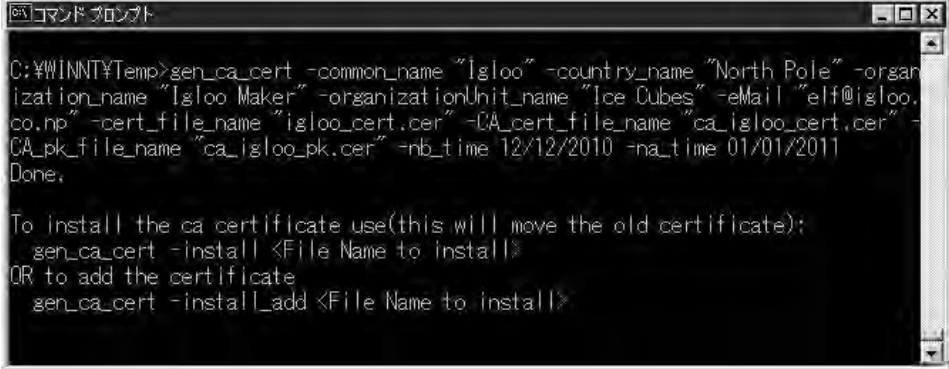
`-country_name` <国名>, `-organization_name` <組織名>, `-organization_unit_name` <部署名>, `-eMail` <電子メール/アドレス>および `-common_name` <マシンの名前, 完全名, または IP アドレス>

CA 証明書ファイルと CA 秘密鍵ファイルは、証明書の作成に必要です。標準設定では、これらのファイルはカレント・ディレクトリに作成され、それぞれ `cacert.cert` および `capvk.cert` という名前が与えられます。ファイル名と格納場所が標準設定以外の場合は、`-CA_cert_file_name` オプションと `-CA_pk_file_name` オプションを使用します。

この結果、ユーティリティを実行したディレクトリに証明書ファイルが作成されます。標準設定では、ファイル名は `cert.cert` です。別のファイル名を与えるには、`-cert_file_name` オプションを使用します。

標準設定では、CA は生成されてから 3 年間有効です。有効期限を変更するには、オプション `-nb_time < dd/mm/yyyy 形式での有効期限の開始日 >` または `-na_time < dd/mm/yyyy 形式での有効期限の終了日 >` を使用します。

次の例では、カレント・ディレクトリに **igloo_cert.cer** というファイルが作成されます。



```
コマンドプロンプト
C:\¥WINNT¥Temp>gen_ca_cert -common_name "Igloo" -country_name "North Pole" -organization_name "Igloo Maker" -organizationUnit_name "Ice Cubes" -eMail "elf@igloo.co.jp" -cert_file_name "igloo_cert.cer" -CA_cert_file_name "ca_igloo_cert.cer" -CA_pk_file_name "ca_igloo_pk.cer" -nb_time 12/12/2010 -na_time 01/01/2011
Done.

To install the ca certificate use(this will move the old certificate):
gen_ca_cert -install <File Name to install>
OR to add the certificate
gen_ca_cert -install_add <File Name to install>
```

- 3 この証明書をインストールするには、`-install <証明書ファイルの名前>` オプションを使用します。1 台のマシンが所有できる証明書は 1 つだけなので、このオプションを使用すると、既存の証明書が破棄されます。

索引

C

CA 441
controller_host 389
controller_path 389

D

declare_rendezvous 関数 212
declare_transaction 関数 212

E

end_transaction 関数 212
ERP/CRM 診断 291
 セットアップ 295
 説明 291
 ブレークダウンの配分 300
 メディアータ 294
 モジュールのアーキテクチャ 293
 モジュールの種類 292
 有効化 300
ERP/CRM メディアータ 294
error_message 関数 212, 213

G

get_host_name 関数 214
get_master_host_name 関数 214
GUI 仮想ユーザ・スクリプト
 GUI 関数の使用方法 212
 LoadRunner との統合 203
 QuickTest での作成 206
 WinRunner での作成 209
GUI 仮想ユーザ, 定義 9

H

hme0 デバイス 433
hostid, IP アドレス・コンポーネント 431
Hostinfo ユーティリティ 404

hosts ファイル 405
HP, IP アドレスの設定 434

I

IP アドレス
 hostid 431
 HP での設定 434
 IP ウィザード 429
 Linux での設定 434
 netid 431
 Solaris での設定 433
 クラス 431
 コントローラからの有効化 436
 サブマスク 431
 複数の設定 427-436
 ロード・ジェネレータへの追加 428
 ロード・ジェネレータ・マシン 427

L

Linux
 IP アドレスの設定 434
LoadRunner
 LoadRunner を使った作業 6
 アプリケーション・テスト 3
 概要 3-15
 仮想ユーザを使った実ユーザのエミュレート 5
 導入計画 26
LoadRunner アナリシス・ユーザーズ・ガイド xiii
LoadRunner インストール・ガイド xiii
LoadRunner コントローラ・ユーザーズ・ガイド xiii
LoadRunner
 テスト工程, 「テスト工程」参照
LoadRunner モニタ・リファレンス xiii
lr_whoami 関数 (GUI 仮想ユーザ) 212

M

Mercury Interactive の Web サイト xv
MI リスナー 274

N

netid, IP アドレス・コンポーネント 431

O

Oracle 11i 診断

診断パスワードの設定 324
323

コントローラでの設定 326

診断パスワード要求の無効化 325

トレース診断の有効化 323

トレース・ファイルのサイズの設定
324

ログの有効化 323

Oracle サーバ設定ダイアログ・ボックス 328

Oracle 設定ダイアログ・ボックス 327

output_message 関数 212

Q

Quality Center

Quality Center への結果の保存 199

Quality Center へのシナリオの保存 197

仮想ユーザ・スクリプトの追加 200

コマンド・ライン引数 438

シナリオを開く 195

スクリプトの管理 191

接続 192

切断 193

統合 191-202

Quality Center からの切断 193

Quality Center プロジェクトからシナリオを開
くダイアログ・ボックス 195

Quality Center プロジェクトへシナリオを保存
ダイアログ・ボックス 197

Quality Center への接続ダイアログ・ボックス
192

QuickTest

QuickTest を使用した GUI スクリプトの
作成 206

QuickTest, LoadRunner への 206

QuickTest, テストでの統合データの保存 207

R

remote_host 390

remote_path 389

rsh

UNIX で使用せずに実行 407

コントローラとの接続の確認 406

RTE 仮想ユーザ・スクリプト

コントローラ内の 12

S

SAP 設定ダイアログ・ボックス 333

SAP 診断

コントローラでの設定 332

Siebel DB Server Configuration ダイアログ・
ボックス 320

Siebel DB 設定ダイアログ・ボックス 318

Siebel DB 診断 315

コントローラでの設定 318

ログの有効化 316

Siebel Server Configuration ダイアログ・ボック
ス 311

Siebel 設定ダイアログ・ボックス 309

Siebel 診断 305

Siebel 診断の有効化 306

Web サーバが DMZ 内にある場合の設
定 307

アプリケーション・サーバ ID の生成
307

アプリケーション・サーバからメディ
エータへのファイルのコピー 308

アプリケーション・サーバの設定 305

コントローラでの設定 309

サーバのパフォーマンス設定の最適化
306

Solaris

IP アドレスの設定 433

SSL

MI リスナーの設定 276

エージェント構成設定 273

概要 441

start_transaction 関数 (GUI) 212

T

TCP/IP の設定 402

U**UNIX**

- rsh 406
- rsh を使用しない 407
- シェル 406
 - ロード・ジェネレータへの接続 399

UNIX LoadRunner エージェント 267**user_data_point 関数 212****W****WAN エミュレーション 95**

- 構成設定 97
- 詳細オプション 93
- セットアップ 96
- 定義済みのプロファイルの設定 93
- 停止と開始 103
- ホストの除外 101

WinRunner

- WinRunner を使用した GUI スクリプトの作成 209

WinRunner, LoadRunner への 209**あ****アプリケーション**

- 構成設定 23
- 使用モデル 24
- 分析 22-24

い**一時停止**

- 仮想ユーザ 229
- モニタ 369

インストール

- ファイアウォール越しの監視機能 282

え**エージェント**

- サマリ・ウィンドウ 251
- 定義 7
- デーモン 408
- トラブルシューティング 423

エキスパート・モード 393-400

- UNIX ロード・ジェネレータへの接続 399
- 一般設定 394

出力の設定 396**デバッグ設定 395****モニタ設定 398****エラー 仮想ユーザのステータス****シナリオ・グループ表示枠 240****エラー処理 364****お****応答時間の測定****GUI 27****エンドツーエンド 26****サーバ 28****ネットワークとサーバ 27****ミドルウェアとサーバ 28****オプション・ダイアログ・ボックス****一般タブ 394****実行環境の設定タブ 175****実行ファイルの保存場所タブ 178****タイムアウト・タブ 172****デバッグ情報タブ 395****パス変換テーブル 390****オンライン・グラフ 367-379****2つのグラフの結合 378****x 軸のスタイル 371****Y 軸の目盛り 372****エクスポート 379****オフラインでのデータ表示 379****解析 383-385****グラフを開く 360****更新率 371****サンプリング頻度 364****線の色 373****測定値の設定 373****測定値の倍率の変更 373****トランザクション・データ 363****表示のカスタマイズ 361****棒グラフ値 372****オンライン・グラフの解析 383-385****オンライン・モニタ 339-344****一時停止 369****エラー処理 364****オフラインでのデータ表示 379****オンライン・グラフの解析 383****起動 358****グラフ 367****グラフ時間 372**

線の表示 / 非表示 375
デバッグ 365
表示形式 372

か

仮想ユーザ

GUI 仮想ユーザ 9
RTE 仮想ユーザ 12
一時停止 229
エラー, 警告, 通知メッセージ 242
仮想ユーザ・ウィンドウ 47
仮想ユーザ情報ダイアログ・ボックス 107
仮想ユーザ・スクリプト・ログ 229
仮想ユーザ・ログ 248
監視 240
グループに追加 69, 70
グループ・リスト 54-56
最大ユーザ負荷のエミュレート 161
実行 229
実行数の決定 29
シナリオ・グループ表示枠のステータス 240
シナリオ実行中のその他の仮想ユーザのアクティブ化 232
種類 7
設定 59
定義 5
停止 229
番号変更 230
表示 239-252
ロード 230
仮想ユーザ・ウィンドウ 63
仮想ユーザ, 『仮想ユーザ』 参照
仮想ユーザ関数
GUI 212
仮想ユーザ・グループ
一時停止 223
仮想ユーザを追加 69, 70
削除 59
作成 54-61
実行 223
スケジュール 156
停止 224
有効化 / 無効化 59
ロード 222

仮想ユーザ数の引き上げ 415
仮想ユーザ・スクリプト 5
Quality Center からの追加 200
仮想ユーザ情報の取得 213
仮想ユーザの総数に対する割合を割り当てる (パーセント・モード) 116
グループのための選択 70
(ゴール指向シナリオ) への定義された目標の百分率割り当て 138
(ゴール指向シナリオ) へのロード・ジェネレータの割り当て 138
コマンド・ライン・オプション 105
削除 (ゴール指向シナリオ) 145
削除 (パーセント・モード) 123
作成 18
シナリオの選択 39
スクリプトの詳細の変更 103, 107
設定 103-108
追加 (ゴール指向シナリオ) 141
追加 (パーセント・モード) 118
定義 5
ファイル 106
編集 107
有効化 / 無効化 (ゴール指向シナリオ) 145
有効化 / 無効化 (マニュアル・シナリオ) 123
ロード・ジェネレータの選択 58
ロード・ジェネレータの割り当て (パーセント・モード) 116
仮想ユーザ・タイプ
GUI 209
仮想ユーザの実行 / 停止ダイアログ・ボックス 232
仮想ユーザの表示 239-252
エージェント・サマリ 251
概要 239
出力メッセージ・ウィンドウ 242
仮想ユーザを追加ダイアログ・ボックス 69
関数
GUI 212
完了-失敗-仮想ユーザのステータスシナリオ・グループ表示枠 240
完了-成功-仮想ユーザのステータスシナリオ・グループ表示枠 240

き

起動

コントローラ 37

機能テスト・スクリプト, 「GUI 仮想ユーザ・スクリプト」も参照

機能テスト・スクリプト, LoadRunner との統合 203

く

グラフ, 「オンライン・グラフ」参照

グラフ時間 372

グラフの設定ダイアログ・ボックス 370

グラフを重ねて表示ダイアログ・ボックス 378

グループ情報ダイアログ・ボックス 60

グループの追加ダイアログ・ボックス 56

け

継続時間 154

結果 182

Quality Center プロジェクトの場所 199

格納場所の指定 182

照合 186

ディレクトリのファイル構造 184

デバッグ対象ファイル 409

名前 182

結果ディレクトリの設定ダイアログ・ボックス

Quality Center プロジェクト 199

ローカルまたはリモート 183

結果の照合ダイアログ・ボックス 187

こ

更新ボタン 104

ゴール指向シナリオ 127-145

ゴールの定義 131

シナリオ・タイプの選択 39

スクリプトへの目標の百分率の割り当て 138

スクリプトへのロード・ジェネレータの割り当て 138

デザイン・タブ 129

コマンド・ライン

オプション, 仮想ユーザ・スクリプト 105

引数 437

コントローラ 40

概要 40

クイック・ツアー 37-48

コマンド **pd** からの実行 437

シナリオ・ファイルの管理 43
定義 5

メッセージの送信 (GUI) 213

コントローラ・ウィンドウ

ステータス・バー 40

タイトル・バー 40

ツールバー 42

デザイン・ビュー 42

メニュー・バー 40

コントローラの起動 37

さ

サーバ・モニタ 287

測定値の追加と削除 286

測定頻度の設定 286

プロパティの設定 283

サーバのルーティング・テーブル 435

作成

仮想ユーザ・グループ 54-61

仮想ユーザ・スクリプト 18

ゴール指向シナリオ 127-145

パーセント・モードでのマニュアル・シナリオ 111-125

マニュアル・シナリオ 51-110

サーバのクローニング 287

サマリ情報ダイアログ・ボックス 190

サンプリング頻度 364

し

実行-仮想ユーザのステータス

シナリオ・グループ表示枠 240

実行環境の設定

共有 (コントローラ) 72

ゴール指向シナリオ (コントローラ) 内の設定 142

コントローラ内の設定 104

設定 (コントローラ) 63

実行時間グラフ

解析 383

実行メモ・ダイアログ・ボックス 251

シナリオ

Quality Center から開く 195

- Quality Center への保存 197
- QuickTest テストの実行 214
- WinRunner テストの実行 214
- 概要 52
- 仮想ユーザ・グループ・モードへの切り替え 124
- 結果ディレクトリ 184
- 結果の照合 186
- ゴール指向シナリオの作成 127-145
- サマリ情報 189
- 実行 219-231
- 実行, GUI 仮想ユーザ・スクリプト 214
- 実行の概要 46-48
- 実行の準備 181-190
- シナリオ開始ダイアログ・ボックス 149
- シナリオ・スクリプト表示枠 129, 131
- シナリオのゴール・ウィンドウ 129, 131
- シナリオのゴールの定義 131
- シナリオの新規作成 44
- シナリオ・ファイルの管理 43-45, 46
- 出力メッセージの表示 242
- 新規シナリオ・ダイアログ・ボックス 52
- スケジュール 153
- 設定 171-180
- 定義 5
- 定義されたゴール指向シナリオ 39
- 定義されたマニュアル・シナリオ 39
- 閉じる 45
- パーセント・モードでのマニュアル・シナリオの作成 111-125
- パーセント・モードへの切り替え 112
- 開く 44
- 保存 45
- マニュアル・シナリオの作成 51-110
- シナリオゴールの編集ダイアログ・ボックス 131
- シナリオ設定タブ 131
- 負荷の動作タブ 131
- プレビューのロード 131
- シナリオの切り替え
 - 仮想ユーザ・グループ・モードへの 124
 - パーセント・モードへの 112
- シナリオの結果の照合 186
- シナリオの実行 219-231
 - アクティブな仮想ユーザの監視 240
 - 概要 46-48, 219
 - 仮想ユーザ・グループの一時停止 223
 - 仮想ユーザ・グループの実行 223
 - 仮想ユーザ・グループの停止 224
 - 仮想ユーザ・グループのロード 222
 - 仮想ユーザのロード 230
 - 継続時間の制限 153
 - 個々の仮想ユーザ・グループの制御 222
 - 個々の仮想ユーザの制御 228
 - その他の仮想ユーザのアクティブ化 232
 - 遅延 148, 153
 - 無人でのシナリオの実行 220
 - メッセージ 242
 - ランデブーからの手作業による仮想ユーザの解放 231
- シナリオの設計
 - パス変換 180
- シナリオの設定
 - 結果の格納場所の指定 182
 - 実行環境の設定 174
 - 実行ファイルの格納場所 176
 - タイムアウト時間 172
- 終了中-仮想ユーザのステータス
 - シナリオ・グループ表示枠 240
- 出力ウィンドウ
 - デバッグ情報 412
- 出力ファイル 413
- 出力メッセージ・ウィンドウ 242-247
 - 更新 245
 - 消去 246
 - メッセージの詳細の表示 246
 - メッセージの並べ替え 247
 - メッセージのファイルへの保存 246
 - メッセージのフィルタ処理 246
 - ログ情報のドリル・ダウン 244
- 準備完了-仮想ユーザのステータス
 - シナリオ・グループ表示枠 240
- 詳細ボタン 246
- 初期化 228
- 初期化-仮想ユーザのステータス

シナリオ・グループ表示枠 240
 初期化クォータ 84
 徐々に終了—仮想ユーザのステータス
 シナリオ・グループ表示枠 240
 新規監視対象サーバプロパティ・ダイアログ
 ボックス 283, 287
 新規グラフを開くダイアログ・ボックス 360
 新規シナリオ・ダイアログ・ボックス 37
 新規ロードジェネレータの追加ダイアログ・
 ボックス 74

診断

ERP/CRM メディエータ 294
 Oracle 11i ログの有効化 323
 Siebel DB ログの有効化 316
 診断結果の表示 304

す

スクリプト, 「仮想ユーザ・スクリプト」参照
 スクリプト情報ダイアログ・ボックス 142
 スクリプト・パス, 相対 108
 スクリプトを追加ダイアログ・ボックス 141
 スケジュールの編集 150
 スケジュール・ビルダ 147-160
 仮想ユーザ・グループ 156
 シナリオの実行 153
 スケジュールの削除 151
 スケジュールの作成 150
 スケジュールの選択 150, 150-160
 スケジュールの名前変更 151
 スケジュールの変更 150
 スケジュールビルダ・ダイアログ・
 ボックス 151
 スケジュール・ビルダ・ダイアログ・ボック
 ス
 継続時間タブ 154
 ランプ・アップ・タブ 153
 ランプ・ダウン・タブ 155
 スケジュール・ビルダ・ダイアログ・ボック
 ス (グループ)
 Start Time tab 156
 継続時間タブ 158
 ランプ・アップ・タブ 157
 ランプ・ダウン・タブ 159
 ステータス・バー 40
 ステートメント
 QuickTest での追加 208

せ

接続

Quality Center への 192
 データベースへ 409
 接続ログ・タブ 400
 設定 367-379
 一般 394
 仮想ユーザ 59
 シナリオ 171-180
 出力 396
 スクリプト 103-108
 スクリプト (ゴール指向シナリオ)
 141-145
 スクリプト (パーセント・モード)
 118-123
 測定頻度 286
 タイムアウト時間 172
 デバッグ 395
 モニタ 398
 ロード・ジェネレータ 52, 74-78, 79-103
 ロード・ジェネレータの設定 79-97
 測定頻度, 設定 286

そ

相対スクリプト・パス 108
 測定値の削除 286
 測定値の設定ダイアログ・ボックス
 詳細タブ 373
 設定タブ 373
 測定値の追加 286
 測定値の表示 / 非表示
 オンライン・モニタ 369
 トランザクション・モニタ 375

た

タイムアウト時間
 コントローラの設定 172
 ダウン—仮想ユーザのステータス
 シナリオ・グループ表示枠 240
 端末サービス 89
 エージェント 90
 仮想ユーザの分配 93
 クライアントの起動 91

ち

- 中止—仮想ユーザのステータス
シナリオ・グループ表示枠 240

つ

- ツールバー 42

て

停止

- 仮想ユーザ 229
- 仮想ユーザ・グループ 224

データベース

- 接続 409

デザイン・タブ

- パーセント・モード 112
- マニュアル・シナリオ 55

デジタル証明書

- MI リスナーの設定 276
- 概要 441

デジタル認証

- エージェント構成設定 273

テスト工程

- 仮想ユーザ・スクリプトの作成 18
- シナリオの監視 19
- シナリオの作成 18
- シナリオの実行 19
- テスト結果の分析 20
- テストの計画 18

テスト目標

- 一般的目標の要約 32–34
- 定義 25–26

デバッグ

- 情報設定 395
- レベル 365

と

統合

- QuickTest の LoadRunner への 206
- WinRunner, LoadRunner への 209

トラブルシューティング

- エージェント 423
- コントローラ 401–425
- ファイアウォール 416

トランザクション 5

- QuickTest スクリプトに含める 207

失敗 411

- ダイアログ・ボックス 241
- 定義 5
- 定義対象の決定 29
- トランザクション・データ 363

ね

ネットワーク

- ネットワーク遅延時間オプション 372

は

パーセント・モード

- 仮想ユーザ・グループ・モードへの切り替え 124
- 仮想ユーザの総数に対する割合をスク립トに割り当てる 116
- 仮想ユーザの総数の定義 114
- シナリオの作成 111–125
- シナリオをパーセント・モードに切り替え 112
- スクリプトへのロード・ジェネレータの割り当て 116

デザイン・タブ 112

ハードウェア

- 通信の確認 402
- テスト対象の選択 30–32

パス変換

- シナリオの設計 180
- スクリプト・パス 109
- 定義 387
- パス変換テーブルの使用 389
- パス変換テーブルの編集 390
- ファイル位置のデバッグ 409
- 例 391

パフォーマンス測定 of the 範囲 26

- パフォーマンスの分析, 定義された 6
- 番号の再割り当て, 仮想ユーザ 230

ふ

ファイアウォール

- MI リスナーのインストール 274
- インストールの設定 258
- 概要 255
- 仮想ユーザの実行 263–279
- コントローラの設定 276
- 設定の概要 259

- 外側からの監視 281–288
- データ収集の準備 282
- トラブルシューティング 260, 416
- ファイアウォール越しに動作するエージェントの設定 271
- ファイアウォール越しの監視機能のインストール 282
- ファイアウォール越しの監視 281–288
 - 概要 281
 - 測定値の追加と削除 286
 - データ収集の準備 282
 - 頻度の設定, 測定値 286
 - プロパティの設定 283
- ファイアウォール越しの監視機能のインストール 282
- ファイアウォール越しの実行 263–279
- ファイアウォールでの作業 255
- ファイル, 仮想ユーザ・スクリプト 106
- 負荷テスト, 定義 3
- 負荷テストの計画 21–34
- 負荷分散 140

- ほ**
- 保留中—仮想ユーザのステータス
シナリオ・グループ表示枠 240

- ま**
- マニュアル・シナリオ
 - 作成 51–110
 - 定義 39
 - デザイン・タブ 55
 - パーセント・モード 111–125
- マルチ IP アドレス 394
 - コントローラへの接続 406
 - 有効化 394

- み**
- ミドルウェア
 - 応答時間の測定 28
 - システム構成 23

- む**
- 無効化
 - 仮想ユーザ・グループ 59
 - 仮想ユーザ・スクリプト (ゴール指向) 145
- 仮想ユーザ・スクリプト (マニュアル・シナリオ) 123

- め**
- メッセージ
 - コンソールへの送信 (GUI) 212
- メッセージの送信
 - GUI 仮想ユーザ 212

- も**
- モニタ
 - オンライン 339–344
- モニタ設定ダイアログ・ボックス 283

- ら**
- ライセンス情報の変更 14
- ランタイム・ビューア
 - コントローラから再生の表示 229
- ランデブー 161–168
 - 解放ポリシーの設定 165
 - 概要 161
 - 仮想ユーザのステータス 240
 - 仮想ユーザを無効にする 166
 - 仮想ユーザを有効にする 166
 - 情報 167
 - 属性 163
 - 属性の設定 163
 - タイムアウト・ポリシーの設定 165
 - 定義 5
 - 手作業による仮想ユーザの解放 231
- ランデブー情報ダイアログ・ボックス 168
- ランプ・アップ 153
- ランプ・ダウン 155

- り**
- リスト 18
 - 仮想ユーザ・グループのリスト 54–56
 - スクリプト・リスト 103–108
 - スクリプト・リスト (ゴール指向シナリオ) 141–145
 - スクリプト・リスト (パーセント・モード) 118–123
 - ランデブー・リスト 161–168
 - ロード・ジェネレータ・リスト 74–78

リソース

Mercury Interactive の Web サイト xv

リモート・エージェント・ディスパッチャ
(プロセス) 7

る

ルーティング・テーブル 435

れ

レジストリの変更 415

ろ

ロード・ジェネレータ 5

IP アドレスの追加 428

構成設定 52, 74-78

選択 69

選択 (パーセント・モード) 116

属性の設定 79-103

追加 77

定義 5

分散 140

変更 77

マルチ IP アドレス 394

ロード・ジェネレータ情報の詳細の表
示 75

ロード・ジェネレータ・ウィンドウ 74

ロード・ジェネレータの情報ダイアログ・
ボックス 79

Unix 環境タブ 82

WAN エミュレーション詳細オプション
99

WAN エミュレーション・タブ 97

仮想ユーザ・ステータス・タブ 88

仮想ユーザの制限タブ 85

実行時クォータ・タブ 84

実行ファイルの保存場所タブ 81

ステータス・タブ 80

端末サービス・タブ 93

ファイアウォール・タブ 86

ロード・ジェネレータの設定 74

UNIX シェル 82

WAN エミュレーション 95

エキスパート・モード 398

仮想ユーザの制限 85

コントローラとの通信の確認 403

実行ファイル 81

初期化クォータ 84

端末サービス 93

ファイアウォール 86

ロード・ジェネレータの接続 75

ロード・ジェネレータの切断 75

ロード・ジェネレータの選択 69

ロード・ジェネレータの無効化 74

ロード・ジェネレータの有効化 74