

LoadRunner

コントローラ・ユーザーズ・ガイド

Version 7.8

LoadRunner コントローラ・ユーザーズ・ガイド, Version 7.8

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 5,701,139; 5,657,438; 5,511,185; 5,870,559; 5,958,008; 5,974,572; 6,138,157; 6,144,962; 6,205,122; 6,237,006; 6,341,310; および 6,360,332。その他の特許は米国およびその他の国で申請中です。すべての権利は弊社に帰属します。

ActiveTest, ActiveTune, Astra, FastTrack, Global SiteReliance, LoadRunner, Mercury Interactive, Mercury Interactive のロゴ, Open Test Architecture, Optane, POPs on Demand, ProTune, QuickTest, RapidTest, SiteReliance, SiteRunner, SiteScope, SiteSeer, TestCenter, TestDirector, TestSuite, Topaz, Topaz AIMS, Topaz Business Process Monitor, Topaz Client Monitor, Topaz Console, Topaz Delta, Topaz Diagnostics, Topaz Global Monitoring Service, Topaz Managed Services, Topaz Open DataSource, Topaz Real User Monitor, Topaz WeatherMap, TurboLoad, Twinlook, Visual Testing, Visual Web Display, WebTest, WebTrace, WinRunner および XRunner は、米国およびその他の国の Mercury Interactive Corporation または Mercury Interactive Corporation が 100% 出資している子会社である Mercury Interactive (Israel) Ltd. の登録商標です。

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 Interactive Corporation 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
1325 Borregas Avenue
Sunnyvale, CA 94089 USA
Tel: (408) 822-5200
Toll Free: (800) TEST-911, (866) TOPAZ-4U
Fax: (408) 822-5300

© 2003 Mercury Interactive Corporation, All rights reserved

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

目次

LoadRunner へようこそ	xi
オンライン・リソース.....	xi
LoadRunner のマニュアル.....	xii
LoadRunner 付属マニュアルの使い方.....	xiii
表記規則.....	xv

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.....	8
第 2 章 : LoadRunner テスト工程	15
ステップ I : テストの計画.....	16
ステップ II : 仮想ユーザ・スクリプトの作成.....	16
ステップ III : シナリオの作成.....	16
ステップ IV : シナリオの実行.....	17
ステップ V : シナリオの監視.....	18
ステップ VI : テスト結果の分析.....	18
第 3 章 : 負荷テストの計画	19
負荷テストの計画について.....	19
アプリケーションの分析.....	20
テスト目標の定義.....	23
LoadRunner の導入計画.....	25
負荷テスト目標の調査.....	31

第 4 章 : LoadRunner コントローラの概要	35
コントローラの起動	36
LoadRunner コントローラの紹介.....	38
シナリオ・ファイルの管理	42
シナリオの実行	45

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	64
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
ターミナル・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	102
スクリプトに対する相対パスの使用	106
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	109
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 ...	109
仮想ユーザの総数の定義	112
スクリプトへのプロパティの割り当て.....	113
スクリプトの設定.....	116
シナリオの仮想ユーザ・グループ・モードへの切り替え.....	122
第 7 章 : ゴール指向シナリオの作成	123
ゴール指向シナリオの計画について	123
ゴール指向シナリオの [デザイン] タブについて.....	126
シナリオのゴールの定義	128
スクリプトへのプロパティの割り当て.....	134
スクリプトの設定	137
第 8 章 : シナリオのスケジュール	143
シナリオのスケジュールについて.....	144
シナリオの開始の遅延.....	145
スケジュールの選択	146
シナリオのスケジュール	149
仮想ユーザ・グループのスケジュール	152
スケジュールされたシナリオへの仮想ユーザの追加	157

第 9 章 : ランデブー・ポイントの使用	159
ランデブー・ポイントの使用について	159
ランデブー属性の設定	161
ランデブー・ポリシーの設定	162
ランデブー・ポイントを有効 / 無効にする	164
ランデブー・ポイントの仮想ユーザを有効 / 無効にする	164
ランデブー情報の表示	166
第 10 章 : シナリオの設定	169
シナリオの設定について	169
シナリオ実行環境の設定	170
タイムアウト時間の設定	172
実行ファイルの格納場所の設定	175
パス変換の指定	178
第 11 章 : シナリオの実行準備	179
シナリオの実行準備について	179
結果の格納場所の指定	180
結果ディレクトリのファイル構造	183
結果の照合	184
シナリオのサマリ情報の設定	186
第 12 章 : TestDirector を使ったシナリオ管理	187
TestDirector を使ったシナリオ管理	187
TestDirector の接続と切断	188
TestDirector プロジェクトからシナリオを開く	192
TestDirector プロジェクトへのシナリオの保存	193
TestDirector プロジェクトへの結果の保存	195
TestDirector プロジェクトからの仮想ユーザ・スクリプトの追加	196

第 3 部 : シナリオの実行

第 13 章 : シナリオの実行	201
シナリオの実行について	201
シナリオ全体の実行	203
仮想ユーザ・グループの制御	204
個々の仮想ユーザの制御	211
ランデブーからの手作業による仮想ユーザの解放	213
実行中のシナリオへの手作業による仮想ユーザの追加	214

第 14 章 : 実行中の仮想ユーザの表示	221
実行中の仮想ユーザの表示について	221
仮想ユーザのステータスの監視	222
[出力メッセージ] ウィンドウの表示	225
仮想ユーザ・スクリプト・ログの表示.....	230
実行メモの記録	233
エージェントのサマリ表示.....	234
第 15 章 : ファイアウォールに関する作業	235
LoadRunner でのファイアウォールの使用について	236
ファイアウォール越しの仮想ユーザの実行および監視の概要	240
ファイアウォールの内側への LoadRunner エージェントの インストール	241
ファイアウォールの内側の LoadRunner エージェントの設定	242
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	251
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	252
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	254
ファイアウォール越しの監視.....	256
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.....	256
サーバ・モニタのプロパティの設定	257
測定値の追加と削除	262
測定頻度の設定	263
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.....	263
トラブルシューティングのヒント.....	264

第 4 部 : シナリオの監視

第 16 章 : オンライン・モニタ	269
オンライン・モニタについて	270
モニタの起動	272
オンライン・モニタ・グラフを開く	274
サーバ・リソースの監視	276
グラフ表示のカスタマイズ.....	277
オンライン・モニタの設定.....	277
モニタ・オプションの設定.....	278
オンライン・グラフの設定.....	281
グラフの結合	284
オンライン・モニタ・グラフについて	285
オンライン測定の設定	287
オンライン・モニタ・グラフのエクスポート.....	292
オフラインでのデータ表示.....	292

第 17 章 : リモート・パフォーマンス監視	293
リモート・パフォーマンス監視について	294
リモート・パフォーマンス・モニタ・サーバのインストール	295
リモート・パフォーマンス・モニタのユーザ設定の構成	295
LoadRunner リモート・パフォーマンス・モニタ・サーバへの接続	298
負荷テスト・データの監視	300
オンライン・グラフ表示	300
オンライン・グラフ設定のカスタマイズ	303
第 18 章 : 実行環境とトランザクションの監視	309
実行時間グラフとトランザクション・グラフについて	309
実行時間 グラフ	310
ユーザ定義データ・ポイント・グラフ	311
トランザクション・モニタ・グラフ	312
トランザクション・モニタを有効にする	313
スクリプトへのトランザクションの追加	314
Web ページ・ブレイクダウンの有効化	315
第 19 章 : Web リソースの監視	317
Web リソースの監視について	317
[秒ごとのヒット数] グラフ	318
[スループット] グラフ	318
[秒ごとの HTTP 応答数] グラフ	319
[秒ごとにダウンロードされたページ] グラフ	321
[秒ごとの再試行数] グラフ	322
[接続] グラフ	322
[秒ごとの接続数] グラフ	322
[秒ごとの SSL 接続] グラフ	322
第 20 章 : システム・リソースの監視	325
システム・リソースの監視について	325
Windows リソース・モニタの設定	327
UNIX リソース・モニタの設定	333
UNIX での rstatd デーモンの設定	336
SNMP リソース・モニタの設定	337
Antara FlameThrower モニタの設定	339
SiteScope モニタの設定	351
第 21 章 : ネットワークの監視	353
ネットワークの監視について	353
UNIX 送信元マシンからのネットワークの監視	355
ネットワーク遅延時間モニタの設定	358
ネットワーク遅延時間グラフの表示	363

第 22 章 :ファイアウォール・サーバ・パフォーマンスの監視	365
ファイアウォール・サーバ・モニタについて.....	365
Check Point FireWall-1 サーバ・モニタの設定.....	366
第 23 章 :Web サーバ・リソースの監視	371
Web サーバ・リソース・モニタについて.....	371
Apache モニタの設定.....	372
Microsoft IIS モニタの設定.....	375
iPlanet/Netscape モニタの設定.....	378
iPlanet (SNMP) モニタの設定.....	382
プロキシ・サーバを使用した監視.....	389
第 24 章 :Web アプリケーション・サーバ・リソースの監視	391
Web アプリケーション・サーバ・リソース・モニタについて.....	392
Ariba モニタの設定.....	392
ATG Dynamo モニタの設定.....	397
BroadVision モニタの設定.....	402
ColdFusion モニタの設定.....	411
Fujitsu INTERSTAGE モニタの設定.....	414
iPlanet (NAS) モニタの設定.....	417
MS Active Server Pages モニタの設定.....	433
Oracle9iAS HTTP モニタの設定.....	435
SilverStream モニタの設定.....	440
WebLogic (SNMP) モニタの設定.....	443
WebLogic (JMX) モニタの設定.....	448
WebSphere モニタの設定.....	454
WebSphere (EPM) モニタの設定.....	465
第 25 章 :データベース・リソースの監視	477
データベース・リソースの監視について.....	477
DB2 モニタの設定.....	478
Oracle モニタの設定.....	494
SQL Server モニタの設定.....	501
Sybase モニタの設定.....	504
第 26 章 :ストリーミング・メディアの監視	513
ストリーミング・メディアの監視について.....	513
Windows Media サーバ・モニタの設定.....	514
RealPlayer サーバ・モニタの設定.....	516
RealPlayer クライアント・オンライン・グラフの表示.....	519
Media Player クライアント・オンライン・グラフの表示.....	520

第 27 章 :ERP/CRM サーバ・リソースの監視	521
ERP/CRM サーバ・リソースの監視について	521
SAP モニタの設定	522
SAP Portal モニタの設定	528
Siebel Web サーバ・モニタの設定	532
Siebel Server Manager モニタの設定	536
第 28 章 :Java パフォーマンスの監視	541
Java パフォーマンスの監視について.....	541
EJB パフォーマンスの監視.....	542
第 29 章 :J2EE パフォーマンスの監視	561
J2EE パフォーマンスの監視について.....	562
アプリケーション・サーバへの J2EE モニタのインストール.....	563
J2EE モニタの初期設定	565
クライアント・マシンでの J2EE モニタの有効化.....	567
アプリケーション・サーバ設定の変更例.....	571
J2EE モニタのトラブルシューティング	582
第 30 章 :Application Deployment ソリューション	583
Application Deployment ソリューションの監視について	583
Citrix MetaFrame Server モニタの設定.....	584
第 31 章 :ミドルウェアのパフォーマンス監視	595
ミドルウェアのパフォーマンスの監視について	595
Tuxedo モニタの設定	596
IBM WebSphere MQ モニタの設定.....	602
第 32 章 :オンライン・モニタのトラブルシューティング	613
サーバ・リソース・モニタのトラブルシューティング	613
ネットワーク遅延モニタのトラブルシューティング.....	616
ネットワークについて考慮すべき事項	618

第 5 部 : 付録

付録 A: LoadRunner オンライン・グラフの解析	623
オンライン・モニタ・グラフ.....	623
付録 B: パス変換の実行	627
パス変換について	627
パス変換テーブルへのエントリの追加	629
パス変換テーブルの編集	631
パス変換の例	632

付録 C: エキスパート・モードでの作業	633
エキスパート・モードへの切り替え方	633
オプション – 一般設定	634
オプション – デバッグ情報の設定	635
オプション – 出力の設定	638
オプション – モニタの設定	640
ロード・ジェネレータ情報 – UNIX 環境の設定	641
ロード・ジェネレータ情報 – 接続ログの設定	642
付録 D: コントローラのトラブルシューティング	645
トラブルシューティングについて	646
LoadRunner 通信	647
ロード・ジェネレータとの通信の失敗	648
AUT データベースへの接続の失敗	653
ファイルへのアクセスの失敗	654
仮想ユーザまたはトランザクションの失敗	655
Windows マシンの仮想ユーザ数の引き上げ	659
ファイアウォールのトラブルシューティング	660
付録 E: サーバ・モニタ・カウンタでの作業	669
モニタの標準設定のカウンタの変更	669
負荷テストに役立つカウンタ	670
付録 F: マルチ IP アドレスの設定	673
マルチ IP アドレスについて	673
ロード・ジェネレータへの IP アドレスの追加	675
IP ウィザードの使用	676
UNIX での複数 IP アドレスの設定	680
ルーティング・テーブルの更新	681
コントローラでの複数 IP アドレス機能の有効化	682
付録 G: コントローラのコマンド・ライン引数	683
コントローラのコマンド・ライン引数について	683
コマンド・ラインからのコントローラの起動	684
TestDirector の引数	685
実行時環境引数	686
付録 H: デジタル証明書の使い方	687
ファイアウォールでのデジタル証明書の使用	687
デジタル証明書の作成と使用	688
索引	693

LoadRunner へようこそ

LoadRunner は、マーキュリー・インタラクティブが提供する、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。負荷を生成するために LoadRunner では、ネットワーク上に分散する数千の仮想ユーザを実行します。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。

オンライン・リソース



LoadRunner には、以下のオンライン・リソースがあります。

最初にお読みください： LoadRunner の最新のお知らせと情報を提供します。

オンライン文書： 全マニュアルを PDF 形式で提供します。オンライン文書は Adobe Acrobat Reader を使って読んだり、印刷したりできます。Acrobat Reader は、LoadRunner のインストール・パッケージに含まれています。LoadRunner オンライン・マニュアルのアップデートについては、マーキュリー・インタラクティブのカスタマー・サポート Web サイトをご覧ください。

LoadRunner オンライン関数リファレンス： 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。『LoadRunner 関数リファレンス』オンライン・マニュアルのアップデートについては、マーキュリー・インタラクティブのカスタマー・サポート Web サイトをご覧ください。

LoadRunner コンテキスト・センシティブ・ヘルプ：LoadRunner の使用中に生じた疑問をすぐに解決できます。このヘルプは、各ダイアログ・ボックスの説明と LoadRunner を使った作業の手順を示します。ウィンドウ上またはウィンドウ内をクリックし、F1 キーを押すと、このヘルプが表示されます。LoadRunner ヘルプ・ファイルのアップデートについては、マーキュリー・インタラクティブのカスタマー・サポート Web サイトをご覧ください。

オンライン技術サポート：普段お使いの Web ブラウザで、マーキュリー・インタラクティブのカスタマー・サポート Web サイトを開きます。このサイトでは、マーキュリー・インタラクティブの最新情報や製品に関する情報をご覧になれます。この Web サイトの URL は、<http://www.mercury.co.jp/support> です。

サポート情報：マーキュリー・インタラクティブの Web サイトとカスタマー・サポート・サイト、世界のマーキュリー・インタラクティブの営業所を示します。

Mercury Interactive の Web サイト：普段お使いの Web ブラウザで、マーキュリー・インタラクティブのホーム・ページ (<http://www.mercury.co.jp>) を開きます。このサイトでは、マーキュリー・インタラクティブの最新情報や製品に関する情報をご覧になれます。

LoadRunner のマニュアル

LoadRunner には、以下の手順について説明するマニュアル一式が付属しています。

- ▶ LoadRunner のインストール
- ▶ 仮想ユーザ・スクリプトの作成
- ▶ LoadRunner コントローラの使用
- ▶ LoadRunner アナリシスの使用

LoadRunner 付属マニュアルの使い方

LoadRunner のマニュアルは、インストール・ガイド、コントローラ・ユーザーズ・ガイド、アナリシス・ユーザーズ・ガイド、および仮想ユーザ・スクリプトの作成に関するマニュアルで構成されています。

インストール・ガイド

LoadRunner のインストール方法については、『**LoadRunner インストール・ガイド**』を参照してください。『LoadRunner インストール・ガイド』では、以下のインストールについて説明します。

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

コントローラ・ユーザーズ・ガイド

LoadRunner の付属マニュアルには、コントローラ・のユーザーズ・ガイドが 1 冊含まれます。

『**LoadRunner コントローラ・ユーザーズ・ガイド**』は、Windows 環境で LoadRunner コントローラ使用して、LoadRunner シナリオを作成し、実行する方法を説明します。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。『コントローラ・ユーザーズ・ガイド』は、LoadRunner のテスト工程の概要を説明します。

アナリシス・ユーザーズ・ガイド

LoadRunner の付属マニュアルには、アナリシスのユーザーズ・ガイドが 1 冊含まれます。

『**LoadRunner アナリシス・ユーザーズ・ガイド**』は、システム・パフォーマンスの分析を行うために、シナリオの実行後、LoadRunner アナリシス・グラフとレポートを作成する方法について、説明します。

仮想ユーザ・スクリプトの作成に関するガイド

LoadRunner の付属マニュアルには、仮想ユーザ・ジェネレータ (VuGen) ユーザーズ・ガイドが 1 冊含まれます。

- ▶ 『**LoadRunner 仮想ユーザ・スクリプトの作成**』は、VuGen を使った仮想ユーザ・スクリプトの作成方法を説明しています。必要に応じ、このマニュアルと併せてオンラインの『**LoadRunner 関数リファレンス**』と、次のマニュアルをお読みください。
- ▶ 『**WinRunner ユーザーズ・ガイド**』は、WinRunner を使った GUI 仮想ユーザ・スクリプトの作成方法を説明します。作成された仮想ユーザ・スクリプトは、Windows プラットフォームで実行できます。『**TSL オンライン・リファレンス**』は、このマニュアルと併せてご利用ください。

情報	参照先
LoadRunner のインストール	『 LoadRunner インストール・ガイド 』
LoadRunner のテスト・プロセス	『 LoadRunner コントローラ・ユーザーズ・ガイド 』
仮想ユーザ・スクリプトの作成	『 LoadRunner 仮想ユーザ・スクリプトの作成 』
シナリオの作成と実行	『 LoadRunner コントローラ・ユーザーズ・ガイド 』
テスト結果の分析	『 LoadRunner アナリシス・ユーザーズ・ガイド 』

表記規則

本書は、次の表記規則に従っています。

1, 2, 3	数字は操作手順を示します。
▶	ブリット記号はオプションまたは特徴を示します。
>	大なり記号はメニュー・レベルを区切ります（例： [ファイル] > [開く]）。
[太字]	アクションを実行する際のインタフェース要素の名前は、 全角の大括弧に 太字 で示します（例：[実行] ボタンをク リックします）。
太字	メソッド名または関数名、メソッドや関数の引数、ファ イル名、パスは、 太字 で示します。
Arial	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、Arial フォントで示します。
<>	ファイル・パスまたは URL アドレスの中の可変部分は、 山括弧で囲んで示します（例：<製品のインストール先 フォルダ> %bin）。
[]	省略可能な引数は、半角の大括弧で囲んで示します。
{ }	引数に割り当てる値の候補は、中括弧で囲んで示します。 値をいずれか 1 つ割り当てる必要があります。
...	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。

第 1 部

LoadRunner について

第 1 章

はじめに

アプリケーションの負荷テストをするために、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します。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。

たとえば、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。たとえば、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。

- ▶ コスト高。人員と機材がどちらも大量に必要。複雑。
- ▶ 特に、多数のテスト要員をまとめ上げ、一斉に作業を行う必要がある点。
- ▶ 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 の自動化ソリューションは、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、たとえば、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 start と end で囲みます。たとえば、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。たとえば、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

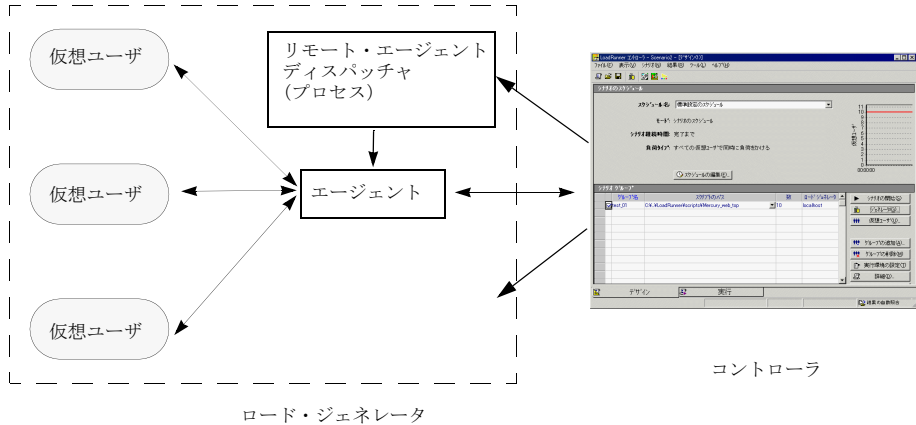
多数のインターネット・ユーザがアクセスするオンライン・バンキングの Web サーバをテストするものとします。この Web サイトは顧客に、振り込みや残高照会などの、各種バンキング・サービスを提供します。このサーバをテストするために、シナリオを作成します。シナリオでは、負荷テスト中にサーバで行われるアクションを定義します。

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。

▶ **クライアント / サーバ**

MSSQLServer, ODBC, Oracle (2 層), 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, Media Player, Multi Protocol Web/WS, Web (HTTP, HTML), Palm, および RealPlayer 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**

Oracle NCA, Peoplesoft (Tuxedo), SAP, および Siebel プロトコルの場合。

▶ **レガシ**

ターミナル・エミュレータ (RTE) の場合。

▶ **メーリング・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**

Tuxedo (6, 7) プロトコルの場合。

▶ **ワイヤレス**

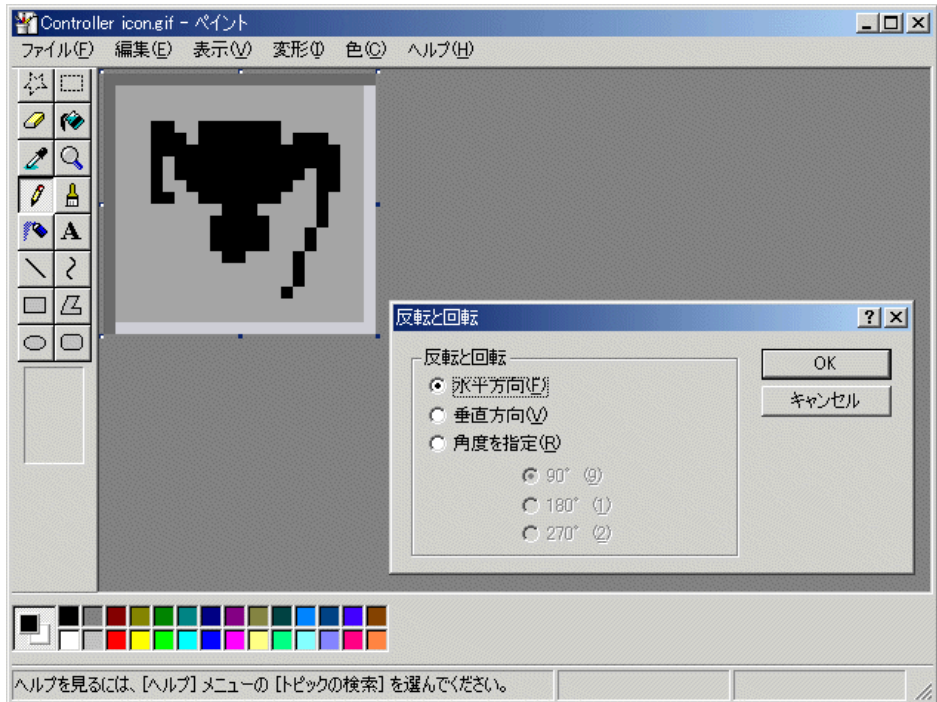
i モード, VoiceXML, WAP プロトコルの場合。

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。開発する各 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。たとえば、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 [保存] を選択する。



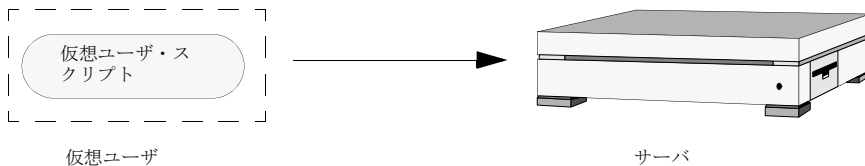
GUI 仮想ユーザがアプリケーションに対して実行する操作は、GUI 仮想ユーザ・スクリプトに定義されます。マージュー・インタラクティブの GUI テスト・ツールを使って、GUI 仮想ユーザ・スクリプトを作成します。WinRunner (Microsoft Windows のアプリケーションの場合) および Astra QuickTest (Web アプリケーションの場合) を使用します。

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、『LoadRunner 仮想ユーザ・スクリプトの作成』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 および SAP 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、LoadRunner の仮想ユーザ・スクリプト・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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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.

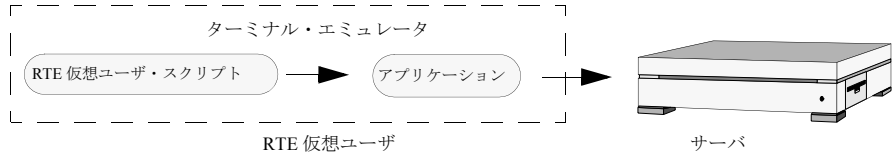
ATM Services
=====
operations:
  1) Withdraw Cash.
  2) Deposit Cash.
  3) Balance Report.
  4) Exit ATM.

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。仮想ユーザ・スクリプト・ジェネレータ (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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』を参照してください。

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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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, Application Deployment, ミドルウェア・パフォーマンス・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 4 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, 『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。

- ▶ 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

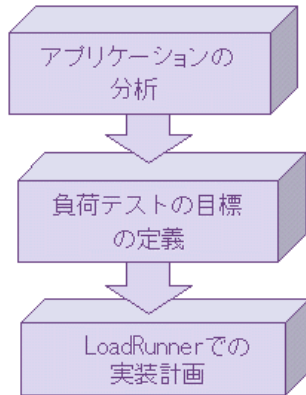
アプリケーション・テストには、ハードウェア、ソフトウェア、および人員のリソースが必要です。テストを始める前に、利用可能なリソースを把握し、効果的な使い方を考えておく必要があります。

- ▶ 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。たとえば、「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。



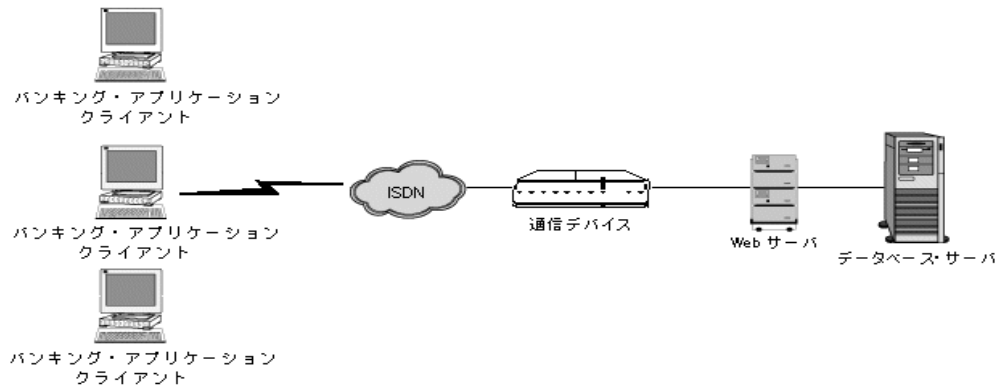
アプリケーションの分析

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・サーバ間のミドル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。

たとえば、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。

たとえば、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。たとえば、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。

以下に、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。

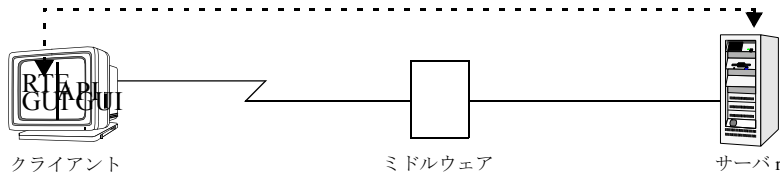
パフォーマンス測定範囲の定義

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。

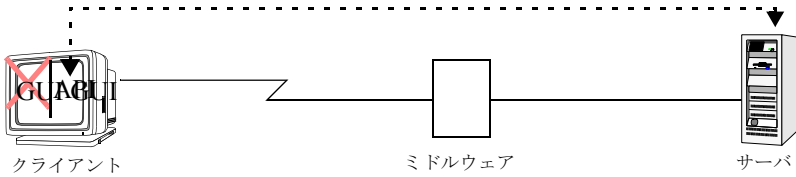
フロント・エンドで GUI 仮想ユーザや RTE 仮想ユーザを実行することにより、ターミナル・エミュレータまたは GUI フロント・エンド、ネットワーク、およびサーバを含む、ネットワーク全体の応答時間を測定できます。



▶ 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

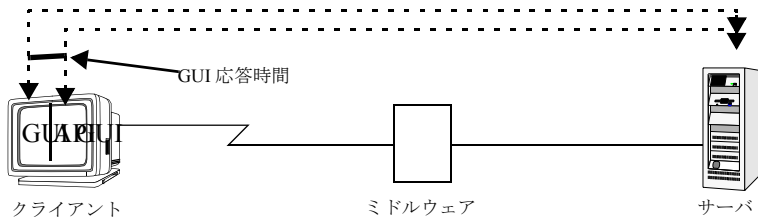
ンから多数の仮想ユーザを実行している場合、負荷がネットワークやサーバの応答時間にどのように影響するか測定できます。



▶ GUI 応答時間の測定 :

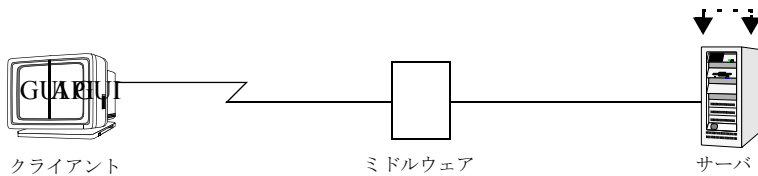
前の 2 つの測定値の差を計算すれば、クライアント・アプリケーション・インタフェースが応答時間に与える影響がわかります。

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



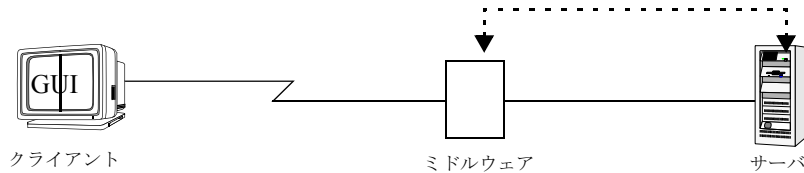
▶ サーバ応答時間の測定 :

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。たとえば、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。たとえば、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。たとえば、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。

たとえば、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
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。

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® サービ ス・パック 6a 以上 Windows 2000 Windows XP	Windows NT® サー ビス・パック 6a 以 上 Windows 2000 Windows XP	Windows NT® 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® 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

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.co.jp/products/loadrunner/technical/> を参照してください。

UNIX 構成設定の要件

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

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。

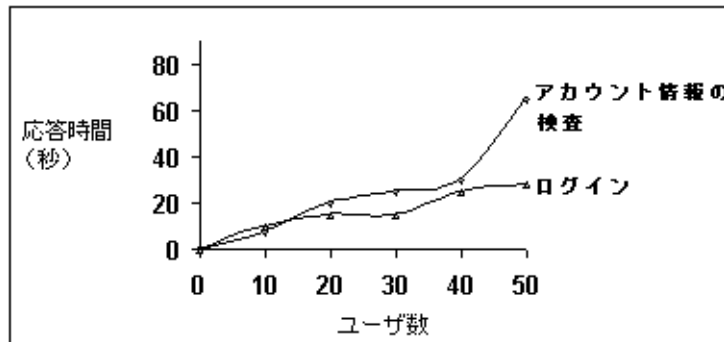
負荷テスト目標の調査

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。たとえば、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。

たとえば、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 : 200MHz, 64MB RAM
- ▶ 設定 2 : 200MHz, 128MB RAM
- ▶ 設定 3 : 266MHz, 128MB 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

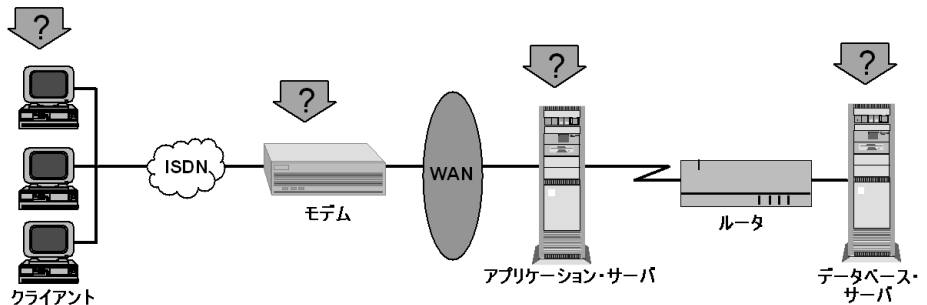
ハードウェアまたはソフトウェアの新しいリリースと以前のリリースを比較するには、回帰テストを実行します。アップグレードが応答時間にどのように影響しているか（ベンチマーク）、また信頼性にどのように影響しているか検査できます。アプリケーション回帰テストは、アップグレードの新しい機能を検査しません。新しいリリースが以前のリリースと同じ程度の効率性と信頼性を持っているかどうか検査します。

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。たとえば、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。

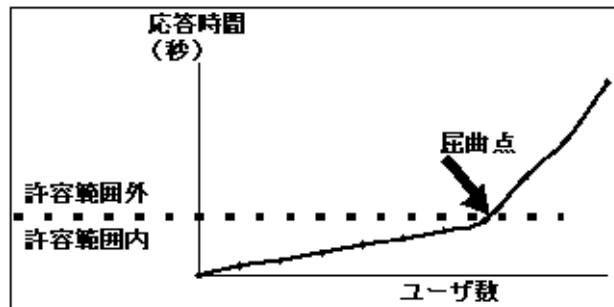
ボトルネックの特定

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、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4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がわかります。

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、以下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

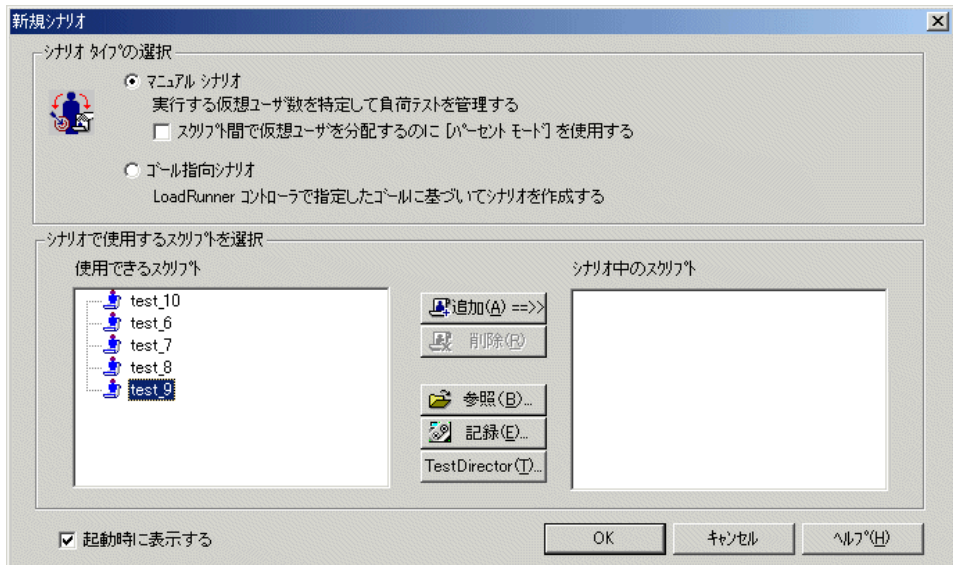
コントローラの起動

『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。

コントローラの起動は、次の手順で行います。



[スタート] > [プログラム] > [LoadRunner] > [Controller] を選択します。[LoadRunner コントローラ] ウィンドウが表示され、その内側に [新規シナリオ] ダイアログ・ボックスが表示されます。



[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。

TestDirector 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、[TestDirector] 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。

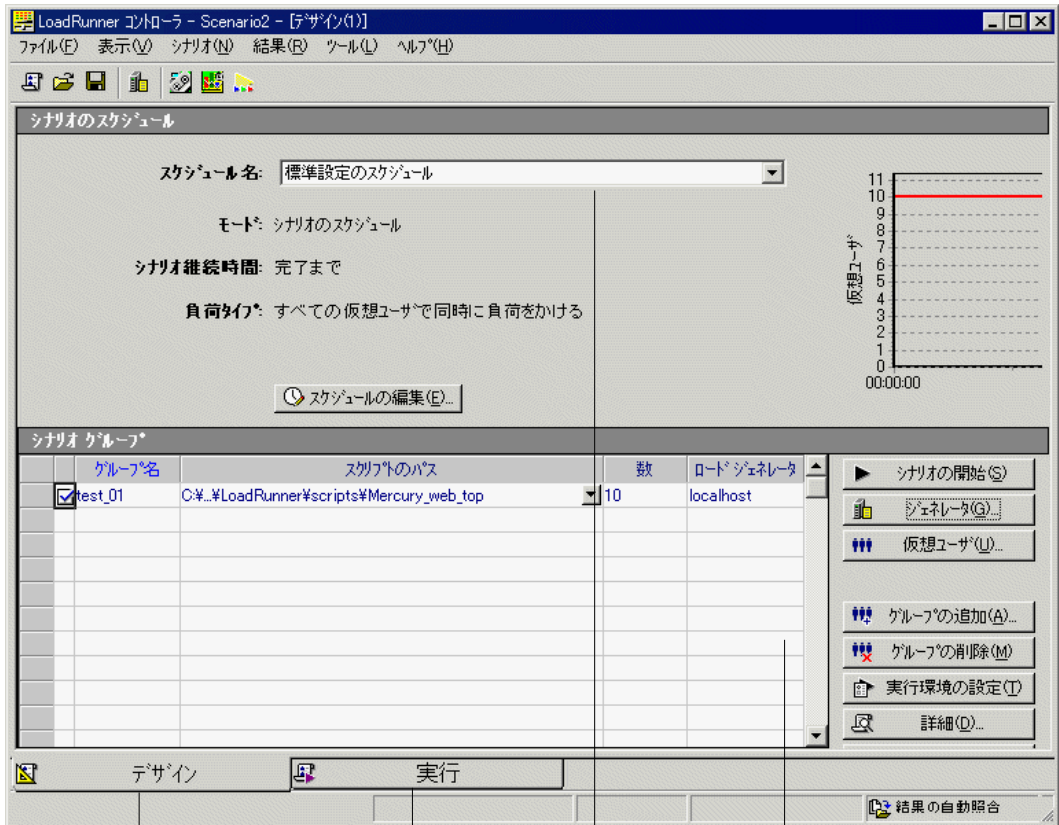
- 2 スクリプトを選択し、[追加] ボタンをクリックして、[シナリオ中のスクリプト] リストにコピーします。
- 3 [シナリオ中のスクリプト] リストからスクリプトを削除するには、[削除] ボタンをクリックします。
- 4 次回新規シナリオを作成する際に、このダイアログ・ボックスが表示されないようにするには、[起動時に表示する] チェック・ボックスをクリアします。シナリオを作成中も、後からスクリプトを追加できます。
- 5 [OK] をクリックして、ダイアログ・ボックスを閉じます。



LoadRunner コントローラの紹介

[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、TestDirector 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

[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。

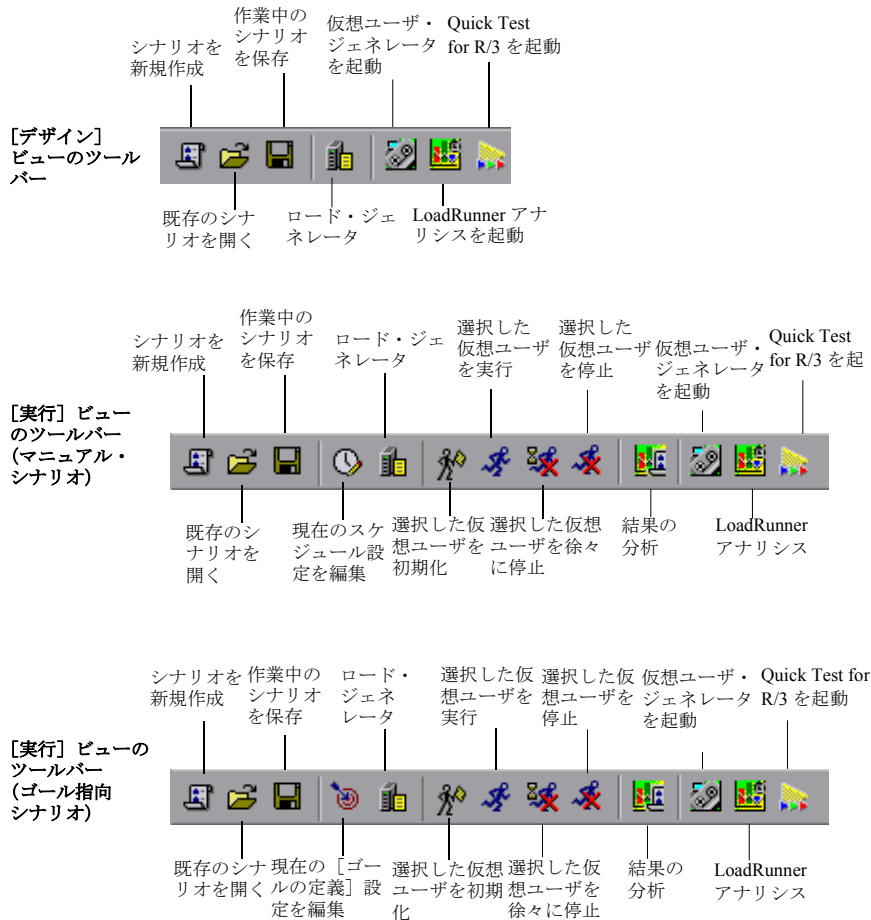
[デザイン] ビュー このビューには、シナリオ内のすべての仮想ユーザ・グループ/スクリプトのリスト、ロード・ジェネレータ・マシン、各グループ/スクリプトに割り当てられた仮想ユーザ数が表示されます。シナリオのスケジュール（マニュアル・シナリオ）またはゴール（ゴール指向シナリオ）についての基本的な情報も表示されます。

[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、[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。



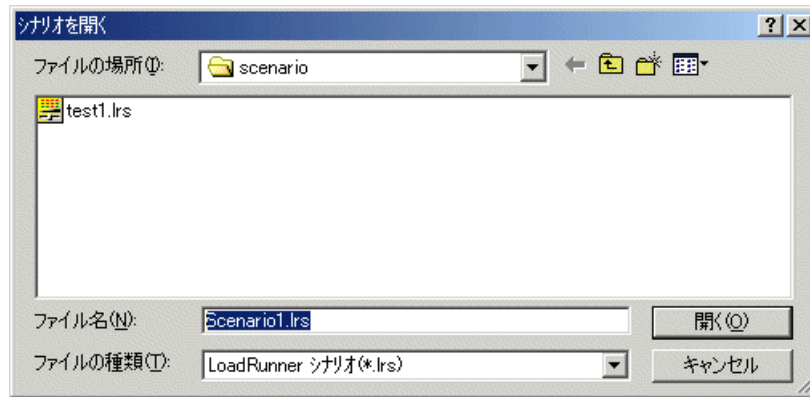
既存のシナリオを開く

[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。[ファイルを開く] 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。

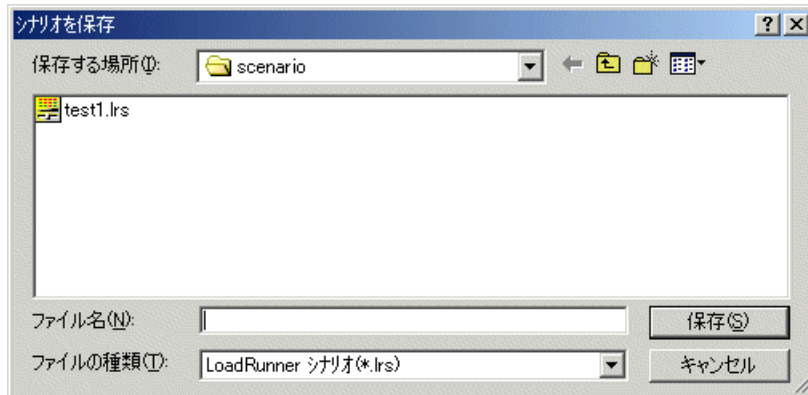
シナリオの保存

[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] 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。

シナリオ グループ

グループ	ユーザ	開始	終了	経過時間	エラー	成功	失敗	合計
test	10			00:00:15	0	0	0	0
test	4			00:00:09	0	0	0	0
test	1	5						
test	1							9
test	10							

シナリオ ステータス

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

利用可能なグラフ

- ユーザ定義データポイント
- エラーの統計
- エラーのある仮想ユーザ
- トランザクション グラフ
 - トランザクション 応答時間
 - 秒ごとのトランザクション (成功)
 - 秒ごとのトランザクション (失敗/停止)
 - 秒ごとのトランザクションの総合計
- Web リソース グラフ
 - 秒ごとのヒット数
 - スループット
 - 秒ごとの HTTP 応答数
 - 秒ごとのダウンロードされたページの...

実行中の仮想ユーザ - シナリオ全体

トランザクション 応答時間 - シナリオ全体

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

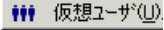
[デザイン]
タブ

[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・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。

ID	ステータス	スクリプト	ロード ジェネレータ	経過時間
1	実行中	_lab10vuser1.usr	localhost	00:00:07
2	実行中	_lab10vuser1.usr	localhost	00:00:08
3	実行中	_lab10vuser1.usr	localhost	00:00:08
4	実行中	_lab10vuser1.usr	localhost	00:00:08
5	実行中	_lab10vuser1.usr	localhost	00:00:07
6	実行中	_lab10vuser1.usr	localhost	00:00:07
7	実行中	_lab10vuser1.usr	localhost	00:00:08
8	実行中	_lab10vuser1.usr	localhost	00:00:08
9	実行中	_lab10vuser1.usr	localhost	00:00:08
10	実行中	_lab10vuser1.usr	localhost	00:00:07

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
実行中の仮想ユーザ	4
経過時間	00:00:52 (時間:分:秒)
秒ごとのヒット数	42.84 (最終 60 秒)
成功したトランザクション	86
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。

タイプ	メッセージコード (4)	サンプル メッセージ テキスト	メッセージ合...	仮想ユーザ	スクリプト	ジェネレータ
	-82202	利用可能な仮想ユーザのライセンス数を超えました。	17	7	2	1
	-27995	ShowItinerary(3) エラー -27995: 要求されたリソースが不足しています。	5	5	2	1
	-27979	Action1.c(42) エラー -27979: 要求されたフォームが見つけられませんでした。	3	1	1	1
	-27790	ShowItinerary(6) エラー -27790: サーバが応答しませんでした。	2	2	1	1

サマリ
メッセージを生成したホストの数です。

[出力メッセージ] ウィンドウの詳細については、225 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、アプリケーション配備、ミドルウェア・パフォーマンス、および Java パフォーマンスを監視できます。オンライン・モニタの詳細については、第 16 章「オンライン・モニタ」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、以下の項目について説明します。

- ▶ 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
- ▶ ターミナル・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

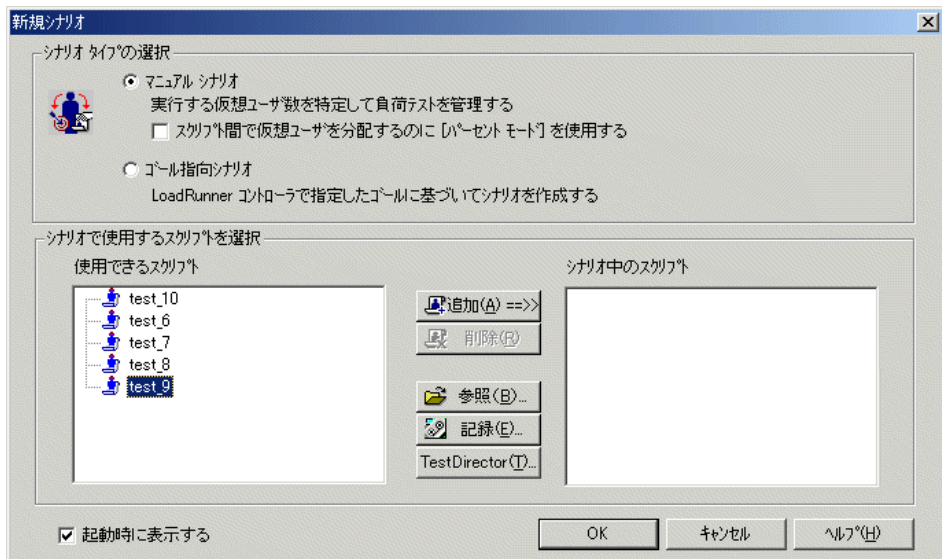
LoadRunner でシステムをテストするには、シナリオを作成する必要があります。シナリオは、テスト・セッションに関する情報が含まれているファイルです。シナリオは、現実のユーザをエミュレートするための手段です。シナリオには、実ユーザをエミュレートする方法についての情報（仮想ユーザ・グループ、仮想ユーザが実行するテスト・スクリプトおよびスクリプトを実行するロード・ジェネレータ・マシン）が含まれます。

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2つの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、[シナリオ中のスクリプト] 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、『LoadRunner 仮想ユーザ・スクリプトの作成』を参照してください。

[TestDirector]：[TestDirector への接続] ダイアログ・ボックスが開き、TestDirector プロジェクトへの接続を開くことができます。

[シナリオ中のスクリプト] : 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。

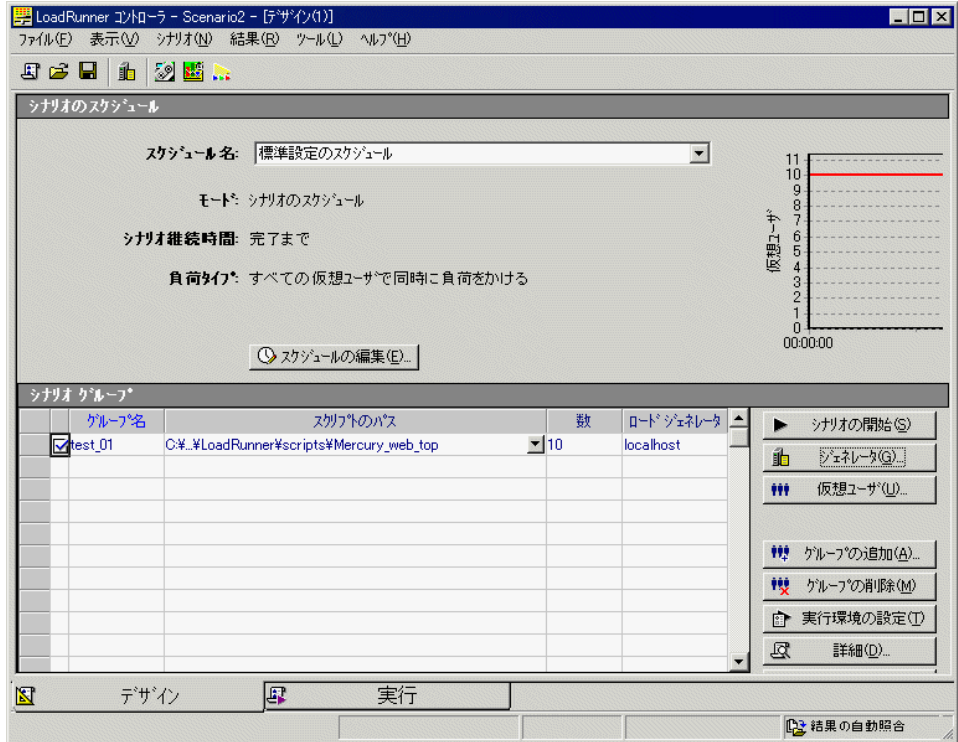
仮想ユーザ・グループの作成

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。たとえば、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。[プレビューのロード] 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、149 ページの [スケジュールの編集] 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、ロード・ジェネレータ・マシン、各グループに割り当てられた仮想ユーザの数が表示されます。

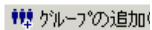
仮想ユーザ・グループまたはシナリオに対して、以下の操作が可能です。

- ▶ 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し、および追加したスクリプトの設定
- ▶ 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

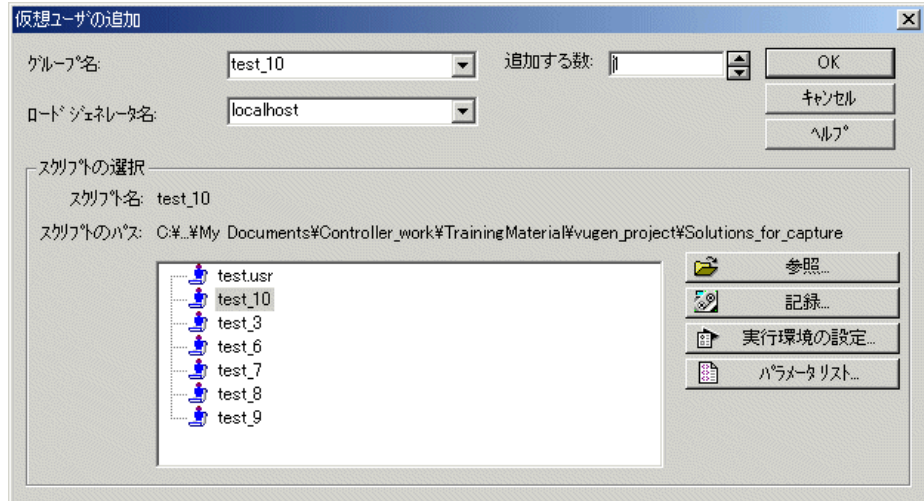
仮想ユーザ・グループの追加

[仮想ユーザの追加] 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・グループの作成は、次の手順で行います。

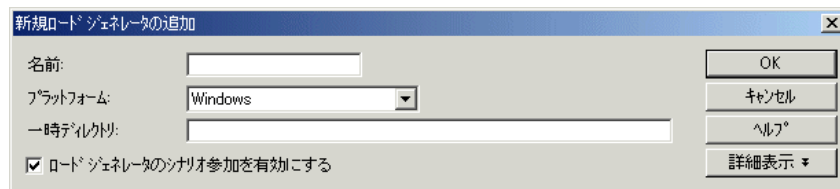


- 1 [シナリオ グループ] 表示枠の右の [グループの追加] ボタンをクリックします。[仮想ユーザの追加] ダイアログ・ボックスが表示されます。



- 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] 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。

[仮想ユーザの追加] 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。

[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。

[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。[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、106 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。

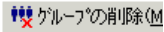
- ▶ **[記録]**：スクリプトの記録を開始できるように、仮想ユーザ・ジェネレータを開きます。スクリプトの記録の詳細については、『**LoadRunner 仮想ユーザ・スクリプトの作成**』を参照してください。

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クリックします。

仮想ユーザ・グループの削除

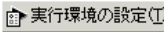
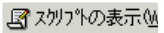


仮想ユーザ・グループを削除するには、[シナリオ グループ] 表示枠の右にある [グループの削除] ボタンをクリックするか、削除する仮想ユーザ・グループを右クリックして [グループの削除] を選択します。

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] 表示枠からの直接的な仮想ユーザ・グループの変更は、次の手順で行います。

- 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、102 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、『LoadRunner 仮想ユーザ・スクリプトの作成』を参照してください。

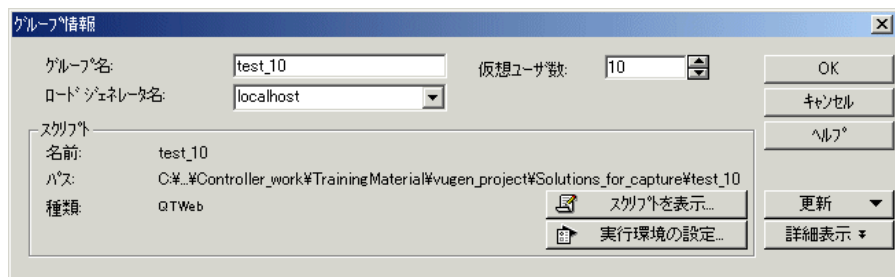
【グループ情報】ダイアログ・ボックスを使った仮想ユーザ・グループの変更

【グループ情報】ダイアログ・ボックスには、仮想ユーザ・グループに関する詳細が表示され、グループの設定を変更できます。

【グループ情報】ダイアログ・ボックスを使った仮想ユーザ・グループの変更は、次の手順で行います。



- 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、102 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、102 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。リストに表示されていないロード・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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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、VuGen のヘルプ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。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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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 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、ロード・ジェネレータ名、または仮想ユーザ数でグループの並べ替えができます。

仮想ユーザ・グループの並べ替えは、次の手順で行います。

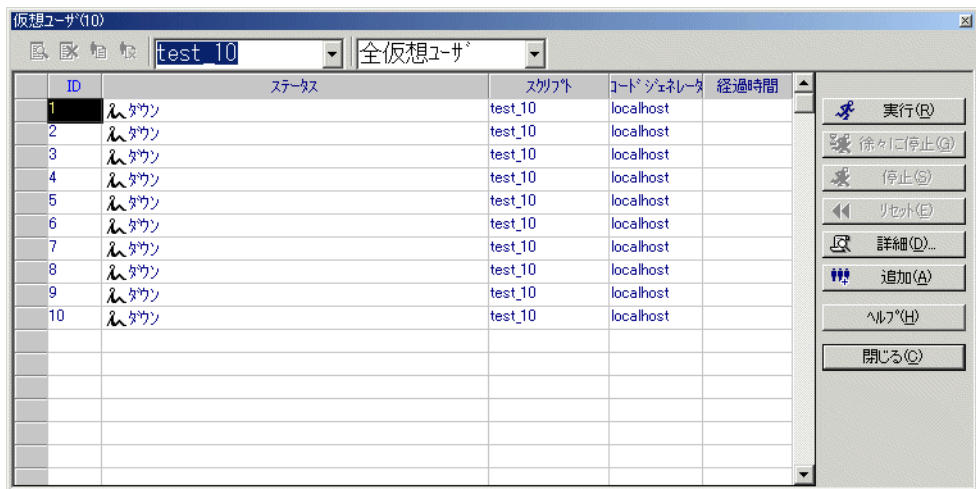
- ▶ グループの並べ替えの基準にするカラムを選択します。カラム見出しをクリックします。
- ▶ または、並べ替えの基準にするカラムの任意の場所を右クリックして、**[グループの並べ替え]** を選択します。**[名前で並べ替え]**、**[パスで並べ替え]**、**[数で並べ替え]**、**[ジェネレータで並べ替え]** のいずれかを選択します。
- ▶ 新しい仮想ユーザ・グループのエントリをコントローラが自動的に並べ替えるようにするには、そのエントリを右クリックして、**[自動並べ替え]** を選択します。

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の定義は、次の手順で行います。

- 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	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た。スクリプトの実行は成功した。

ステータス	説明
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)。
終了中	仮想ユーザは実行を終えたか、停止されて終了中。
停止	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。

[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。

[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。

[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 [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、VuGen のヘルプを参照してください。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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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。

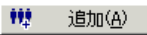
- ▶ [スクリプト名]：この仮想ユーザで使用するスクリプトを選択します。選択したスクリプトが [スクリプト名] カラムに表示されます。
- ▶ [スクリプトのパス]：スクリプト・ディレクトリのパスが表示されます。
- ▶ [参照]：別のディレクトリからスクリプトを選択できます。VB 仮想ユーザ・スクリプトを選択するには、.usr ファイルを探します。
- ▶ [記録]：スクリプトの記録を開始できるように、仮想ユーザ・ジェネレータを開きます。スクリプトの記録の詳細については、『LoadRunner 仮想ユーザ・スクリプトの作成』を参照してください。
- ▶ [実行環境の設定]：[実行環境設定] 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、VuGen のヘルプを参照してください。

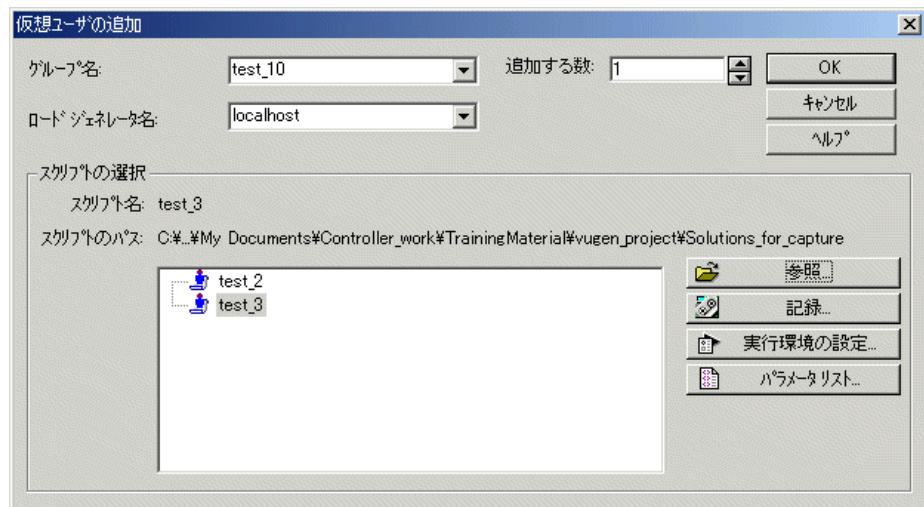
仮想ユーザ・グループへの仮想ユーザの追加

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、214 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・グループへの仮想ユーザの追加は、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、[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。[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、106 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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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、VuGen のヘルプ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、102 ページ「スクリプトの設定」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。実行環境の設定を表示する方法はいくつかあります。

- ▶ [グループ情報] ダイアログ・ボックスで、[実行環境の設定] をクリックします。
- ▶ 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。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。

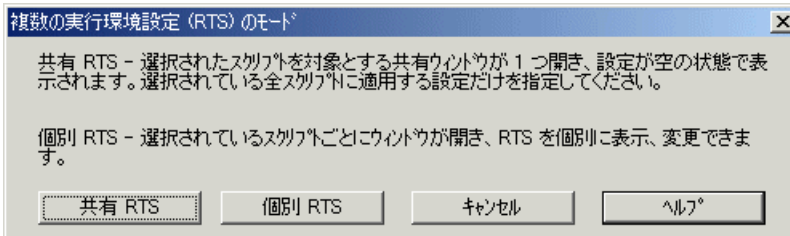
各実行環境の設定の詳細については、『LoadRunner 仮想ユーザ・スクリプトの作成』を参照してください。

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、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修正するために、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せん。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。たとえば、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（たとえば [ブラウザのエミュレーション] ノードの [変更] および [詳細設定] ボタン）はすべて無効になります。

以下のノードは共有モードでは表示されません。

- ▶ [Java 環境の設定 : 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、「プロパティ」カラムと「値」カラムで構成されるテーブルが含まれるノード（たとえば、[Oracle NCA : クライアントのエミュレーション] ノード）。

ロード・ジェネレータの設定

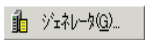
ロード・ジェネレータの属性の設定は、[ロードジェネレータ] ダイアログ・ボックスを使って、ロード・ジェネレータ・リストにロード・ジェネレータを追加しているときに設定できます。また既存のロード・ジェネレータの属性は随時変更できます。

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。ロード・ジェネレータのプロパティを個別に設定するには、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。たとえば、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・ジェネレータの設定は、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。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。[切断] をクリックすると、ロード・ジェネレータの [ステータス] が「準備完了」から「ダウン」に変わります。
- 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。

[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が表示されます。

[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] : [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色で表示されます。

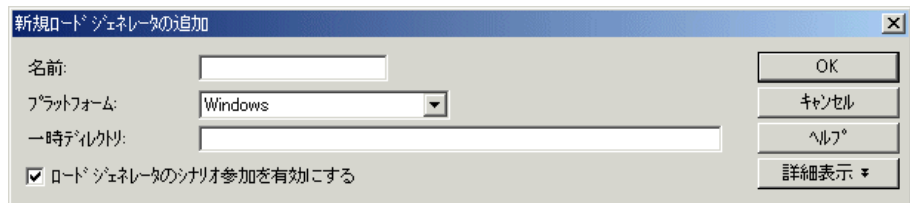
注：コントローラは、Windows ロード・ジェネレータ・マシンの CPU 使用率を監視し、ロード・ジェネレータが過負荷になった場合には、仮想ユーザのロードを自動的に停止します。このダイアログ・ボックス内のアイコンを使用して、マシンの CPU 使用率のステータスを監視できます。ロード・ジェネレータの CPU 使用率に問題が生じると、ロード・ジェネレータ名の左側にあるアイコンに黄色の線が表示されます。マシンが過負荷になると、アイコンに赤色の線が表示されます。

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・ジェネレータの追加またはロード・ジェネレータ情報の変更は、次の手順で行います。

- 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 ロード・ジェネレータを削除するには、[削除] をクリックします。
- 8 [閉じる] 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] : 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。

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、仮想ユーザのステータス、仮想ユーザの制限、接続ログ（エキスパート・モード）、ファイアウォール、そして WAN エミュレーションです。

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・ジェネレータの設定は、次の手順で行います。

- 1 [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・ボックスで、[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、[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 ステータス]、[端末サービス]、[WAN エミュレーション]、[ファイアウォール]（ロード・ジェネレータがローカルホストでない場合）の各タブが表示されます。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、633 ページ「エキスパート・モードでの作業」を参照してください。
- 2 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 設定するロード・ジェネレータ設定を含んだタブを選択し、設定を行います。
- 4 [OK] をクリックして、[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。

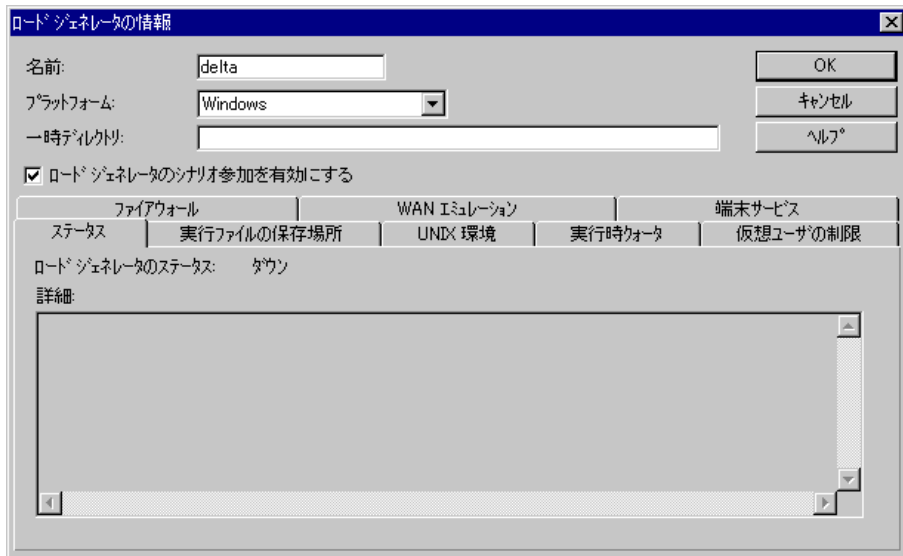
[プラットフォーム]: ロード・ジェネレータを実行するプラットフォームのタイプを選択します。

[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・ジェネレータのステータスの詳細を表示するには、[ステータス] タブを選択します。

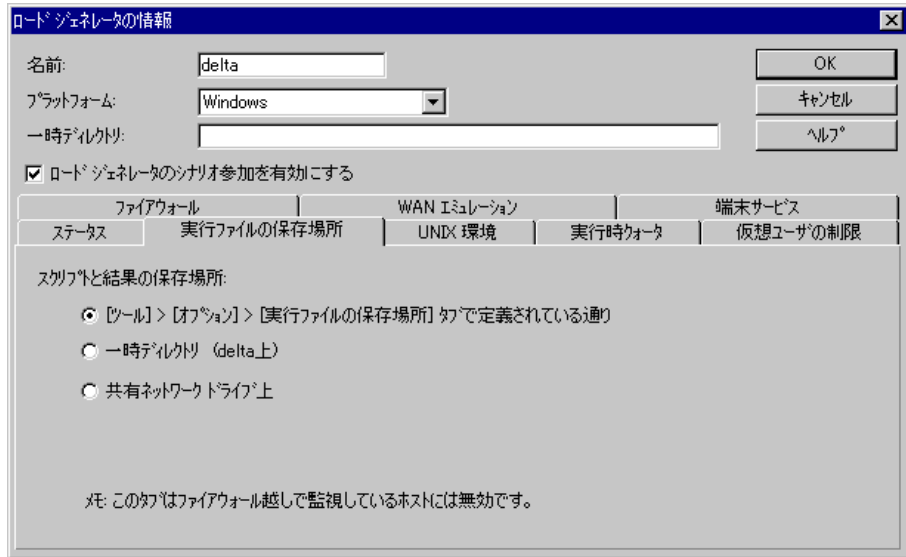


[ロード ジェネレータのステータス]: ロード・ジェネレータのステータスが表示されます。

[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

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】：次のオプションのいずれかを選択します。

- ▶ [[ツール] > [オプション] > [実行ファイルの保存場所] 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

各 UNIX ロード・ジェネレータのログイン・パラメータおよびシェルの種類を設定するには、[UNIX 環境] タブを選択します。

The screenshot shows the 'Load Generator Information' dialog box with the 'UNIX Environment' tab selected. The 'Name' field contains 'delta', and the 'Platform' dropdown is set to 'UNIX'. The 'Enable scenario participation for load generators' checkbox is checked. The 'Logon' section has an unchecked 'Name' checkbox and an unchecked 'Use lowercase characters in logon name' checkbox. The 'Shell Environment Settings' section has an unchecked 'Do not use RSH' checkbox, a 'Standard shell' dropdown set to 'csh', and an empty 'Initialization command' field. A note at the bottom states: 'Note: This tab is disabled when the virtual user is running, or when monitoring the host through a firewall.' The 'Standard Settings' button is visible at the bottom right.

[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、たとえば **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 633 ページ「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・ジェネレータでの標準設定のシェルとして、**csh** (C シェル—標準設定)、**bsh** (Bourne シェル)、または **ksh** (Korn シェル) を選択します。

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 (たとえば、**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。たとえば、**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。

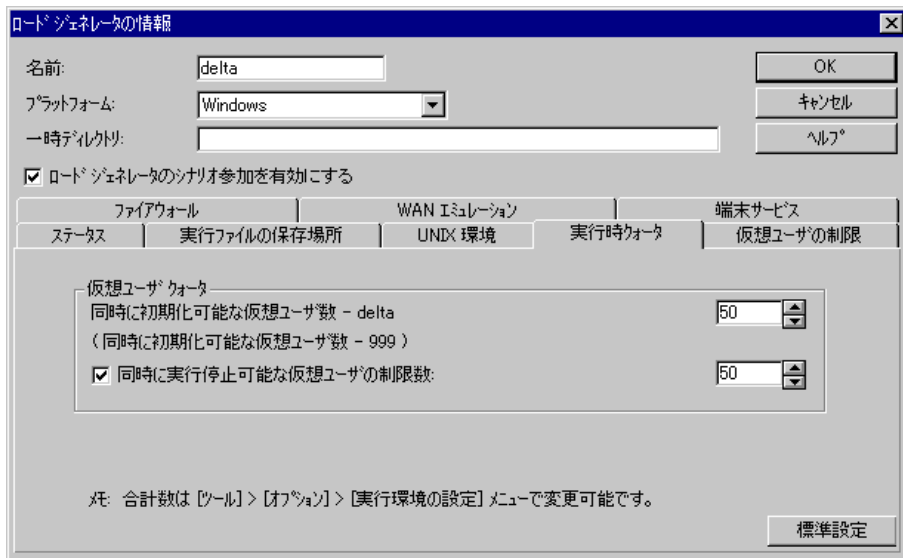
```
. .profile;
```

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、仮想ユーザを一度に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】：

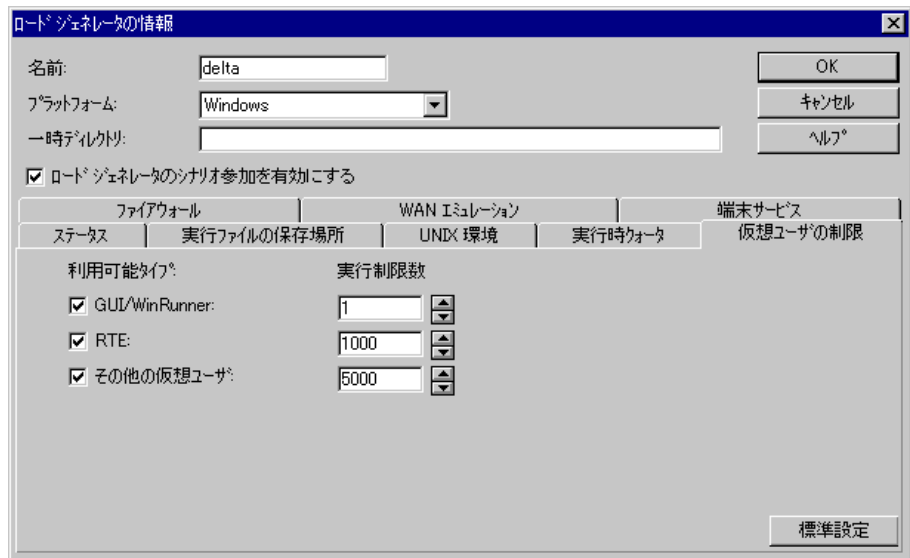
- ▶ 【同時に初期化可能な仮想ユーザ数 - 現在のロード・ジェネレータ】：現在のロード・ジェネレータで一度に初期化できる仮想ユーザの最大数を選択します。
- ▶ 【同時に実行停止可能な仮想ユーザの制限数】：現在のロード・ジェネレータで一度に停止できる仮想ユーザの最大数を選択します。

【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・ジェネレータが実行できる GUI 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。

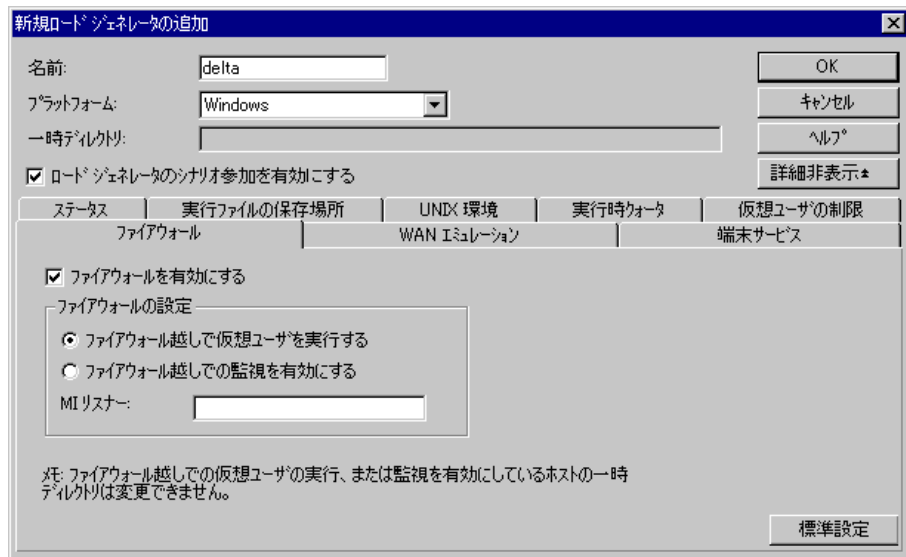
[実行制限数]: ロード・ジェネレータが実行する各タイプの仮想ユーザの最大数を選択します。

[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、[ヘルプ] > [LoadRunner コントローラのバージョン情報] を選択します。

[ファイアウォール] タブ

ファイアウォール越しの監視または実行ができるようにするには、[ファイアウォール] タブを選択します。



[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] :

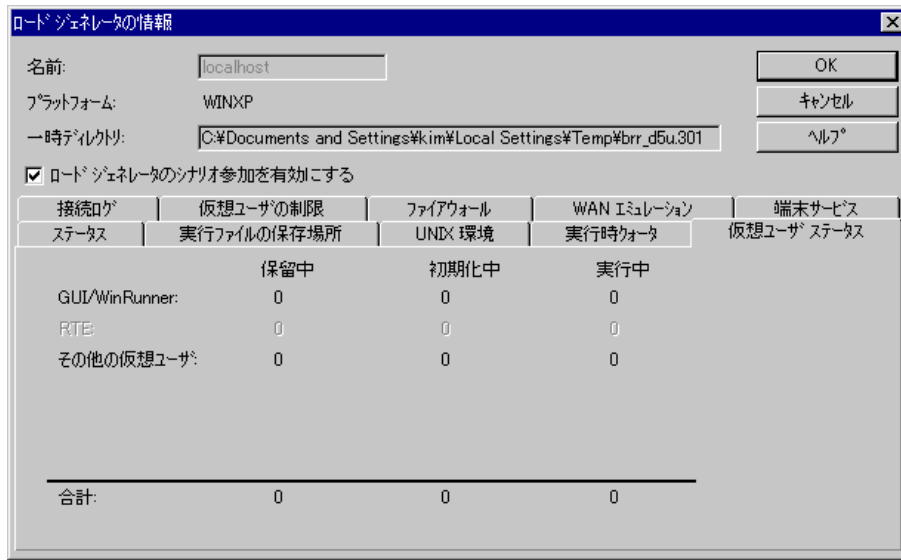
- ▶ [ファイアウォール越しでの監視を有効にする] : LoadRunner がファイアウォール越しにロード・ジェネレータ・マシンを監視できるようにします。
- ▶ [ファイアウォール越しで仮想ユーザを実行する] : LoadRunner がファイアウォールの外にあるロード・ジェネレータで仮想ユーザを実行できるようにします。

[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

選択したロード・ジェネレータ・マシンに接続されたすべての仮想ユーザのステータスを見るには、【仮想ユーザステータス】タブを選択します。



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。

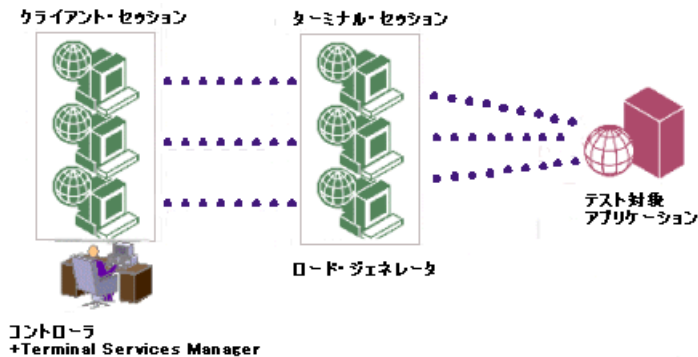
ターミナル・サービスの設定

LoadRunner の Terminal Services Manager を使用すると、ターミナル・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、ターミナル・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 仮想ユーザごとにターミナル・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。

ターミナル・サービスについて

ターミナル・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。ターミナル・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。ターミナル・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、ターミナル・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、ターミナル・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。



Terminal Services Manager

ターミナル・サーバ・クライアントでは、複数のターミナル・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ターミナルの数と（十分な数のターミナル・セッションが動作している場合）、各ターミナル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、次の手順で行います。

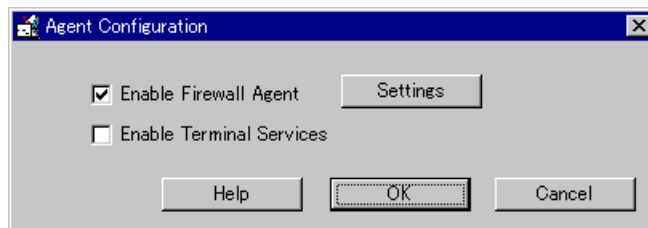
- ▶ ロード・ジェネレータ・マシンでターミナル・サーバ・エージェントを設定します。
- ▶ コントローラ・マシンでターミナル・クライアント・セッションを開始します。
- ▶ LoadRunner Terminal Services Manager を使って、ターミナル・サーバで仮想ユーザを分配します。

ターミナル・サーバ・エージェントの設定

ロード・ジェネレータ・マシンのターミナル・サーバ・エージェントを設定する前に、ターミナル・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、『**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・マシンでターミナル・サーバ・エージェントを設定するには、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] > [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。[Agent Configuration] ダイアログ・ボックスが開きます。



- 3 **[Enable Terminal Services]** チェック・ボックスを選択します。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、**[Enable Firewall Agent]** チェック・ボックスを選択し、**[Settings]** をクリックします。エージェントの設定については、249 ページ「エージェントの設定」を参照してください。

[OK] をクリックします。

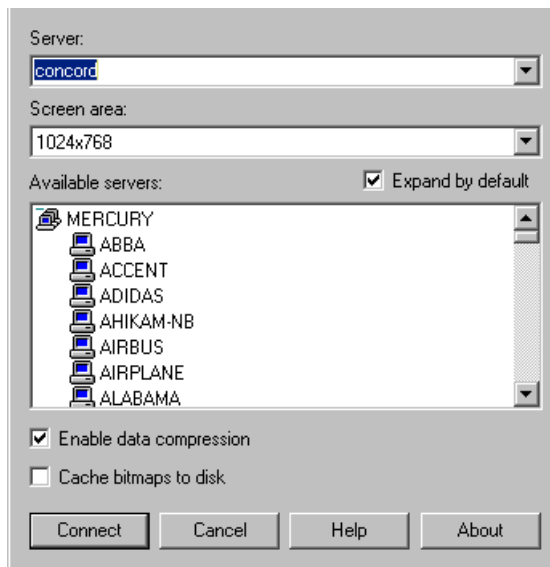
- 4 デスクトップのショートカットをダブルクリックするか、**[スタート]** > **[プログラム]** > **[LoadRunner]** > **[LoadRunner Agent Service/Process]** を選択して LoadRunner エージェントを再起動します。

ターミナル・クライアント・セッションの開始

ターミナル・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。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、Terminal Server 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・マシンでターミナル・クライアント・セッションを開くには、次の手順で行います。

- 1 **[スタート]** > **[プログラム]** > **[Terminal Services Client]** > **[Terminal Services Client]** を選択します。**[Terminal Services Client]** ダイアログ・ボックスが開きます。



- 2 [Server] ボックスで、ターミナル・サーバの 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ターミナル・サーバを選択します。
- 3 [Screen Area] で、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 [Connect] をクリックします。Windows のログオン・ダイアログ・ボックスが開きます。



- 5 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。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ターミナルごとに、ターミナル・サーバ・クライアント・セッションを開く必要があります。

ターミナル・サーバでの仮想ユーザの分配

ターミナル・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。

ロードジェネレータの情報

名前: spark

プラットフォーム: Windows

一時ディレクトリ:

ロードジェネレータのシナリオ参加を有効にする

ステータス	実行ファイルの保存場所	UNIX 環境	実行時コマンド	仮想ユーザの制限
ファイアウォール		WAN エミュレーション		端末サービス

端末サービス マネージャを有効にする

端末数: 2

各端末に割り当てる仮想ユーザ数の上限: 50

標準設定

[名前]: ターミナル・サーバの名前。実行するインスタンスの数に関係なく、ターミナル・サーバの名前を [ロードジェネレータ] リストに追加する必要があるのは1回だけです。

[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。

[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。

[端末サービス マネージャを有効にする]: ターミナル・サーバ上で動作しているロード・ジェネレータへのターミナル・サービス設定の適用を有効にします。

[端末数]: 負荷テストで使用するターミナルの数を入力します。シナリオ実行中に仮想ユーザを実行するターミナルごとに、ターミナル・クライアント・セッションを開く必要があります。

[各端末に割り当てる仮想ユーザ数の上限]：各ターミナルで実行する仮想ユーザの最大数を入力します。ターミナル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。

[標準設定]：ターミナルの数を 2 に、ターミナル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

コントローラとターミナル・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、ターミナル・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] > [LoadRunner] > [LoadRunner Agent Service/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。

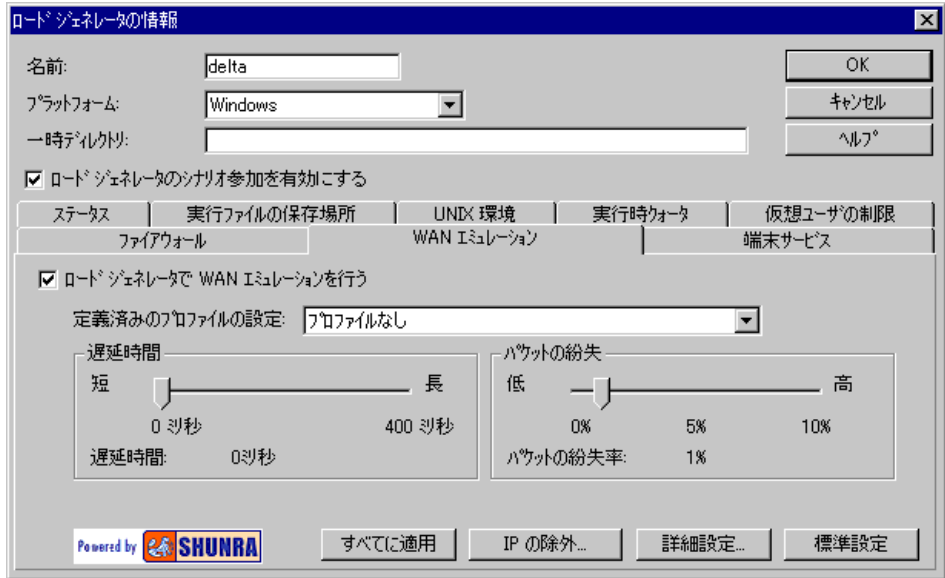
WAN エミュレータのセットアップ

Shunra WAN エミュレータを使用するには、LoadRunner Controller 7.8 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、『**LoadRunner コントローラ・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。ライセンス情報については、マーキュリー・インタラクティブのカスタマー・サポート Web サイト (<http://support.mercuryinteractive.com>) 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 : 100 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 0ms です。

[パケットの紛失率] : 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。

[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。

[IP の除外] : **[IP 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 IP アドレスを除外できます。詳細については、99 ページ「WAN エミュレーションからの IP アドレスの除外」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、98 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望みの値にします。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のフラグメンテーションが発生する（最大伝送単位が短いため）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 IP アドレスの除外

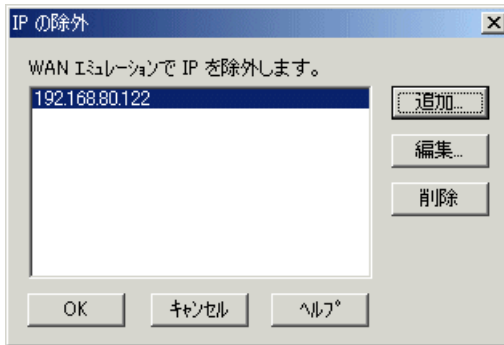
状況に応じて、WAN エミュレーションから特定の IP アドレスを除外できます。つまり、指定された IP アドレスに対するトラフィックに影響を与えないように、WAN エミュレータを設定できます。エミュレーションの影響を受けないネットワーク・トラフィックは、WAN の影響を一切受けず、WAN エミュレーション監視レポートにも含まれません。

エミュレートされる WAN から IP アドレスを除外することがあるのは、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。

[IP の除外] ダイアログ・ボックスについて

特定の IP アドレスを 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 [IP の除外] を選択します。



[追加] : [IP の追加] ダイアログ・ボックスが開きます。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 IP アドレスを [IP を除外] リストから選択し、[マシンを編集] ダイアログ・ボックスでその IP アドレスに変更を加えることができます。

[削除] : IP アドレスを [IP を除外] リストから削除します。

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、[IP を除外] 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再開

シナリオ実行中、いつでも WAN エミュレーションの停止と再開ができます。

WAN エミュレーションの停止と再開は、次の手順で行います。

- 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再開するには、[シナリオ] > [WAN エミュレーションの再開] を選択します。

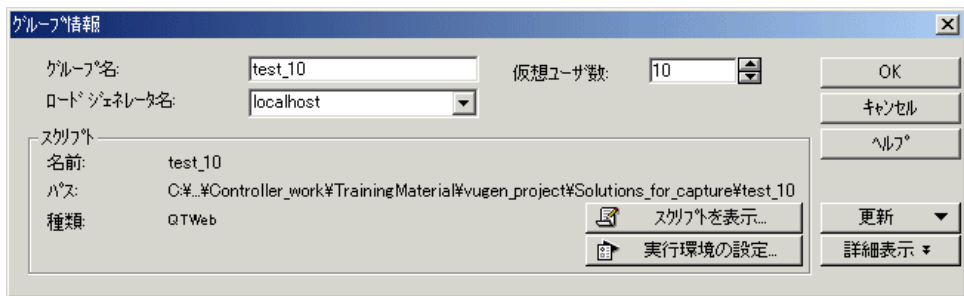
スクリプトの設定

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と詳細の表示は、次の手順で行います。

- 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グループ名、パス、およびタイプが表示されます。

 詳細(D)...



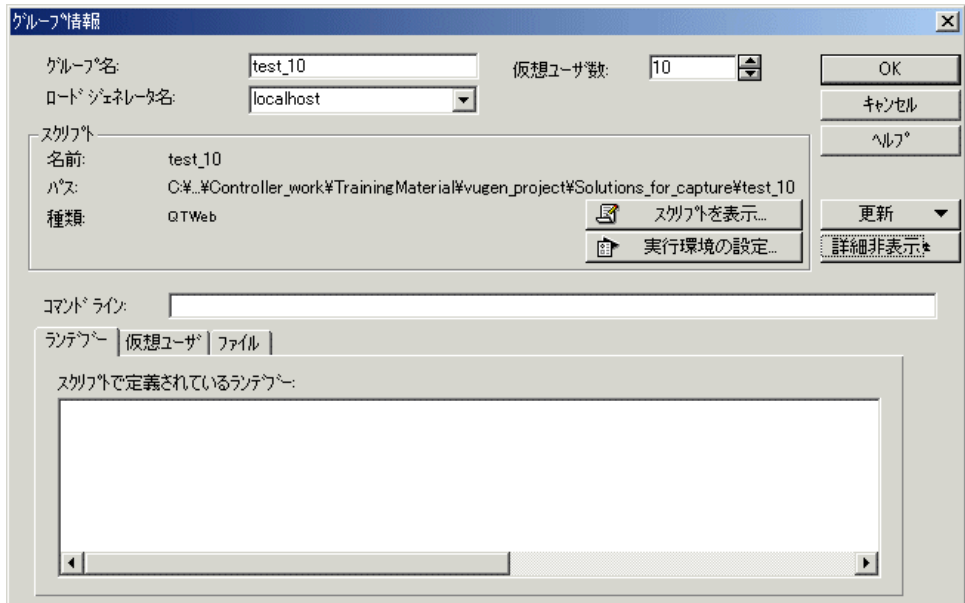
- 2 [実行環境の設定] をクリックして、スクリプトの実行環境を設定します（任意）。詳細については、71 ページ「仮想ユーザ実行環境の設定」を参照してください。

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の VuGen が起動します。スクリプトの編集の詳細については、『LoadRunner 仮想ユーザ・スクリプトの作成』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

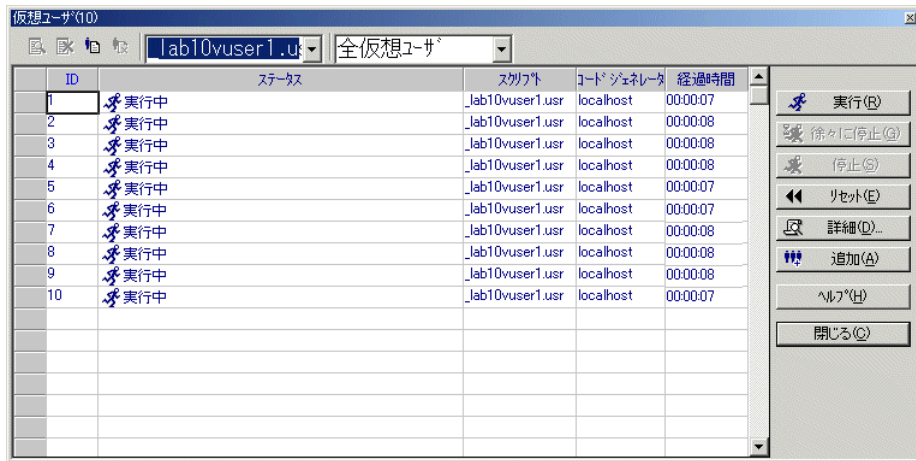
コマンド・ライン引数の値をスクリプトに渡す方法の詳細については、『LoadRunner 仮想ユーザ・スクリプトの作成』を参照してください。

- 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。
- 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。

- 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。



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、106 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。

- 2 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の VuGen が起動します。スクリプトの編集の詳細については、『**LoadRunner 仮想ユーザ・スクリプトの作成**』を参照してください。
- 3 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を記録中に行うには、[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。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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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。

たとえば、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。たとえば、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、以下の項目について説明します。

- ▶ 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。

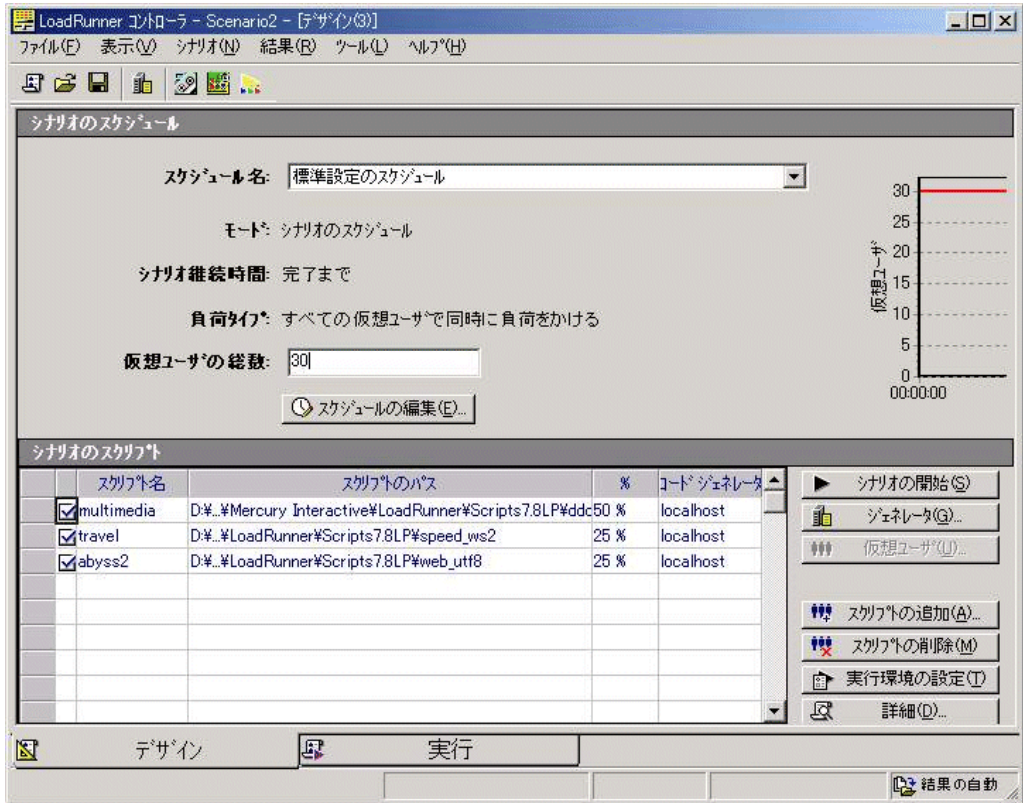
新しいシナリオを作成するときに [新規シナリオ] ダイアログ・ボックスで [スクリプト間で仮想ユーザを分配するのに [パーセントモード] を使用する] を選択することによって、パーセントモードに直接入ることができます。また、[シナリオ] > [シナリオを次のモードに切り替える：パーセントモード] を選択することによって、仮想ユーザ・グループ・モードで作成したシナリオをパーセントモードに切り替えることができます。

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。[プレビューのロード] 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、149 ページの [スケジュールの編集] ダイアログ・ボックスを参照してください。

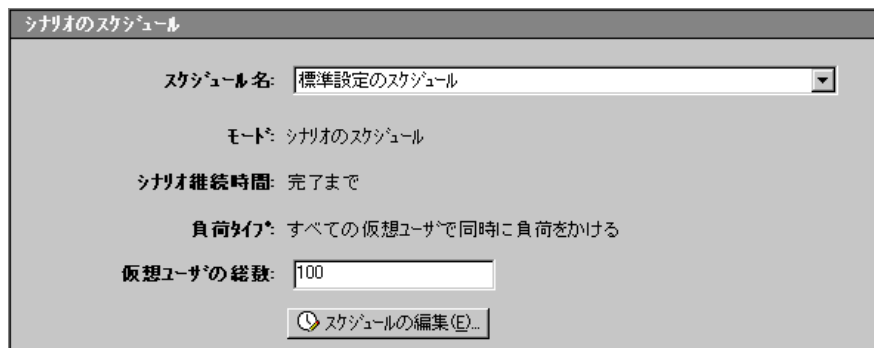
[シナリオのスクリプト] 表示枠には、すべての有効および無効な仮想ユーザ・スクリプト、各スクリプトのパス、ロード・ジェネレータ・マシン、各スクリプトに割り当てられた仮想ユーザの総数の割合のリストが表示されます。

仮想ユーザ・スクリプトまたはシナリオに対して、以下の操作が可能です。

- ▶ 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、および追加したマシンの設定
- ▶ 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

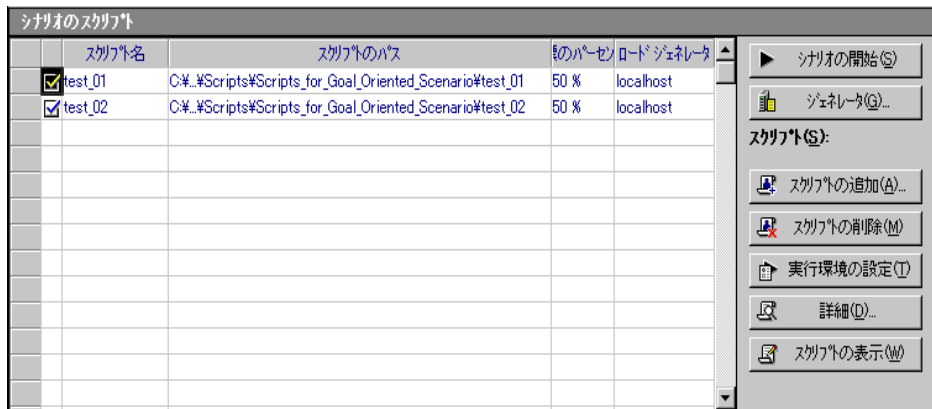
パーセントモードでシナリオを作成する場合、スクリプトごとに仮想ユーザの数を定義するのではなく、シナリオ内で使う仮想ユーザの総数を定義します。[シナリオのスケジュール] ウィンドウに仮想ユーザの総数を入力します。



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。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。

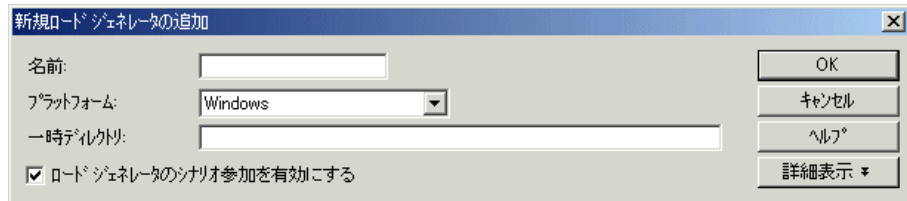
- ▶ 仮想ユーザの総数のうち、スクリプトに割り当てる割合。
- ▶ 仮想ユーザがスクリプトを実行するロード・ジェネレータ。

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] 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を追加] 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] 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] 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、136 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。たとえば、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、75 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。

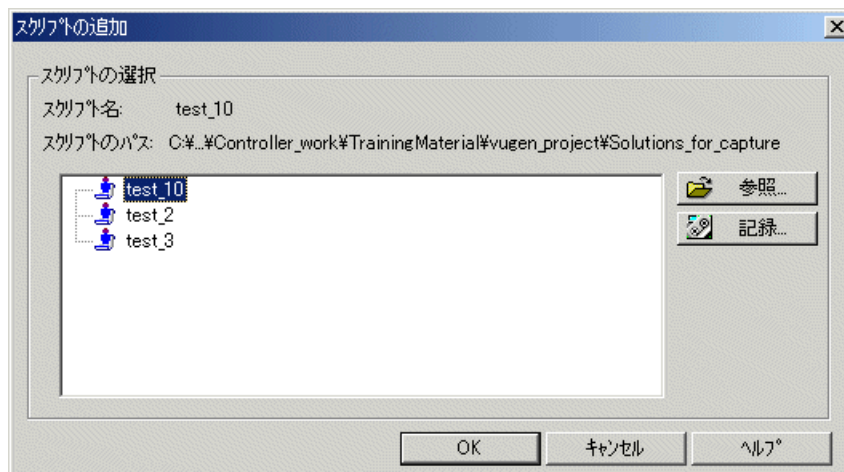
シナリオに参加しているすべてのロード・ジェネレータのグローバル設定を構成するには、LoadRunner の [オプション] ダイアログ・ボックスを使います。詳細については、第 10 章「シナリオの設定」を参照してください。

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 [パス] ボックスの右の [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、106 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] ウィンドウに新しいスクリプト情報を入力します。

[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、『LoadRunner 仮想ユーザ・スクリプトの作成』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。

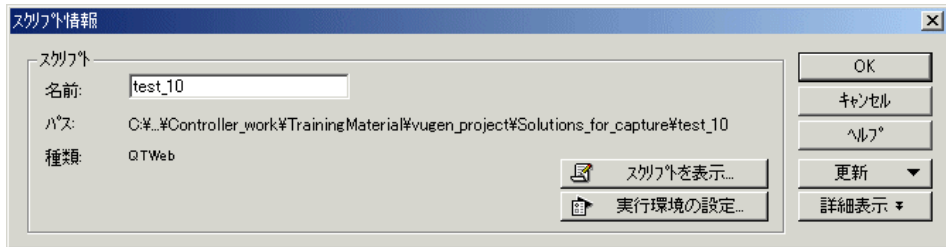
スクリプト情報の表示

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。

 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。

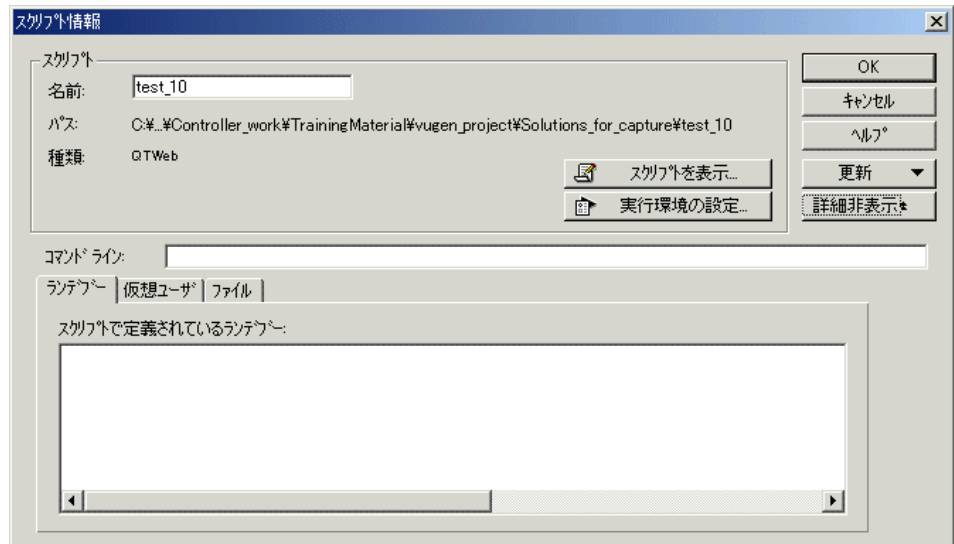
実行環境の設定の詳細については、『LoadRunner 仮想ユーザ・スクリプトの作成』を参照してください。

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、『LoadRunner 仮想ユーザ・スクリプトの作成』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

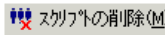
コマンド・ライン引数の値をスクリプトに渡す方法の詳細については、『**LoadRunner 仮想ユーザ・スクリプトの作成**』を参照してください。

- 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。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。

[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。
- ▶ **[スクリプトの表示]** : スクリプトを編集できるように、を開きます。スクリプトの編集の詳細については、『**LoadRunner 仮想ユーザ・スクリプトの作成**』を参照してください。
- ▶ **[実行環境の設定]** : **[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。実行環境の設定の詳細については、VuGen のヘルプ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。たとえば、`-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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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。

パーセントモード・シナリオの詳細については、スクリプト情報の表示を参照してください。ゴール指向シナリオの詳細については、スクリプト情報の表示を参照してください。

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切り替える：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、109 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。

シナリオ切り替えの前に、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 モードの切り替え時に警告を表示] 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、以下の項目について説明します。

- ▶ 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 128 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、128 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、132 ページ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

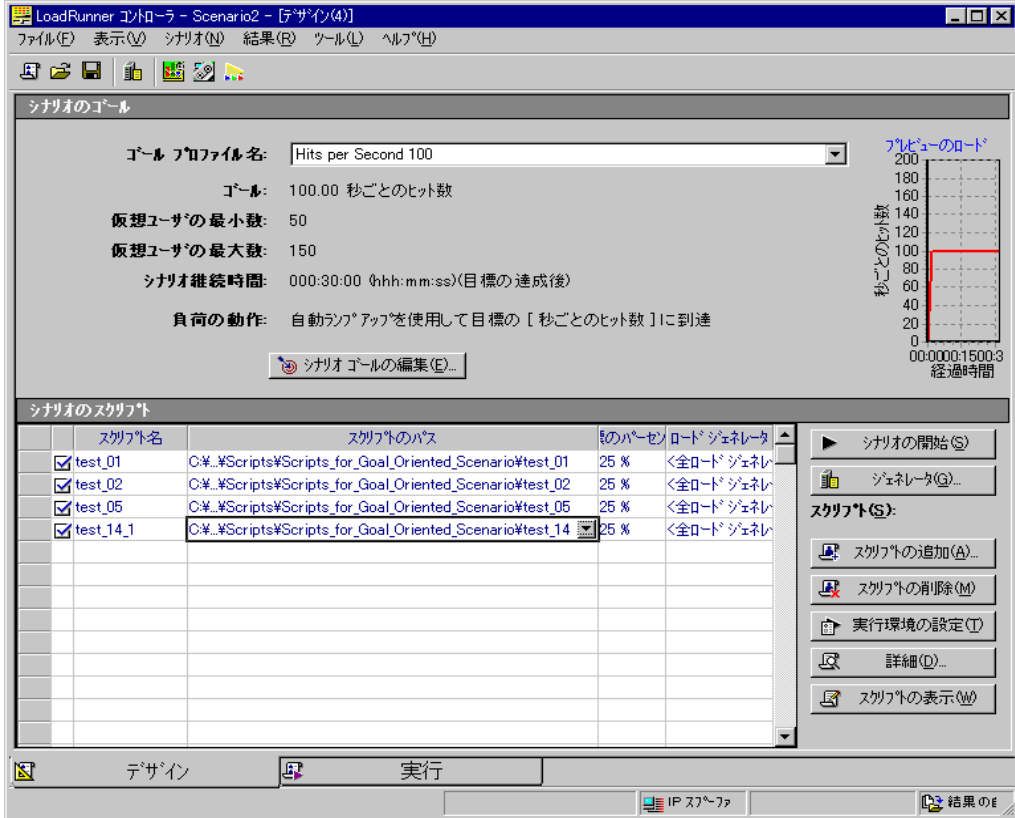
指定のトランザクション応答時間を超えずに、同時に実行できる仮想ユーザ数をテストする場合は、トランザクション応答時間のゴール・タイプを指定することをお勧めします。テストするスクリプトのトランザクション名と、実行する LoadRunner の最小および最大仮想ユーザ数の範囲を指定します。トランザクション応答時間は、あらかじめ定義されているしきい値に合わせて指定します。たとえば、電子商取引サイトへのログインで顧客を5秒以上待たせないようにしたい場合には、最大許容トランザクション応答時間を5秒に指定します。最小および最大仮想ユーザ数を、同時に処理できるようにしたい最小および最大顧客数に設定します。

シナリオが、定義した最大トランザクション応答時間に達さなければ、想定している同時サービス顧客数にサーバは適切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：トランザクション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ョン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応答時間です。ゴール・タイプの定義の詳細については、128 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、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、「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。

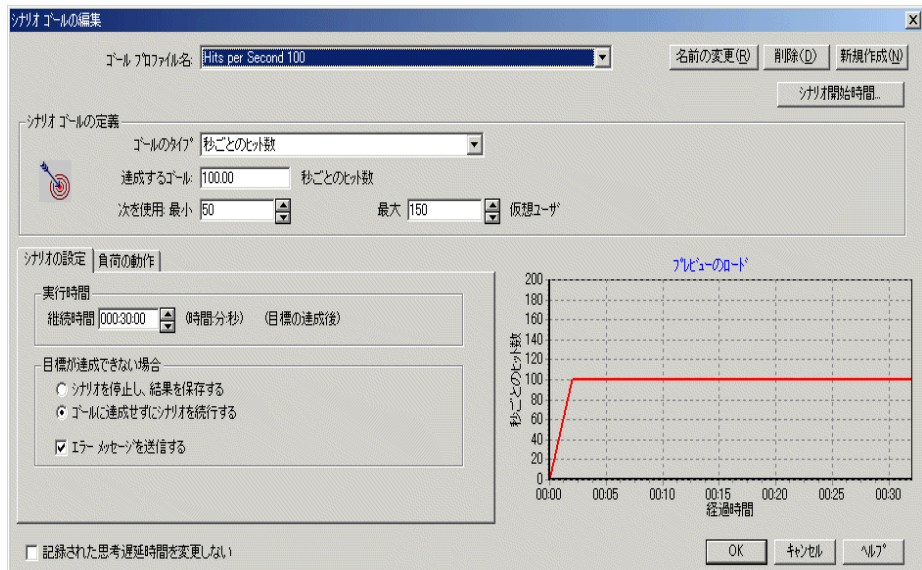
- ▶ ゴール・プロファイル名とゴール・タイプを定義する
- ▶ 新規のスクリプトをシナリオに追加して設定する
- ▶ 1つまたは複数のロード・ジェネレータ・マシンをスクリプトに追加し、そのマシンを設定する
- ▶ シナリオのスクリプトを有効または無効にする
- ▶ シナリオの継続時間とランプ・アップ動作を定義する
- ▶ シナリオを実行する
- ▶ シナリオを停止する
- ▶ シナリオをリセットする
- ▶ シナリオの実行結果に関する設定を行う

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。
 - ▶ [仮想ユーザ] を選択した場合は、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] を選択した場合は、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] を選択した場合は、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] を選択した場合は、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。
- ▶ [分ごとのページ数] を選択した場合は、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。

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] タブでは、目標達成後、シナリオの実行を継続する時間の長さを選択します。
- 5 LoadRunner が定義された目標を達成できなかった場合、シナリオの実行を停止して、シナリオの結果を保存するか、シナリオの実行を続行するかを選択します。目標が達成されなかった場合に、LoadRunner がエラー・メッセージを送信するようにするには、[エラーメッセージを送信する] を選択します。
- 6 [負荷の動作] タブを選択します。[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] ゴール・タイプを選択した場合、LoadRunner が、各バッチで標準設定の仮想ユーザ数を自動的に追加することによって、あるいは一定時間経過後に、目標を達成するようにします。[分ごとのページ数]、[仮想ユーザ]、または [秒ごとのヒット数] ゴール・タイプを選択した場合、LoadRunner が、各バッチで標準設定の仮想ユーザ数を自動的に追加することによって、一定時間経過後に、あるいは徐々の増加（時間 x ごとに仮想ユーザ数 / ページ数 / ヒット数を x ずつ）によって、目標を達成するようにします。

- 7 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。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。
- 8 **[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**]: [**ゴール プロファイル名の変更**] 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**]: [**ゴール プロファイル名の変更**] 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 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)
- ▶ 仮想ユーザ
- ▶ 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
- ▶ トランザクション応答時間

[シナリオの設定] タブ

131 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。

[負荷の動作] タブ

131 ページ「[負荷の動作] タブについて」を参照してください。

[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。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。

[プレビューのロード]：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として、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 / トランザクション数」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、最初に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。

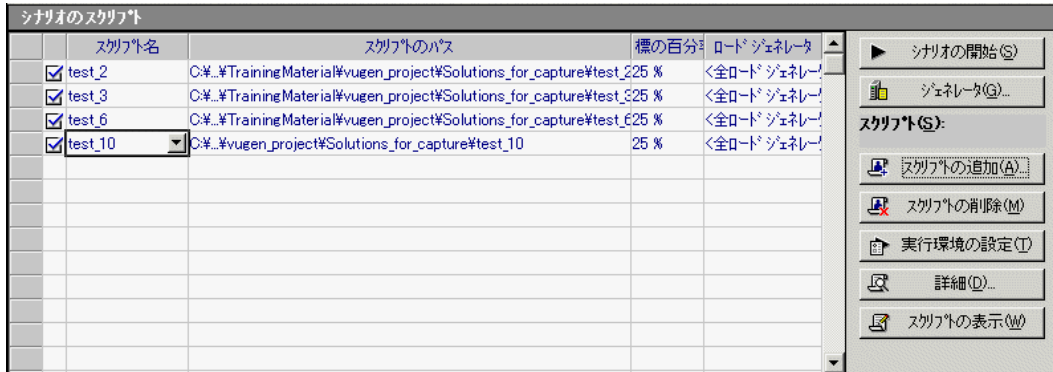
コントローラが定義した最大数の仮想ユーザを起動してもゴールを達成できない場合は、LoadRunnerは仮想ユーザごとの目標ヒット数、トランザクション数、またはページ数を再計算し、最大数の仮想ユーザを同時に実行して、もう一度定義された目標の達成を試みます。

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。



[目標の百分率] 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。

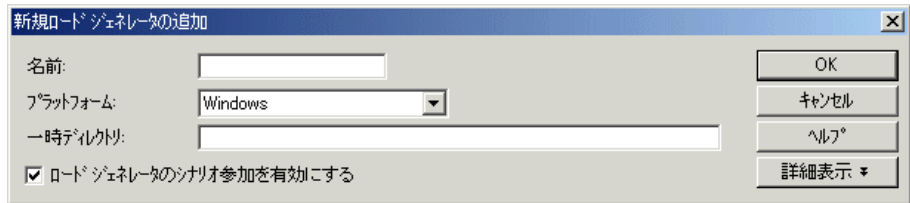
スクリプトに割り当てる仮想ユーザの割合の変更は、次の手順で行います。

スクリプトの [目標の百分率] 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を変更するには、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] 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。



[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。

[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。

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。

- 3 [OK] をクリックして、[新規ロードジェネレータを追加] 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] 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] 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。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。たとえば、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。

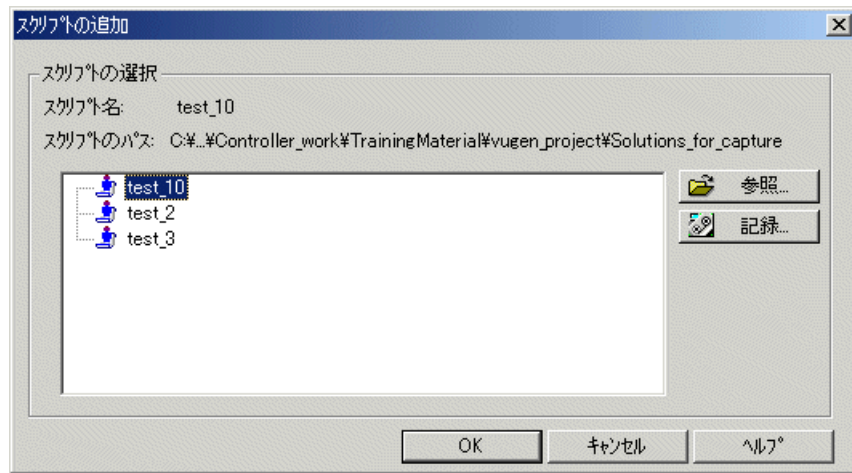
注：負荷分散が行えるのは、ゴール指向シナリオとパーセントモードのマニュアル・シナリオだけです。

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 [パス] ボックスの右の [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、106 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

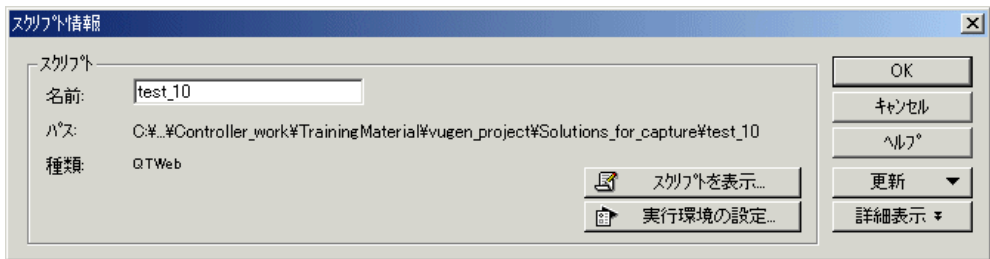
リストに追加したスクリプトは、選択して詳細表示したり、編集したり、有効化または無効化したり、実行環境の設定を変更したりできます。

スクリプトの詳細表示は、次の手順で行います。



詳細(D)...

- 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。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。

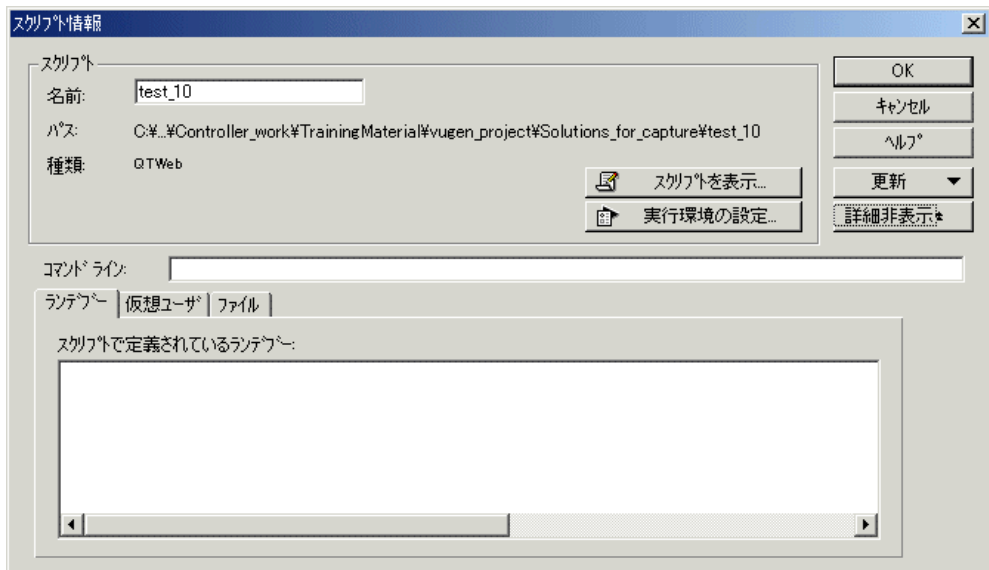
実行環境の設定の詳細については、『LoadRunner 仮想ユーザ・スクリプトの作成』を参照してください。

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、『LoadRunner 仮想ユーザ・スクリプトの作成』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。

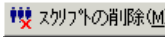
- 4 [詳細表示] をクリックして、[スクリプト情報] ダイアログ・ボックスを拡張すると、さらに多くのスクリプト情報が表示されます。



- 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、『LoadRunner 仮想ユーザ・スクリプトの作成』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。

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、以下の項目について説明します。

- ▶ 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、[プレビューのロード] グラフに表示されます。

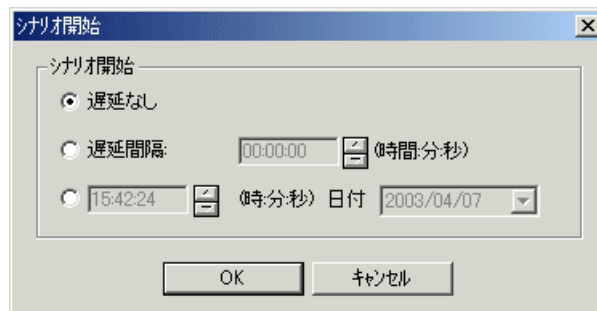
注：仮想ユーザ・スクリプトのランデブー・ポイントは、スケジュールが設定されたシナリオに干渉します。スクリプトにランデブー・ポイントが含まれている場合、シナリオはスケジュールどおりに実行されません。

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 X] 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 (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。
- ▶ [(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] ダイアログ・ボックスを使用して、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。

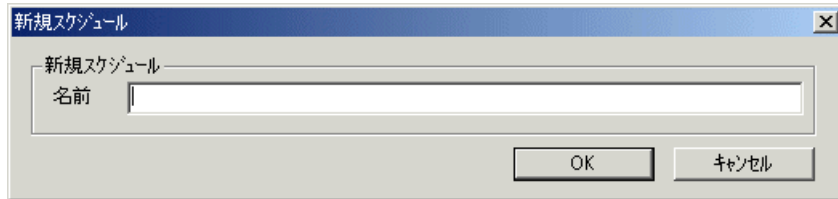
スケジュールの選択

[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 ([Slow Ramp Up] または [Ramp Up]) 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。

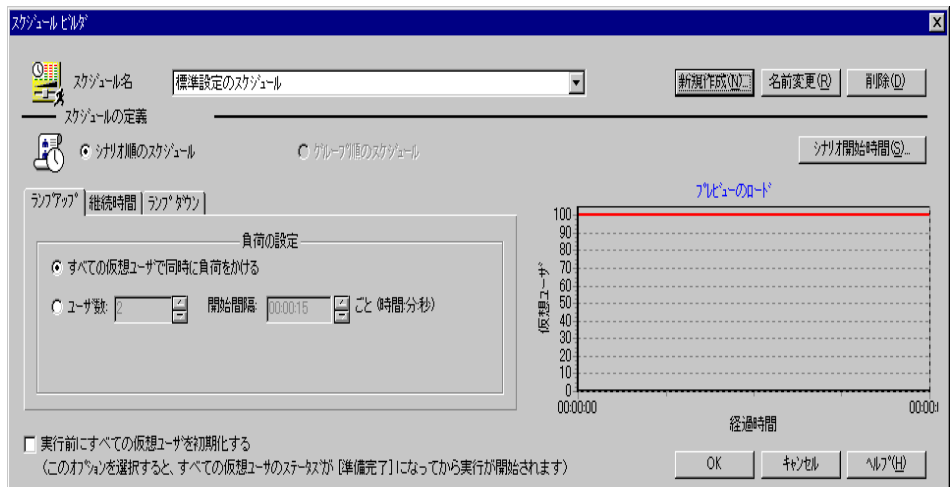
- 1 [シナリオのスケジュール] 表示枠の [スケジュール名] ボックスから [**新規スケジュール**] を選択します。[新規スケジュール] ダイアログ・ボックスが表示されます。



- 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、[Slow Ramp Up] または [Ramp Up] 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。

注：[シナリオ開始] ダイアログ・ボックスを使用して、後でマニュアル・シナリオまたはゴール指向シナリオを開始するよう LoadRunner に指示できます。

[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」、「Ramp Up」、「Slow Ramp Up」の3つの標準設定の名前が表示されます。「Ramp Up」は仮想ユーザを一定の割合で段階的に開始します。「Slow Ramp Up」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・ボックスでシナリオの開始時間を遅らせることができます。

[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
 - ▶ [継続時間] タブ
 - ▶ [ランプダウン] タブ
- ▶ [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、スケジュール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
 - ▶ [ランプアップ] タブ

- ▶ [継続時間] タブ
- ▶ [ランプ ダウン] タブ

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。

[**プレビューのロード**]：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。

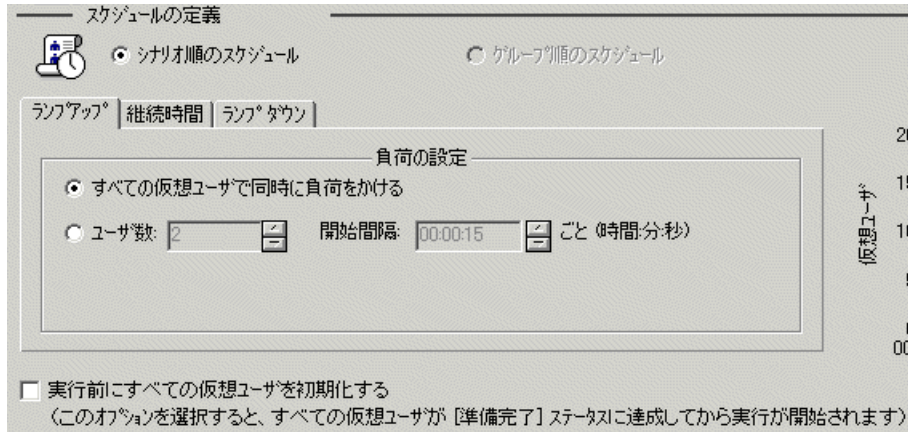
シナリオのスケジュール

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。

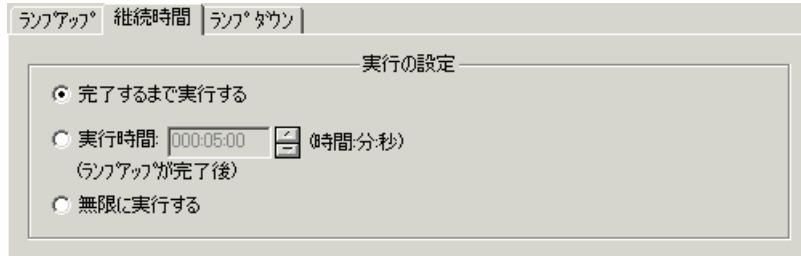


- 2 シナリオを開始する方法を指定するには、[ランプアップ] タブをクリックします。次のオプションのいずれかを選択します。
 - ▶ [すべての仮想ユーザで同時に負荷をかける]：シナリオ内のすべての仮想ユーザを同時に開始します。
 - ▶ [ユーザ数：X 開始間隔：X ごと (時間：分：秒)]：指定された数の仮想ユーザを同時に開始し、仮想ユーザのランプ・アップの開始間隔として指定された時間待機します。

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。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あります。

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。



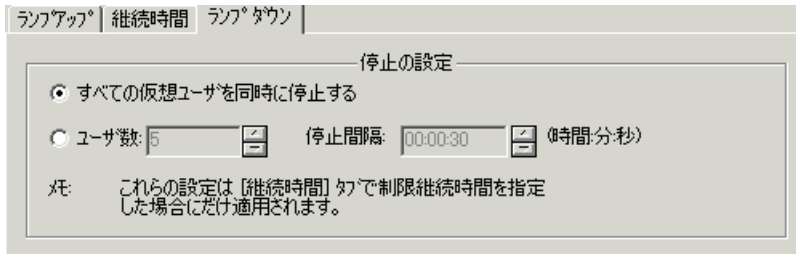
次のオプションのいずれかを選択します。

- ▶ **[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ランプ・アップされ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。

- 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 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。

- 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

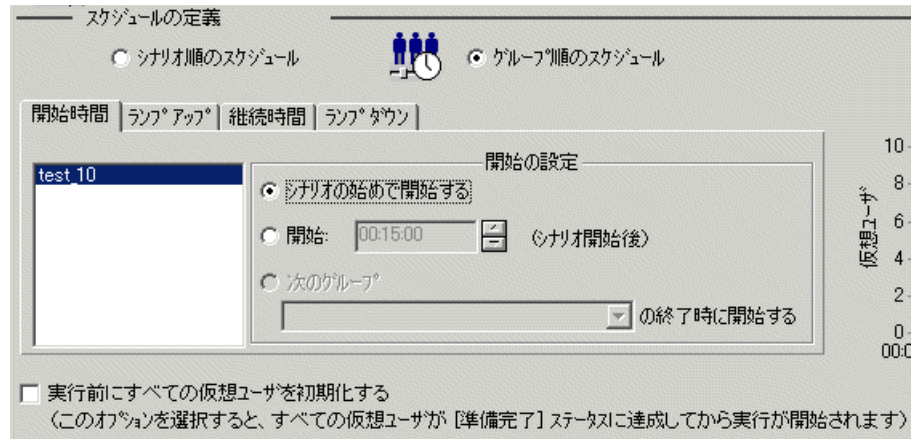
仮想ユーザ・グループの作成後、次の項目を設定して、グループのスクリプト実行スケジュールを設定できます。

- ▶ 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。

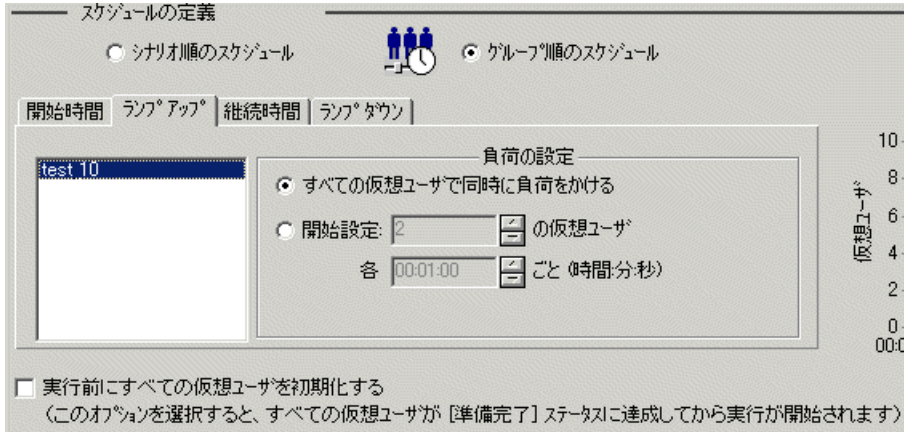
仮想ユーザ・グループのスケジュールの作成は、次の手順で行います。

- 1 [グループ順のスケジュール] オプションを選択します。



- 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、[開始時間] タブをクリックします。次の3つのオプションからいずれかを選択します。
 - ▶ [シナリオの始めて開始する]
 - ▶ [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。



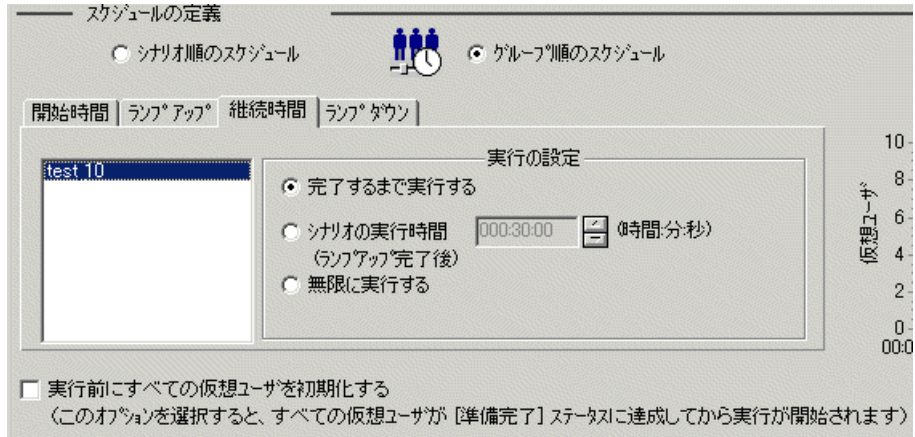
次のオプションのいずれかを選択します。

- ▶ [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。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あります。

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。



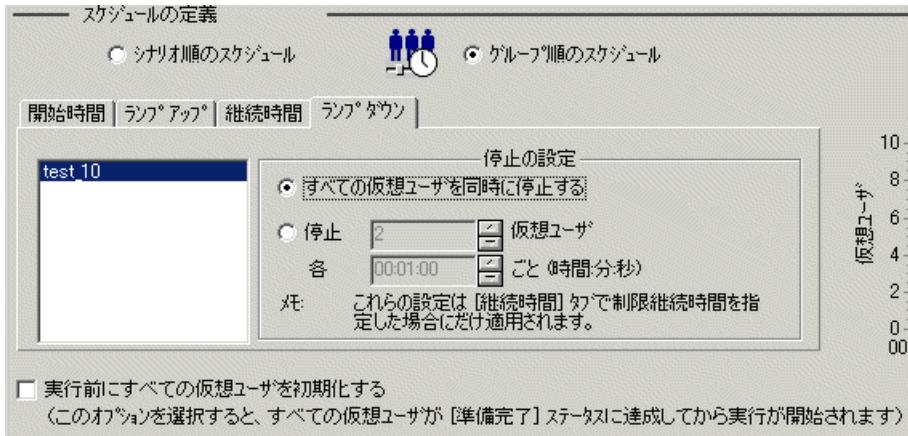
次のオプションのいずれかを選択します。

- ▶ **[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ランプ・アップされ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。

- 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。
- ▶ [ユーザ数：X 停止間隔：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。たとえば、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。

手作業で制御される仮想ユーザの詳細については、214 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、以下の項目について説明します。

- ▶ 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、162 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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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

たとえば、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、162 ページ「ランデブー・ポリシーの設定」を参照してください。

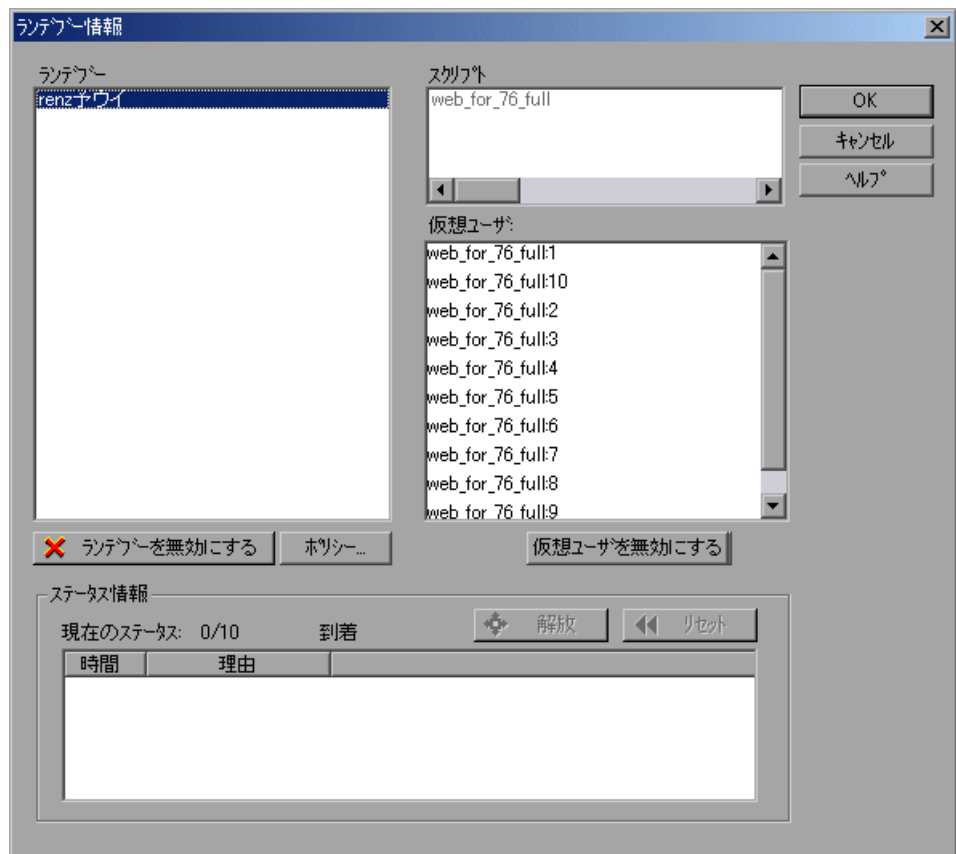
- 5 シナリオを実行します。

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。たとえば、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 13 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

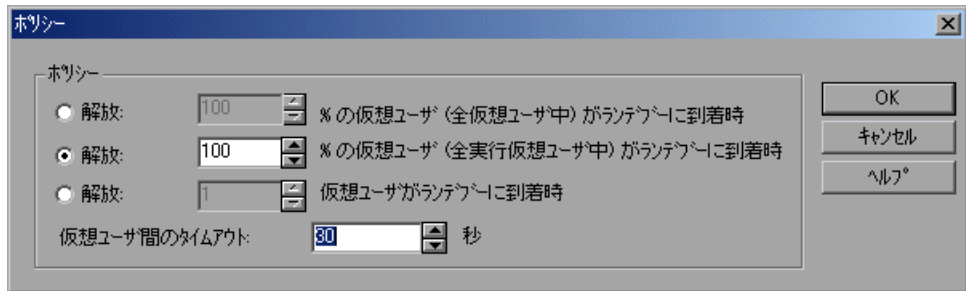
ランデブー・ポリシーを設定して、仮想ユーザでランデブー・ポイントをどのように扱うかを決めます。各ランデブーに、次のポリシー属性を設定します。

解放ポリシー ランデブーから一度に解放される仮想ユーザの数を設定します。

タイムアウト時間 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です。

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。したがって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。

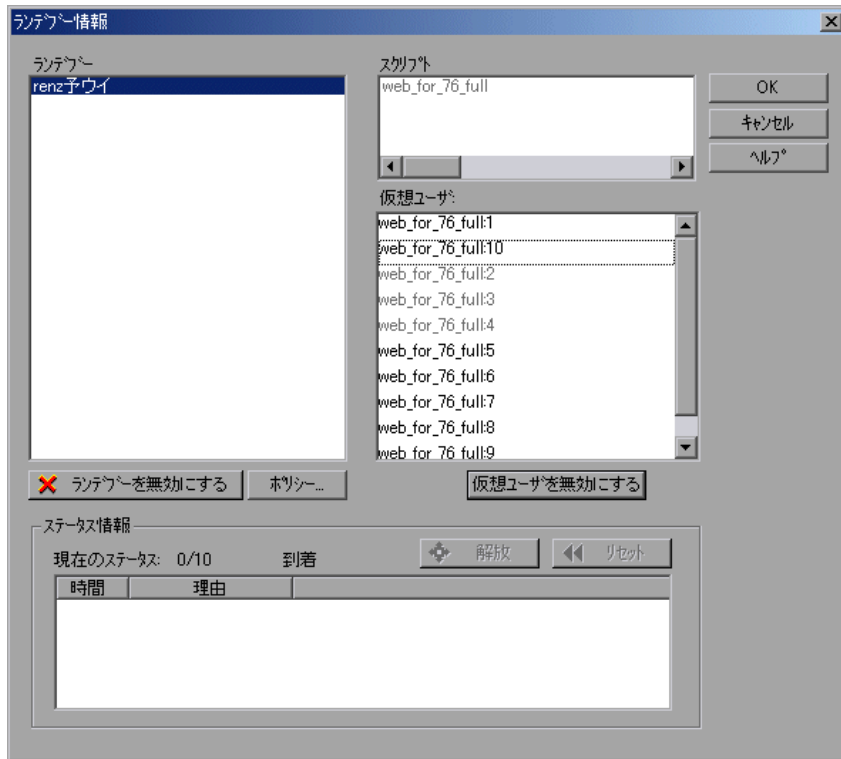
ランデブー・ポイントの仮想ユーザを有効 / 無効にする

は、シナリオのすべての仮想ユーザではなく、LoadRunner 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。

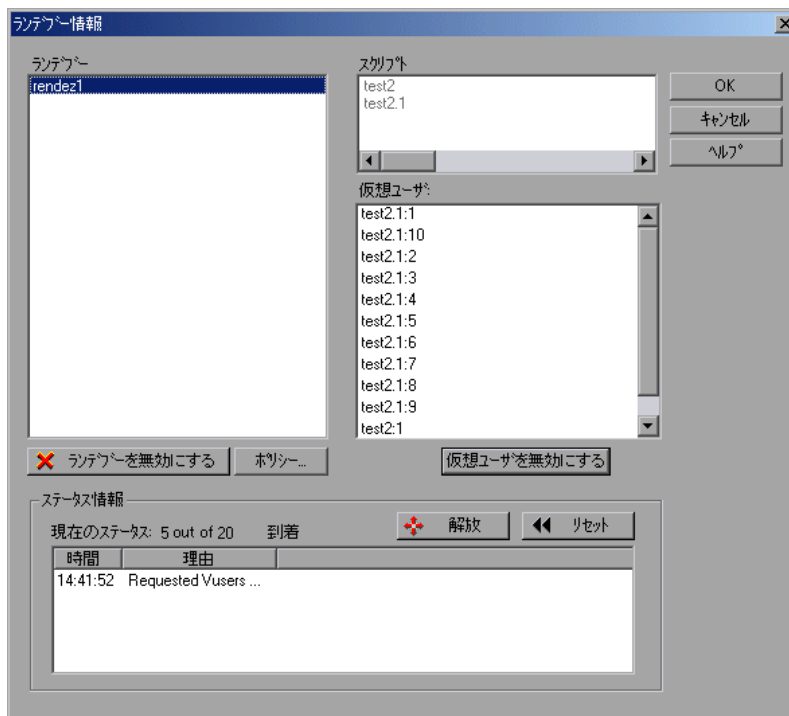
[時間]：ランデブー・ポイントで仮想ユーザが解放された時間。

[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。

[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。

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 / 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。

- ▶ [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。
- ▶ [開放]：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、以下の項目について説明します。

- ▶ シナリオ実行環境の設定
- ▶ タイムアウト時間の設定
- ▶ 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、通常は一度設定するだけで済みます。

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、『LoadRunner 仮想ユーザ・スクリプトの作成』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 16 章「オンライン・モニタ」を参照してください。

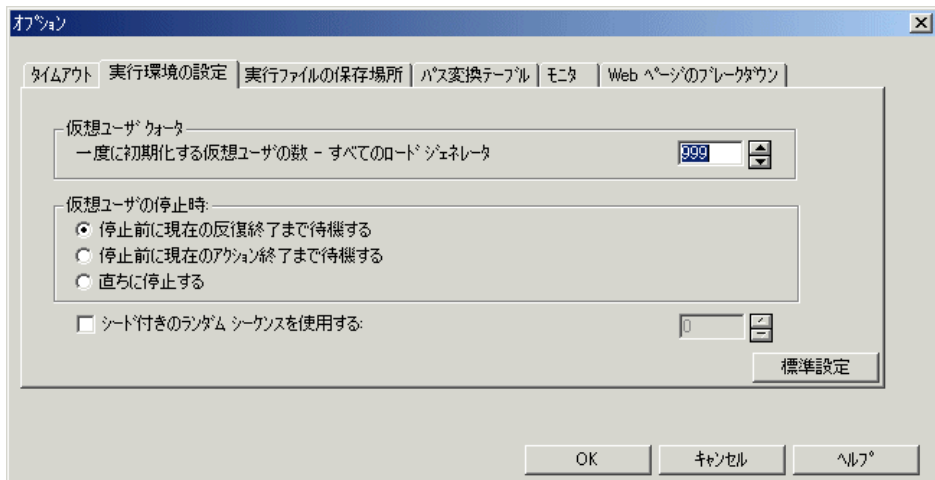
LoadRunner のエキスパート・モードでは、LoadRunner エージェントとその他の LoadRunner コンポーネントに対するさらに多くの設定ができます。詳細については、付録 C「エキスパート・モードでの作業」を参照してください。

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。

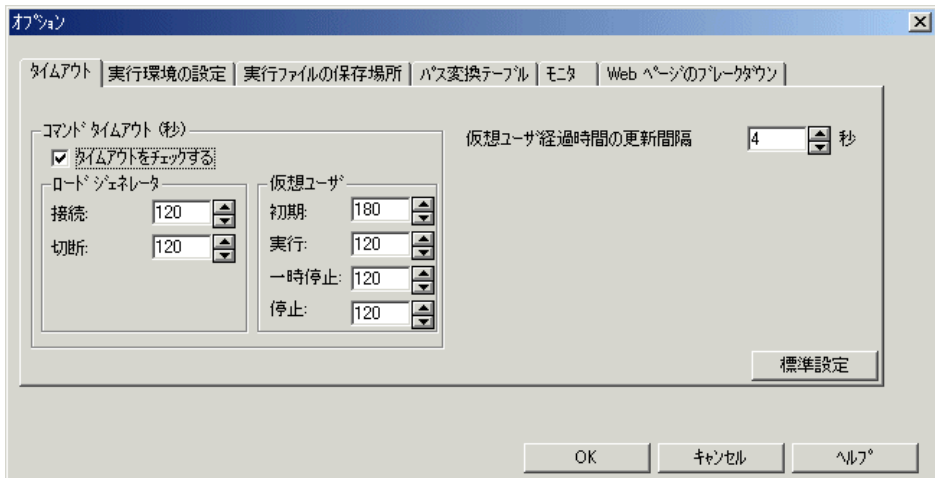
[**シード付きのランダムシーケンスを使用する**]：LoadRunner がランダム・シーケンスにシード値を使用できるようにします。各シード値は、テスト実行に使用されるランダム値のシーケンスを表します。同じシード値を使用している場合はいつも同じ値のシーケンスがシナリオ内の仮想ユーザに割り当てられます。この設定は、データ・ファイルの値を割り当てるためにランダム方式を使って、パラメータ化された仮想ユーザ・スクリプトに適用されます。この設定は、記録される思考遅延時間のランダム・パーセンテージにも影響します (VuGen のヘルプで [実行環境設定] ダイアログ・ボックスに関する情報を参照してください)。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。

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] タブをクリックします。



- 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。たとえば、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 [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。

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、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

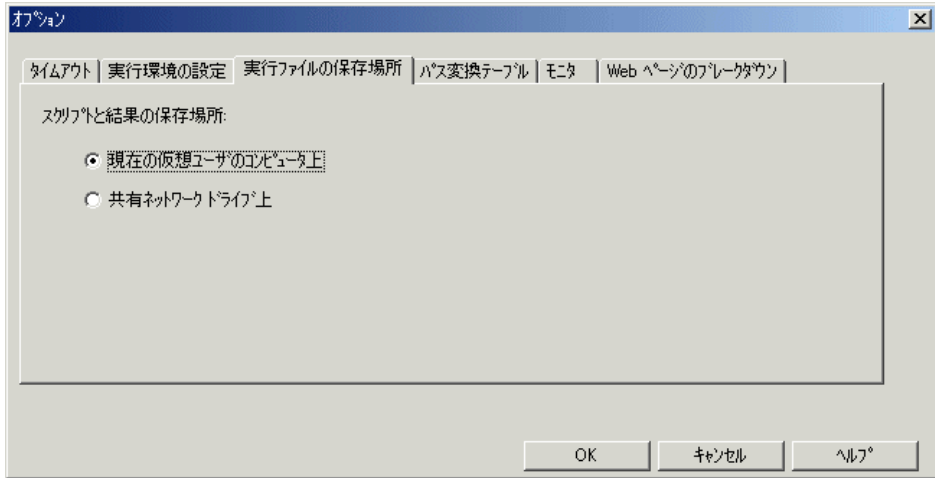
仮想ユーザを実行すると、コントローラは関係する仮想ユーザ・スクリプトのコピーを仮想ユーザのロード・ジェネレータに送信します。このスクリプトは、ロード・ジェネレータの実行ファイルの一時ディレクトリに保存されます。

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13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] タブをクリックします。



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、(175 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、以下の項目について説明します。

- ▶ 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。

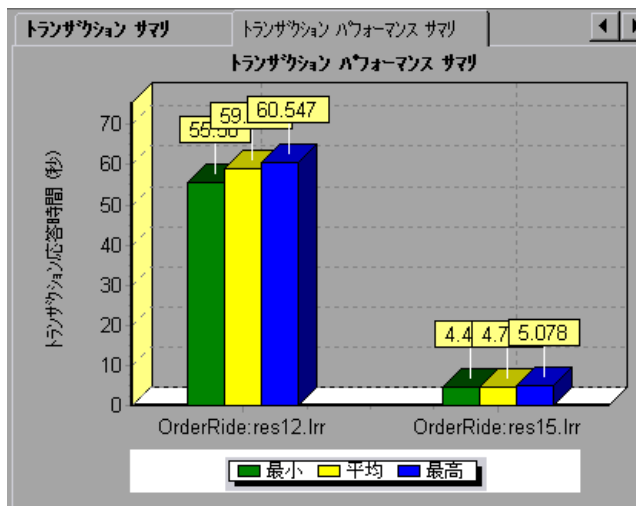
タイムアウト、出力、クォータなどの 1 回限りの構成設定については、第 10 章「シナリオの設定」を参照してください。

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、第 10 章「シナリオの設定」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。

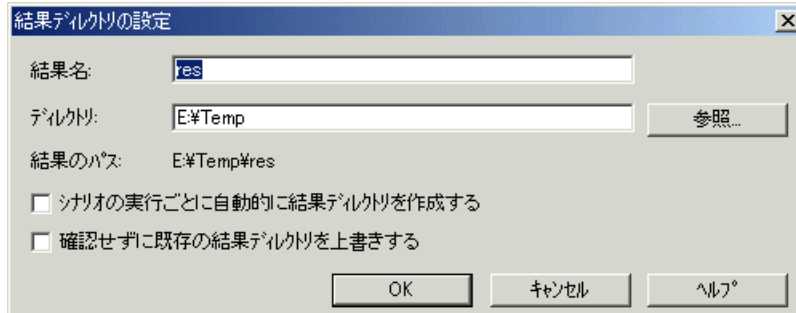


クロス結果グラフの詳細については、『**LoadRunner アナリシス・ユーザーズ・ガイド**』を参照してください。

注：また、マーキュリー・インタラクティブの Web ベース・テスト管理プログラム TestDirector 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「TestDirector 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： TestDirector 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。

[**結果の名前**]：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。















[**TestDirector**] (TestDirector に接続中のみ)：TestDirector テスト・セットに結果を保存できます。

[**ファイル システム**] (TestDirector 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。

 results	_____	結果ディレクトリ
 test2	_____	結果の名前
 _t_rep.eve	_____	イベント・ファイル
 collate.txt	_____	照合ファイル
 localhost_1.eve	_____	ホスト・イベント・ファイル
 offline.dat	_____	オフライン・データ・ファイル
 off_1.def	_____	定義ファイル
 output.mdb	_____	出力データベース
 remote_results.txt	_____	リモート結果ファイル
 test2.lrr	_____	結果ファイル
 travel.cfg	_____	仮想ユーザ cfg ファイル
 travel.usp	_____	仮想ユーザ usp ファイル
 log	_____	仮想ユーザ・ログ・ディレクトリ
 sum_data	_____	サマリ・データ・ディレクトリ

- ▶ 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。
- ▶ **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・エンジンで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、『**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。180 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、169 ページ「シナリオの設定について」を参照してください。

自動照合を無効にし、オプションの横のチェック・マークをクリアするには、**[結果]** > **[結果の自動照合]** を選択します。結果を手作業で照合する場合は、**[結果]** > **[結果の照合]** を選択します。**[ファイルの照合]** 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。結果の照合を停止し、ダイアログ・ボックスを閉じるには、**[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、634 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も、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]** > **[結果を再照合]** 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。

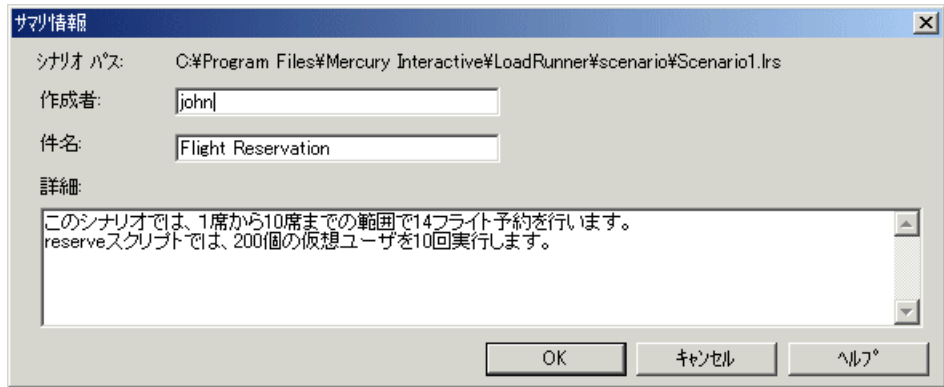
LoadRunner は、分析データを生成する前に、以前に照合が行われていなければ自動的に結果を照合します。

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

TestDirector を使ったシナリオ管理

LoadRunner と TestDirector の統合により、TestDirector を使って LoadRunner のシナリオを管理できます。TestDirector 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、以下の項目について説明します。

- ▶ TestDirector の接続と切断
- ▶ TestDirector プロジェクトからシナリオを開く
- ▶ TestDirector プロジェクトへのシナリオの保存
- ▶ TestDirector プロジェクトへの結果の保存
- ▶ TestDirector プロジェクトからの仮想ユーザ・スクリプトの追加

TestDirector を使ったシナリオ管理

LoadRunner は、マーキュリー・インタラクティブの Web ベースのテスト管理ツール TestDirector と組み合わせて使えます。TestDirector 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 TestDirector 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 TestDirector プロジェクトにアクセスするには、TestDirector 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。

TestDirector を使った作業の詳細については、『**TestDirector ユーザーズ・ガイド**』を参照してください。

TestDirector の接続と切断

LoadRunner と TestDirector の両方を使って作業している場合には、LoadRunner から TestDirector プロジェクトとやり取りできます。テスト工程中、LoadRunner と TestDirector プロジェクトはいつでも接続または切断できます。

TestDirector 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 TestDirector Web サーバまたはリモート TestDirector Web サーバに接続します。このサーバは、LoadRunner と TestDirector プロジェクトの間の接続を処理します。

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。TestDirector 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 TestDirector への接続は、次の手順で行います。

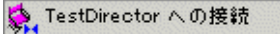
- 1 コントローラで、[ツール] > [TestDirector への接続] を選択します。
[TestDirector への接続] ダイアログ・ボックスが表示されます。



- 2 [サーバ] ボックスに、TestDirector 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 [Connect] 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] セクションの [ドメイン] にドメイン名を入力します。
- 5 [プロジェクト] ボックスで、TestDirector プロジェクトを選択します。
- 6 [ユーザ名] ボックスに、ユーザ名を入力します。
- 7 [パスワード] ボックスに、パスワードを入力します。
- 8 [Connect] をクリックして、選択したプロジェクトに LoadRunner を接続します。
選択したプロジェクトへの接続が確立されると、[プロジェクト] 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。
- 9 起動時に TestDirector 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。
- 10 [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する** 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、LoadRunner を TestDirector に接続するときに、パスワードの入力が必要になります。
- 11 [閉じる] をクリックして、[TestDirector 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 TestDirector 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。

 TestDirector への接続

TestDirector からの LoadRunner の切断

選択した TestDirector プロジェクトと Web サーバから LoadRunner を切断できます。

LoadRunner の TestDirector からの切断は、次の手順で行います。

- 1 コントローラで、[ツール] > [TestDirector への接続] を選択します。
[TestDirector 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 [Disconnect] 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 [Disconnect] をクリックします。
- 4 [閉じる] をクリックして、[TestDirector への接続] ダイアログ・ボックスを閉じます。

[TestDirector への接続] ダイアログ・ボックスについて

[TestDirector への接続] ダイアログ・ボックスを使用して、TestDirector プロジェクトに接続できます。TestDirector は、シナリオ、シナリオ結果、仮想ユーザ・スクリプトの整理と管理を行いやすいよう、プロジェクト・リポジトリを使用します。

[サーバへの接続]：TestDirector プロジェクトを使用する前に、そのプロジェクトが属するサーバに接続する必要があります。

- ▶ [サーバ]：TestDirector プロジェクトの属するサーバの名前を入力します。
- ▶ [Connect]：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 TestDirector データベース・サーバに接続したら、プロジェクトを選択し、そのプロジェクトのユーザ名、ドメイン名、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。
- ▶ [プロジェクト]：接続するプロジェクトを選択します。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]：ユーザの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 TestDirector 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。

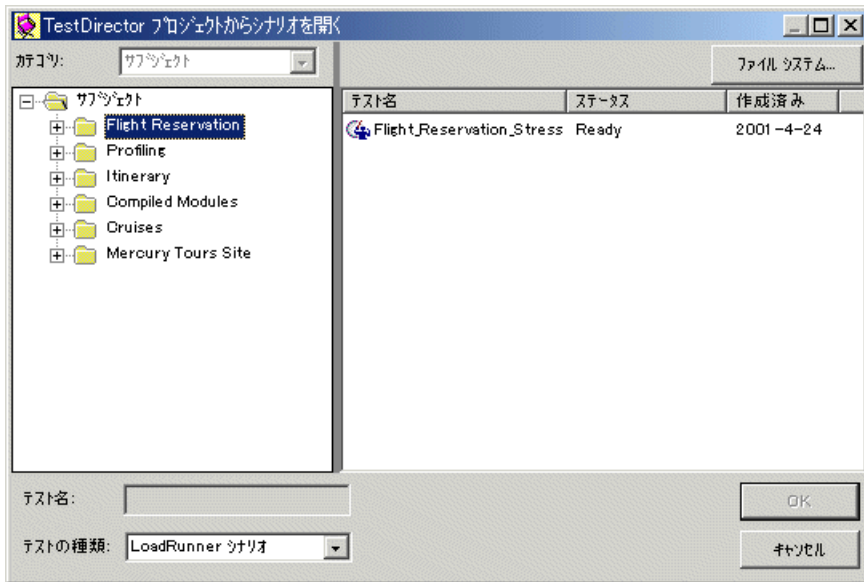
[起動時に再接続できるようにパスワードを保存する]：これが選択されている場合は、ログイン・プロセスを自動化するために、指定したパスワードがレジストリに保存されます。

TestDirector プロジェクトからシナリオを開く

LoadRunner を TestDirector プロジェクトに接続する場合、TestDirector 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。

TestDirector プロジェクトからシナリオを開くには、次の手順で行います。

- 1 TestDirector サーバに接続します（188 ページ「TestDirector 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、[既存のシナリオを開く] アイコンをクリックします。[TestDirector 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、[ファイル システム] 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（[シナリオを開く] ダイアログ・ボックスから [TestDirector 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、[TestDirector] ボタンをクリックします）。

- 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 [テスト名] リストに表示されます。
- 4 [テスト名] リストからシナリオを選択します。読み取り専用の [テスト名] 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。TestDirector 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、[TestDirector 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。

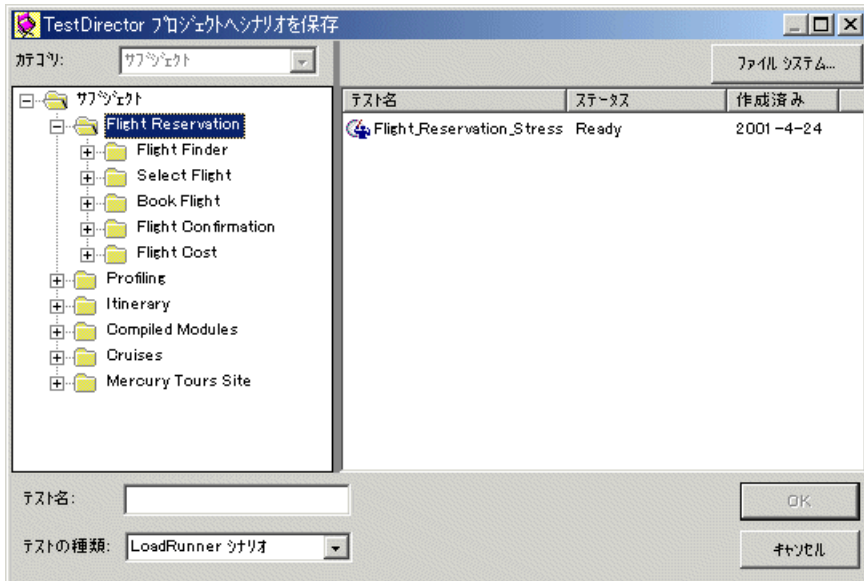
TestDirector プロジェクトへのシナリオの保存

LoadRunner が TestDirector 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せん。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の直ちに表示したりできます。

シナリオの TestDirector プロジェクトへの保存は、次の手順で行います。

- 1 TestDirector サーバに接続します (188 ページ「TestDirector 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。
[TestDirector 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、[ファイル システム] ボタンをクリックします。[シナリオを保存] ダイアログ・ボックスが表示されます。([シナリオを保存] ダイアログ・ボックスから [TestDirector プロジェクトへシナリオを保存] ダイアログ・ボックスに戻るには、[TestDirector] ボタンをクリックします)。

- 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 [テスト名] 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 TestDirector を起動したときに、TestDirector のテスト計画ツリーに新しいシナリオが表示されます。

TestDirector プロジェクトへの結果の保存

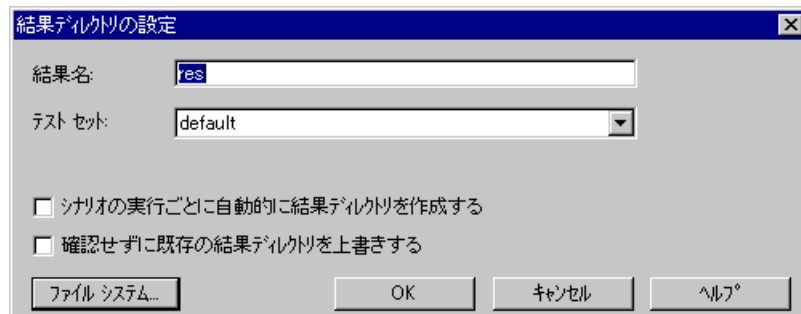
シナリオを実行する前に、結果の格納場所を設定します。LoadRunner が TestDirector プロジェクトに接続されている場合、実行結果はテスト・セットに保存されます。標準ファイル・システムを使って、結果をディスクに保存することもできます。

実行結果の TestDirector プロジェクトへの保存は、次の手順で行います。

- 1 TestDirector サーバに接続します（188 ページ「TestDirector 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 [TestDirector] 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、ユーザー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。

TestDirector プロジェクトからの仮想ユーザ・スクリプトの追加

TestDirector 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に追加できます。

マニュアル・シナリオへの仮想ユーザ・スクリプトの追加

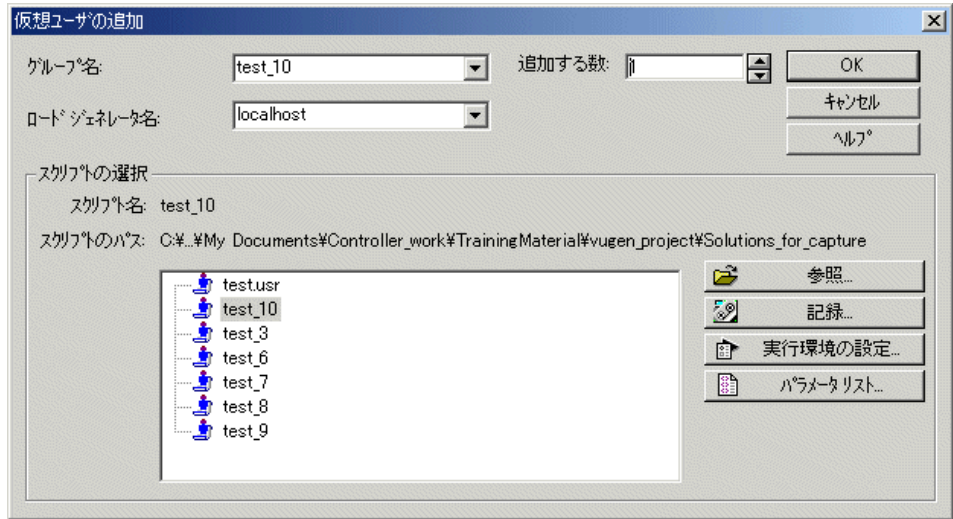
マニュアル・シナリオを作成している場合、[グループの追加] ダイアログ・ボックスを使って仮想ユーザ・スクリプトを追加できます。

マニュアル・シナリオへの仮想ユーザ・スクリプトの追加は、次の手順で行います。

- 1 TestDirector サーバに接続します (188 ページ「TestDirector 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。
[仮想ユーザの追加] 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] ボタンをクリックします。[TestDirector プロジェクトからテストを開く] 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。
- 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_td

- 5 [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。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。

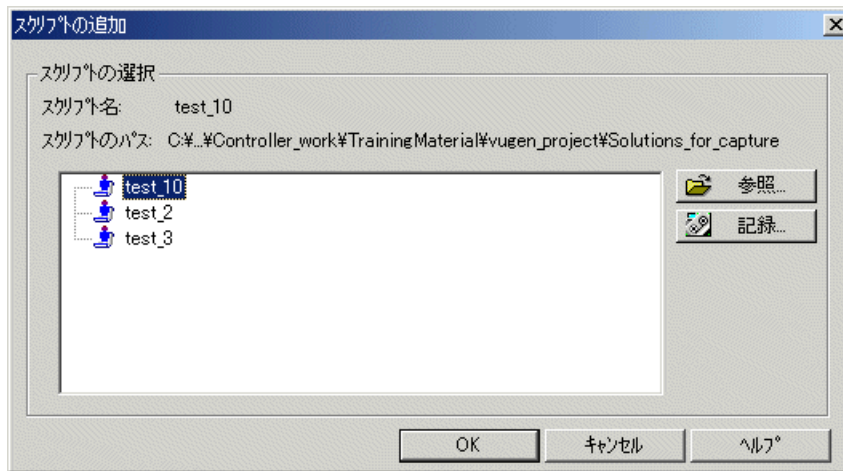
ゴール指向シナリオへの仮想ユーザ・スクリプトの追加

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への仮想ユーザ・スクリプトの追加は、次の手順で行います。

- 1 TestDirector サーバに接続します（188 ページ「TestDirector 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。

スクリプトの追加(A)



- 3 [参照] ボタンをクリックします。[TestDirector プロジェクトからテストを開く] 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。
- 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_td

- 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 3 部

シナリオの実行

第 13 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、以下の項目について説明します。

- ▶ 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 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、170 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、214 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。204 ページ「仮想ユーザ・グループの制御」と211 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。

シナリオ全体の実行は、次の手順で行います。

- 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] タブをクリックします。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] ボタンをクリックします。コントローラは、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。

▶ シナリオの開始

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。

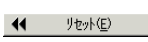
シナリオの実行中に、個々の仮想ユーザおよび仮想ユーザ・グループを操作できます。これについては、204 ページ「仮想ユーザ・グループの制御」と211 ページ「個々の仮想ユーザの制御」で説明します。

- 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。



- 5 [シナリオ] > [停止] を選択するか、[停止] ボタン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] ボタン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」）にリセットされます。

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、仮想ユーザの初期化、実行、停止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できます。

仮想ユーザ・グループの初期化は、次の手順で行います。

- 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] ボタンをクリックするか、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・グループの実行は、次の手順で行います。

- 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**」ボタンをクリックするか、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、230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。

仮想ユーザの一時停止は、次の手順で行います。

- 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**」ボタンをクリックするか、仮想ユーザ・グループ (1つまたは複数) 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] ステータスではない場合、「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) に戻ります。

仮想ユーザ・グループのリセットは、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 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】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。

- ▶ 【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。
- ▶ 【直ちに停止する】：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。
- ▶ 【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。
- ▶ 【仮想ユーザの実行/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。
- ▶ **[仮想ユーザの初期化]** : 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。[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。アイコンをクリックすると、個々のトランザクションとエラーの詳細が表示されます。

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。

[グラフ]： 利用可能なグラフのリストを表示するには、[表示] > [利用可能なグラフを表示] 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] > [利用可能なグラフを非表示] を選択するか、[利用可能なグラフ] リストの右上隅にある [X] 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、グラフに対して以下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、測定値を対象に以下の操作を実行できます。

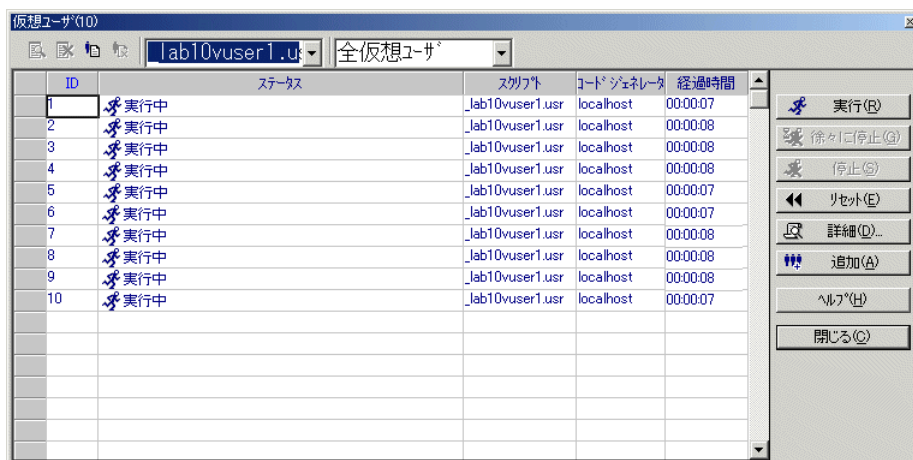
- ▶ 測定値の追加 / 削除
- ▶ 測定値の表示 / 非表示
- ▶ 測定値の設定
- ▶ 測定値の説明

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。

個々の仮想ユーザの制御は、次の手順で行います。

- 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。



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。

注：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。

- ▶ 仮想ユーザを選択して [リセット] をクリックすると、ステータスが [ダウン] に戻ります。
- ▶ 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。
- ▶ 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。



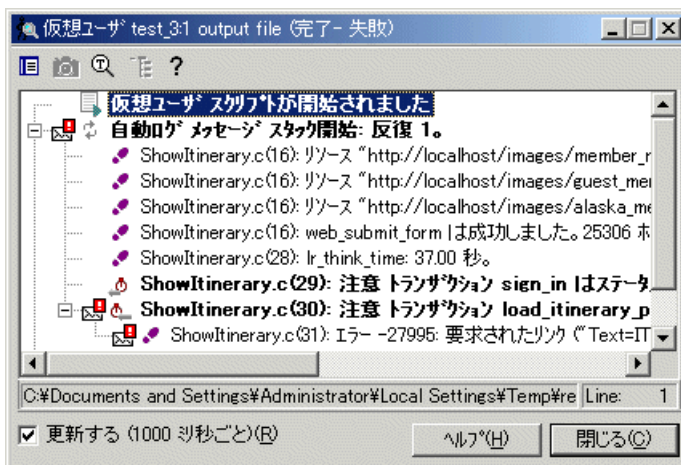
[オプション] メニューの項目を使用すると、表示するコントロールの種類を選択できます。

[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、**[仮想ユーザ ログを非表示]** ボタンをクリックします。仮想ユーザ・スクリプト・ログの詳細については、230 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。

ランデブーからの手作業による仮想ユーザの解放は、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]** 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。



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、157 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。

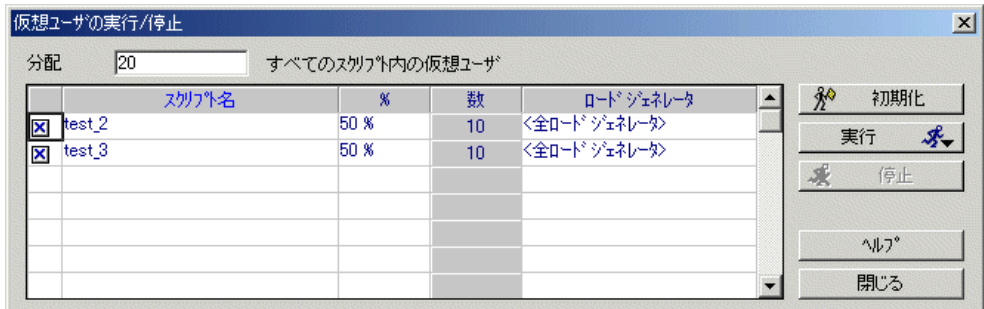
実行中のシナリオへの仮想ユーザの追加は、次の手順で行います。

- 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] ボタンをクリックします。[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(仮)...



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が 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。

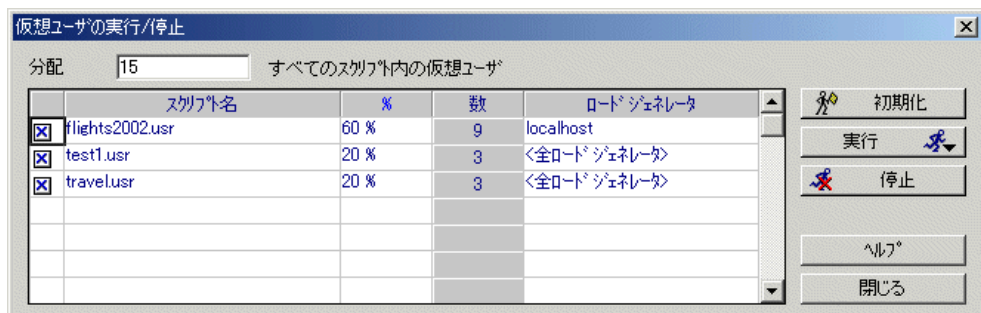
パーセントモードの場合、複数のロード・ジェネレータを選択して、仮想ユーザ・スクリプトを実行できます。[ロードジェネレータ名] リストからロード・ジェネレータを選択し、[OK] をクリックします。このリストのすべてのロード・ジェネレータを使用するには、[全ロードジェネレータ] ボタンをクリックします。

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] ボタンをクリックして、追加した仮想ユーザの数を初期化します。
- 6 [実行] ボタンをクリックし、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。

この割合にしたがい、[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]、[中止]）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、136 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 :

- ▶ [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。
- ▶ [%] : 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)。
- ▶ [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。
- ▶ [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]：次のオプションのいずれかを選択します。

- ▶ [初期化して実行]：シナリオ内のすでに初期化されている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。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]：指定し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、170 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 14 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、以下の項目について説明します。

- ▶ 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 <code>init</code> 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た。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)。
終了中	仮想ユーザは実行を終えたか、停止されて終了中。
停止	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
実行中の仮想ユーザ	4	
経過時間	00:00:52 (時間:分:秒)	
秒ごとのヒット数	42.84 (最終 60 秒)	
成功したトランザクション	86	🔍
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。

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。

[停止] : 中止されたトランザクションの数が表示されます。

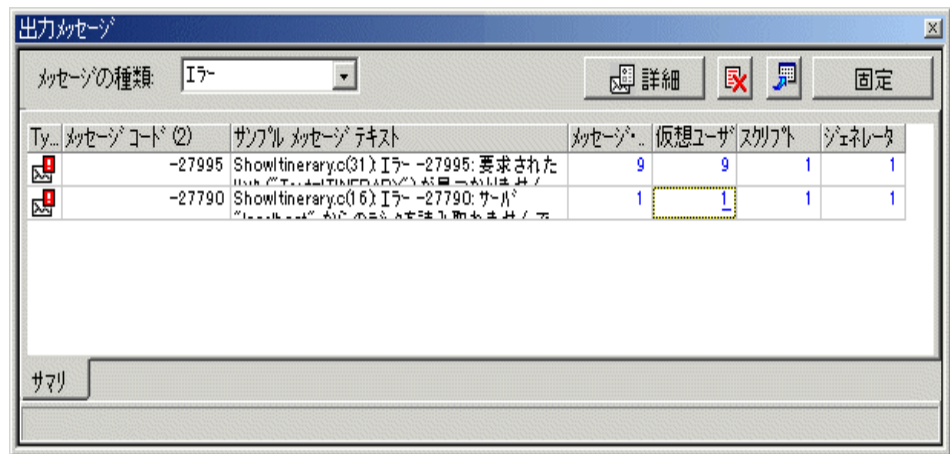
[出力メッセージ] ウィンドウの表示

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。

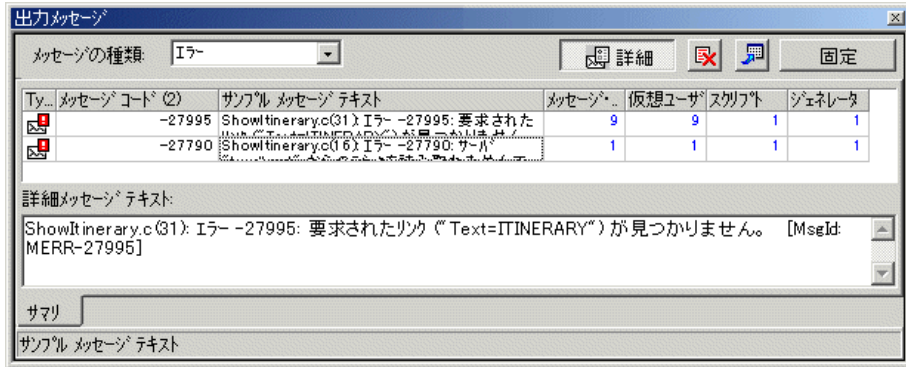
[出力メッセージ] ウィンドウにメッセージを表示するには、次の手順を行います。

- 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、エラーのログ情報のリストが表示されます。



- 2 [メッセージの種類] ボックスで、フィルタにか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。

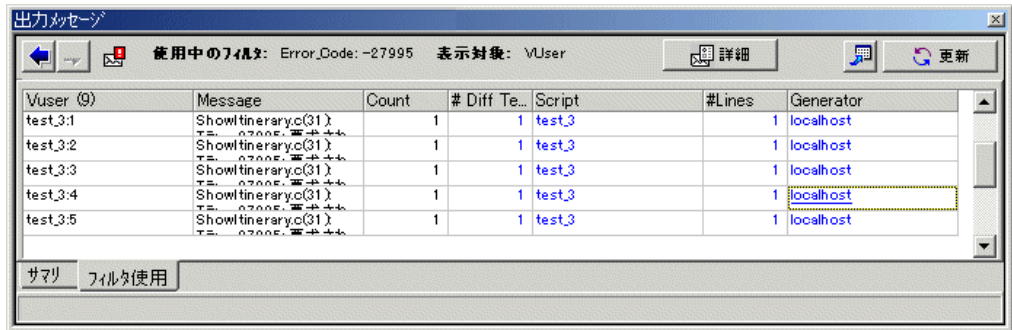


- 4 メッセージ、仮想ユーザ、スクリプト、またはロード・ジェネレータごとのログ情報の詳細を表示するには、それぞれのカラムの青いリンクをクリックします。詳細については、226 ページ「ログ情報の詳細表示」を参照してください。

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 [詳細] タブには、ドリル・ダウンされたビューがメッセージ、仮想ユーザ・スクリプト、またはロード・ジェネレータごとに表示されます。

たとえば、[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 [出力メッセージ] ウィンドウに表示されます。



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 [アクション] か [行番号]) 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=00 (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 [詳細] 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 [前を表示] ボタンと [次を表示] ボタンをクリックします。

[出カメッセージ] 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。

[タイプ]：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。

▶ [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]：たとえば `lr_output_message` 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、付録 C「オプション - デバッグ情報の設定」を参照してください。



▶ [バッチ]：バッチ・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。

[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。



[ビューのエクスポート]：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。

注 : 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。

【詳細】 タブ

【詳細】 タブには、ドリル・ダウンされたビューが、メッセージ、仮想ユーザ、スクリプト、またはロード・ジェネレータごとに表示されます。たとえば、【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【詳細】 タブに表示されます。



【前を表示 / 次を表示】：さまざまなドリル・ダウン階層間を移動できます。



【メッセージのタイプ 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。

【アクティブ 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。



【ビューのエクスポート】：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【詳細】 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

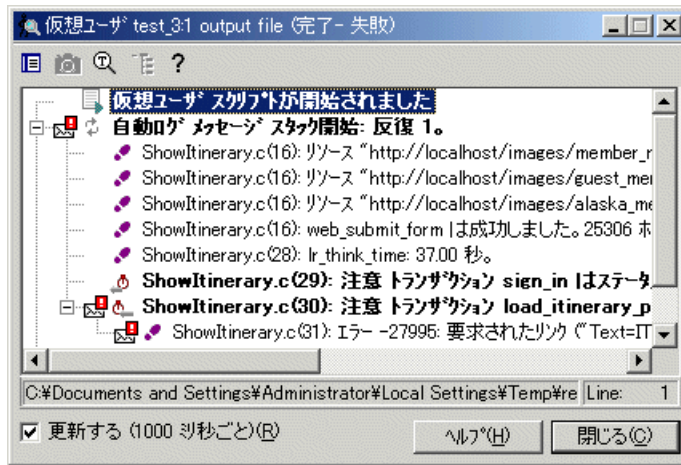
シナリオの実行中、実行中の各仮想ユーザに関する実行時の情報が格納されているログを表示できます。

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。標準設定では 1000 ミリ秒ごとに更新されます。



標準の更新間隔設定の変更については、638 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 [エラー発生時にスナップショットを起動する] 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・ビューに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] チェック・ボックスをクリアします。

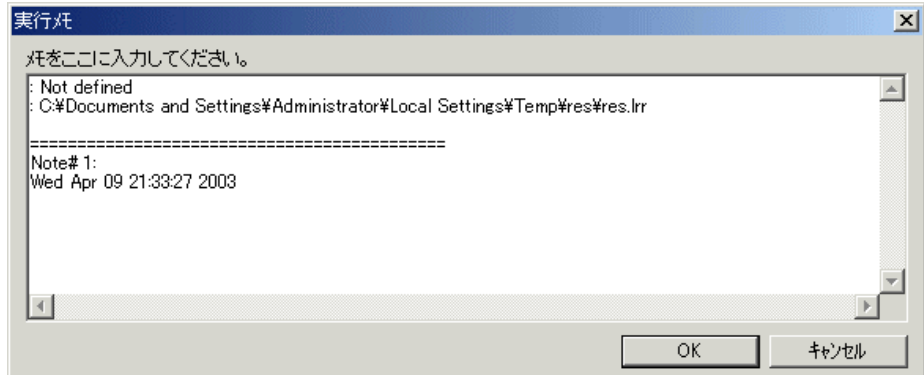
注：標準の更新間隔設定は、638 ページ「オプション - 出力の設定」で変更できます。

コピー：仮想ユーザ・ログからテキストをコピーできます。仮想ログ内で選択したテキストを右クリックし、[コピー] をクリックします。

パスをステータス・バーからコピー：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、[パスをステータス・バーからコピー] をクリックします。

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。



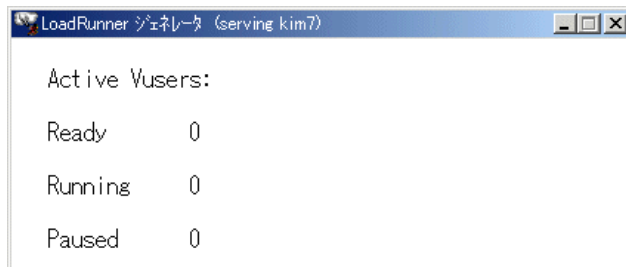
実行メモの記録は、次の手順で行います。

- 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 [OK] 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 [Ready], [Running], [Paused] 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最前面に表示されます。このウィンドウは、いつでも最小化または復元できます。



The screenshot shows a window titled "LoadRunner ジェネレータ (serving kim7)". Inside the window, the text "Active Users:" is followed by a list of agent states and their counts:

Ready	0
Running	0
Paused	0

第 15 章

ファイアウォールに関する作業

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、ファイアウォールの背後にある仮想ユーザの実行やサーバの監視が可能です。

本章では、以下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および監視の概要
- ▶ ファイアウォールの内側への 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監視
- ▶ 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
- ▶ サーバ・モニタのプロパティの設定
- ▶ 測定値の追加と削除
- ▶ 測定頻度の設定
- ▶ 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
- ▶ トラブルシューティングのヒント

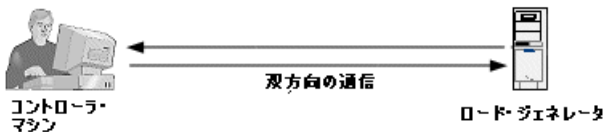
LoadRunner でのファイアウォールの使用について

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。

たとえば、メール・ポート（23）を除いたネットワーク内のすべてのポートに外部からアクセスできないようにしたり、メール・ポートと Web ポート（80）を除いたネットワーク内のすべてのポートから外部への接続を防ぐことができます。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。

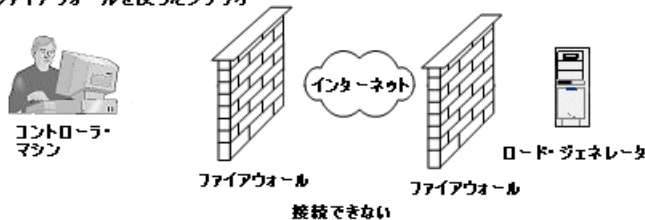
通常の LoadRunner シナリオ（ファイアウォール越しの処理を行わないシナリオ）では、コントローラは、リモート・マシンで実行中の LoadRunner エージェントに直接アクセスできます。これにより、コントローラはリモート・マシンに直接接続できます。

通常のLoadRunnerシナリオ



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。

ファイアウォールを使ったシナリオ

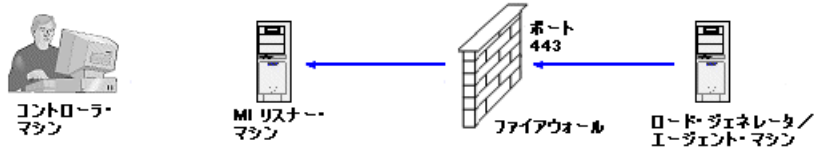


LoadRunner は、ファイアウォール上の標準 SSL ポート（ポート 443）を使用する HTTPS またはセキュア TCP/IP に基づいた通信メカニズムを使用することで、この問題を解決します。詳細については、238 ページ「システムの設定」を参照してください。

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（「メディアエータ」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



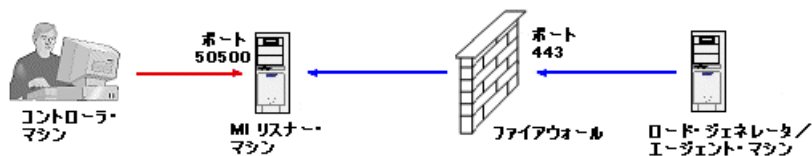
LoadRunner エージェントが MI リスナーへの接続を確立すると、MI リスナーはエージェントから渡されたシンボル名を使って、エージェントへの接続のリストを維持します。コントローラは、MI リスナーに接続するとき、ポート 50500 を経由して MI リスナーと通信します。

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。これで、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

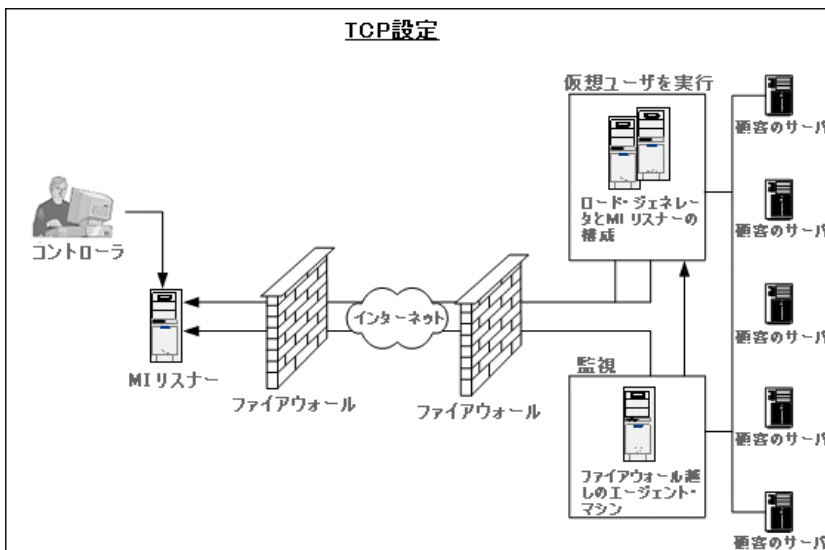


システムの設定

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。

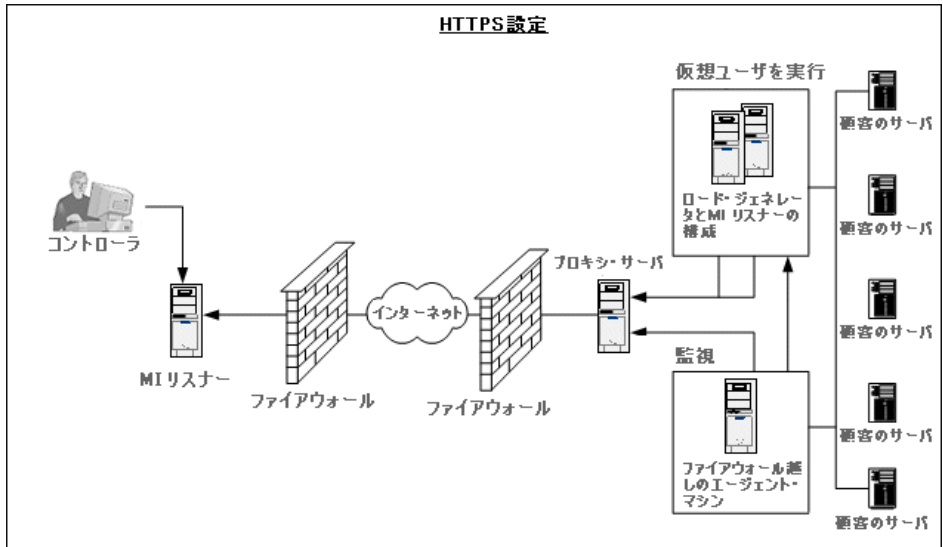
TCP 設定

TCP 設定では、ファイアウォール 1 の内側にあるすべての LoadRunner 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。



ファイアウォール越しの仮想ユーザの実行および監視の概要

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、接続を行います。手順 4～8 が必要なのは、ファイアウォールの内側にあるサーバを監視する場合だけです。

- 1 **LoadRunner エージェントは、仮想ユーザを実行するマシン、またはファイアウォールの内側にある監視対象のサーバにインストールします。**

エージェントが、仮想ユーザを実行するマシン、またはファイアウォールの内側にある監視対象のサーバにインストールされていることを確認します。LoadRunner エージェントは Windows マシンでも UNIX マシンでも実行できます。241 ページ「ファイアウォールの内側への 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。**

仮想ユーザを実行するマシンの LoadRunner エージェント、または、監視対象サーバのメディアータとして機能するエージェントを設定します。手順については、242 ページ「ファイアウォールの内側の 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ます。**

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。251 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 **「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。**

ファイアウォール越しにサーバを監視する場合は、ファイアウォールの内側にあるエージェント・マシンにこのコンポーネントをインストールします。このマシンは、コントローラと監視対象サーバとの間でメディアータとして機能します。「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する場所の詳細については、236 ページ「LoadRunner でのファイアウォールの使用について」の項の図を参照してください。設定の手順については、256 ページ「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」を参照してください。

5 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。

MI リスナーのインストールの詳細については、『**LoadRunner インストール・ガイド**』を参照してください。MI リスナーをインストールする場所の詳細については、238 ページ「システムの設定」の項の図を参照してください。

6 MI リスナー・マシンを設定します。

各 MI リスナー・マシンのセキュリティ属性を設定します。252 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。

7 コントローラ・マシンを設定します。

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。254 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。

8 サーバ・モニタのプロパティを設定します。

サーバ・モニタのプロパティと測定の頻度を設定します。257 ページ「サーバ・モニタのプロパティの設定」、262 ページ「測定値の追加と削除」、および 263 ページ「測定頻度の設定」を参照してください。

ファイアウォールの内側への 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または監視するには、仮想ユーザを実行するロード・ジェネレータ・マシンか、ファイアウォールの内側にある監視対象サーバに LoadRunner エージェントをインストールします。エージェントは、Windows サービスとして、または「スタートアップ」フォルダから実行される実行可能ファイルとして追加されます。

ファイアウォール越しの仮想ユーザの実行

セットアップからロード・ジェネレータのインストールを実行した場合は、ファイアウォールの内側にあるロード・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。LoadRunner エージェントがインストール済みかどうかを確認するには、**[スタート]** >

[プログラム] > [LoadRunner] > [LoadRunner Agent Service/Process] の順にクリックします。LoadRunner のメニューの選択肢のリストに [Agent Service] または [Agent Process] が表示された場合は、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 コントローラの 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、238 ページ「システムの設定」の項の図を参照してください。

ファイアウォール越しの監視

「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にある監視対象のサーバにインストールします。詳細については、256 ページ「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」を参照してください。「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する場所の詳細については、238 ページ「システムの設定」の項の図を参照してください。

ファイアウォールの内側の 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

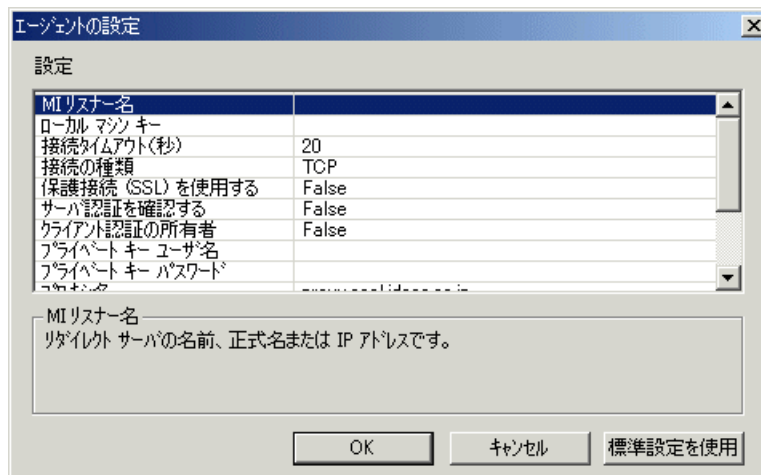
LoadRunner エージェントを Windows マシンで設定するには、次の手順で行います。

- 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] > [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 [Enable Firewall Agent] チェック・ボックスを選択し、[Settings] をクリックします。



[エージェントの設定] ダイアログ・ボックスが開きます。



- 4 各オプションを 249 ページ「エージェントの設定」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、標準設定の値をすべての設定に割り当てる場合は [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] > [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 での 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。249 ページ「エージェントの設定」の説明に従って、各オプションを設定します。

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 =
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。

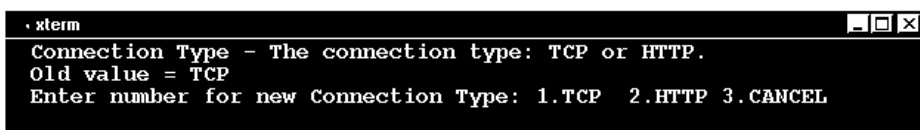
```
MI Listener Name - The name, full name or IP address of the redirection server.
Old value =
Enter new MI Listener Name.
bunji
Change MI Listener Name from "" to "bunji"? 1.OK 2.CANCEL 3.FIX
```

- 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。

新しい値を破棄して設定を再び変更するには、「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。

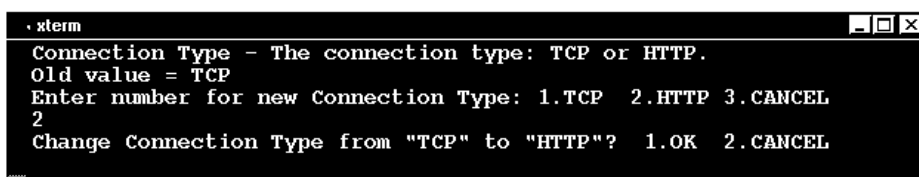
- 1 設定メニューで「4」を入力し、次の画面を表示します。



```
. 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。



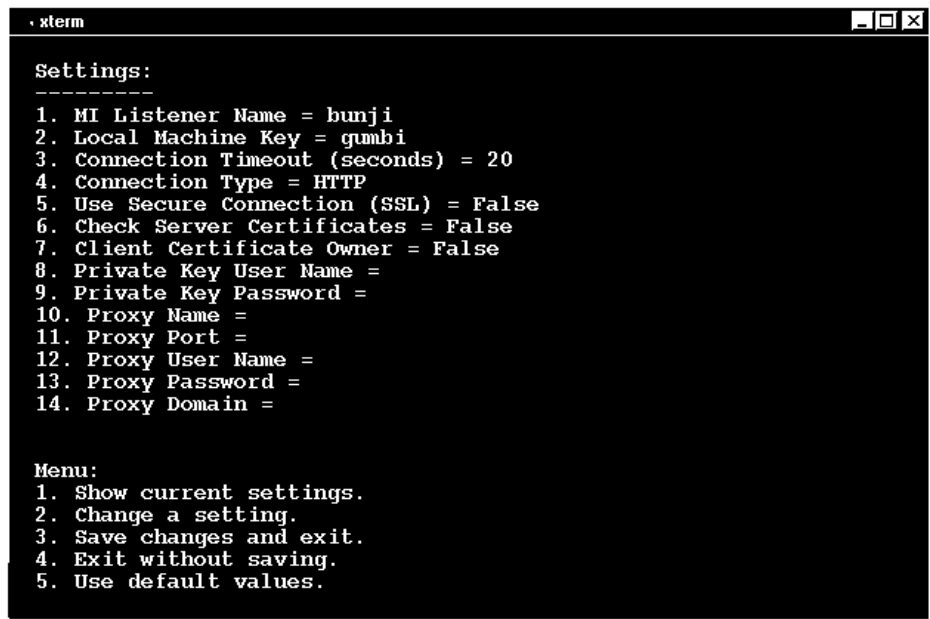
```
. xterm
Connection Type - The connection type: TCP or HTTP.
Old value = TCP
Enter number for new Connection Type: 1.TCP 2.HTTP 3.CANCEL
2
Change Connection Type from "TCP" to "HTTP"? 1.OK 2.CANCEL
```

- 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を使用するには（249 ページ「エージェントの設定」で説明されているように）、「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、付録 D 「コントローラのトラブルシューティング」の「UNIX シェル」を参照してください。

エージェントの設定

オプション	標準設定値	説明
MI リスナー名 (MI Listener name)	なし	マーキュリー・インタラクティブ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 (Local Machine Key)	なし	MI リスナー・マシンを経由して、ファイアウォールの背後にあるコントローラ・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。
接続タイムアウト (秒) (Connection Timeout (seconds))	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。
接続の種類 (Connection Type)	TCP	使用している構成に応じて、TCP または HTTP のどちらかを選択します。
サーバ・ユーザ名 (Server User Name)	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。
サーバ・パスワード (Server Password)	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 (Server Domain)	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にだけ必須です。
保護接続 (SSL) を使用する (Use Secure Connection (SSL))	False	Secure Sockets Layer プロトコルを使用して接続する場合は「True」を選択します。

オプション	標準設定値	説明
サーバ認証を確認する (Check Server Certificates)	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を確認する場合は、「 Medium 」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、「 High 」を選択します。この設定を使用できるのは、「 Use Secure Connection 」オプションが「 True 」に設定されている場合だけです。
クライアント認証の所有者 (Client Certificate Owner)	False	SSL 証明書をロードする場合は「 True 」を選択します。接続の許可を得るために、サーバによっては証明書が要求される場合があります。このオプションが意味を持つのは、「 Use Secure Connection 」オプションが「 True 」に設定されている場合だけです。
プライベートキー パスワード (Private Key Password)	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、「 Client Certificate Owner 」オプションが「 True 」に設定されている場合だけです。
プロキシ名 (Proxy Name)	< IE プロキシ・サーバ名 >または、なし	プロキシ・サーバの名前。このオプションは、「 Connection Type 」オプションが「 HTTP 」に設定されている場合は必須です。
プロキシポート (Proxy Port)	< IE プロキシ・サーバ・ポート >または、なし	プロキシ・サーバの接続ポート。このオプションは、「 Connection Type 」オプションが「 HTTP 」に設定されている場合は必須です。
プロキシ・ユーザ名 (Proxy User Name)	なし	プロキシ・サーバへの接続権限のあるユーザのユーザ名。

オプション	標準設定値	説明
プロキシ・パスワード (Proxy Password)	なし	ユーザのパスワード。
プロキシ・ドメイン (Proxy Domain)	なし	プロキシ・サーバの設定で定義されている場合は、ユーザのドメイン。このオプションは、NTLM が使用されている場合にだけ必須です。

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

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。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」(Connection Timeout) 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」(Proxy Port) 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。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」(Connection Timeout) 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。

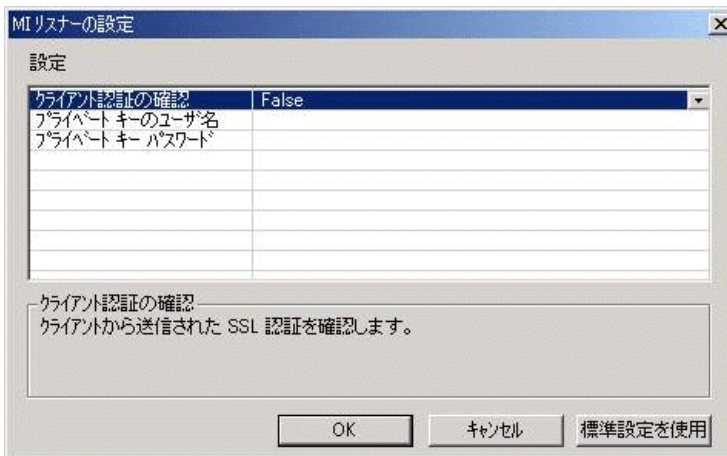
ファイアウォールの外側の MI リスナーのインストールと設定

ファイアウォール越しに仮想ユーザの実行や監視を行うには、コントローラ・マシンと同じ LAN 内の、ファイアウォールの外側にある 1 台または複数のマシンに 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、『**LoadRunner インストール・ガイド**』を参照してください。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。

注：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] を選択し、MI リスナー上の LoadRunner エージェントを停止します。
- 3 [スタート] > [プログラム] > [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 253 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] > [LoadRunner] から実行して、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妨げます。

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確認 (Check Client Certificate)	False	クライアントに対して接続時に SSL 証明書を送信するよう要求し、証明書を認証する場合は、「True」を選択します。
秘密鍵ユーザ名 (Private Key User Name)	なし	SSL 証明書認証処理に必要な場合があるユーザ名。
秘密鍵パスワード (Private Key Password)	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・マシンの間の一意的接続を確立する必要があります。この接続は、マーキュリー・インタラクティブのリスナー・マシンである MI リスナーを通じて確立します。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。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] > [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。
- 2 [ジェネレータ] をクリックして、[ロードジェネレータ] ウィンドウを表示します。[名前] フィールドに、サーバのシンボル名を入力します。この名前は、[エージェントの設定] ダイアログ・ボックスの「ローカルマシンキー」設定に入力した名前と同じです。次の例では、サーバ名は「gumbi」です。

サーバが UNIX サーバの場合、[プラットフォーム] フィールドの値を **UNIX** に変更します。



- 3 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。

- 4 [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。この例では、MI リスナーは「bunji」です。
- 5 [ファイアウォールの設定] セクションで、次のオプションのどちらかを選択します。
- ▶ [ファイアウォール越して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 エミュレーションを使用する場合は、MI リスナー・マシンの IP アドレスを WAN エミュレーションの [IP の除外] リストに追加する必要があります。詳細については、99 ページ「WAN エミュレーションからの IP アドレスの除外」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ー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。

ファイアウォール越しの監視

ファイアウォールの外側からサーバを監視するには、ファイアウォールの内側の指定のマシンに「**ファイアウォール越しの監視**」機能を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に、サーバ・モニタ中継マシン（「メディアエータ」と呼びます）が設定されます。その後、監視対象サーバを指定し、LoadRunner メディアエータ・マシン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コンポーネントのインストール

「**ファイアウォール越しの監視**」は、LoadRunner のインストール時にすでにインストールされている可能性があります。このコンポーネントがインストール済みかどうかを確認するには、[スタート] > [プログラム] > [LoadRunner] > [Advanced Settings] の順にクリックします。LoadRunner 選択肢の一覧に [Monitor Configuration] 項目が表示されていたら、「**ファイアウォール越しの監視**」はすでにインストールされています。252 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」に進んでください。

まだインストールされていない場合は、次の x いずれかの方法で、「ファイアウォール越しの監視」をメディアエータ・マシン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。

- ▶ マーキュリー・インタラクティブのカスタマー・サポート Web サイト (<http://support.mercuryinteractive.com>) から「ファイアウォール越しの監視」のファイル入手します。「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。

LoadRunner のカスタム・インストールの方法については、『**LoadRunner インストール・ガイド**』を参照してください。

注：「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後、257 ページ「サーバ・モニタのプロパティの設定」に進む前に、240 ページ「ファイアウォール越しの仮想ユーザの実行および監視の概要」で説明されている手順をすべて完了してください。

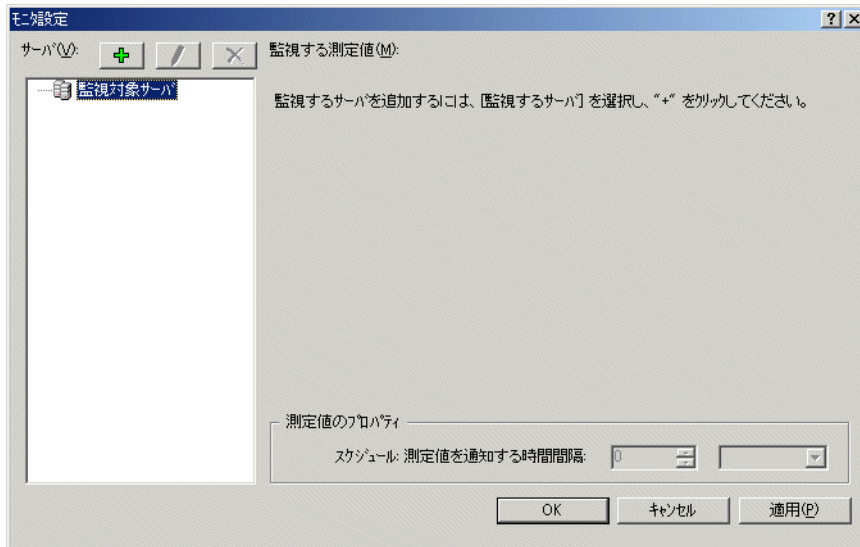
サーバ・モニタのプロパティの設定

LoadRunner エージェント、「ファイアウォール越しの監視」コンポーネント、MI リスナー、およびコントローラ・マシンをインストールして設定した後、メディアータ・マシンによって監視するサーバ測定値を選択する必要があります。

サーバ・モニタのプロパティは、[モニタ設定] ダイアログ・ボックスを使って、メディアータ・マシンから設定します。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。

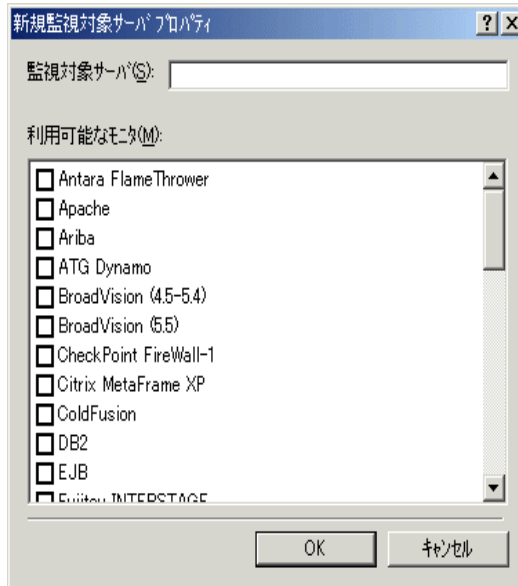
サーバ・モニタのプロパティを設定するには、次の手順で行います。

- 1 [スタート] > [プログラム] > [LoadRunner] > [Advanced Settings] > [Monitor Configuration] を選択します。LoadRunner の完全インストールを行っ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。

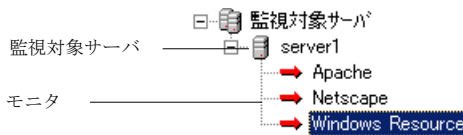
注：同時に複数のサーバを追加するには、サーバ名または IP アドレスをカンマで区切ります。次に例を示します。

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・キーで使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、LoadRunner コントローラで [ヘルプ] > [LoadRunner のバージョン情報] を選択します。

- 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。測定値の選択の詳細については、262 ページ「測定値の追加と削除」を参照してください。

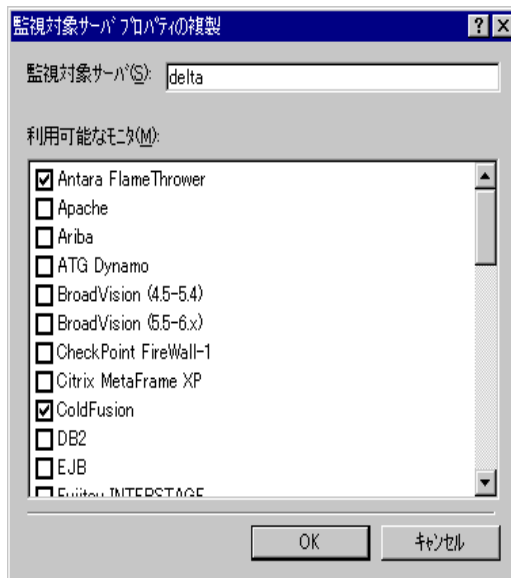
- 6 監視対象サーバをリストにさらに追加するには、1～5 の手順を繰り返します。
- 7 [適用] をクリックして、設定を保存します。

監視対象サーバのプロパティの複製

複数のサーバ・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・ボックスを使って、選択したサーバのプロパティを複製できます。

監視対象サーバのプロパティを複製するには、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、複製するサーバを右クリックして [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、同じサーバ・モニタ・プロパティを使用す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

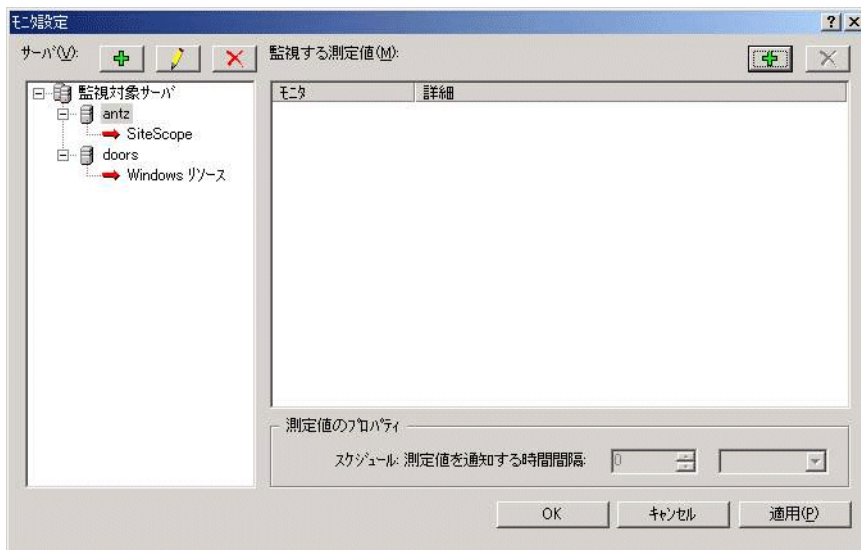
- 複製元のサーバにおいて選択されているモニタが [利用可能なモニタ] リストに表示されます。必要に応じて複製サーバのための追加のモニタを選択します。
- [OK] をクリックして、[監視対象サーバモニタの複製] ダイアログ・ボックスを閉じます。複製されたサーバが [監視対象サーバ] リストに表示されます。
- [適用] をクリックして、設定を保存します。

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。に LoadRunner 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。

- [監視対象サーバ] リストからサーバを選択します。



- 右上隅にある [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。
- 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、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。

測定値のプロパティ

スケジュール: 測定値を通知する時間間隔: 1 Minute(s)

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。

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 (358 ページ「ネットワーク遅延時間モニタの設定」を参照)、次の作業を手順 3 (359 ページ) に追加します。

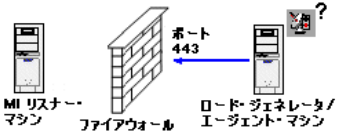
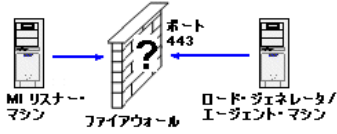

次の形式に従って、[次のマシンからネットワーク遅延を監視する] セクションに監視元マシンのサーバ名または IP アドレスを入力します。 < MI リスナー・マシン > : < 監視元マシンのローカル・キー > 監視元マシンのローカル・キーは、監視元マシンの LoadRunner エージェントを設定するときに選択した一意のキーです。次に例を示します。

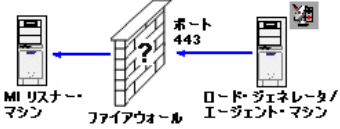
12.12.12.3:vds

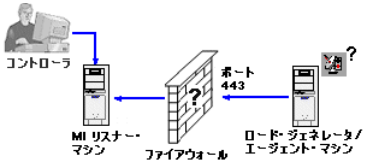
トラブルシューティングのヒント

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。

次の表に、接続の問題のトラブルシューティングに関するヒントを示します。

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、[Agent Settings] の [FireWall] セクションで、「FirewallServiceActive=1」が設定されていません。242 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 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。 注：ファイアウォール越しに仮想ユーザを実行するには、双方向の通信が必要です。したがって、このテストを MI リスナーに対しても実行する必要があります。次のように入力します。 telnet <エージェントの IP アドレス> 443</p>
<p>ポート 443 が使用可能かどうか確認するには</p>  <p>Webサーバ 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 Webサーバ</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>

確認	解決策
<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>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>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。 ● [Agent Settings] と [Agent Configuration] が正しく設定されているかどうか確認します。242 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\magentproc.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。[サービス] > [LoadRunner Agent Service] までブラウズし、このサービスのプロパティを「ローカル システム アカウント」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間の接続を確認するには</p>  <p>コントローラ 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>	<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 (257 ページ「サーバ・モニタのプロパティの設定」を参照してください)。 • メディエータ・マシンで LoadRunner エージェント・プロセスを起動します (242 ページ「ファイアウォールの内側の 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・マシンの名前を [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 (254 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)。 • 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・マシン上のログ・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・マシンで 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・マシンが MI リスナーに接続するまで待つてから、コントローラをメディエータ・マシンに接続します。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・マシンを再起動します。

第4部

シナリオの監視

第 16 章

オンライン・モニタ

LoadRunner の実行時間、トランザクション、Web リソース、システム・リソース、ネットワーク遅延、ファイアウォール・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、データベース・サーバ・リソース、ストリーミング・メディア・リソース、ERP/CRM サーバ・リソース、Java パフォーマンス、アプリケーション配備、ミドルウェア・パフォーマンス・モニタの各モニタを使って、シナリオの実行をオンラインで監視できます。

それぞれのモニタについては、以降の章で解説します。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。

- ▶ モニタの起動
- ▶ オンライン・モニタ・グラフを開く
- ▶ サーバ・リソースの監視
- ▶ グラフ表示のカスタマイズ
- ▶ オンライン・モニタの設定
- ▶ モニタ・オプションの設定
- ▶ オンライン・グラフの設定
- ▶ グラフの結合
- ▶ オンライン・モニタ・グラフについて
- ▶ オンライン測定の設定
- ▶ 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

オンライン・モニタについて

LoadRunner には、次のオンライン・モニタが用意されています。

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。また、仮想ユーザ・スクリプトでユーザが定義したポイントのリアルタイムの値を示す [ユーザ定義データ ポイント] グラフも提供します。

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、第 18 章「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・モニタは、シナリオ実行時に、Web サーバでの統計値を測定します。このモニタでは、シナリオ実行時の 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。Web リソース・モニタの詳細については、第 19 章「Web リソースの監視」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, および Antara FlameThrower のリソースを測定します。システム・リソース・モニタを起動するには、シナリオを実行する前に、モニタ・オプションを設定する必要があります。このオプションの設定の詳細については、第 20 章「システム・リソースの監視」を参照してください。

ネットワーク遅延モニタは、システムのネットワーク遅延に関する情報を示します。ネットワーク遅延モニタを起動するには、シナリオを実行する前に、監視対象のネットワーク・パスを設定する必要があります。詳細については、第 21 章「ネットワークの監視」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。ファイアウォール・モニタを起動するには、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する必要があります。詳細については、第 22 章「ファイアウォール・サーバ・パフォーマンスの監視」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。Web サーバ・リソース・モニタを起動するには、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する必要があります。詳細については、第 23 章「Web サーバ・リソースの監視」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、シナリオ実行時に、Web アプリケーション・サーバの統計値を測定します。Web アプリケーション・サーバ・リソース・モニタを起動するには、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する必要があります。詳細については、第 24 章「Web アプリケーション・サーバ・リソースの監視」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、SQL サーバ、Oracle、Sybase、DB2 データベースに関する統計値を測定します。データベース・サーバ・リソース・モニタを起動するには、シナリオを実行する前に、監視対象の測定値のリストを作成する必要があります。詳細については、第 25 章「データベース・リソースの監視」を参照してください。

ストリーミング・メディア・モニタは、Windows MediaServer 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。ストリーミング・メディア・モニタを起動するには、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する必要があります。詳細については、第 26 章「ストリーミング・メディアの監視」を参照してください。

ERP/CRM サーバ・リソース・モニタは、シナリオ実行中に、SAP R/3 システム・サーバ、SAP Portal、Siebel Web サーバ、Siebel サーバ・マネージャ・サーバの統計値を測定します。ERP/CRM サーバ・リソース・モニタを起動するには、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する必要があります。詳細については、第 27 章「ERP/CRM サーバ・リソースの監視」を参照してください。

Java パフォーマンス・モニタは、J2EE および EJB サーバ・マシンを使用している Java 2 Platform, Enterprise Edition (J2EE) オブジェクト、Enterprise Java Bean (EJB) オブジェクトの統計値を測定します。Java パフォーマンス・モニタを起動するには、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する必要があります。詳細については、第 28 章「Java パフォーマンスの監視」と第 29 章「J2EE パフォーマンスの監視」を参照してください。

Application Deployment ソリューション・モニタは、シナリオ実行時に Citrix MetaFrame XP と MetaFrame 1.8 サーバの統計値を測定します。Application Deployment ソリューション・モニタを起動するには、シナリオを実行する前に、モニタ・オプションを設定する必要があります。このオプションの設定の詳細については、第 30 章「Application Deployment ソリューション」を参照してください。

ミドルウェア・パフォーマンス・モニタは、シナリオ実行時に TUXEDO および IBM WebSphere MQ サーバの統計値を測定します。ミドルウェア・パフォーマンス・モニタを起動するには、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する必要があります。詳細については、第 31 章「ミドルウェアのパフォーマンス監視」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、『LoadRunner アナリシス・ユーザーズ・ガイド』を参照してください。

注：LoadRunner のモニタの詳細については、マーキュリー・インタラクティブの Web サイト (http://www-heva.mercuryinteractive.com/products/loadrunner/load_testing_monitors/supported.html 英語のみ) を参照してください。

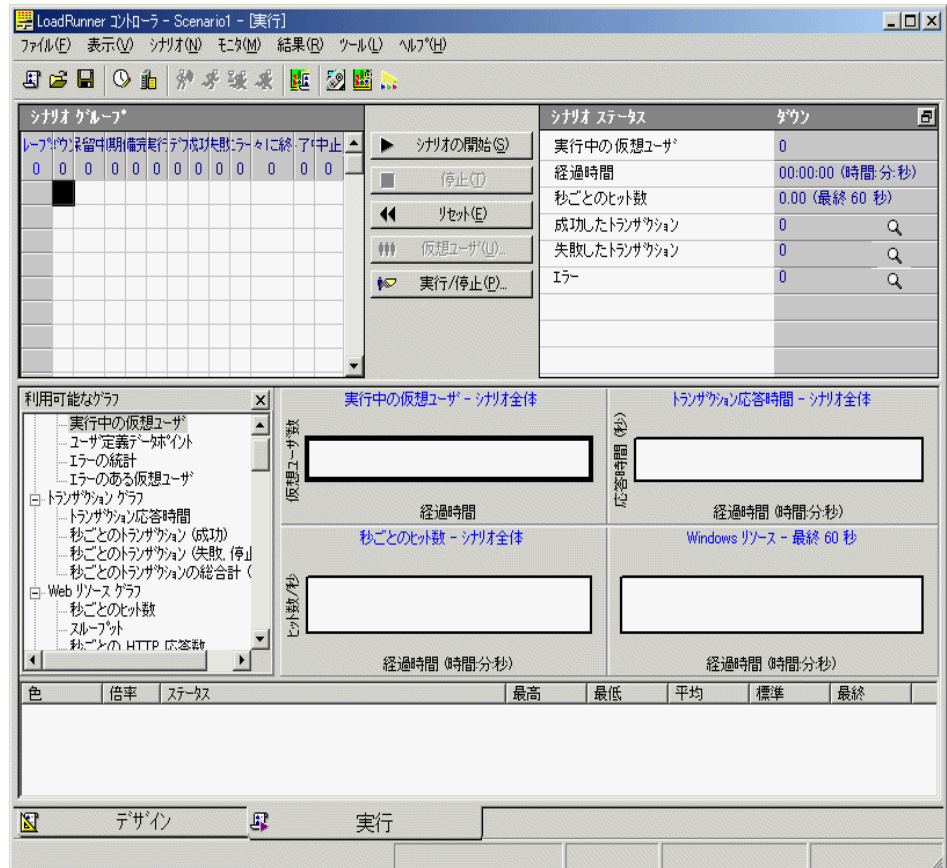
モニタの起動

オンライン・モニタを使って、仮想ユーザのステータス、エラー、トランザクション、システム・リソース、Web リソース、ネットワーク遅延、ファイアウォール・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、データベース・サーバ・リソース、ストリーミング・メディア・リソース、ERP/CRM サーバ・リソース、Java パフォーマンス、アプリケーション配備、およびミドルウェア・パフォーマンス・モニタを監視します。

オンライン・モニタを起動するには、次の手順で行います。

- 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] ボタンをクリックするか、[シナリオ] > [開始] を選択します。
- 2 [実行] タブをクリックします。[シナリオ グループ] 表示枠の下に、標準設定のグラフが表示されます。

▶ シナリオの開始(S)



- 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] > [利用可能なグラフを表示 / 非表示] 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・ツリー表示を非表示にするには、[表示] > [利用可能なグラフを表示 / 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 [X] 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。

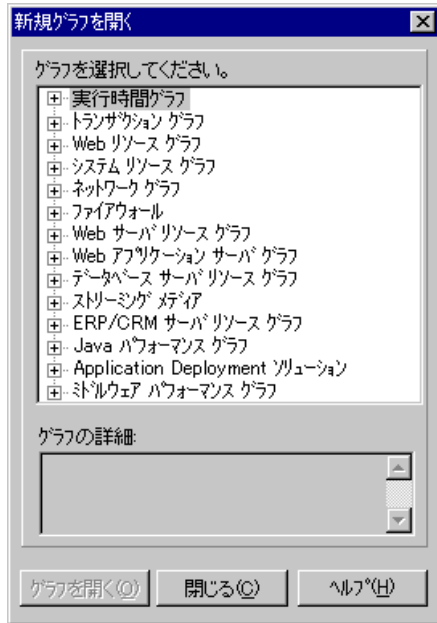
注：トランザクションが実行されていない場合、トランザクション・モニタ・グラフにはデータが表示されません。また、シナリオを実行する前に、監視対象のリソースのリストを作成していないと、システム・リソース、ネットワーク、ファイアウォール、Web サーバ、Web アプリケーション・サーバ、データベース、ストリーミング・メディア、ERP リソース、Java パフォーマンスのグラフにはデータが表示されません。

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、次の手順で行います。

- 1 [モニタ] > [オンライン グラフ] > [新規グラフの追加] 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] をクリックします。グラフ表示領域にグラフが表示されます。

サーバ・リソースの監視

サーバ・リソースの監視では、シナリオの実行中にコントローラで監視する対象となるマシンを指定できます。

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 LoadRunner は、選択されたリソース・モニタをテスト実行中に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。

マシンのリソースを監視するには、グラフ表示領域で監視するグラフを開き、[モニタ] > [測定値の追加] をクリックします。[監視するサーバマシン] ダイアログ・ボックスが開きます。

[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。監視するマシンの名前または IP アドレスと、プラットフォームを入力します。
- ▶ [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）。個々のモニタの詳細については、対応する項を参照してください。

[リソース測定場所]：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]：監視の対象となる選択したマシン上の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。

[説明]：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、対応する項を参照してください。

グラフ表示のカスタマイズ

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・グラフの表示をカスタマイズするには、次の手順で行います。

[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。[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。

オンライン・モニタの設定

LoadRunner では、オンライン・モニタの設定ができます。グラフのサンプリング間隔、線の色、倍率など、グラフの測定値やプロパティを設定できます。

モニタ・オプション：グローバル・サンプリング頻度、エラー処理、デバッグ、頻度設定。詳細については、278 ページ「モニタ・オプションの設定」を参照してください。

グラフのプロパティ：更新頻度、表示タイプ、X 軸のグラフ時間、Y 軸の目盛。詳細については、281 ページ「オンライン・グラフの設定」を参照してください。

測定値の設定：線の色、Y 軸の目盛、線の表示と非表示の切り替え。詳細については、287 ページ「オンライン測定の設定」を参照してください。

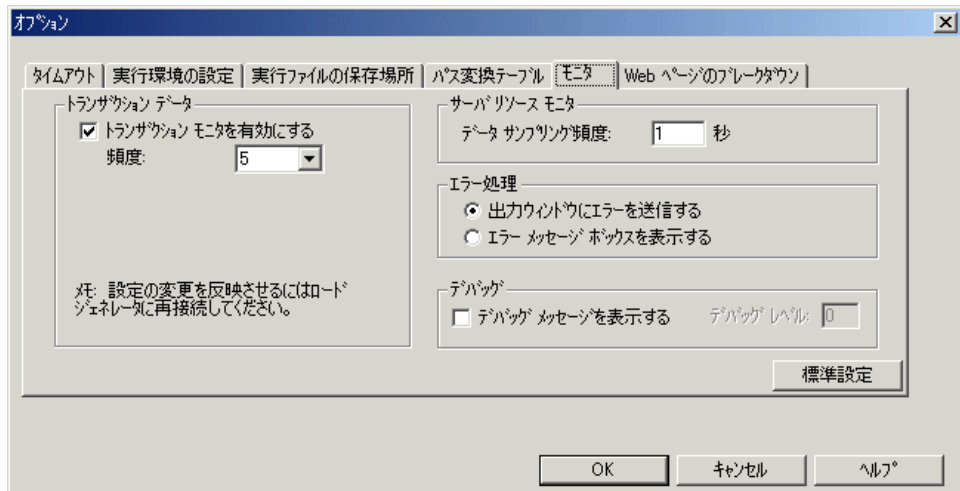
シナリオを保存すると、オンライン・モニタの設定も保存されます。

モニタ・オプションの設定

シナリオを実行する前に、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。

モニタ・オプションを設定するには、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] を選択し、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。

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 サンプリ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] ダイアログ・ボックスの [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。

[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リングする頻度（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、第 18 章「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・モニタの動作を設定します。

- ▶ **[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。LoadRunner がデータを監視するためにシナリオに対して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、281 ページ「オンライン・グラフの設定」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。たとえば、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 がエラー・メッセージをどのように発行するかを制御します。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] をクリックする必要があります。

[**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。

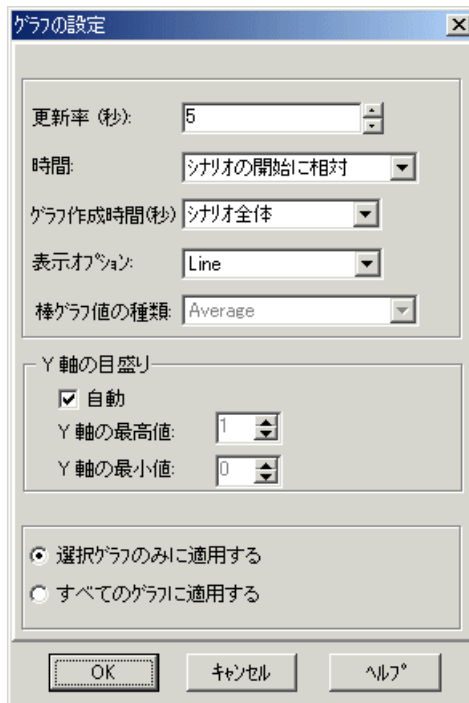
オンライン・グラフの設定

オンライン・グラフについて、次の設定をカスタマイズできます。更新率、時間、グラフ作成時間、表示オプション、棒グラフ値、Y 軸の目盛り、ネットワーク遅延時間ビュー。

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、次の手順で行います。

- 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て [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 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。
- 3 [更新率 (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。

- 4 [時間] ボックスから X 軸のスタイルを選択します。
- 5 [グラフ作成時間 (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。
- 6 [表示オプション] ボックスからグラフのスタイルを選択します。
- 7 選択した表示タイプが「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。[Average] (平均値), [Last Value] (最新の値), [Minimum] (最小値), および [Maximum] (最大値) のいずれかを選択できます。
- 8 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。
- 9 設定をすべてのグラフに適用するか、選択したグラフだけに適用するかを選択します (ネットワーク遅延時間グラフでは指定できません)。
- 10 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。
- 11 [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

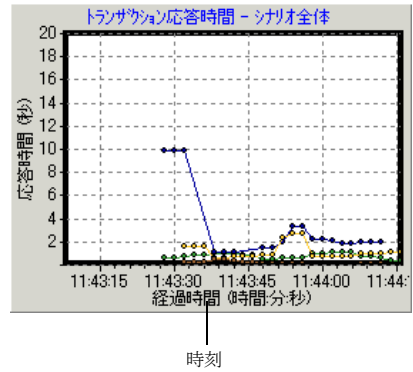
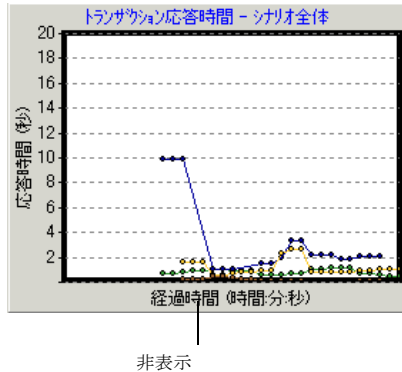
[グラフの設定] ダイアログ・ボックスでは、オンライン・グラフの設定をカスタマイズできます。

[更新率 (秒)] : グラフを新しいデータで更新する間隔。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。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を指定できます。

- ▶ [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] : シナリオの開始からの相対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。



[**グラフ作成時間 (秒)**]: 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、グラフ作成時間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、グラフ作成時間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。

[**表示オプション**]: LoadRunner に対して、グラフを線グラフと棒グラフのどちらか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。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**]: 選択した表示オプションが「Bar」(棒グラフ) の場合は、[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。[**Average**] (平均値)、[**Last Value**] (最新の値)、[**Minimum**] (最小値)、および [**Maximum**] (最大値) のいずれかを選択できます。

[**Y 軸の目盛り**]: 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**] を選択します。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。

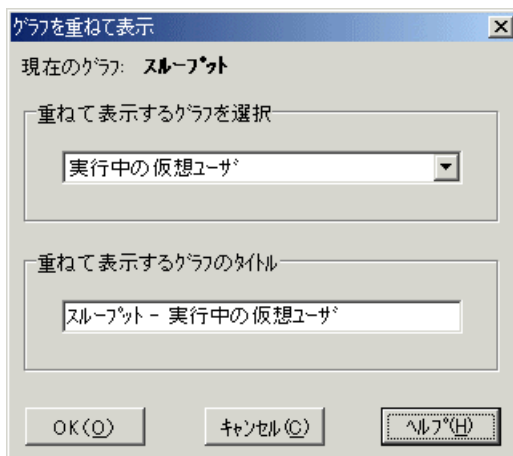
グラフの結合

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、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、次の手順で行います。

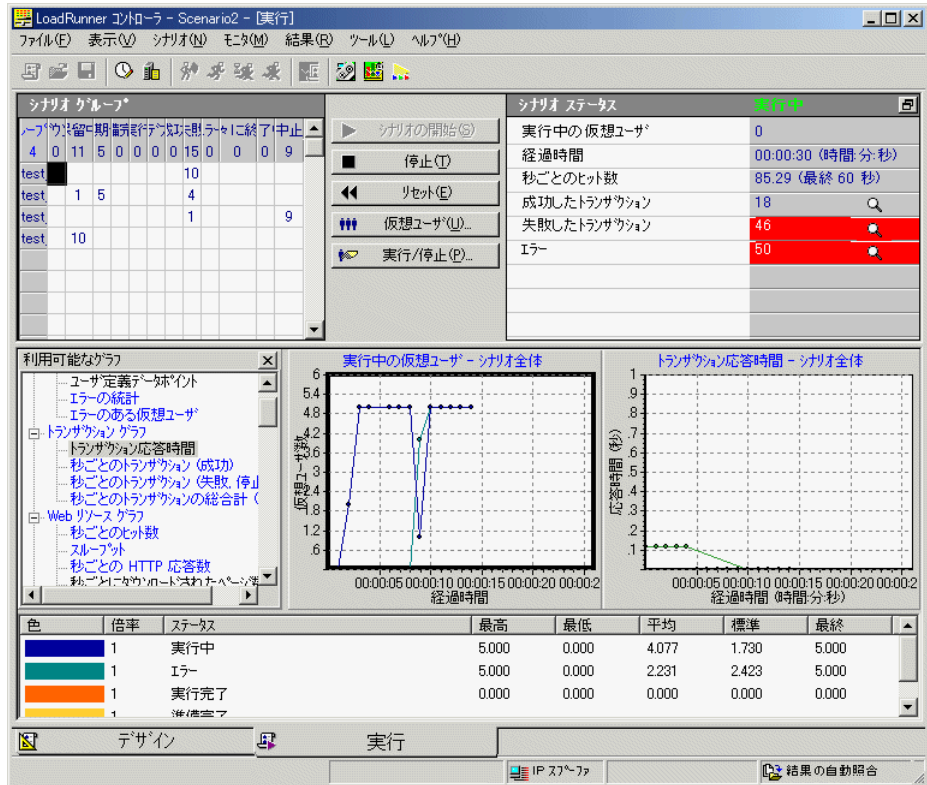
- 1 合成対象のグラフを右クリックして、**[グラフを重ねて表示]** を選択します。**[グラフを重ねて表示]** ダイアログ・ボックスが開きます。



- 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・グラフについて

オンライン・モニタ・グラフには、シナリオによって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。各測定値は色付きの線でグラフに示され、グラフの下の凡例にも同じ色で表示されます。凡例には、選択したグラフの測定値だけが表示されます。



注：ゴール指向シナリオでは、定義したゴールも該当するグラフに表示されます。

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。

特定の線に選択して、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青で表示されます。
- ▶ **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。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、そのグラフを選択して [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。

オンライン測定の設定

[測定値の設定] ダイアログ・ボックスで、次のオンライン測定の設定ができます。

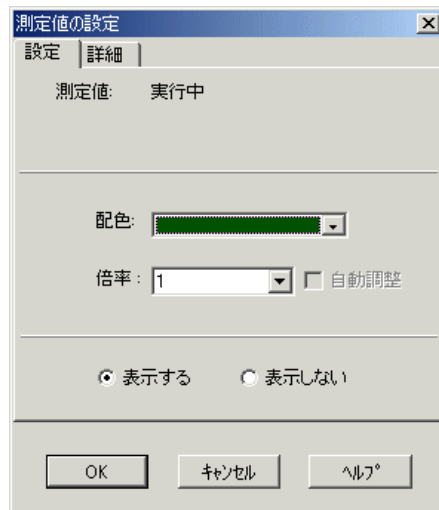
- ▶ 線の色
- ▶ 測定値の倍率
- ▶ トランザクションの表示

線の色

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。

測定値の線の色を変更するには、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適用し、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	倍率	トランザクション	最高	最低	平均	標準	最終
	1	vuser_end_Transaction	1.257	0.051	0.179	0.269	0.053
	1	BookFlight_Transaction	13.397	0.341	1.297	2.070	1.077
	1	vuser_init_Transaction	9.888	0.754	3.803	3.620	2.017
	1	ShowItinerary_Transaction	7.735	0.159	0.832	1.255	0.420

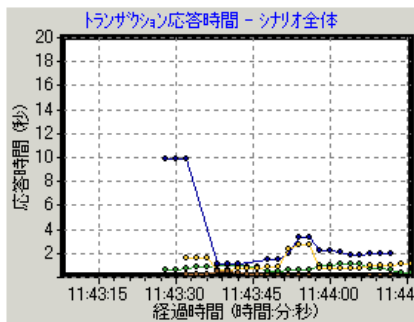
測定値の倍率

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。たとえば、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。

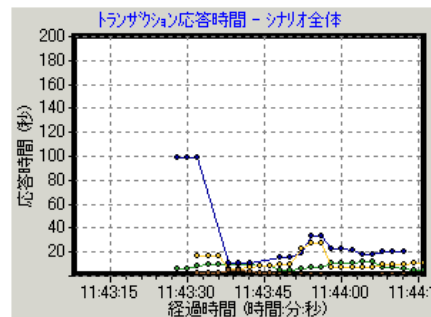
測定値の倍率を設定するには、次の手順で行います。

- 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、[自動調整] 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適用し、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。



倍率 = 1



倍率 = 10

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	倍率	トランザクション	最高	最低	平均	標準	最終
	10	user_end_Transaction	1.257	0.051	0.179	0.269	0.053
	1	BookFlight_Transaction	13.397	0.341	1.297	2.070	1.077

↑
倍率

LoadRunner では [自動調整] 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

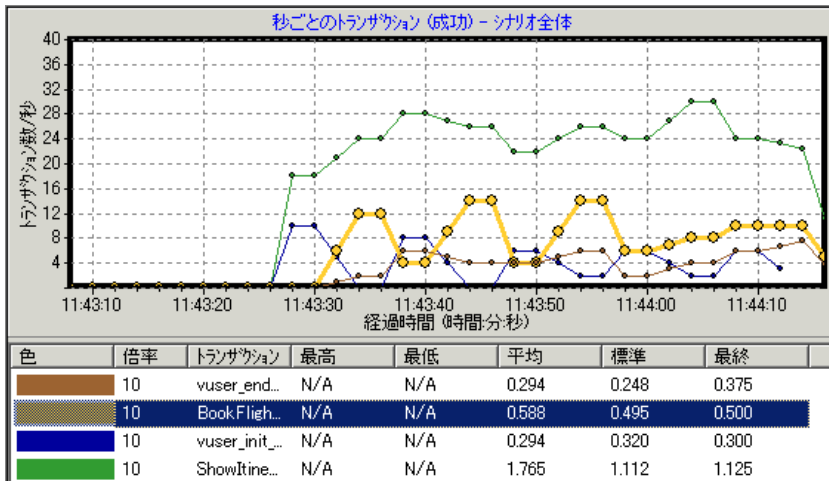
標準設定では、トランザクション・モニタにトランザクション・リストの各項目に対応する線が表示されます。特定の測定値に注目するために、監視中の任意のトランザクションの線を非表示にできます。

トランザクションを表示または非表示にするには、次の手順で行います。

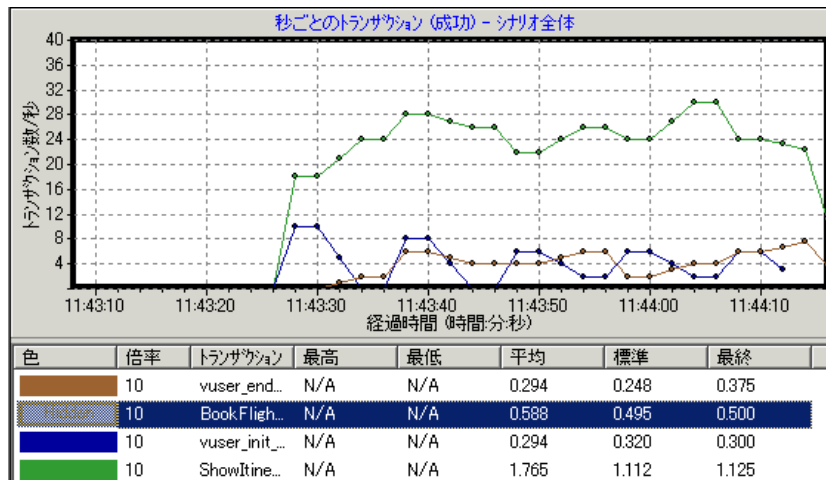
- 1 測定値を非表示にするには、[表示しない] をクリックします。非表示のソースを表示するには、[表示する] をクリックします。
- 2 [OK] をクリックして設定を承認し、ダイアログ・ボックスを閉じます。

また、凡例の測定値を右クリックして [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測定値の表示 / 非表示を切り替えることができます。

次の例では、各測定値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【Network Type】と表示されます。

【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。たとえば、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 を掛ける必要があります。

【自動調整】：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。

【表示する】：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。

【表示しない】：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）。

【説明】：選択した測定値の説明が表示されます。

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。グラフをすべてエクスポートすることも、選択したグラフだけをエクスポートすることもできます。

オンライン・グラフを HTML 形式でエクスポートするには、次の手順で行います。

- 1 特定のグラフをエクスポートするには、エクスポート対象のグラフを選び、**[モニタ] > [オンライングラフ] > [HTMLへエクスポート]** を選択します。**[ファイル名とパスの選択]** ダイアログ・ボックスが開きます。
- 2 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。**[ファイル名とパスの選択]** ダイアログ・ボックスが開きます。
- 3 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、『**LoadRunner アナリシス・ユーザーズ・ガイド**』を参照してください。

第 17 章

リモート・パフォーマンス監視

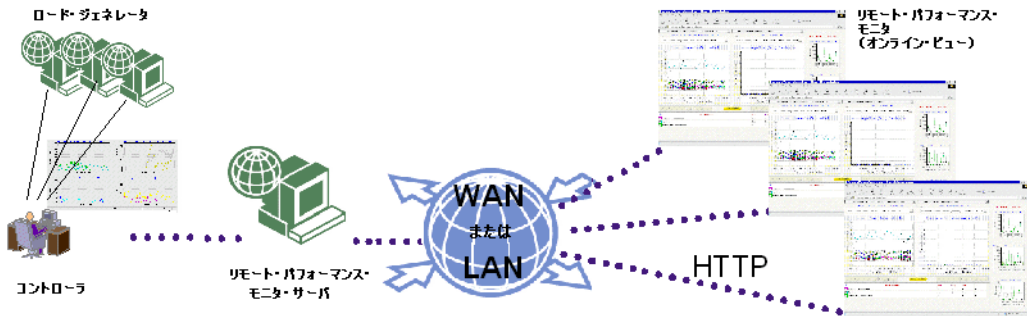
リモート・パフォーマンス監視により、Web ブラウザを使用してリモート・ロケーションから複数の人が LoadRunner シナリオを監視できるようになります。そのため、コントローラ・マシンへアクセスしなくても、ライセンス数分の参加者がテスト結果をオンラインで同時に見ることができます。リモートから監視する人たちは、現在の負荷テストから監視するグラフを選択したり、必要に応じてグラフの設定をカスタマイズしたりできます。

本章では、以下の項目について説明します。

- ▶ リモート・パフォーマンス・モニタ・サーバのインストール
- ▶ LoadRunner リモート・パフォーマンス・モニタ・サーバへの接続
- ▶ 負荷テスト・データの監視
- ▶ オンライン・グラフ表示
- ▶ オンライン・グラフ設定のカスタマイズ

リモート・パフォーマンス監視について

負荷テスト実行中、リモート・パフォーマンス・モニタによって、仮想ユーザがサーバ上で生成する負荷の状況をグラフ表示で見ることができます。ユーザは、Web サーバに接続されている Web ブラウザで、負荷テストのデータを見ることができます。



リモート・パフォーマンス・モニタ・サーバには、ASP ページが実装されている Web サイトと、負荷テスト・グラフが含まれるファイル・サーバが含まれます。リモート・パフォーマンス・モニタ・サーバは、コントローラのオンライン・コンポーネントとやり取りを行い、負荷テストを表示できる同時ユーザ数をライセンスの内容に応じて管理します。

コントローラは必要に応じてオンライン・グラフを生成するためにサーバとやり取りを行うアプリケーションを実行します。

グラフとモニタ測定値の詳細については、第 16 章「オンライン・モニタ」を参照してください。

リモート・パフォーマンス・モニタ・サーバのインストール

リモート・ロケーションからサーバのパフォーマンスを監視するには、LoadRunner コントローラ 7.8 の CD からリモート・パフォーマンス・モニタ・サーバをインストールする必要があります。

リモート・パフォーマンス・モニタ・サーバのインストールの詳細については、『LoadRunner インストール・ガイド』を参照してください。

インストールの要件

リモート・パフォーマンス・モニタ・サーバによりリモート・パフォーマンス・モニタ用 IIS Web サーバが構成されます。それには次のコンポーネントをインストールしたマシンが必要です。

IIS サーバ	5.0
オペレーティング・システム	Windows 2000 Server, Windows 2000 Advanced Server
クライアント・ブラウザ	Internet Explorer 5.0 以降, Netscape 6.2 以降

IIS Web サーバはコントローラおよびリモート・パフォーマンス・モニタと通信してユーザの要求を処理し、オンラインのグラフとグラフの凡例を作成します。

リモート・パフォーマンス・モニタのユーザ設定の構成

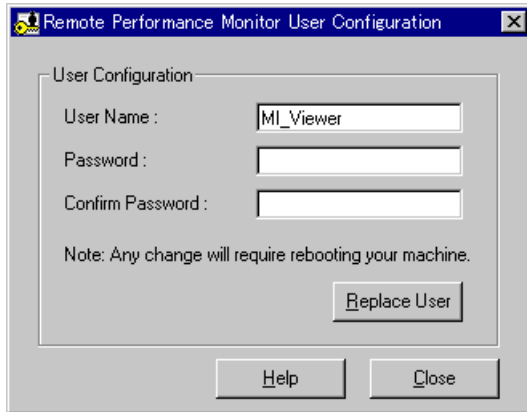
リモート・パフォーマンス・モニタのユーザ構成ツールにより、標準設定または、Web サーバにリモート・パフォーマンス・モニタをインストールしたときに使用したユーザ定義のユーザ名とパスワードを変更できます。このツールを使って、コントローラ・マシン上のリモート・パフォーマンス・モニタのユーザ設定の更新もできます。LoadRunner は、Web サーバとコントロール・マシン間の認証にユーザ名とパスワードを使用するので、この情報は両方のマシンで同じでなければなりません。

ユーザ構成

コントローラ・マシンと Web サーバ・マシン両方でリモート・パフォーマンス・モニタのユーザ構成ツールを使用して、ユーザ設定を構成する必要があります。

コントローラ側のユーザ設定の変更は、次の手順で行います。

- 1 [スタート] > [プログラム] > [LoadRunner] > [Tools] > [RPM User Configuration] を選択して、コントロール・マシンでリモート・パフォーマンス・モニタのユーザ構成ツールを開きます。



- 2 [Remote Performance Monitor User Configuration] ダイアログ・ボックスで、ユーザ名とパスワードを入力し、パスワードの確認をします。
- 3 [Replace User] ボタンをクリックします。構成プログラムによってマシンを再起動するよう求められます。再起動は後で行ってもかまいません。

注： [Replace User] ボタンをクリックすると、すぐに変更が行われます。ただし、マシンを再起動してからでないとシステムは正しく動作しません。

Web サーバのユーザ設定の変更は、次の手順で行います。

- 1 [スタート] > [プログラム] > [RPM Server] を選択して、Web サーバでリモート・パフォーマンス・モニタのユーザ設定ツールを開きます。
- 2 [Remote Performance Monitor User Configuration] ダイアログ・ボックスで、コントローラ・マシンで入力したのと同じユーザ名とパスワードを入力します。パスワードの確認を行います。
- 3 [Replace User] ボタンをクリックします。構成プログラムによってマシンを再起動するよう求められます。

注：[Replace User] ボタンをクリックすると、すぐに変更が行われます。ただし、マシンを再起動してからでないとシステムは正しく動作しません。

注：リモート・パフォーマンス・モニタのユーザ名とパスワードは IIS Web サーバで自動的に更新されます。

LoadRunner リモート・パフォーマンス・モニタ・サーバへの接続

LoadRunner リモート・パフォーマンス・モニタに接続するには、Web ブラウザで次のパスを入力します。

`http:// < IIS Web サーバ・マシンの名前 > /remoteview`

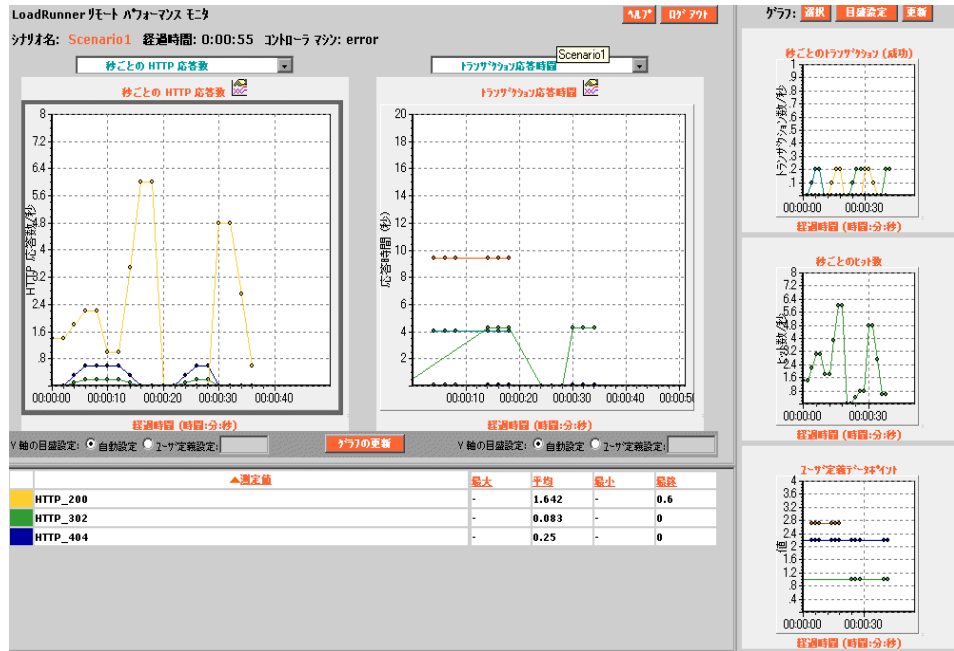
LoadRunner リモート・パフォーマンス・モニタのログオン・ページが開きます。



LoadRunner リモート・パフォーマンス・モニタへのログオンは、次の手順で行います。

- 1 [ユーザ ID] ボックスに、Admin と入力します。
- 2 [パスワード] ボックスに、Admin と入力します。
- 3 [コントローラ マシン] ボックスに、アクセスするコントローラ・マシンの名前または IP アドレスを入力します。

- 4 [Login] をクリックします。LoadRunner リモート・パフォーマンス・モニタのページが開きます。



標準設定では、左のグラフが選択されて、その測定値が測定値凡例に表示されます。

注：20 分間ブラウザの活動がない場合、リモート・パフォーマンス監視セッションはタイムアウトになります。セッションを続けるには、ログインし直す必要があります。

負荷テスト・データの監視

負荷テスト実行中に負荷テスト・データを監視して、テストのステータスおよび Web サーバに対する負荷の影響の概要を素早く把握できます。

リモート・パフォーマンス・モニタのページの上部に、現在実行中のテストのステータスが表示されます。

シナリオ名: **Scenario1** 経過時間: 0:00:52 コントローラ マシン: error

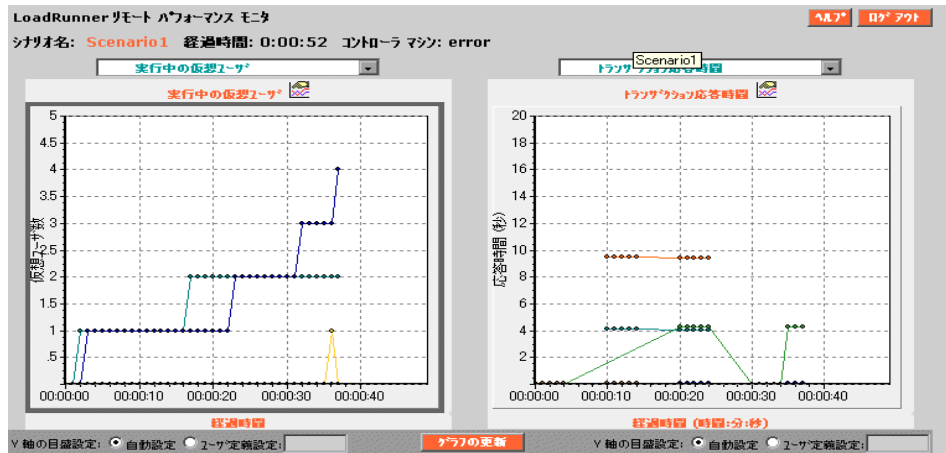
リモート・パフォーマンス・モニタのページには、実行中のテストの名前、テストの経過時間、およびコントローラ・マシンの名前が表示されます。

オンライン・グラフ表示

グラフは表示枠に分かれて 5 つのグラフを同時に表示できます。大きいグラフが 2 つと小さいグラフが 3 つです。さらに、凡例にグラフ測定値が表示されます。

負荷テスト中にグラフを表示するには、次の手順で行います。

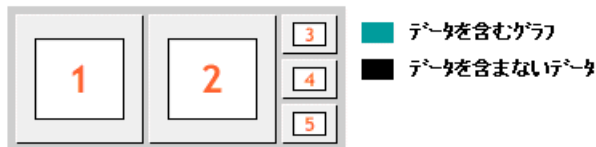
- 1 大きいグラフ表示枠にグラフを表示するには、大きいグラフ表示枠の上にあるドロップダウン・グラフ・リストからグラフを選択します。選択したグラフがページに再ロードされます。



注：使用可能なグラフはドロップダウン・グラフ・リストに緑色で表示されます。使用できない（黒）グラフを選択すると、空の表示枠が現れます。

- 小さいグラフ表示枠にグラフを表示するか、または画面に表示されているグラフのどれかを変更するには、小さいグラフの上にある「**選択**」ボタンをクリックします。「グラフの選択」ウィンドウが開きます。

Select Graphs



グラフ名	1	2	3	4	5
実行中の仮想ユーザ	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
ユーザ定義データポイント	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
エラーの統計	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
エラーのある仮想ユーザ	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
トランザクション応答時間	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
秒ごとのトランザクション (成功)	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
秒ごとのトランザクション (失敗, 停止)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
秒ごとのトランザクションの総合計 (成功)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
秒ごとのヒット数	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
スループット	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
秒ごとの HTTP 応答数	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
秒ごとにダウンロードされたページ数	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
秒ごとの再試行数	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Real クライアント	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

- リストから任意のグラフとそれが表示される位置を選択します。上の図では、位置に番号が振ってあります。
- [**OK**] をクリックして、「グラフの選択」ウィンドウを閉じます。選択したグラフがリモート・パフォーマンス・モニタのページに表示されます。

グラフの凡例

大きいグラフのウィンドウにはグラフの測定値を表示できます。標準設定では、リモート・ビューアが開くと、左のグラフが選択されて（グレーの境界が強調表示される）、その測定値が測定値凡例に表示されます。

注：小さいグラフの測定値は凡例に表示できません。表示するには、測定するリソースを大きいグラフとして開く必要があります。

グラフの凡例の表示は、次の手順で行います。

- 1 グラフ表示枠をクリックしてグラフを選択します。グラフがグレーの境界線で強調表示され、その測定値が凡例に表示されます。
- 2 つのグラフの測定値を表示するには、大きいグラフを選択します。凡例が縦に分割され、両方のグラフの測定値が表示されます。
- 2 凡例には各測定値の最大値、平均値、最小値、直前の値に関する詳細が表示されます。これらの値の1つで測定値をソートするには、列見出し（最大、平均、最小、最終）をクリックします。列見出しの脇にアイコンが表示されて、測定値が昇順、降順どちらでソートされているかがわかります。
- 3 グラフの凡例を閉じるには、グラフ表示枠をクリックします。

オンライン・グラフ設定のカスタマイズ

負荷テスト実行中に、リモート・パフォーマンス・モニタから、次のオンライン・グラフ設定を変更できます。

- ▶ グラフの倍率設定
- ▶ 更新率
- ▶ グラフ測定値の構成

標準設定を変更しても、現在実行中のテストにしか適用されず、将来の負荷テスト用に保存されることはありません。

グラフの倍率設定

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。X 軸は [経過時間] を表し、変更できません。標準設定では、LoadRunner は [自動] オプションを使います。このオプションでは、グラフの表示に最適なように自動的に測定値を設定します。

大きいグラフの倍率を設定するには、次の手順で行います。

- 1 大きいグラフの下にある Y 軸の値の部分で、[ユーザ定義設定] を選択し、ボックスに値を入力して、[グラフの更新] をクリックします。指定した値を Y 軸の上限としてグラフが再描画されます。
- 2 通常の倍率のグラフを表示するには、[自動] を選択して、[グラフの更新] をクリックします。

小さいグラフの倍率を設定するには、次の手順で行います。

- 1 小さいグラフの上にある [拡大縮小] ボタンをクリックします。[グラフの拡大縮小] ウィンドウが開きます。



301 ページの [グラフの選択] ウィンドウに示す 3, 4, 5 の位置にある小さいグラフの Y 軸の測定値を変更できます。

- 2 異なる倍率を使用するには、[ユーザ定義設定] を選択して、ボックスに値を入力します。
- 3 通常の倍率のグラフを表示するには、[自動] を選択します。
- 4 [OK] をクリックして、[グラフの選択] ウィンドウを閉じます。指定した値を Y 軸の上限としてグラフが再描画されます。

注：グラフの倍率の詳細については、第 16 章「オンライン・モニタ」の「測定値の倍率」を参照してください。

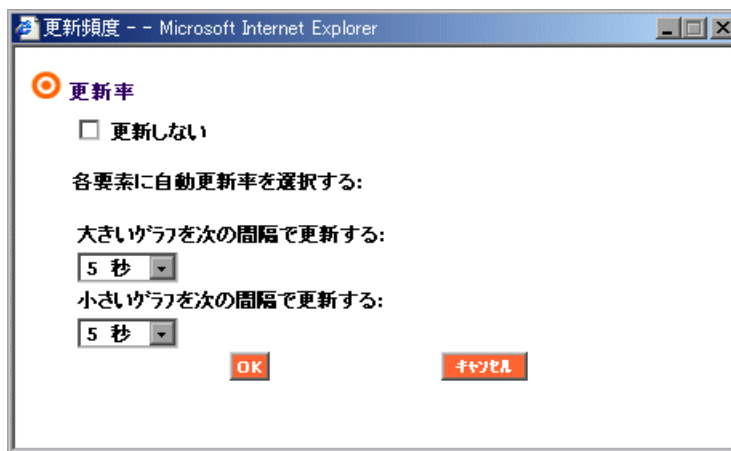
更新率

標準設定では、リモート・パフォーマンス・モニタのページのグラフは 5 秒ごとに更新されます。[更新率] オプションを使用して、標準設定の更新率を変更できます。更新率の値を大きくすると、グラフの更新頻度が少なくなります。

注：大きい負荷テストでは、小さいグラフに対しては大きい更新率の使用が推奨されます。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。

標準設定の更新率を変更するには、次の手順で行います。

- 1 小さいグラフの上にある [更新] ボタンをクリックします。[更新率] ウィンドウが開きます。



標準設定では、更新オプションが有効になっており、5 秒間隔ですべてのグラフが自動的に更新されます。自動更新を無効にするには、[更新しない] チェック・ボックスを選択します。

- 2 大きいグラフと小さいグラフの更新頻度を選択します。
- 3 [OK] をクリックして自動更新率を変更し、負荷テストの実行ページに戻ります。

注：グラフと凡例を直ちに更新するには [グラフの更新] ボタンをクリックします。

グラフの測定値の構成

グラフをカスタマイズして選択した測定値を表示，非表示，または強調表示にできます。

グラフの測定値を構成するには，次の手順で行います。



- 1 大きいグラフの表示枠の上部にあるアイコンをクリックしてグラフの測定値を構成します。グラフの測定値の構成ページが開きます。



- 2 [表示] チェック・ボックスを選択してグラフに測定値を表示します。標準設定では，グラフの測定値がすべてグラフに表示されます。1つの測定値をグラフに表示されないようにするには，[表示] チェック・ボックスをクリアします。1つの測定値をグラフに太字で強調表示するには，[太字] チェック・ボックスを選択します。

表示カラムの [Select/Deselect All] チェック・ボックスを選択すると，グラフにすべての測定値が表示されます。[Select/Deselect All] チェック・ボックスをクリアすると，グラフからすべての測定値が削除されます。

すべての測定値をグラフに太字で強調表示するには，Bold カラムの [Select/Deselect All] チェック・ボックスを選択します。太字の強調表示を削除するには，[Select/Deselect All] チェック・ボックスをクリアします。

- 3 **[OK]** をクリックして、グラフの測定値の構成ページを閉じます。グラフが更新されると、新しいグラフの設定構成が反映されます。

ログアウト

LoadRunner リモート・パフォーマンス・モニタからログアウトするには、ページの上にある **[ログアウト]** ボタンをクリックします。

第 18 章

実行環境とトランザクションの監視

シナリオの実行時に、LoadRunner の実行時間モニタとトランザクション・モニタを使って、実行環境のステータスとトランザクションのパフォーマンスのグラフを表示できます。

本章では、以下の項目について説明します。

- ▶ 実行時間 グラフ
- ▶ ユーザ定義データ・ポイント・グラフ
- ▶ トランザクション・モニタ・グラフ
- ▶ トランザクション・モニタを有効にする
- ▶ スクリプトへのトランザクションの追加
- ▶ Web ページ・ブレイクダウンの有効化





実行時間グラフとトランザクション・グラフについて

実行時間モニタは、シナリオに参加している仮想ユーザのステータス情報や仮想ユーザが生成するエラーの数と種類を表示します。また、仮想ユーザ・スクリプトでユーザが定義したポイントのリアルタイムの値を示す [ユーザ定義データ・ポイント] グラフも提供します。

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。トランザクションの詳細については、314 ページ「スクリプトへのトランザクションの追加」を参照してください。

実行時間グラフ

モニタの**実行中の仮想ユーザ**・グラフには、すべてのロード・ジェネレータ・マシンで現在のシナリオを実行した場合の仮想ユーザのステータスが表示されます。グラフには実行している仮想ユーザの数が表示されます。グラフの凡例に含まれる情報は、各ステータスの仮想ユーザの数を示します。

色	倍率	ステータス	最高	最低	平均	標準	最終
	1	実行完了	20.000	0.000	6.000	6.370	20.000
	1	実行中	5.000	0.000	2.343	1.956	0.000
	1	準備完了	5.000	0.000	0.143	0.424	0.000
	1	エラー	0.000	0.000	0.000	0.000	0.000

各仮想ユーザの [ステータス] フィールドには、仮想ユーザの現在のステータスが表示されます。次の表に、各仮想ユーザのステータスを説明します。

ステータス	説明
実行	すべてのロード・ジェネレータ上で現在実行している仮想ユーザの合計数
準備完了	スクリプトの初期化セクションを完了した実行可能な仮想ユーザの数
実行完了	実行が完了した仮想ユーザの数。成功した仮想ユーザと失敗した仮想ユーザの両方が含まれます。
エラー	実行の際にエラーが発生した仮想ユーザの数。エラーの詳細については、[仮想ユーザ] ウィンドウの [ステータス] フィールドまたは [出力メッセージ] ウィンドウを参照してください。

モニタの**エラーの統計**グラフには、シナリオ実行の経過秒ごとに増加するエラーの数の詳細が表示されます。エラーは、エラーの発生元によりグループ分けされます。たとえば、スクリプト内の場所やロード・ジェネレータ名などに分けられます。

エラーのある仮想ユーザ・グラフには、シナリオ実行中にエラーが発生した仮想ユーザの数の詳細が表示されます。エラーは、エラーの発生元によりグループ分けされます。

ユーザ定義データ・ポイント・グラフ

ユーザ定義データ・ポイント・グラフは、ユーザが定義したデータ・ポイントのリアルタイムの値を示します。データ・ポイントは、仮想ユーザ・スクリプトの適切な箇所に `lr_user_data_point` 関数を挿入することで定義します（GUI 仮想ユーザの場合 `user_data_point`、Java 仮想ユーザの場合 `lr.user_data_point`）。

```
Action1()
{
    lr_think_time(1);
    lr_user_data_point ("data_point_1",1);
    lr_user_data_point ("data_point_2",2);
    return 0;
}
```

Web や Oracle NCA などのグラフィカルなスクリプト表現をサポートする仮想ユーザ・プロトコルでは、データ・ポイントを「ユーザ定義」ステップとして挿入します。データ・ポイント情報は、スクリプトが関数またはステップを実行するたびに収集されます。データ・ポイントの詳細については、『[LoadRunner オンライン関数リファレンス](#)』を参照してください。

標準設定では、LoadRunner は、1つのグラフにすべてのデータ・ポイントを表示します。凡例には、各データ・ポイントに関する情報が表示されます。必要に応じて、グラフの下の凡例を使って特定のデータ・ポイントを非表示にできます。

シナリオの実行後に、オフラインでデータ・ポイントを表示することもできます。詳細については、『[LoadRunner アナリシス・ユーザーズ・ガイド](#)』を参照してください。

トランザクション・モニタ・グラフ

トランザクション・モニタは、次のグラフを表示します。

- ▶ トランザクション応答時間
- ▶ 秒ごとのトランザクション（成功）
- ▶ 秒ごとのトランザクション（失敗，中止）
- ▶ 秒ごとのトランザクションの総合計（成功）

トランザクション応答時間グラフには、シナリオを開始してからの経過時間（X 軸）の関数として、トランザクションの平均応答時間（Y 軸）が秒単位で表示されます。

秒ごとのトランザクション（成功） グラフには、シナリオを開始してからの経過時間（X 軸）の関数として、1 秒あたりに成功したトランザクションの数（Y 軸）が表示されます。

秒ごとのトランザクション（失敗，中止） グラフには、シナリオを開始してからの経過時間（X 軸）の関数として、1 秒あたりの失敗や停止のトランザクションの数（Y 軸）が表示されます。

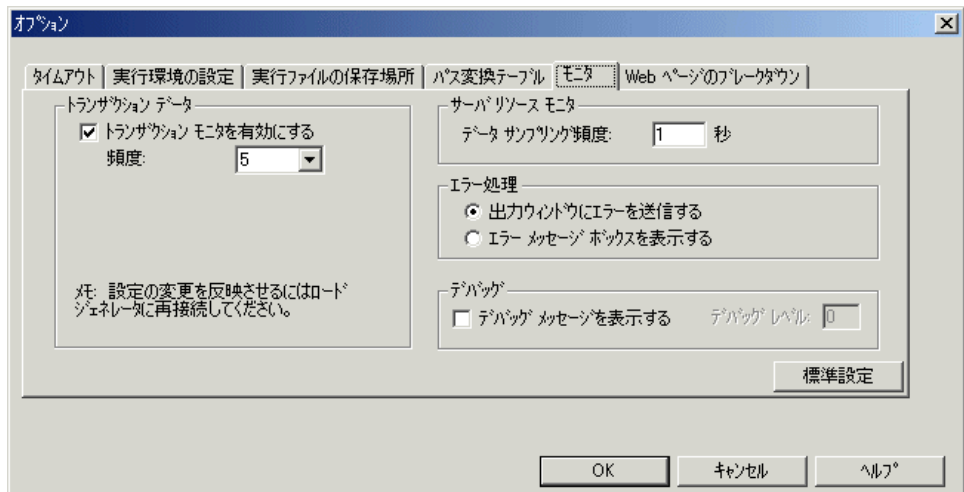
秒ごとのトランザクションの総合計（成功） グラフには、シナリオを開始してからの経過時間（X 軸）の関数として、1 秒あたりに成功して完了したトランザクションの総数（Y 軸）が表示されます。

トランザクション・モニタを有効にする

トランザクション・モニタは標準設定で有効になっており、シナリオの開始時に仮想ユーザのトランザクションを監視し始めます。リソースを節約するために、トランザクション・モニタを無効にすることもできます。

トランザクション・モニタを有効にするには、次の手順で行います。

- 1 [ツール] > [オプション] を選択して、[モニタ] タブを選びます。



- 2 [トランザクション モニタを有効にする] チェック・ボックスを選択して、トランザクションの監視を有効にします。トランザクションの監視を無効にするには、[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ます。

スクリプトへのトランザクションの追加

仮想ユーザ・スクリプトにトランザクションが定義されていない場合、オンライン・グラフにデータが表示されません。既存のスクリプトにトランザクションを追加するには、適切なツールを使ってそのスクリプトを編集します。次の表に、各スクリプト・タイプに対応するスクリプト作成ツールを示します。

スクリプト・タイプ	編集ツール
GUI ウィンドウ	WinRunner
非 GUI ウィンドウ	VuGen (仮想ユーザ・ジェネレータ)
SAP	QuickTest for SAP

スクリプトへのトランザクションの追加は、次の手順で行います。

- 1 [デザイン] タブをクリックして、仮想ユーザ・グループとスクリプトのリストを表示します。
- 2 仮想ユーザ・グループのスクリプトを編集するには、グループを選択し、[シナリオグループ] ウィンドウの右にある [スクリプトの表示] ボタンをクリックします。スクリプト作成ツールが起動します。

個々の仮想ユーザのスクリプトを編集するには、[仮想ユーザ] をクリックします。編集する仮想ユーザを右クリックして [スクリプトの表示] を選択し、スクリプト作成ツールを起動します。

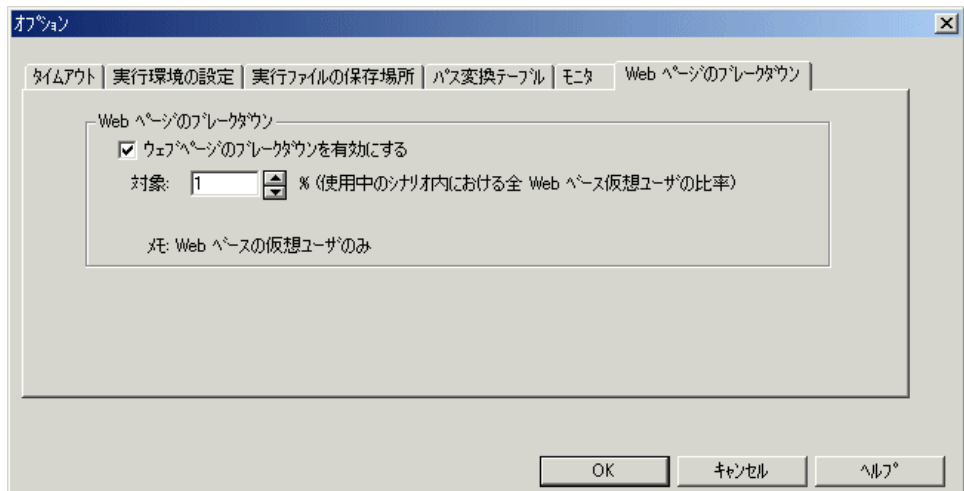
- 3 トランザクション開始とトランザクション終了の関数またはマーカをスクリプト全体に挿入します。

詳細については、「**LoadRunner へようこそ**」に記載されている適切なユーザーズ・ガイドを参照してください。

Web ページ・ブレイクダウンの有効化

アナリシスで、スクリプトに定義されている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を表示する [Web ページブレイクダウン] グラフを生成するには、シナリオを実行する前にコントローラで Web ページ・ブレイクダウン機能を有効にしておく必要があります。

[ツール] > [オプション] を選択して、[Web ページのブレイクダウン] タブを選びます。



[Web ページ・ブレイクダウンの有効化]

- ▶ [ウェブページのブレイクダウンを有効にする]: アナリシスで Web ページ・ブレイクダウン・グラフを表示できるようにします。
- ▶ [対象 X % (使用中のシナリオ内における Web ベース仮想ユーザの比率)]: 実行するトランザクション・ブレイクダウンに Web 仮想ユーザの割合を指定します。

注: この機能は、Web ベースの仮想ユーザにのみ適用されます。

[Web ページブレイクダウン] グラフの詳細については、『LoadRunner アナリシス・ユーザズ・ガイド』を参照してください。

第 19 章

Web リソースの監視

LoadRunner の Web リソース・モニタを使って、Web サーバのパフォーマンスに関する情報を取得できます。

本章では、以下の項目について説明します。

- ▶ [秒ごとのヒット数] グラフ
- ▶ [スループット] グラフ
- ▶ [秒ごとの HTTP 応答数] グラフ
- ▶ [秒ごとにダウンロードされたページ] グラフ
- ▶ [秒ごとの再試行数] グラフ
- ▶ [接続] グラフ
- ▶ [秒ごとの接続数] グラフ
- ▶ [秒ごとの SSL 接続] グラフ

Web リソースの監視について

Web リソース・モニタでは、Web サーバのスループット、シナリオの実行中に発生した秒ごとのヒット数、秒ごとの HTTP 応答数、Web サーバから返された HTTP ステータス・コード（「要求が成功しました」、「ページが見つかりません」などの HTTP 要求のステータスを示す）、秒ごとにダウンロードされたページ、秒ごとの再試行数を分析できます。

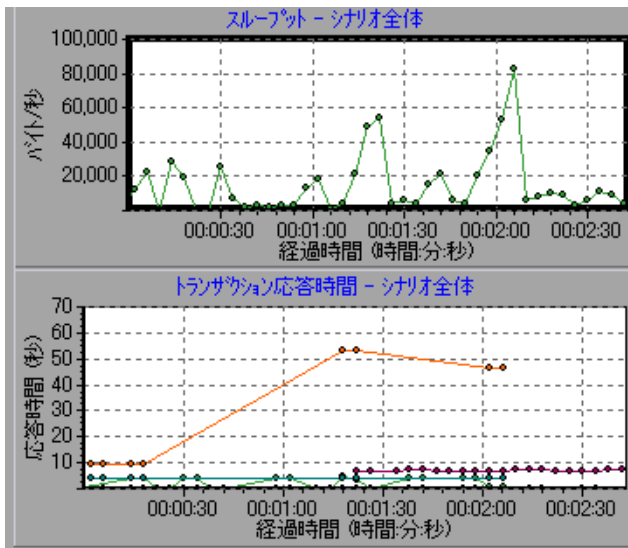
[秒ごとのヒット数] グラフ

[秒ごとのヒット数] グラフには、シナリオを開始してからの経過時間（X 軸）の関数として、Web サーバに対するヒット（HTTP 要求）数（Y 軸）が表示されます。このグラフには、シナリオ全体、最後の 60 秒、180 秒、600 秒、3600 秒を表示できます。このグラフと [トランザクション応答時間] グラフを比較して、ヒットの数がトランザクション・パフォーマンスにどのように影響するかを知ることができます。

[スループット] グラフ

[スループット] グラフには、シナリオ実行中の秒ごと（X 軸）の、Web サーバ上のスループット（Y 軸）が表示されます。スループットはバイト単位で測定され、任意の指定された 1 秒間に、仮想ユーザがサーバから受け取ったデータ量を表します。このグラフと [トランザクション応答時間] グラフを比較して、スループットがトランザクション・パフォーマンスにどのように影響するかを知ることができます。

次の例では、[トランザクション応答時間] グラフと [スループット] グラフを比較しています。グラフからは、スループットが減少するにつれてトランザクション応答時間も短くなっているように見えます。スループットのピークが生じたのは、シナリオ開始後約 1 分の時点です。応答時間が最も長かったのも、この時間です。



[秒ごとの HTTP 応答数] グラフ

[秒ごとの HTTP 応答数] グラフには、シナリオ実行中の秒ごと（X 軸）の、Web サーバから返された HTTP ステータス・コード（「要求が成功しました」、「ページが見つかりません」などの HTTP 要求のステータスを示す）の数（Y 軸）が、ステータス・コード別にグループ分けして表示されます。スクリプトによって（「Group By」関数を使って）グラフに表示された結果をグループ分けして、エラー・コードを生成したスクリプトを検索できます。

次の表は、HTTP ステータス・コードの一覧です。

コード	説明
200	OK
201	Created
202	Accepted
203	Non-Authoritative Information
204	No Content
205	Reset Content
206	Partial Content
300	Multiple Choices
301	Moved Permanently
302	Found
303	See Other
304	Not Modified
305	Use Proxy
307	Temporary Redirect
400	Bad Request
401	Unauthorized
402	Payment Required

コード	説明
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Request Entity Too Large
414	Request - URI Too Large
415	Unsupported Media Type
416	Requested range not satisfiable
417	Expectation Failed
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	HTTP Version not supported

上記のステータス・コードの詳細については、
<http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10> を参照してください。

[秒ごとにダウンロードされたページ] グラフ

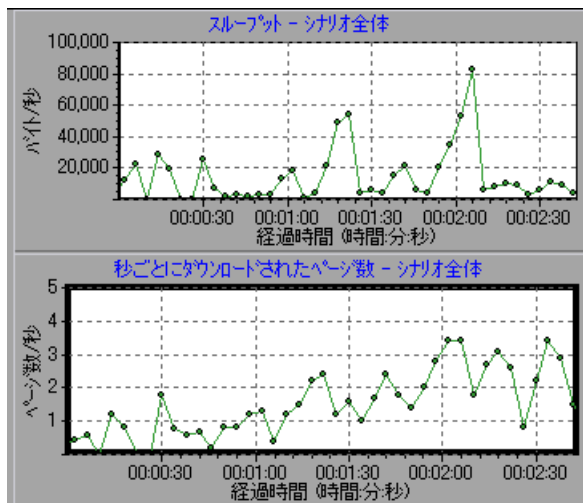
[秒ごとにダウンロードされたページ] グラフには、シナリオ実行中の秒ごとの、サーバからダウンロードされた Web ページ数 (Y 軸) が表示されます。このグラフは、仮想ユーザが生成する負荷の量をダウンロードされた Web ページ数を単位として測るのに使用できます。

注: [秒ごとにダウンロードされたページ数] グラフを表示するには、シナリオを実行する前に、スクリプトの [実行環境の設定] ダイアログ・ボックスの [お気に入り] タブで、[秒ごとのページ数 (HTML モードのみ)] を選択しておく必要があります。

[スループット] グラフ同様、[秒ごとにダウンロードされたページ数] グラフは、任意の時点で仮想ユーザがサーバから受信したデータ量を表します。

- ▶ [スループット] グラフは、各リソースとそのサイズ (たとえば、1つ1つの .gif ファイルや Web ページのサイズなど) を考慮します。
- ▶ [秒ごとにダウンロードされたページ数] グラフは、ページ数だけを測ります。

次の例では、[スループット] グラフと [秒ごとにダウンロードされたページ数] グラフを比較しています。このグラフでは、スループットと秒ごとにダウンロードされたページ数は比例していないように見えます。たとえば、シナリオを開始してから 15 ~ 16 秒の間、スループットは低下しているのに、秒ごとにダウンロードされたページ数は増加しています。



[秒ごとの再試行数] グラフ

[秒ごとの再試行数] グラフには、シナリオを開始してからの経過時間（X 軸）の関数として、Web サーバに対する接続回数（Y 軸）が表示されます。サーバの接続が再試行されるのは、最初の接続が許可されなかった場合、プロキシ認証が必要な場合、最初の接続がサーバによって閉じられた場合、サーバへの最初の接続を確立できなかった場合、またはサーバが最初にロード・ジェネレータの IP アドレスを変換できなかった場合です。

[接続] グラフ

[接続] グラフには、シナリオ実行中の各時点での（X 軸）、開いている TCP/IP 接続の数（Y 軸）が表示されます。1 つの HTML ページ内に複数の異なる Web アドレスへのリンクが存在する場合、その HTML ページから複数の接続が開かれる可能性があります。各 Web サーバごとに 2 つずつ接続が開かれます。

このグラフは、追加の接続がいつ必要になるかを知るのに役立ちます。たとえば、接続の数が安定状態に達し、トランザクション応答時間が急速に増大している場合には、接続数を増やせば、パフォーマンスが大幅に改善される（トランザクション応答時間が短縮される）でしょう。

[秒ごとの接続数] グラフ

[秒ごとの接続数] グラフには、シナリオ実行中の秒ごとの（X 軸）、新規に開かれた TCP/IP 接続の数（Y 軸）が表示されます。この数は、秒ごとのヒット数よりはるかに小さい数でなければなりません。サーバ、ルータ、およびネットワーク・リソースの消費という観点から見て、新規の TCP/IP 接続は非常に高価だからです。各要求ごとに新しい接続を開く代わりに、多くの HTTP 要求が同じ接続を使用するのが理想的です。

[秒ごとの SSL 接続] グラフ

[秒ごとの SSL 接続] グラフには、シナリオ実行中の秒ごとの（X 軸）、新規または再利用で開かれた SSL 接続の数（Y 軸）が表示されます。SSL 接続は、セキュア・サーバへの TCP/IP 接続が開かれた後で、ブラウザによって開かれます。

新規の SSL 接続を作成すると大量のリソースが消費されるので、新規に開く SSL 接続の数はできるだけ少なくする必要があります。いったん確立した SSL

接続を再利用するべきです。1 つの仮想ユーザにつき複数の新規 SSL 接続を開くべきではありません。[実行環境の設定] メニューの [ブラウザのエミュレーション] タブで、反復ごとに新規の仮想ユーザをシミュレートするように ProTune を設定した場合、それぞれの反復の仮想ユーザごとの新規 SSL 接続は 1 つだけにすべきです。秒ごとの新規 TCP/IP 接続と SSL 接続の数はごく少数にとどめるのが理想です。

第 20 章

システム・リソースの監視

LoadRunner のシステム・リソース・モニタを使って、シナリオの実行中にマシンのシステム・リソースの使用状況を監視できます。

本章では、以下の項目について説明します。

- ▶ Windows リソース・モニタの設定
- ▶ UNIX リソース・モニタの設定
- ▶ UNIX での rstatd デーモンの設定
- ▶ SNMP リソース・モニタの設定
- ▶ Antara FlameThrower モニタの設定
- ▶ SiteScope モニタの設定

システム・リソースの監視について

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、システム・リソースの使用状況です。LoadRunner のリソース・モニタを使って、シナリオ実行中にマシンの Windows, UNIX, SNMP, Antara FlameThrower, および SiteScope リソースを監視して、特定のマシンで発生したボトルネックの原因を判定できます。

Windows の測定値は、Windows の [パフォーマンス モニタ] に組み込まれているカウンタに対応しています。

UNIX の測定項目には、**rstatd** デーモンが対象とする次の項目が含まれます。Average load (平均負荷)、衝突率、コンテキスト切り替え率、CPU Utilization (利用率)、着信パケット・エラー率、着信パケット率、中断率、発信パケット・エラー率、発信パケット率、ページイン率、ページアウト率、Paging rate (ページング率)、スワップイン率、スワップアウト率、システム・モード CPU 使用率、ユーザ・モード CPU 使用率。

注：監視対象のすべての UNIX マシンで、**rstatd** デーモンを設定する必要があります。**rstatd** デーモンの設定方法の詳細については、UNIX の **man** ページまたは 336 ページ「UNIX での rstatd デーモンの設定」を参照してください。

SNMP モニタで、SNMP (Simple Network Management Protocol) を使用しているマシンを監視できます。SNMP の監視は、プラットフォームに依存しません。

Antara FlameThrower モニタは次のパフォーマンス・カウンタを測定できます。Layer, TCP, HTTP, SSL/HTTPS, Sticky SLB, FTP, SMPT, POP3, DNS, および Attacks。

SiteScope モニタは、サーバ、ネットワーク、およびプロセッサのパフォーマンス・カウンタを測定できます。SiteScope モニタが監視できるパフォーマンス・カウンタの詳細については、SiteScope 関連ドキュメントを参照してください。

リソース・モニタは、シナリオの実行時に自動的に有効になります。ただし、監視対象マシンを指定し、マシンごとに監視対象リソースを指定する必要があります。シナリオの実行中にマシンとリソースを追加または削除することもできます。

Windows リソース・モニタの設定

Windows NT および Windows 2000 の測定値は、Windows のパフォーマンス・モニタに組み込まれているカウンタに対応しています。

注：ファイアウォール越しに Windows NT または 2000 マシンを監視するには、139 番ポートで TCP を使用します。

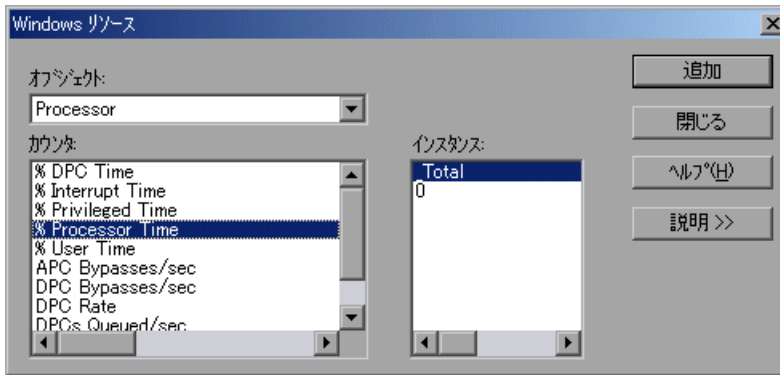
Windows リソース・モニタの設定は、次の手順で行います。

- 1 グラフ・ツリーで Windows リソース・グラフをクリックし、[実行] ビューの右の表示枠までドラッグします。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Windows リソース] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象マシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [Windows リソース] ダイアログ・ボックスの [リソース測定場所] 以下の項目で、監視対象の測定値を選択します。

使用可能なパフォーマンス・カウンタの一覧については、329 ページを参照してください。

注：Windows マシン・モニタの標準設定のカウンタを変更するには、669 ページ「モニタの標準設定のカウンタの変更」を参照してください。Windows 2000 マシンを監視する場合、NT マシンの標準設定のカウンタがいくつか利用できないことがあります（「% Total CPU」や「Interrupts/sec」など）。Windows 2000 に適したカウンタを選択するには、手順 5 に進みます。

- 5 追加の測定値を選択するには、[追加] をクリックします。ダイアログ・ボックスが開き、使用可能な測定値とサーバのプロパティが表示されます。



- 6 オブジェクト、カウンタ、インスタンスを選択します。Ctrl キーを使って、複数のカウンタを選択できます。インスタンスが意味を持つのは、強調表示されたカウンタの複数のインスタンスが実行されている場合だけです。各カウンタの説明を表示するには、[説明>>] をクリックしてダイアログ・ボックスを拡張します。
- 7 [追加] をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。
- 8 [Windows リソース] 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。

注： Windows ドメイン・セキュリティを使っていないリモート Windows マシンを監視するには、リモート Windows マシンでコントローラ・マシンを認証する必要があります。コントローラ・マシンを認証するには、アカウントを作成するか、コントローラへのログオンに使われるアカウントのパスワードを変更して、リモートで監視されている Windows マシンへのログオンに使うパスワードとユーザ名に一致させます。リモート Windows マシンが他のマシンのリソースを要求すると、リソースを要求しているマシンのログイン名とパスワードが送信されます。

Windows リソース・パフォーマンス・カウンタ

Windows マシンの場合，次の標準設定の測定値が利用できます。

対象	測定値	説明
System	% Total Processor Time	システム上のすべてのプロセッサが非アイドル・スレッドを実行するために使用中となる時間の平均的な割合。マルチプロセッサ・システムで，すべてのプロセッサが常にビジー状態ならば，この値は 100% です。すべてのプロセッサが 50% の時間だけビジー状態ならば，この値は 50% です。4 分の 1 のプロセッサが 100% の時間ビジー状態ならば，この値は 25% です。この値は，何らかの処理を行うために費やされた時間の割合です。各プロセッサにはアイドル・プロセス内のアイドル・スレッドが割り当てられます。アイドル・スレッドによって，ほかのスレッドが使用していない非生産的なプロセッサ・サイクルが消費されます。
System	File Data Operations/sec	コンピュータがファイル・システム・デバイスに対する読み書き操作を行う頻度。これには，ファイル制御操作は含まれません。

対象	測定値	説明
Processor	% Processor Time (Windows 2000)	プロセッサが非アイドル・スレッドを実行している時間の割合。このカウンタは、プロセッサの動作状況を示す重要な指標となります。この値は、プロセッサがアイドル・プロセスのスレッドを実行するのに費やす時間をサンプリング間隔ごとに測定し、その値を100%から引くことによって算出されます（各プロセッサには、他のスレッドの実行準備ができていないときにサイクルを消費するアイドル・スレッドが割り当てられています）。この値は、あるサンプリング時点から次のサンプリング時点までの間に何らかの有用な処理を行うために費やされた時間の割合です。このカウンタは、サンプリング間隔の間に観察されたビジー状態の時間の平均な割合を示します。この値は、サービスがアクティブではなかった時間を監視し、その値を100%から引くことによって算出されます。
System	Processor Queue Length	スレッド数で表されるプロセッサ・キューの瞬間的な長さを表します。このカウンタは、スレッド・カウンタの監視も行われていなければ、常に0です。プロセッサはすべて、スレッドがプロセッサ・サイクルを待機する単独のキューを使用します。この長さには、現在実行中のスレッドは含まれません。一般に、プロセッサ・キューが常に2より長い場合、プロセッサが輻輳状態であることを意味します。このカウンタは瞬間的な値を示すもので、一定時間における平均値ではありません。

対象	測定値	説明
メモリ	Page Faults/sec	プロセッサ内で発生したページ・フォルトの回数。ページ・フォルトは、プロセッサが、メイン・メモリ上のワーキング・セットに存在しない仮想メモリ・ページを参照したときに発生します。問題のページがスタンバイ・リスト上に存在する場合（したがってすでにメイン・メモリ中にある場合）、また、そのページを共有している別のプロセスによってそのページが使用されている場合には、ディスクからそのページが取り出されることはありません。
PhysicalDisk	% Disk Time	選択したディスク・ドライブで読み取りまたは書き込みの要求を処理するためにビジー状態となっていた経過時間の割合。
メモリ	Pool Nonpaged Bytes	ページング対象外プールのバイト数。ページング対象外プールはシステム・メモリ領域の1つで、オペレーティング・システムの各コンポーネントが指定されたタスクを実行するとき、この領域に一定の空間を確保します。ページング対象外プールのページは、ページング・ファイルにページ・アウトすることはできません。これらのページは割り当てられている限り、メイン・メモリに存在します。

対象	測定値	説明
メモリ	Pages/sec	参照時にメモリに入っていなかったページへのメモリ参照を解決するために、ディスクから読み取られたページ数またはディスクに書き込まれたページ数。このカウンタは、Pages Input/sec および Pages Output/sec の合計です。このカウンタには、システム・キャッシュに代わってアプリケーションのファイル・データにアクセスするためのページング・トラフィックが含まれます。この値には、キャッシュ対象外マップ済みメモリ・ファイルとメモリの間で読み書きされるページも含まれます。メモリが過度に使用される点（つまり、スラッシング）、およびその結果生じる可能性のある過剰なページングが気になる場合には、このカウンタを観察することが重要になります。
System	Total Interrupts/sec	コンピュータがハードウェア割り込みを受信して処理する頻度。割り込みを生成する可能性があるデバイスとしては、システム・タイマ、マウス、データ通信回線、ネットワーク・インタフェース・カードなどの周辺機器があります。このカウンタにより、これらのデバイスがコンピュータ全体から見てどの程度使用されているのか確認できます。「Processor: Interrupts/sec」も参照してください。
Objects	Threads	データ収集時のコンピュータのスレッド数。このカウンタは瞬間的な値を示すもので、一定時間における平均値ではありません。スレッドとは、プロセッサで命令を実行できる、基本的な実行単位です。
Process	Private Bytes	プロセスによって割り当てられ、ほかのプロセスとは共有できないバイト数の最新の値。

UNIX リソース・モニタの設定

UNIX カーネルの統計情報の測定値には、**rstatd** デーモンが扱う次の項目が含まれます。平均負荷、衝突率、コンテキスト切り替え率、CPU 利用率、着信パケット・エラー率、着信パケット率、中断率、発信パケット・エラー率、発信パケット率、ページイン率、ページアウト率、ページング率、スワップイン率、スワップアウト率、システム・モード CPU 使用率、ユーザ・モード CPU 使用率。

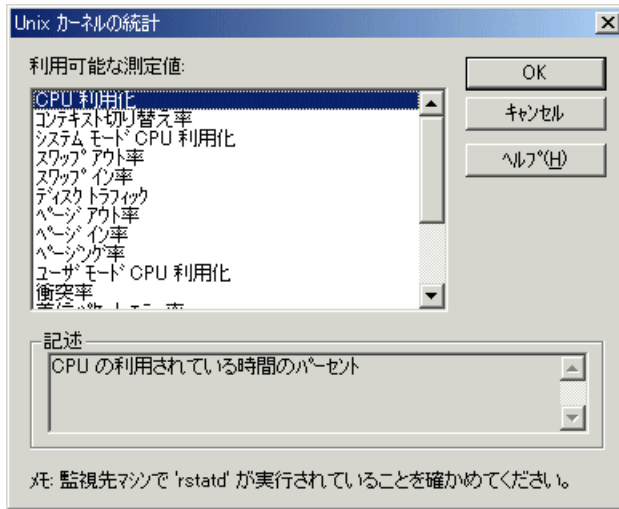
UNIX リソース・モニタの設定は、次の手順で行います。

- 1 グラフ・ツリーで UNIX リソース・グラフをクリックし、[実行] ビューの右の表示枠までドラッグします。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [UNIX リソース] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象マシンのサーバ名または IP アドレスを入力します。プラットフォーム・リストから [UNIX] を選択して、[OK] をクリックします。
- 4 [UNIX リソース] ダイアログ・ボックスの [リソース測定場所] 項目で、監視対象の標準設定の測定値を選択します。

使用可能なパフォーマンス・カウンタの一覧については、335 ページを参照してください。

注：UNIX モニタの標準設定のカウンタを変更する方法については、669 ページ「モニタの標準設定のカウンタの変更」を参照してください。

- 追加の測定値を選択するには、**[追加]** をクリックします。**[Unix カーネルの統計]** ダイアログ・ボックスが表示され、利用可能な測定値およびサーバのプロパティが表示されます。



- モニタ・リストに UNIX 測定値を追加するには、必要な測定値を選択して **[OK]** をクリックします。
- [UNIX カーネルの統計]** ダイアログ・ボックスで **[OK]** をクリックすると、UNIX モニタが起動します。

注： 監視対象の UNIX マシンで `rstatd` デーモンが適切に設定され、実行されていること確認します。詳細については、336 ページ「UNIX での `rstatd` デーモンの設定」を参照してください。

UNIX リソース・パフォーマンス・カウンタ

UNIX マシンには、次の標準の測定値が使用できます。

測定値	説明
平均負荷	直前の 1 分間に同時に「READY」状態であったプロセスの平均数。
衝突率	Ethernet で検出された秒ごとの衝突回数。
コンテキスト切り替え率	プロセス間またはスレッド間の秒ごとの切り替え回数。
CPU 利用率	CPU が使用された時間の割合。
ディスク・トラフィック	ディスク転送速度。
着信パケット・エラー率	Ethernet パケット受信中の秒ごとのエラー数。
着信パケット率	秒ごとの受信 Ethernet パケット数。
中断率	秒ごとのデバイスの割り込み回数。
発信パケット・エラー率	Ethernet パケット送信中の秒ごとのエラー数。
発信パケット率	秒ごとの送信 Ethernet パケット数。
ページイン率	物理メモリに読み込まれた秒ごとのページ数。
ページアウト率	ページファイルに書き込まれた、または、物理メモリから削除された秒ごとのページ数。
ページング率	物理メモリに読み込まれた、または、ページファイルに書き込まれた秒ごとのページ数。
スワップイン率	スワップインされたプロセス数。
スワップアウト率	スワップアウトされたプロセス数。
システム・モード CPU 利用化	CPU がシステム・モードで使用された時間の割合。
ユーザ・モード CPU 利用化	CPU がユーザ・モードで使用された時間の割合。

UNIX での rstatd デーモンの設定

UNIX リソースを監視するには、rstatd デーモンを設定する必要があります。マシンが rstatd リクエストを受信すると、マシン上の inetd が自動的に rstatd を起動するため、すでに起動されている場合があります。

rstatd デーモンが設定されているかどうかの確認は、次の手順で行います。

rup コマンドは、rstatd の設定を含む、様々なマシンの統計情報を報告します。マシンの統計情報を表示するには、次のコマンドを実行します。

```
>rup host
```

また、**lr_host_monitor** を使って、何らかの統計値が返されるかどうかを確認することもできます。

このコマンドによって何らかの統計値が返された場合には、rstatd デーモンが設定され、起動されています。統計値が返されなかった場合またはエラー・メッセージを受信した場合は、rstatd デーモンが設定されていません。

rstatd デーモンの設定は、次の手順で行います。

- 1 **su root** コマンドを実行します。
- 2 **/etc/inetd.conf** を開いて、(rstatd で始まる) rstatd 行を見つけます。コメントアウトされている (行頭に「#」が付いている) 場合は、このコメント指示子を削除して、ファイルを保存します。
- 3 コマンド・ラインから、次のコマンドを実行します。

```
kill -1 inet_pid
```

ここで **inet_pid** は、inetd プロセスのプロセス ID です。このコマンドを実行すると、inetd によって **/etc/inetd.conf** ファイルが再スキャンされ、コメントアウトされていないすべてのデーモン (rstatd デーモンを含む) が登録されます。

- 4 再度 **rup** を実行します。

ここでも rstatd が設定されていないことが示された場合は、システム管理者に連絡してください。

注：ファイアウォール越しに UNIX マシンを監視するには、UNIX ユーティリティ、`rpcinfo` を実行して、`rstatd` のポート番号を特定する必要があります。`rpcinfo -p <ホスト名>` を実行すると、ポート番号と一緒にホストの `portmapper` に登録されているすべての RPC サーバのリストが返されます。このリストは `rstatd` を停止して再実行しないと変更されません。ファイアウォールには、ポートではなく、RPC プログラム番号を開けるものもあります。この場合、プログラム 100001 を開きます。バージョン番号を指定するように指示された場合は、バージョン 3 と 4 を指定します。

SNMP リソース・モニタの設定

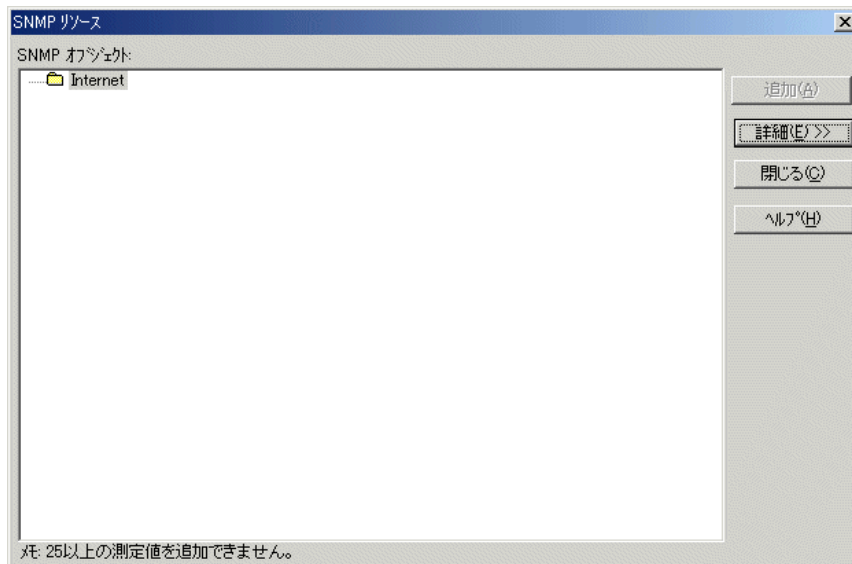
SNMP リソース・モニタで、SNMP (Simple Network Management Protocol) を使用して SNMP エージェントを実行する任意のマシンを監視できます。

注：`snmp.cfg` ファイルにポート番号を指定できます。ポートを指定しないと、LoadRunner は、SNMP の標準設定の 161 番ポートに接続します。また、次の形式でマシン名の指定もできます。`<サーバ名> : <ポート番号>` ファイアウォール越しに SNMP リソースを監視するには、161 番または 162 番ポートを使います。

SNMP リソース・モニタの設定は、次の手順で行います。

- 1 グラフ・ツリーで SNMP リソース・グラフをクリックし、[実行] ビューの右の表示枠までドラッグします。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [SNMP] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象マシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [SNMP] ダイアログ・ボックスの [リソース測定場所] 項目で [追加] をクリックして、監視対象の測定値を選択します。

[SNMP リソース] ダイアログ・ボックスが表示されます。



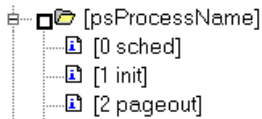
- 5 SNMP オブジェクト・ツリーを参照します。
- 6 オブジェクトを測定するには、対象のオブジェクトを選択して [追加] をクリックします。各リソースの説明を表示するには、[詳細 >>] をクリックしてダイアログ・ボックスを拡張します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。

注：SNMP モニタは、測定値を 25 項目まで監視できます。

- 7 [SNMP] 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。

監視するリソースのリストは、シナリオ実行中に随時変更できます。リモート・マシン上のリソースを監視するときに、シナリオがアクティブである必要はありません。

注：SNMP モニタの測定値情報レベルを向上させるには、測定値のリストの文字列値（測定値の数値に追加）と、名前修飾子（測定値名の固有部として表示される文字列値）を有効にします。名前修飾子を用いた測定値の次の例では、ProcessName の文字列値（sched）が、そのインスタンス ID（0）に加えて表示されます。



この機能を有効にするには、次の行を < **LoadRunner** のインストール先フォルダ > %dat%monitors%snmp.cfg ファイルに追加します。

SNMP_show_string_nodes=1 使用上の注意：名前修飾子は複数選択できますが、使用されるのは、階層の先頭の修飾子です。[SNMP] ダイアログ・ボックスが表示されるたびに、snmp.cfg ファイルから情報が再び読み取られます。「名前修飾子付き」と「名前修飾子なし」という具合に同じ測定項目を 2 度追加することはできません。2 度追加すると、エラー・メッセージが表示されます。

Antara FlameThrower モニタの設定

[Antara FlameThrower 設定] ダイアログ・ボックスを使って、Antara FlameThrower サーバを監視するための測定値を選択します。

Antara FlameThrower モニタの設定は、次の手順で行います。

- 1 グラフ・ツリーで Antara FlameThrower グラフをクリックして、[実行] ビューの右の表示枠までドラッグします。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Antara FlameThrower] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。次の形式に従って、サーバ名または IP アドレスを入力します。
<サーバ名> : <ポート番号>

次に例を示します。merc1:12135

マシンを実行するプラットフォームを選択して、[OK] をクリックします。

- 4 [Antara FlameThrower] ダイアログ・ボックスの [リソース測定場所] 項目の [追加] をクリックして、監視対象の測定値を選択します。[Antara FlameThrower 設定] ダイアログ・ボックスが表示されます。
- 5 [測定コンポーネント] ツリーが表示されます。
- 6 [Antara FlameThrower 設定] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。

使用可能なパフォーマンス・カウンタの一覧については、340 ページを参照してください。
- 7 [Antara FlameThrower モニタ設定] ダイアログ・ボックスと [Antara FlameThrower] ダイアログ・ボックスで [OK] をクリックすると、Antara FlameThrower モニタが起動します。

Antara FlameThrower パフォーマンス・カウンタ

以下の表に、監視可能なカウンタの説明を示します。

レイヤ・パフォーマンス・カウンタ

測定値	説明
TxBytes	送信された Layer 2 データ・バイトの総数。
TxByteRate (/sec)	送信された Layer 2 データ・バイトの秒ごとの総数。
TxFrames	送信されたパケットの総数。
TxFrameRate (/sec)	送信されたパケットの秒ごとの数。
RxBytes	受信された Layer 2 データ・バイトの総数。
RxByteRate (/sec)	受信された Layer 2 データ・バイトの秒ごとの数。
RxFrames	受信されたパケットの総数。
RxFrameRate (/sec)	受信されたパケットの秒ごとの数。

TCP パフォーマンス・カウンタ

測定値	説明
ActiveTCPConns	現在アクティブになっている TCP 接続の総数。
SuccTCPConns	受信された SYN ACK パケットの総数。
SuccTCPConnRate(/sec)	受信された SYN ACK パケットの秒ごとの数。
TCPConnLatency (milisec)	SYN パケットの送信から SYN ACK 応答パケットの受信までの間隔 (ミリ秒)。
MinTCPConnLatency(milisecc)	最小 TCPConnectionLatency (ミリ秒)。
MaxTCPConnLatency(milisecc)	最大 TCPConnectionLatency (ミリ秒)。
TCPsndConnClose	送信された FIN または FIN ACK パケットの総数 (クライアント)。
TCPrcvConnClose	受信された FIN または FIN ACK パケットの総数 (クライアント)。
TCPsndResets	送信した RST パケットの総数。
TCPrcvResets	受信された RST パケットの総数。

測定値	説明
SYNSent	送信された SYN パケットの総数。
SYNSentRate (/sec)	送信された SYN パケットの秒ごとの数。
SYNAckSent	送信された SYN ACK パケットの総数。
SYNAckRate (/sec)	送信された SYN ACK パケットの秒ごとの数。

HTTP パフォーマンス・カウンタ

測定値	説明
HTTPRequests	送信された HTTP 要求コマンド・パケットの総数。
HTTPRequestRate(/sec)	送信された HTTP 要求パケットの秒ごとの数。
AvgHTTPDataLatency(milisecs)	過去 1 秒の平均 HTTP データ遅延 (ミリ秒)。
HTTPDataLatency(milisecs)	要求パケットの送信から応答の受信までの間隔 (ミリ秒)。
DataThroughput(bytes/sec)	HTTP サーバから受信したデータ・バイトの秒ごとの数。
MinHTTPDataLatency(milisecs)	最小 HTTPDataLatency (ミリ秒)。
MaxHTTPDataLatency(milisecs)	最大 HTTPDataLatency (ミリ秒)。
MinDataThroughput(bytes/sec)	最小 HTTPDataThroughput (バイト / 秒)。
MaxDataThroughput(bytes/sec)	最大 HTTPDataThroughput (バイト / 秒)。
SuccHTTPRequests	受信された成功 HTTP 要求応答 (200 OK) の総数。
SuccHTTPRequestRate(/sec)	受信された成功 HTTP 要求応答 (200 OK) の秒ごとの数。
UnsuccHTTPRequests	失敗した HTTP 要求の数。

SSL/HTTPS パフォーマンス・カウンタ

測定値	説明
SSLConnections	クライアントによって送信された ClientHello メッセージの数。
SSLConnectionRate(/sec)	送信された ClientHello メッセージの秒ごとの数。
SuccSSLConnections	成功した SSL 接続の数。成功した接続とは、サーバからのハンドシェイク完了メッセージがクライアントによってエラーなく受信された接続です。
SuccSSLConnectionRate(/sec)	成功した SSL 接続の秒ごとの数。
SSLAlertErrors	クライアントによって受信された SSL 警告メッセージの数（例：bad_record_mac, decryption_failed, handshake_failure など）。
SuccSSLResumedSessions	正しく再開された SSL セッションの数。
FailedSSLResumedSessions	再開できなかった SSL セッションの数。

Sticky SLB パフォーマンス・カウンタ

測定値	説明
CookieAuthenticationFail	サーバが認証しなかったクッキーの数。
SuccCookieAuthentication	サーバが認証したクッキーの数。
SSLClientHellos	サーバに送信された Client Hello パケットの数。
SSLServerHellos	サーバからクライアントに返送された Server Hello パケットの数。
SSLSessionsFailed	サーバが認証しなかった Session ID の数。
SSLSessionsResumed	サーバが認証した Session ID の数。
succSSLClientHellos	クライアントによって受信された Client Hello リプライ、またはサーバによって受信されたパケットの数。
succSSLServerHellos	クライアントによって受信された Server Hello の数。

FTP パフォーマンス・カウンタ

測定値	説明
TPUsers	送信された FTP User コマンド・パケットの総数。
FTPUserRate (/sec)	送信された FTP User コマンド・パケットの秒ごとの数。
FTPUserLatency(miliseconds)	FTP User パケットの送信から応答の受信までの間隔 (ミリ秒)。
MinFTPUserLatency(miliseconds)	最小 FTP ユーザ待ち時間 (ミリ秒)。
MaxFTPUserLatency(miliseconds)	最大 FTP ユーザ待ち時間 (ミリ秒)。
SuccFTPUsers	受信された成功 FTP User コマンド応答の総数。
SuccFTPUserRate(/sec)	受信された成功 FTP User コマンド応答の秒ごとの数。
FTPPasses	送信された FTP PASS パケットの総数。
FTPPassRate (/sec)	送信された FTP PASS パケットの秒ごとの数。

測定値	説明
FTPPassLatency(milise cs)	FTP PASS パケットの送信から応答の受信までの間隔 (ミリ秒)。
MinFTPPassLatency(milise cs)	最小 FTPPassLatency (ミリ秒)。
MaxFTPPassLatency(milise cs)	最大 FTPPassLatency (ミリ秒)。
SuccFTPPasses	受信された成功 FTP PASS 応答の総数。
SuccFTPPassRate(/sec)	受信された FTP PASS 応答の秒ごとの数。
FTPControlConnections	FTP クライアントによって送信された SYN パケットの総数。
FTPControlConnectionRate(/sec)	FTP クライアントによって送信された SYN パケットの秒ごとの数。
SuccFTPControlConnections	FTP クライアントによって受信された SYN ACK パケットの総数。
SuccFTPControlConnectionRate(/sec)	FTP クライアントによって受信された SYN ACK パケットの秒ごとの数。
FTPDataConnections	FTP クライアントによって受信された SYN ACK パケットの秒ごとの数。
FTPDataConnectionRate(/sec)	FTP クライアントによって送信された、または FTP サーバによって受信された SYN ACK パケットの秒ごとの数。
SuccFTPDataConnections	FTP クライアントによって送信された、または FTP サーバによって受信された SYN ACK パケットの総数。
SuccFTPDataConnectionRate(/sec)	FTP サーバによって受信された SYN ACK パケットの秒ごとの数。
FtpAuthFailed	FTP クライアントによって受信されたエラー応答の総数。
FTPGets	クライアントの Get 要求の総数。
FTPPuts	クライアントの Put 要求の総数。

測定値	説明
SuccFTPGets	成功した（データがサーバからクライアントに正しく送信された）Get 要求の総数。
SuccFTPPuts	成功した（データがクライアントからサーバに正しく送信された）Put 要求の総数。

SMTP パフォーマンス・カウンタ

測定値	説明
SMTPHelos	送信された HELO パケットの総数。
SMTPHeloRate (/sec)	送信された HELO パケットの秒ごとの数。
SMTPHeloLatency(miliseecs)	HELO パケットの送信から応答の受信までの間隔（ミリ秒）。
MinSMTPHeloLatency(miliseecs)	最小 SMTPHeloLatency（ミリ秒）。
MaxSMTPHeloLatency(miliseecs)	最大 SMTPHeloLatency（ミリ秒）。
SuccSMTPHelos	成功した HELO 応答の受信総数。
SuccSMTPHeloRate(/sec)	受信された HELO リプライの秒ごとの数。
SMTPMailFroms	送信された Mail From パケットの総数。
SMTPMailFromRate (/sec)	送信された Mail From パケットの秒ごとの数。
SMTPMailFromLatency(miliseecs)	Mail From パケットの送信から応答の受信までの間隔（ミリ秒）。
MinSMTPMailFromLatency(miliseecs)	最小 SMTPMailFromLatency（ミリ秒）。
MaxSMTPMailFromLatency(miliseecs)	最大 SMTPMailFromLatency（ミリ秒）。
SuccSMTPMailFroms	受信された成功 Mail From 応答の総数。
SuccSMTPMailFromRate (/sec)	受信された成功 Mail From 応答の秒ごとの数。

測定値	説明
SMTPRcptTos	送信された RcptTo パケットの総数。
SMTPRcptToRate(/sec)	送信された RcptTo パケットの秒ごとの数。
SMTPRcptToLatency(miliseecs)	RcptTo パケットの送信から応答の受信までの間隔 (ミリ秒)。
MinSMTPRcptToLatency(miliseecs)	最小 SMTPRcptToLatency (ミリ秒)。
MaxSMTPRcptToLatency(miliseecs)	最大 SMTPRcptToLatency (ミリ秒)。
SuccSMTPRcptTos	受信された成功 RcptTo 応答の総数。
SuccSMTPRcptToRate(/sec)	受信された成功 RcptTo 応答の秒ごとの数。
SMTPDdatas	送信された Data パケットの総数。
SMTPDataRate (/sec)	送信された Data パケットの秒ごとの数。
SMTPDataLatency(miliseecs)	Data パケットの送信から応答の受信までの間隔 (ミリ秒)。
MinSMTPDataLatency(miliseecs)	最小 SMTPDataLatency (ミリ秒)。
MaxSMTPDataLatency(miliseecs)	最大 SMTPDataLatency (ミリ秒)。
SuccSMTPDdatas	受信された成功 Data 応答の総数。
SuccSMTPDataRate(/sec)	受信された成功 Data 応答の秒ごとの数。

POP3 パフォーマンス・カウンタ

測定値	説明
POP3Users	送信された Pop3 User コマンド・パケットの総数。
POP3UserRate (/sec)	送信された Pop3 User コマンド・パケットの秒ごとの数。
POP3UserLatency(miliseecs)	Pop3 User コマンド・パケットの送信から応答の受信までの間隔 (ミリ秒)。
MinPOP3UserLatency(miliseecs)	最小 POP3UserLatency (ミリ秒)。

測定値	説明
MaxPOP3UserLatency(milisecs)	最大 POP3UserLatency (ミリ秒)。
SuccPOP3Users	受信された成功 Pop3 User 応答の総数。
SuccPOP3UserRate(/sec)	受信された成功 Pop3 User 応答の秒ごとの数。
POP3Passes	送信された Pop3 Pass コマンド・パケットの総数。
POP3PassRate (/sec)	送信された Pop3 Pass コマンド・パケットの秒ごとの数。
POP3PassLatency(milisecs)	Pop3 Pass パケットの送信から応答の受信までの間隔 (ミリ秒)。
MinPOP3PassLatency(milisecs)	最小 POP3PassLatency (ミリ秒)。
MaxPOP3PassLatency(milisecs)	最大 POP3PassLatency (ミリ秒)。
SuccPOP3Passes	受信された成功 Pop3 Pass 応答の総数。
SuccPOP3PassRate(/sec)	受信された成功 Pop3 Pass 応答の秒ごとの数。
POP3Stats	送信された Pop3 Stat コマンド・パケットの総数。
POP3StatRate (/sec)	送信された Pop3 Stat コマンド・パケットの秒ごとの数。
POP3StatLatency(milisecs)	Pop3 Stat パケットの送信から応答の受信までの間隔 (ミリ秒)。
MinPOP3StatLatency(milisecs)	最小 POP3StatLatency (ミリ秒)。
MaxPOP3StatLatency(milisecs)	最大 POP3StatLatency (ミリ秒)。
SuccPOP3Stats	受信された成功 Pop3 Stat 応答の総数。
SuccPOP3StatRate(/sec)	受信された成功 Pop3 Stat 応答の秒ごとの数。
POP3Lists	送信された Pop3 List コマンド・パケットの総数。
POP3ListRate (/sec)	送信された Pop3 List コマンド・パケットの秒ごとの数。
POP3ListLatency(milisecs)	Pop3 List パケットの送信から応答の受信までの間隔 (ミリ秒)。

測定値	説明
MinPOP3ListLatency(mili secs)	最小 POP3ListLatency (ミリ秒)。
MaxPOP3ListLatency(mili secs)	最大 POP3ListLatency (ミリ秒)。
SuccPOP3Lists	受信された成功 Pop3Lists の総数。
SuccPOP3ListRate(/sec)	受信された成功 Pop3Lists の秒ごとの数。
POP3Retrs	送信された Pop3 Retr パケットの総数。
POP3RetrRate (/sec)	送信された Pop3 Retr パケットの秒ごとの数。
POP3RetrLatency(miliseconds)	Pop3 Retr パケットの送信から応答の受信までの間隔 (ミリ秒)。
MinPOP3RetrLatency(mili secs)	最小 POP3RetrLatency (ミリ秒)。
MaxPOP3RetrLatency(mil isecs)	最大 POP3RetrLatency (ミリ秒)。
SuccPOP3Retrs	受信された成功 Pop3Retrs の総数。
SuccPOP3RetrRate(/sec)	受信された成功 Pop3Retrs の秒ごとの数。

DNS パフォーマンス・カウンタ

測定値	説明
SuccPrimaryDNSRequest	プライマリ DNS サーバに行われた Successful DNS 要求の総数。
SuccSecondaryDNSRequest	セカンダリ DNS サーバに行われた Successful DNS 要求の総数。
SuccDNSDataRequestRate (/sec)	送信された Successful DNS 要求パケットの秒ごとの数。
PrimaryDNSFailure	プライマリ DNS サーバから受信した DNS 要求の失敗の総数。
PrimaryDNSRequest	プライマリ DNS サーバに対して行われた DNS 要求の総数。

測定値	説明
SecondaryDNSFailure	セカンダリ DNS サーバから受信した DNS 要求の失敗の総数。
SecondaryDNSRequest	セカンダリ DNS サーバに対して行われた DNS 要求の総数。
MinDNSDataLatency	最小 DNS データ待ち時間 (ミリ秒)。
MaxDNSDataLatency	最大 DNS データ待ち時間 (ミリ秒)。
CurDNSDataLatency	DNS 要求の送信から応答の受信までの間隔 (ミリ秒)。
DNSDataRequestRate(/sec)	送信された DNS 要求パケットの秒ごとの数。
NoOfReTransmission	再送信された DNS 要求パケットの総数。
NoOfAnswers	DNS 要求パケットへの回答の総数。

Attacks パフォーマンス・カウンタ

測定値	説明
Attacks	送信された攻撃パケットの総数 (すべての攻撃)
AttackRate (/sec)	送信された攻撃パケットの秒ごとの数 (ARP, Land, Ping, SYN, および Smurf)。
Havoc Flood	生成された Havoc パケットの数 (Stacheldraht のみ)。
Icmp Flood	生成された ICMP 攻撃パケットの数 (TFN, TFN2K, および Stacheldraht)。
Mix Flood	生成された Mix パケットの数 (TFN2K のみ)。
Mstream Flood	生成された Mstream パケットの数 (Stacheldraht のみ)。
Null Flood	生成された Null パケットの数 (Stacheldraht のみ)。
Smurf Flood	生成された Smurf 攻撃パケットの数 (TFN, TFN2K, および Stacheldraht)。
Syn Flood	生成された SYN 攻撃パケットの数 (TFN, TFN2K, および Stacheldraht)。
Targa Flood	生成された Targa パケットの数 (TFN2K のみ)。
Udp Flood	生成された UDP パケットの数 (すべての DDoS 攻撃のみ)。

SiteScope モニタの設定

[SiteScope Monitor 設定] ダイアログ・ボックスを使って、SiteScope からポーリングする測定値を選択します。

SiteScope を設定する前に、次のことを行います。

- 1 SiteScope がサーバにインストールされていることを確認します。SiteScope はコントローラ・マシンにインストールできますが、専用のサーバにインストールすることをお勧めします。
- 2 SiteScope がインストールされているマシンで、SiteScope が必要なサーバを監視するように構成します。SiteScope サーバの設定の詳細については、『SiteScope ユーザーズ・ガイド』を参照してください。

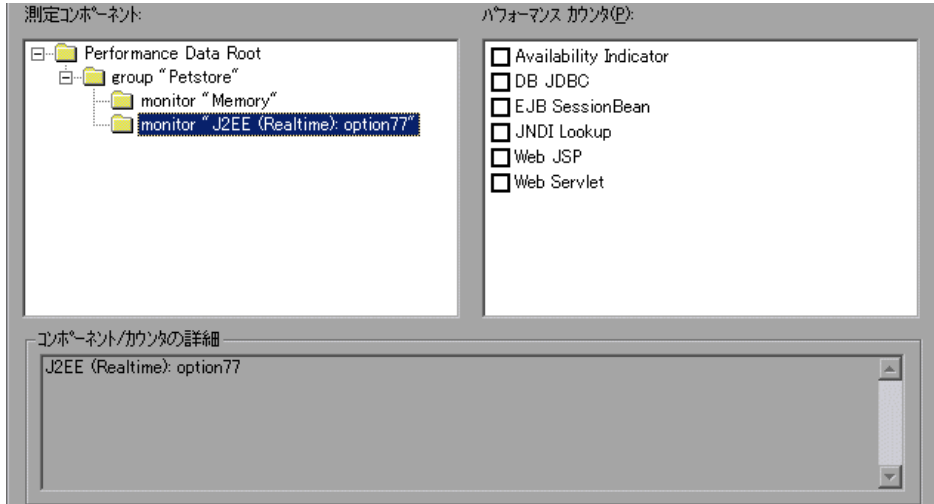
注：モニタに名前を割り当てるときは、モニタ名にサーバ名を含めます。そうすることにより、モニタが属するホストについての混乱を避けられます。

- 3 SiteScope が監視対象のサーバから必要なデータを収集していることを確認します。[SiteScope Panel] で、サーバ・マシンに対してポーリングを行うモニタ・グループを選択し、そのモニタの [Status] カラムにサーバ測定値のリストが表示されていることを確認します。

SiteScope モニタの設定は、次の手順で行います。

- 1 グラフ・ツリーで SiteScope グラフをクリックし、[実行] ビューの右の表示枠までドラッグします。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [SiteScope] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

- 4 [SiteScope] ダイアログ・ボックスの [リソース測定場所] 項目で [追加] をクリックして、監視対象の測定値を選択します。[SiteScope Monitor 設定] ダイアログ・ボックスに利用可能な測定値が表示されます。
- 5 [測定コンポーネント] ツリーを目的のコンポーネントまでたどっていきます。



- 6 [SiteScope Monitor 設定] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。
- 7 [SiteScope Monitor 設定] ダイアログ・ボックスと [SiteScope] ダイアログ・ボックスで [OK] をクリックすると、SiteScope モニタが起動します。

注： SiteScope の標準サンプリング頻度は 10 分で、最小頻度は 15 秒です。SiteScope は 1 度に 1 つのコントローラでのみ監視できます。

第 21 章

ネットワークの監視

ネットワーク・モニタを使用して、ネットワークがシナリオ内で遅延を引き起こしているかどうかを判定できます。また、問題が生じているネットワーク・セグメントを特定することもできます。

注：ネットワーク・モニタを実行するには、監視を実行する Windows マシンに対する管理者権限が必要です（ICMP プロトコルを使用する場合を除く）。

本章では、以下の項目について説明します。

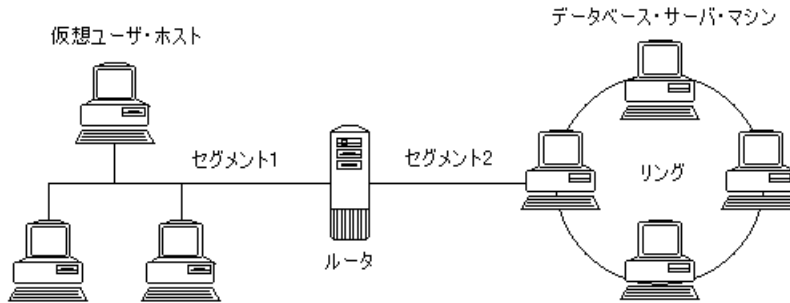
- ▶ UNIX 送信元マシンからのネットワークの監視
- ▶ ネットワーク遅延時間モニタの設定
- ▶ ネットワーク遅延時間グラフの表示

ネットワークの監視について

アプリケーションのパフォーマンスにおいて、ネットワークの設定は非常に大切です。設計が適切でないと、クライアントの動作速度が許容可能なレベルを下回ることがあります。

実際の Web システムまたはクライアント/サーバ・システムには、多数のネットワーク・セグメントがあります。パフォーマンスの悪い 1 つのネットワーク・セグメントが、システム全体に影響を与えることもあります。

次の図は、一般的なネットワークを示しています。データは、サーバ・マシンから仮想ユーザ・マシンに到達するまでに、複数のセグメントを経由します。



ネットワーク・モニタは、ネットワークのパフォーマンスを測定するために、ネットワークを経由してデータの packets を送信します。パケットが返ると、ネットワーク・モニタは、そのパケットが、要求されたノードに行って戻ってくるのにかかった時間を算出します。この時間がネットワーク遅延時間グラフに遅延として表示されます。

オンラインのネットワーク遅延時間グラフを使って、ネットワーク関連の問題を特定して、対処できます。

注：送信元マシンから各ノードまでの間の遅延は、同時ながら個別に測定されます。したがって、送信元マシンからノードの1つまでの遅延が、送信元マシンから送信先マシンまでのパスの全体の遅延より大きくなる場合があります。

UNIX 送信元マシンからのネットワークの監視

UNIX マシンで、UDP または ICMP プロトコルを使用してネットワークの監視を実行できます。UNIX 送信元マシンからネットワークの監視を実行する前に、次の手順を行います。

- ▶ root アクセス権限を **merc_webtrace** プロセスに割り当てることで、送信元マシンを設定します。
- ▶ rsh またはエージェントのいずれかを介して送信元マシンに接続するための調整を必要に応じて行います。

送信元マシンの設定

ローカルに **LoadRunner** がインストールされている送信元マシンを設定するには、次の手順で行います。

root アクセス権限を **merc_webtrace** プロセスに割り当てるには、次のように、**merc_webtrace** のアクセス権限に s ビットを追加します。

- 1 送信元マシンに root としてログインします。
- 2 `cd < LoadRunner のインストール先フォルダ > bin` と入力し、**bin** ディレクトリに移ります。
- 3 `chown root merc_webtrace` と入力し、**merc_webtrace** ファイルの所有者を root に変更します。
- 4 `chmod +s merc_webtrace` と入力して、ファイルのアクセス権限に s ビットを追加します。
- 5 確認のため、`ls -l merc_webtrace` と入力します。アクセス権限は、`-rwsrwsr-x` のようになっているはずです。

ネットワーク上に **LoadRunner** がインストールされている、送信元マシンを設定するには、次の手順で行います。

LoadRunner ネットワーク・インストールでは、**merc_webtrace** プロセスは、送信元マシンのディスク上ではなく、ネットワーク上にあります。以下の手順では、**merc_webtrace** ファイルをローカル・ディスクにコピーし、**mdrv.dat** を設定してプロセスが認識されるようにし、root アクセス権限を **merc_webtrace** に割り当てます。

- 1 **merc_webtrace** を < **LoadRunner** のインストール先フォルダ > /bin から送信元マシンのローカル・ディスクの任意の場所にコピーします。たとえば、ファイルを /local/ < LR <ディレクトリ> にコピーするには、**cp /net/tools/ < LoadRunner のインストール先フォルダ > /bin/merc_webtrace /local/LR** と入力します。

注：同じネットワーク・インストールを使う送信元マシンは、すべて、同じ **mdrv.dat** を使用するため、**merc_webtrace** を各ローカル・ディスクの同じディレクトリ・パス（たとえば、/local/ < LR >）にコピーする必要があります。

- 2 次の行を、< **LoadRunner** のインストール先フォルダ > /dat/**mdrv.dat** ファイルの [monitors_server] セクションに追加します。

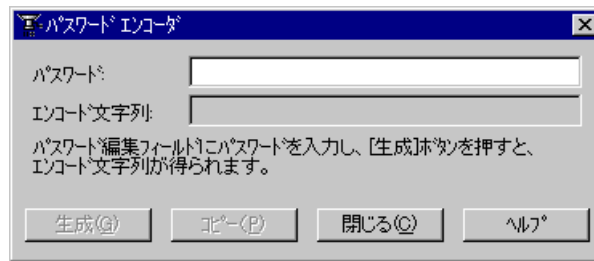
```
ExtCmdLine=-merc_webtrace_path/local/xxx
```
- 3 送信元マシンに root としてログインします。
- 4 **cd** < **LoadRunner** のインストール先フォルダ > /bin と入力し、**bin** ディレクトリに移ります。
- 5 **chown root merc_webtrace** と入力し、**merc_webtrace** ファイルの所有者を root に変更します。
- 6 **chmod +s merc_webtrace** と入力して、ファイルのアクセス権限に s ビットを追加します。
- 7 確認のため、**ls -l merc_webtrace** と入力します。アクセス権限は、-rwsrwsr-x のようになっているはずです。

rsh を介した送信元マシンへの接続

コントローラが、rsh を介して送信元マシンに接続されている場合（デフォルトの接続モード）、エージェント・デーモンを起動する必要はありません。ネットワーク・モニタを初めて実行するときは、実行に先立って、暗号化されたユーザ名およびパスワードをネットワーク・モニタ設定ファイルに入力します。

暗号化されたユーザ名およびパスワードを作成するには、次の手順で行います。

- 1 Windows のタスクバーで [スタート] をクリックし、[プログラム] > [LoadRunner] > [ツール] をポイントして、[Password Encoder] をクリックします。[パスワードエンコーダ] ウィンドウが開きます。



- 2 [パスワード] ボックスに、rsh ユーザ名とパスワードを垂直バー記号で区切って入力します。たとえば、「myname|mypw」のように入力します。
- 3 [生成] をクリックします。暗号化された文字列が [エンコード文字列] フィールドに表示されます。
- 4 [コピー] をクリックして、暗号化された文字列をクリップボードにコピーします。
- 5 次の行を、< LoadRunner のインストール先フォルダ > /dat/monitors/ndm.cfg ファイルの [hosts] セクションに追加します。
Host = <クリップボードからコピーした暗号化された文字列>
- 6 現在のシナリオを閉じて、開きます。LoadRunner が更新された構成ファイルを読み込み、送信元マシンを監視用として認識します。

エージェントを介した送信元マシンへの接続

コントローラが rsh を介して送信元マシンに接続されていない場合、ネットワーク・モニタを実行する前に、エージェント・デーモンが送信元マシンで動作していることを確認してください。rsh を使用しない方法の詳細については、付録 D 「コントローラのトラブルシューティング」の「UNIX シェル」という項を参照してください。

エージェント・デーモンを起動するには、次の手順で行います。

rsh を使用しない場合には、送信元マシンでエージェント・デーモンを起動します。

- 1 < LoadRunner のインストール先フォルダ > /bin ディレクトリで `m_daemon_setup -install` と入力します。
- 2 ネットワーク・モニタを起動するとき、必ずエージェント・デーモンが動作していることを確認してください。
- 3 ネットワーク遅延モニタのエージェント・デーモンを停止するには、`m_daemon_setup -remove` と入力します。

ネットワーク遅延時間モニタの設定

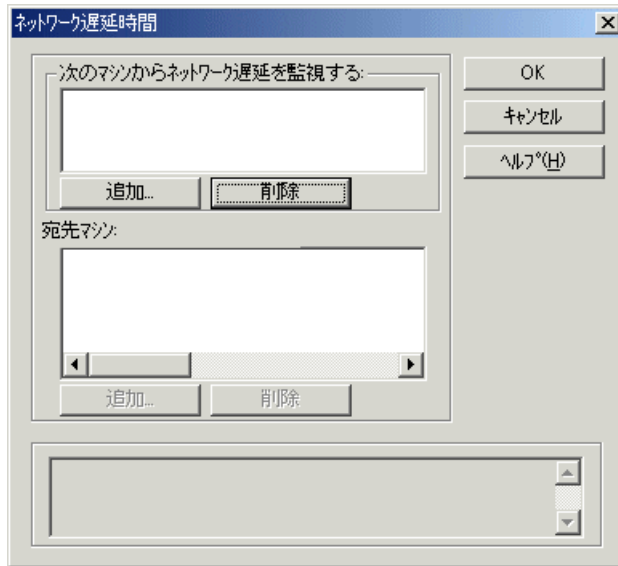
ネットワーク・モニタは、シナリオを実行する前に、コントローラの [実行] ビューで設定します。[ネットワーク遅延時間] および [ネットワーク遅延監視先マシンの追加] ダイアログ・ボックスを使って、監視対象のネットワーク・パスを選択します。

注：ネットワークを監視するには、送信元マシンに LoadRunner エージェントをインストールする必要があります。監視対象マシンに LoadRunner エージェントをインストールする必要はありません。

ネットワーク・モニタを設定するには、次の手順で行います。

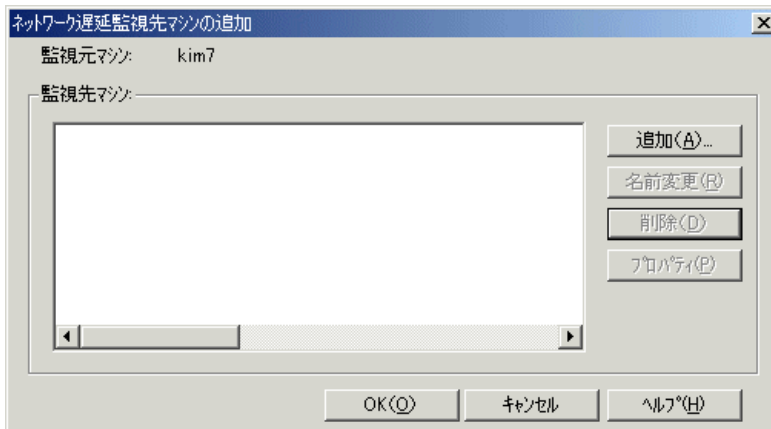
- 1 グラフ・ツリーの中で [ネットワーク遅延時間] グラフを選択して、右の表示枠までドラッグします。

- 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。[ネットワーク遅延時間] ダイアログ・ボックスが開きます。



- [次のマシンからネットワーク遅延を監視する] セクションで [追加] をクリックして、ネットワーク・パスの監視を開始する送信元マシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- [ネットワーク遅延時間] ダイアログ・ボックスの [宛先マシン] セクションで [追加] をクリックして、監視対象のパスの最終的な送信先となるマシンの

名前を入力します。[ネットワーク遅延監視先マシンの追加] ダイアログ・ボックスが開きます。



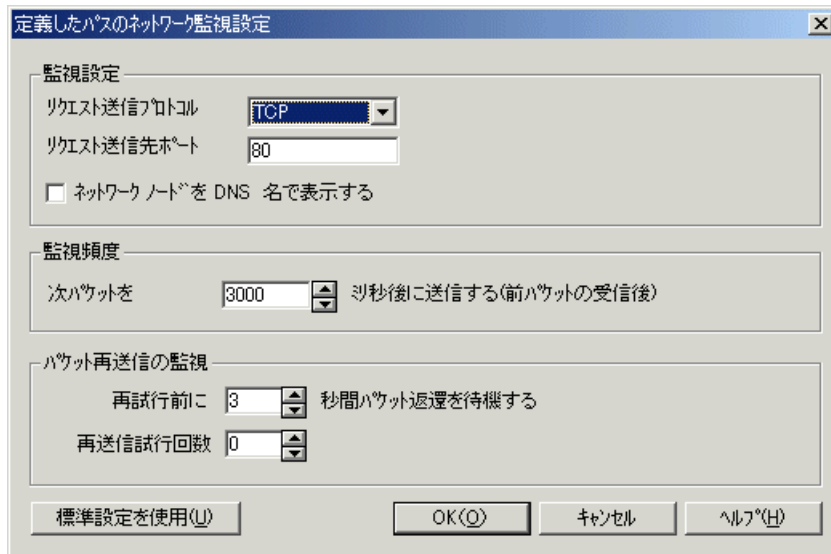
- 5 [追加] をクリックし、監視先のマシンの名前を入力して [OK] をクリックします。[ネットワーク遅延監視先マシンの追加] ダイアログ・ボックスにマシン名が表示されます。監視する各パスについてこの手順を繰り返します。

注：監視先マシンが **localhost** の場合は、**localhost** ではなく、ローカル・マシンの名前を入力してください。

マシン名を変更するには、[名前変更] をクリックして、新しいマシン名を入力します。

マシンを削除するには、削除するマシンを選択して [削除] をクリックします。

- 6 [プロパティ] をクリックして、ネットワーク・モニタの設定を追加指定します。
[定義したパスのネットワーク監視設定] ダイアログ・ボックスが開きます。



- 7 [監視設定] セクションでプロトコルを選択して、ネットワーク・パスで使用されているポート番号を入力します。ネットワーク・モニタでは、TCP、UDP、ICMP の3つのプロトコルをサポートしています。標準設定のプロトコルを使用することをお勧めします。Windows の標準設定は TCP で、UNIX では UDP です。
- 8 IP アドレスの他に、ネットワーク・パス上にある各ノードの DNS 名を表示するには、[ネットワークノードをDNS名で表示する] を選択します。このオプションを選択すると、ネットワーク・モニタの速度が遅くなります。
- 9 [監視頻度] セクションで、パケットを受信してから次のパケットを送信するまでモニタが待機する時間をミリ秒単位で指定します。標準設定値は 3000 ミリ秒です。実行されるシナリオが長く、安定したものである場合は、間隔を数秒長くすることもできます。

- 10 [パケット再送信の監視] セクションで、パケットが返送されるまで最長で何秒待機してからパケットの再送信を試みるかを指定します。標準の値は3秒です。ネットワークの規模が大きく、負荷が高い場合は（低容量のインターネット接続など）、この値を数秒大きくする必要があります。ネットワークの規模が小さい場合は（LAN など）、値を小さくできます。

また、パケットが最初に返送されなくなったときから、ネットワーク・モニタでパケットを最大何回まで再送信するかを選択します。標準設定の値は0回です。

ファイアウォール越しのネットワークの監視

送信元と送信先のマシンの間にファイアウォールが設置されているネットワークを監視する場合、ネットワーク・データ・パケットが送信先に到達するようにファイアウォールを設定する必要があります。

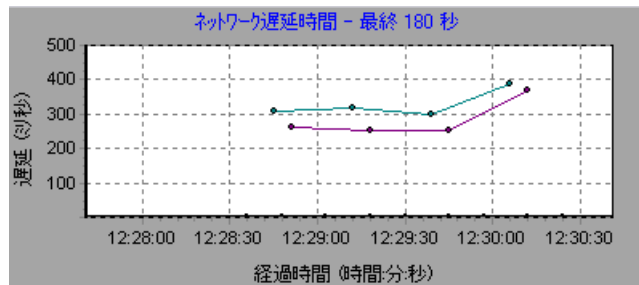
- ▶ TCP プロトコルを使用する場合、送信先マシンを保護するファイアウォールによって、送信 ICMP_TIMEEXCEEDED パケット（マシンからファイアウォールの外側に送信されるパケット）がブロックされないようにしなければなりません。また、送信元マシンを保護するファイアウォールで、ICMP_TIMEEXCEEDED パケットを内側に通過させ、TCP パケットが外に出られるようにする必要があります。
- ▶ ICMP プロトコルを使用する場合、送信先マシンのファイアウォールで、着信 ICMP_ECHO_REQUEST パケットまたは送信 ICMP_ECHO_REPLY および ICMP_ECHO_TIMEEXCEEDED パケットがブロックされないようにする必要があります。また、送信元マシンを保護するファイアウォールで、ICMP_ECHO_REPLY および ICMP_ECHO_TIMEEXCEEDED パケットを内側に通過させ、ICMP_ECHO_REQUEST パケットが外に出られるようにする必要があります。
- ▶ UDP プロトコルを使用する場合、UDP プロトコルで送信元マシンから送信先マシンにアクセスできることを確認します。送信先マシンのファイアウォールで、送信 ICMP_DEST_UNREACHABLE および ICMP_ECHO_TIMEEXCEEDED パケットがブロックされないようにします。また、送信元マシンを保護するファイアウォールで、ICMP_DEST_UNREACHABLE および ICMP_ECHO_TIMEEXCEEDED パケットを内側に通過できるようにする必要があります。

注：コントローラ・マシンと送信元マシン間にファイアウォールが設置されているときにネットワーク遅延モニタを実行するには、ファイアウォール越しの監視のために LoadRunner エージェント、MI リスナーおよびネットワーク・モニタを設定する必要があります。詳細については、242 ページ「ファイアウォールの内側の LoadRunner エージェントの設定」、252 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」、および 263 ページ「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」を参照してください。

ネットワーク遅延時間グラフの表示

[ネットワーク遅延時間] グラフには、シナリオを開始してからの経過時間（X 軸）の関数として、送信元マシンと送信先マシンの間のパス全体で発生する遅延時間（Y 軸）が表示されます。

グラフの中では、[ネットワーク遅延時間監視先マシンの追加] ダイアログ・ボックスで定義した各パスが、異なる色の線で示されます。



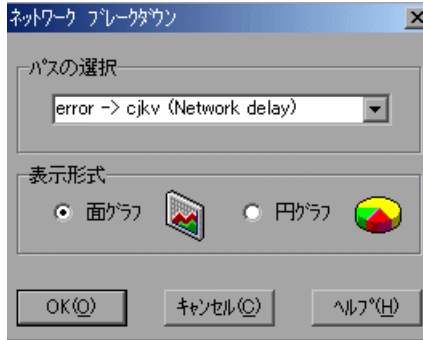
凡例に表示されている測定値の DNS 名を表示するには、グラフを右クリックして [DNS 名として表示] を選択します。

送信元マシンから各ノードまでの遅延時間をネットワーク・パスとともに表示するには、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・ボックスで、[サブパス] をクリックします。

また、パスの各セグメントの遅延時間も表示できます。

ネットワーク・セグメントの遅延時間を表示するには、次の手順で行います。

- 1 [ネットワーク遅延時間] グラフを右クリックして、[セグメントを表示] を選択します。[ネットワーク ブレークダウン] ダイアログ・ボックスが開きます。



- 2 ブレークダウンするパスを選択します。
- 3 選択したグラフのネットワーク・セグメントの表示形式として、面グラフまたは円グラフを選択します。
- 4 [OK] をクリックして、[ネットワーク ブレークダウン] ダイアログ・ボックスを閉じます。指定したパス上にあるネットワーク・セグメントの遅延時間が、グラフ表示領域に表示されます。

注：セグメントの遅延は近似値であり、正確な値が測定されるネットワーク・パス遅延とは必ずしも一致しません。パスの各セグメントの遅延は、監視元マシンからあるノードまでの遅延を計算し、監視元マシンからほかのノードまでの遅延を差し引くことによって見積ります。たとえば、セグメント B から C の遅延は、監視元マシンからポイント C までの遅延を測定し、監視元マシンからポイント B までの遅延を差し引いて計算します。

パス全体の遅延時間の表示に戻るには、右クリック・メニューで [セグメントを非表示] を選択します。

第 22 章

ファイアウォール・サーバ・パフォーマンスの監視

サーバのパフォーマンスのボトルネックを特定するために、シナリオ実行中に、ファイアウォール・サーバを監視できます。

本章では、以下の項目について説明します。

- ▶ Check Point FireWall-1 サーバ・モニタの設定

ファイアウォール・サーバ・モニタについて

ファイアウォール・サーバ・オンライン・モニタは、シナリオ実行中に、ファイアウォール・サーバのパフォーマンスを測定します。パフォーマンス・データを取得するために、シナリオを実行する前にファイアウォール・サーバを起動して、監視対象の統計値と測定値を指定する必要があります。

Check Point FireWall-1 サーバ・モニタの設定

Check Point FireWall-1 サーバを監視するには、Check Point FireWall-1 サーバ・モニタで測定するカウンタを選択する必要があります。カウンタは、[Check Point FireWall-1 SNMP リソース] ダイアログ・ボックスで選択します。

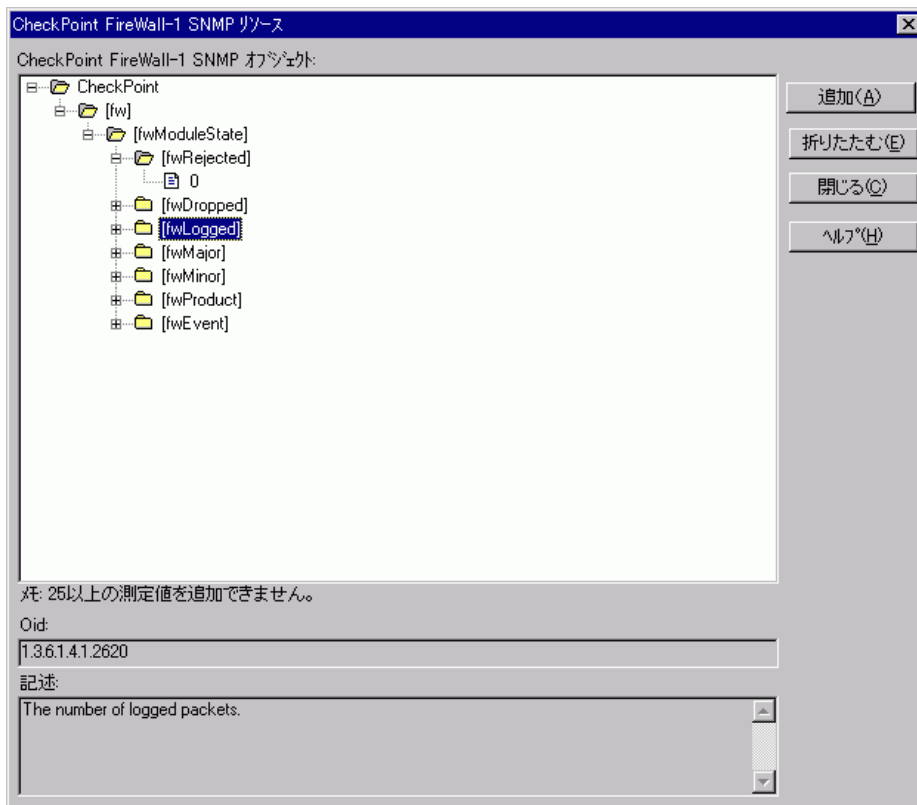
Check Point FireWall-1 サーバ・モニタの設定は、次の手順で行います。

- 1 グラフ・ツリーで CheckPoint FireWall-1 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Check Point FireWall-1] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視するマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

注： `snmp.cfg` ファイルにポート番号を指定できます。ポート番号を指定しなければ、LoadRunner は、CheckPoint FireWall-1 SNMP エージェントの標準設定の 260 番ポートに接続します。[マシンの追加] ダイアログ・ボックスで、次の形式でマシン名およびポート番号を指定することもできます。

<マシン名> : <ポート番号>

- 4 [Check Point FireWall-1] ダイアログ・ボックスの [リソース測定場所] 項目で、[追加] をクリックします。[Check Point FireWall-1 SNMP リソース] ダイアログ・ボックスが表示されます。



5 監視する測定値を選択します。

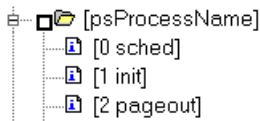
測定可能なパフォーマンス・カウンタの一覧については、369 ページを参照してください。

6 [追加] をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。

注：CheckPoint FireWall-1 モニタでは、測定値を 25 個まで監視できます。

7 [Check Point FireWall-1] 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。

注：Check Point FireWall-1 モニタの測定値情報レベルを上げるには、(測定項目の数値だけでなく) 測定項目の文字列値も表示されるようにし、(文字列値を測定項目の識別名として表示する) 名前修飾子を有効にします。名前修飾子を使用している次の測定値の例では、ProcessName の文字列値 (sched) が、そのインスタンス ID (0) のほかに表示されます。



```
[psProcessName]
├── [0 sched]
├── [1 init]
└── [2 pageout]
```

この機能を有効にするには、次の行を **LoadRunner** のインストール先フォルダ > %dat%monitors%snmp.cfg ファイルに追加します。

SNMP_show_string_nodes=1

使用上の注意：名前修飾子は複数選択できますが、使用されるのは、階層の先頭の修飾子です。[Check Point FireWall-1 リソース] ダイアログ・ボックスが開くたびに、snmp.cfg ファイルから情報が読み込み直されます。「名前修飾子付き」と「名前修飾子なし」という具合に同じ測定項目を 2 度追加することはできません。2 度追加すると、エラー・メッセージが表示されます。

Check Point FireWall-1 パフォーマンス・カウンタ

次の標準のカウンタを監視できます。

測定値	説明
fwRejected	拒否されたパケットの数。
fwDropped	ドロップされたパケットの数。
fwLogged	ログを取ったパケットの数。

第 23 章

Web サーバ・リソースの監視

LoadRunner の Web サーバ・リソース・モニタを使って、シナリオの実行中に Apache, Microsoft IIS, iPlanet (SNMP), iPlanet/Netscape サーバを監視して、サーバのパフォーマンスのボトルネックを特定することができます。

本章では、以下の項目について説明します。

- ▶ Apache モニタの設定
- ▶ Microsoft IIS モニタの設定
- ▶ iPlanet/Netscape モニタの設定
- ▶ iPlanet (SNMP) モニタの設定
- ▶ プロキシ・サーバを使用した監視

Web サーバ・リソース・モニタについて

Web サーバ・リソース・モニタは、シナリオの実行時の Apache, Microsoft IIS, iPlanet (SNMP), iPlanet/Netscape Web サーバのリソースの使用状況に関する情報を取得します。このデータを取得するには、シナリオを実行する前に、サーバのオンライン・モニタを起動して測定対象のリソースを指定する必要があります。

モニタ測定値の選択手順およびモニタの設定手順は、サーバの種類によって異なります。以降の項では、サーバの種類ごとに設定手順を説明します。

注：カウンタの中には、サーバでの最初の負荷テストで、Web サーバのパフォーマンスを調べたり、ボトルネックの原因を特定したりするのに特に便利なカウンタもあります。これらのカウンタの詳細については、670 ページ「負荷テストに役立つカウンタ」を参照してください。

Apache モニタの設定

Apache サーバを監視するには、サーバの統計情報の URL を知っておく必要があります。統計情報の URL を検証する簡単な方法は、ブラウザで URL を表示してみることです。

URL は、次の形式で指定します。

`http:// <サーバ名 /IP アドレス> : <ポート番号> /server-status?auto`

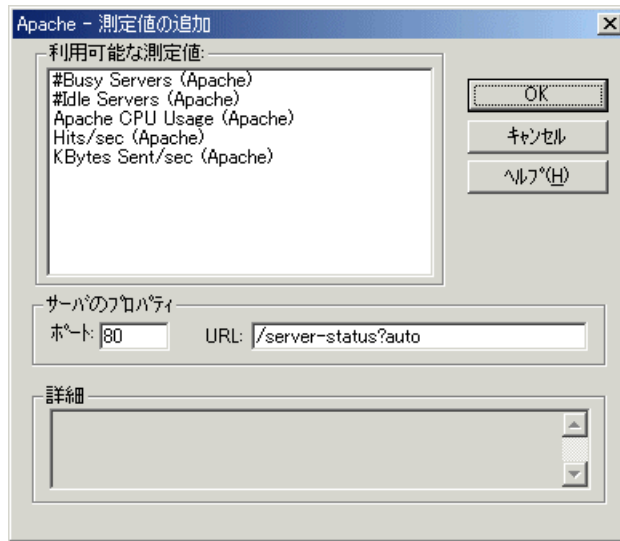
次に例を示します。

`http://stimpdy:80/server-status?auto`

Apache モニタを設定するには、次の手順で行います。

- 1 グラフ・ツリーで [Apache] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Apache] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [Apache] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。

[Apache - 測定値の追加] ダイアログ・ボックスが開き、使用可能な測定値とサーバのプロパティが表示されます。



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。測定可能なパフォーマンス・カウンタの一覧については、374 ページを参照してください。

- 5 [サーバのプロパティ] セクションにポート番号と URL (サーバ名を除く) を入力して、**[OK]** をクリックします。標準設定の URL は、`/server-status?auto` です。
- 6 [Apache] 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。

注： 標準設定のポート番号と URL は、サーバによって異なります。Web サーバの管理者に確認してください。

Apache パフォーマンス・カウンタ

次の表に、監視できる測定値およびサーバのプロパティの説明を示します。

測定値	説明
# Busy Servers	Busy 状態にあるサーバの数。
# Idle Servers	Idle 状態にあるサーバの数。
Apache CPU Usage	Apache サーバが CPU を使用した時間の割合。
Hits/sec	HTTP 要求の秒ごとの数。
KBytes Sent/sec	Web サーバがデータ・バイトを送信する速度。

標準設定のサーバのプロパティを変更するには、次の手順で行います。

- 1 < LoadRunner のインストール先フォルダ > \data\monitors ディレクトリにある **apache.cfg** ファイルを開きます。
- 2 **Delimiter=:** ステートメントの後にある、次のパラメータを編集します。

InfoURL サーバの統計情報の URL

ServerPort サーバ・ポート番号

SamplingRate LoadRunner モニタが、サーバの統計情報のポーリングを行う頻度（ミリ秒単位）。この値が 1000 より大きい場合、LoadRunner は、サンプリング頻度としてこの値を使いません。それ以外の場合は、[オプション] ダイアログ・ボックスの [モニタ] タブに定義されているサンプリング頻度を使います。

注：ファイアウォール越しに Apache サーバを監視するには、Web サーバのポート（標準設定では 80 番）を使用します。

Microsoft IIS モニタの設定

[MS IIS] ダイアログ・ボックスを使って、Microsoft IIS サーバ・モニタの測定値を選択します。

注：ファイアウォール越しに IIS サーバを監視するには、139 番ポートで TCP を使用します。

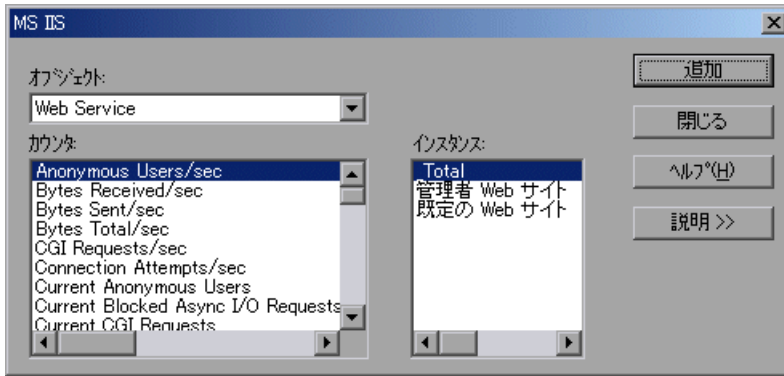
IIS サーバ・モニタを設定するには、次の手順で行います。

- 1 グラフ・ツリーで [MS IIS] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [MS IIS] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [MS IIS] ダイアログ・ボックスの [リソース測定場所] セクションで、監視対象の測定値を選択します。

測定可能なパフォーマンス・カウンタの一覧については、376 ページを参照してください。

注：Microsoft IIS モニタの標準設定のカウンタを変更するには、669 ページ「モニタの標準設定のカウンタの変更」を参照してください。

- 5 追加の測定値を選択するには、**[追加]** をクリックします。ダイアログ・ボックスが開き、Web Service オブジェクト、そのカウンタ、およびインスタンスが表示されます。



- 6 カウンタとインスタンスを選択します。**Ctrl** キーを使って、複数のカウンタを選択できます。インスタンスが意味を持つのは、強調表示されたカウンタの複数のインスタンスが実行されている場合だけです。各カウンタの説明を表示するには、**[説明 >>]** をクリックしてダイアログ・ボックスを拡張します。
- 7 **[追加]** をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、**[閉じる]** をクリックします。
- 8 **[MS IIS]** 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。

Microsoft IIS パフォーマンス・カウンタ

以下の表に、標準で監視可能な測定値の説明を示します。

対象	測定値	説明
Web Service	Bytes Sent/sec	Web サービスがデータ・バイトを送信する速度。
Web Service	Bytes Received/sec	Web サービスがデータ・バイトを受信する速度。

対象	測定値	説明
Web Service	Get Requests/sec	GET メソッドを使用する HTTP 要求の秒ごとの数。通常の GET 要求は、基本的なファイルの取得またはイメージ・マップに使用しますが、フォームにも使用できます。
Web Service	Post Requests/sec	POST メソッドを使用する HTTP 要求の秒ごとの数。通常、POST メソッドは、フォームまたはゲートウェイの要求に使用されます。
Web Service	Maximum Connections	Web サービスとの間で確立された同時接続の最大数。
Web Service	Current Connections	Web サービスとの間で現在確立されている接続の数。
Web Service	Current NonAnonymous Users	Web サービスに対して非匿名接続を行っている現在のユーザの数。
Web Service	Not Found Errors/sec	要求されたドキュメントが見つからなかったためにサーバが処理できなかった要求の秒ごとの数。通常、この種のエラーは、HTTP のエラー・コード 404 としてクライアントに通知されます。
Process	Private Bytes	プロセスによって割り当てられ、ほかのプロセスとは共有できないバイト数の最新の値。

iPlanet/Netscape モニタの設定

iPlanet または Netscape サーバを監視するには、管理サーバの URL を知っておく必要があります。管理サーバの URL を検証する簡単な方法は、ブラウザで URL を表示してみることです。

URL は、次の形式で指定します。

`http:// <管理サーバ名 /IP アドレス> : <ポート番号> /https- <管理サーバ名 /IP アドレス> /bin/sitemon?doit`

次に例を示します。

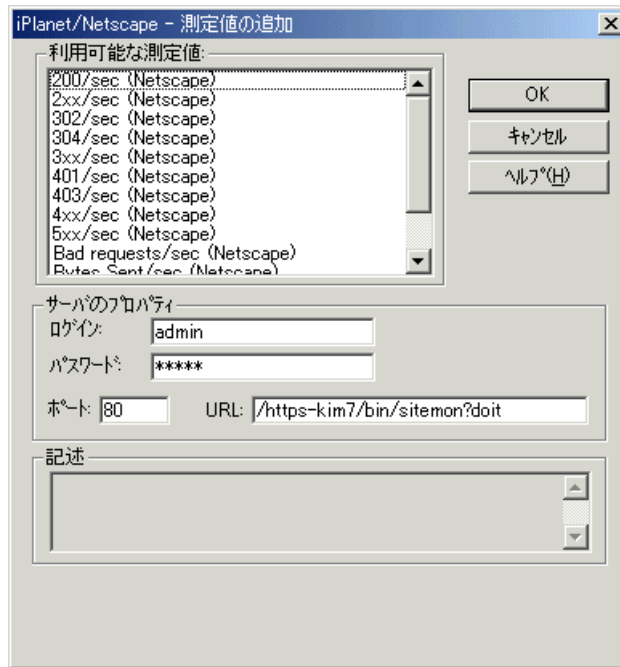
`http://lazarus:12000/https-lazarus.mercury.co.il/bin/sitemon?doit`

注：サーバの設定によっては、IP アドレスではなく、管理サーバ名を URL に含めなければなりません。また、管理サーバ名は iPlanet/Netscape サーバ名と異なる場合があります。

コントローラから iPlanet/Netscape モニタを起動するには、次の手順で行います。

- 1 グラフ・ツリーで [iPlanet/Netscape] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [iPlanet/Netscape] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [iPlanet/Netscape] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。

[iPlanet/Netscape - 測定値の追加] ダイアログ・ボックスが開き、使用可能な測定値とサーバのプロパティが表示されます。



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。測定可能なパフォーマンス・カウンタの一覧については、380 ページを参照してください。

5 [サーバのプロパティ] セクションの情報を入力します。

- ▶ ユーザのログイン名とパスワードを入力します。このユーザには、サーバの管理者権限が必要です。
- ▶ ポート番号と URL（サーバ名を除く）を入力して、[OK] をクリックします。標準設定の URL は、/https- <管理サーバ> /bin/sitemon?doit です。

6 [iPlanet/Netscape] 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。

注：標準設定のポート番号と URL は、サーバによって異なります。Web サーバの管理者に確認してください。サーバの設定によっては、IP アドレスではなく、サーバ名を URL に含めなければなりません。

Netscape パフォーマンス・カウンタ

次の表に、監視できる測定値およびサーバのプロパティの説明を示します。

測定値	説明
200/sec	サーバによって処理された成功トランザクションの秒ごとの数。
2xx/sec	サーバが 200 から 299 までのステータス・コードを処理した秒ごとの数。
302/sec	サーバによって処理された、再配置された URL の秒ごとの数。
304/sec	サーバがユーザに、サーバから新しいバージョンを取得するのではなく、URL のローカル・コピーを使用するように伝えた要求の秒ごとの数。
3xx/sec	サーバが 300 から 399 までのステータス・コードを処理した秒ごとの数。
401/sec	サーバによって処理された認証が拒否された要求の秒ごとの数。
403/sec	サーバによって処理された URL アクセス禁止ステータス・コードの秒ごとの数。
4xx/sec	サーバが 400 から 499 までのステータス・コードを処理した秒ごとの数。
5xx/sec	サーバが 500 以上のステータス・コードを処理した秒ごとの数。
Bad requests/sec	サーバが不正な要求を処理した秒ごとの数。
Bytes sent/sec	Web サーバがデータ・バイトを送信した速度。
Hits/sec	HTTP 要求の秒ごとの数。
xxx/sec	HTTP ステータス・コードを返したタイムアウトやその他のエラーを除いた、サーバによって処理されたすべてのステータス・コード (2xx ~ 5xx) の秒ごとの数。

標準設定のサーバのプロパティを変更するには、次の手順で行います。

- 1 < LoadRunner のインストール先フォルダ > %dat%monitors ディレクトリにある Netscape.cfg ファイルを開きます。
- 2 [Netscape] セクションで次のパラメータを編集します。

Counter	LoadRunner iPlanet/Netscape モニタが表示するカウンタ数。この値は、ファイルで定義されたカウンタ数と同じである必要があります。
InfoURL	サーバの統計情報の URL
ServerPort	サーバ・ポート番号
ServerLogin	サーバへのログイン名
ServerPassword	ログイン名に対応するログイン・パスワード
SamplingRate	LoadRunner モニタが、サーバの統計情報のポーリングを行う頻度（ミリ秒単位）。この値が 1000 より大きい場合、LoadRunner は、サンプリング頻度としてこの値を使います。それ以外の場合は、[オプション] ダイアログ・ボックスの [モニタ] タブに定義されているサンプリング頻度を使います。

注：ファイアウォール越しに iPlanet/Netscape サーバを監視するには、iPlanet/Netscape Administration サーバ・ポートを使用します。このポートは、サーバのインストール中に設定します。

iPlanet (SNMP) モニタの設定

iPlanet (SNMP) モニタは、Simple Network Management Protocol (SNMP) を使って iPlanet (SNMP) サーバの統計値を取得します。[iPlanet (SNMP) リソース] ダイアログ・ボックスを使って、iPlanet (SNMP) モニタの測定値を定義します。

注： iPlanet (SNMP) サーバを監視するには、サーバの構成によって 161 番ポートまたは 162 番ポートのどちらかを使用します。

iPlanet (SNMP) リソース・モニタを設定するには、次の手順で行います。

- 1 グラフ・ツリーで [iPlanet (SNMP)] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [iPlanet (SNMP)] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

注： iPlanet SNMP エージェントを実行しているポートの番号が SNMP の標準のポート番号と異なる場合には、ポート番号を定義する必要があります。[マシンの追加] ダイアログ・ボックスに、次の情報を入力します。

<サーバ名：ポート番号>

次に例を示します。

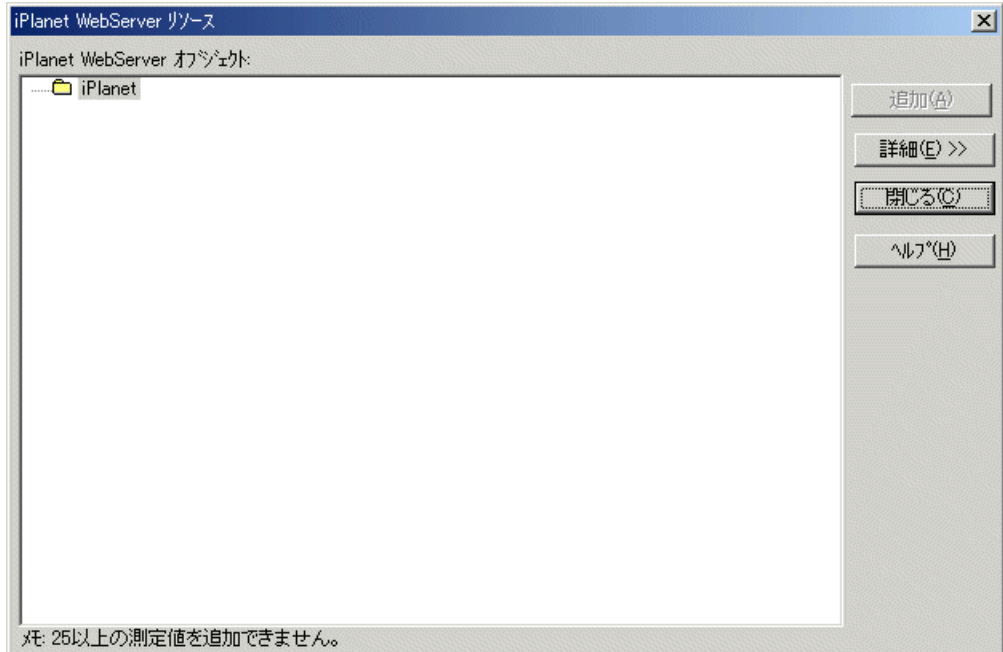
digi:8888

また、< LoadRunner のインストール先フォルダ > %dat%monitors にある設定ファイル **snmp.cfg** で、iPlanet サーバの標準設定のポートを定義することもできます。たとえば、iPlanet サーバで SNMP エージェントによって使用されているポート番号が 8888 の場合、**snmp.cfg** ファイルを次のように編集します。

```
; iPlanet (WebServer)
[cm_snmp_mon_iws60]
port=8888
```

- 4 [iPlanet (SNMP)] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。

[iPlanet WebServer リソース] ダイアログ・ボックスが開きます。



- 5 [iPlanet WebServer オブジェクト] ツリーが表示されます。

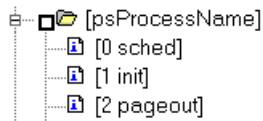
測定可能なパフォーマンス・カウンタの一覧については、385 ページを参照してください。

- 6 オブジェクトを測定するには、対象のオブジェクトを選択して [追加] をクリックします。各リソースの説明を表示するには、[詳細 >>] をクリックしてダイアログ・ボックスを拡張します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。

注： iPlanet (SNMP) モニタでは測定値を 25 個まで監視できます。

- 7 [iPlanet (SNMP)] 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。

注：iPlanet (SNMP) モニタの測定情報のレベルを上げるには、(測定項目の数値だけでなく) 測定項目の文字列値も表示されるようにし、(文字列値を測定項目の識別名として表示する) 名前修飾子を有効にします。名前修飾子を使用している次の測定値の例では、**ProcessName** の文字列値 (**sched**) がそのインスタンス ID (**0**) のほかに表示されます。



```
├── [psProcessName]
│   ├── [0 sched]
│   ├── [1 init]
│   └── [2 pageout]
```

この機能を有効にするには、次の行を **LoadRunner** のインストール先フォルダ > `%dat%\monitors\snmp.cfg` ファイルに追加します。

SNMP_show_string_nodes=1

使用上の注意：名前修飾子は複数選択できますが、使用されるのは、階層の先頭の修飾子です。[iPlanet (SNMP) リソース] ダイアログ・ボックスが開くたびに、**snmp.cfg** ファイルから情報が読み込み直されます。「名前修飾子付き」と「名前修飾子なし」という具合に同じ測定項目を2度追加することはできません。2度追加すると、エラー・メッセージが表示されます。

iPlanet (SNMP) パフォーマンス・カウンタ

次の表に、監視できる測定値およびサーバのプロパティの説明を示します。

測定値	説明
iwsInstanceTable	iPlanet Web サーバ・インスタンス。
iwsInstanceEntry	iPlanet Web サーバ・インスタンス。
iwsInstanceIndex	サーバ・インスタンスのインデックス。
iwsInstanceId	サーバ・インスタンスの識別子。
iwsInstanceVersion	サーバ・インスタンスのソフトウェア・バージョン。
iwsInstanceDescription	サーバ・インスタンスの説明。
iwsInstanceOrganization	サーバ・インスタンスに対応する担当責任組織。
iwsInstanceContact	サーバ・インスタンスに対応する 1 人または複数人の担当責任者の連絡先情報。
iwsInstanceLocation	サーバ・インスタンスの場所。
iwsInstanceStatus	サーバ・インスタンスのステータス。
iwsInstanceUptime	サーバ・インスタンスの連続稼動時間。
iwsInstanceDeathCount	サーバ・インスタンス・プロセスが異常停止した回数。
iwsInstanceRequests	処理した要求の数。
iwsInstanceInOctets	受信したオクテットの数。
iwsInstanceOutOctets	送信したオクテットの数。
iwsInstanceCount2xx	発行された 200 レベル (Successful) 応答の数。
iwsInstanceCount3xx	発行された 300 レベル (Redirection) 応答の数。
iwsInstanceCount4xx	発行された 400 レベル (Client Error) 応答の数。
iwsInstanceCount5xx	発行された 500 レベル (Server Error) 応答の数。
iwsInstanceCountOther	発行されたその他の (2xx, 3xx, 4xx, 5xx のどれでもない) 応答の数。
iwsInstanceCount200	発行された 200 (OK) 応答の数。

測定値	説明
iwsInstanceCount302	発行された 302 (Moved Temporarily) 応答の数。
iwsInstanceCount304	発行された 304 (Not Modified) 応答の数。
iwsInstanceCount400	発行された 400 (Bad Request) 応答の数。
iwsInstanceCount401	発行された 401 (Unauthorized) 応答の数。
iwsInstanceCount403	発行された 403 (Forbidden) 応答の数。
iwsInstanceCount404	発行された 404 (Not Found) 応答の数。
iwsInstanceCount503	発行された 503 (Unavailable) 応答の数。
iwsInstanceLoad1Minute Average	システムの 1 分間の平均負荷。
iwsInstanceLoad5Minute Average	システムの 5 分間の平均負荷。
iwsInstanceLoad15Minute Average	システムの 15 分間の平均負荷。
iwsInstanceNetworkInOctets	ネットワークで送信されたオクテットの秒ごとの数。
iwsInstanceNetworkOutOctets	ネットワークで受信されたオクテットの秒ごとの数。
iwsVsTable	iPlanet Web サーバ仮想サーバ群。
iwsVsEntry	iPlanet Web サーバ仮想サーバ。
iwsVsIndex	仮想サーバのインデックス。
iwsVsId	仮想サーバの識別子。
iwsVsRequests	処理した要求の数。
iwsVsInOctets	受信したオクテットの数。
iwsVsOutOctets	送信したオクテットの数。
iwsVsCount2xx	発行された 200 レベル (Successful) 応答の数。
iwsVsCount3xx	発行された 300 レベル (Redirection) 応答の数。
iwsVsCount4xx	発行された 400 レベル (Client Error) 応答の数。
iwsVsCount5xx	発行された 500 レベル (Server Error) 応答の数。

測定値	説明
iwsVsCountOther	発行されたその他の (2xx, 3xx, 4xx, 5xx のどれでもない) 応答の数。
iwsVsCount200	発行された 200 (OK) 応答の数。
iwsVsCount302	発行された 302 (Moved Temporarily) 応答の数。
iwsVsCount304	発行された 304 (Not Modified) 応答の数。
iwsVsCount400	発行された 400 (Bad Request) 応答の数。
iwsVsCount401	発行された 401 (Unauthorized) 応答の数。
iwsVsCount403	発行された 403 (Forbidden) 応答の数。
iwsVsCount404	発行された 404 (Not Found) 応答の数。
iwsVsCount503	発行された 503 (Unavailable) 応答の数。
iwsProcessTable	iPlanet Web サーバ・プロセス群。
iwsProcessEntry	iPlanet Web サーバ・プロセス。
iwsProcessIndex	プロセスのインデックス。
iwsProcessId	オペレーティング・システム・プロセス識別子。
iwsProcessThreadCount	要求処理スレッドの数。
iwsProcessThreadIdle	現在アイドル状態の要求処理スレッドの数。
iwsProcessConnectionQueue Count	現在接続キュー内にある接続の数。
iwsProcessConnectionQueue Peak	過去に同時にキューに置かれた接続の最大数。
iwsProcessConnectionQueue Max	接続キューに置くことのできる接続の最大数。
iwsProcessConnectionQueue Total	受け付けられた接続の数。
iwsProcessConnectionQueue Overflows	接続キューのオーバーフローのために拒否された接続の数。
iwsProcessKeepaliveCount	現在キープアライブ・キュー内にある接続の数。

測定値	説明
iwsProcessKeepaliveMax	キープアライブ・キューに置くことのできる接続の最大数。
iwsProcessSizeVirtual	プロセスのサイズ (KB)。
iwsProcessSizeResident	プロセスの常駐サイズ (KB)。
iwsProcessFractionSystemMemoryUsage	システム・メモリ内のプロセス・メモリの割合。
iwsListenTable	iPlanet Web サーバ・リッスン・ソケット群。
iwsListenEntry	iPlanet Web サーバ・リッスン・ソケット。
iwsListenIndex	リッスン・ソケットのインデックス。
iwsListenId	リッスン・ソケットの識別子。
iwsProcessTable	ソケットがリッスンしているアドレス。
iwsListenPort	ソケットがリッスンしているポート。
iwsListenSecurity	暗号化サポート。
iwsThreadPoolTable	iPlanet Web サーバ・スレッド・プール群。
iwsThreadPoolEntry	iPlanet Web サーバ・スレッド・プール。
iwsThreadPoolIndex	スレッド・プールのインデックス。
iwsThreadPoolId	スレッド・プールの識別子。
iwsThreadPoolCount	キューに置かれた要求の数。
iwsThreadPoolPeak	過去に同時にキューに置かれた要求の最大数。
iwsThreadPoolMax	キューに置くことのできる要求の最大数。
iwsCpuTable	iPlanet Web サーバ CPU 群。
iwsCpuEntry	iPlanet Web サーバ CPU。
iwsCpuIndex	CPU のインデックス。
iwsCpuId	CPU の識別子。
iwsCpuIdleTime	CPU のアイドル時間。

測定値	説明
<code>iwsCpuUserTime</code>	CPU のユーザ時間。
<code>iwsCpuKernelTime</code>	CPU のカーネル時間。

プロキシ・サーバを使用した監視

LoadRunner では、コントローラと監視対象のサーバの間にプロキシ・サーバがある場合に、Apache および Netscape モニタを使用して監視できます。監視を行うには、Apache モニタでは **< LoadRunner のインストール先フォルダ > %dat%monitors%apache.cfg**、Netscape モニタでは **< LoadRunner のインストール先フォルダ > %dat%monitors%Netscape.cfg** にある設定ファイルで設定を定義します。

設定を定義する前に、プロキシ設定を、Internet Explorer の接続設定または設定ファイルのプロキシ設定のどちらから取得するかを LoadRunner に指定します。

LoadRunner で Internet Explorer 接続からプロキシ設定を取得するには、次の手順で行います。

- 1 設定ファイルのプロキシ設定 (Proxy Setting) セクションで「**useProxy**」変数に値「1」を割り当てます。
- 2 プロキシでユーザ名、パスワード、ドメインが必要な場合は、これらのパラメータを「**proxyUsername**」, 「**proxyPassword**」, 「**proxyDomain**」の各行に指定します。

LoadRunner で設定ファイルからプロキシ設定を取得するには、次の手順で行います。

- 1 設定ファイルのプロキシ設定 (Proxy Setting) セクションで「**httpProxy**」行にプロキシ情報を入力します。次の形式を使用します。

```
[ <プロトコル> =][ <スキーム> :/] <プロキシ> [: <ポート> ] [ <プロトコル> =][ <スキーム> :/] <プロキシ> [: <ポート> ] ]
```

次に例を示します。

```
httpProxy=http=http://my_http_proxy:8080 https=https://my_https_proxy:9000
```

- 2 プロキシでユーザ名、パスワード、ドメインが必要な場合は、これらのパラメータを「**proxyUsername**」, 「**proxyPassword**」, 「**proxyDomain**」の各行に指定します。

LoadRunner をサーバに（プロキシ設定をすべて無視して）直接接続するには、次の手順で行います。

設定ファイルのプロキシ設定（Proxy Setting）セクションで「**useProxy**」変数に値「0」を割り当てます。

第 24 章

Web アプリケーション・サーバ・リソースの監視

シナリオの実行中に Web アプリケーション・サーバを監視して、アプリケーション・サーバ・パフォーマンスのボトルネックを特定するには、LoadRunner の Web アプリケーション・サーバ・リソース・モニタを使います。

本章では、以下の項目について説明します。

- ▶ Ariba モニタの設定
- ▶ ATG Dynamo モニタの設定
- ▶ BroadVision モニタの設定
- ▶ ColdFusion モニタの設定
- ▶ Fujitsu INTERSTAGE モニタの設定
- ▶ iPlanet (NAS) モニタの設定
- ▶ MS Active Server Pages モニタの設定
- ▶ Oracle9iAS HTTP モニタの設定
- ▶ SilverStream モニタの設定
- ▶ WebLogic (SNMP) モニタの設定
- ▶ WebLogic (JMX) モニタの設定
- ▶ WebSphere モニタの設定
- ▶ WebSphere (EPM) モニタの設定

Web アプリケーション・サーバ・リソース・モニタについて

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), WebSphere アプリケーション・サーバのリソースの使用状況に関する情報を表示します。パフォーマンス・データを取得するには、シナリオを実行する前に、サーバのオンライン・モニタを起動して測定対象のリソースを指定する必要があります。

モニタ測定値の選択手順およびモニタの設定手順は、サーバの種類によって異なります。以降の項では、サーバの種類ごとに設定手順を説明します。

Ariba モニタの設定

[Ariba Monitor 設定] ダイアログ・ボックスを使って、Ariba サーバを監視するための測定値を選択します。

注：ファイアウォール越しに Ariba サーバを監視するために使用するポートは、サーバの構成によって異なります。

Ariba モニタの設定は、次の手順で行います。

- 1 グラフ・ツリーで [Ariba] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Ariba] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。次の形式に従って、サーバ名または IP アドレスを入力します。

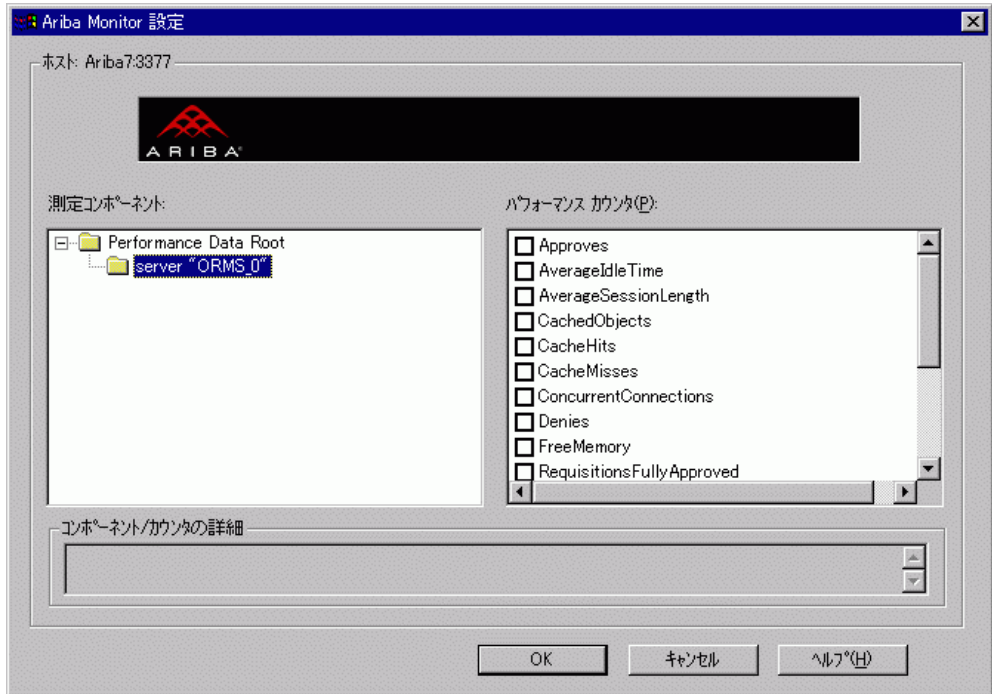
<サーバ名> : <ポート番号>

例を次に示します。

merc1:12130

マシンを実行するプラットフォームを選択して、[OK] をクリックします。

- 4 [Ariba] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。[Ariba Monitor 設定] ダイアログ・ボックスが開きます。
- 5 [測定コンポーネント] ツリーが表示されます。



- 6 [Ariba Monitor 設定] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。使用できるパフォーマンス・カウンタのリストについては、394 ページを参照してください。
- 7 [Ariba Monitor 設定] ダイアログ・ボックスと [Ariba] ダイアログ・ボックスで [OK] をクリックすると、Ariba モニタが起動します。

XML アクセシビリティ検証

パフォーマンス XML ファイルを表示できるのは、XML と互換性のあるブラウザだけです。

XML ファイルがアクセス可能かどうか検証するには、次の手順で行います。

ブラウザで XML ファイルを表示します。URL は、次の形式で指定します。

`http:// <サーバ名 : ポート番号> /metrics?query=getStats`

次に例を示します。

`http://merc1:12130/metrics?query=getStats`

注：ブラウザが XML と互換性があっても、次のエラーが返される場合があります。

The XML page cannot be displayed.

この場合、ブラウザには表示できませんが、XML ファイルは所定の場所に存在し、Ariba パフォーマンス・モニタでそのファイルにアクセスできます。

Ariba パフォーマンス・カウンタ

以下の表に、監視可能なカウンタの説明を示します。

主要なサーバ・パフォーマンス・カウンタ

測定値	説明
Requisitions Finished	この値を取得した瞬間におけるワーカ・キューの長さを示す測定値。ワーカ・キューが長いほど、ユーザ要求の処理は遅れます。
Worker Queue Length	この値を取得した瞬間におけるワーカ・キューの長さを示す測定値。ワーカ・キューが長いほど、ユーザ要求の処理は遅れます。
Concurrent Connections	この値を取得した瞬間における、同時ユーザ接続数を示す測定値。
Total Connections	Ariba Buyer が開始されてからの同時ユーザ接続の累積数。

測定値	説明
Total Memory	この値を取得する瞬間において Ariba Buyer が使用しているメモリ量を示す測定値 (KB)。
Free Memory	この値を取得する瞬間において使用されていない予約メモリ量を示す測定値 (バイト)。
Up Time	Ariba Buyer の開始されたときからの稼働時間 (時間と分)。
Number of Threads	この値を取得する瞬間において存在しているサーバ・スレッドの数を示す測定値。
Number of Cached Objects	この値を取得する瞬間においてメモリに格納されている Ariba Buyer オブジェクトの数を示す測定値。
Average Session Length	前のサンプリング時間以降ログ・アウトしたすべてのユーザのユーザ・セッションの平均の長さ (秒)。この値は、ユーザがサーバに接続している時間の平均を表します。
Average Idle Time	前のサンプリング時間以降のアクティブなすべてのユーザの平均アイドル時間 (秒)。アイドル時間とは、同じユーザによる連続した 2 つのユーザ要求の間の時間のことです。
Approves	サンプリング時間中に発生した承認の累積数。Approve は、1 つの Approvable を承認するユーザで構成されます。
Submits	前のサンプリング時間以降送信された Approvable の累積数。
Denies	前のサンプリング時間以降、拒否された送信 Approvable の累積数。
Object Cache Accesses	前のサンプリング時間以降のオブジェクト・キャッシュへの累積アクセス数 (読み取りと書き込みの両方)。
Object Cache Hits	前のサンプリング時間以降の、オブジェクト・キャッシュへの成功した累積アクセス数 (キャッシュ・ヒット数)。

システム関連パフォーマンス・カウンタ

測定値	説明
Database Response Time	前のサンプリング時間以降のデータベース・リクエストに対する平均応答時間（秒）。
Buyer to DB server Traffic	前のサンプリング時間以降、Ariba Buyer がデータベース・サーバに送信した累積バイト数。
DB to Buyer server Traffic	前のサンプリング時間以降、データベース・サーバが Ariba Buyer に送信した累積バイト数。
Database Query Packets	前のサンプリング時間以降、Ariba Buyer がデータベース・サーバに送信した平均パケット数。
Database Response Packets	前のサンプリング時間以降、データベース・サーバが Ariba Buyer に送信した平均パケット数。

ATG Dynamo モニタの設定

ATG Dynamo モニタは、SNMP を使って ATG Dynamo サーバの統計値を取得します。[ATG Dynamo リソース] ダイアログ・ボックスを使って、ATG Dynamo モニタの測定値を定義します。

ATG Dynamo サーバ・モニタの設定は、次の手順で行います。

- 1 グラフ・ツリーで [ATG Dynamo] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [ATG Dynamo] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

注： ATG SNMP エージェントによって使用されているポートが、ATG SNMP の標準のポート番号である 8870 と異なる場合には、ポート番号を定義する必要があります。ATG サーバの標準のポート番号は、< LoadRunner のインストール先フォルダ > %dat%monitors にある **snmp.cfg** 構成ファイルで定義できます。たとえば、ATG システムで SNMP エージェントによって使用されているポート番号が 8888 の場合、**snmp.cfg** ファイルを次のように編集します。

```
; ATG Dynamo
[cm_snmp_mon_atg]
port=8888
```

[マシンの追加] ダイアログ・ボックスで、次の形式でマシン名およびポート番号を指定することもできます。

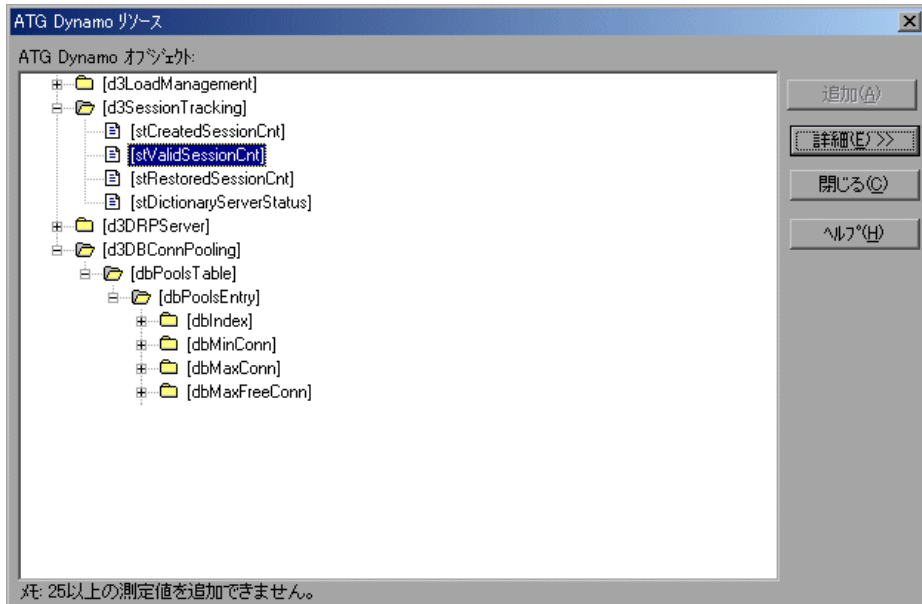
<サーバ名:ポート番号>

次に例を示します。

```
digi:8888
```

- 4 [ATG Dynamo] ダイアログ・ボックスの [リソース測定場所] セクションの [追加] をクリックします。[ATG Dynamo リソース] ダイアログ・ボックスが開きます。

- 5 [ATG Dynamo オブジェクト] ツリーから、監視対象の測定値を選択します。



測定可能なパフォーマンス・カウンタの一覧については、399 ページを参照してください。

- 6 [追加] をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。

注：ATG Dynamo モニタでは、測定値を 25 項目まで監視できます。

- 7 [ATG Dynamo] 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。

注： ATG Dynamo モニタの測定情報のレベルを上げるには、（測定項目の数値だけでなく）測定項目の文字列値も表示されるようにし、（文字列値を測定項目の識別名として表示する）名前修飾子を有効にします。名前修飾子を使用している次の測定値の例では、**ProcessName** の文字列値（**sched**）がそのインスタンス ID（0）のほかに表示されます。

```

├─ [psProcessName]
│  └─ [0 sched]
│     └─ [1 init]
│        └─ [2 pageout]

```

この機能を有効にするには、次の行を < **LoadRunner** のインストール先フォルダ > `¥dat¥monitors¥snmp.cfg` ファイルに追加します。

`SNMP_show_string_nodes=1`

使用上の注意：名前修飾子は複数選択できますが、使用されるのは、階層の先頭の修飾子です。[ATG Dynamo リソース] ダイアログ・ボックスが開くたびに、`snmp.cfg` ファイルから情報が読み込み直されます。「名前修飾子付き」と「名前修飾子なし」という具合に同じ測定項目を 2 度追加することはできません。2 度追加すると、エラー・メッセージが表示されます。

ATG Dynamo パフォーマンス・カウンタ

以下の表に、監視可能な測定値の説明を示します。

d3System

測定値	説明
<code>sysTotalMem</code>	現在オブジェクトの割り当てに利用できるメモリの総量 (バイト)。
<code>sysFreeMem</code>	将来的にオブジェクトの割り当てに利用できるメモリの、現在の推定総量 (バイト)。
<code>sysNumInfoMsgs</code>	書き込まれたシステム・グローバル情報メッセージの数。
<code>sysNumWarningMsgs</code>	書き込まれたシステム・グローバル警告メッセージの数。
<code>sysNumErrorMsgs</code>	書き込まれたシステム・グローバル・エラー・メッセージの数。

d3LoadManagement

測定値	説明
lmIsManager	Dynamo が負荷マネージャを実行している場合は真。
lmManagerIndex	負荷管理エンティティの一覧の先頭からの Dynamo のオフセットを返します。
lmIsPrimaryManager	負荷マネージャが実行中のプライマリ・マネージャである場合は真。
lmServicingCMs	負荷マネージャが、接続モジュール・ポーリング間隔として設定された時間内に接続モジュール要求をサービスした場合は真。
lmCMLDRPPort	接続モジュール・エージェントのポート。
lmIndex	管理される各エンティティの一意の値。
lmSNMPPort	エントリの SNMP エージェントのポート。
lmProbability	エントリに新規セッションが割り当てられる確率。
lmNewSessions	エントリが新規セッションを受け付けているかどうか、また、負荷マネージャがエントリへの新規セッションの送信を許可しているかどうかを示します。この値には、 lmNewSessionOverride が示すオーバーライドも含まれます。
lmNewSessionOverride	サーバが新規セッションを受け付けているかどうかを示すオーバーライド設定。

d3SessionTracking

測定値	説明
stCreatedSessionCnt	作成されたセッションの数。
stValidSessionCnt	有効なセッションの数。
stRestoredSessionCnt	サーバに移行されたセッションの数。
StDictionaryServerStatus	d3Session Tracking。

d3DRPServer

測定値	説明
drpPort	DRP サーバのポート。
drpTotalReqsServed	サービスされた DRP 要求の総数。
drpTotalReqTime	全 DRP 要求の合計サービス時間（ミリ秒）。
drpAvgReqTime	各 DRP 要求の平均サービス時間（ミリ秒）。
drpNewessions	Dynamo が新規セッションを受け付けている場合は真。

d3DBConnPooling

測定値	説明
dbPoolsEntry	プールの設定と現在のステータスに関する情報が含まれているプール・サービス・エントリ。
dbIndex	各プール・サービスの一意の値。
dbPoolID	データベース接続プール・サービスの名前。
dbMinConn	プールされた接続の最小数。
dbMaxConn	プールされた接続の最大数。
dbMaxFreeConn	同時に存在し得る空き状態のプールされた接続の最大数。
dbBlocking	プールがチェック・アウトを阻止するかどうかを示します。
dbConnOut	チェック・アウトされた接続の数を返します。
dbFreeResources	プール中の空き状態の接続の数を返します。この値は、現在チェック・アウトされていない、実際に作成された接続数を表します。プールの最大接続数までさらにいくつの接続が作成できるかは表しません。
dbTotalResources	プールの総接続数を返します。この値は、実際に作成された接続数を表し、さらにいくつの接続を作成してプールで使用できるかは表しません。

BroadVision モニタの設定

BroadVision サーバを監視するには、サーバでサービスを起動する権限をクライアントに与える必要があります。

注：ファイアウォール越しに BroadVision サーバを監視するために使用するポートは、サーバの構成によって異なります。

BroadVision サーバに対する権限をクライアントに与えるには、次の手順で行います。

- ▶ 次の Iona Technologies (Orbix) コマンドを使って、ロード・ジェネレータ・マシンのユーザとアクセス権限を設定します。

```
chmodit [-h <ホスト> ] [-v] { <サーバ> | -a <ディレクトリ> }  
{i{+,-}{user.group} | l{+,-}{user.group} }
```

- ▶ BroadVision モニタに接続しようとして問題が生じた場合には、「all」に対する権限を定義し直す必要がある場合があります。

「all」に対する権限を呼び出すには、BroadVision サーバのコマンド・プロンプトで次のコマンドを入力します。

```
# chmodit <サーバ> i+all
```

「all」の権限を開始するには、BroadVision サーバのコマンド・プロンプトで次のコマンドを入力します。

```
# chmodit <サーバ> l+all
```

- ▶ または、ORBIX_ACL を設定します。BroadVision/Orbix 設定ファイルで ORBIX_ACL=i+all l+all を設定すると、権限が「all」に対して与えられます。

さらに、BroadVision サーバを監視するには、コントローラ・マシンに JDK 1.2 以降がインストールされている必要があります。

JDK 1.2 は、Web サイト <http://java.sun.com/products/jdk/1.2/> からダウンロードして、インストール手順に従ってインストールできます。

モニタを有効にする前に、Java 環境が正しく設定されていることを確認します。

Java 環境を設定するには、次の手順で行います。

- 1 Windows レジストリを開きます。
- 2 レジストリには、JDK 1.2 のインストール先ディレクトリにある Java 実行ファイル (java.exe) への正しいパスが含まれている必要があります。次のレジストリ・キーを確認します。

```
HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows¥CurrentVersion¥App Paths¥java.exe
```

- 3 レジストリには、JRE 1.2 のインストール先ディレクトリにある Java ランタイム環境 (JRE) への正しいパスが含まれている必要があります。次のレジストリ・キーを確認します。

```
HKEY_LOCAL_MACHINE¥SOFTWARE¥JavaSoft¥Java Runtime Environment¥1.2¥JavaHome
```

BroadVision オンライン・モニタを設定するには、次の手順で行います。

- 1 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 2 [BroadVision] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、次の形式に従って BroadVision サーバの名前または IP アドレスをポート番号とともに入力します。

<サーバ名> : <ポート番号>

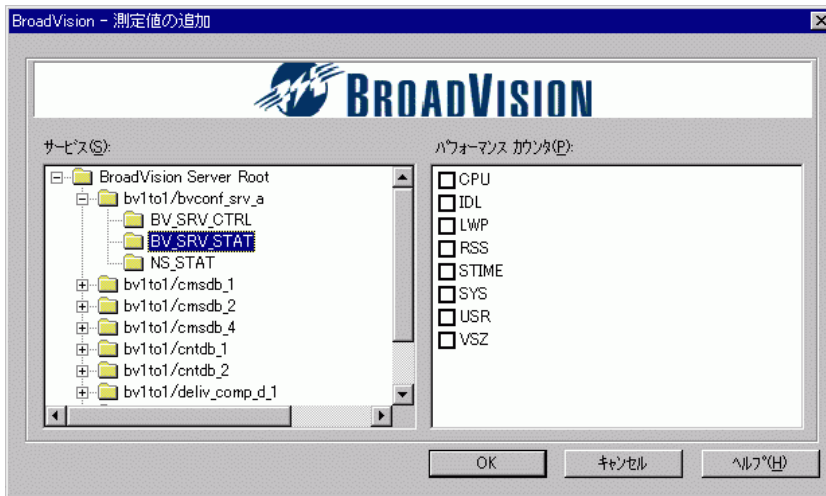
例を次に示します。

```
dnsqa:1221
```

マシン・プラットフォームを選択して、[OK] をクリックします。

- 3 [BroadVision] ダイアログ・ボックスの [リソース測定場所] セクションの [追加] をクリックします。

[BroadVision –測定値の追加] ダイアログ・ボックスが開き、使用可能な測定値が表示されます。



- 4 [サービス] ツリーを参照して、[BroadVision –測定値の追加] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。

測定可能なパフォーマンス・カウンタの一覧については、404 ページを参照してください。

- 5 [BroadVision –測定値の追加] ダイアログ・ボックスと [BroadVision] ダイアログ・ボックスで [OK] をクリックすると、BroadVision モニタが起動します。

BroadVision パフォーマンス・カウンタ

次の表に、監視の可能なサーバおよびサービスを示します。

サーバ	複数インスタンス	説明
adm_srv	不可	One-To-One ユーザ管理サーバ。必ず 1 つ必要です。
alert_srv	不可	警告サーバ。Alert システムに対する直接の IDL 関数呼び出しを処理します。
bvconf_srv	不可	One-To-One 構成管理サーバ。必ず 1 つ必要です。

サーバ	複数インスタンス	説明
cmsdb	可	訪問者管理データベース・サーバ。
cntdb	可	コンテンツ・データベース・サーバ。
deliv_smtp_d	可	電子メール・タイプのメッセージの通知配信サーバ。このサーバの各インスタンスは、「1」から始まる通し番号の ID を割り当てられている必要があります。
deliv_comp_d	不可	通知配信完了プロセッサ。
extdbacc	可	外部データベース・アクセッサ。外部データ・ソースごとに最低 1 つは必要です。
genericdb	不可	汎用データベース・アクセッサ。アプリケーションから明示的に呼び出された場合に、アプリケーションからのコンテンツ問い合わせ要求を処理します。これは、One-To-One Command Center でも使用されます。
hostmgr	可	One-To-One 処理に参加するが One-To-One サーバは実行しない各マシンのホスト・マネージャ・プロセス。たとえば、サーバだけが実行されるマシンでは hostmgr が必要です。このリストのサーバのうちの 1 つがすでにあるマシンでは、別の hostmgr は必要ありません。
g1_ofbe_srv	不可	注文処理バックエンド・サーバ。
g1_ofdb	可	注文処理データベース・サーバ。
g1_om_srv	不可	注文管理サーバ。
pmtassign_d	不可	支払いアーカイビング・デーモン。定期的に請求書テーブルを検査し、支払い処理が完了したレコードを探して、そのレコードをアーカイブ・テーブルに移すという方法で、支払いレコードをアーカイブに転送します。
pmthdlr_d	可	各支払い処理メソッド。要求があったときに定期的に認証を取得する、1 つ以上の認証デーモンが必要です。

サーバ	複数インスタンス	説明
pmtsettle_d	可	支払い決済デーモン。決済を要する、関連支払い処理メソッドの注文がないかデータベースを定期的に検査し、その処理を承認します。
sched_poll_d	不可	通知スケジュール・ポーラ。通知を行うタイミングを判断するために、データベース・テーブルを走査します。
sched_srv	可	通知スケジュール・サーバ。訪問者通知メッセージを生成するスクリプトを実行します。

パフォーマンス・カウンタ

各サーバおよびサービスのパフォーマンス・カウンタは、サービスの種類に応じて論理グループに分類されます。

次の項では、グループごとに利用可能なすべてのカウンタを説明します。サービスによっては、同じグループでもカウンタの数が異なる場合もあります。

カウンタ・グループ

- ▶ BV_DB_STAT
- ▶ BV_SRV_CTRL
- ▶ BV_SRV_STAT
- ▶ NS_STAT
- ▶ BV_CACHE_STAT
- ▶ JS_SCRIPT_CTRL
- ▶ JS_SCRIPT_STAT

BV_DB_STAT

データベース・アクセッサ・プロセスには、BV_DB_STAT メモリ・ブロックから取得できる付加的な統計データがあります。これらの統計データは、選択、更新、挿入、削除、およびストアド・プロシージャの実行回数など、データベース・アクセスに関する情報を提供します。

- ▶ DELETE – 削除の実行回数
- ▶ INSERT – 挿入の実行回数

- ▶ SELECT — 選択の実行回数
- ▶ SPROC — ストアド・プロシージャの実行回数
- ▶ UPDATE — 更新の実行回数

BV_SRV_CTRL

- ▶ SHUTDOWN

NS_STAT

NS プロセスは、現在の One-To-One 環境の名前空間を表示します。また、名前空間のオブジェクトを更新させることもできます。

- ▶ Bind
- ▶ List
- ▶ New
- ▶ Rebnd
- ▶ Rsolv
- ▶ Unbnd

BV_SRV_STAT

Interaction Manager プロセスの表示には、現在のセッション数、接続数、アイドル・セッション数、使用中のスレッド数、および処理された CGI 要求の数に関する情報が含まれます。

- ▶ **HOST** — プロセスを実行中のホスト・マシン。
- ▶ **ID** — プロセスのインスタンス（複数のプロセスを **bv1to1.conf** ファイルで設定可能）、または、Interaction Manager のエンジンの ID。
- ▶ **CGI** — 処理された CGI 要求の現在の総数。
- ▶ **CONN** — 接続の現在の総数。
- ▶ **CPU** — このプロセスによって消費された CPU の割合。あるプロセスが CPU 時間の大部分を使用している場合は、そのプロセスをほかのホストに移動するか、場合によってはほかのマシンで動作する追加プロセスを作成することを検討します。これらの指定はどちらも、**bv1to1.conf** ファイルで行います。報告された CPU の割合は、1つのプロセッサに関するものです。プロセッサが4つ搭載されているマシンで、サーバが1つのCPUを占有している場合、この統計値

は 100% となりますが、Windows のタスク・マネージャでは 25% となります。この統計値は、Windows のパフォーマンス・モニタの「% Processor Time」と一致します。

- ▶ **GROUP** – プロセス・グループ (**bv1to1.conf** ファイルで設定)、または、Interaction Manager アプリケーション名。
- ▶ **STIME** – サーバの開始時刻。通常、この開始時間は互いに近い時間です。時間が離れている場合は、サーバがクラッシュし、自動的に再起動されたことを示します。
- ▶ **IDL** – モニタへの要求を除く、受信した IDL 要求の総数。
- ▶ **IdIQ**
- ▶ **JOB**
- ▶ **LWP** – 軽量プロセス (スレッド) の数。
- ▶ **RSS** – サーバ・プロセスの常駐メモリ・サイズ (KB)。
- ▶ **STIME** – システムの開始時刻。
- ▶ **SESS** – 接続の現在の総数。
- ▶ **SYS** – システム・モードでの累積 CPU 使用時間 (秒)。
- ▶ **THR** – スレッドの現在の総数。
- ▶ **USR** – ユーザ・モードでの累積 CPU 使用時間 (秒)。
- ▶ **VSZ** – サーバ・プロセスの仮想メモリ・サイズ (KB)。プロセスのサイズが増大している場合、メモリ・リークが発生している可能性があります。そのプロセスが Interaction Manager プロセスであれば、原因はコンポーネントか動的オブジェクトである可能性が高いといえます (ただし、Interaction Manager サーバは通常の動作として、ガーベジ・コレクションによってサイズが増減します)。

BV_CACHE_STAT

要求キャッシュのステータスを監視します。

各要求について次のカウンタが利用できます。

- ▶ **CNT-Request_Name-HIT** – キャッシュにある要求の総数。
- ▶ **CNT-Request_Name-MAX** – キャッシュの最大サイズ (バイト)。
- ▶ **CNT-Request_Name-SWAP** – キャッシュからスワップされた項目の総数。

- ▶ **CNT-Request_Name-MISS** – キャッシュになかった要求の総数。
- ▶ **CNT-Request_Name-SIZE** – 現在キャッシュにある項目の総数。

キャッシュの測定値

以下の項目について、キャッシュの測定値を利用できます。

- ▶ **AD**
- ▶ **ALERTSCHED** – 通知スケジュールは、BV_ALERTSCHED テーブルと BV_MSGSCHED テーブルで設定します。これは、One-To-One Comamnd Center のユーザが設定するか、またはアプリケーションによって設定されます。
- ▶ **CATEGORY_CONTENT**
- ▶ **DISCUSSION** – One-To-One ディスカッション・グループには、モデレートされたメッセージ・システム、およびトピック別に並べられるメッセージ・スレッドが備わっています。ディスカッション・グループの個々のメッセージを作成、検索、削除するには、Discussion グループ・インタフェースを使用します。ディスカッション・グループを作成、削除、検索するには、汎用のコンテンツ管理 API を使用します。また、BV_DiscussionDB オブジェクトを使用すれば、ディスカッション・グループのデータベースのスレッドとメッセージにアクセスできます。
- ▶ **EXT_FIN_PRODUCT**
- ▶ **EDITORIAL** – Editorials コンテンツ・モジュールを使用すれば、パーソナライズされた編集コンテンツをポイント・キャストおよびコミュニティ・キャストしたり、出版テキストを自社の One-To-One サイトで販売したりできます。また、投資レポートや週刊コラムなどの編集コンテンツを外部のライターや出版社に依頼したり、自社独自の記事、レビュー、レポート、およびその他の情報媒体を作成したりできます。編集コンテンツとしては、テキストのほかに画像、音声、音楽、および映像も使用できます。
- ▶ **INCENTIVE** – 販売奨励金が入ります。
- ▶ **MSGSCHED** – 訪問者メッセージのジョブの指定が入ります。通知スケジュールは、BV_ALERTSCHED テーブルと BV_MSGSCHED テーブルで設定します。これは、One-To-One Comamnd Center のユーザが設定するか、またはアプリケーションによって設定されます。
- ▶ **MSGSCRIPT** – 訪問者のメッセージと警告メッセージを生成する JavaScripts に関する記述が入ります。また、ターゲット・メッセージと警告メッセージを生成する JavaScripts に関する記述が入ります。Command Center を使用し、

Notifications グループの Visitor Messages モジュールを選択することによって、メッセージ・スクリプト情報をこのテーブルに追加します。詳細については、『Command Center User’s Guide』を参照してください。

- ▶ **PRODUCT** – BV_PRODUCT には、訪問者が購入できる製品に関する情報が入ります。
- ▶ **QUERY** – BV_QUERY には、問い合わせが入ります。
- ▶ **SCRIPT** – BV_SCRIPT には、ページ・スクリプトが入ります。
- ▶ **SECURITIES**
- ▶ **TEMPLATE** – Templates コンテンツ・モジュールによって、自社の One-To-One サイトで使用されているあらゆる BroadVision ページ・テンプレートを、コンテンツ・データベースに格納できます。サイト開発者が One-To-One Web サイトを作成する 1 つの方法として、One-To-One Design Center アプリケーションで、BroadVision ページ・テンプレートを BroadVision ダイナミック・オブジェクトと組み合わせるという方法があります。開発者がこれらのページ・テンプレートを使用している場合は、Command Center を使ってページ・テンプレートをコンテンツ・データベースに入れ、管理できます。サイトに BroadVision ページ・テンプレートが使われていない場合は、このコンテンツ・モジュールは使用されません。

JS_SCRIPT_CTRL

- ▶ CACHE
- ▶ DUMP
- ▶ FLUSH
- ▶ METER
- ▶ TRACE

JS_SCRIPT_STAT

- ▶ ALLOC
- ▶ ERROR
- ▶ FAIL
- ▶ JSPERR
- ▶ RELEASE

- ▶ STOP
- ▶ SUCC
- ▶ SYNTAX

ColdFusion モニタの設定

[ColdFusion] ダイアログ・ボックスを使って、ColdFusion サーバを監視するための測定値を選択します。

注：ColdFusion モニタは HTTP を介して動作し、UNIX プラットフォームをサポートします。Windows プラットフォーム上の ColdFusion サーバを監視する場合は、Windows リソース・モニタを使うこともできます。

ColdFusion モニタ環境を設定するには、次の手順で行います。

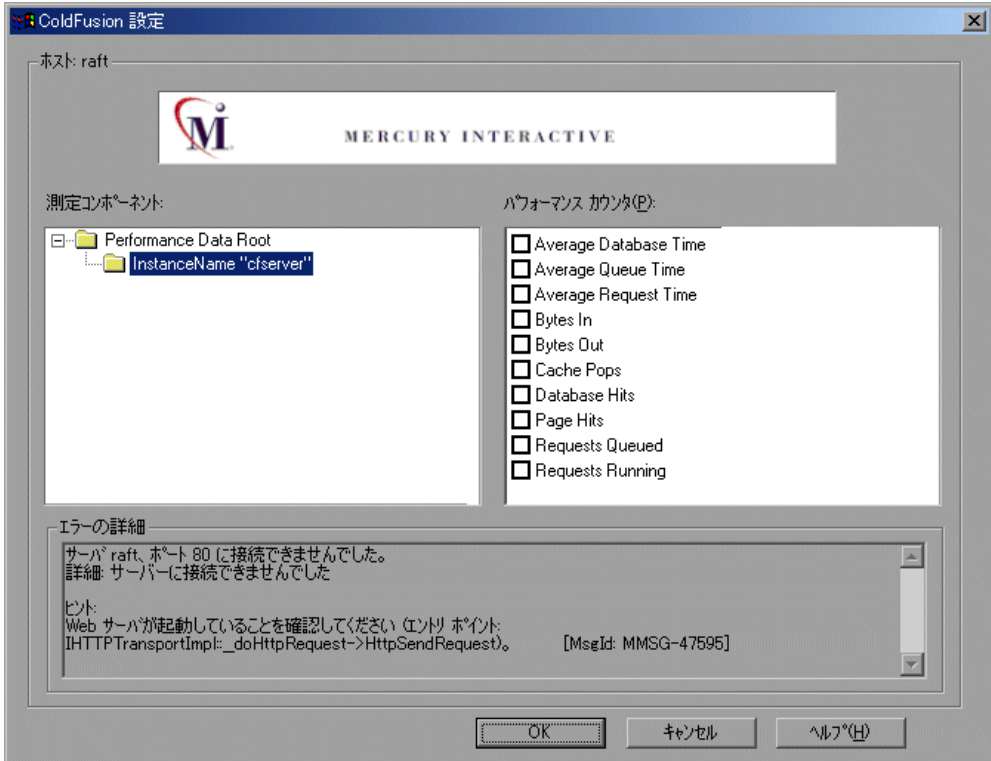
< LoadRunner のインストール先 > %data%monitors%perfmon.cfm ファイルを < ColdFusion のホーム > %cfide%administrator ディレクトリにコピーします。標準設定では、ColdFusion モニタは、< ColdFusion のホーム > %cfide%administrator%perfmon.cfm ファイルの有無を確認します。

注：ファイアウォール越しに ColdFusion サーバを監視するために使用するポートは、サーバの構成によって異なります。

ColdFusion モニタを設定するには、次の手順で行います。

- 1 グラフ・ツリーで [ColdFusion] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [ColdFusion] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

- 4 [ColdFusion] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。[ColdFusion 設定] ダイアログ・ボックスに、使用可能な測定値が表示されます。
- 5 [測定コンポーネント] ツリーが表示されます。



- 6 [ColdFusion 設定] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。
測定可能なパフォーマンス・カウンタの一覧については、413 ページを参照してください。
- 7 [ColdFusion 設定] ダイアログ・ボックスと [ColdFusion] ダイアログ・ボックスで [OK] をクリックすると、ColdFusion モニタが起動します。

ColdFusion パフォーマンス・カウンタ

以下の表に、標準で監視可能なカウンタの説明を示します。

測定値	説明
Avg. Database Time (msec)	ColdFusion がデータベース要求を処理するのに要する時間量の平均（ミリ秒）。
Avg. Queue Time (msec)	ColdFusion が要求の処理を開始する前に、要求が ColdFusion の入力キューで待機していた平均時間（ミリ秒）。
Avg Req Time (msec)	ColdFusion が要求を処理するのに要する合計時間の平均（ミリ秒）。この値には、一般的なページ処理時間に加えて、キュー時間とデータベース処理時間の両方が含まれます。
Bytes In/sec	ColdFusion サーバに送られる秒ごとのバイト数。
Bytes Out/sec	ColdFusion サーバが返す秒ごとのバイト数。
Cache Pops	キャッシュ・ポップ数。
Database Hits/sec	ColdFusion サーバが生成する秒ごとのデータベース・ヒット数。
Page Hits/sec	ColdFusion サーバが処理する秒ごとの Web ページ数。
Queued Requests	現在、ColdFusion サーバによって処理されるのを待機している要求の数。
Running Requests	現在、ColdFusion サーバによってアクティブに処理されている要求の数。
Timed Out Requests	非活動タイムアウトによりタイムアウトとなった要求の数。

Fujitsu INTERSTAGE モニタの設定

Fujitsu INTERSTAGE モニタは、SNMP を使って Fujitsu INTERSTAGE サーバの統計値を取得します。[Fujitsu INTERSTAGE SNMP リソース] ダイアログ・ボックスを使って、Fujitsu INTERSTAGE モニタの測定値を定義します。

Fujitsu INTERSTAGE モニタを設定するには、次の手順で行います。

- 1 グラフ・ツリーで [Fujitsu INTERSTAGE] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Fujitsu INTERSTAGE] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

注：Fujitsu INTERSTAGE SNMP エージェントが動作する SNMP ポートの番号が、標準設定の番号である 161 と異なる場合、ポート番号を定義する必要があります。[マシンの追加] ダイアログ・ボックスに次の情報を入力します。

<サーバ名：ポート番号>

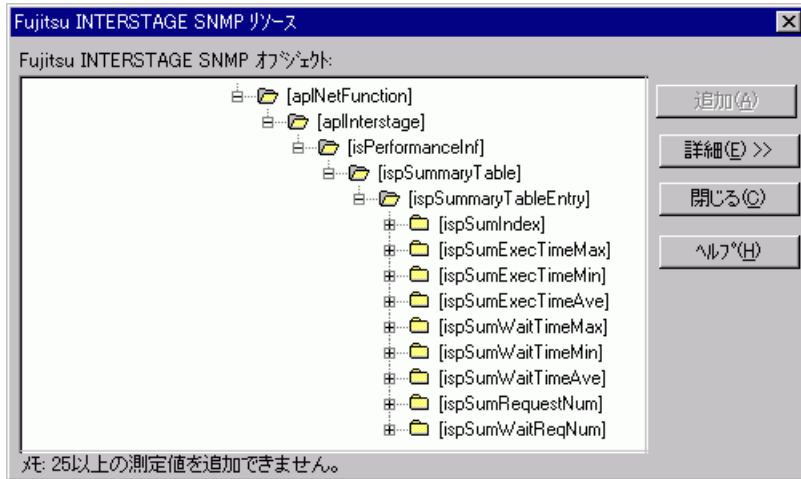
次に例を示します。

digj:8888

また、< LoadRunner のインストール先フォルダ > %dat%monitors にある設定ファイル **snmp.cfg** で、Fujitsu INTERSTAGE サーバの標準設定のポートを定義することもできます。たとえば、Fujitsu INTERSTAGE システムの SNMP エージェントによって使用されているポート番号が 8888 の場合、**snmp.cfg** ファイルを次のように編集します。

```
; Fujitsu INTERSTAGE
[cm_snmp_mon_isp]
port=8888
```

- 4 [Fujitsu INTERSTAGE] ダイアログ・ボックスの [リソース測定場所] セクションの [追加] をクリックします。[Fujitsu INTERSTAGE SNMP リソース] ダイアログ・ボックスが開きます。
- 5 [Fujitsu INTERSTAGE SNMP オブジェクト] ツリーから、監視対象の測定値を選択します。



測定可能なパフォーマンス・カウンタの一覧については、416 ページを参照してください。

- 6 [追加] をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。

注： Fujitsu INTERSTAGE モニタでは測定値を 25 個まで監視できます。

- 7 [Fujitsu INTERSTAGE] 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。

Fujitsu INTERSTAGE パフォーマンス・カウンタ

以下の表に、監視可能な測定値の説明を示します。

測定値	説明
IspSumObjectName	パフォーマンス情報が測定されるアプリケーションのオブジェクト名。
IspSumExecTimeMax	一定時間内におけるアプリケーションの最大処理時間。
IspSumExecTimeMin	一定時間内におけるアプリケーションの最小処理時間。
IspSumExecTimeAve	一定時間内におけるアプリケーションの平均処理時間。
IspSumWaitTimeMax	開始要求の発行後、INTERSTAGE がアプリケーションを開始するまでの最長時間。
IspSumWaitTimeMin	開始要求の発行後、INTERSTAGE がアプリケーションを開始するまでの最短時間。
IspSumWaitTimeAve	開始要求の発行後、INTERSTAGE がアプリケーションを開始するまでの平均時間。
IspSumRequestNum	アプリケーションの開始要求の数。
IspSumWaitReqNum	アプリケーションの起動を待機している要求の数。

iPlanet (NAS) モニタの設定

iPlanet (NAS) モニタは、SNMP を使って iPlanet (NAS) サーバの統計値を取得します。[iPlanet (NAS) リソース] ダイアログ・ボックスを使って、iPlanet (NAS) モニタの測定値を定義します。モニタを設定する前に、SNMP 監視用のアプリケーション・サーバを設定する必要があります。

この項では、iPlanet アプリケーション・サーバでの SNMP 監視の設定について簡単に説明します。この説明は iPlanet のドキュメントを補足するためのもので、代わりになるものではありません。SNMP のレポートイング・アーキテクチャと考え方の説明については、iPlanet のドキュメントを参照してください。

注：次の説明では、SNMP の統計情報を SNMP の標準のポート番号である 161 番のポートを通して収集することを前提としています。

SNMP の概要

- ▶ Solaris にはネイティブな SNMP エージェントである `snmpdx` があります。`snmpdx` は、`/etc/rc3.d/S76snmpdx` スクリプトによってブート時に自動的に起動されます。このデーモンは SNMP の標準の 161 番ポートを使って通信します。ポート番号は、`-p <ポート>` オプションを指定して変更できます。
- ▶ iPlanet 製品には独自の SNMP エージェントが付属しています。ホストごとに、ネットワーク管理ステーションと通信する「マスタ・エージェント」と、各種 iPlanet 製品からデータを収集して、統計情報をマスタ・エージェントに転送する 1 つまたは複数の「サブエージェント」があるという構成になっています。マスタ・エージェントも標準設定では 161 番ポートと通信するようになっています。
- ▶ Solaris の SNMP エージェントと iPlanet の SNMP エージェントの両方を実行するには、プロキシを使用して Sun のエージェントが iPlanet エージェントのサブエージェントに見えるようにする必要があります。

手順の概要

- ▶ システムに root としてログインします。
- ▶ Solaris SNMP エージェントのポート番号を変更します。
- ▶ iPlanet のエージェント「magt」と「sagt」を設定して実行します。
- ▶ Solaris SNMP エージェントを起動します。
- ▶ iPlanet Application Server を SNMP 統計用に構成します。
- ▶ iPlanet Directory Server と iPlanet Web Server (オプション) 用の SNMP サブエージェントを起動します。

Solaris SNMP エージェントのポート番号を変更するには、次の手順で行います。

- 1 システムに root としてログインします (root ユーザだけがポート番号を変更してエージェントを実行できます)。
- 2 `/etc/rc2.d/K76snmpdx stop` を実行して SNMP エージェントを停止します。
- 3 `/etc/rc3.d/S76snmpdx` を編集して、非標準のポート番号で Solaris デーモンを実行します。1161 番に変える場合の例を示します。次のように置き換えます。
`/usr/lib/snmp/snmpdx -y -c /etc/snmp/conf` を、`/usr/lib/snmp/snmpdx -p 1161 -y -c /etc/snmp/conf` に置き換えます。

iPlanet のエージェント「magt」と「sagt」を設定して実行するには、次の手順で行います。

マスタ・エージェント、プロキシ・エージェント、およびスタートアップ・スクリプトは < ias のインストール先ディレクトリ > /snmp にあります。

- 1 `S75snmpagt` スクリプトで、環境変数 `GX_ROOTDIR` に 1 行追加して iAS のインストール先を指すようにします。たとえば、iPlanet Application Server が `/usr/iplanet/ias6/ias` にインストールされている場合には、次のように追加します。

```
GX_ROOTDIR=/usr/iplanet/ias6/iasexprt GX_ROOTDIR
```

- 2 スクリプト `S75snmpagt` を `/etc/rc3.d` にコピーします。
- 3 `chmod 755 /etc/rc3.d/S75snmpagt`
- 4 `In /etc/rc3.d/S75snmpagt /etc/rc2.d/K07snmpagt`
- 5 システム情報とトラップの設定ができます。

次の例では、システムの所有者と場所に関する情報を追加し、SNMP トラップをネットワーク管理ステーション (mde.uk.sun.com) に送信するようにしています。

```
COMMUNITY public
ALLOW ALL OPERATIONS
INITIAL sysLocation "Under Joe Bloggs' Desk in Headquarters"
INITIAL sysContact "Joe Bloggs
Email: Joe.Bloggs@Sun.COM
Voice: +1 650 555 1212"
MANAGER mde.uk.sun.com
SEND ALL TRAPS TO PORT 162
WITH COMMUNITY public
```

注：プロキシ・エージェントの設定ファイル (CONFIG_SAGT) を編集する必要はありません。

- 6 次のコマンドを実行して iPlanet エージェントを起動します。
- ```
/etc/rc3.d/S75snmpagt start
```

**Solaris の SNMP エージェントを起動するには、次の手順で行います。**

次のコマンドを実行して Solaris SNMP エージェントを再起動します。

```
/etc/rc3.d/S76snmpdx start
```

**iPlanet Application Server を SNMP 統計用に設定するには、次の手順で行います。**

- 1 iPlanet Application Server の管理者ツール **ksvradmin** を起動します。
- 2 [General View] で、管理対象とするインスタンスの名前を選択します。
- 3 管理フレームの [SNMP] タブをクリックします。
- 4 [Enable SNMP Administration and Monitoring] および [Enable SNMP Debug] を選択します。
- 5 [Connection Attempt Interval] フィールドに「60」と入力して、**ksvradmin** を終了します。

- 6 次のコマンドで iPlanet Application Server を再起動します。

```
iascontrol stop
iascontrol kill
iascontrol start
```

- 7 ログ・ファイル < **ias** のインストール先ディレクトリ > /logs/ias.log を調べて、アプリケーション・サーバが正常にマスタ・エージェントに接続したことを確認します。次の行があるはずです。  
kas> SNMP: Connected to master agent

**iPlanet Web サーバ用の SNMP サブエージェントを起動するには、次の手順で行います。**

- 1 Web ブラウザを使って iPlanet Web サーバにアクセスします。
- 2 管理の対象にする Web サーバを選択して、[**Manage**] ボタンをクリックします。
- 3 [**Monitor**] タブを選択し、ページの左側にある [**SNMP Subagent Configuration**] をクリックします。
- 4 設定情報を入力して、[**Enable SNMP Statistics Collection**] ラジオ・ボタンをオンにします。
- 5 [**SNMP Subagent Control**] をクリックします。
- 6 [**Start**] ボタンをクリックします。

**iPlanet Directory Server 用の SNMP サブエージェントを起動するには、次の手順で行います。**

- 1 Netscape Administration Console を使用して iPlanet サーバを管理します。
- 2 [**Configuration**] タブを選択します。
- 3 [Configuration] フレームの [**SNMP**] タブをクリックします。
- 4 [**Enable statistics collection**] チェック・ボックスを選択します。
- 5 "Master Host" を "localhost" に設定します。
- 6 "Master port" を 199 に設定します。
- 7 他のフィールドに適切な情報を入力します。
- 8 [**Start Subagent**] ボタンをクリックします。

## 要約

SNMP 管理ツールを使用して 161 番ポートを通じて SNMP マスタ・エージェントに問い合わせをします。Solaris SNMP エージェントのほか、設定した iPlanet サブエージェントによるすべての情報が表示されます。

ブート・スクリプトを設定したので、次回 Solaris をブートするときは、Sun と iPlanet の SNMP エージェントは自動的に起動されます。

## コントローラでの iPlanet (NAS) モニタの設定

iPlanet の SNMP サービスを設定したら、iPlanet (NAS) モニタで測定するカウンタを選択する必要があります。測定値は、[iPlanet (NAS) リソース] ダイアログ・ボックスで選択します。

**iPlanet (NAS) リソース・モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [iPlanet (NAS)] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [iPlanet (NAS)] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

注：iPlanet SNMP エージェントを実行しているポートの番号が SNMP の標準のポート番号と異なる場合には、ポート番号を定義する必要があります。[マシンの追加] ダイアログ・ボックスに、次の情報を入力します。

<サーバ名：ポート番号>

次に例を示します。

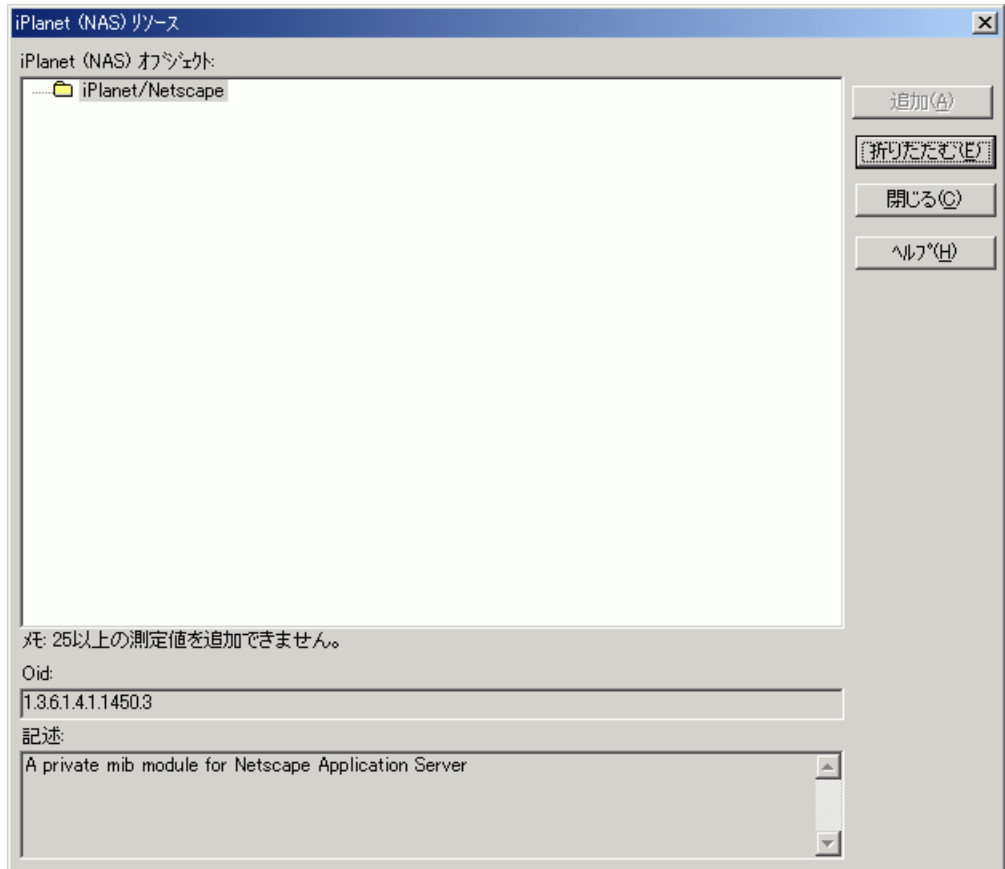
dig:8888

また、< LoadRunner のインストール先フォルダ > %dat%monitors にある設定ファイル **snmp.cfg** で、iPlanet サーバの標準設定のポートを定義することもできます。たとえば、iPlanet サーバで SNMP エージェントによって使用されているポート番号が 8888 の場合、**snmp.cfg** ファイルを次のように編集します。

```
; iPlanet (NAS)
[cm_snmp_mon_nas]
port=8888
```

- 
- 4 [iPlanet (NAS)] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。

[iPlanet (NAS) リソース] ダイアログ・ボックスが開きます。



- 5 [iPlanet (NAS) オブジェクト] ツリーを参照します。

測定可能なパフォーマンス・カウンタの一覧については、424 ページを参照してください。

- 6 オブジェクトを測定するには、対象のオブジェクトを選択して [追加] をクリックします。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。

---

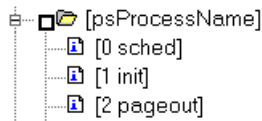
注：iPlanet（NAS）モニタでは測定値を 25 個まで監視できます。

---

- 7 [iPlanet（NAS）] 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。

---

注：iPlanet（NAS）モニタの測定情報のレベルを上げるには、（測定項目の数値だけでなく）測定項目の文字列値も表示されるようにし、（文字列値を測定項目の識別名として表示する）名前修飾子を有効にします。名前修飾子を使用している次の測定値の例では、ProcessName の文字列値（sched）がそのインスタンス ID（0）のほかに表示されます。



```
[-] [psProcessName]
 [0 sched]
 [1 init]
 [2 pageout]
```

この機能を有効にするには、次の行を < LoadRunner のインストール先フォルダ > %dat%monitors%snmp.cfg ファイルに追加します。

SNMP\_show\_string\_nodes=1

使用上の注意：名前修飾子は複数選択できますが、使用されるのは、階層の先頭の修飾子です。[iPlanet（NAS）リソース] ダイアログ・ボックスが開くたびに、snmp.cfg ファイルから情報が読み込み直されます。「名前修飾子付き」と「名前修飾子なし」という具合に同じ測定項目を 2 度追加することはできません。2 度追加すると、エラー・メッセージが表示されます。

---

## iPlanet（NAS）パフォーマンス・カウンタ

以下の表に、監視可能なカウンタの説明を示します。



## Netscape パフォーマンス・カウンタ

| 測定値                          | 説明                         |
|------------------------------|----------------------------|
| <b>nasKesEngConnRetries</b>  | 管理サーバがエンジンに接続を試みる最大回数。     |
| <b>nasKesEngMaxRestart</b>   | 失敗の後、管理サーバがエンジンを再起動する最大回数。 |
| <b>nasKesEngAutoStart</b>    | 管理サーバの起動時にすべてのエンジンを起動します。  |
| <b>nasKesConfigHeartBeat</b> | 活動通知。                      |

## KES パフォーマンス・カウンタ

| 測定値                               | 説明                                                                                                                          |
|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| <b>nasKesId</b>                   | このエンジンが属している KES の ID。                                                                                                      |
| <b>nasKesMinThread</b>            | エンジンごとのスレッドの標準の最少数。                                                                                                         |
| <b>nasKesMaxThread</b>            | エンジンごとのスレッドの標準の最大数。                                                                                                         |
| <b>nasKesLoadBalancer Disable</b> | 負荷バランス・サービスを有効または無効にします。                                                                                                    |
| <b>nasKesCpuLoad</b>              | このホストでの CPU 総使用量。                                                                                                           |
| <b>nasKesDiskLoad</b>             | このホストでのディスク総使用量。                                                                                                            |
| <b>nasKesMemLoad</b>              | このホストでのメモリ総使用量。                                                                                                             |
| <b>nasKesRequestLoad</b>          | この NAS に対する要求の数。                                                                                                            |
| <b>nasKesCpuLoadFactor</b>        | サーバ負荷を計算する際の CPU 使用量の相対的重要度。この数字は、パーセント単位で指定します。サーバ負荷の係数 (CPUload, DiskLoad, MemLoad, および ExecReqs) の合計は、100% にならなくてはなりません。 |
| <b>nasKesDiskLoadFactor</b>       | サーバ負荷を計算する際のディスク使用量の相対的重要度。この数字は、パーセント単位で指定します。サーバ負荷の係数 (CPUload, DiskLoad, MemLoad, および ExecReqs) の合計は、100% にならなくてはなりません。  |
| <b>nasKesMemLoadFactor</b>        | サーバ負荷を計算する際のメモリ使用量の相対的重要度。この数字は、パーセント単位で指定します。サーバ負荷の係数 (CPUload, DiskLoad, MemLoad, および ExecReqs) の合計は、100% にならなくてはなりません。   |

| 測定値                                     | 説明                                                                                                                                                                       |
|-----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>nasKesAppLogics RunningFactor</b>    | AppLogic の実行パフォーマンスを計算する際の、AppLogic が実行される回数の相対的重要度。この数字は、パーセント単位で指定します。エージェント負荷の係数 (ResultCached, AvgExecTime, LastExecTime, および ServerLoad) の合計は、100% にならなくてはなりません。    |
| <b>nasKesResultsCached Factor</b>       | AppLogic の実行パフォーマンスを計算する際の、AppLogic がキャッシュされた結果の相対的重要度。この数字は、パーセント単位で指定します。エージェント負荷の係数 (ResultCached, AvgExecTime, LastExecTime, および ServerLoad) の合計は、100% にならなくてはなりません。 |
| <b>nasKesAvgExecTime Factor</b>         | AppLogic の実行パフォーマンスを計算する際の、AppLogic の平均実行時間の相対的重要度。この数字は、パーセント単位で指定します。エージェント負荷の係数 (ResultCached, AvgExecTime, LastExecTime, および ServerLoad) の合計は、100% にならなくてはなりません。     |
| <b>nasKesLastExecTime Factor</b>        | AppLogic の実行パフォーマンスを計算する際の、AppLogic の最後の実行時間の相対的重要度。この数字は、パーセント単位で指定します。エージェント負荷の係数 (ResultCached, AvgExecTime, LastExecTime, および ServerLoad) の合計は、100% にならなくてはなりません。    |
| <b>nasKesHitsFactor</b>                 | AppLogic の実行パフォーマンスを計算する際の、AppLogic 実行の数の相対的重要度。この数字は、パーセント単位で指定します。エージェント負荷の係数 (ResultCached, AvgExecTime, LastExecTime, および ServerLoad) の合計は、100% にならなくてはなりません。        |
| <b>nasKesServerLoadFactor</b>           | AppLogic 実行パフォーマンスを計算する際の、サーバ負荷 (4つのサーバ負荷係数を使用して計算される) の相対的重要度。エージェント負荷の係数 (ResultCached, AvgExecTime, LastExecTime, および ServerLoad) の合計は、100% にならなくてはなりません。             |
| <b>nasKesBroadcastInterval</b>          | ロードバランサ・デーモンから試みられるブロードキャストの間隔 (秒)。                                                                                                                                      |
| <b>nasKesApplogic BroadcastInterval</b> | クラスタ内のすべてのサーバをまたぐ AppLogic 負荷情報のブロードキャストの間隔 (秒)。これは nasKesBroacastInterval よりも大きくなくてはなりません。                                                                              |

| 測定値                                   | 説明                                                                                           |
|---------------------------------------|----------------------------------------------------------------------------------------------|
| <b>nasKesServerBroadcastInterval</b>  | クラスタ内のすべてのサーバをまたぐサーバ負荷情報の各ブロードキャストの間隔 (秒)。これは <b>nasKesBroacastInterval</b> よりも大きくなくてはなりません。 |
| <b>nasKesServerLoadUpdateInterval</b> | サーバ負荷情報の更新の間隔 (秒)。サーバ負荷の更新には、更新が行われるときまでにサンプリングされたサーバ負荷データが適用されます。                           |
| <b>nasKesCpuLoadUpdateInterval</b>    | CPU 使用量のサンプリングの間隔 (秒)。                                                                       |
| <b>nasKesDiskLoadUpdateInterval</b>   | ディスク使用量のサンプリングの間隔 (秒)。                                                                       |
| <b>nasKesMemLoadUpdateInterval</b>    | メモリ・スラッシュのサンプリングの間隔 (秒)。                                                                     |
| <b>nasKesTotalReqsUpdateInterval</b>  | 要求数のサンプリングの間隔 (秒)。                                                                           |
| <b>nasKesMaxHops</b>                  | 要求の負荷を別のサーバに分散できる最大回数。                                                                       |
| <b>nasKesODBCReqMinThread</b>         | 非同期の要求を処理するために用意されているスレッドの最小数。                                                               |
| <b>nasKesODBCReqMaxThread</b>         | 非同期の要求を処理するために用意されているスレッドの最大数。                                                               |
| <b>nasKesODBCCacheMaxConns</b>        | NAS とデータベース間で開いている接続の最大数。                                                                    |
| <b>nasKesODBCCacheFreeSlots</b>       | NAS とデータベース間で確立されているキャッシュされた接続の最大数。                                                          |
| <b>nasKesODBCCacheTimeout</b>         | アイドル状態の接続が切断されるまでの時間。                                                                        |
| <b>nasKesODBCCacheInterval</b>        | キャッシュ・クリーナが、指定されたタイムアウトよりも長い間アイドル状態にある接続を切断しようと試みる間隔 (秒)。                                    |
| <b>nasKesODBCConnGiveupTime</b>       | ドライバがデータベースへの接続を試みる最大回数。                                                                     |
| <b>nasKesODBCCacheDebug</b>           | 接続キャッシュのデバッグ情報を有効にします。                                                                       |

| 測定値                                     | 説明                                                        |
|-----------------------------------------|-----------------------------------------------------------|
| <b>nasKesODBCResultSet<br/>InitRows</b> | データベースから一度に取得される行数。                                       |
| <b>nasKesODBCResultSet<br/>MaxRows</b>  | キャッシュされた結果セットに含めることができる最大行数。                              |
| <b>nasKesODBCResultSet<br/>MaxSize</b>  | ドライバがキャッシュする結果セットの最大サイズ。                                  |
| <b>nasKesODBCSqlDebug</b>               | SQL デバッグ情報を有効にします。                                        |
| <b>nasKesODBCEnable<br/>Parser</b>      | SQL 解析を有効にします。                                            |
| <b>nasKesORCLReqMin<br/>Thread</b>      | 非同期の要求を処理するために用意されているスレッドの最小数。                            |
| <b>nasKesORCLReqMax<br/>Thread</b>      | 非同期の要求を処理するために用意されているスレッドの最大数。                            |
| <b>nasKesORCLCache<br/>MaxConns</b>     | NAS とデータベース間で開いている接続の最大数。                                 |
| <b>nasKesORCLCache<br/>FreeSlots</b>    | NAS とデータベース間で確立されているキャッシュされた接続の最大数。                       |
| <b>nasKesORCLCache<br/>Timeout</b>      | アイドル状態の接続が切断されるまでの時間。                                     |
| <b>nasKesORCLCache<br/>Interval</b>     | キャッシュ・クリーナが、指定されたタイムアウトよりも長い間アイドル状態にある接続を切断しようと試みる間隔 (秒)。 |
| <b>nasKesORCLConn<br/>GiveupTime</b>    | ドライバが Oracle への接続を試みる最大回数。                                |
| <b>nasKesORCLCache<br/>Debug</b>        | 接続キャッシュのデバッグ情報を有効にします。                                    |
| <b>nasKesORCLResultSet<br/>InitRows</b> | データベースから一度に取得される行数。                                       |
| <b>nasKesORCLResultSet<br/>MaxRows</b>  | キャッシュされた結果セットに含めることができる最大行数。                              |
| <b>nasKesORCLResultSet<br/>MaxSize</b>  | ドライバがキャッシュする結果セットの最大サイズ。                                  |

| 測定値                                | 説明                                  |
|------------------------------------|-------------------------------------|
| <b>nasKesORCLSqlDebug</b>          | SQL デバッグ情報を有効にします。                  |
| <b>nasKesSYBReqMin Thread</b>      | 非同期の要求を処理するために用意されているスレッドの最小数。      |
| <b>nasKesSYBReqMax Thread</b>      | 非同期の要求を処理するために用意されているスレッドの最大数。      |
| <b>nasKesSYBCacheMax Conns</b>     | NAS とデータベース間で開いている接続の最大数。           |
| <b>nasKesSYBCacheFree Slots</b>    | NAS とデータベース間で確立されているキャッシュされた接続の最大数。 |
| <b>nasKesSYBCache Timeout</b>      | アイドル状態の接続が切断されるまでの時間。               |
| <b>nasKesSYBCache Interval</b>     | キャッシュされた接続どうしの間隔。                   |
| <b>nasKesSYBConn GiveupTime</b>    | ドライバが Sybase への接続を試みる最大回数。          |
| <b>nasKesSYBCacheDebug</b>         | 接続キャッシュのデバッグ情報を有効にします。              |
| <b>nasKesSYBResultSet InitRows</b> | データベースから一度に取得される行数。                 |
| <b>nasKesSYBResultSet MaxRows</b>  | キャッシュされた結果セットに含めることができる最大行数。        |
| <b>nasKesSYBResultSet MaxSize</b>  | ドライバがキャッシュする結果セットの最大サイズ。            |

## エンジン・パフォーマンス・カウンタ

| 測定値                      | 説明                                                                                  |
|--------------------------|-------------------------------------------------------------------------------------|
| <b>nasEngKesPort</b>     | このエンジンがサービスを提供する KXS のポート。これはオブジェクト ID の一部として提供されるため、作成後の変更はできません。                  |
| <b>nasEngPort</b>        | このエンジンがリスンしている TCP/IP ポート。このポートは、エンジンの作成時にだけ指定できます。これを変更することはできません。                 |
| <b>nasEngType</b>        | エンジンの種類。executive (0), Java (1000), C++ (3000) のいずれか。                               |
| <b>nasEngId</b>          | ID は、0 から 1 つずつ増えていきます。ID は変更できません。                                                 |
| <b>nasEngName</b>        | このエンジンの名前。これは、kes, kxs, または kjs を含む情報文字列です。                                         |
| <b>nasEngNewConsole</b>  | 新しいコンソール・ウィンドウで各エンジンを起動します。                                                         |
| <b>nasEngStatus</b>      | エンジンの追加、削除、有効/無効の切り替えに使用するステータス・カラム。エンジンを作成するには、1 つ設定する必要があります。これは、RFC 1443 に準拠します。 |
| <b>nasEngMinThread</b>   | エンジンごとのスレッドの標準の最少数。                                                                 |
| <b>nasEngMaxThread</b>   | エンジンごとのスレッドの標準の最大数。                                                                 |
| <b>nasEngReqRate</b>     | 要求が到着する割合。                                                                          |
| <b>nasEngTotalReq</b>    | エンジンの起動以来処理された要求の総数。                                                                |
| <b>nasEngReqNow</b>      | 処理中の要求の数。                                                                           |
| <b>nasEngReqWait</b>     | サービスの提供を待機している要求。                                                                   |
| <b>nasEngReqReady</b>    | サービスを受ける準備ができていない要求。                                                                |
| <b>nasEngAvgReqTime</b>  | 要求の平均処理時間。                                                                          |
| <b>nasEngThreadNow</b>   | 要求マネージャが使用するスレッドの数。                                                                 |
| <b>nasEngThreadWait</b>  | アイドル状態のスレッドの数。                                                                      |
| <b>nasEngWebReqQueue</b> | キューで待機している Web 要求の数。                                                                |

| 測定値                                 | 説明                           |
|-------------------------------------|------------------------------|
| <b>nasEngFailedReq</b>              | 失敗した要求の数。                    |
| <b>nasEngTotalConn</b>              | 開いている接続の総数。                  |
| <b>nasEngTotalConnNow</b>           | 使用中の接続の総数。                   |
| <b>nasEngTotalAccept</b>            | 受信要求をリスンしている接続の総数。           |
| <b>nasEngTotalAcceptNow</b>         | 使用中の受信要求をリスンしている接続の総数。       |
| <b>nasEngTotalSent</b>              | 送信されたパケットの総数。                |
| <b>nasEngTotalSentBytes</b>         | 送信された総バイト数。                  |
| <b>nasEngTotalRecv</b>              | 受信されたパケットの総数。                |
| <b>nasEngTotalRecvBytes</b>         | 受信した総バイト数。                   |
| <b>nasEngBindTotal</b>              | 起動後にバインドされた AppLogic の数。     |
| <b>nasEngBindTotalCached</b>        | 起動後にキャッシュされた AppLogic の数。    |
| <b>nasEngTotalThreads</b>           | このプロセスで作成されたスレッドの総数。         |
| <b>nasEngCurrentThreads</b>         | このプロセスで使用中のスレッドの総数。          |
| <b>nasEngSleepingThreads</b>        | このプロセスでスリープ状態のスレッドの数。        |
| <b>nasEngDAETotalQuery</b>          | 起動後に実行されたクエリーの総数。            |
| <b>nasEngDAEQueryNow</b>            | 処理中のクエリーの数。                  |
| <b>nasEngDAETotalConn</b>           | 起動後に作成された論理接続の数。             |
| <b>nasEngDAEConnNow</b>             | 使用中の論理接続の数。                  |
| <b>nasEngDAECacheCount</b>          | キャッシュの数。                     |
| <b>nasEngODBCQueryTotal</b>         | 起動後に実行されたクエリーの総数。            |
| <b>nasEngODBCPreparedQueryTotal</b> | 起動後に実行された ODBC が準備したクエリーの総数。 |
| <b>nasEngODBCConnTotal</b>          | 起動後に開かれた接続の総数。               |
| <b>nasEngODBCConnNow</b>            | 現在開いている接続の数。                 |
| <b>nasEngORCLQueryTotal</b>         | 起動後に実行されたクエリーの総数。            |

| 測定値                                 | 説明                          |
|-------------------------------------|-----------------------------|
| <b>nasEngORCLPreparedQueryTotal</b> | 起動後に実行された準備されているクエリーの総数。    |
| <b>nasEngORCLConnTotal</b>          | 起動後に Oracle との間で確立された接続の総数。 |
| <b>nasEngORCLConnNow</b>            | 現在 Oracle との間で開いている接続の数。    |
| <b>nasEngSYBQueryTotal</b>          | 起動後にドライバによって処理されたクエリーの総数。   |
| <b>nasEngSYBPreparedQueryTotal</b>  | 起動後に処理された準備済みのクエリーの総数。      |
| <b>nasEngSYBConnTotal</b>           | 起動後に開かれた接続の総数。              |
| <b>nasEngSYBConnNow</b>             | 現在開いている SYB 接続の数。           |
| <b>nasStatusTrapEntry</b>           | KES の定義。                    |
| <b>nasTrapKesIpAddress</b>          | KES ホストの IP アドレス。           |
| <b>nasTrapKesPort</b>               | この NAS の主エンジンのポート。          |
| <b>nasTrapEngPort</b>               | このイベントを生成しているエンジンのポート。      |
| <b>nasTrapEngState</b>              | このイベントを生成しているエンジンのポート。      |



## MS Active Server Pages モニタの設定

[MS Active Server Pages] ダイアログ・ボックスを使って、Microsoft ASP アプリケーション・サーバを監視するための測定値を選択します。

---

**注：**ファイアウォール越しに ASP サーバを監視するには、139 番ポートで TCP を使用します。

---

**ASP モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [MS Active Server Pages] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [MS Active Server Pages] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [MS Active Server Pages] ダイアログ・ボックスの [リソース測定場所] セクションで、監視対象の測定値を選択します。

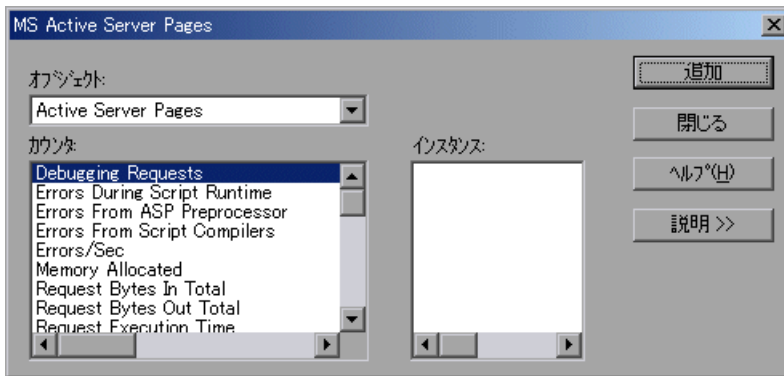
測定可能なパフォーマンス・カウンタの一覧については、434 ページを参照してください。

---

**注：**Microsoft ASP モニタの標準設定のカウンタを変更するには、669 ページ「モニタの標準設定のカウンタの変更」を参照してください。

---

- 追加の測定値を選択するには、[追加] をクリックします。ダイアログ・ボックスが開き、Active Server Pages オブジェクト、そのカウンタ、およびインスタンスが表示されます。



- カウンタとインスタンスを選択します。**Ctrl** キーを使って、複数のカウンタを選択できます。インスタンスが意味を持つのは、強調表示されたカウンタの複数のインスタンスが実行されている場合だけです。各カウンタの説明を表示するには、[説明 >>] をクリックしてダイアログ・ボックスを拡張します。
- [追加] をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。
- [MS Active Server Pages] 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。

### MS Active Server Pages パフォーマンス・カウンタ

以下の表に、標準で監視可能なカウンタの説明を示します。

| 測定値                | 説明                                 |
|--------------------|------------------------------------|
| Errors per Second  | 秒ごとのエラー数。                          |
| Requests Wait Time | 直前の要求がキューで待機していた時間（ミリ秒）。           |
| Requests Executing | 現在実行している要求の数。                      |
| Requests Queued    | キューでサービスを待機している要求の数。               |
| Requests Rejected  | 処理に必要なリソースが不足していたために実行されなかった要求の総数。 |

| 測定値                                  | 説明                                               |
|--------------------------------------|--------------------------------------------------|
| <b>Requests Not Found</b>            | ファイルが見つからなかった要求の数。                               |
| <b>Requests/sec</b>                  | 秒ごとの実行要求数。                                       |
| <b>Memory Allocated</b>              | Active Server Pages によって現在割り当てられているメモリの総量 (バイト)。 |
| <b>Errors During Script Run-Time</b> | 実行時エラーにより失敗した要求の数。                               |
| <b>Sessions Current</b>              | サービスされているセッションの数。                                |
| <b>Transactions/sec</b>              | 開始したトランザクションの秒ごとの数。                              |

## Oracle9iAS HTTP モニタの設定

[Oracle HTTP Server Monitor 設定] ダイアログ・ボックスを使って、Oracle9iAS HTTP サーバを監視するための測定値を選択します。監視する測定値を選択する前に Oracle 9iAS HTTP サーバを起動します。

---

注：ファイアウォール越しに Oracle9iAS HTTP サーバを監視するために使用するポートは、サーバの構成によって異なります。

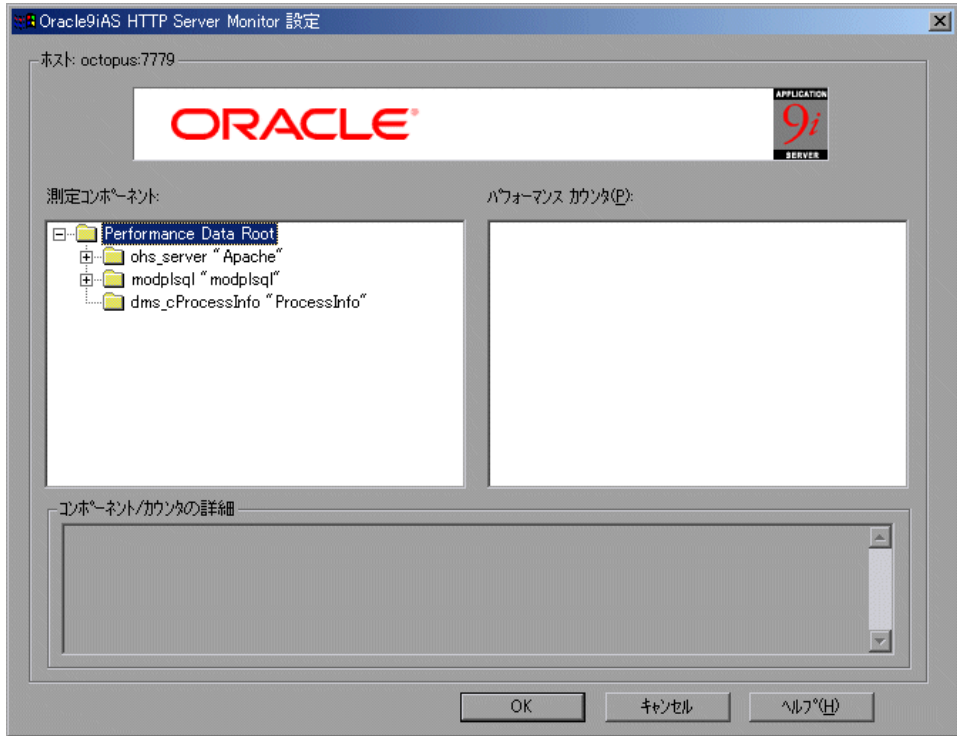
---

**Oracle9iAS HTTP モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [Oracle9iAS HTTP] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Oracle9iAS HTTP] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。任意のプラットフォームを選択して、[OK] をクリックします。
- 4 [Oracle9iAS HTTP Server] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。[Oracle

**HTTP Server Monitor 設定** ダイアログ・ボックスが開き、監視可能なカウンタが表示されます。

- 5 [測定コンポーネント] ツリーが表示されます。



- 6 [Oracle HTTP Server Monitor 設定] ウィンドウの右の表示枠で、必要なマシン処理カウンタまたはアプリケーション・サーバ・パフォーマンス・カウンタを選択します。

測定可能なパフォーマンス・カウンタの一覧については、437 ページを参照してください。

- 7 [Oracle HTTP Server Monitor 設定] ダイアログ・ボックスと [Oracle9iAS HTTP Server] ダイアログ・ボックスで [OK] をクリックすると、Oracle9iAS HTTP モニタが起動します。

## Oracle9iAS HTTP サーバ・パフォーマンスカウンタ

次の表に、監視可能ないくつかのモジュールの説明を示します。

| 測定値                      | 説明                                         |
|--------------------------|--------------------------------------------|
| <b>mod_mime.c</b>        | ファイルの拡張子によってドキュメントの形式を判定します。               |
| <b>mod_mime_magic.c</b>  | 「マジック・ナンバー」によってドキュメントの形式を判定します。            |
| <b>mod_auth_anon.c</b>   | 認証が必要な領域に匿名ユーザ・アクセスできるようにします。              |
| <b>mod_auth_dbm.c</b>    | DBM ファイルを使用して、ユーザ認証を行います。                  |
| <b>mod_auth_digest.c</b> | MD5 認証を提供します。                              |
| <b>mod_cern_meta.c</b>   | HTTP ヘッダ・メタファイルをサポートします。                   |
| <b>mod_digest.c</b>      | MD5 認証を提供します (mod_auth_digest の登場により使用廃止)。 |
| <b>mod_expires.c</b>     | Expires: ヘッダをリソースに適用します。                   |
| <b>mod_headers.c</b>     | 指定した HTTP ヘッダをリソースに追加します。                  |
| <b>mod_proxy.c</b>       | プロキシ・キャッシュ機能を提供します。                        |
| <b>mod_rewrite.c</b>     | 正規表現を使用して URI からファイル名へのマッピングを行います。         |
| <b>mod_speling.c</b>     | URL のスペルミスを自動修正します。                        |
| <b>mod_info.c</b>        | サーバ設定情報を提供します。                             |
| <b>mod_status.c</b>      | サーバの状態を表示します。                              |
| <b>mod_usertrack.c</b>   | クッキーを使用してユーザを追跡できるようにします。                  |
| <b>mod_dms.c</b>         | DMS Apache 統計にアクセスできるようにします。               |
| <b>mod_perl.c</b>        | Perl スクリプトを実行します。                          |
| <b>mod_fastcgi.c</b>     | 長期実行プログラムへの CGI アクセスをサポートします。              |
| <b>mod_ssl.c</b>         | SSL サポートを提供します。                            |
| <b>mod_plsql.c</b>       | Oracle ストアド・プロシージャに対する要求を処理します。            |

| 測定値                      | 説明                                                                |
|--------------------------|-------------------------------------------------------------------|
| <b>mod_isapi.c</b>       | Windows ISAPI 拡張子サポートを提供します。                                      |
| <b>mod_setenvif.c</b>    | クライアント情報に基づいて環境変数を設定します。                                          |
| <b>mod_actions.c</b>     | メディアの種類や要求メソッドに基づいて CGI を実行します。                                   |
| <b>mod_imap.c</b>        | イメージ・マップ・ファイルを処理します。                                              |
| <b>mod_asis.c</b>        | 固有の HTTP ヘッダを含むファイルを送信します。                                        |
| <b>mod_log_config.c</b>  | mod_log_common の代わりとなるユーザ定義ログ機能を提供します。                            |
| <b>mod_env.c</b>         | CGI スクリプトに環境を渡します。                                                |
| <b>mod_alias.c</b>       | ドキュメント・ツリー内のホスト・ファイル・システムの項目を割り当て、URL をリダイレクトします。                 |
| <b>mod_userdir.c</b>     | ユーザのホーム・ディレクトリを処理します。                                             |
| <b>mod_cgi.c</b>         | CGI スクリプトを起動します。                                                  |
| <b>mod_dir.c</b>         | 基本ディレクトリを処理します。                                                   |
| <b>mod_autoindex.c</b>   | ディレクトリの一覧を自動的に取得します。                                              |
| <b>mod_include.c</b>     | サーバ解析ドキュメントを取得します。                                                |
| <b>mod_negotiation.c</b> | コンテンツ・ネゴシエーションを処理します。                                             |
| <b>mod_auth.c</b>        | テキスト・ファイルを使用してユーザ認証を行います。                                         |
| <b>mod_access.c</b>      | クライアントのホスト名または IP アドレスに基づいたアクセス制御を提供します。                          |
| <b>mod_so.c</b>          | モジュールの実行時の読み込みをサポートします (UNIX では .so, Win32 では .dll)。              |
| <b>mod_oprocmgr.c</b>    | JServ プロセスを監視し、このプロセスが失敗すると再起動します。                                |
| <b>mod_jserv.c</b>       | HTTP 要求を JServ サーバにルートします。巡回方式で新しい要求を複数の JServ に分散して負荷のバランスを取ります。 |

| 測定値                | 説明                                           |
|--------------------|----------------------------------------------|
| <b>mod_ose.c</b>   | 要求を Oracle のデータベース・サーバに組み込まれている JVM にルートします。 |
| <b>http_core.c</b> | 静的 Web ページに対する要求を処理します。                      |

以下の表に、監視可能なカウンタの説明を示します。

| 測定値                         | 説明                                |
|-----------------------------|-----------------------------------|
| <b>handle.minTime</b>       | モジュール・ハンドラでの最短処理時間。               |
| <b>handle.avg</b>           | モジュール・ハンドラでの平均処理時間。               |
| <b>handle.active</b>        | 現在ハンドル処理されているスレッド数。               |
| <b>handle.time</b>          | モジュール・ハンドラでの処理時間の合計。              |
| <b>handle.completed</b>     | ハンドル処理が完了した回数。                    |
| <b>request.maxTime</b>      | HTTP 要求のサービスに要する最長時間。             |
| <b>request.minTime</b>      | HTTP 要求のサービスに要する最短時間。             |
| <b>request.avg</b>          | HTTP 要求のサービスに要する平均時間。             |
| <b>request.active</b>       | 現在、要求処理の段階にあるスレッド数。               |
| <b>request.time</b>         | HTTP 要求のサービスに要する時間の総計。            |
| <b>request.completed</b>    | 要求処理が完了した回数。                      |
| <b>connection.maxTime</b>   | 任意の HTTP 接続サービスにかかる最長時間。          |
| <b>connection.minTime</b>   | 任意の HTTP 接続サービスにかかる最短時間。          |
| <b>connection.avg</b>       | 任意の HTTP 接続サービスにかかる平均処理時間。        |
| <b>connection.active</b>    | 現在オープン状態のスレッドを持っている接続数。           |
| <b>connection.time</b>      | HTTP 接続のサービスにかかる時間の合計。            |
| <b>connection.completed</b> | 接続処理が完了した回数。                      |
| <b>numMods.value</b>        | ロードされたモジュールの数。                    |
| <b>childFinish.count</b>    | 理由にかかわらず、Apache 親サーバが子サーバを起動した回数。 |

| 測定値                           | 説明                                                                    |
|-------------------------------|-----------------------------------------------------------------------|
| <b>childStart.count</b>       | 「子」が「秩序正しく」終了した回数。エラーまたはクラッシュによって終了した回数は、childFinish.countでカウントされません。 |
| <b>Decline.count</b>          | 各モジュールが HTTP 要求を拒否した回数。                                               |
| <b>internalRedirect.count</b> | 「内部リダイレクト」によって任意のモジュールが別のモジュールに制御を渡した回数。                              |
| <b>cpuTime.value</b>          | Apache サーバ上のすべてのプロセスで使用されている CPU 時間の合計（ミリ秒）。                          |
| <b>heapSize.value</b>         | Apache サーバ上の全プロセスで使用されるヒープ・メモリの合計（キロバイトで測定）。                          |
| <b>pid.value</b>              | 親 Apache プロセスのプロセス識別子。                                                |
| <b>upTime.value</b>           | サーバの連続稼働時間（ミリ秒）。                                                      |

## SilverStream モニタの設定

SilverStream サーバを監視するには、サーバの統計情報の URL を知っておく必要があります。統計情報の URL を検証する簡単な方法は、ブラウザからその情報にアクセスして見ることです。

URL は、次の形式で指定します。

`http:// <サーバ名 /IP アドレス><ポート番号> /SilverStream/Statistics`

次に例を示します。

`http://199.203.78.57:80/SilverStream/Statistics`

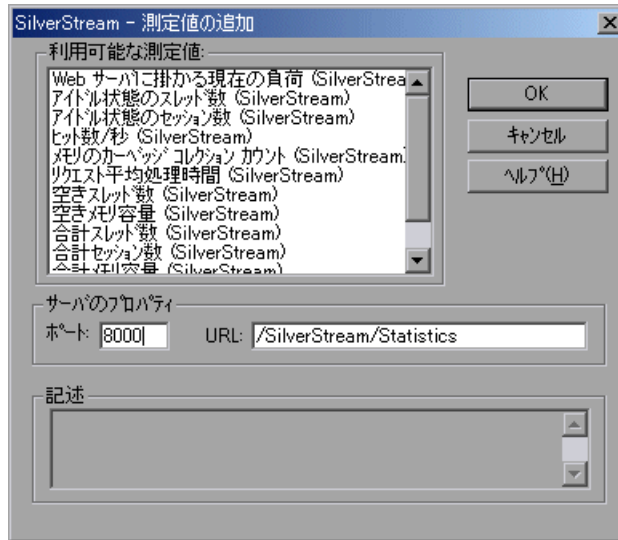
**SilverStream モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [SilverStream] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。



- 3 [SilverStream] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [SilverStream] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。

ダイアログ・ボックスが開き、使用可能な測定値とサーバのプロパティが表示されます。



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。測定可能なパフォーマンス・カウンタの一覧については、442 ページを参照してください。

- 5 [サーバのプロパティ] セクションにポート番号と URL (サーバ名を除く) を入力して、[OK] をクリックします。標準設定の URL は /SilverStream/Statistics です。
- 6 [SilverStream] 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。

---

**注：**標準設定のポート番号と URL は、サーバによって異なります。Web サーバの管理者に確認してください。

---

**標準設定のサーバのプロパティを変更するには、次の手順で行います。**

- 1 < LoadRunner のインストール先フォルダ > \¥dat¥ monitors ディレクトリにある **SilverStream.cfg** ファイルを開きます。
- 2 ファイルの末尾にある次のパラメータを編集します。

|                     |                                                                                                                                                            |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>InfoURL</b>      | サーバの統計情報の URL                                                                                                                                              |
| <b>ServerPort</b>   | サーバ・ポート番号                                                                                                                                                  |
| <b>SamplingRate</b> | LoadRunner モニタが、サーバの統計情報のポーリングを行う頻度（ミリ秒単位）。この値が 1000 より大きい場合、LoadRunner は、サンプリング頻度としてこの値を使いません。それ以外の場合は、[オプション] ダイアログ・ボックスの [モニタ] タブに定義されているサンプリング頻度を使います。 |

---

**注：**ファイアウォール越しに SilverStream サーバを監視するには、Web サーバのポート（標準設定では 80 番）を使用します。

---

## SilverStream パフォーマンス・カウンタ

次の表に、監視できる測定値およびサーバのプロパティの説明を示します。

| 測定値                                 | 説明                                 |
|-------------------------------------|------------------------------------|
| <b>#Idle Sessions</b>               | アイドル状態のセッション数。                     |
| <b>Avg. Request processing time</b> | 要求の平均処理時間。                         |
| <b>Bytes Sent/sec</b>               | Web サーバがデータ・バイトを送信する速度。            |
| <b>Current load on Web Server</b>   | SilverStream サーバが使用した負荷の割合（倍率 25）。 |

| 測定値                             | 説明                                          |
|---------------------------------|---------------------------------------------|
| Hits/sec                        | HTTP 要求の秒ごとの数。                              |
| Total sessions                  | セッションの総数。                                   |
| Free memory                     | 将来的にオブジェクトの割り当てが可能な、Java 仮想マシンの現在のメモリの総量。   |
| Total memory                    | Java 仮想マシンのメモリの総量。                          |
| Memory Garbage Collection Count | サーバが開始されてから、JAVA ガーベジ・コレクタが実行された総回数。        |
| Free threads                    | クライアント接続に関係しておらず、直ちに使用可能なスレッドの現在の数。         |
| Idle threads                    | クライアント接続に関係しているが、現在ユーザ・リクエストを処理していないスレッドの数。 |
| Total threads                   | 割り当てられたクライアント・スレッドの総数。                      |

## WebLogic (SNMP) モニタの設定

WebLogic (SNMP) モニタは、SNMP を使ってサーバの統計値を取得します。このモニタを使用するには、6.0 以前のバージョンの WebLogic がインストールされていること、また SNMP エージェントがインストールされ、アクティブになっていることを確認してください。SNMP エージェントのインストール手順については、<http://www.weblogic.com/docs51/admindocs/snmpagent.html> を参照してください。

---

注：WebLogic (SNMP) サーバを監視するには、サーバの構成によって 161 番ポートまたは 162 番ポートのどちらかを使用します。

---

**WebLogic (SNMP) モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [WebLogic (SNMP)] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。

- 3 [WebLogic (SNMP)] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

---

注：WebLogic SNMP エージェントを実行しているポートの番号が SNMP の標準のポート番号と異なる場合には、ポート番号を定義する必要があります。

[マシンの追加] ダイアログ・ボックスに、次の情報を入力します。

<サーバ名：ポート番号>

次に例を示します。

digi:8888

また、<LoadRunner のインストール先フォルダ>¥dat¥monitors にある設定ファイル **snmp.cfg** で、WebLogic サーバの標準設定のポートを定義することもできます。たとえば、WebLogic サーバで SNMP エージェントによって使用されているポート番号が 8888 の場合、**snmp.cfg** ファイルを次のように編集します。

```
; WebLogic
[cm_snmp_mon_isp]
port=8888
```

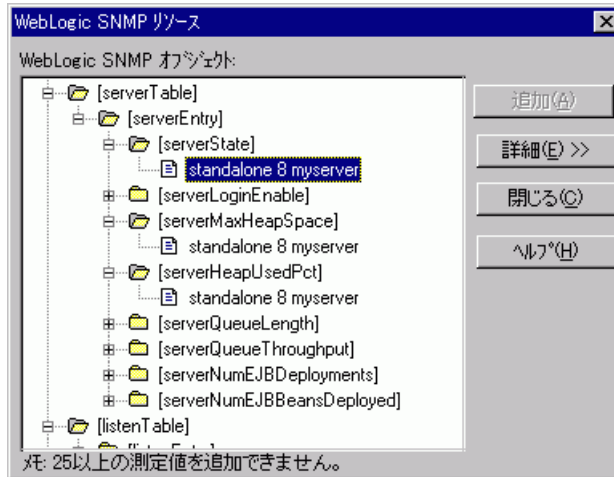
- 
- 4 [WebLogic (SNMP)] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。[WebLogic SNMP リソース] ダイアログ・ボックスに、使用可能な測定値が表示されます。

---

注：WebLogic (SNMP) モニタでは測定値を 25 個まで監視できます。

---

- 5 [WebLogic SNMP オブジェクト] ツリーが表示されます。



- 6 オブジェクトを測定するには、対象のオブジェクトを選択して [追加] をクリックします。
- 測定可能なパフォーマンス・カウンタの一覧については、445 ページを参照してください。
- 7 必要なオブジェクトを選択して追加したら、[閉じる] をクリックします。
- 8 [WebLogic (SNMP)] 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。

## WebLogic (SNMP) パフォーマンス・カウンタ

以下の表に、監視できる測定値およびサーバのプロパティの説明を示します。

### ServerTable

ServerTable には、エージェントによって監視されているすべての WebLogic (SNMP) サーバが一覧表示されます。サーバがこのテーブルに表示されるには、少なくとも 1 度はクラスタのメンバとして接続されるか報告されるかする

必要があります。サーバは、クラスタで現在使用されているとき、またはクラスタに加わってからすぐに、クラスタのメンバとして報告されます。

| 測定値                              | 説明                                                                                                              |
|----------------------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>ServerState</b>               | SNMP エージェントによって推測される、WebLogic サーバの状態。 <b>Up</b> は、エージェントがサーバに接続できることを示し、 <b>Down</b> はエージェントがサーバに接続できないことを示します。 |
| <b>ServerLoginEnable</b>         | クライアント・ログインがサーバで可能になっている場合は真。                                                                                   |
| <b>ServerMaxHeapSpace</b>        | このサーバの最大ヒープ・サイズ (KB)。                                                                                           |
| <b>ServerHeapUsedPct</b>         | サーバで現在使用中のヒープ・スペースの割合。                                                                                          |
| <b>ServerQueueLength</b>         | サーバの実行キューの現在の長さ。                                                                                                |
| <b>ServerQueueThroughput</b>     | 実行キューの現在のスループット。1 秒間に処理される要求数で表される。                                                                             |
| <b>ServerNumEJBDeployment</b>    | サーバが把握している EJB 配置ユニットの総数。                                                                                       |
| <b>ServerNumEJBBeansDeployed</b> | サーバでアクティブに配置された EJB ビーンの総数。                                                                                     |

### ListenTable

ListenTable は、プロトコル、IP アドレス、およびサーバがリスンしているポートの組み合わせのセットです。このテーブルには、各サーバに対して複数のエントリがあります。つまり、(protocol, ipAddr, port) の組み合わせごとに 1 つのサーバが対応します。クラスタリングが使用される場合、クラスタリング関連の MIB オブジェクトの優先度がより高くなります。

| 測定値               | 説明     |
|-------------------|--------|
| <b>ListenPort</b> | ポート番号。 |

| 測定値                  | 説明                                                                                                                                                                                                                |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>ListenAdminOK</b> | 管理要求がこの組み合わせ (protocol, ipAddr, port) に対して認められる場合は真, 認められない場合は偽。                                                                                                                                                  |
| <b>ListenState</b>   | この組み合わせ (protocol, ipAddr, port) がサーバで使用可能な場合は Listening となります。使用不可能な場合は not Listening となります。サーバは, サーバの Login Enable の状態が偽のとき, リッスンしていますが, 新しいクライアントは受け付けていないことがあります。この場合, 既存のクライアントは動作を継続しますが, 新しいクライアントは動作しません。 |

### ClassPathTable

ClassPathTable は, Java, WebLogic (SNMP) サーバ, およびサーブレットのクラスパス (CP) 要素のテーブルです。このテーブルには, 各サーバに対して複数のエントリがあります。また, サーバのパスごとに複数のエントリがある場合もあります。クラスタリングが使用される場合, クラスタリング関連の MIB オブジェクトの優先度がより高くなります。

| 測定値            | 説明                                                                                                                                                                                                                                         |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CPTYPE</b>  | CP 要素の種類。Java, WebLogic, servlet のどれかです。Java CPTYPE は, cpElement が標準 Java クラスパスの要素の 1 つであることを示します。WebLogic CPTYPE は, cpElement が weblogic.class.path の要素の 1 つであることを示します。また, servlet CPTYPE は, cpElement が動的サーブレット・クラスパスの要素の 1 つであることを示します。 |
| <b>CPIndex</b> | パス内の要素の位置。このインデックスは 1 から始まります。                                                                                                                                                                                                             |

## WebLogic (JMX) モニタの設定

BEA WebLogic (JMX) モニタでは、Java JMX インタフェースからサーバ上のランタイム MBeans にアクセスします。MBean とは、パフォーマンス・データを保持するコンテナです。

WebLogic (JMX) モニタを使用する前に、コントローラ・マシンに Java 1.3 以降をインストールしてください。Java 1.3 またはそれ以降が既にインストールされているにもかかわらず、標準の Java バージョンが使用されていない場合は、更新バージョンの完全パスを指定します。パスは、**< LoadRunner インストール先フォルダ > %dat%monitors%WebLogicMon.ini** ファイルで指定します。

[WebLogicMon] セクションで JVM エントリを編集します。次に例を示します。

```
JVM="E:%Program Files%JavaSoft%JRE%1.3.1%bin%javaw.exe
```

注：BEA WebLogic (JMX) モニタを使用するには、サーバ上にバージョン 6.0 以降の WebLogic をインストールしてください。

### 監視のための権限の設定

ユーザが MBeans を監視できるように権限を設定する必要があります。

権限を設定するには、次の手順で行います。

- 1 WebLogic のコンソール ([http:// <ホスト:ポート> /console](http://<ホスト:ポート>/console)) を開きます。
- 2 左側のツリーで、[**Security (セキュリティ)**] > [**ACLs**] を選択します。  
WebLogic バージョン 6.1 のコンソールで作業している場合は、画面右側の [**Create a new ACL.... (新規 ACL の作成)**] をクリックします。
- 3 [New ACL Name (新規 ACL 名)] ボックスで、「weblogic.admin.mbean」と入力し、[作成] をクリックします。  
WebLogic バージョン 6.1 のコンソールで作業している場合は、画面右側の [**Add a new Permission... (新規パーミッションの追加)**] をクリックします。
- 4 [New Permission (新規パーミッション)] ボックス (WebLogic バージョン 6.1 のコンソールの場合は [Permission (パーミッション)] ボックス、で「access」と入力します。WebLogic 6.0 のコンソールでは、[**Create (作成)**] をクリックします。



- 5 [Users (ユーザ)] ボックスおよび [Groups (グループ)] ボックスでは、監視に使用したい任意のユーザまたはグループ名を入力します。
- 6 WebLogic バージョン 6.0 のコンソールで、[Grant Permission (パーミッションを認可)] をクリックします。WebLogic バージョン 6.1 のコンソールでは、[Apply (適用)] をクリックします。

### サーバからのクラスの読み込み

WebLogic (JMX) モニタは、ClasspathServlet と呼ばれる組み込みサーバを利用して、直接サーバから自動的にクラスを読み込みます。これには、インストールが簡単で、バージョンに依存しないという利点があります。ただし、初めてクラスを読み込むときにパフォーマンスが多少低下するほか（サーブレットのサイズが大きいため）、サーブレットが利用できなくなる可能性があります。

サーブレットが使用不可になっているか、サーブレットを利用しない場合は、ファイル・システムからクラスを直接読み込むことができます。

ファイル・システムから直接クラスを読み込むには、次の手順で行います。

- 1 アプリケーション・サーバのインストール先フォルダ（lib フォルダの下）から < LoadRunner のインストール先フォルダ > %classes に **weblogic.jar** ファイルをコピーします。
- 2 クラス・ファイルが標準の < LoadRunner のインストール先フォルダ > がない場合、< LoadRunner のインストール先フォルダ > %dat%monitors%WebLogicMon.ini ファイルの中で、このクラス・ファイルへの完全パスを指定する必要があります。このファイルの中で、Weblogic=weblogic.jar という行を Weblogic=Weblogic=weblogic.jar に変更します。

## WebLogic (JMX) モニタの設定

[BEA WebLogic Monitor 設定] ダイアログ・ボックスを使って、WebLogic (JMX) アプリケーション・サーバを監視するための測定値を選択します。

WebLogic (JMX) モニタの設定は、次の手順で行います。

- 1 グラフ・ツリーで [WebLogic (JMX)] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [WebLogic (JMX)] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。次の形式に従って、サーバ名または IP アドレスを入力します。

<サーバ名> : <ポート番号>

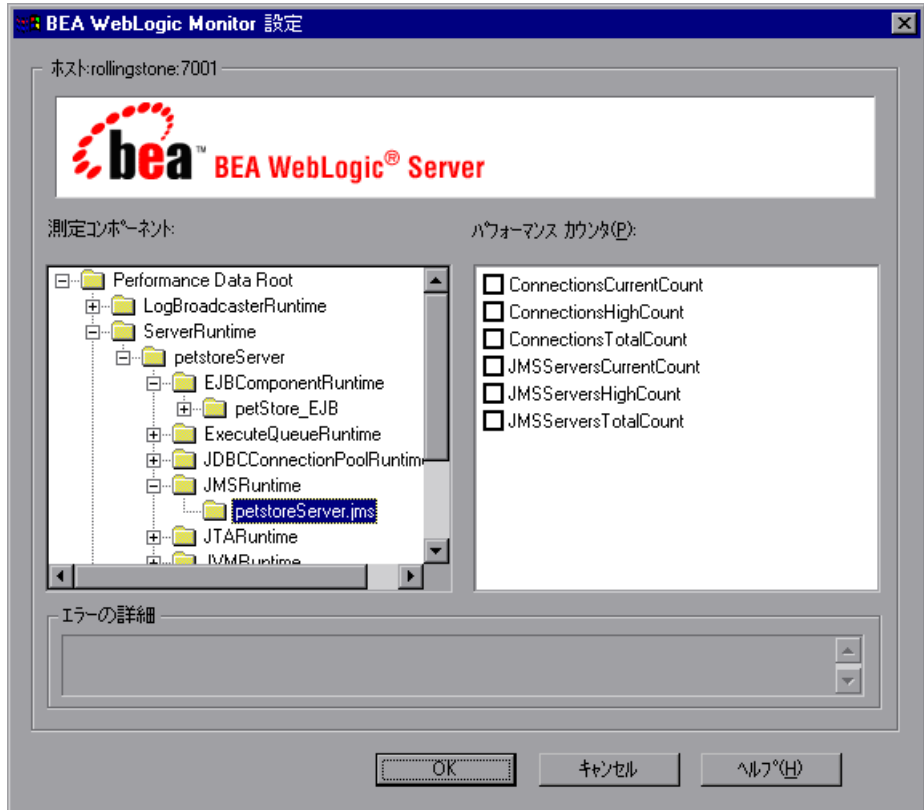
例を次に示します。

mercury:8111

マシンを実行するプラットフォームを選択して、[OK] をクリックします。

- 4 [WebLogic (JMX)] ダイアログ・ボックスの [リソース測定場所] セクションの [追加] をクリックします。[ログイン情報の入力] ダイアログ・ボックスに、WebLogic サーバに対する管理権限を持つユーザのユーザ名とパスワードを入力します。[BEA WebLogic Monitor 設定] ダイアログ・ボックスが開きます。ユーザ権限の作成の詳細については、448 ページ「監視のための権限の設定」を参照してください。

- 5 [測定コンポーネント] ツリーが表示されます。



測定可能なパフォーマンス・カウンタの一覧については、452 ページを参照してください。

- 6 [BEA WebLogic Monitor 設定] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。
- 7 [BEA WebLogic Monitor 設定] ダイアログ・ボックスと [WebLogic (JMX)] ダイアログ・ボックスで [OK] をクリックすると、WebLogic (JMX) モニタが起動します。

## BEA WebLogic (JMX) パフォーマンス・カウンタ

WebLogic (JMX) サーバに使用できる測定項目を以下に示します。

### LogBroadcasterRuntime

| 測定値                    | 説明                                                  |
|------------------------|-----------------------------------------------------|
| <b>MessagesLogged</b>  | 現在の WebLogic サーバのインスタンスによって生成されたログ・メッセージの総数。        |
| <b>Registered</b>      | このオブジェクトによって表される MBean の登録が解除された場合は「false」（偽）を返します。 |
| <b>CachingDisabled</b> | プロキシでのキャッシュを無効にするプライベート・プロパティ。                      |

### ServerRuntime

次の各測定項目のカテゴリに含まれる測定値の詳細については、マーキュリー・インタラクティブの「Load Testing Monitors」Web サイト ([http://www-heva.mercuryinteractive.com/products/loadrunner/load\\_testing\\_monitors/bealogic.html](http://www-heva.mercuryinteractive.com/products/loadrunner/load_testing_monitors/bealogic.html)) を参照してください。

- ▶ ServletRuntime
- ▶ WebAppComponentRuntime
- ▶ EJBStatefulHomeRuntime
- ▶ JTARuntime
- ▶ JVMRuntime
- ▶ EJBEntityHomeRuntime
- ▶ DomainRuntime
- ▶ EJBComponentRuntime
- ▶ DomainLogHandlerRuntime
- ▶ JDBCConnectionPoolRuntime
- ▶ ExecuteQueueRuntime
- ▶ ClusterRuntime
- ▶ JMSRuntime

- ▶ TimeServiceRuntime
- ▶ EJBStatelessHomeRuntime
- ▶ WLECConnectionServiceRuntime

### ServerSecurityRuntime

| 測定値                                       | 説明                                                  |
|-------------------------------------------|-----------------------------------------------------|
| <b>UnlockedUsersTotalCount</b>            | サーバ上でユーザがアンロックされた回数を返します。                           |
| <b>InvalidLoginUsersHighCount</b>         | サーバへの無効なログイン試行数が際立って多いユーザ数を返します。                    |
| <b>LoginAttemptsWhileLockedTotalCount</b> | ユーザがロックされている間の無効なログイン試行数の累計を返します。                   |
| <b>Registered</b>                         | このオブジェクトによって表される MBean の登録が解除された場合は「false」（偽）を返します。 |
| <b>LockedUsersCurrentCount</b>            | 現在サーバ上でロックされているユーザ数を返します。                           |
| <b>CachingDisabled</b>                    | プロキシでのキャッシュを無効にするプライベート・プロパティ。                      |
| <b>InvalidLoginAttemptsTotalCount</b>     | サーバ上での無効なログイン試行数の累計を返します。                           |
| <b>UserLockoutTotalCount</b>              | サーバ上で行われたユーザ・ロックアウトの累計数を返します。                       |

## WebSphere モニタの設定

WebSphere モニタを使用して、IBM WebSphere Application Server 3.x, 4.x, 5.x の可用性とサーバ統計値の監視が行えます。

WebSphere バージョン 5.x を監視するには、IBM WebSphere の「Installing a New Application」ウィザードを使用してアプリケーション・サーバにパフォーマンス・サーブレットを配備する必要があります。

**WebSphere 5.x 用にアプリケーション・サーバにパフォーマンス・サーブレットを配備するには、次の手順で行います。**

- 1 管理コンソールから、コンソール・ナビゲーション・ツリーで **[Applications]** > **[Install New Application]** をクリックします。
- 2 「Path」には、サーバ・マシンのソース・アプリケーション・ファイル（「PerfServletApp.ear」）の完全パス名を指定し、**[Next]** をクリックします。
- 3 **[Generate Default Bindings]** チェック・ボックスを選択して、**[Next]** をクリックします。
- 4 **[Install New Application]** ページで、**[Summary]** をクリックし、**[Cell/Node/Server]** オプションを選択します。**[Click here]** をクリックします。
- 5 **[Map modules to application servers]** パネルで、アプリケーション・ファイルのインストール先のサーバを **[Clusters and Servers]** リストから選択し、**[Module]** を選択して、すべてのアプリケーション・モジュールを選択します。
- 6 **[Next]** をクリックし、**[Summary]** パネルで、**[Finish]** をクリックします。
- 7 サーブレットが正しく実行されていること、およびパフォーマンス・データが生成されていることを確認します。パフォーマンス・データにアクセスできることを簡単に確認するには、ブラウザにパフォーマンス・データを表示します。URL は、次の形式で指定します。

`http://<サーバ名 : ポート番号> / <サーブレット・フォルダ>  
/com.ibm.ivb.epm.servlet.PerformanceServlet`

次に例を示します。

`http://websphere.mercury.co.il:81/servlet/com.ibm.ivb.epm.servlet.PerformanceServlet`

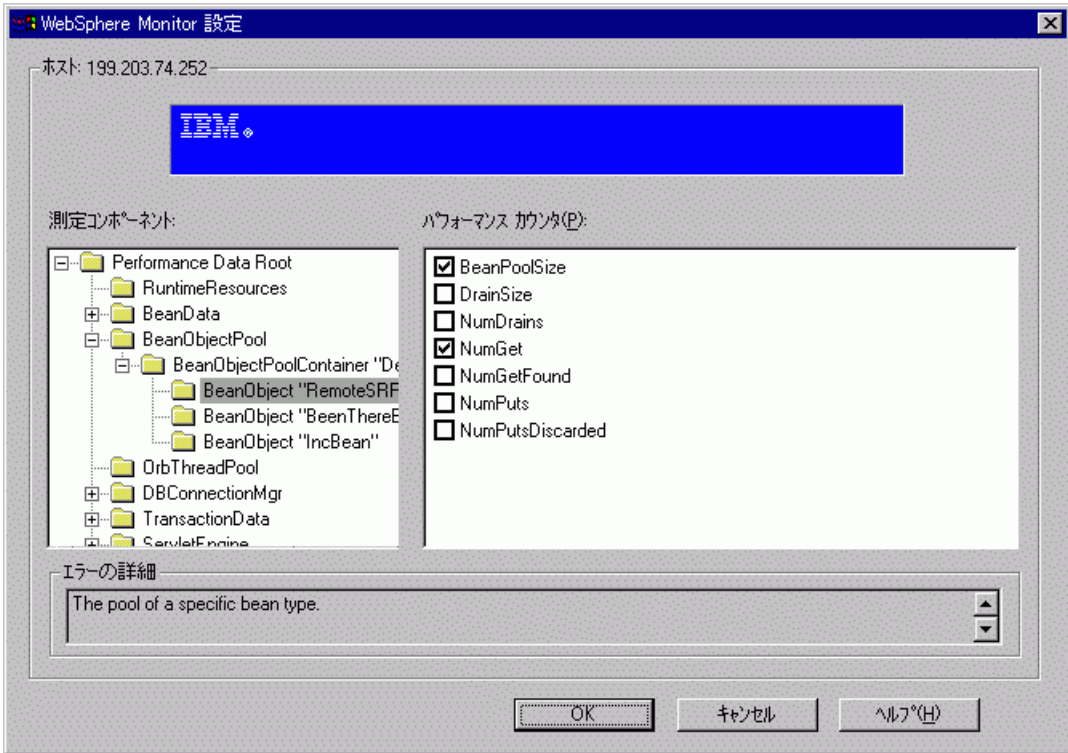
注：パフォーマンス XML ファイルを表示できるのは、XML と互換性のあるブラウザだけです。

---

**WebSphere または WebLogic 4.x ~ 5.x モニタの設定は、次の手順で行います。**

- 1 グラフ・ツリーで [WebSphere] または [WebSphere 4.x - 5.x] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [WebSphere] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [WebSphere] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。[WebSphere Monitor 設定] ダイアログ・ボックスに、使用可能な測定値が表示されます。

5 [測定コンポーネント] ツリーが表示されます。



- 6 [WebSphere Monitor 設定] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。使用できるパフォーマンス・カウンタのリストについては、459 ページを参照してください。
- 7 [WebSphere Monitor 設定] ダイアログ・ボックスと [WebSphere] ダイアログ・ボックスで [OK] をクリックすると、WebSphere モニタが起動します。

---

**注：**ファイアウォール越しに WebSphere サーバを監視するために使用するポートは、サーバの構成によって異なります。

---



サブレット・ディレクトリ用に別の Web エイリアスを指定するには、次の手順で行います。

標準設定では、LoadRunner は「servlet」をサブレット・ディレクトリの Web エイリアスとして使います。たとえば、WebSphere サーバ・マシンの名前が mercury でサブレット・ディレクトリのパスが次のとおりだとします。

```
E:¥AppServer¥hosts¥default_host¥default_app¥servlets
```

LoadRunner XML ファイルを次の URL にリクエストします。

```
http://mercury/servlet/com.ibm.ivb.epm.servlet.PerformanceServlet
```

servlet には、サブレット・ディレクトリ用の Web エイリアスが入ります。

サブレット・ディレクトリの Web エイリアスが servlet ではない場合は、[マシンの追加] ダイアログ・ボックスでサブレット・ディレクトリの Web エイリアスを次の形式で指定しなければなりません。

```
http://<サーバ名:ポート番号>/<サブレット・ディレクトリのエイリアス>
```

例を次に示します。

```
http://mercury/servlet2
```

この方法で、アプリケーション・サーバを必要な数だけ監視できます（インストール先のマシンが 1 台か複数かは問いません）。

**標準設定のアプリケーションのほかに、別のアプリケーションを監視するには、次の手順で行います。**

アプリケーションが同じマシンにインストールされているか、別々のマシンにインストールされているかに関係なく、アプリケーションを必要な数だけ監視できます。

- 1 標準設定のアプリケーションの Servlets ディレクトリにコピーしたファイルと同じファイルを、監視対象にする他の Web アプリケーションの Servlets ディレクトリにコピーします。
- 2 WebSphere コンソールで、各 Web アプリケーションの設定に com.ibm.ivb.epm.servlet.PerformanceServlet を追加します。
- 3 監視の対象にする Web アプリケーションを次の形式で WebSphere パフォーマンス・モニタに追加します。

```
http://<server:port_number>/<servlet_dir_alias>/servlet
```

例を次に示します。

<http://mercury/servlet3/servlet>

**WebSphere バージョン 3.5.x** での作業は、次の手順で行います。

- 1 3.5.x の EPM カウンタは、標準では「none」に設定されています。EPM カウンタを有効にするには、WebSphere 管理のコンソールのブラウザの中で監視対象アプリケーション・サーバを選択します。
- 2 アプリケーション・サーバを右クリックして、[**Performance**] を選択します。ポップアップ・ウィンドウから [Performance Modules] を選択します。
- 3 [Performance Modules] を右クリックして、パフォーマンス・レベルを選択します。さまざまなレベルのカウンタを選択することによって、アプリケーション・サーバで、さまざまなレベルのパフォーマンス・データを管理できるようになります。
- 4 [Set] ボタンをクリックします。
- 5 バージョン 3.5.2 および 3.5.3 では、Servlet カウンタは無効にされています。Servlet カウンタを有効にするには、< WAS\_HOME > \lib\ibmwebas.jar にある ccom/ibm/servlet/appserver.properties ファイルの内容を修正する必要があります。

**jar** ファイルを取り出し、appserver.properties を次のように変更します。

```
#listeners.application=com.ibm.servlet.engine.EPMApplicationListener
com.ibm.servlet.debug.OLTServletManager
listeners.application=
```

上記を次のように修正します。

```
listeners.application=com.ibm.servlet.engine.EPMApplicationListener
com.ibm.servlet.debug.OLTServletManager
#listeners.application=
```

- 6 **jar** ファイルを再パッケージします。

## WebSphere パフォーマンス・カウンタ

以下の表に、監視可能なカウンタの説明を示します。

### Run-Time Resources

Java 仮想マシンにのランタイムおよび ORB に関するリソースが含まれます。

| 測定値                | 説明                        |
|--------------------|---------------------------|
| <b>MemoryFree</b>  | Java 仮想マシンの空きメモリ容量。       |
| <b>MemoryTotal</b> | Java 仮想マシンに割り当てられたメモリの総量。 |
| <b>MemoryUse</b>   | Java 仮想マシン内で使用中のメモリの総量。   |

### BeanData

サーバ上のすべてのホームは、ホームに配置されたビーンの種類に応じてパフォーマンス・データを提供します。トップ・レベルのビーン・データには、すべてのコンテナのデータの集合が含まれます。

| 測定値                           | 説明                                                                  |
|-------------------------------|---------------------------------------------------------------------|
| <b>BeanCreates</b>            | 作成されたビーンの数。「ステートフル」または「エンティティ」である個々のビーンが対象となります。                    |
| <b>EntityBeanCreates</b>      | 作成されたエンティティ・ビーンの数。                                                  |
| <b>BeanRemoves</b>            | 削除された特定のビーンに関係のあるエンティティ・ビーンの数。「ステートフル」または「エンティティ」である個々のビーンが対象となります。 |
| <b>EntityBeanRemoves</b>      | 削除されたエンティティ・ビーンの数。                                                  |
| <b>StatefulBeanCreates</b>    | 作成されたステートフル・ビーンの数。                                                  |
| <b>StatefulBeanRemoves</b>    | 削除されたステートフル・ビーンの数。                                                  |
| <b>BeanPassivates</b>         | 特定のビーンに関係のある、不活性化されたビーンの数。「ステートフル」または「エンティティ」である個々のビーンが対象となります。     |
| <b>EntityBeanPassivates</b>   | 不活性化されたエンティティ・ビーンの数。                                                |
| <b>StatefulBeanPassivates</b> | 不活性化されたステートフル・ビーンの数。                                                |

| 測定値                              | 説明                                                                      |
|----------------------------------|-------------------------------------------------------------------------|
| <b>BeanActivates</b>             | 特定のビーンに関係のある、活性化されたビーンの数。「ステートフル」または「エンティティ」である個々のビーンが対象となります。          |
| <b>EntityBeanActivates</b>       | 活性化されたエンティティ・ビーンの数。                                                     |
| <b>StatefulBeanActivates</b>     | 活性化されたステートフル・ビーンの数。                                                     |
| <b>BeanLoads</b>                 | ビーン・データがロードされた回数。エンティティが対象となります。                                        |
| <b>BeanStores</b>                | ビーン・データがデータベースに格納された回数。エンティティが対象となります。                                  |
| <b>BeanInstantiates</b>          | ビーン・オブジェクトが作成された回数。ビーンの種類に関係なく、個々のビーンが対象となります。                          |
| <b>StatelessBeanInstantiates</b> | ステートレス・セッション・ビーン・オブジェクトが作成された回数。                                        |
| <b>StatefulBeanInstantiates</b>  | ステートフル・セッション・ビーン・オブジェクトが作成された回数。                                        |
| <b>EntityBeanInstantiates</b>    | エンティティ・ビーン・オブジェクトが作成された回数。                                              |
| <b>BeanDestroys</b>              | 個々のビーン・オブジェクトが破棄された回数。ビーンの種類に関係なく、あらゆるビーンが対象となります。                      |
| <b>StatelessBeanDestroys</b>     | ステートレス・セッション・ビーン・オブジェクトが破棄された回数。                                        |
| <b>StatefulBeanDestroys</b>      | ステートフル・セッション・ビーン・オブジェクトが破棄された回数。                                        |
| <b>EntityBeanDestroys</b>        | エンティティ・ビーン・オブジェクトが作成された回数。                                              |
| <b>BeansActive</b>               | 特定のビーンに関係のあるアクティブなビーンの実インスタンスの平均数。「ステートフル」または「エンティティ」である個々のビーンが対象となります。 |
| <b>EntityBeansActive</b>         | アクティブなエンティティ・ビーンの平均数。                                                   |
| <b>StatefulBeansActive</b>       | アクティブなセッション・ビーンの平均数。                                                    |

| 測定値                       | 説明                                                                      |
|---------------------------|-------------------------------------------------------------------------|
| <b>BeansLive</b>          | インスタンス化されているがまだ破棄されていない特定の種類のビーン・オブジェクトの平均数。ビーンの種類に関係なく、個々のビーンが対象となります。 |
| <b>StatelessBeansLive</b> | インスタンス化されているがまだ破棄されていない、ステートレス・セッション・ビーン・オブジェクトの平均数。                    |
| <b>StatefulBeansLive</b>  | インスタンス化されているがまだ破棄されていない、ステートフル・セッション・ビーン・オブジェクトの平均数。                    |
| <b>EntityBeansLive</b>    | インスタンス化されているがまだ破棄されていない、エンティティ・ビーン・オブジェクトの平均数。                          |
| <b>BeanMethodRT</b>       | 該当するビーンに対するリモート・インタフェースで定義されているすべてのメソッドの平均応答時間。すべてのビーンが対象となります。         |
| <b>BeanMethodActive</b>   | 同時に処理されているメソッドの平均数。すべてのビーンが対象となります。                                     |
| <b>BeanMethodCalls</b>    | 該当するビーンの方法が呼び出された合計回数。                                                  |

### BeanObjectPool

サーバには、ビーン・オブジェクトのキャッシュが保持されます。各ホームにはキャッシュがあるので、コンテナごとに1つの **BeanObjectPoolContainer** があります。トップ・レベルの **BeanObjectPool** には、すべてのコンテナ・データの集合が含まれます。

| 測定値                            | 説明                               |
|--------------------------------|----------------------------------|
| <b>BeanObjectPoolContainer</b> | 特定の種類のビーンのプール。                   |
| <b>BeanObject</b>              | ホーム専用のプール。                       |
| <b>NumGet</b>                  | プールからオブジェクトを検索する呼び出しの回数。         |
| <b>NumGetFound</b>             | プールに対する呼び出しのうち、使用可能なビーンが検出された回数。 |
| <b>NumPuts</b>                 | プールに解放されたビーンの数。                  |

| 測定値                     | 説明                                     |
|-------------------------|----------------------------------------|
| <b>NumPutsDiscarded</b> | プールがいっぱいであったために、プールに解放されたビーンが破棄された回数。  |
| <b>NumDrains</b>        | デーモンが、プールがアイドル状態なのを検出し、プールを掃除しようとした回数。 |
| <b>DrainSize</b>        | デーモンの掃除によって破棄されたビーンの平均数。               |
| <b>BeanPoolSize</b>     | プール内のビーンの平均数。                          |

### OrbThreadPool

サーバ上の ORB スレッド・プールに関するリソースを以下に示します。

| 測定値                      | 説明                                     |
|--------------------------|----------------------------------------|
| <b>ActiveThreads</b>     | プール内のアクティブ・スレッドの平均数。                   |
| <b>TotalThreads</b>      | プール内のスレッドの平均数。                         |
| <b>PercentTimeMaxed</b>  | プール内のスレッド数が適正最大数に到達、またはそれを超えた時間の割合の平均。 |
| <b>ThreadCreates</b>     | 作成されたスレッドの数。                           |
| <b>ThreadDestroys</b>    | 破棄されたスレッドの数。                           |
| <b>ConfiguredMaxSize</b> | プールされるスレッドの、設定された最大数。                  |

### DBConnectionMgr

データベース接続マネージャに関するリソースを以下に示します。データベース接続マネージャは、一連のデータ・ソース、および各パフォーマンス測定値のトップ・レベルの集合で構成されます。

| 測定値                       | 説明                                   |
|---------------------------|--------------------------------------|
| <b>DataSource</b>         | 「name」属性によって指定された特定のデータ・ソースに関するリソース。 |
| <b>ConnectionCreates</b>  | 作成された接続の数。                           |
| <b>ConnectionDestroys</b> | 解放された接続の数。                           |

| 測定値                           | 説明                   |
|-------------------------------|----------------------|
| <b>ConnectionPoolSize</b>     | プールの平均サイズ。つまり、平均接続数。 |
| <b>ConnectionAllocates</b>    | 接続が割り当てられた回数。        |
| <b>ConnectionWaiters</b>      | 接続を待機しているスレッドの平均数。   |
| <b>ConnectionWaitTime</b>     | 接続許可の平均時間（秒）。        |
| <b>ConnectionTime</b>         | 接続が使用されている平均時間（秒）。   |
| <b>ConnectionPercentUsed</b>  | 使用中であるプールの割合の平均。     |
| <b>ConnectionPercentMaxed</b> | すべての接続が使用中である時間の割合。  |

### TransactionData

トランザクションに関係のあるリソースを以下に示します。

| 測定値                           | 説明                                 |
|-------------------------------|------------------------------------|
| <b>NumTransactions</b>        | 処理されたトランザクションの数。                   |
| <b>ActiveTransactions</b>     | アクティブ・トランザクションの平均数。                |
| <b>TransactionRT</b>          | 各トランザクションの平均所要時間。                  |
| <b>BeanObjectCount</b>        | トランザクションに関与しているビーン・オブジェクト・プールの平均数。 |
| <b>RolledBack</b>             | ロール・バックされたトランザクションの数。              |
| <b>Committed</b>              | コミットされたトランザクションの数。                 |
| <b>LocalTransactions</b>      | ローカルだったトランザクションの数。                 |
| <b>TransactionMethodCount</b> | 各トランザクションの一部として呼び出されたメソッドの平均数。     |
| <b>Timeouts</b>               | 非活動タイムアウトによりタイムアウトとなったトランザクションの数。  |
| <b>TransactionSuspended</b>   | トランザクションが中断された平均回数。                |

## ServletEngine

サーブレットと JSP に関係のあるリソースを以下に示します。

| 測定値                       | 説明                                 |
|---------------------------|------------------------------------|
| <b>ServletsLoaded</b>     | 現在ロードされているサーブレットの数。                |
| <b>ServletRequests</b>    | サービスされた要求の数。                       |
| <b>CurrentRequests</b>    | 現在サービスされている要求の数。                   |
| <b>ServletRT</b>          | 各要求の平均応答時間。                        |
| <b>ServletsActive</b>     | 要求をアクティブに処理しているサーブレットの平均数。         |
| <b>ServletIdle</b>        | サーバがアイドル状態である時間（つまり、最後の要求からの時間）。   |
| <b>ServletErrors</b>      | エラーまたは例外となった要求の数。                  |
| <b>ServletBeanCalls</b>   | サーブレットによって実行されたビーン・メソッド呼び出しの回数。    |
| <b>ServletBeanCreates</b> | サーブレットによって実行されたビーン参照の回数。           |
| <b>ServletDBCalls</b>     | サーブレットによって実行されたデータベース呼び出しの回数。      |
| <b>ServletDBConAlloc</b>  | サーブレットによって割り当てられたデータベース接続の数。       |
| <b>SessionLoads</b>       | サーブレットのセッション・データがデータベースから読み取られた回数。 |
| <b>SessionStores</b>      | サーブレットのセッション・データがデータベースに格納された回数。   |
| <b>SessionSize</b>        | セッション・データの平均サイズ（バイト）。              |
| <b>LoadedSince</b>        | サーバがロードされてから経過した時間（UNC 時間）。        |



## Sessions

HTTP セッション・プールに関する一般的な測定値を以下に示します。

| 測定値                        | 説明                                                      |
|----------------------------|---------------------------------------------------------|
| <b>SessionsCreated</b>     | サーバで作成されたセッションの数。                                       |
| <b>SessionsActive</b>      | 現在のアクティブ・セッションの数。                                       |
| <b>SessionsInvalidated</b> | 無効になったセッションの数。データベース・モードでセッションを使用している場合は、有効ではないことがあります。 |
| <b>SessionLifetime</b>     | 無効になったセッションの統計データ。まだ有効であるセッションは含まれません。                  |

## WebSphere (EPM) モニタの設定

IBM WebSphere アプリケーション・サーバ (3.5.x) を監視するには、コントローラ・マシンに IBM WebSphere の管理コンソールをあらかじめインストールしておかなければなりません。セキュリティ・キーリングのコピーが必要になる場合もあります。

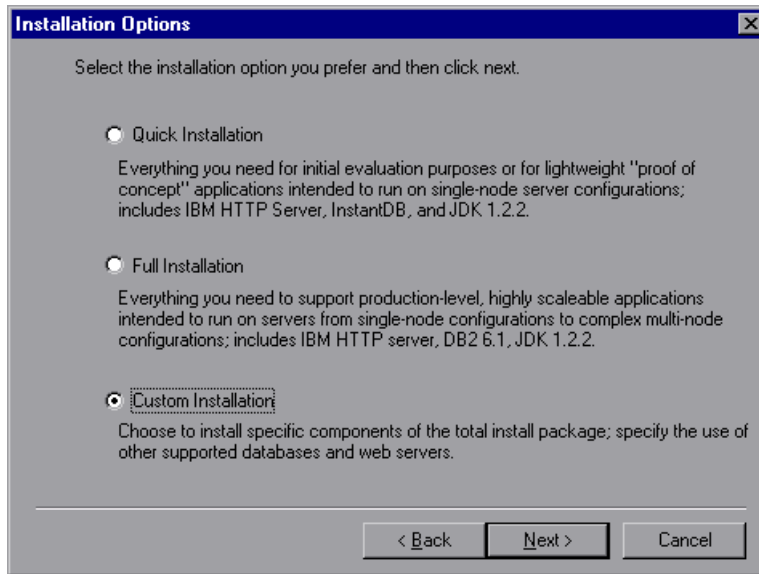
## IBM WebSphere 管理コンソールのインストール

- 1 WebSphere 3.5 Windows NT の配布 CD-ROM から、WebSphere インストール・プログラムを起動します。[WebSphere Application Server] ダイアログ・ボックスが開きます。

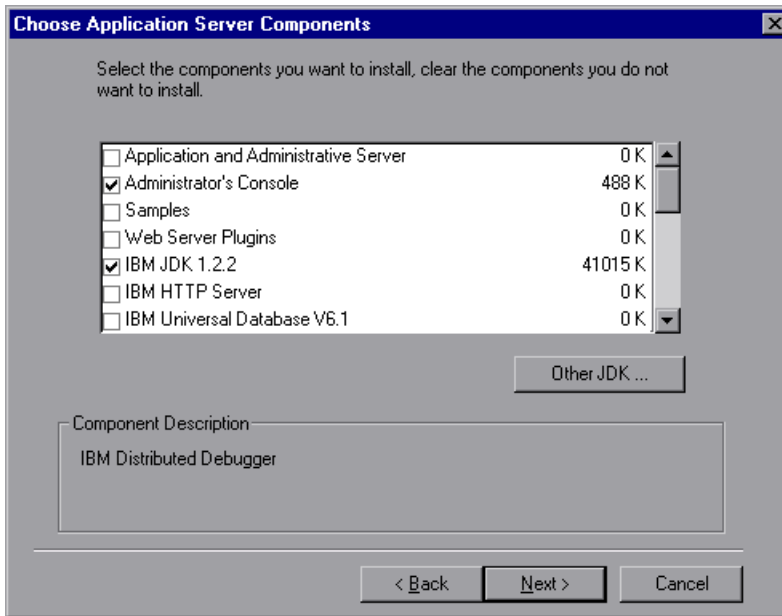


- 2 WebSphere で使用する Web サーバをすべてシャットダウンするよう指示されますが、無視します。この手順は、管理コンソールのインストールには関係ありません。それ以外の手順を進めます。

- 3 [Next] をクリックして先に進みます。[Installation Options] ダイアログ・ボックスが開きます。

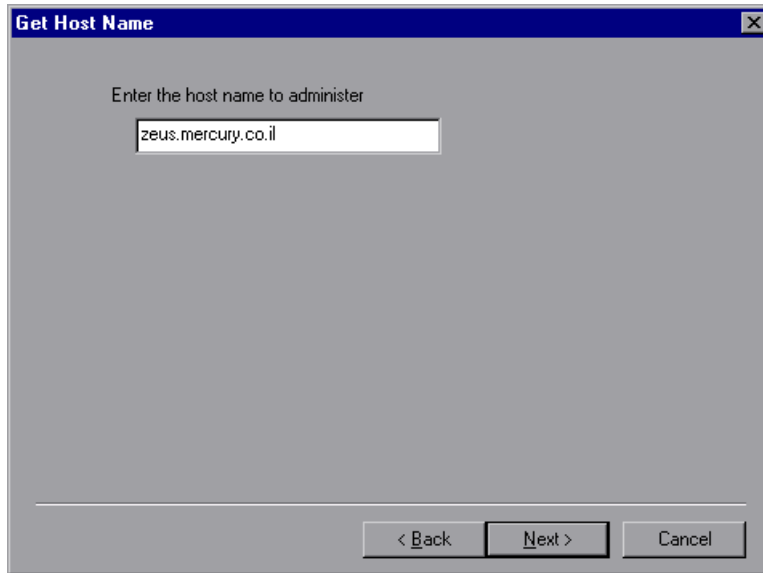


- 4 [Custom Installation] を選択して, [Next] をクリックします。[Choose Application Server Components] ダイアログ・ボックスが開きます。



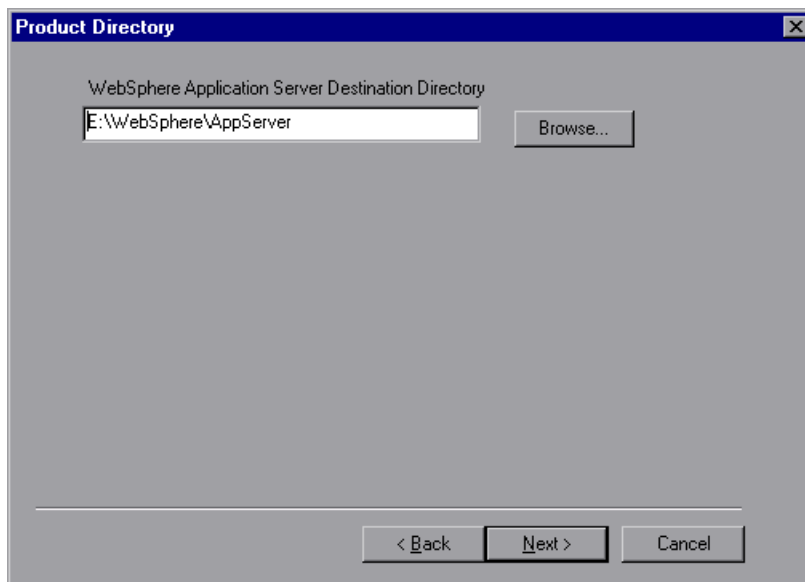
- 5 [Administrator's Console] および [IBM JDK 1.2.2] を選択します。その他のオプションはすべてクリアします。

- 6 [Next] をクリックします。[Get Host Name] ダイアログ・ボックスが開きます。



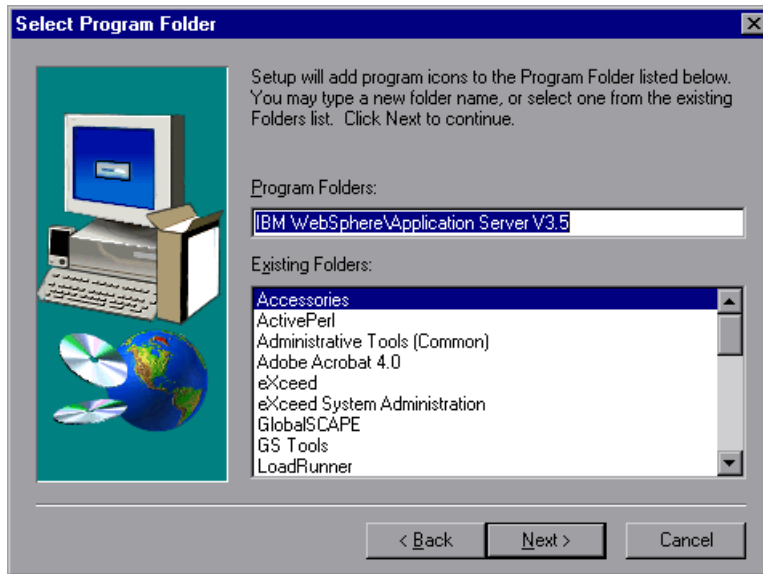
- 7 監視対象マシンの名前を入力します。

- 8 [Next] をクリックします。[Product Directory] ダイアログ・ボックスが開きます。

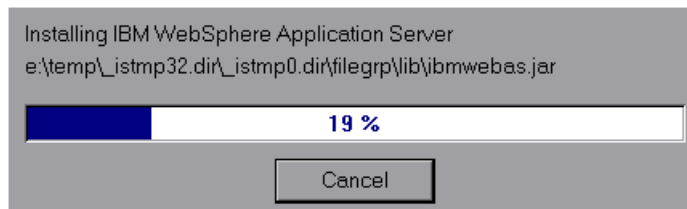


- 9 管理コンソールのインストール先フォルダを指定します。別の場所を選択するには、[Browse] をクリックして標準設定のフォルダ以外のフォルダを選択し、[OK] をクリックします。

- 10 **[Next]** をクリックします。**[Select Program Folder]** ダイアログ・ボックスが開きます。



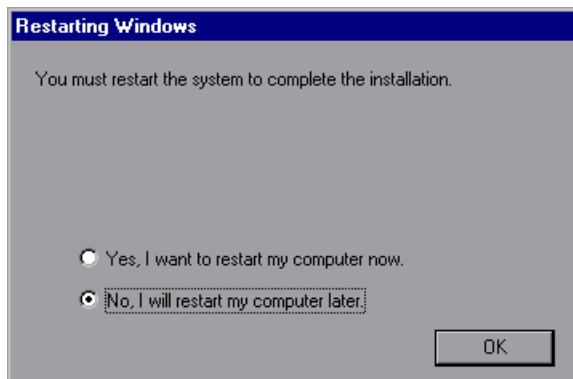
- 11 プログラム・フォルダを指定するか、標準設定の IBM WebSphere¥Application Server V3.5 フォルダを受け入れます。
- 12 **[Next]** をクリックします。インストール処理が開始します。インストール処理を一時停止または中止するには、**[Cancel]** をクリックします。



インストールが完了すると、[Setup Complete] ダイアログ・ボックスが開きます。



- 13 プログラムを起動する前に readme ファイルを表示するには、[Setup Complete] ダイアログ・ボックスで該当するチェック・ボックスを選択します。readme ファイルは、[スタート] > [プログラム] > [Application Server V3.5] > [IBM WebSphere] > [README] を選択することで随時表示できます。
- 14 [Finish] をクリックして、インストール・プログラム終了します。[Restarting Windows] ダイアログ・ボックスが開きます。





- 15 今すぐコンピュータを再起動してインストールを完了するか（推奨）、後で再起動してインストールを完了するかを選択します。
- 16 [OK] をクリックすると、管理コンソールのインストールが完了します。

### セキュリティ・キーリングのコピー

WebSphere サーバでセキュリティを有効にした場合、WebSphere サーバから管理クライアントにセキュリティ・キーリングをコピーする必要があります（セキュリティが有効になっているかどうかは、管理コンソールを管理サーバに接続できるかどうかで確認できます）。キーリングとは、サーバがクライアントの識別に使用する証明書です。

サーバの `lib` フォルダからクライアントの `lib` フォルダに、キーリングが含まれている `jar` ファイルをコピーする必要があります。また、監視側のクライアントのコマンド・ラインに、キーリングが含まれている `.jar` ファイルを追加する必要があります。

---

**注：** このファイル (`353Keyring.jar`) で使用されるキーリングは、バージョン 3.52 以前を使用するサーバにインストールしておく必要のある IBM ダミー・キーリングです。サーバのバージョンが 3.52 以前で、IBM ダミー・キーリングをすでに使用している場合、この行を変更する必要はありません。また、サーバのバージョンが 3.53 以降で、IBM ダミー・キーリングを使用している場合は、何もする必要はありません。

---

キーリングをコピーするには、次の手順で行います。

- 1 サーバから管理クライアントの `lib` フォルダ（標準設定では `C:\¥Websphere¥Appserver¥lib`）にキーリングの `jar` ファイルをコピーします。  
 キーリングが含まれる `jar` ファイル `xxxKeyring.jar` は、標準では次の場所にあります。
 

|          |                                          |
|----------|------------------------------------------|
| NT サーバ   | <code>C:\¥Websphere¥Appserver¥lib</code> |
| UNIX サーバ | <code>OPT/websphere/Appserver/lib</code> |

- 2 テキスト・エディタで、**< LoadRunner のインストール先フォルダ >**  
**¥dat¥monitors¥WebSphere35Mon.ini** ファイルを開きます。
- 3 次の行を見つけます。  
JVM\_CLASSES4=C:¥WebSphere¥AppServer¥lib¥353Keyring.jar

---

注：WebSphere を標準設定と異なる場所にインストールしてある場合、この行は異なります。

---

- 4 **353Keyring.jar** を、使用しているキーリングに変更します。

### WebSphere 3.5.x サーバでの EPM カウンタの有効化

標準の設定では「none」となっている EPM カウンタを有効にするには、WebSphere の管理コンソール・ブラウザの中で監視対象アプリケーションを右クリックして、**[Performance]** を選択します。表示されるダイアログ・ボックスで、**[Performance Modules]** ツリーを展開します。パフォーマンス・データの各種レベルを管理するには、パフォーマンス・モジュールを右クリックして、パフォーマンス・レベルを選択します。**[Set]** ボタンをクリックします。

または、アプリケーション・サーバが起動していることを確認して、WebSphere の管理コンソール・ブラウザの **[Advanced (詳細設定)]** タブを選択し、**[EPM Specification]** ボックスに次のように入力します。  
epm=high:epm.beanMethodData=none

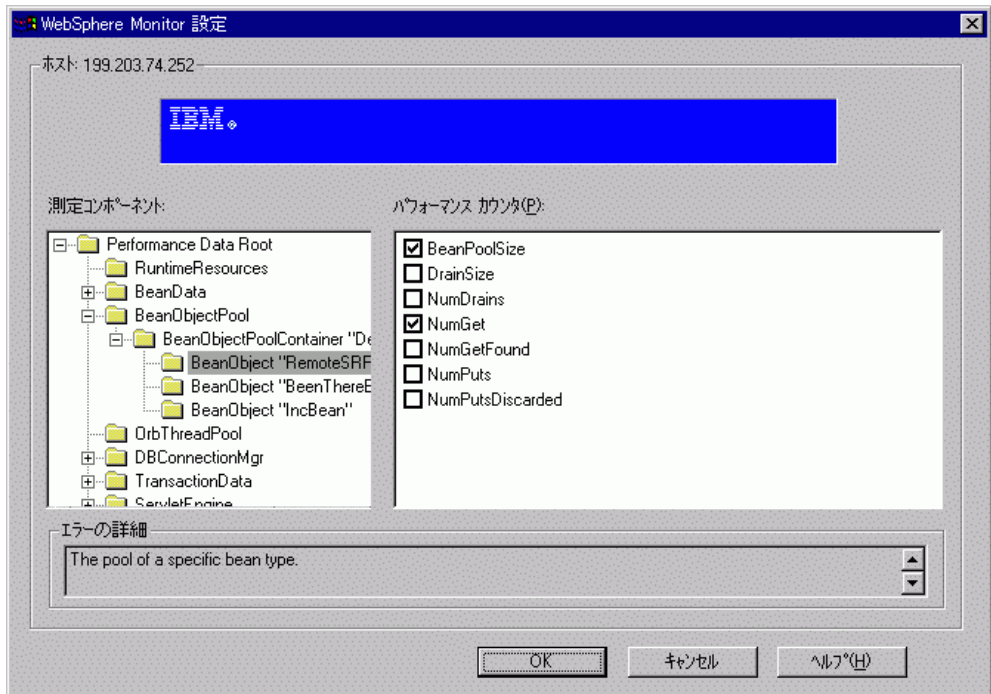
### WebSphere (EPM) モニタの設定

WebSphere の管理コンソールをインストールして EPM カウンタを有効にしたら、WebSphere (EPM) モニタを起動できます。

**WebSphere EPM モニタを起動するには、次の手順で行います。**

- 1 グラフ・ツリーで **[WebLogic (EPM)]** 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。
- 2 対象のグラフを右クリックして、**[測定値の追加]** を選択するか、**[モニタ] > [測定値の追加]** を選択します。

- 3 [WebSphere (EPM)] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [WebSphere (EPM)] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。[WebSphere Monitor 設定] ダイアログ・ボックスに、使用可能な測定値が表示されます。
- 5 [測定コンポーネント] ツリーが表示されます。



- 6 [WebSphere Monitor 設定] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。使用できるパフォーマンス・カウンタのリストについては、459 ページを参照してください。
- 7 [WebSphere Monitor 設定] ダイアログ・ボックスと [WebSphere (EPM)] ダイアログ・ボックスで [OK] をクリックすると、WebSphere モニタが起動します。



# 第 25 章

---

## データベース・リソースの監視

LoadRunner のデータベース・サーバ・リソース・モニタを使って、シナリオ実行時に DB2, Oracle, SQL Server, または Sybase データベース・リソースの使用状況を監視できます。

本章では、以下の項目について説明します。

- ▶ DB2 モニタの設定
- ▶ Oracle モニタの設定
- ▶ SQL Server モニタの設定
- ▶ Sybase モニタの設定

### データベース・リソースの監視について

DB2, Oracle, SQL Server, または Sybase データベース・サーバ・リソース・モニタは、DB2, Oracle, SQL Server, または Sybase データベース・サーバの統計値を測定します。シナリオ実行時にこれらのモニタを使って、データベース・サーバのパフォーマンスのボトルネックを特定できます。

シナリオを実行する前に、それぞれのデータベース・サーバで監視対象の測定値を設定します。DB2, Oracle, Sybase モニタを実行するには、監視対象のデータベース・サーバにクライアント・ライブラリをインストールする必要があります。

## DB2 モニタの設定

DB2 データベース・サーバ・モニタは、シナリオ実行中に DB2 データベースのリソースの使用状況を測定します。

---

**注：**データベースと連携して動作しているアプリケーションがない場合、監視の対象にできるのはデータベース・マネージャ・インスタンスだけです。

---

DB2 データベース・サーバを監視するには、あらかじめ DB2 モニタ環境を設定しておく必要があります。

**DB2 モニタ環境を設定するには、次の手順で行います。**

- 1 コントローラ・マシンにすべてのクライアント・ファイルとライブラリをインストールします。
- 2 [スタート] > [プログラム] > [DB2 for Windows NT] > [Control Center] を選択します。DB2 サーバの（管理者権限を持つ）ユーザ名とパスワードを入力します。
- 3 表示されるコンソールで [システム] を右クリックして、[追加] を選択します。
- 4 ダイアログ・ボックスに、次の設定を入力します。

**System Name :** <サーバ名>

**Remote Instance :** DB2

**Host Name :** <サーバ名>

**Service Name :** DB2 サーバ・ポート。標準設定の値は 50000 です。

- 5 [取得] をクリックした後、[OK] をクリックします。

---

**注：**[取得] をクリックした後にエラー・メッセージが返された場合は、手順 3 ~ 4 を繰り返して [OK] をクリックします。

---

- 6 コンソール・ツリーで、<サーバ名>ノードを展開します。
- 7 [インスタンス] を右クリックして、[追加] を選択します。

- 8 ダイアログ・ボックスに、次の設定を入力します。

**Remote Instance** : DB2

**Instance Name** : コントローラから呼び出すデータベース・インスタンス

**Host Name** : <サーバ名>

**Service Name** : DB2 サーバ・ポート。標準設定の値は 50000 です。

- 9 [OK] をクリックして、Control Center を閉じます。

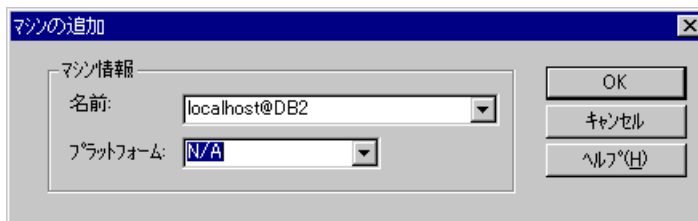
---

注：1つの監視セッションで対象にできるデータベース・マネージャ・インスタンスは1つだけです。

---

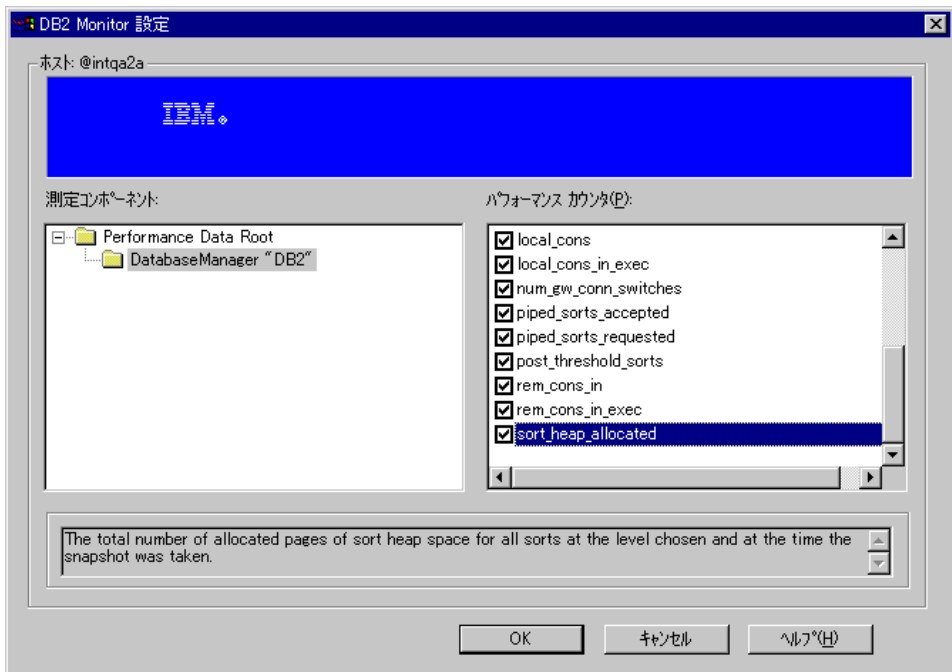
**DB2 モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [DB2] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。[DB2] ダイアログ・ボックスが開きます。
- 3 このダイアログ・ボックスの [リソース測定場所] セクションの [追加] ボタンをクリックします。[マシンを追加] ダイアログ・ボックスが開きます。
- 4 [名前] ボックスに、DB2 サーバのマシン名、続いて「@」記号、DB2 Control Center で指定したデータベース・インスタンスを入力します。[プラットフォーム] ボックスで、[N/A] を選択します。



[OK] をクリックして、入力した情報を保存してダイアログ・ボックスを閉じます。

- 5 [DB2] ダイアログ・ボックスの [リソース測定場所] セクションの [追加] をクリックします。表示されたダイアログ・ボックスに DB2 サーバのユーザ名とパスワードを入力して、[OK] をクリックします。[DB2 Monitor 設定] ダイアログ・ボックスが開きます。
- 6 [測定コンポーネント] ツリーを展開して、監視対象のメソッドとカウンタを選択します。



使用可能なパフォーマンス・カウンタの一覧については、481 ページを参照してください。

- 7 [追加] をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。
- 8 [DB2] 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。



## DB2 パフォーマンス・カウンタ

以下の表に、標準で監視可能なカウンタの説明を示します。

## データベース・マネージャ

| 測定値                              | 説明                                                                     |
|----------------------------------|------------------------------------------------------------------------|
| <b>rem_cons_in</b>               | リモート・クライアントによって監視対象のデータベース・マネージャのインスタンスに対して開始された現在の接続の数。               |
| <b>rem_cons_in_exec</b>          | 現在データベースに接続されていて、監視対象のデータベース・マネージャのインスタンス内で作業単位を処理しているリモート・アプリケーションの数。 |
| <b>local_cons</b>                | 監視対象のデータベース・マネージャのインスタンス内で現在データベースに接続されているローカル・アプリケーションの数。             |
| <b>local_cons_in_exec</b>        | 監視対象のデータベース・マネージャのインスタンス内で作業単位を処理しているローカル・アプリケーションの数。                  |
| <b>con_local_databases</b>       | アプリケーションが接続されているローカル・データベースの数。                                         |
| <b>agents_registered</b>         | 監視対象のデータベース・マネージャのインスタンスに登録されているエージェントの数（調整プログラム・エージェントおよびサブエージェント）。   |
| <b>agents_waiting_on_token</b>   | データベース・マネージャでトランザクションを実行するためにトークンを待っているエージェントの数。                       |
| <b>idle_agents</b>               | エージェント・プール内のエージェントで、現在アプリケーションに割り当てられていない「アイドル」となっているエージェントの数。         |
| <b>agents_from_pool</b>          | エージェント・プールから割り当てられたエージェントの数。                                           |
| <b>agents_created_empty_pool</b> | エージェント・プールが空だったために作成されたエージェントの数。                                       |

| 測定値                          | 説明                                                                                                |
|------------------------------|---------------------------------------------------------------------------------------------------|
| <b>agents_stolen</b>         | アプリケーションからエージェントが「スチールされた」回数。アプリケーションに関連付けられたアイドル・エージェントが、ほかのアプリケーションに再割り当てされたときに、「スチールされた」と言います。 |
| <b>comm_private_mem</b>      | スナップショット時にデータベース・マネージャのインスタンスが現在コミットしているプライベート・メモリの量。                                             |
| <b>inactive_gw_agents</b>    | DRDA 接続プール内の DRDA エージェントで、DRDA データベースに接続されてプライム状態になっているが、アクティブになっていない DRDA エージェントの数。              |
| <b>num_gw_conn_switches</b>  | エージェント・プールのエージェントが、接続されてプライム状態になっていたにもかかわらず別の DRDA データベースで使用するために「スチールされた」回数。                     |
| <b>sort_heap_allocated</b>   | スナップショット時に、選択したレベルでの全ソートに割り当てられているソート・ヒープ領域の総ページ数。                                                |
| <b>post_threshold_sorts</b>  | ソート・ヒープのしきい値に達した後に、ヒープを要求したソートの数。                                                                 |
| <b>piped_sorts_requested</b> | 要求されたパイプ・ソートの数。                                                                                   |
| <b>piped_sorts_accepted</b>  | 受け付けられたパイプ・ソートの数。                                                                                 |

## データベース

| 測定値                   | 説明                                                   |
|-----------------------|------------------------------------------------------|
| <b>appls_cur_cons</b> | 現在データベースに接続されているアプリケーションの数を表示します。                    |
| <b>appls_in_db2</b>   | 現在データベースに接続されていて、データベース・マネージャが現在要求を処理しているアプリケーションの数。 |

| 測定値                              | 説明                                                                             |
|----------------------------------|--------------------------------------------------------------------------------|
| <b>total_sec_cons</b>            | サブエージェントがノードにおいてデータベースに行った接続の数。                                                |
| <b>num_assoc_agents</b>          | アプリケーション・レベルでは、アプリケーションに関連付けられているサブエージェントの数。データベース・レベルでは、全アプリケーションのサブエージェントの数。 |
| <b>sort_heap_allocated</b>       | スナップショット時に、選択したレベルでの全ソートに割り当てられているソート・ヒープ領域の総ページ数。                             |
| <b>total_sorts</b>               | 実行されたソートの総数。                                                                   |
| <b>total_sort_time</b>           | 実行された全ソートの合計経過時間（ミリ秒）。                                                         |
| <b>sort_overflows</b>            | ソート・ヒープが足りなくなり、一時保存用のディスク領域が必要となったソートの総計。                                      |
| <b>active_sorts</b>              | 現在、データベースでソート・ヒープが割り当てられているソートの数。                                              |
| <b>total_hash_joins</b>          | 実行されたハッシュ結合の総数。                                                                |
| <b>total_hash_loops</b>          | 利用可能なソート・ヒープ容量よりもハッシュ結合の単一パーティションの方が大きかった回数の総計。                                |
| <b>hash_join_overflows</b>       | ハッシュ結合データが、利用可能なソート・ヒープ容量を超過した回数。                                              |
| <b>hash_join_small_overflows</b> | ハッシュ結合データによる利用可能なソート・ヒープ容量の超過が、10%以下だった回数。                                     |
| <b>pool_data_l_reads</b>         | バッファ・プールを通ったデータ・ページの論理読み取り要求の数。                                                |
| <b>pool_data_p_reads</b>         | データ・ページをバッファ・プールに読み込むためにI/Oを必要とした、読み取り要求の数。                                    |
| <b>pool_data_writes</b>          | バッファ・プール・データ・ページがディスクに物理的に書き込まれた回数。                                            |
| <b>pool_index_l_reads</b>        | バッファ・プールを通った索引ページへの論理読み取り要求の数。                                                 |

| 測定値                              | 説明                                                                                                                       |
|----------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| <b>pool_index_p_reads</b>        | 索引ページをバッファ・プールに置くための物理的読み取り要求の数。                                                                                         |
| <b>pool_index_writes</b>         | バッファ・プール索引ページがディスクに物理的に書き込まれた回数。                                                                                         |
| <b>pool_read_time</b>            | データまたは索引ページが、ディスクからバッファ・プールに物理的に読み込まれる原因となった読み取り要求の処理にかかった経過時間の合計。                                                       |
| <b>pool_write_time</b>           | データまたは索引ページをバッファ・プールからディスクに物理的に書き込む時間の合計。                                                                                |
| <b>files_closed</b>              | 閉じられたデータベース・ファイルの総数。                                                                                                     |
| <b>pool_async_data_reads</b>     | バッファ・プールに非同期的に読み込まれるページ数。                                                                                                |
| <b>pool_async_data_writes</b>    | バッファ・プール・データ・ページが、非同期ページ・クリーナ、またはプリフェッチ機能によってディスクに物理的に書き込まれた回数。プリフェッチ機能は先読みされているページのために空き容量を用意するためにダーティ・ページを書き込むことがあります。 |
| <b>pool_async_index_writes</b>   | バッファ・プール索引ページが、非同期ページ・クリーナまたはプリフェッチ機能によってディスクに物理的に書き込まれた回数。プリフェッチ機能は先読みされているページのために空き容量を用意するためにダーティ・ページを書き込むことがあります。     |
| <b>pool_async_index_reads</b>    | プリフェッチ機能により、非同期でバッファ・プールに読み込まれた索引ページの数。                                                                                  |
| <b>pool_async_read_time</b>      | データベース・マネージャのプリフェッチ機能による読み取り所要時間の合計。                                                                                     |
| <b>pool_async_write_time</b>     | データベース・マネージャのページ・クリーナによって、データまたは索引ページがバッファ・プールからディスクに書き込まれた所要時間の合計。                                                      |
| <b>pool_async_data_read_reqs</b> | 非同期読み取り要求の数。                                                                                                             |

| 測定値                            | 説明                                                         |
|--------------------------------|------------------------------------------------------------|
| <b>pool_lsn_gap_clns</b>       | 使用ログ容量が、データベースで定義された範囲に到達したことにより、ページ・クリーナが呼び出された回数。        |
| <b>pool_drty_pg_steal_clns</b> | データベースのビクティム・バッファ置換の間に必要とされた同期書き込みのために、ページ・クリーナが呼び出された回数。  |
| <b>pool_drty_pg_thrsh_clns</b> | バッファ・プールが、データベースのダーティ・ページのしきい値に到達したためにページ・クリーナが呼び出された回数。   |
| <b>prefetch_wait_time</b>      | I/O サーバ（プリフェッチ機能）が、ページをバッファ・プールにロードし終えるまで、アプリケーションが待機した時間。 |
| <b>pool_data_to_estore</b>     | 拡張記憶域にコピーされたバッファ・プール・データ・ページの数。                            |
| <b>pool_index_to_estore</b>    | 拡張記憶域にコピーされたバッファ・プール索引ページの数。                               |
| <b>pool_data_from_estore</b>   | 拡張記憶域からコピーされたバッファ・プール・データ・ページの数。                           |
| <b>pool_index_from_estore</b>  | 拡張記憶域からコピーされたバッファ・プール索引ページの数。                              |
| <b>direct_reads</b>            | バッファ・プールを使用しない読み取り操作の回数。                                   |
| <b>direct_writes</b>           | バッファ・プールを使用しない書き込み操作の回数。                                   |
| <b>direct_read_reqs</b>        | 1つ以上のデータ・セクタで直接読み取りを行うための要求数。                              |
| <b>direct_write_reqs</b>       | 1つ以上のデータ・セクタで直接書き込みを行うための要求数。                              |
| <b>direct_read_time</b>        | 直接読み取りの所要時間（ミリ秒）。                                          |
| <b>direct_write_time</b>       | 直接書き込みの所要時間（ミリ秒）。                                          |
| <b>cat_cache_lookups</b>       | 表記述子情報を取得するためにカタログ・キャッシュが参照された回数。                          |

| 測定値                            | 説明                                                                                                |
|--------------------------------|---------------------------------------------------------------------------------------------------|
| <b>cat_cache_inserts</b>       | システムが、カタログ・キャッシュに表記述子情報の挿入しようとした回数。                                                               |
| <b>cat_cache_overflows</b>     | カタログ・キャッシュ容量がいっぱいであったためにカタログ・キャッシュへの挿入が失敗した回数。                                                    |
| <b>cat_cache_heap_full</b>     | データベース・ヒープでヒープがいっぱいであったために、カタログ・キャッシュへの挿入が失敗した回数。                                                 |
| <b>pkg_cache_lookups</b>       | アプリケーションが、パッケージ・キャッシュでセクションまたはパッケージを検索した回数。データベース・レベルでは、データベースの開始以降、または監視データのリセット以降の参照回数の合計を示します。 |
| <b>pkg_cache_inserts</b>       | 要求セクションが使用できないために、パッケージ・キャッシュにロードされた回数。このカウントには、システムによる暗黙の準備も含まれます。                               |
| <b>pkg_cache_num_overflows</b> | パッケージ・キャッシュが割り当てられたメモリからオーバーフローした回数。                                                              |
| <b>appl_section_lookups</b>    | アプリケーションによる SQL 作業域からの SQL セクションの参照数。                                                             |
| <b>appl_section_inserts</b>    | アプリケーションによる SQL 作業域からの SQL セクションの挿入数。                                                             |
| <b>sec_logs_allocated</b>      | 現在データベースに使用されている 2 次ログ・ファイルの総数。                                                                   |
| <b>log_reads</b>               | ログ機能がディスクから読み取ったログ・ページの数。                                                                         |
| <b>log_writes</b>              | ログ機能がディスクに書き込んだログ・ページの数。                                                                          |
| <b>total_log_used</b>          | データベースで現在使用中のアクティブなログ・スペースの合計 (バイト)。                                                              |
| <b>locks_held</b>              | 現在保持されているロックの数。                                                                                   |
| <b>lock_list_in_use</b>        | 使用中のロック・リスト・メモリの合計 (バイト)。                                                                         |
| <b>deadlocks</b>               | 発生したデッドロックの総数。                                                                                    |

| 測定値                       | 説明                                                                        |
|---------------------------|---------------------------------------------------------------------------|
| <b>lock_escal</b>         | ロックが複数の行ロックから表ロックにエスカレートした回数。                                             |
| <b>x_lock_escal</b>       | ロックが、複数の行ロックから1つの排他的な表ロックにエスカレートした回数、または、行の排他的ロックに起因して、表ロックが排他的ロックになった回数。 |
| <b>lock_timeouts</b>      | オブジェクトをロックする要求が承諾されずにタイムアウトになった回数。                                        |
| <b>lock_waits</b>         | アプリケーションまたは接続がロック発生まで待機した回数の合計。                                           |
| <b>lock_wait_time</b>     | ロックできるまで待機した時間の総計。                                                        |
| <b>locks_waiting</b>      | ロックを待機しているエージェントの数。                                                       |
| <b>rows_deleted</b>       | 行の削除を試みた回数。                                                               |
| <b>rows_inserted</b>      | 行の挿入を試みた回数。                                                               |
| <b>rows_updated</b>       | 行の更新を試みた回数。                                                               |
| <b>rows_selected</b>      | 選択の結果、アプリケーションに戻された行の数。                                                   |
| <b>int_rows_deleted</b>   | 内部活動の結果として、データベースから削除された行の数。                                              |
| <b>int_rows_updated</b>   | 内部活動の結果としてデータベースから更新された行の数。                                               |
| <b>int_rows_inserted</b>  | トリガによって生じた内部活動の結果としてデータベースに挿入された行の数。                                      |
| <b>static_sql_stmts</b>   | 試行された静的 SQL ステートメントの数。                                                    |
| <b>dynamic_sql_stmts</b>  | 試行された動的 SQL ステートメントの数。                                                    |
| <b>failed_sql_stmts</b>   | 試行された SQL ステートメント数のうち、失敗した数。                                              |
| <b>commit_sql_stmts</b>   | 試行された SQL COMMIT ステートメントの総数。                                              |
| <b>rollback_sql_stmts</b> | 試行された SQL ROLLBACK ステートメントの総数。                                            |
| <b>select_sql_stmts</b>   | 実行された SQL SELECT ステートメントの数。                                               |

| 測定値                          | 説明                                                                                                           |
|------------------------------|--------------------------------------------------------------------------------------------------------------|
| <b>uid_sql_stmts</b>         | 実行された SQL UPDATE, INSERT, DELETE ステートメントの数。                                                                  |
| <b>ddl_sql_stmts</b>         | 実行された SQL データ定義言語 (DDL) ステートメントの数。                                                                           |
| <b>int_auto_rebinds</b>      | 試行された自動再バインド (または再コンパイル) の数。                                                                                 |
| <b>int_commits</b>           | データベース・マネージャにより内部的に行われたコミットの総数。                                                                              |
| <b>int_rollback</b>          | データベース・マネージャにより内部的に行われたロールバックの総数。                                                                            |
| <b>int_deadlock_rollback</b> | デッドロックによりデータベース・マネージャが行った強制ロールバックの総数。ロールバックは、デッドロックを解決するために、データベース・マネージャによって選択されたアプリケーション内の現行作業単位に対して実行されます。 |
| <b>binds_precompiles</b>     | 試行されたバインドおよびプリコンパイルの数。                                                                                       |

## アプリケーション

| 測定値                     | 説明                                                                                                |
|-------------------------|---------------------------------------------------------------------------------------------------|
| <b>agents_stolen</b>    | アプリケーションからエージェントが「スチールされた」回数。アプリケーションに関連付けられたアイドル・エージェントが、ほかのアプリケーションに再割り当てされたときに、「スチールされた」と言います。 |
| <b>num_assoc_agents</b> | アプリケーション・レベルでは、アプリケーションに関連付けられているサブエージェントの数。データベース・レベルでは、全アプリケーションのサブエージェントの数。                    |
| <b>total_sorts</b>      | 実行されたソートの総数。                                                                                      |
| <b>total_sort_time</b>  | 実行された全ソートの合計経過時間 (ミリ秒)。                                                                           |



| 測定値                              | 説明                                                                 |
|----------------------------------|--------------------------------------------------------------------|
| <b>sort_overflows</b>            | ソート・ヒープが足りなくなり、一時保存用のディスク領域が必要となったソートの総計。                          |
| <b>total_hash_joins</b>          | 実行されたハッシュ結合の総数。                                                    |
| <b>total_hash_loops</b>          | 利用可能なソート・ヒープ容量よりもハッシュ結合の単一パーティションの方が大きかった回数の総計。                    |
| <b>hash_join_overflows</b>       | ハッシュ結合データが、利用可能なソート・ヒープ容量を超過した回数。                                  |
| <b>hash_join_small_overflows</b> | ハッシュ結合データによる利用可能なソート・ヒープ容量の超過が、10%以下だった回数。                         |
| <b>pool_data_l_reads</b>         | バッファ・プールを通ったデータ・ページの論理読み取り要求の数。                                    |
| <b>pool_data_p_reads</b>         | データ・ページをバッファ・プールに読み込むためにI/Oを必要とした、読み取り要求の数。                        |
| <b>pool_data_writes</b>          | バッファ・プール・データ・ページがディスクに物理的に書き込まれた回数。                                |
| <b>pool_index_l_reads</b>        | バッファ・プールを通った索引ページへの論理読み取り要求の数。                                     |
| <b>pool_index_p_reads</b>        | 索引ページをバッファ・プールに置くための物理的読み取り要求の数。                                   |
| <b>pool_index_writes</b>         | バッファ・プール索引ページがディスクに物理的に書き込まれた回数。                                   |
| <b>pool_read_time</b>            | データまたは索引ページが、ディスクからバッファ・プールに物理的に読み込まれる原因となった読み取り要求の処理にかかった経過時間の合計。 |
| <b>prefetch_wait_time</b>        | I/Oサーバ（プリフェッチ機能）が、ページをバッファ・プールにロードし終えるまで、アプリケーションが待機した時間。          |
| <b>pool_data_to_estore</b>       | 拡張記憶域にコピーされたバッファ・プール・データ・ページの数。                                    |
| <b>pool_index_to_estore</b>      | 拡張記憶域にコピーされたバッファ・プール索引ページの数。                                       |

| 測定値                           | 説明                                                                                                |
|-------------------------------|---------------------------------------------------------------------------------------------------|
| <b>pool_data_from_estore</b>  | 拡張記憶域からコピーされたバッファ・プール・データ・ページの数。                                                                  |
| <b>pool_index_from_estore</b> | 拡張記憶域からコピーされたバッファ・プール索引ページの数。                                                                     |
| <b>direct_reads</b>           | バッファ・プールを使用しない読み取り操作の回数。                                                                          |
| <b>direct_writes</b>          | バッファ・プールを使用しない書き込み操作の回数。                                                                          |
| <b>direct_read_reqs</b>       | 1つ以上のデータ・セクタで直接読み取りを行うための要求数。                                                                     |
| <b>direct_write_reqs</b>      | 1つ以上のデータ・セクタで直接書き込みを行うための要求数。                                                                     |
| <b>direct_read_time</b>       | 直接読み取りの所要時間（ミリ秒）。                                                                                 |
| <b>direct_write_time</b>      | 直接書き込みの所要時間（ミリ秒）。                                                                                 |
| <b>cat_cache_lookups</b>      | 表記述子情報を取得するためにカタログ・キャッシュが参照された回数。                                                                 |
| <b>cat_cache_inserts</b>      | システムが、カタログ・キャッシュに表記述子情報の挿入しようとした回数。                                                               |
| <b>cat_cache_overflows</b>    | カタログ・キャッシュ容量がいっぱいであったためにカタログ・キャッシュへの挿入が失敗した回数。                                                    |
| <b>cat_cache_heap_full</b>    | データベース・ヒープでヒープがいっぱいであったために、カタログ・キャッシュへの挿入が失敗した回数。                                                 |
| <b>pkg_cache_lookups</b>      | アプリケーションが、パッケージ・キャッシュでセクションまたはパッケージを検索した回数。データベース・レベルでは、データベースの開始以降、または監視データのリセット以降の参照回数の合計を示します。 |
| <b>pkg_cache_inserts</b>      | 要求セクションが使用できないために、パッケージ・キャッシュにロードされた回数。このカウントには、システムによる暗黙の準備も含まれます。                               |
| <b>appl_section_lookups</b>   | アプリケーションによる SQL 作業域からの SQL セクションの参照数。                                                             |

| 測定値                         | 説明                                                                        |
|-----------------------------|---------------------------------------------------------------------------|
| <b>appl_section_inserts</b> | アプリケーションによる SQL 作業域からの SQL セクションの挿入数。                                     |
| <b>uow_log_space_used</b>   | 監視されているアプリケーションの現在の作業単位に使用されているログ領域の量 (バイト)。                              |
| <b>locks_held</b>           | 現在保持されているロックの数。                                                           |
| <b>deadlocks</b>            | 発生したデッドロックの総数。                                                            |
| <b>lock_escal</b>           | ロックが複数の行ロックから表ロックにエスカレートした回数。                                             |
| <b>x_lock_escal</b>         | ロックが、複数の行ロックから1つの排他的な表ロックにエスカレートした回数、または、行の排他的ロックに起因して、表ロックが排他的ロックになった回数。 |
| <b>lock_timeouts</b>        | オブジェクトをロックする要求が承諾されずにタイムアウトになった回数。                                        |
| <b>lock_waits</b>           | アプリケーションまたは接続がロック発生まで待機した回数の合計。                                           |
| <b>lock_wait_time</b>       | ロックできるまで待機した時間の総計。                                                        |
| <b>locks_waiting</b>        | ロックを待機しているエージェントの数。                                                       |
| <b>uow_lock_wait_time</b>   | この作業単位がロックを待機した時間の合計。                                                     |
| <b>rows_deleted</b>         | 行の削除を試みた回数。                                                               |
| <b>rows_inserted</b>        | 行の挿入を試みた回数。                                                               |
| <b>rows_updated</b>         | 行の更新を試みた回数。                                                               |
| <b>rows_selected</b>        | 選択の結果、アプリケーションに戻された行の数。                                                   |
| <b>rows_written</b>         | 表内で変更 (挿入, 削除, 更新) があつた行の数。                                               |
| <b>rows_read</b>            | 表から読み取られた行数。                                                              |
| <b>int_rows_deleted</b>     | 内部活動の結果として、データベースから削除された行の数。                                              |
| <b>int_rows_updated</b>     | 内部活動の結果としてデータベースから更新された行の数。                                               |

| 測定値                       | 説明                                                             |
|---------------------------|----------------------------------------------------------------|
| <b>int_rows_inserted</b>  | トリガによって生じた内部活動の結果としてデータベースに挿入された行の数。                           |
| <b>open_rem_curs</b>      | このアプリケーションで開いている「open_rem_curs_blk」でカウントされるカーソルも含めたリモート・カーソル数。 |
| <b>open_rem_curs_blk</b>  | このアプリケーションで現在開いているリモート・ブロッキング・カーソル数。                           |
| <b>rej_curs_blk</b>       | サーバで I/O ブロック要求が拒否され、ノン・ブロック I/O に変換された総数。                     |
| <b>acc_curs_blk</b>       | I/O ブロック要求が受け入れられた回数。                                          |
| <b>open_loc_curs</b>      | このアプリケーションで開いている「open_loc_curs_blk」でカウントされるカーソルも含めたローカル・カーソル数。 |
| <b>open_loc_curs_blk</b>  | このアプリケーションで開いているローカル・ブロッキング・カーソル数。                             |
| <b>static_sql_stmts</b>   | 試行された静的 SQL ステートメントの数。                                         |
| <b>dynamic_sql_stmts</b>  | 試行された動的 SQL ステートメントの数。                                         |
| <b>failed_sql_stmts</b>   | 試行された SQL ステートメント数のうち、失敗した数。                                   |
| <b>commit_sql_stmts</b>   | 試行された SQL COMMIT ステートメントの総数。                                   |
| <b>rollback_sql_stmts</b> | 試行された SQL ROLLBACK ステートメントの総数。                                 |
| <b>select_sql_stmts</b>   | 実行された SQL SELECT ステートメントの数。                                    |
| <b>uid_sql_stmts</b>      | 実行された SQL UPDATE, INSERT, DELETE ステートメントの数。                    |
| <b>ddl_sql_stmts</b>      | 実行された SQL データ定義言語 (DDL) ステートメントの数。                             |
| <b>int_auto_rebinds</b>   | 試行された自動再バインド (または再コンパイル) の数。                                   |
| <b>int_commits</b>        | データベース・マネージャにより内部的に行われたコミットの総数。                                |

| 測定値                          | 説明                                                                                                           |
|------------------------------|--------------------------------------------------------------------------------------------------------------|
| <b>int_rollback</b>          | データベース・マネージャにより内部的に行われたロールバックの総数。                                                                            |
| <b>int_deadlock_rollback</b> | デッドロックによりデータベース・マネージャが行った強制ロールバックの総数。ロールバックは、デッドロックを解決するために、データベース・マネージャによって選択されたアプリケーション内の現行作業単位に対して実行されます。 |
| <b>binds_precompiles</b>     | 試行されたバインドおよびプリコンパイルの数。                                                                                       |

---

**注：**ファイアウォール越しに Oracle サーバを監視するために使用するポートは、Oracle サーバの構成によって異なります。クライアントとサーバの間の接続の設定情報は、Oracle クライアントの **tnsnames.ora** ファイルにあります。

---

## Oracle モニタの設定

Oracle サーバ・モニタでは、Oracle の V\$ テーブルである V\$SESSTAT および V\$SYSSTAT のほか、ユーザ定義クエリーでユーザが定義したその他のカウンタの情報が測定されます。Oracle サーバを監視するには、次の手順に従って監視環境を設定する必要があります。

---

**注：**ファイアウォール越しに Oracle サーバを監視するために使用するポートは、Oracle サーバの構成によって異なります。クライアントとサーバの間の接続の設定情報は、Oracle クライアントの **tnsnames.ora** ファイルにあります。

---

**Oracle モニタ環境を設定するには、次の手順で行います。**

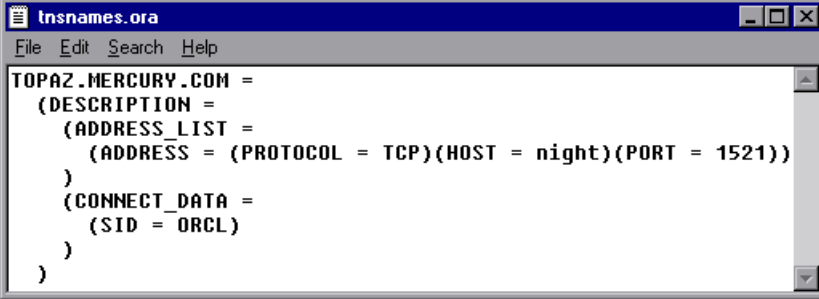
- 1 コントローラ・マシンに Oracle クライアント・ライブラリがインストールされていることを確認します。
- 2 パス環境変数に **%OracleHome%\bin** が含まれていることを確認します。含まれていない場合は、追加します。
- 3 コントローラ・マシンで、Oracle クライアントが監視対象の Oracle サーバと通信できるように **tnsnames.ora** ファイルを設定します。

接続パラメータは、テキスト・エディタを使って **tnsnames.ora** ファイルを手作業で編集するか、Oracle サービス設定ツールを使用（たとえば [スタート] > [プログラム] > [Oracle for Windows NT] > [Oracle Net8 Easy Config] を選択）して設定できます。

次の項目を指定します。

- ▶ Oracle インスタンスの新しいサービス名（TNS 名）
- ▶ TCP プロトコル
- ▶ ホスト名（監視対象のサーバ・マシンの名前）
- ▶ ポート番号（通常は 1521）
- ▶ データベース SID（標準設定の SID は ORCL）

例を次に示します。



```
tnsnames.ora
File Edit Search Help
TOPAZ.MERCURY.COM =
 (DESCRIPTION =
 (ADDRESS_LIST =
 (ADDRESS = (PROTOCOL = TCP)(HOST = night)(PORT = 1521))
)
 (CONNECT_DATA =
 (SID = ORCL)
)
)
)
```

---

注：Oracle モニタを実行するコントローラ・マシンには、32 ビットの Oracle クライアントだけをインストールします。コントローラ・マシンに 16 ビットと 32 ビットの Oracle クライアントが両方インストールされている場合は、16 ビットのクライアントをアンインストールしておきます。

---

- 4 データベース管理者からサービス用のユーザ名とパスワードを入手して、コントローラに Oracle の V\$ テーブル (V\$SESSTAT, V\$SYSSTAT, V\$STATNAME, V\$INSTANCE, V\$SESSION) に対するデータベース管理者権限を付与します。
- 5 コントローラ・マシンから **tns ping** を実行して、Oracle サーバとの接続を確認します。Oracle サーバにアクセスするアプリケーション・サーバからの通信を制限する DMZ またはファイアウォールの背後に Oracle サーバがある場合は、接続に問題が起こることがあります。
- 6 レジストリが、現在使用している Oracle のバージョンに対応して更新されていることと、HKEY\_LOCAL\_MACHINE¥SOFTWARE¥ORACLE キーがあることを確認します。
- 7 監視対象の Oracle サーバが起動されていて、稼動中であることを確認します。

---

注：同時に複数の Oracle データベース・サーバを監視できます。

---

- 8 コントローラ・マシンから SQL\*Plus を実行して、該当するユーザ名 / パスワード / サーバの組み合わせを使って Oracle サーバ (1 台または複数) にログインしてみます。
- 9 `SELECT * FROM V$SYSSTAT` と入力して、Oracle サーバの V\$SYSSTAT テーブルを表示できることを確認します。同様のクエリーを使って、Oracle サーバの V\$SESSTAT, V\$SESSION, V\$INSTANCE, V\$STATNAME, V\$PROCESS テーブルを表示できることを確認します。Oracle の bin ディレクトリが検索パスに含まれていることを確認します。
- 10 各モニタのサンプリング間隔 (秒単位) を変更するには、LoadRunner のインストール先フォルダにある `dat\monitors\mmon.cfg` ファイルを編集する必要があります。標準設定の間隔は 10 秒です。

---

**注 :** Oracle モニタのサンプリング間隔の最小値は 10 秒。サンプリング間隔を 10 秒未満に設定すると、Oracle モニタは 10 秒間隔で監視を続けます。

---

---

**注 :** Oracle 環境の設定で不具合が発生した場合は、Oracle サーバによって発行されるエラー・メッセージを参照してください。

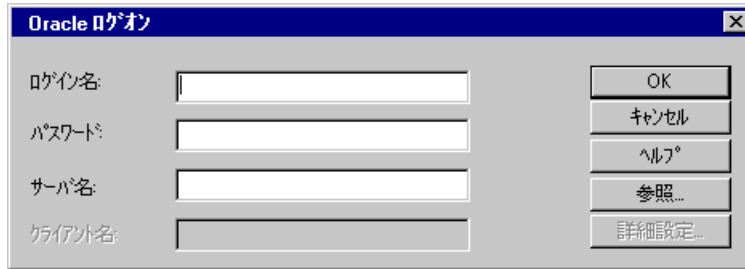
---

**Oracle モニタを設定するには、次の手順で行います。**

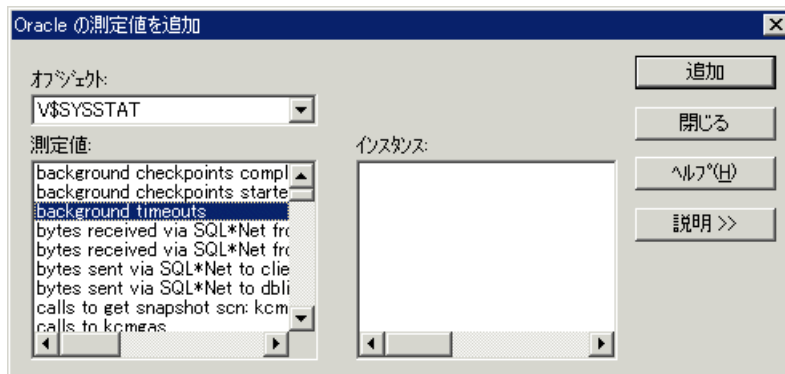
- 1 グラフ・ツリーで [Oracle] グラフをクリックして、[実行] ウィンドウの右の表示枠までドラッグします。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Oracle] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。任意のプラットフォームを選択して、[OK] をクリックします。
- 4 [Oracle] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。



[Oracle ログオン] ダイアログ・ボックスが開きます。



- 5 [ログイン名], [パスワード], [サーバ名] を入力して, [OK] をクリックします。[Oracle の測定値を追加] ダイアログ・ボックスが開きます。



- 6 オブジェクト, 測定値, インスタンスを選択します。Ctrl キーを使って複数の測定値を選択できます。インスタンスが意味を持つのは, 強調表示された測定値の複数のインスタンスが実行されている場合だけです。各測定値の説明を表示するには, [説明 >>] をクリックしてダイアログ・ボックスを拡張します。ユーザ定義クエリーの作成手順については, 499 ページ「ユーザ定義クエリー」を参照してください。

使用可能なパフォーマンス・カウンタの一覧については, 498 ページ「Oracle パフォーマンス・カウンタ」を参照してください。

- 7 [追加] をクリックして, 選択した測定値をリソースのリストに追加します。必要なリソースをすべてリストに追加したら, [閉じる] をクリックします。
- 8 [Oracle] 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。

注：標準設定では、データベースはカウンタの絶対値を返します。ただし、**dat%monitors%vmon.cfg** ファイルの **IsRate** 設定を 1 に変更することで、カウンタの変化率（単位時間ごとのカウンタの変化）を報告するようにデータベースに指定できます。

## Oracle パフォーマンス・カウンタ

Oracle サーバの監視時に最も一般的に使用される測定項目を以下に示します（V\$SYSSTAT テーブルから）。

| 測定値                                           | 説明                                                                                                                                                                                                               |
|-----------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CPU used by this session</b>               | ユーザ呼び出しの開始から終了までの間にセッションによって使用される CPU 時間（10 ミリ秒単位）。ユーザ呼び出しの中には 10 ミリ秒以内に完了するものもあり、結果として、ユーザ呼び出しの開始時間と終了時間が同じになることがあります。その場合は、統計値に 0 ミリ秒が加えられます。特にコンテキスト・スイッチが多く発生しているシステムでは、オペレーティング・システムのレポートで同様の問題が起こることがあります。 |
| <b>Bytes received via SQL*Net from client</b> | Net8 を介してクライアントから受信した総バイト数。                                                                                                                                                                                      |
| <b>Logons current</b>                         | 現在の総ログイン数。                                                                                                                                                                                                       |
| <b>Opens of replaced files</b>                | プロセスのファイル・キャッシュになかったため、再度開く必要があったファイルの総数。                                                                                                                                                                        |
| <b>User calls</b>                             | Oracle では、ログイン、解析、または実行するごとに、関連するユーザ呼び出しのデータ構造を追跡するために、リソース（Call State Objects）が割り当てられます。動作状況を確認するときは、RPI 呼び出しに対するユーザ呼び出しの割合を調べれば、ユーザが Oracle に送信している要求の種類に応じて、どの程度の内部作業が発生しているかがわかります。                          |
| <b>SQL*Net roundtrips to/from client</b>      | クライアントと送受信された Net8 メッセージの総数。                                                                                                                                                                                     |

| 測定値                                     | 説明                                                                                                                                                                                                                           |
|-----------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Bytes sent via SQL*Net to client</b> | フォアグラウンド・プロセスからクライアントに送信された総バイト数。                                                                                                                                                                                            |
| <b>Opened cursors current</b>           | 現在オープンしているカーソルの総数。                                                                                                                                                                                                           |
| <b>DB block changes</b>                 | この統計値は、一貫性維持と密接に関連しており、更新および削除作業の対象となったすべてのブロックに対してSGAにおいて加えられた変更の総数がカウントされます。これらの変更によってREDOログ・エントリが生成されており、トランザクションがコミットされると、データベースに対して変更が確定されます。この値は、データベースの大まかな総作業量を示します。また、バッファに変更が加えられる割合を(場合によってはトランザクションごとのレベルで)示します。 |
| <b>Total file opens</b>                 | インスタンスによって実行されているファイル・オープンの総数。各プロセスは、データベースを操作するために、多くのファイル(コントロール・ファイル、ログ・ファイル、データベース・ファイル)を必要とします。                                                                                                                         |

### ユーザ定義クエリー

ユーザ定義クエリー機能を使って、Oracle データベースを対象に独自のクエリーを定義して、Oracle オンライン・モニタ・グラフにクエリーの結果(1つの数値)を表示できます。独自のクエリーを定義することで、Oracle モニタによって現在提供されているV\$SYSSTAT およびV\$SESSTAT テーブルのカウントだけでなく、パフォーマンス情報を含むほかのテーブルも監視できます。

ユーザ定義クエリーを作成するには、次の手順で行います。

- 1 **vmon.cfg** ファイルの 3 行目にある **CustomCounters=** は、作成するユーザ定義カウンタの数を示します。
- 2 **vmon.cfg** ファイルに、新しいカウンタ用の新しいセクションを作成します。各セクションの形式は次のとおりです。

```
[Custom2]
```

```
Name=Number of sessions
```

```
Description=This counter returns the number of sessions active.
```

```
Query=SELECT COUNT(*) FROM V$SESSION
```

```
IsRate=1
```

- 3 [Custom#] 行には、新しいユーザ定義カウンタに、次の順番のカウンタ番号を指定します。ユーザ定義カウンタは 0 から始まる連番でなければなりません。
- 4 Name 行には、新しいカウンタ名を入力します。
- 5 Description 行には、ヘルプ・メッセージに含めるカウンタの説明を入力します。
- 6 Query 行には、データベースから 1 行だけを返す SQL クエリーのテキストを (**vmon.cfg** ファイルの複数行に分けずに 1 行として) 入力します。返される行には 1 つの列 (数値) が含まれている必要があります。

---

注：ユーザ定義クエリーは 512 文字を越えてはなりません。

---

- 7 IsRate 行には、カウンタを絶対値として報告するようにデータベースに対して指定する場合は 0 を入力します。単位時間ごとのカウンタの変化を報告するようにデータベースに指定するには、1 を入力します。

---

注：ユーザ定義クエリーが負の値を返してはなりません。

---

## SQL Server モニタの設定

SQL Server モニタは、SQL Server マシンの標準の Windows リソースを測定します。

---

**注：**ファイアウォール越しに SQL Server を監視するには、139 番ポートで TCP を使用します。

---

**SQL Server モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [SQL Server] グラフをクリックし、[実行] ウィンドウの右の表示枠までドラッグします。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [MS SQL サーバ] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [MS SQL サーバ] ダイアログ・ボックスの [リソース測定場所] セクションで、監視対象の測定値を選択します。

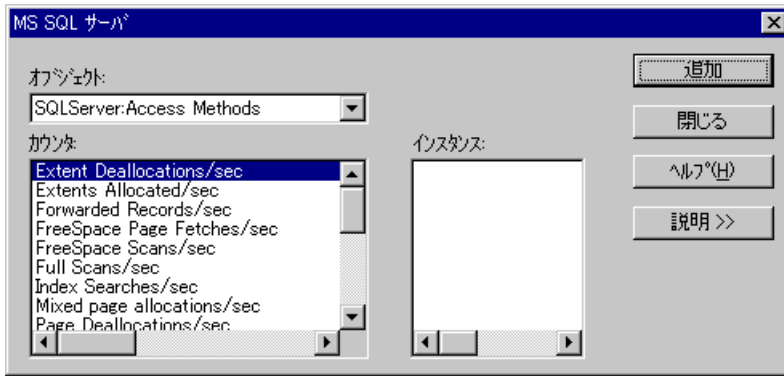
使用可能なパフォーマンス・カウンタの一覧については、503 ページを参照してください。

---

**注：**SQL Server モニタの標準設定のカウンタを変更するには、669 ページ「モニタの標準設定のカウンタの変更」を参照してください。

---

- 追加の測定値を選択するには、**[追加]** をクリックします。ダイアログ・ボックスが開き、SQL Server オブジェクト、そのカウンタ、インスタンスが表示されます。



- カウンタとインスタンスを選択します。**Ctrl** キーを使って、複数のカウンタを選択できます。インスタンスが意味を持つのは、強調表示されたカウンタの複数のインスタンスが実行されている場合だけです。各カウンタの説明を表示するには、**[説明 >>]** をクリックしてダイアログ・ボックスを拡張します。
- [追加]** をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、**[閉じる]** をクリックします。
- [MS SQL サーバ]** 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。

---

**注：**カウンタの中には、サーバでの最初の負荷テストで、SQL Server のパフォーマンスを調べたり、ボトルネックの原因を特定したりするのに特に便利なカウンタもあります。これらのカウンタの詳細については、670 ページ「負荷テストに役立つカウンタ」を参照してください。

---

## SQL Server パフォーマンス・カウンタ

次の表に、SQL Server 6.5 で監視可能な標準のカウンタを示します。

| 測定値                                    | 説明                                                                                                                                                                                                                                                                                                                                                                         |
|----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>% Total Processor Time (NT)</b>     | システム上のすべてのプロセッサが非アイドル・スレッドを実行するために使用中となる時間の平均的な割合。マルチプロセッサ・システムで、すべてのプロセッサが常にビジー状態ならば、この値は 100% です。すべてのプロセッサが 50% の時間だけビジー状態ならば、この値は 50% です。4 分の 1 のプロセッサが 100% の時間ビジー状態ならば、この値は 25% です。この値は、何らかの処理を行うために費やされた時間の割合です。各プロセッサにはアイドル・プロセス内のアイドル・スレッドが割り当てられます。アイドル・スレッドによって、ほかのスレッドが使用していない非生産的なプロセッサ・サイクルが消費されます。                                                           |
| <b>% Processor Time (Windows 2000)</b> | プロセッサが非アイドル・スレッドを実行している時間の割合。このカウンタは、プロセッサの動作状況を示す重要な指標となります。この値は、プロセッサがアイドル・プロセスのスレッドを実行するのに費やす時間をサンプリング間隔ごとに測定し、その値を 100% から引くことによって算出されます（各プロセッサには、ほかのスレッドの実行準備ができていないときにサイクルを消費するアイドル・スレッドが割り当てられています）。この値は、あるサンプリング時点から次のサンプリング時点までの間に何らかの有用な処理を行うために費やされ時間の割合です。このカウンタは、サンプリング間隔の間に観察されたビジー状態の時間の平均的な割合を示します。この値は、サービスがアクティブではなかった時間を監視し、その値を 100% から引くことによって算出されます。 |
| <b>Cache Hit Ratio</b>                 | 要求されたデータ・ページが（ディスクから読み出される代わりに）データ・キャッシュで見つかった回数の割合。                                                                                                                                                                                                                                                                                                                       |
| <b>I/O - Batch Writes/sec</b>          | バッチ I/O によってディスクに書き出される 2 KB ページの秒ごとの数。バッチ I/O は、主にチェックポイント・スレッドが使用します。                                                                                                                                                                                                                                                                                                    |
| <b>I/O-Lazy Writes/sec</b>             | レイジー・ライタによってディスクにフラッシュされる 2 KB ページの秒ごとの数。                                                                                                                                                                                                                                                                                                                                  |

| 測定値                             | 説明                                 |
|---------------------------------|------------------------------------|
| <b>I/O - Outstanding Reads</b>  | 保留されている物理読み取りの数。                   |
| <b>I/O - Outstanding Writes</b> | 保留されている物理書き込みの数。                   |
| <b>I/O - Page Reads/sec</b>     | 物理ページ読み取りの秒ごとの数。                   |
| <b>I/O - Transactions/sec</b>   | 実行された Transact-SQL コマンド・バッチの秒ごとの数。 |
| <b>User Connections</b>         | オープン・ユーザ接続の数。                      |

Sybase モニタでは、Windows および UNIX 上の Sybase Adaptive Server Enterprise (Sybase ASE) サーバ (バージョン 11 以降) を監視できます。このモニタは、Adaptive Server Enterprise Monitor Server 経由で Sybase ASE サーバに接続し、Sybase に組み込まれている標準ライブラリを使ってサーバから測定値を取得します。

## Sybase モニタの設定

Sybase モニタでは、Windows および UNIX 上の Sybase Adaptive Server Enterprise (Sybase ASE) サーバ (バージョン 11 以降) を監視できます。このモニタは、Adaptive Server Enterprise Monitor Server 経由で Sybase ASE サーバに接続し、Sybase に組み込まれている標準ライブラリを使ってサーバから測定値を取得します。

---

**注：**監視対象のサーバに接続するときは、Sybase ASE サーバではなく Adaptive Server Enterprise Monitor Server に接続します。Adaptive Server Enterprise Monitor Server は、Sybase ASE サーバと同じマシンで実行され、Sybase ASE サーバからパフォーマンス情報を取得するアプリケーションです。通常、Adaptive Server Enterprise Monitor Server の名前は Sybase サーバと同じで、サフィックス **\_ms** が付きます。

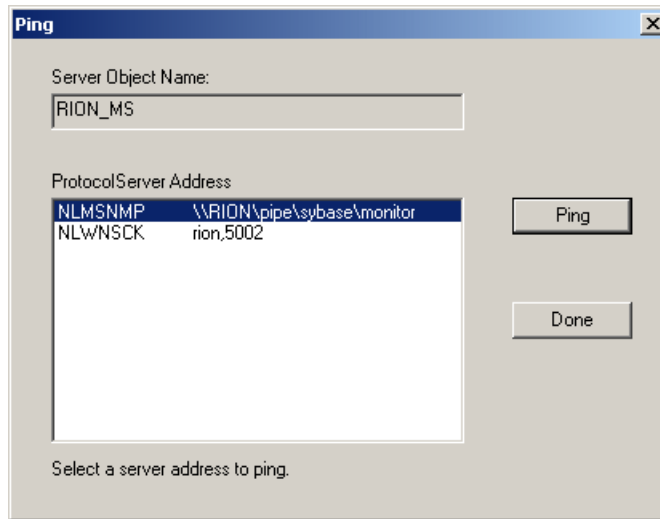
---

Sybase ASE サーバを監視するには、まず Sybase モニタ環境を設定する必要があります。



**Sybase モニタ環境を設定するには、次の手順で行います。**

- 1 コントローラ・マシンに Sybase クライアント・ファイルとライブラリをインストールします。
- 2 コントローラ・マシンでクライアントとサーバの間の接続を検証します。接続を検証するには、Sybase クライアントの **dsedit** ツールを使って、Adaptive Server Enterprise Monitor Server に対して ping を実行します。



---

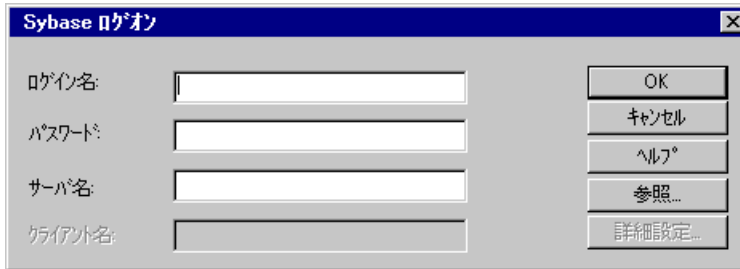
**注：**ファイアウォール越しに Sybase サーバを監視するために使用するポートは、Sybase サーバの構成によって異なります。クライアントとサーバの間の接続の設定情報は、Sybase クライアントの **sql.ini** ファイルにあります。

---

**Sybase ASE モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [Sybase] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。

- 3 [Sybase] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。任意のプラットフォームを選択して、[OK] をクリックします。
- 4 [Sybase] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。  
[Sybase ログオン] ダイアログ・ボックスが開きます。



- 5 Sybase ASE サーバの管理者権限を持っているユーザのログイン名とパスワード、および Adaptive Server Enterprise Monitor Server の名前（通常は Sybase サーバと同じ名前前でサフィックス `_ms` が付く）を入力します。
- 6 [OK] をクリックします。[Sybase の測定値の追加] ダイアログ・ボックスが開きます。



- 7 オブジェクト、測定値、インスタンスを選択します。**Ctrl** キーを使って複数の測定値を選択できます。インスタンスが意味を持つのは、強調表示された測定値の複数のインスタンスが実行されている場合だけです。各測定値の説明を表示するには、[説明 >>] をクリックしてダイアログ・ボックスを拡張します。

使用可能なパフォーマンス・カウンタの一覧については、507 ページを参照してください。

- 8 [追加] をクリックして、選択した測定値をリソースのリストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。
- 9 [Sybase] 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。

### Sybase パフォーマンス・カウンタ

Sybase サーバの監視には、次の測定値が利用できます。

| 対象      | 測定値                        | 説明                            |
|---------|----------------------------|-------------------------------|
| Network | Average packet size (Read) | 受信したネットワーク・パケットの数。            |
|         | Average packet size (Send) | 送信したネットワーク・パケットの数。            |
|         | Network bytes (Read)       | サンプリング間隔の間に受信したバイト数。          |
|         | Network bytes (Read)/sec   | 秒ごとの受信バイト数。                   |
|         | Network bytes (Send)       | サンプリング間隔の間に送信したバイト数。          |
|         | Network bytes (Send)/sec   | 秒ごとの送信バイト数。                   |
|         | Network packets (Read)     | サンプリング間隔の間に受信したネットワーク・パケットの数。 |
|         | Network packets (Read)/sec | 秒ごとの受信ネットワーク・パケット数。           |
|         | Network packets (Send)     | サンプリング間隔の間に送信されたネットワーク・パケット数。 |
|         | Network packets (Send)/sec | 秒ごとの送信ネットワーク・パケット数。           |
| Memory  | Memory                     | ページ・キャッシュに割り当てられたメモリ量 (バイト)。  |

| 対象                       | 測定値                                | 説明                                   |
|--------------------------|------------------------------------|--------------------------------------|
| <b>Disk</b>              | Reads                              | データベース・デバイスからの読み取り数。                 |
|                          | Writes                             | データベース・デバイスへの書き込み数。                  |
|                          | Waits                              | デバイスへのアクセスを待機しなければならなかった回数。          |
| <b>Disk</b>              | Grants                             | デバイスへのアクセスが許可された回数。                  |
| <b>Engine</b>            | Server is busy (%)                 | Adaptive Server がビジー状態である時間の割合。      |
|                          | CPU time                           | エンジンでビジー時間がどれだけあったか。                 |
|                          | Logical pages (Read)               | キャッシュまたはデータベース・デバイスからのデータ・ページの読み込み数。 |
|                          | Pages from disk (Read)             | データ・キャッシュから行われなかったデータ・ページの読み込み数。     |
|                          | Pages stored                       | データベース・デバイスに書き込まれなかったデータ・ページ数。       |
| <b>Stored Procedures</b> | Executed (sampling period)         | サンプリング間隔の間にストアド・プロシージャが実行された回数。      |
|                          | Executed (session)                 | セッション中にストアド・プロシージャが実行された回数。          |
|                          | Average duration (sampling period) | サンプリング間隔の間にストアド・プロシージャの実行に要した時間 (秒)。 |
|                          | Average duration (session)         | セッション中に、ストアド・プロシージャの実行に要した時間 (秒)。    |
| <b>Locks</b>             | % Requests                         | 完了したロック要求の割合。                        |
|                          | Locks count                        | ロック数。これは、累積値です。                      |

| 対象             | 測定値                       | 説明                                               |
|----------------|---------------------------|--------------------------------------------------|
| <b>Locks</b>   | Granted immediately       | ほかのロックが解放されるのを待たずに、すぐに承認されたロックの数。                |
|                | Granted after wait        | ほかのロックが解放されるのを待ってから承認されたロックの数。                   |
|                | Not granted               | 要求されたにもかかわらず承認されなかったロックの数。                       |
|                | Wait time (avg.)          | ロックの平均待ち時間。                                      |
| <b>SqlSrvr</b> | Locks/sec                 | ロック数。これは、累積値です。                                  |
|                | % Processor time (server) | Adaptive Server がビジー状態である時間の割合。                  |
|                | Transactions              | コミットされた Transact-SQL ステートメント・ブロック (トランザクション) の数。 |
|                | Deadlocks                 | デッドロックの数を報告します。                                  |
| <b>Cache</b>   | % Hits                    | データ・ページの読み込みが、物理的ページ読み込みではなく、キャッシュから行われた回数の割合。   |
|                | Pages (Read)              | キャッシュまたはデータベース・デバイスからのデータ・ページの読み込み数。             |
|                | Pages (Read)/sec          | キャッシュまたはデータベース・デバイスからのデータ・ページの秒ごとの読み込み数。         |

| 対象                 | 測定値                        | 説明                                                              |
|--------------------|----------------------------|-----------------------------------------------------------------|
|                    | Pages from disk (Read)     | データ・キャッシュから行われなかったデータ・ページの読み込み数。                                |
|                    | Pages from disk (Read)/sec | データ・キャッシュから行われなかった、データ・ページの秒ごとの読み込み数。                           |
|                    | Pages (Write)              | データベース・デバイスに書き込まれなかったデータ・ページ数。                                  |
|                    | Pages (Write)/sec          | データベース・デバイスに書き込まれた秒ごとのデータ・ページ数。                                 |
| <b>Process</b>     | % Processor time (process) | 全プロセスが「実行」状態になっている時間に対して、アプリケーションを実行しているプロセスが「実行」状態になっている時間の割合。 |
|                    | Locks/sec                  | プロセスごとにロックの数。これは、累積値です。                                         |
|                    | % Cache hit                | プロセスによって、データ・ページの読み込みが、物理的ページ読み込みではなく、キャッシュから行われた回数の割合。         |
|                    | Pages (Write)              | プロセスによってデータベース・デバイスに書き込まれたデータ・ページ数。                             |
| <b>Transaction</b> | Transactions               | セッション中にコミットされた Transact-SQL ステートメント・ブロック (トランザクション) の数。         |
|                    | Rows (Deleted)             | セッション中にデータ・テーブルから削除された行数。                                       |

| 対象                 | 測定値                  | 説明                                                        |
|--------------------|----------------------|-----------------------------------------------------------|
| <b>Transaction</b> | Inserts              | セッション中にデータベースに挿入があった回数。                                   |
|                    | Updates              | セッション中のデータ・テーブルへの更新。                                      |
|                    | Updates in place     | セッション中の選択範囲内、および選択範囲外のサイズの大きい更新の総計（遅延された更新を除く）。           |
|                    | Transactions/sec     | セッション中にコミットされた Transact-SQL ステートメント・ブロック（トランザクション）の秒ごとの数。 |
|                    | Rows (Deleted)/sec   | データベース・テーブルから削除された秒ごとの行数。                                 |
|                    | Inserts/sec          | データベース・テーブルへの挿入の秒ごとの数。                                    |
|                    | Updates/sec          | データベース・テーブルへの更新の秒ごとの数。                                    |
|                    | Updates in place/sec | 選択範囲内、および選択範囲外のサイズの大きい更新の秒ごとの総計（遅延された更新を除く）。              |





# 第 26 章

---

## ストリーミング・メディアの監視

サーバとクライアントのパフォーマンスのボトルネックを特定するために、シナリオ実行時に Windows Media サーバおよび RealPlayer オーディオ/ビデオ・サーバのほか、RealPlayer および Media Player クライアントを監視できます。

本章では、以下の項目について説明します。

- ▶ Windows Media サーバ・モニタの設定
- ▶ RealPlayer サーバ・モニタの設定
- ▶ RealPlayer クライアント・オンライン・グラフの表示
- ▶ Media Player クライアント・オンライン・グラフの表示

---

注：ストリーミング・メディア関数を含むスクリプトの記録手順については、『**LoadRunner 仮想ユーザ・スクリプトの作成**』を参照してください。

---

### ストリーミング・メディアの監視について

ストリーミング・メディア・モニタは、Windows Media サーバと RealPlayer オーディオ/ビデオ・サーバのほか、RealPlayer および Media Player クライアントのパフォーマンス情報を提供します。Windows Media サーバおよび RealPlayer サーバのパフォーマンス・データを取得するには、シナリオを実行する前にストリーミング・メディア・モニタを起動して、監視対象の統計値と測定値を指定する必要があります。RealPlayer クライアントと Media Player クライアントの場合には、事前にシナリオを起動したり設定したりする必要はありません。

## Windows Media サーバ・モニタの設定

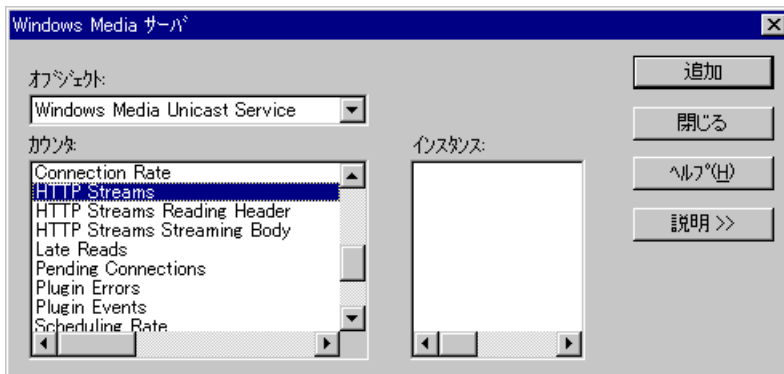
Windows Media サーバを監視するには、まず Windows Media サーバ・モニタで測定するカウンタを選択する必要があります。これらのカウンタは、[Windows Media サーバ] ダイアログ・ボックスを使って選択します。

**Windows Media サーバ・モニタの設定は、次の手順で行います。**

- 1 グラフ・ツリーで [Windows Media サーバ] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Windows Media サーバ] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [Windows Media サーバ] ダイアログ・ボックスの [リソース測定場所] セクションで、監視対象の測定値を選択します。

測定可能なパフォーマンス・カウンタの一覧については、515 ページを参照してください。

- 5 追加の測定値を選択するには、[追加] をクリックします。[Windows Media サーバ] ダイアログ・ボックスが開き、Windows Media ユニキャスト・サービス・オブジェクト、そのカウンタ、インスタンスが表示されます。



- 6 カウンタとインスタンスを選択します。**Ctrl** キーを使って、複数のカウンタを選択できます。インスタンスが意味を持つのは、強調表示されたカウンタの複数のインスタンスが実行されている場合だけです。各カウンタの説明を表示するには、[説明 >>] をクリックしてダイアログ・ボックスを拡張します。
- 7 [追加] をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。
- 8 [Windows Media サーバ] 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。

## Windows Media サーバ・パフォーマンス・カウンタ

以下の表に、標準で監視可能なカウンタの説明を示します。

| 測定値                                   | 説明                                                                                      |
|---------------------------------------|-----------------------------------------------------------------------------------------|
| Active Live Unicast Streams (Windows) | ストリーミングされているライブ・ユニキャスト・ストリームの数。                                                         |
| Active Streams                        | ストリーミングされているストリームの数。                                                                    |
| Active TCP Streams                    | ストリーミングされている TCP ストリームの数。                                                               |
| Active UDP Streams                    | ストリーミングされている UDP ストリームの数。                                                               |
| Aggregate Read Rate                   | ファイル読み取りの合計速度 (バイト/秒)。                                                                  |
| Aggregate Send Rate                   | ストリーム伝送の合計速度 (バイト/秒)。                                                                   |
| Connected Clients                     | サーバに接続されたクライアントの数。                                                                      |
| Connection Rate                       | クライアントがサーバに接続されている速度。                                                                   |
| Consoles                              | 現在サーバに接続されているコントローラの数。                                                                  |
| HTTP Streams                          | ストリーミングされている HTTP ストリームの数。                                                              |
| Late Reads                            | 秒ごとの遅延読み込み完了数。                                                                          |
| Pending Connections                   | サーバへの接続を試みているが、まだ接続していないクライアントの数。サーバが許容量の上限に近い状態で稼働していて、多数の接続要求をすばやく処理できないと、この値が高くなります。 |
| Stations                              | 現在サーバに存在するステーション・オブジェクトの数。                                                              |

| 測定値           | 説明                        |
|---------------|---------------------------|
| Streams       | 現在サーバに存在するストリーム・オブジェクトの数。 |
| Stream Errors | 発生したエラーの秒ごとの累積数。          |

## RealPlayer サーバ・モニタの設定

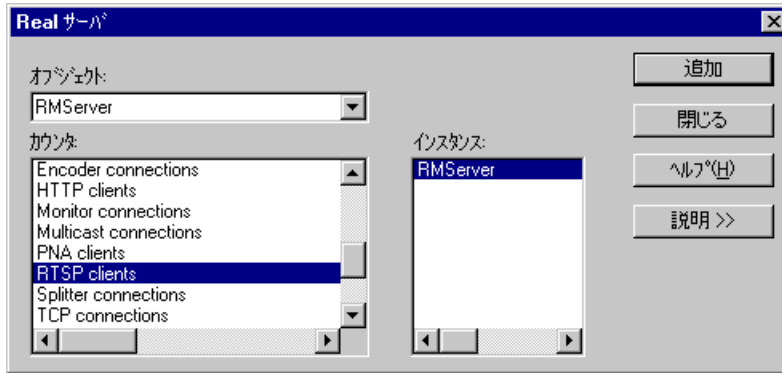
RealPlayer サーバを監視するには、まず RealPlayer サーバ・モニタで測定するカウンタを選択する必要があります。これらのカウンタは、[Real サーバ] ダイアログ・ボックスを使って選択します。

**RealPlayer サーバ・モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [Real サーバ] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Real サーバ] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

- 4 [Real サーバ] ダイアログ・ボックスの [リソース測定場所] の下の項目で [追加] をクリックして、監視対象の測定値を選択します。

[Real サーバ] ダイアログ・ボックスがもう 1 つ開き、監視可能なカウンタが表示されます。



- 5 カウンタとインスタンスを選択します。Ctrl キーを使って、複数のカウンタを選択できます。インスタンスが意味を持つのは、強調表示されたカウンタの複数のインスタンスが実行されている場合だけです。各カウンタの説明を表示するには、[説明 >>] をクリックしてダイアログ・ボックスを拡張します。

測定可能なパフォーマンス・カウンタの一覧については、518 ページを参照してください。

- 6 [追加] をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。
- 7 [Real サーバ] 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。

## Real サーバ・パフォーマンス・カウンタ

以下の表に、標準で監視可能なカウンタの説明を示します。

| 測定値                          | 説明                          |
|------------------------------|-----------------------------|
| <b>Encoder Connections</b>   | アクティブなエンコーダ接続の数。            |
| <b>HTTP Clients</b>          | HTTP を使用しているアクティブなクライアントの数。 |
| <b>Monitor Connections</b>   | アクティブなサーバ・モニタ接続の数。          |
| <b>Multicast Connections</b> | アクティブなマルチキャスト接続の数。          |
| <b>PNA Clients</b>           | PNA を使用しているアクティブなクライアントの数。  |
| <b>RTSP Clients</b>          | RTSP を使用しているアクティブなクライアントの数。 |
| <b>Splitter Connections</b>  | アクティブなスプリッタ接続の数。            |
| <b>TCP Connections</b>       | アクティブな TCP 接続の数。            |
| <b>Total Bandwidth</b>       | 消費されている秒ごとのビット数。            |
| <b>Total Clients</b>         | アクティブなクライアントの総数。            |
| <b>UDP Clients</b>           | アクティブな UDP 接続の数。            |

## RealPlayer クライアント・オンライン・グラフの表示

RealPlayer クライアントのオンライン・モニタ・グラフは、[Real クライアント] グラフをグラフ・ツリーから [実行] ウィンドウの右の表示枠までドラッグすることで表示できます。

[Real クライアント] グラフで監視される測定項目を以下に示します。

| 測定値                                          | 説明                                   |
|----------------------------------------------|--------------------------------------|
| <b>Current Bandwidth (Kbits/sec)</b>         | 直前の秒におけるキロバイト数。                      |
| <b>Buffering Event Time (sec)</b>            | バッファリングに要した平均時間。                     |
| <b>Network Performance</b>                   | 現在の帯域幅とクリップの実際の帯域幅との割合。              |
| <b>Percentage of Recovered Packets</b>       | 回復されたエラー・パケットの割合。                    |
| <b>Percentage of Lost Packets</b>            | 喪失したパケットの割合。                         |
| <b>Percentage of Late Packets</b>            | 遅延したパケットの割合。                         |
| <b>Time to First Frame Appearance (sec)</b>  | 最初のフレームが現れるまでの時間（再生の開始時点から測定）。       |
| <b>Number of Buffering Events</b>            | 全バッファリング・イベントの平均数。                   |
| <b>Number of Buffering Seek Events</b>       | シーク操作に起因するバッファリング・イベントの平均数。          |
| <b>Buffering Seek Time</b>                   | シーク操作に起因するバッファリング・イベントに要した平均時間。      |
| <b>Number of Buffering Congestion Events</b> | ネットワークの輻輳に起因するバッファリング・イベントの平均数。      |
| <b>Buffering Congestion Time</b>             | ネットワークの輻輳に起因するバッファリング・イベントに要した平均時間。  |
| <b>Number of Buffering Live Pause Events</b> | ライブ放送の一時停止に起因するバッファリング・イベントの平均数。     |
| <b>Buffering Live Pause Time</b>             | ライブ放送の一時停止に起因するバッファリング・イベントに要した平均時間。 |

## Media Player クライアント・オンライン・グラフの表示

Windows Media Player クライアントのオンライン・モニタ・グラフは、[Windows Media Player] グラフをグラフ・ツリーから [実行] ウィンドウの右の表示枠までドラッグすることで表示できます。

[Media Player クライアント] で監視される測定項目を以下に示します。

| 測定値                                      | 説明                                                                                                                         |
|------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|
| <b>Average Buffering Events</b>          | この測定値には、メディアの内容が不十分だったために、Media Player クライアントが受信メディア・データのバッファリングを行う必要のあった回数。                                               |
| <b>Average Buffering Time (sec)</b>      | メディア・クリップの再生を続行するのに十分なデータ量のメディア・データを Media Player クライアントが待機した時間。                                                           |
| <b>Current bandwidth (Kbits/sec)</b>     | 受信した秒ごとのキロバイト数。                                                                                                            |
| <b>Number of Packets</b>                 | サーバから特定のメディア・クリップ用に送信されたパケット数。                                                                                             |
| <b>Stream Interruptions</b>              | メディア・クリップの再生中に、Media Player クライアントで発生した中断の回数。この測定値には、メディア・プレイヤー・クライアントが受信メディア・データをバッファリングを行う必要のあった回数と、再生中に発生したエラーの数が含まれます。 |
| <b>Stream Quality (Packet-level)</b>     | 総パケット数に対する受信パケットの割合。                                                                                                       |
| <b>Stream Quality (Sampling-level)</b>   | 遅延なく受信したストリームの割合（受信遅延なし）。                                                                                                  |
| <b>Total number of recovered packets</b> | 回復された喪失パケットの数。この値はネットワーク再生の場合にだけ適用されます。                                                                                    |
| <b>Total number of lost packets</b>      | 回復されなかった喪失パケットの数。この値はネットワーク再生の場合にだけ適用されます。                                                                                 |



# 第 27 章

---

## ERP/CRM サーバ・リソースの監視

シナリオの実行時に、ERP/CRM サーバのリソースを監視して、サーバのパフォーマンスのボトルネックを特定できます。

本章では、以下の項目について説明します。

- ▶ SAP モニタの設定
- ▶ SAP Portal モニタの設定
- ▶ Siebel Web サーバ・モニタの設定
- ▶ Siebel Server Manager モニタの設定

### ERP/CRM サーバ・リソースの監視について

ERP/CRM サーバ・リソース・モニタは、SAP R/3 システム、SAP Portal、Siebel Server Manager、および Siebel Web サーバに関するパフォーマンス情報を取得します。

このデータを取得するには、シナリオを実行する前に ERP/CRM サーバ・リソース・モニタを起動して、監視対象の統計値と測定値を指定する必要があります。

## SAP モニタの設定

SAP R/3 システム・サーバに関するデータを取得するには、シナリオを実行する前に ERP/CRM サーバ・リソース・モニタを起動して、監視対象にする統計値と測定値を指定する必要があります。

SAP モニタを使って以下を表示できます。

- ▶ 各 SAP システムに設定されているインスタンスの数。
- ▶ すべてのアプリケーション・インスタンス（ログオンしたものだけでなく）のデータ。
- ▶ 使用されているトランザクションとそれら呼び出しているユーザ。
- ▶ 個々のインスタンスで作業を行っているユーザの数。
- ▶ すべてのインスタンスの最近のパフォーマンス履歴。
- ▶ 応答時間の分布。
- ▶ 任意のアプリケーション・サーバのリソースの消費量。
- ▶ 当日または最近のアプリケーション・サーバの作業負荷。

SAP R/3 システム・サーバを監視する前に、SAP モニタの環境を設定する必要があります。

### SAP モニタ環境の設定

SAP R/3 システム・サーバを監視する前に、サーバ・モニタ環境を設定する必要があります。

**SAP モニタ環境を設定するには、次の手順で行います。**

- 1 コントローラ・マシンに SAP GUI クライアント・ファイルをインストールします。
- 2 **F6** キーを押して、st03 トランザクションにアクセスして **last minute load** の情報を照会できるか確認します。この機能が有効になっていない場合、コントローラ・マシンの SAP R/3 クライアントから、コントローラに定義されているユーザ名とパスワードを使って有効にします。

## SAP モニタの設定

SAP R/3 システム・サーバを監視するには、SAP モニタで測定するカウンタを選択する必要があります。カウンタは、[SAP の測定値を追加] ダイアログ・ボックスで選択します。

---

**注：**SAP R/3 パフォーマンス・モニタは、SAP R/3 サーバがインストールされているオペレーティング・システムやプラットフォームに関係なく、SAP サーバ 3.1 ～ 4.6 をサポートします。

---

**SAP モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [SAP] 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。
  - 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
  - 3 [SAP] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 

**注：**[マシンの追加] ダイアログ・ボックスで、次の形式でシステム番号と IP アドレスを指定することもできます。

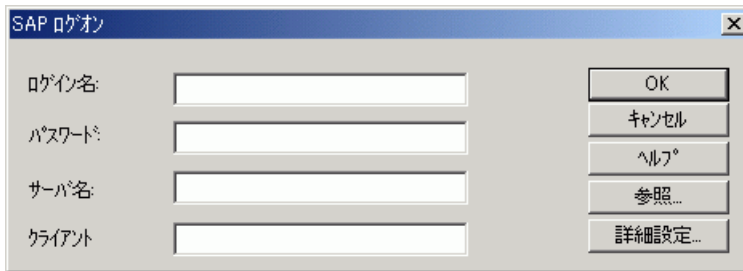
<システム番号 :IP アドレス>

次に例を示します。

199.35.106.162:00

---

- 4 [SAP] ダイアログ・ボックスの [リソース測定場所] セクションの [追加] をクリックします。[SAP ログオン] ダイアログ・ボックスが開きます。



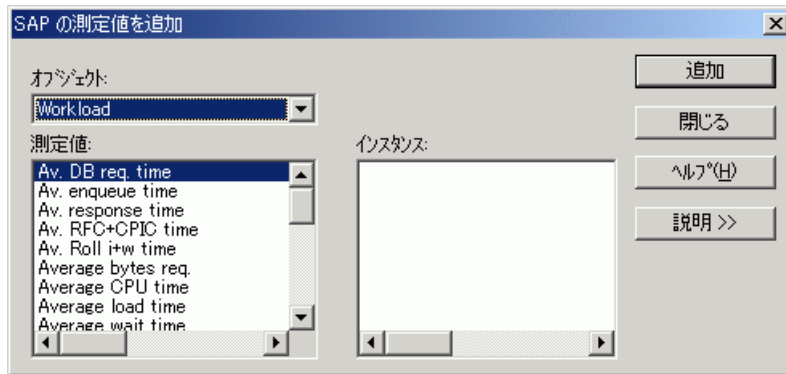
- 5 [ログイン名], [パスワード], [サーバ名], [クライアント] を入力します。

---

**注：**ルータを経由して SAP モニタに接続する場合、ルータ文字列を [サーバ名] フィールドにルータ指定文字列を入力する必要があります。ルータ指定文字列の形式は次のとおりです：< RouterString/ServerIP/S/sapdpxx >  
RouterString には /H/ < IP\_ADDRESS > /H/ < IP\_ADDRESS > /H/, ServerIP にはアプリケーション・サーバの IP アドレス、xx にはシステム番号を指定します。たとえば、ルータ指定文字列が /H/199.35.107.9/H/204.79.199.244/H/ で、アプリケーション・サーバの IP アドレスが 172.20.11.6、そして、システム番号が 00 の場合、[サーバ名] フィールドには次の文字列を入力します。  
/H/199.35.107.9/H/204.79.199.244/H/172.20.11.6/S/sapdp00

---

- 6 [OK] をクリックします。[SAP の測定値を追加] ダイアログ・ボックスが開きます。



- 7 オブジェクト、測定値、インスタンスを選択します。Ctrl キーを使って複数の測定値を選択できます。インスタンスが意味を持つのは、強調表示された測定値の複数のインスタンスが実行されている場合だけです。各測定値の説明を表示するには、[説明 >>] をクリックしてダイアログ・ボックスを拡張します。

使用可能なパフォーマンス・カウンタの一覧については、526 ページを参照してください。

- 8 [追加] をクリックして、選択した測定値をリソースのリストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。
- 9 [SAP] 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。

## SAP パフォーマンス・カウンタ

以下に、最もよく監視の対象にされるカウンタを示します。

| 測定値                          | 説明                                                                                                                                                                                                                                                  |
|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Average CPU time</b>      | ワーク・プロセスで使用される平均 CPU 時間。                                                                                                                                                                                                                            |
| <b>Average response time</b> | ダイアログがディスクパッチャー・ワーク・プロセスに要求を送信した時点から、ダイアログの処理を経て、ダイアログが完了してデータがプレゼンテーション層に渡されるまでを計測した平均応答時間。SAP GUI とディスクパッチャーの間の応答時間はこの値には含まれていません。                                                                                                                |
| <b>Average wait time</b>     | 未処理のダイアログ・ステップがディスクパッチャー・キューで空きワーク・プロセスを待機する時間。通常は、ディスクパッチャー・ワーク・プロセスはダイアログ・ステップから要求を受信した直後にダイアログ・ステップをアプリケーション・プロセスに渡します。その場合の平均待機時間は数ミリ秒です。アプリケーション・サーバまたはシステム全体に大きな負荷がかかっている場合、ディスクパッチャー・キューが長くなります。                                             |
| <b>Average load time</b>     | ABAP ソース・コードや画面情報などのオブジェクトをデータベースからロードして生成するのにかかる時間。                                                                                                                                                                                                |
| <b>Database calls</b>        | データベースに送信された解析済みの要求の数。                                                                                                                                                                                                                              |
| <b>Database requests</b>     | データベース内のデータを要求する論理 ABAP 要求数。これらの要求は R/3 データベース・インタフェースを通して渡され、個別のデータベース呼び出しに解析されます。データベース呼び出しとデータベース要求の比率は重要です。テーブル内の情報へのアクセスが SAP バッファでバッファリングされていると、データベース・サーバへのデータベース呼び出しは不要となります。したがって、呼び出しと要求の対比からテーブル・バッファ処理の総体的な効率を知ることができます。適切な比率は 1:10 です。 |
| <b>Roll ins</b>              | ロール・イン・ユーザ・コンテキスト数。                                                                                                                                                                                                                                 |
| <b>Roll outs</b>             | ロール・アウト・ユーザ・コンテキスト数。                                                                                                                                                                                                                                |
| <b>Roll in time</b>          | ロール・インの処理時間。                                                                                                                                                                                                                                        |

| 測定値                                     | 説明                                                                                                                                                        |
|-----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Roll out time</b>                    | ロール・アウトの処理時間。                                                                                                                                             |
| <b>Roll wait time</b>                   | ロール領域のキュー時間。同期 RFC が呼び出されると、ダイアログ・ステップが完了していなくても、ワーク・プロセスはロール・アウトを実行し、ロール領域で RFC の終了を待機します。ロール領域では、RFC サーバ・プログラムは送信されてくるほかの RFC も待機できます。                  |
| <b>Average time per logical DB call</b> | データベース・システムに送られたすべてのコマンドへの平均応答時間（ミリ秒）。これはデータベース・サーバの CPU、ネットワーク、バッファリングの処理能力、データベース・サーバの入力/出力処理能力などに依存します。バッファリングされたテーブルのアクセス時間は何倍も高速であるため測定項目には含まれていません。 |

## SAP Portal モニタの設定

SAP Portal モニタを使用するには、まず SiteScope マシン上で SAP Portal モニタを設定してから、SAP Portal モニタによる測定の対象にするカウンタを選択する必要があります。カウンタは、コントローラの [SAP Portal] ダイアログ・ボックスで選択します。

SAP Portal サーバを監視する前に、サーバ・モニタ環境を設定する必要があります。

**SAP Portal モニタを設定する前に、次の作業を行います。**

- 1 SiteScope がサーバにインストールされていることを確認します。SiteScope は、SAP Portal サーバの監視に使用するアプリケーションです。SiteScope はコントローラ・マシンにインストールできますが、専用のサーバにインストールすることをお勧めします。
- 2 SiteScope をインストールしたマシンで、必要な SAP Portal マシンを SiteScope が監視するように設定します。SiteScope サーバの設定の詳細については、『SiteScope ユーザーズ・ガイド』を参照してください。

---

**注：**モニタに名前を割り当てるときは、モニタ名にサーバ名を含めます。そうすることにより、モニタが属するホストについての混乱を避けられます。

---

- 3 SiteScope が監視対象のサーバから必要なデータを収集していることを確認します。[SiteScope Panel] で、SAP Portal サーバ・マシンに対してポーリングを行うモニタ・グループを選択し、そのモニタの [Status] カラムにサーバ測定値のリストが表示されていることを確認します。

**SiteScope マシンで SAP Portal モニタを設定するには、次の手順で行います。**

- 1 モニタ・アドインをインストールしてから SiteScope を再起動し、SiteScope がインストールされているマシンから次の URL を開いて、SAP Portal の SWSE ページへの接続を確認します。

http:// < SAP Portal サーバ > /sapportal

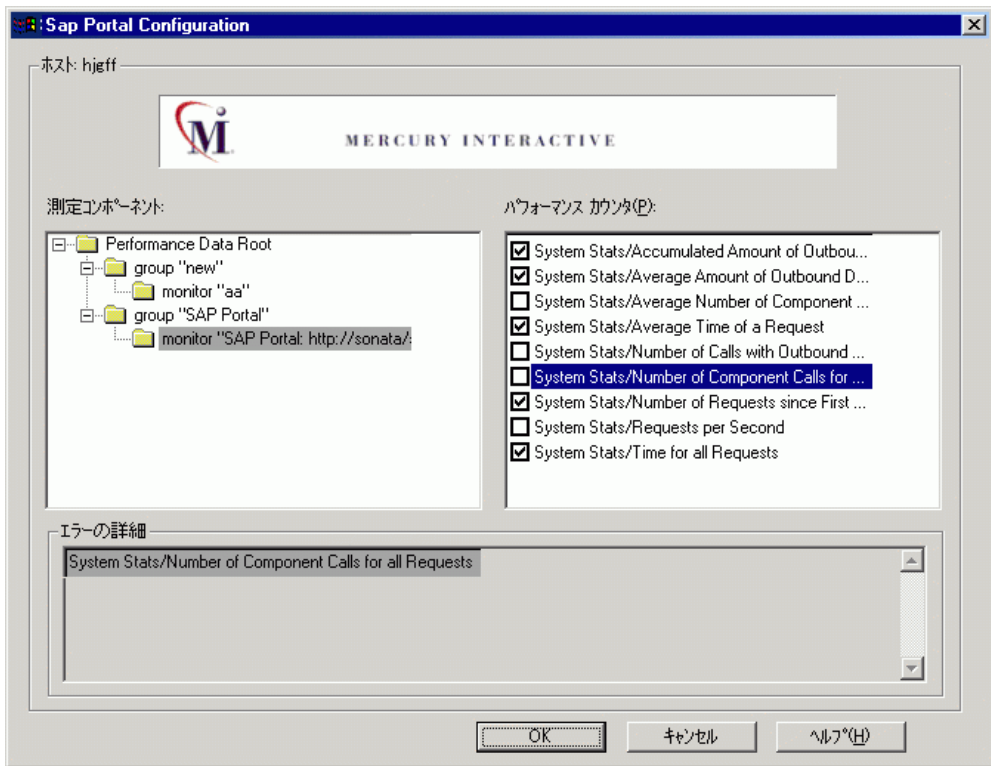


- 2 [SiteScope Add Monitors] ページを開き, [Add SAP Portal Monitor] をクリックします。
- 3 監視の対象にする SWE stats ページの URL を [server] フィールドに入力します。次に例を示します。  
 http:// < SAP Portal サーバ > /sapportal
- 4 監視の対象にするアプリケーションの名前を [Application] フィールドに入力します。
- 5 ユーザ名とパスワードを入力します (必要な場合)。
- 6 [Choose Counters] をクリックし, 監視の対象にするカウンタを選択します。
- 7 もう一度 [Choose Counters] をクリックし, [Add Monitor] をクリックします。

コントローラ・マシンで SAP Portal モニタを設定するには, 次の手順で行います。

- 1 グラフ・ツリーで [SAP Portal] 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。
- 2 対象のグラフを右クリックして, [測定値の追加] を選択するか, [モニタ] > [測定値の追加] を選択します。
- 3 [SAP Portal] ダイアログ・ボックスの [監視するサーバ マシン] セクションで [追加] をクリックして, 監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して, [OK] をクリックします。
- 4 [SAP Portal] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして, 監視対象の測定値を選択します。[SAP Portal Configuration] ダイアログ・ボックスに利用可能な測定値が表示されます。

5 [測定コンポーネント] ツリーを目的のコンポーネントまでたどっていきます。



6 [SAP Portal Configuration] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。

測定値の一覧については、531 ページ「SAP Portal パフォーマンス・カウンタ」を参照してください。

7 [SAP Portal Configuration] ダイアログ・ボックスと [SAP Portal] ダイアログ・ボックスで [OK] をクリックすると、SAP Portal モニタが起動します。

## SAP Portal パフォーマンス・カウンタ

以下の表に、標準で監視可能なカウンタの説明を示します。

| 測定値                                                          | 説明                           |
|--------------------------------------------------------------|------------------------------|
| <b>Accumulated Amount of Outbound Data (bytes)</b>           | 送信データの累積量 (バイト)。             |
| <b>Time for all Requests (ms)</b>                            | すべての要求を処理するのに要した時間の合計 (ミリ秒)。 |
| <b>Average Amount of Outbound Data per Request (bytes)</b>   | 要求ごとの送信データの平均量 (バイト)。        |
| <b>Average Number of Component Calls per Request (bytes)</b> | 要求ごとのコンポーネント呼び出しの平均数 (バイト)。  |
| <b>Average Time of a Request (ms)</b>                        | 1 つの要求を処理するのに要した時間の平均 (ミリ秒)。 |
| <b>Number of Calls with Outbound Data</b>                    | 送信データを伴う呼び出しの総数。             |
| <b>Number of Component Calls for all Requests</b>            | すべての要求のコンポーネント呼び出しの総数。       |
| <b>Number of Requests since First Request</b>                | 最初の要求からこれまでの要求の総数。           |
| <b>Requests per Second</b>                                   | 1 秒あたりに発行された要求の数。            |
| <b>Time Stamp of First Request</b>                           | 最初の要求のタイム・スタンプ。              |

## Siebel Web サーバ・モニタの設定

Siebel Web サーバ・モニタを使用するには、まず SiteScope マシン上で Siebel Web サーバ・モニタを設定してから、Siebel Web サーバ・モニタによる測定の対象にするカウンタを選択する必要があります。カウンタは、コントローラの [Siebel Web Server] ダイアログ・ボックスで選択します。

**Siebel Web サーバ・モニタを設定する前に、次の作業を行います。**

- 1 SiteScope がサーバにインストールされていることを確認します。SiteScope は、Siebel サーバの監視に使用するアプリケーションです。SiteScope はコントローラ・マシンにインストールできますが、専用のサーバにインストールすることをお勧めします。
- 2 SiteScope をインストールしたマシンで、必要な SAP Web Server マシンを SiteScope が監視するように設定します。SiteScope サーバの設定の詳細については、『SiteScope ユーザーズ・ガイド』を参照してください。

---

**注：**モニタに名前を割り当てるときは、モニタ名にサーバ名を含めます。そうすることにより、モニタが属するホストについての混乱を避けられます。

---

- 3 SiteScope が監視対象のサーバから必要なデータを収集していることを確認します。[SiteScope Panel] で、Siebel Web サーバ・マシンに対してポーリングを行うモニタ・グループを選択し、そのモニタの [Status] カラムにサーバ測定値のリストが表示されていることを確認します。

**SiteScope マシンで Siebel Web サーバ・モニタを設定するには、次の手順で行います。**

- 1 モニタ・アドインをインストールしてから SiteScope を再起動し、SiteScope がインストールされているマシンから次の URL を開いて、Siebel の SWSE ページへの接続を確認します。

`http:// < Siebel Web サーバ > /callcenter/_stats.swe`

- 2 [SiteScope Panel] で、[Create Group] をクリックしてグループの名前を入力し、[Add] をクリックします。
- 3 [Add to Group] セクションで [Monitor] をクリックし、モニタのリストから [Siebel Web Server] を選択します。

- 4 監視の対象にする SWE stats ページの URL を [server] フィールドに入力します。次に例を示します。

http:// < Siebel Web サーバ > /callcenter/\_stats.swe

- 5 監視の対象にするアプリケーションの名前を [Application] フィールドに入力します。次に例を示します。

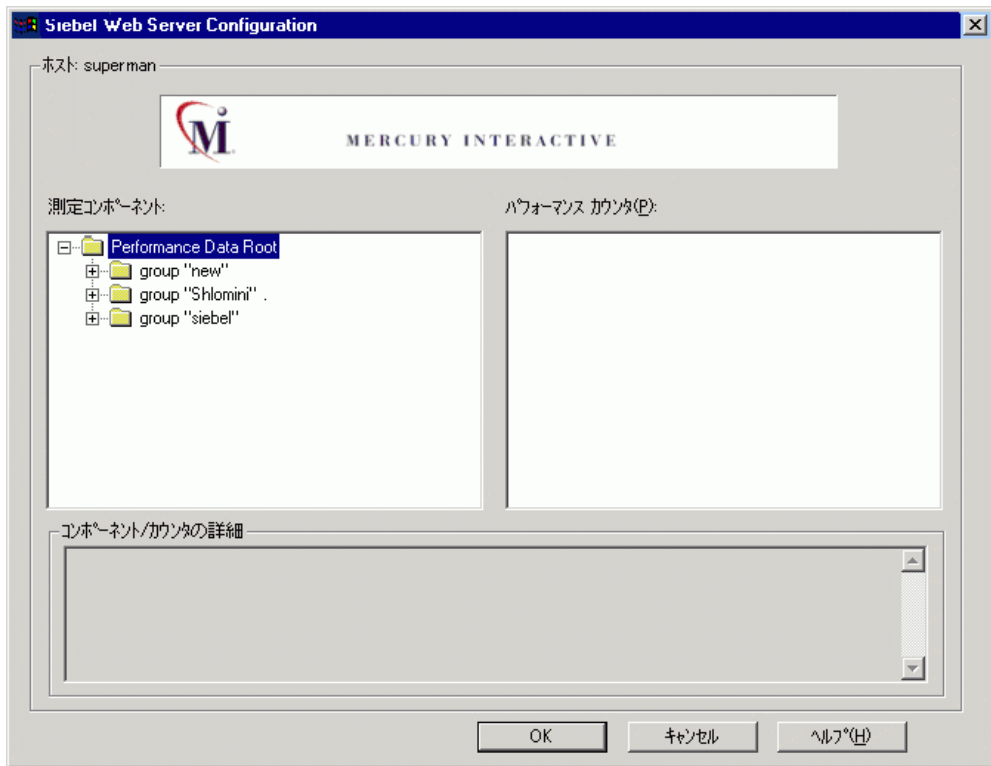
Callcenter

- 6 ユーザ名とパスワードを入力します (必要な場合)。
- 7 [Choose Counters] をクリックし、監視の対象にするカウンタを選択します。
- 8 もう一度 [Choose Counters] をクリックし、[Add Monitor] をクリックします。

コントローラ・マシンで Siebel Web サーバ・モニタを設定するには、次の手順で行います。

- 1 グラフ・ツリーで [Siebel Web Server] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Siebel Web Server] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [Siebel Web Server] ダイアログ・ボックスの [リソース測定場所] 項目で [追加] をクリックして、監視対象の測定値を選択します。[Siebel Web Server Configuration] ダイアログ・ボックスに利用可能な測定値が表示されます。

5 [測定コンポーネント] ツリーを目的のコンポーネントまでたどっていきます。



6 [Siebel Web Server Configuration] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。

使用可能なパフォーマンス・カウンタの一覧については、535 ページを参照してください。

7 [Siebel Web Server Configuration] ダイアログ・ボックスと [Siebel Web Server] ダイアログ・ボックスで [OK] をクリックすると、Siebel Web サーバ・モニタが起動します。

**Siebel Web サーバ・パフォーマンス・カウンタ**

以下の表に、標準で監視可能なカウンタの説明を示します。

| 測定値                                               | 説明                       |
|---------------------------------------------------|--------------------------|
| <b>Anonymous sessions requested from the pool</b> | プールから要求された匿名セッションの数。     |
| <b>Open Session Time</b>                          | ユーザがシステムにログオンするのに要した時間。  |
| <b>Anon Session Removed</b>                       | プールから削除された匿名セッションの数。     |
| <b>Anon Session Available</b>                     | プール内にある使用可能な匿名セッションの数。   |
| <b>Anonymous sessions returns to the pool</b>     | プールに返された匿名セッションの数。       |
| <b>Response Time</b>                              | ユーザ要求への応答にかかった時間。        |
| <b>Close Session Time</b>                         | ユーザがシステムからログオフするのに要した時間。 |
| <b>Request Time</b>                               | ユーザ要求の処理に要した時間。          |

## Siebel Server Manager モニタの設定

Siebel Server Manager のパフォーマンスを監視するには、まず SiteScope マシンに Siebel Server Manager クライアントをインストールする必要があります。その後、Siebel Server Manager モニタで測定するカウンタを選択します。カウンタは、コントローラの [Siebel Server Manager] ダイアログ・ボックスで選択します。

**Siebel Server Manager モニタを設定する前に、次の作業を行います。**

- 1 SiteScope がサーバにインストールされていることを確認します。SiteScope は、Siebel Server Manager サーバの監視に使用されるアプリケーションです。SiteScope はコントローラ・マシンにインストールできますが、専用のサーバにインストールすることをお勧めします。
- 2 SiteScope をインストールしたマシンで、必要な Siebel Server Manager マシンを SiteScope が監視するように設定します。SiteScope サーバの設定の詳細については、『SiteScope ユーザーズ・ガイド』を参照してください。

---

**注：**モニタに名前を割り当てるときは、モニタ名にサーバ名を含めます。そうすることにより、モニタが属するホストについての混乱を避けられます。

---

- 3 SiteScope が監視対象のサーバから必要なデータを収集していることを確認します。[SiteScope Panel] で、Siebel Server Manager マシンに対してポーリングを行うモニタ・グループを選択し、そのモニタの [Status] カラムにサーバ測定値のリストが表示されていることを確認します。

**SiteScope マシンで Siebel Server Manager クライアントを設定するには、次の手順で行います。**

- 1 モニタ・アドインをインストールしてから SiteScope を再起動し、SiteScope がインストールされているマシンから次の URL を開いて、Siebel の SWSE ページへの接続を確認します。

`http:// < Siebel サーバ > /callcenter/_stats.swe`

- 2 [SiteScope Panel] で、[Create Group] をクリックしてグループの名前を入力し、[Add] をクリックします。
- 3 [Add to Group] セクションで [Monitor] をクリックし、モニタのリストから [Siebel Server Manager] を選択します。

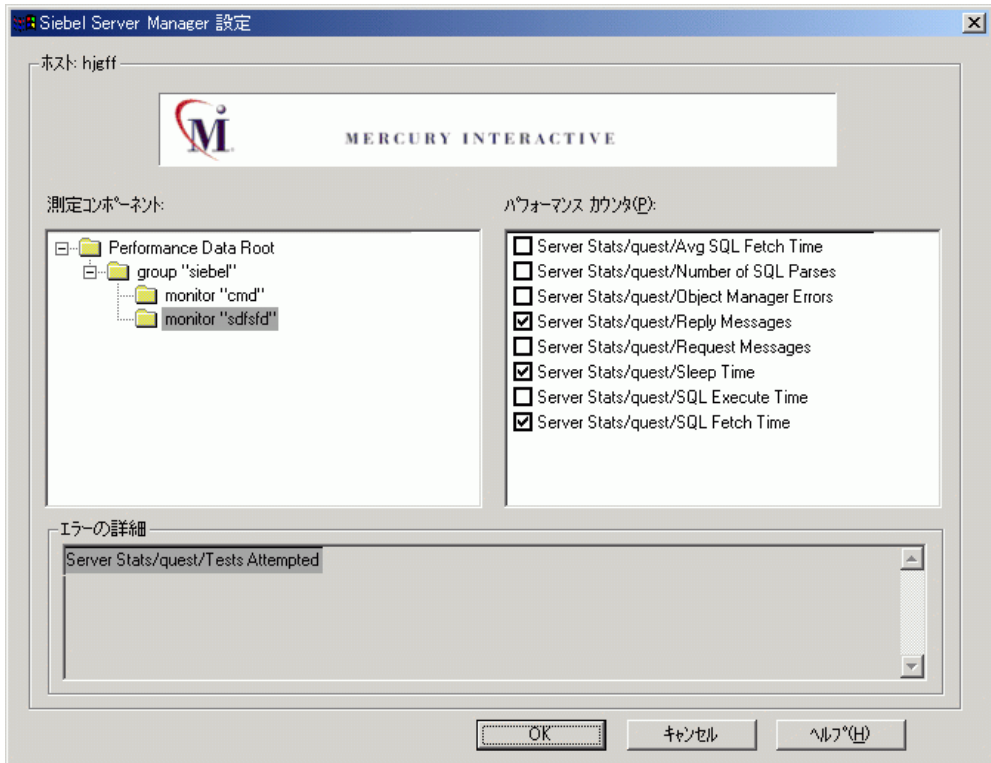


- 4 Siebel サーバの名前を [Application server] フィールドに, Enterprise Server の名前を [Enterprise server] フィールドに, Gateway Server の名前を [Gateway server] フィールドに, それぞれ入力します。
- 5 Siebel Server Manager へのパスを [Path to Script] フィールドに入力します。
- 6 ユーザ名とパスワードを入力します (必要な場合)。
- 7 [Choose Counters] をクリックし, 監視の対象にするカウンタを選択します。
- 8 もう一度 [Choose Counters] をクリックし, [Add Monitor] をクリックします。

**コントローラ・マシンで Siebel Server Manager モニタを設定するには, 次の手順で行います。**

- 1 グラフ・ツリーで [Siebel Server Manager] 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。
- 2 対象のグラフを右クリックして, [測定値の追加] を選択するか, [モニタ] > [測定値の追加] を選択します。
- 3 [Siebel Server Manager] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして, 監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して, [OK] をクリックします。
- 4 [Siebel Server Manager] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして, 監視対象の測定値を選択します。[Siebel Server Manager 設定] ダイアログ・ボックスに利用可能な測定値が表示されます。

5 [測定コンポーネント] ツリーを目的のコンポーネントまでたどっていきます。



6 [Siebel Server Manager 設定] ウィンドウの右の表示枠で、必要なパフォーマンス・カウンタを選択します。

使用可能なパフォーマンス・カウンタの一覧については、539 ページを参照してください。

7 [Siebel Server Manager 設定] ダイアログ・ボックスと [Siebel Server Manager] ダイアログ・ボックスで [OK] をクリックすると、Siebel Server Manager モニタが起動します。

## Siebel Server Manager パフォーマンス・カウンタ

以下の表に、標準で監視可能なカウンタの説明を示します。

| 測定値                                 | 説明                     |
|-------------------------------------|------------------------|
| <b>Average Connect Time</b>         | 平均接続時間。                |
| <b>Average Reply Size</b>           | ユーザ応答の平均サイズ。           |
| <b>Average Request Size</b>         | ユーザ要求の平均サイズ。           |
| <b>Average Requests Per Session</b> | セッションあたりのユーザ要求の平均数。    |
| <b>Average Response Time</b>        | サーバが要求に応答するのに要した時間の平均。 |
| <b>Average Think Time</b>           | 要求に応答するまでの思考遅延時間の平均。   |
| <b>Avg SQL Execute Time</b>         | 平均 SQL 実行時間。           |
| <b>Avg SQL Fetch Time</b>           | 平均 SQL フェッチ時間。         |
| <b>Avg SQL Parse Time</b>           | 平均 SQL 解析時間。           |
| <b>CPU Time</b>                     | ワーク・プロセスで使用された CPU 時間。 |
| <b>Elapsed Time</b>                 | 経過時間の合計。               |
| <b>Num of DBConn Retries</b>        | データベース接続の再試行回数。        |
| <b>Num of DLRbk Retries</b>         | DLRbk 再試行の回数。          |
| <b>Num of Exhausted Retries</b>     | 時間切れになった再試行の総数。        |
| <b>Number of SQL Executes</b>       | SQL 実行の総数。             |
| <b>Number of SQL Fetches</b>        | SQL フェッチの総数。           |
| <b>Number of SQL Parses</b>         | SQL 解析の総数。             |
| <b>Number of Sleeps</b>             | スリープの数。                |
| <b>Object Manager Errors</b>        | オブジェクト・マネージャ・エラーの総数。   |
| <b>Reply Messages</b>               | 応答メッセージの総数。            |
| <b>Request Messages</b>             | 要求メッセージの総数。            |

| 測定値                        | 説明                |
|----------------------------|-------------------|
| <b>SQL Execute Time</b>    | SQL 実行時間の合計。      |
| <b>SQL Fetch Time</b>      | SQL フェッチ時間の合計。    |
| <b>SQL Parse Time</b>      | SQL 解析時間の合計。      |
| <b>Sleep Time</b>          | スリープ時間の合計。        |
| <b>Tests Attempted</b>     | テストを試みた回数。        |
| <b>Tests Failed</b>        | 失敗したテストの数。        |
| <b>Tests Successful</b>    | 成功したテストの数。        |
| <b>Total Reply Size</b>    | 応答の合計サイズ (バイト単位)。 |
| <b>Total Request Size</b>  | 要求の合計サイズ (バイト単位)。 |
| <b>Total Response Time</b> | 応答時間の合計。          |
| <b>Total Tasks</b>         | タスクの総数。           |
| <b>Total Think Time</b>    | 思考遅延時間の合計。        |

# 第 28 章

---

## Java パフォーマンスの監視

Java パフォーマンス・モニタを使って、シナリオの実行時に、Java 2 Platform, Enterprise Edition (J2EE) オブジェクト、Enterprise Java Bean (EJB) オブジェクト、および Java ベースのアプリケーションのリソースの使用状況を監視できます。

本章では、以下の項目について説明します。

- ▶ EJB パフォーマンスの監視

---

**注：**J2EE パフォーマンス・モニタについては第 29 章「J2EE パフォーマンスの監視」で別に説明します。

---

## Java パフォーマンスの監視について

Java パフォーマンス・モニタは、シナリオ実行時に、Java 2 Platform, Enterprise Edition (J2EE) オブジェクト、Enterprise Java Bean (EJB) オブジェクト、および Java ベースのアプリケーションに関するパフォーマンス情報を表示します。パフォーマンス・データを取得するには、シナリオを実行する前に Java パフォーマンス・モニタを起動して、監視対象の統計値と測定値を指定する必要があります。

## EJB パフォーマンスの監視

EJB パフォーマンス・モニタを使って、シナリオ実行時に WebLogic, WebSphere または Oracle 9iAS アプリケーション・サーバ上の Enterprise Java Bean (EJB) オブジェクトを監視できます。

### サポート対応表

| アプリケーション・サーバ        | バージョン              | プラットフォーム              |
|---------------------|--------------------|-----------------------|
| WebLogic WebLogic   | 4.x, 5.x, 6.x, 7.x | Windows, Solaris, AIX |
| WebSphere WebSphere | 3.x, 4.x           | Windows, Solaris, AIX |
| Oracle 9i           | 1.0.2.2            | Windows, Solaris, AIX |

EJB オブジェクトを監視するには、まずアプリケーション・サーバ・マシンに EJB モニタをインストールして、モニタ・ディテクタを実行し、EJB モニタをアクティブにしておく必要があります。次に、クライアント・マシンの EJB モニタを設定するために、モニタで測定するカウンタを選択します。

---

**注：**サーバ側のインストール先には、EJB 仮想ユーザ・スクリプトを作成するための新しい EJBDetector サポート・ファイルが含まれています。EJBDetector の詳細については、『LoadRunner 仮想ユーザ・スクリプトの作成』を参照してください。

---

### EJB モニタのインストールとモニタ・ディテクタの実行

EJB オブジェクトを監視するには、あらかじめアプリケーション・サーバ・マシンに EJB モニタのサポート・ファイルをインストールしておき、有効な JDK 環境が用意されていることを確認する必要があります。次に、バッチ・ファイルまたはコマンド・ラインからモニタ・ディテクタを実行して、EJB モニタで監視を行うための準備をします。

**EJB モニタのサポート・ファイルをインストールするには、次の手順で行います。**

マーキュリー・インタラクティブの EJB サポート・ファイル用のホーム・ディレクトリ（たとえば MERC\_MONITOR\_HOME）を作成し、そこに **LoadRunner CD > add-ins\Monitors\J2EE\Windows\jmonitor\_ <プラットフォーム> .jar** ファイルを展開します。

UNIX プラットフォームの場合には、jar ユーティリティを使用してインストール用の Jar ファイルを抽出します。

MERC\_MONITOR\_HOME ディレクトリに移動して、次のコマンドを入力します。

```
jar -xvf <jmonitor_ <プラットフォーム> .jar へのパス>
```

**バッチ・ファイルからモニタ・ディテクタを実行するには、次の手順で行います。**

1 **env.cmd** (NT) または **env.sh** (UNIX) ファイルを開いて、次の変数を設定します。

**JAVA\_HOME** JDK のルート・ディレクトリを指定します。

**APP\_SERVER\_DRIVE** アプリケーション・サーバがインストールされているドライブを指定します (NT のみ)。

**DETECTOR\_INS\_DIR** ディテクタのルート・ディレクトリを指定します。

**APP\_SERVER\_ROOT** 次のガイドラインに従ってください。

**BEA WebLogic サーバ 4.x および 5.x** : アプリケーション・サーバのルート・ディレクトリを指定します。

**BEA WebLogic サーバ 6.x および 7.x** : ドメイン・フォルダの完全パスを指定します。

**WebSphere サーバ 3.x および 4.0.x** : アプリケーション・サーバのルート・ディレクトリを指定します。

**Oracle OC4J** : アプリケーション・サーバのルート・ディレクトリを指定します。

**Sun J2EE サーバ** : 配備可能な **.ear** ファイルまたは複数の **.ear** ファイルが格納されているディレクトリへの完全パスを指定します。

**EJB\_DIR\_LIST**  
(オプション)

配備可能な **.ear** または **.jar** ファイルのほか、テスト対象の EJB が使用するその他のクラス・ディレクトリまたは **.jar** ファイルを、セミコロン (;) で区切ったディレクトリとファイルのリストとして指定します。

- 2 **Mon\_Detector.cmd** (NT) または **Mon\_Detector.sh** (UNIX) バッチ・ファイルを実行して、配備されている EJB に関する情報を収集します。モニタ・ディテクタを実行すると、`< MERC_MONITOR_HOME > %dat` ディレクトリに次の3つのファイルが作成されます。**ejb\_monitor.hooks**, **cjhook.ini** および **regmon.properties**。これらのファイルには、アプリケーション・サーバで検出された EJB に関する情報が記録されています。

---

注：アプリケーション・サーバで EJB を追加、変更、削除するたびに、モニタ・ディテクタを実行する必要があります。

---

コマンド・ラインからモニタ・ディテクタを実行するには、次の手順で行います。

- 1 `< MERC_MONITOR_HOME > %classes, < MERC_MONITOR_HOME > %dat, < MERC_MONITOR_HOME > %classes%xcerces.jar` ファイルを CLASSPATH 環境変数に追加します。
- 2 コマンド・ラインで `java MonDetect <検索ルート・ディレクトリ>` を実行して、配備されている EJB に関する情報を収集します。

**<検索ルート・ディレクトリ>** EJB を検索する場所として、1つ以上のディレクトリまたはファイルをセミコロンで区切って指定します。次のガイドラインに従ってください。

**BEA WebLogic サーバ 4.x および 5.x**：アプリケーション・サーバのルート・ディレクトリを指定します。

**BEA WebLogic サーバ 6.x および 7.x**：ドメイン・フォルダの完全パス、続いてルート・ディレクトリを指定します。

**WebSphere サーバ 3.x および 4.0.x**：アプリケーション・サーバのルート・ディレクトリを指定します。

**Oracle OC4J**：アプリケーション・サーバのルート・ディレクトリを指定します。

**Sun J2EE サーバ**：配備可能な **.ear** ファイルまたは複数の **.ear** ファイルが格納されているディレクトリへの完全パスを指定します。



検索するディレクトリやファイルのリストを指定することもできます。指定しない場合、CLASSPATH が検索されます。

モニタ・ディテクタを実行すると、< MERC\_MONITOR\_HOME > %dat ディレクトリに次の 3 つのファイルが作成されます：**ejb\_monitor.hooks**、**cjhook.ini**、および **regmon.properties**。これらのファイルには、アプリケーション・サーバで検出された EJB に関する情報が記録されています。

---

**注：**アプリケーション・サーバで EJB を追加、変更、削除するたびに、モニタ・ディテクタを実行する必要があります。

---

## アプリケーション・サーバでの EJB モニタの設定

WebLogic, WebSphere, または Oracle 9iAS マシンにマーキュリー・インタラクティブの EJB モニタ・サポート・ファイルをインストールしたら、EJB モニタのサポート機能を有効にした状態でアプリケーション・サーバが実行されるように設定する必要があります。

---

**注：**環境変数は、以降に示してある順序で設定することが重要です。

---

## WebLogic Server

WebLogic 4.x ~ 5.x サーバ, WebLogic 6.x サーバおよび WebLogic 7.x サーバの設定方法は異なります。

**WebLogic 4.x ~ 5.x サーバを設定するには、次の手順で行います。**

- 1 < WebLogic ホーム > **startWeblogic.cmd** ファイルをバックアップするため、このファイルを < WebLogic ホーム > %startWeblogicMercury.cmd にコピーします。
- 2 < WebLogic ホーム > %startWeblogicMercury.cmd ファイルを開きます。
- 3 このファイルの runWebLogicJava セクションで、WEBLOGIC\_CLASSPATH 環境設定の後に次の環境変数を設定します。

Windows プラットフォームの場合：

```
set MERC_MONITOR_HOME= < MERC モニタ・ホーム・ディレクトリ>
set CLASSPATH=%MERC_MONITOR_HOME%\dat
set JAVA_CLASSPATH=
%MERC_MONITOR_HOME%\dat;%MERC_MONITOR_HOME%\classes;%
MERC_MONITOR_HOME%\classes\xerces.jar;%JAVA_CLASSPATH%
set PATH=%PATH%;%MERC_MONITOR_HOME%\bin
```

UNIX プラットフォームの場合：

```
MERC_MONITOR_HOME < EJB モニタ・ホーム・ディレクトリ>
CLASSPATH ${MERC_MONITOR_HOME}\dat
JAVA_CLASSPATH ${MERC_MONITOR_HOME}\dat:${MERC_MONITOR_H
OME}\classes:${MERC_MONITOR_HOME}\classes\xerces.jar:${JAVA_CLA
SSPATH}
LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:${MERC_MONITOR_HOME}\bi
n
export CLASSPATH
export LD_LIBRARY_PATH
export JAVA_CLASSPATH
```

---

注：IBM AIX プラットフォームの場合は、**LD\_LIBRARY\_PATH** の代わりに **LIBPATH** を指定します。< **EJB モニタのホーム・ディレクトリ**>を EJB モニタのインストール先ルート・ディレクトリで置き換えます。UNIX プラットフォームの場合は、ライブラリ・パス変数をエクスポートする必要がある場合もあります。

---

- 4 ファイルの同じセクションで、コマンド・ラインに次のパラメータを追加します。

`-Xrunjdkhook`

たとえば、Windows プラットフォームの場合は次のようになります。

```
%JAVA_HOME%\bin%java -ms64m -mx64m -Xrunjdkhook -
classpath%JAVA_CLASSPATH%
-Dweblogic.class.path=%WEBLOGIC_CLASSPATH% -Dweblogic.home=
-Djava.security.manager -Djava.security.policy==.%weblogic.policy
weblogic.Server
```

---

**注：** Solaris にインストールされている場合のみ JDK 1.2.x を使用している場合は、コマンド・ラインに次のパラメータを追加します。

`-Dweblogic.classloader.preprocessor=com.mercuryinteractive.aim.MercuryWL5Preprocessor`

たとえば、Windows プラットフォームの場合は次のようになります。

```
%JAVA_HOME%\bin%java -ms64m -mx64m -classpath %JAVA_CLASSPATH%
-Dweblogic.classloader.preprocessor=com.mercuryinteractive.aim.MercuryWL5Preprocessor
-Dweblogic.class.path=%WEBLOGIC_CLASSPATH% -Dweblogic.home=
-Djava.security.manager -Djava.security.policy==.%weblogic.policy weblogic.Server
```

---

- 5 < WebLogic ホーム > `%startWeblogicMercury.cmd` ファイルを実行します。

**WebLogic 6.x サーバを設定するには、次の手順で行います。**

- 1 < WebLogic ホーム > `%config%` <ドメイン名> `%startWeblogic.cmd` ファイルをバックアップするため、このファイルを < WebLogic ホーム > `%config%` <ドメイン名> `%startWeblogicMercury.cmd` にコピーします。
- 2 < WebLogic ホーム > `%config%` <ドメイン名> `%startWeblogicMercury.cmd` ファイルを開きます。

- 3 このファイルの runWebLogic セクションで、次の環境変数を設定します。

Windows プラットフォームの場合：

```
set MERC_MONITOR_HOME= < MERC_MONITOR_HOME ディレクトリ>
set CLASSPATH=
%CLASSPATH%;%MERC_MONITOR_HOME%\dat;%MERC_MONITOR_
HOME%\classes;%MERC_MONITOR_HOME%\classes\xerces.jar
set PATH=%PATH%;%MERC_MONITOR_HOME%\bin
```

UNIX プラットフォームの場合：

```
MERC_MONITOR_HOME < EJB モニタ・ホーム・ディレクトリ>
CLASSPATH ${JAVA_CLASSPATH};${MERC_MONITOR_HOME}/dat:$
{MERC_MONITOR_HOME}/classes:${MERC_MONITOR_HOME}/
classes/xerces.jar
LD_LIBRARY_PATH
${LD_LIBRARY_PATH};${MERC_MONITOR_HOME}/bin
export CLASSPATH
export LD_LIBRARY_PATH
```

---

注：IBM AIX プラットフォームの場合は、**LD\_LIBRARY\_PATH** の代わりに **LIBPATH** を指定します。< **EJB モニタのホーム・ディレクトリ**>を EJB モニタのインストール先ルート・ディレクトリで置き換えます。UNIX プラットフォームの場合は、ライブラリ・パス変数をエクスポートする必要がある場合もあります。

---

- 4 ファイルの同じセクションで、コマンド・ラインに次のパラメータを追加します。

-Xrunjdkhook

たとえば、Windows プラットフォームの場合は次のようになります。

```
"%JAVA_HOME%\bin\java" -hotspot -ms64m -mx64m -Xrunjdkhook
-classpath %CLASSPATH% -Dweblogic.Domain=mydomain
-Dweblogic.Name=myserver "-Dbea.home=f:%bea" &dlq;
-Djava.security.policy=f:%bea%\wserver6.0\lib\weblogic.policy"
-Dweblogic.management.password=%WLS_PW% weblogic.Server
```

- 5 < WebLogic ホーム > %config% < ドメイン名 > %startWeblogicMercury.cmd  
ファイルを実行します。

**WebLogic 7.x サーバを設定するには、次の手順で行います。**

- 1 < WebLogic ホーム > %server%bin%startwls.cmd ファイルをバックアップするため、このファイルを < WebLogic ホーム > %server%bin%startwlsMercury.cmd にコピーします。
- 2 < WebLogic ホーム > %server%bin%startwlsMercury.cmd ファイルを開きます。
- 3 このファイルの runWebLogic セクションで、次の環境変数を設定します。

Windows プラットフォームの場合：

```
set MERC_MONITOR_HOME= < MERC_MONITOR_HOME ディレクトリ >
set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes;%MERC_MONITOR_HOME%\classes
%xerces.jar
set PATH=%PATH%;%MERC_MONITOR_HOME%\bin
```

UNIX プラットフォームの場合：

```
MERC_MONITOR_HOME < EJB モニタ・ホーム・ディレクトリ > >
CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:$MERC_
MONITOR_HOME/classes:$MERC_MONITOR_HOME/classes/xerces.jar
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$MERC_MONITOR_HOME/
bin
export CLASSPATH
export LD_LIBRARY_PATH
```

---

**注：** IBM AIX プラットフォームの場合は、**LD\_LIBRARY\_PATH** の代わりに **LIBPATH** を指定します。 < EJB モニタのホーム・ディレクトリ > を EJB モニタのインストール先ルート・ディレクトリで置き換えます。UNIX プラットフォームの場合は、ライブラリ・パス変数をエクスポートする必要がある場合もあります。

---

- 4 ファイルの同じセクションで、コマンド・ラインに次のパラメータを追加します。  
-Xrunjdkhook

たとえば、Windows プラットフォームの場合は次のようになります。

```
"%JAVA_HOME%\bin\java" -hotspot -ms64m -mx64m -Xrunjdkhook
-classpath %CLASSPATH% -Dweblogic.Domain=mydomain
-Dweblogic.Name=myserver "-Dbea.home=f:\bea"
"-Djava.security.policy==f:\bea\wlserver6.0\lib\weblogic.policy"
-Dweblogic.management.password=%WLS_PW% weblogic.Server
```

- 5 <ドメイン名> ¥startWeblogic.cmd ファイルをバックアップするため、このファイルを <ドメイン名> ¥startWeblogicMercury.cmd にコピーします。
- 6 <ドメイン名> ¥startWeblogicMercury.cmd ファイルを開きます。
- 7 weblogic サーバ呼び出しを探します。たとえば、  
  
D: ¥bea¥weblogic700¥server¥bin¥startWLS.cmd
- 8 この呼び出しの **startWLS.cmd** を **startWLSMercury.cmd** に変更して、ファイルを保存します。
- 9 <ドメイン名> ¥startWeblogicMercury.cmd ファイルを実行します。

### WebSphere サーバ 3.0 および 3.5

標準設定では、WebSphere 3.x アプリケーション・サーバは、マシンの起動時に自動サービスとして実行されます。マーキュリー・インタラクティブでは、現時点では自動サービスとして実行される WebSphere サーバで LoadRunner による EJB の監視をサポートしていないため、標準設定の WebSphere サーバの起動方法を **手動** に変更する必要があります。

**標準設定の WebSphere 3.x サーバの起動方法を変更するには、次の手順で行います。**

- 1 [スタート] > [設定] > [コントロールパネル] > [管理ツール] > [サービス] を選択します。
- 2 [IBM WS AdminServer] を選択して、[停止] ボタンをクリックします。
- 3 [IBM WS AdminServer] をダブルクリックして、[手動] 起動タイプを選択します。

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

これで、自動サービスを使わずに、< WebSphere ホーム>  
¥AppServer¥bin¥debug¥adminserver.bat からサーバを起動できます。

**LoadRunner EJB モニタのサポート機能を WebSphere 3.x サーバに追加するには、次の手順で行います。**

- 1 < WebSphere ホーム> ¥AppServer¥bin¥debug¥adminserver.bat ファイルのバックアップ・コピーを作成します。
- 2 < WebSphere ホーム> ¥AppServer¥bin¥debug¥adminserver.bat ファイルを開きます。
- 3 次の環境変数を SET\_CP セクションの末尾に追加します。

Windows プラットフォームの場合：

```
set CLASSPATH= < MERC_MONITOR_HOME > ¥dat;
< MERC_MONITOR_HOME > ¥classes; < MERC_MONITOR_HOME >
¥classes¥xerces.jar; %CLASSPATH%
set PATH=%PATH%; < MERC_MONITOR_HOME > ¥bin
```

UNIX プラットフォームの場合：

```
CLASSPATH${MERC_MONITOR_HOME}/dat:${MERC_MONITOR_HOME}/
classes:${MERC_MONITOR_HOME}/classes/xerces.jar:${CLASSPATH}
LD_LIBRARY_PATH
${LD_LIBRARY_PATH}:${MERC_MONITOR_HOME}/bin
export CLASSPATH
export LD_LIBRARY_PATH
```

---

**注：** IBM AIX プラットフォームの場合は、**LD\_LIBRARY\_PATH** の代わりに **LIBPATH** を指定します。< EJB モニタのホーム・ディレクトリ>を EJB モニタのインストール先ルート・ディレクトリで置き換えます。UNIX プラットフォームの場合は、ライブラリ・パス変数をエクスポートする必要がある場合もあります。

---

---

**注** : Solaris にインストールされている場合のみ JRE1.2.x で作業している場合、次に示す IBM の Web サイトまたは FTP サイトからパッチ・ファイル PQ46831.jar をダウンロードする必要があります。

<http://www-3.ibm.com/software/webservers/appserv/efix-archive.html>  
<ftp://ftp.software.ibm.com/software/websphere/appserv/support/fixes/pq46831/>

サーバのバージョンに合うバージョンをダウンロードしてください。パッチ・ファイルをクラスパスに追加します。

```
setenv CLASSPATH PQ46831.jar:${CLASSPATH}
```

---

- 4 **adminserver.bat** ファイルを実行します。
- 5 WebSphere 管理コンソール・アドバンスド版を開いて、[ビュー] > [トポロジ] を選択します。
- 6 [<サーバ・マシン名>] > [Default Server] を選択して、[WebSphere 管理ドメイン] ツリーを展開します。
- 7 [アプリケーション・サーバ:Default Server] ウィンドウで [一般] タブを選択します。
- 8 コマンド・ライン引数ボックスに -Xrunjdkhook と入力して、[適用] をクリックします。

JDK1.1.7 IBM が含まれる WebSphere 3.0 サーバで作業している場合、[環境] をダブルクリックします。[変数名] ボックスに `_CLASSLOAD_HOOK`、[値] ボックスに `jdkhook` と入力します。[追加] ボタン、[OK] ボタン、[適用] ボタンをクリックします。



---

**注：** Solaris にインストールされている場合のみ J2RE1.2.x が含まれる WebSphere 3.5 サーバで作業している場合、[コマンドラインの引数] ボックスに下記を入力して [適用] をクリックします。

```
-Dcom.ibm.ejs.sm.server.ServiceInitializer=com.ibm.ejs.sm.server.WilyInitializer
-Dcom.ibm.websphere.introscope.implClass=com.mercuryinteractive.aim.
MercuryWASPreprocessor
```

---

- 9 [WebSphere 管理コンソールアドバンスド版] を閉じます。
- 10 **adminserver.bat** ファイルを閉じて再起動します。

## WebSphere サーバ 4.0

WebSphere 4.0 サーバは、startServerBasic.bat ファイルまたは startServer.bat ファイルを使って起動できます。

**WebSphere 4.0 サーバを設定するには、次の手順で行います。**

- 1 WebSphere 管理サーバが実行されていることを確認し、管理コンソールを起動します。
- 2 [WebSphere 管理ドメイン] ツリーで、ノード、ホスト名、アプリケーション・サーバのサブツリーを展開し、Default Server（または JMonitor の対象にするアプリケーション・サーバ）を選択します。
- 3 Windows 2000/NT、Solaris の場合は、[一般] タブをクリックして、次の変数を [環境] ボックスに追加します。

---

**注：** < EJB モニタのホーム・ディレクトリ > を EJB モニタのインストール先ルート・ディレクトリで置き換えます。

---

Windows 2000/NT の場合：

```
name=PATH
value= < EJB モニタのホーム・ディレクトリ > %bin
```

Solaris の場合 :

```
name=LD_LIBRARY_PATH
value= < EJB モニタのホーム・ディレクトリ> /bin
```

[OK] をクリックして, [Environment Editor] ダイアログ・ボックスを閉じます。

AIX の場合 :

環境変数 LIBPATH を変更した場合, EJB モニタのライブラリを /usr/lib ディレクトリにリンクする必要があります。

次のコマンドを追加します。

```
#ln -s < EJB モニタのホーム・ディレクトリ> /bin/libcjhooke_mon.so
/usr/lib/libcjhooke_mon.so
#ln -s < EJB モニタのホーム・ディレクトリ> /bin/libconfig.so
/usr/lib/libconfig.so
#ln -s < EJB モニタのホーム・ディレクトリ> /bin/libjdkhook.so
/usr/lib/libjdkhook.so
#ln -s < EJB モニタのホーム・ディレクトリ> /bin/libmllib_ds.so
/usr/lib/libcjhooke_mon.so
#ln -s < EJB モニタのホーム・ディレクトリ> /bin/libmosifs.so
/usr/lib/libmosifs.so
#ln -s < EJB モニタのホーム・ディレクトリ> /bin/libthrutil.so
/usr/lib/libthrutil.so
```

---

**注 :** リンクを作成するには, 通常, root 権限が必要になります。または, リンクを WebSphere の /bin ディレクトリ (通常は /usr/WebSphere/AppServer/bin) に入れることもできます。

---

- 4 WebSphere 管理コンソールの [JVM の設定] タブをクリックして、次の値を classpath に追加します。

---

**注：** < EJB モニタのホーム・ディレクトリ > を EJB モニタのインストール先ルート・ディレクトリで置き換えます。

---

Windows 2000/NT の場合：

```
< EJB モニタのホーム・ディレクトリ > %dat
< EJB モニタのホーム・ディレクトリ > %classes
< EJB モニタのホーム・ディレクトリ > %classes%xerces.jar
```

Solaris または AIX の場合：

```
< EJB モニタのホーム・ディレクトリ > /dat
< EJB モニタのホーム・ディレクトリ > /classes
< EJB Monitor のホーム・ディレクトリ > /classes/xerces.jar
```

---

**注：** Solaris にインストールされている場合のみ JRE1.2.x で作業している場合、次に示す IBM の Web サイトまたは FTP サイトからパッチ・ファイル PQ46831.jar をダウンロードする必要があります。

<http://www-3.ibm.com/software/webservers/appserv/efix-archive.html>  
<ftp://ftp.software.ibm.com/software/websphere/appserv/support/fixes/pq46831/>

サーバのバージョンに合うバージョンをダウンロードしてください。次の値を classpath に追加します。

```
< EJB モニタのホーム・ディレクトリ > /classes/PQ46831.jar
```

---

- 5 [JVM の詳細設定] ボタンをクリックします。コマンド・ラインの引数フィールドに、Windows 2000/NT, Solaris, AIX に対して次の値を追加します。

-Xrunjdkhook

---

注：Solaris にインストールされている場合のみ JRE1.2.x で作業をしている場合には、-Xrunjdkhook の代わりに、次の値を追加します。

-Dcom.ibm.ejs.sm.server.ServiceInitializer=com.ibm.ejs.sm.server.WilyInitializer  
-Dcom.ibm.websphere.introscope.implClass=com.mercuryinteractive.aim.  
MercuryWASPreprocessor

---

- 6 [OK] ボタンをクリックし、[適用] ボタンをクリックして、アプリケーション・サーバの変更内容を保存します。これで、LoadRunnerEJB モニタを使用して、WebSphere サーバを起動したり停止したりできます。

### Oracle 9iAS サーバ

Oracle 9iAS アプリケーション・サーバでサポート・ファイルを設定して JDK 環境を設定したら、NT マシンでは **oc4jMonitor.cmd** ファイルを、UNIX マシンでは **oc4jMonitor.sh** ファイルを実行します。EJB モニタのサポート機能が有効になっているアプリケーション・サーバが起動します。

## クライアント・マシンでの EJB モニタの設定

EJB パフォーマンスを監視するには、EJB モニタで測定するカウンタを選択する必要があります。カウンタは、コントローラの [EJB モニタ設定] ダイアログ・ボックスを使って選択します。

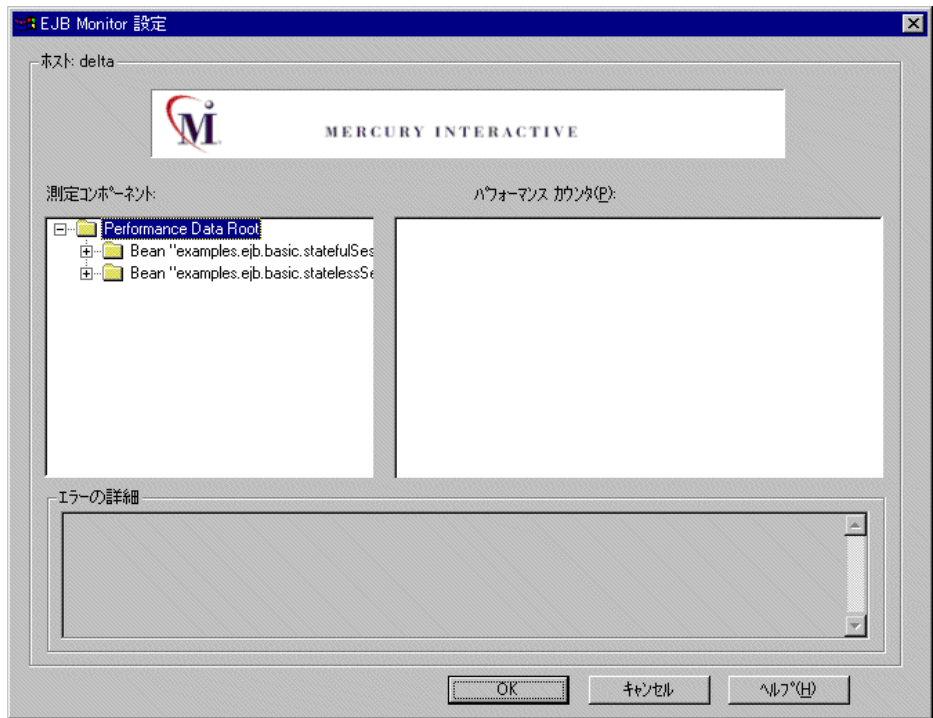
**EJB モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで EJB 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。[EJB] ダイアログ・ボックスが開きます。



- 3 [監視するサーバマシン] ボックスで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

- 4 [EJB] ダイアログ・ボックスの [リソース測定場所] セクションの [追加] をクリックします。[EJB Monitor 設定] ダイアログ・ボックスが開き、使用可能な EJB が表示されます。



- 5 [測定コンポーネント] ツリーを展開して、監視対象のメソッドとカウンタを選択します。

測定可能なパフォーマンス・カウンタの一覧については、559 ページを参照してください。

- 6 [EJB Monitor 設定] ダイアログ・ボックスと [EJB] ダイアログ・ボックスで [OK] をクリックすると、EJB モニタが起動します。

## EJB パフォーマンス・カウンタ

各メソッドで次のカウンタを監視できます。

| 測定値                            | 説明                           |
|--------------------------------|------------------------------|
| <b>Average Response Time</b>   | 監視対象 EJB オブジェクトの平均応答時間（ミリ秒）。 |
| <b>Method Calls per Second</b> | EJB オブジェクト・メソッドの秒ごとの呼び出しの回数。 |





# 第 29 章

---

## J2EE パフォーマンスの監視

J2EE パフォーマンス・モニタによりアプリケーション・サーバ上の J2EE コンポーネントを徹底的に検査できます（サーブレット、JSP、EJB、JNDI、JDBC、および DB SQL 呼び出し）。

本章では、以下の項目について説明します。

- ▶ アプリケーション・サーバへの J2EE モニタのインストール
- ▶ J2EE モニタの初期設定
- ▶ クライアント・マシンでの J2EE モニタの有効化
- ▶ アプリケーション・サーバ設定の変更例
- ▶ J2EE モニタのトラブルシューティング

## J2EE パフォーマンスの監視について

J2EE モニタでは J2EE コンポーネントごとに次の情報が表示されます。

- ▶ メソッド/クエリーごとの平均応答時間
- ▶ 秒ごとのメソッド呼び出し回数

J2EE アーキテクチャをこのように監視することで、システム内の動作状況の全体像を把握できます。エンド・ユーザから見た応答時間を、Web サーバの動作 (サーブレットおよび JSP の動作データ)、アプリケーション・サーバの動作 (JNDI および EJB の動作データ)、およびデータベース要求のバックエンド動作 (JDBC メソッドおよび SQL クエリーの動作) と非常に簡単に突き合わせて相関関係を把握することができます。

J2EE モニタを使用することで、LoadRunner で、J2EE コンポーネントに関する情報を収集するためにアプリケーション・サーバにインストールされたエージェントを利用して、シナリオ実行時に J2EE コンポーネントの測定値を分析できます。これらの測定値は、J2EE モニタに含まれる Web サーバを通じて、アプリケーション・サーバから LoadRunner コントローラに送り返されます。J2EE モニタは、IBM WebSphere、BEA WebLogic、Oracle 9iAS、JBoss などの主要なアプリケーション・サーバに対応しています。対応しているアプリケーション・サーバの詳細については、563 ページ「サポート対応表」を参照してください。

---

**注：**J2EE モニタには MSXML 3.0 以降が必要です (Internet Explorer 6.0 に含まれています)。MSXML 3.0 は Microsoft MSDN Web サイト (<http://msdn.microsoft.com/library/default.asp?url=/downloads/list/xmlgeneral.asp>) からインストールできます。

---

## アプリケーション・サーバへの J2EE モニタのインストール

J2EE オブジェクトを監視するには、まず J2EE モニタをアプリケーション・サーバ・マシンにインストールして起動する必要があります。次に、モニタで測定するカウンタを選択して、クライアント・マシンの J2EE モニタを設定します。

J2EE パフォーマンス・モニタを使って、シナリオ実行時に WebLogic, WebSphere, Oracle 9iAS または JBoss アプリケーション・サーバ上の Java 2 Platform, Enterprise Edition (J2EE) オブジェクトを監視できます。

### サポート対応表

| アプリケーション・サーバ | バージョン                   | プラットフォーム              |
|--------------|-------------------------|-----------------------|
| WebLogic     | 4.x, 5.x, 6.x, 7.0, 8.1 | Windows, Solaris, AIX |
| WebSphere    | 3.x, 4.x                | Windows, Solaris, AIX |
| Oracle 9iAS  | 1.0.2.2                 | Windows, Solaris, AIX |
| JBoss        | 2.4.x, 3.04             | Windows, Solaris, AIX |

アプリケーション・サーバに J2EE モニタをインストールするには、次の手順で行います。

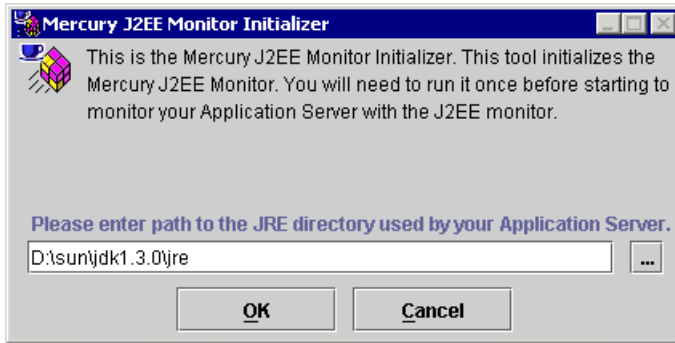
- 1 アプリケーション・サーバ・マシン上にホーム・ディレクトリ（たとえば、J2EEMonitor）を作成して、このディレクトリにインストール・ファイル <LoadRunner CD > ¥Add-ins¥J2EE¥jmonitor\_ <プラットフォーム> .jar ファイルを展開します。

インストール・ファイルを展開するための WinZip がない場合は、次のコマンドでインストール・ファイルを抽出してください。

```
< JDK > ¥bin¥jar.exe -xf <インストール・ファイル>
```

jar ファイルから抽出された UNIX スクリプトは、実行権限を失っている可能性があります。これを修正するには、J2EEMonitor ホーム・ディレクトリを開いて、chmod +x \*.sh コマンドを実行して権限を変更します。

- 2 < J2EEMonitor ホーム・ディレクトリ > %classes にある sipatool.jar ファイルをダブルクリックして、Mercury J2EE Monitor Initializer を起動します。



UNIX プラットフォームで作業している場合、または拡張子 **.jar** がシステムの中で Java 実行環境と関連付けられていない場合は、次のように **sipatool.jar** を実行します。

< J2EEMonitor ホーム・ディレクトリ > %classes ディレクトリから、`java -jar sipatool.jar` と入力します。

---

注：Mercury J2EE Monitor Initializer が表示されない場合は、次のように **sipatool.jar** を実行します。< J2EEMonitor ホーム・ディレクトリ > %classes ディレクトリから、< Java パス > % -jar sipatool.jar -nogui と入力します。アプリケーション・サーバ JDK の Java プロセスのフル・パスを使用してください。

---

- 3 Mercury J2EE Monitor Initializer で、アプリケーション・サーバの Java ホーム・ディレクトリへのパスを入力し、[OK] をクリックしてツールを実行します。
- 4 アプリケーション・サーバのコマンド・ライン引数に、`-Xbootclasspath/p: < J2EEMonitor ホーム・ディレクトリ > %classes%boot` を追加します。

WebLogic, WebSphere, Oracle 9iAS または JBoss アプリケーション・サーバの構文については、571 ページ「アプリケーション・サーバ設定の変更例」を参照してください。

## J2EE モニタの初期設定

アプリケーション・サーバへの J2EE モニタのインストールでは、フック機構、動作モード、JDBC および EJB 情報の取得について設定が行われます。

**フック機構**：J2EE モニタでは Mercury J2EE Monitor Initializer と Java フック・ライブラリを使用します。

**動作モード**：J2EE モニタでは、Auto Discovery 動作モードを使用します。このモードでは、ビジネス・プロセスに実際に参加している J2EE コンポーネント（サーブレット、JSP、JNDI、EJB および JDBC）がシステムによって自動的に検出されます。

**JDBC 情報の取得**：JDBC 情報取得の設定により、JDBC 呼び出しから返されるデータが決まります。標準設定では、J2EE モニタは JDBC の動作（たとえば、SELECT、UPDATE、CREATE）に基づいて測定データを伝送します。この設定を変更する方法については、566 ページ「JDBC 情報取得の設定」を参照してください。

**EJB 情報の取得**：EJB 情報取得の設定により、EJB 呼び出しから返されるデータが決まります。標準設定では、J2EE モニタはコンテナ・メソッド（ejbPassivate()、ejbCreate() など）を測定するように設定されていません。この設定を変更する方法については、566 ページ「EJB 情報取得の設定」を参照してください。

---

**注**：他の設定方法の詳細については、マーキュリー・インタラクティブのカスタマー・サポートにお問い合わせください。

---

## JDBC 情報取得の設定

### JDBC 情報取得の設定

- 1 < J2EEMonitor ホーム・ディレクトリ > %dat%\monitor.properties を開きます。
- 2 monitor.jdbc.mode プロパティに次のいずれかを入力します。
  - ▶ 他の（非 JDBC）測定メソッド呼び出し同様に JDBC メソッド呼び出しを測定する場合は、1 を入力。
  - ▶ JDBC の処理（たとえば、SELECT、UPDATE、CREATE）に基づいて測定データを集計する場合は、2 を入力。
  - ▶ 特定の SQL ステートメント（操作の内容、操作対象のテーブル、対象ステートメントの他のパラメータを含む）に基づいて測定データを集計する場合は、3 を入力。

---

注：長さが 3000 文字を越える SQL ステートメントはサポートされていません。

---

## EJB 情報取得の設定

コンテナ・メソッドを含むように EJB 情報の取得を設定するには、次の手順で行います。

- 1 < J2EEMonitor ホーム・ディレクトリ > %dat%\java\_monitor.ini を開きます。
- 2 このファイルの EJB\_CONFIG セクションで、hook\_files=auto\_detect の設定を次のように変更します。

```
hook_files=auto_detect_container
```

## クライアント・マシンでの J2EE モニタの有効化

J2EE パフォーマンスを監視するには、J2EE モニタで測定するカウンタを選択する必要があります。カウンタは、コントローラの [J2EE Monitor 設定] ダイアログ・ボックスを使って選択します。

**J2EE モニタを設定する前に、次のことを行います。**

Auto Discovery モード (J2EE モニタの標準動作モード) では、ビジネス・プロセスに参加しているコンポーネント (サーブレット, JSP, JNDI, EJB, JDBC) のメソッドがシステムによって検出され、そのオブジェクトだけが測定されます。

Auto Discovery プロセスを起動するには、アプリケーション・サーバを起動し、アプリケーション・サーバに対する負荷テストで使う仮想ユーザ・スクリプトを実行します。これによって、監視の対象にできる測定値のリストをコントローラから取得できるようになります。

---

**注：**次に同スクリプトを実行するときは、監視対象のメソッドとカウンタを選択する前に仮想ユーザを実行する必要はありません。

---

J2EE モニタを設定するには、次の手順で行います。

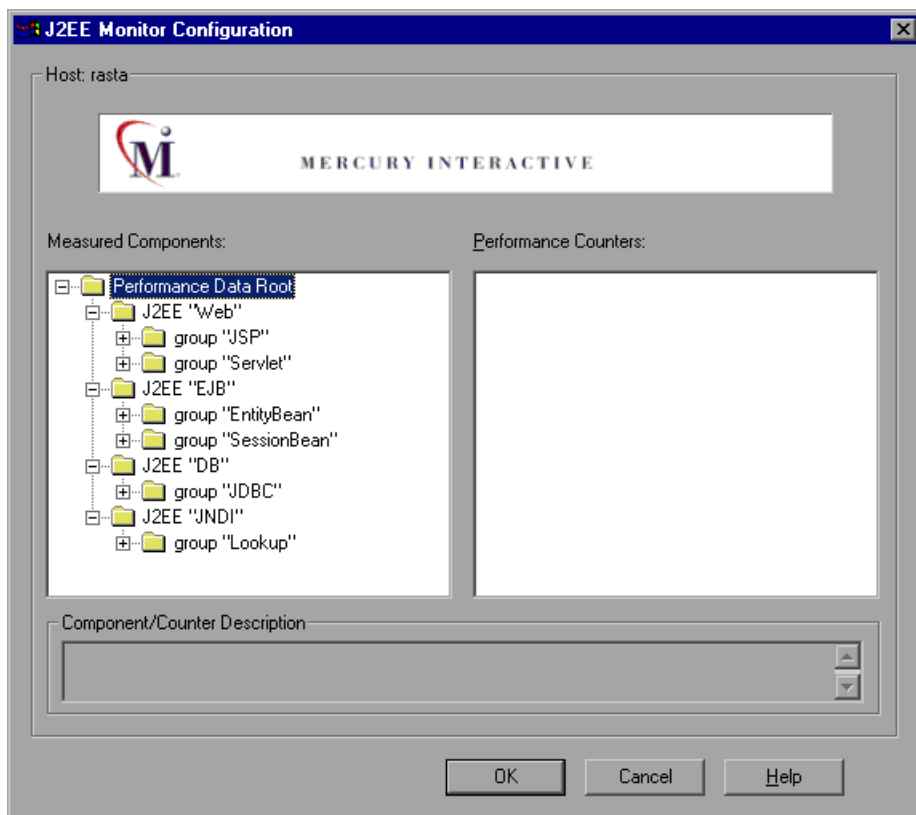
- 1 グラフ・ツリーで [J2EE] グラフをクリックして、[実行] ウィンドウの右の表示枠までドラッグします。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。[J2EE] ダイアログ・ボックスが開きます。



- 3 [監視するサーバマシン] ボックスで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。



- 4 [J2EE] ダイアログ・ボックスの [リソース測定場所] セクションの [追加] をクリックします。[J2EE Monitor 設定] ダイアログ・ボックスが開き、使用可能な J2EE カウンタが表示されます。



- 5 [測定コンポーネント] ツリーを展開して、監視対象のメソッドとカウンタを選択します。使用可能なパフォーマンス・カウンタの一覧については、570 ページを参照してください。

**注：**アナリシスに表示できる測定名の長さは 255 文字までです。測定名がこの上限を越えていると、カウンタ名が切り捨てられ、一意の ID (UID) が与えられます。同じカウンタで複数の異なるイベントを監視したり、クロス結果グラフを作成したりする場合、UID はそのまま変わらずに残ります。測定名の切り捨ては次のように行われます。標準プレフィックス / 切り捨てカウンタ名 < UID > / 監視対象イベント。

次に例を示します。

```
/DB/JDBC/weblogic.jdbc.rmi.SerialPreparedStatement/int executeUpdate()/INSERT INTO
orders (orderid _ userid _ orderdate _ shipaddr1 _ shipaddr2 _ shipcity _ shipstate _
shipzip _ shipcountry _ billaddr1 _ billaddr2 _ b <1> /Average Response Time
```

測定値の完全な名前は [測定値の説明] ボックスに表示されます。

- 6 [J2EE Monitor Configuration] ダイアログ・ボックスと [J2EE] ダイアログ・ボックスで [OK] をクリックすると、J2EE モニタが起動します。

## J2EE パフォーマンス・カウンタ

各メソッドで次のカウンタを監視できます。

| 測定値                            | 説明                              |
|--------------------------------|---------------------------------|
| <b>Average Response Time</b>   | 監視対象の J2EE オブジェクトの平均応答時間 (ミリ秒)。 |
| <b>Method Calls per Second</b> | 秒ごとの J2EE オブジェクト・メソッドの呼び出しの回数。  |

## アプリケーション・サーバ設定の変更例

アプリケーション・サーバにマーキュリー・インタラクティブの J2EE モニタ・ファイルをインストールした場合は、J2EE モニタのサポート機能を有効にした状態でサーバが実行されるように設定されています。この項では、次のアプリケーション・サーバの設定を変更する例を示します。

- ▶ WebLogic - バージョン 4.x ~ 5.x
- ▶ WebLogic - バージョン 6.x
- ▶ WebLogic - バージョン 7.x
- ▶ WebLogic - バージョン 8.1
- ▶ WebSphere サーバ - バージョン 3.x
- ▶ WebSphere サーバ - バージョン 4.x
- ▶ Oracle 9iAS サーバ
- ▶ JBoss 2.4.x ~ 3.04 サーバ

---

注：環境変数は、次の順序で設定してください。

---

### WebLogic サーバ

WebLogic サーバの設定方法は、バージョンによって異なります。

## WebLogic - バージョン 4.x ~ 5.x

WebLogic 4.x ~ 5.x サーバを設定するには、次の手順で行います。

- 1 < WebLogic ホーム > `startWeblogic.cmd` ファイルをバックアップするため、このファイルを < WebLogic ホーム > `¥startWeblogicMercury.cmd` にコピーします。
- 2 < WebLogic ホーム > `¥startWeblogicMercury.cmd` ファイルを開きます。
- 3 サーバの起動に使用している Java コマンド・ラインの直前に、次の変数を追加します。

Windows プラットフォームの場合：

```
set MERC_MONITOR_HOME= < J2EEMonitor ホーム・ディレクトリ >
set JAVA_CLASSPATH=%JAVA_CLASSPATH%;
%MERC_MONITOR_HOME%¥dat;
%MERC_MONITOR_HOME%¥classes¥xerces.jar
```

UNIX プラットフォームの場合 (csh)：

```
MERC_MONITOR_HOME= < J2EEMonitor ホーム・ディレクトリ >
JAVACLASSPATH=$JAVACLASSPATH:
$MERC_MONITOR_HOME/classes/xerces.jar
```

- 4 ファイルの同じセクションで、Java コマンド・ラインに次のパラメータを追加します。

`-Xbootclasspath/p:%MERC_MONITOR_HOME%¥classes¥boot`

例を示します。

```
%JAVA_HOME%¥bin¥java -ms64m -mx64m --
Xbootclasspath/p:%MERC_MONITOR_HOME%¥classes¥boot-
Dweblogic.class.path=%WEBLOGIC_CLASSPATH% -Dweblogic.home=-
Djava.security.manager-Djava.security.policy=.%¥weblogic.policy weblogic.Server
```

- 5 < WebLogic ホーム > `¥startWeblogicMercury.cmd` ファイルを実行します。

## WebLogic - バージョン 6.x

WebLogic 6.x サーバを設定するには、次の手順で行います。

- 1 < WebLogic ホーム > %config% < ドメイン名 > %startWeblogic.cmd ファイルをバックアップするため、このファイルを < WebLogic ホーム > %config% < ドメイン名 > %startWeblogicMercury.cmd にコピーします。
- 2 < WebLogic ホーム > %config% < ドメイン名 > %startWeblogicMercury.cmd ファイルを開きます。
- 3 サーバの起動を行う Java コマンド・ラインの直前に、次の変数を追加します。

Windows プラットフォームの場合：

```
set MERC_MONITOR_HOME= < J2EEMonitor ホーム・ディレクトリ >
set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes\xerces.jar
```

UNIX プラットフォームの場合：

```
MERC_MONITOR_HOME= < J2EEMonitor ホーム・ディレクトリ >
CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:
%MERC_MONITOR_HOME/classes/xerces.jar
```

- 4 ファイルの同じセクションで、コマンド・ラインに次のパラメータを追加します。  
-Xbootclasspath/p:%MERC\_MONITOR\_HOME%\classes%boot

例を示します。

```
"%JAVA_HOME%\bin\java" -hotspot -ms64m -mx64m -
Xbootclasspath/p:%MERC_MONITOR_HOME%\classes%boot -classpath
%CLASSPATH% -Dweblogic.Domain=mydomain-Dweblogic.Name=myserver "-
Dbea.home=f:%bea" "-
Djava.security.policy==f:%bea%\wlsrserver6.0/lib/weblogic.policy" -
Dweblogic.management.password=%WLS_PW% weblogic.Server
```

- 5 < WebLogic ホーム > %config% < ドメイン名 > %startWeblogicMercury.cmd ファイルを実行します。

## WebLogic - バージョン 7.x

WebLogic 7.x サーバを設定するには、次の手順で行います。

- 1 < WebLogic ホーム > %server%\bin\startwls.cmd ファイルをバックアップするため、このファイルを < WebLogic ホーム > %server%\bin\startwlsMercury.cmd にコピーします。
- 2 < WebLogic ホーム > %server%\bin\startwlsMercury.cmd ファイルを開きます。
- 3 サーバの起動を行う Java コマンド・ラインの直前に、次の変数を追加します。

Windows プラットフォームの場合：

```
set MERC_MONITOR_HOME= < J2EEMonitor ホーム・ディレクトリ >
set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes\xerces.jar
```

UNIX プラットフォームの場合：

```
MERC_MONITOR_HOME= < J2EEMonitor ホーム・ディレクトリ >
CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:
$MERC_MONITOR_HOME/classes/xerces.jar
```

- 4 ファイルの同じセクションで、コマンド・ラインに次のパラメータを追加します。

```
-Xbootclasspath/p:%MERC_MONITOR_HOME%\classes%boot
```

例を示します。

```
"%JAVA_HOME%\bin\java" -hotspot -ms64m -mx64m -
Xbootclasspath/p:%MERC_MONITOR_HOME%\classes%boot -classpath
%CLASSPATH% -Dweblogic.Domain=mydomain -Dweblogic.Name=myserver "-
Dbea.home=f:%bea" "-
Djava.security.policy==f:%bea%\wlsrver6.0/lib/weblogic.policy" -
Dweblogic.management.password=%WLS_PW% weblogic.Server
```

- 5 <ドメイン名> %startWeblogic.cmd ファイルをバックアップするため、このファイルを <ドメイン名> %startWeblogicMercury.cmd にコピーします。
- 6 <ドメイン名> %startWeblogicMercury.cmd ファイルを開きます。

- 7 Weblogic サーバ呼び出しを探します。次に例を示します。  
call D:\bea\weblogic700\server\bin\startWLS.cmd
- 8 この呼び出しの **startWLS.cmd** を **startWLSMercury.cmd** に変更して、ファイルを保存します。

## WebLogic - バージョン 8.1

**WebLogic 8.1** サーバを設定するには、次の手順で行います。

- 1 モニタ・マシンの **Monitor Home**で、次の行を変更します。

変更前 :

```
japata.hook.class.name=com.mercuryinteractive.japata.cjhook_mon.CJHookItJava
```

変更後 :

```
japata.hook.class.name=com.mercuryinteractive.japata.cjhook_mon.CJHookItC
```

- 2 < **WebLogic** ホーム > **samples** < ドメイン > 中にある、アプリケーションの起動に使用されるバッチ・ファイルのバックアップを作成します。バッチ・ファイルの名前は「start」で始まり、その後アプリケーションの名前が続きます (たとえば **startwlsMercury.cmd** ファイル)。
- 3 バッチ・ファイルの中で、サーバの起動を行う Java コマンド・ラインの直前に、次の変数を追加します。

Windows プラットフォームの場合 :

```
set MERC_MONITOR_HOME= <モニタのインストール先ディレクトリ>
set
CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\classes\jdom.jar;
%MERC_MONITOR_HOME%\classes\xalan.jar;%MERC_MONITOR_HOME%\c
lasses\xerces.jar;%MERC_MONITOR_HOME%\classes\xml-
apis.jar;%MERC_MONITOR_HOME%\classes;%MERC_MONITOR_HOME%\cla
sses\sqlkey.jar;%MERC_MONITOR_HOME%\classes\antlr.jarset
PATH=%PATH%;%MERC_MONITOR_HOME%\bin
```

UNIX プラットフォームの場合 :

```
set MERC_MONITOR_HOME= <モニタのインストール先ディレクトリ>
set
CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\classes\jdom.jar;
%MERC_MONITOR_HOME%\classes\xalan.jar;%MERC_MONITOR_HOME%\c
lasses\xerces.jar;%MERC_MONITOR_HOME%\classes\xml-
apis.jar;%MERC_MONITOR_HOME%\classes;%MERC_MONITOR_HOME%\c
lasses\sqlkey.jar;%MERC_MONITOR_HOME%\classes\antlr.jarset
PATH=%PATH%;%MERC_MONITOR_HOME%\bin
```

- 4 ファイルの同じセクションで、 コマンド・ラインに次のパラメータを追加します。  
-Xbootclasspath/p:%MERC\_MONITOR\_HOME%\classes\boot -Xrunjdhook  
例を示します。

```
%JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% -
Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot-Xrunjdkhook
%JAVA_OPTIONS% -Dweblogic.Name=%SERVER_NAME% -
Dweblogic.management.username=%WLS_USER% -
Dweblogic.management.password=%WLS_PW% -
Dweblogic.ProductionModeEnabled=%STARTMODE% -
Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy"
weblogic.Server
```

- 5 <ドメイン名> ¥startWeblogic.cmd を<ドメイン名>  
¥startWeblogicMercury.cmd という名前の新規ファイルとしてコピーします。
- 6 <ドメイン名> ¥startWeblogicMercury.cmd を開きます。
- 7 Weblogic サーバ呼び出しを探します。次に例を示します。

```
call D:\bea\weblogic700\server\bin\startWLS.cmd
```

- 8 この呼び出しの startWLS.cmd を startWLSMercury.cmd に変更して、 ファイル  
を保存します。



## WebSphere サーバ - バージョン 3.x

標準設定では、WebSphere 3.x アプリケーション・サーバは Windows 上で、マシンの起動時に自動サービスとして実行されます。マーキュリー・インタラクティブでは、現時点では自動サービスとして実行される WebSphere サーバで LoadRunner による J2EE の監視をサポートしていないため、標準設定の WebSphere サーバの起動方法を**手動**に変更する必要があります。

**標準設定の WebSphere 3.x サーバの起動方法を変更するには、次の手順で行います。**

- 1 [スタート] > [設定] > [コントロールパネル] を選択します。
- 2 [サービス] をダブルクリックします。
- 3 [IBM WS AdminServer] を選択して、[停止] ボタンをクリックします。
- 4 [IBM WS AdminServer] をダブルクリックして、[手動] 起動タイプを選択します。
- 5 [OK] をクリックし、設定を保存してダイアログ・ボックスを閉じます。

これで、自動サービスを使わずに、< WebSphere ホーム >  
¥AppServer¥bin¥debug¥adminserver.bat からサーバを起動できます。

**LoadRunner J2EE モニタのサポート機能を WebSphere 3.x サーバに追加するには、次の手順で行います。**

- 1 < WebSphere ホーム > ¥AppServer¥bin¥debug¥adminserver.bat ファイルのバックアップ・コピーを作成します。
- 2 < WebSphere ホーム > ¥AppServer¥bin¥debug¥adminserver.bat ファイルを開きます。

- 3 次の環境変数を SET\_CP セクションの末尾に追加します。

Windows プラットフォームの場合：

```
set MERC_MONITOR_HOME= < J2EEMonitor ホーム・ディレクトリ>
set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes\xerces.jar
```

UNIX プラットフォームの場合：

```
MERC_MONITOR_HOME= < J2EEMonitor ホーム・ディレクトリ>
CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:
$MERC_MONITOR_HOME/classes/xerces.jar
export MERC_MONITOR_HOME
```

- 4 **adminserver.bat** ファイルを実行します。
- 5 WebSphere 管理コンソール・アドバンスド版を開いて、[ビュー] > [トポロジ] を選択します。
- 6 [<サーバ・マシン名>] > [Default Server] を選択して、[WebSphere 管理ドメイン] ツリーを展開します。
- 7 [アプリケーション・サーバ:Default Server] ウィンドウで [一般] タブを選択します。
- 8 コマンド・ライン引数ボックスに次を追加します。

```
-Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
```

[適用] をクリックします。

JDK1.1.7 IBM が含まれる WebSphere 3.0 サーバで作業している場合は、[環境] をダブルクリックします。[変数名] ボックスに `_CLASSLOAD_HOOK`、[値] ボックスに `jdkhook` と入力します。[追加] ボタン、[OK] ボタン、[適用] ボタンをクリックします。

- 9 Windows 2000/NT, Solaris の場合は, [一般] タブから [Environment Editor] ダイアログ・ボックスを開き, 次の変数を [環境] ボックスに追加します。

Windows 2000/NT の場合 :

```
name=CLASSPATH
value= < J2EEMonitor ホーム・ディレクトリ> %dat
```

Solaris の場合 :

```
name=CLASSPATH
value= < J2EEMonitor ホーム・ディレクトリ> /dat
```

[OK] をクリックして, [Environment Editor] ダイアログ・ボックスを閉じます。

- 10 [WebSphere 管理コンソールアドバンスド版] を閉じます。
- 11 **adminserver.bat** ファイルを閉じて再起動します。

## WebSphere サーバ - バージョン 4.x

WebSphere 4.x サーバは, startServerBasic.bat ファイルまたは startServer.bat ファイルを使って起動できます。

**WebSphere 4.x サーバを設定するには, 次の手順で行います。**

- 1 WebSphere 管理サーバが実行されていることを確認し, 管理コンソールを起動します。
- 2 [WebSphere 管理ドメイン] ツリーで, ノード, ホスト名, アプリケーション・サーバのサブツリーを展開し, Default Server (または J2EE の対象にするアプリケーション・サーバ) を選択します。
- 3 「Default Server」を右クリックし, メニューから [プロパティ] を選択して, [一般] タブをクリックします。
- 4 Windows 2000/NT, Solaris の場合は, [一般] タブから [Environment Editor] ダイアログ・ボックスを開き, 次の変数を [環境] ボックスに追加します。

Windows 2000/NT の場合 :

```
name=CLASSPATH
value= < J2EEMonitor ホーム・ディレクトリ> %dat;
< J2EEMonitor のホーム・ディレクトリ> %classes%xerces.jar
```

Solaris の場合 :

```
name=CLASSPATH
value= < J2EEMonitor ホーム・ディレクトリ> /dat:
< J2EEMonitor のホーム・ディレクトリ> /classes/xerces.jar
```

[OK] をクリックして、[Environment Editor] ダイアログ・ボックスを閉じます。

- 5 [JVM の詳細設定] タブをクリックして、JVM の詳細設定を選択します。コマンド・ラインの引数フィールドに、Windows 2000/NT, Solaris, AIX に対して次の値を追加します。

```
-Xbootclasspath/p:%MERC_MONITOR_HOME%¥classes¥boot
```

- 6 [OK] ボタンをクリックし、[適用] ボタンをクリックして、アプリケーション・サーバの変更内容を保存します。これで、LoadRunner J2EE モニタを使用して、WebSphere サーバを起動したり停止したりできます。

## Oracle 9iAS サーバ

- 1 **env.cmd** ファイル (Unix プラットフォームでは **env.sh**) を次のように編集します。

—環境変数 **JAVA\_HOME** が、アプリケーション・サーバの実行に使用される Java 仮想マシンの場所を指すようにする。

—環境変数 **DETECTOR\_INS\_DIR** がモニタのインストール場所を指すようにする。

—環境変数 **APP\_SERVER\_DRIVE** でアプリケーション・サーバが格納されているドライブを指定する (たとえば, D:)。UNIX プラットフォームではこの変数を変更しないでください。

—環境変数 **APP\_SERVER\_ROOT** でアプリケーション・サーバのルート・ディレクトリを指定する。

- 2 **oc4jMonitor.cmd** (UNIX プラットフォームでは **oc4jMonitor.sh**) を実行します。

## JBoss 2.4.x ~ 3.04 サーバ

- 1 < JBoss ホーム > %run.bat (UNIX プラットフォームで run.sh) のバックアップ・コピーを、 < JBoss ホーム > %runMercury.bat (UNIX では runMercury.sh) に作成します。
- 2 < JBoss ホーム > %runMercury.bat ファイル (UNIX では runMercury.sh) を開きます。

サーバの起動に使用している Java コマンド・ラインの直前に、次の変数を追加します。

Windows プラットフォームの場合：

```
set MERC_MONITOR_HOME= < J2EEMonitor ホーム・ディレクトリ >
set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes\xerces.jar
```

UNIX プラットフォームの場合：

```
MERC_MONITOR_HOME < J2EEMonitor ホーム・ディレクトリ >
CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:
%MERC_MONITOR_HOME/classes/xerces.jar
```

- 3 ファイルの同じセクションで、コマンド・ラインに次のパラメータを追加します。  
-Xbootclasspath/p:%MERC\_MONITOR\_HOME%\classes%boot

例を示します。

```
%JAVA_HOME%\bin%java -ms64m -mx64m -
Xbootclasspath/p:%MERC_MONITOR_HOME%\classes%boot-
Dweblogic.class.path=%WEBLOGIC_CLASSPATH% -Dweblogic.home=. -
Djava.security.manager-Djava.security.policy==.%weblogic.policy weblogic.Server
```

- 4 < JBoss ホーム > %runMercury.bat ファイル (UNIX プラットフォームでは runMercury.sh) を実行します。

## J2EE モニタのトラブルシューティング

### 標準設定のポートの変更

J2EE モニタは、標準設定では、ポート 2004 を使用して LoadRunner と通信します。このポートがすでに他に使用されている場合は、次のようにして別のポートを選択できます。

- 1 アプリケーション・サーバ・マシンで、< **J2EEMonitor** のホーム・ディレクトリ > `%dat%\monitor.properties` を開き、プロパティ `webservice.monitor.port` に指定されているポート番号を変更します。
- 2 LoadRunner マシンで、< **LoadRunner** のインストール先フォルダ > `%dat%\monitors%\xmlmonitorshared.ini` を開き、「DefaultPort」キーの下にある `[mon_j2ee]` セクションに指定されているポート番号を変更します。

### 初期化エラー

「UnsupportedClassVersionError」、 「NoSuchMethodError」、 「NoClassDefFoundError」などのアプリケーション・サーバの初期化エラーが発生した場合、Mercury J2EE Monitor Initializer を使用して指定した JDK のバージョンと、アプリケーション・サーバの起動に使用された実際の JDK のバージョンが異なる可能性があります。

アプリケーション・サーバによって現在使用されている JDK と同じ JDK を選択していることを確認してください。アプリケーション・サーバを別の JDK で動作するように切り替えた場合は、Mercury J2EE Monitor Initializer を再実行する必要があります。

# 第30章

---

## Application Deployment ソリューション

LoadRunner の Application Deployment ソリューション・モニタを使えば、シナリオ実行時に Citrix MetaFrame XP または MetaFrame 1.8 サーバを監視して、サーバ・パフォーマンスのボトルネックを特定できます。

本章では、以下の項目について説明します。

- ▶ Citrix MetaFrame Server モニタの設定

### Application Deployment ソリューションの監視について

LoadRunner の Citrix MetaFrame XP モニタには、シナリオ実行時の Citrix MetaFrame XP および MetaFrame 1.8 サーバのアプリケーション配布の使用状況に関する情報が表示されます。パフォーマンス・データを取得するには、シナリオを実行する前に、サーバのオンライン・モニタを起動して測定対象のリソースを指定する必要があります。

## Citrix MetaFrame Server モニタの設定

Citrix サーバのパフォーマンスを監視するには、まずアプリケーション・サーバ・マシン上で Citrix MetaFrame XP モニタを起動し、Citrix サーバで監視するカウンタを有効にする必要があります。その後で、Citrix MetaFrame XP モニタで測定するカウンタを選択します。カウンタは、コントローラの [Citrix MetaFrame XP] ダイアログ・ボックスを使って選択します。

---

**注：**ファイアウォール越しに Citrix MetaFrame サーバを監視するために使用するポートは、サーバの構成によって異なります。

---

**モニタを設定する前に、次のことを行います。**

- 1 コントローラ・マシンから、ネットワーク・ドライブを Citrix サーバ・マシンに割り当てます。これにより、コントローラがリソース・カウンタにアクセスするのに必要な認証が与えられます。
- 2 コントローラ・マシンからパフォーマンス・モニタを起動して Citrix サーバ上でカウンタを有効にします。Citrix モニタ上の ICA セッション・オブジェクトと同じカウンタを監視できるようになります。
- 3 監視対象にできる測定値のリストをコントローラから取得できるようにするには、まず仮想ユーザを初期化してからシナリオを実行する必要があります。仮想ユーザを初期化したら、Citrix Monitor を設定して ICA セッション・カウンタを追加できます。

---

**注：**インスタンスを監視する測定項目は、現在実行中の Citrix セッションにのみ有効です。このシナリオを再度実行する場合は、インスタンスを対象にしている測定項目を再設定する必要があります。

---



---

**注**：さまざまなインスタンスを監視するには、サーバのログインとログアウト手順を、スクリプトの Action セクションではなく、**Vuser\_init** セクションと **Vuser\_end** セクションにそれぞれ記録します。詳細については、『**LoadRunner 仮想ユーザ・スクリプトの作成**』を参照してください。

---

**Citrix MetaFrame サーバ・モニタを設定するには、次の手順で行います。**

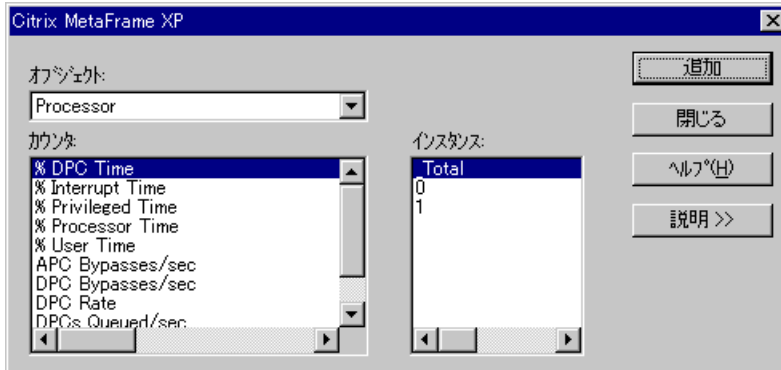
- 1 グラフ・ツリーで [Citrix MetaFrame XP] グラフをクリックして、[実行] ウィンドウの右の表示枠までドラッグします。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [Citrix MetaFrame XP] ダイアログ・ボックスの [監視するサーバマシン] セクションで [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [Citrix MetaFrame XP] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。測定可能なパフォーマンス・カウンタの一覧については、587 ページ「Citrix MetaFrame パフォーマンス・カウンタ」を参照してください。

---

**注**：[追加] をクリックした後にダイアログ・ボックスがフリーズした場合は、Citrix サーバ・マシンに localhost キャッシュを再構築する必要がある場合があります。詳細については、Citrix の Knowledge Base (<http://knowledgebase.citrix.com/cgi-bin/webcgi.exe?New,KB=CitrixKB>) でドキュメント番号 CTX003648 および CTX759510 を参照してください。

---

- 追加の測定値を選択するには、[追加] をクリックします。ダイアログ・ボックスが開き、Citrix オブジェクト、そのカウンタ、インスタンスが表示されます。カウンタを表示するオブジェクトを選択します。LoadRunner によって、選択したオブジェクトのカウンタが [カウンタ] 表示枠に表示されます。



- カウンタとインスタンスを選択します。**Ctrl** キーを使って、複数のカウンタを選択できます。インスタンスが意味を持つのは、強調表示されたカウンタの複数のインスタンスが実行されている場合だけです。各カウンタの説明を表示するには、[説明 >>] をクリックしてダイアログ・ボックスを拡張します。
- [追加] をクリックして、選択したカウンタをリソース・リストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。
- [Citrix MetaFrame XP] 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。

## Citrix MetaFrame パフォーマンス・カウンタ

次の表に、測定可能ないくつかのカウンタの説明を示します。

### 非仮想カウンタ

| 測定値                             | 説明                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>% Disk Time</b>              | 選択したディスク・ドライブで読み取りまたは書き込みの要求を処理するためにビジー状態となっていた経過時間の割合。                                                                                                                                                                                                                                                                                                                 |
| <b>% Processor Time</b>         | プロセッサが非アイドル・スレッドを実行している時間の割合。このカウンタは、プロセッサの動作状況を示す重要な指標となります。この値は、プロセッサがアイドル・プロセスのスレッドを実行するのに費やす時間をサンプル間隔ごとに測定し、その値を 100% から引くことによって算出されます（各プロセッサには、ほかのスレッドの実行準備ができていないときにサイクルを消費するアイドル・スレッドが割り当てられています）。この値は、あるサンプリング時点から次のサンプリング時点までの間に何らかの有用な処理を行うために費やされ時間の割合です。このカウンタは、サンプリング間隔の間に観察されたビジー状態の時間の平均な割合を示します。この値は、サービスがアクティブではなかった時間を監視し、その値を 100% から引くことによって算出されます。 |
| <b>File Data Operations/sec</b> | コンピュータがファイル・システム・デバイスに対する読み書き操作を行う頻度。これには、ファイル制御操作は含まれません。                                                                                                                                                                                                                                                                                                              |

| 測定値                              | 説明                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Interrupts/sec</b>            | <p>プロセッサが受け付けてサービスしている秒ごとのハードウェア割り込み数の平均。これには、DPC は含まれません。DPC は別にカウントされます。この値は、システム・クロック、マウス、ディスク・ドライブ、データ通信回線、ネットワーク・インタフェース・カードその他の周辺機器など、割り込みを生成するデバイスの動作状況を示す間接的な指標となります。通常これらのデバイスは、タスクの実行が完了したときや、対処を必要とするときに、プロセッサに割り込みをかけます。割り込みの間、通常のスレッドの実行は中断されます。ほとんどのシステム・クロックは、10 ミリ秒ごとにプロセッサに割り込みをかけることで、割り込み活動のバックグラウンドを作成します。このカウンタには、最後の2回のサンプリングで観測された値の差をサンプリング間隔で割った値が表示されます。</p> |
| <b>Output Session Line Speed</b> | <p>セッションにおけるサーバからクライアントへの回線速度 (bps)。</p>                                                                                                                                                                                                                                                                                                                                               |
| <b>Input Session Line Speed</b>  | <p>セッションにおけるクライアントからサーバへの回線速度 (bps)。</p>                                                                                                                                                                                                                                                                                                                                               |
| <b>Page Faults/sec</b>           | <p>プロセッサで発生したページ・フォルトの回数。ページ・フォルトは、プロセッサが、メイン・メモリ上のワーキング・セットに存在しない仮想メモリ・ページを参照したときに発生します。問題のページがスタンバイ・リスト上に存在する場合（したがってすでにメイン・メモリ中にある場合）、また、そのページを共有している別のプロセスによってそのページが使用されている場合には、ディスクからそのページが取り出されることはありません。</p>                                                                                                                                                                    |

| 測定値                              | 説明                                                                                                                                                                                                                                                                                                                                                  |
|----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Pages/sec</b>                 | 参照時にメモリに入っていなかったページへのメモリ参照を解決するために、ディスクから読み取られたページ数またはディスクに書き込まれたページ数。このカウンタは、 <b>Pages Input/sec</b> および <b>Pages Output/sec</b> の合計です。このカウンタには、システム・キャッシュに代わってアプリケーションのファイル・データにアクセスするためのページング・トラフィックが含まれます。この値には、キャッシュ対象外マップ済みメモリ・ファイルとメモリの間で読み書きされるページも含まれます。メモリが過度に使用される点（つまり、スラッシング）、およびその結果生じる可能性のある過剰なページングが気になる場合には、このカウンタを観察することが重要になります。 |
| <b>Pool Nonpaged Bytes</b>       | ページング対象外プールのバイト数。ページング対象外プールはシステム・メモリ領域の 1 つで、オペレーティング・システムの各コンポーネントが指定されたタスクを実行するとき、この領域に一定の空間を確保します。ページング対象外プールのページはページング・ファイルにページ・アウトすることはできず、割り当てられている限りメイン・メモリに残ります。                                                                                                                                                                           |
| <b>Private Bytes</b>             | このプロセスによって割り当てられ、ほかのプロセスとは共有できないバイト数の最新の値。                                                                                                                                                                                                                                                                                                          |
| <b>Processor Queue Length</b>    | スレッド数で表されるプロセッサ・キューの瞬間的な長さ。このカウンタは、スレッド・カウンタの監視も行われていなければ、常に 0 です。プロセッサはすべて、スレッドがプロセッサ・サイクルを待機する単独のキューを使用します。この長さには、現在実行中のスレッドは含まれません。一般に、プロセッサ・キューが常に 2 より長い場合、プロセッサが輻輳状態であることを意味します。このカウンタは瞬間的な値を示すもので、一定時間における平均値ではありません。                                                                                                                        |
| <b>Threads</b>                   | データ収集時のコンピュータのスレッド数。このカウンタは瞬間的な値を示すもので、一定時間における平均値ではありません。スレッドとは、プロセッサで命令を実行できる、基本的な実行単位です。                                                                                                                                                                                                                                                         |
| <b>Latency - Session Average</b> | セッションの有効期間におけるクライアント遅延の平均。                                                                                                                                                                                                                                                                                                                          |

| 測定値                                | 説明                                      |
|------------------------------------|-----------------------------------------|
| <b>Latency - Last Recorded</b>     | このセッションに対して最後に記録された遅延測定値。               |
| <b>Latency - Session Deviation</b> | セッションに対して測定された最小値と最大値との差。               |
| <b>Input Session Bandwidth</b>     | セッションにおけるクライアントからサーバへのトラフィックの帯域幅 (bps)。 |
| <b>Input Session Compression</b>   | セッションにおけるクライアントからサーバへのトラフィックの圧縮率。       |
| <b>Output Session Bandwidth</b>    | セッションにおけるサーバからクライアントへのトラフィックの帯域幅 (bps)。 |
| <b>Output Session Compression</b>  | セッションにおけるサーバからクライアントへのトラフィックの圧縮率。       |
| <b>Output Session Linespeed</b>    | セッションにおけるサーバからクライアントへの回線速度 (bps)。       |

### 仮想チャネル・カウンタ

| 測定値                                    | 説明                                                        |
|----------------------------------------|-----------------------------------------------------------|
| <b>Input Audio Bandwidth</b>           | オーディオ・マッピング・チャネルにおけるクライアントからサーバへのトラフィックの帯域幅 (bps)。        |
| <b>Input Clipboard Bandwidth</b>       | クリップボード・マッピング・チャネルにおけるクライアントからサーバへのトラフィックの帯域幅 (bps)。      |
| <b>Input COM1 Bandwidth</b>            | COM1 チャネルにおけるクライアントからサーバへのトラフィックの帯域幅 (bps)。               |
| <b>Input COM2 Bandwidth</b>            | COM2 チャネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。              |
| <b>Input COM Bandwidth</b>             | COM チャネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。               |
| <b>Input Control Channel Bandwidth</b> | ICA コントロール・チャネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。        |
| <b>Input Drive Bandwidth</b>           | クライアントのドライブ・マッピング・チャネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。 |

| 測定値                               | 説明                                                                        |
|-----------------------------------|---------------------------------------------------------------------------|
| <b>Input Font Data Bandwidth</b>  | ローカル・テキスト・エコー・フォントおよびキーボード・レイアウト・チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。 |
| <b>Input Licensing Bandwidth</b>  | ライセンスング・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                          |
| <b>Input LPT1 Bandwidth</b>       | LPT1 チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。                             |
| <b>Input LPT2 Bandwidth</b>       | LPT2 チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。                             |
| <b>Input Management Bandwidth</b> | クライアント管理チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。                          |
| <b>Input PN Bandwidth</b>         | プログラム隣接チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅。この値は bps 単位で測定されます。              |
| <b>Input Printer Bandwidth</b>    | プリンタ・スプーラ・チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。                        |
| <b>Input Seamless Bandwidth</b>   | シームレス・チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。                            |
| <b>Input Text Echo Bandwidth</b>  | ローカル・テキスト・エコー・データ・チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。                |
| <b>Input Thinwire Bandwidth</b>   | Thinwire (グラフィックス) チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。               |
| <b>Input VideoFrame Bandwidth</b> | VideoFrame チャンネルにおける、クライアントからサーバへのトラフィックの帯域幅 (bps)。                       |
| <b>Output Audio Bandwidth</b>     | オーディオ・マッピング・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                      |
| <b>Output Clipboard Bandwidth</b> | クリップボード・マッピング・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                    |
| <b>Output COM1 Bandwidth</b>      | COM1 チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                             |

| 測定値                                     | 説明                                                                        |
|-----------------------------------------|---------------------------------------------------------------------------|
| <b>Output COM2 Bandwidth</b>            | COM2 チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                             |
| <b>Output COM Bandwidth</b>             | COM チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                              |
| <b>Output Control Channel Bandwidth</b> | ICA コントロール・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                       |
| <b>Output Drive Bandwidth</b>           | クライアント・ドライブ・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                      |
| <b>Output Font Data Bandwidth</b>       | ローカル・テキスト・エコー・フォントおよびキーボード・レイアウト・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。 |
| <b>Output Licensing Bandwidth</b>       | ライセンスング・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                          |
| <b>Output LPT1 Bandwidth</b>            | LPT1 チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                             |
| <b>Output LPT2 Bandwidth</b>            | LPT2 チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                             |
| <b>Output Management Bandwidth</b>      | クライアント管理チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                          |
| <b>Output PN Bandwidth</b>              | プログラム隣接チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                           |
| <b>Output Printer Bandwidth</b>         | プリンタ・スプーラ・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                        |
| <b>Output Seamless Bandwidth</b>        | シームレス・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                            |
| <b>Output Text Echo Bandwidth</b>       | ローカル・テキスト・エコー・データ・チャンネルにおける、サーバからクライアントへのトラフィックの帯域幅 (bps)。                |



| 測定値                                | 説明                                                           |
|------------------------------------|--------------------------------------------------------------|
| <b>Output Thinwire Bandwidth</b>   | Thinwire (グラフィックス) チャンネルにおける, サーバからクライアントへのトラフィックの帯域幅 (bps)。 |
| <b>Output VideoFrame Bandwidth</b> | VideoFrame チャンネルにおける, サーバからクライアントへのトラフィックの帯域幅 (bps)。         |



# 第 31 章

## ミドルウェアのパフォーマンス監視

LoadRunner のミドルウェア・パフォーマンス・モニタを使えば、シナリオ実行時に Tuxedo および IBM WebSphere MQ サーバを監視して、サーバ・パフォーマンスのボトルネックを特定できます。

本章では、以下の項目について説明します。

- ▶ Tuxedo モニタの設定
- ▶ IBM WebSphere MQ モニタの設定

### ミドルウェアのパフォーマンスの監視について

トランザクションの応答時間における最大の要因は、ミドルウェアのパフォーマンスの状況です。LoadRunner のミドルウェア・パフォーマンス・モニタは、シナリオ実行時の Tuxedo および IBM WebSphere MQ サーバのミドルウェア・パフォーマンスの状況に関する情報を提供します。パフォーマンス・データを取得するには、シナリオを実行する前に、サーバのオンライン・モニタを起動して測定対象のリソースを指定する必要があります。

Tuxedo モニタでは、Tuxedo システムのサーバ、ロード・ジェネレータ・マシン、ワークステーション・ハンドラ、キューを監視できます。Tuxedo モニタを実行するには、監視対象マシンに Tuxedo クライアント・ライブラリをインストールする必要があります。

IBM WebSphere MQ (バージョン 5.x) サーバでのチャンネルおよびキューのパフォーマンス・カウンタの監視には、IBM WebSphere MQ を使用します。

モニタ測定値の選択手順およびモニタの設定手順は、サーバの種類によって異なります。以降の項では、サーバの種類ごとに設定手順を説明します。

## Tuxedo モニタの設定

Tuxedo モニタを使って、Tuxedo クライアントのパフォーマンスを測定して、表示できます。

コントローラ・マシンに Tuxedo 7.1 以降がインストールされていれば、一度に複数の Tuxedo アプリケーション・サーバを監視できます。ただし、コントローラ・マシンに Tuxedo 6.5 以前がインストールされている場合は、一度に監視できる Tuxedo アプリケーション・サーバは 1 つだけです。

**モニタを設定する前に、次のことを行います。**

- 1 Tuxedo ワークステーション・クライアント（ネイティブ・クライアントでなく）がコントローラ・マシンにインストールされていることを確認します。Tuxedo 6.x サーバを使用している場合は Tuxedo 6.x クライアントを、Tuxedo 7.1 以降のサーバを使用している場合は Tuxedo 7.1 以降のクライアントを使用します。Tuxedo 6.5 またはそれ以前のサーバを使用している場合でも、WSINTOPPRE71 環境変数を「yes」に設定すれば、Tuxedo 7.1 以降のクライアントを使ってサーバを監視できます。

---

**注：**Tuxedo ワークステーション・クライアントはネットワークをまたいでアプリケーション・サーバと通信するため、同じマシン上で Tuxedo アプリケーション・サーバを実行する必要はありません。ネイティブ・クライアントは、Tuxedo アプリケーション・サーバが同じ Tuxedo ドメインに属している場合だけでなく、その Tuxedo アプリケーション・サーバと通信できます。

---

- 2 コントローラ・マシンで、TUXDIR 変数に Tuxedo のインストール先ディレクトリを設定し（たとえば V:\environ\Tuxedo8.0）、PATH 変数に Tuxedo の bin ディレクトリを追加して、Tuxedo 環境変数を定義します。
- 3 ワークステーション・リスナー（WSL）プロセスが実行されるように、Tuxedo アプリケーション・サーバを設定します。これによって、アプリケーション・サーバがワークステーション・クライアントからの要求を受け付けるようになります。アプリケーション・サーバへの接続に使用するアドレスとポート番号は、WSL プロセス専用のアドレスとポート番号と同じである必要があります。

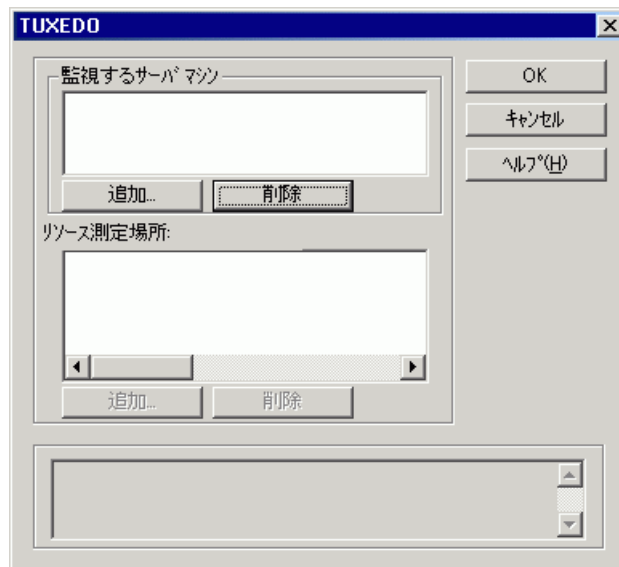
---

注：WSL の設定の詳細については、BEA Tuxedo の Web サイト (<http://edocs.beasys.com/tuxedo/tux81/rf5/rf5101.htm#1534543>) を参照してください。

---

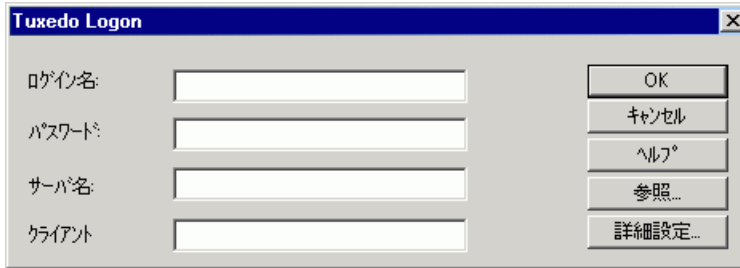
**Tuxedo モニタを設定するには、次の手順で行います。**

- 1 グラフ・ツリーで [Tuxedo] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。[Tuxedo] ダイアログ・ボックスが表示されます。



- 3 [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。マシンを実行するプラットフォームを選択して、[OK] をクリックします。

- 4 Tuxedo サーバにログオンするために、[リソース測定場所] 項目で [追加] をクリックします。ログイン名、パスワード、サーバ名、クライアント名など、Tuxedo サーバに関する情報を入力します。以前に監視された Tuxedo サーバがある場合には、そのサーバの名前が [サーバ名] ボックスに表示されます。



---

**注：**ログイン情報は、記録済みのスクリプトのディレクトリにある **tpinit.ini** ファイルの Logon セクションにあります。値を手作業で入力するよりも、[参照] ボタンを使用して記録済みのスクリプトから **tpinit.ini** ファイルを選択することをお勧めします。

---

**tpinit.ini** ファイルを使用して Tuxedo モニタの正しい設定を取得するには、[参照] ボタンをクリックして、その LoadRunner スクリプトの **tpinit.ini** ファイルを探します。また、記録済みのスクリプトの **lrt\_tpinitialize** ステートメントからクライアント名を判断することもできます。

次の **tpinit.ini** ファイルの例では、Tuxedo モニタは、**bankapp** という名前のクライアントと、65535 番ポートを使用している **URANUS** というサーバを使うように設定されています。ログオン・ユーザ名は **Smith**、パスワードは **mypasswd** です。

```
[Logon]
LogonServername=//URANUS:65535
LogonUserName=Smith
LogonCltName=bankapp
LogonGrpName=
LogonPasswd=myspasswd
LogonData=
```

すでに必要な値がわかっている場合は、ダイアログ・ボックスに手作業で入力できます。サーバ名の形式は、// <マシン名> : <ポート番号> です。マシン名の代わりに IP アドレスの指定も可能です。古いバージョンの Tuxedo で使用される 16 進数形式もサポートされます。引用符は使用できません。

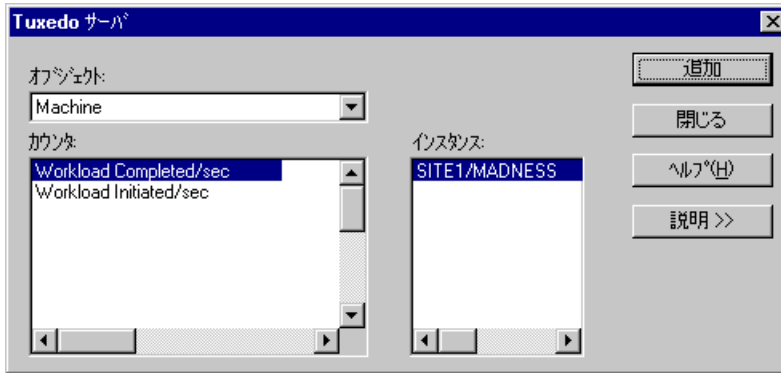
Tuxedo モニタを認証する必要がある場合は、[**詳細設定**] ボタンをクリックし、データ・ボックスに認証データを 16 進文字列（「0x」で始まる文字列）として入力します。認証データの値は、既存の Tuxedo アプリケーションの **tpinit.ini** ファイルから取得できます。

---

**注：**Tuxedo 6.5 またはそれ以前のバージョンを使用している場合、モニタは、コントローラ・セッション中に 1 つのアプリケーション・サーバにだけ接続できます。アプリケーション・サーバに接続されると、コントローラを閉じるまで、そのサーバだけがモニタによって使用されます。モニタからすべてのカウンタが削除されても、この状態は続きます。

---

- 5 [OK] をクリックします。[Tuxedo サーバ] ダイアログ・ボックスが開きます。



- 6 [オブジェクト] リストから Tuxedo オブジェクトを選択します。監視対象の測定値とインスタンスを選択します。使用可能なパフォーマンス・カウンタの一覧については、600 ページを参照してください。
- 7 [追加] をクリックして、選択したカウンタをリソースのリストに追加します。必要なリソースをすべてリストに追加したら、[閉じる] をクリックします。
- 8 [Tuxedo] 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。

### Tuxedo パフォーマンス・カウンタ

以下の表に、使用可能な Tuxedo モニタ測定項目を示します。

| モニタ    | 測定項目                                                                                                                 |
|--------|----------------------------------------------------------------------------------------------------------------------|
| Server | <b>Request per second</b> – 処理された秒ごとのサーバ要求数。                                                                         |
|        | <b>Workload per second</b> – 作業負荷 (Workload) とは、サーバ要求の加重測定値のことです。要求の中には、ほかと異なる重みを持つものもあります。標準では、作業負荷は常に要求の数の 50 倍です。 |



| モニタ                              | 測定項目                                                                                                                    |
|----------------------------------|-------------------------------------------------------------------------------------------------------------------------|
| <b>Machine</b>                   | <b>Workload completed per second</b> 一作業が完了したマシンの全サーバにおける単位時間当たりの作業負荷の合計。                                               |
|                                  | <b>Workload initiated per second</b> 一作業が開始されたマシンの全サーバにおける単位時間当たりの作業負荷の合計。                                              |
|                                  | <b>Current Accessers</b> 一このマシンで直接、またはこのマシンのワークステーション・ハンドラを通じて、アプリケーションに現在アクセスしている、クライアントおよびサーバの数。                      |
|                                  | <b>Current Clients</b> 一現在このマシンにログインしている、ネイティブおよびワークステーションの両方のクライアントの数。                                                 |
|                                  | <b>Current Transactions</b> 一このマシンの使用中トランザクション・テーブル・エントリの数。                                                             |
| <b>Queue</b>                     | <b>Bytes on queue</b> 一キューで待機している全メッセージの合計バイト数。                                                                         |
|                                  | <b>Messages on queue</b> 一キューで待機している要求の総数。この値は標準では0です。                                                                  |
| <b>Workstation Handler (WSH)</b> | <b>Bytes received per second</b> 一ワークステーション・ハンドラによって受信された、単位時間当たりの合計バイト数。                                               |
|                                  | <b>Bytes sent per second</b> 一ワークステーション・ハンドラによってクライアントに返された、単位時間当たりの合計バイト数。                                             |
|                                  | <b>Messages received per second</b> 一ワークステーション・ハンドラによって受信された、単位時間当たりのメッセージ数。                                            |
|                                  | <b>Messages sent per second</b> 一ワークステーション・ハンドラによってクライアントに返された、単位時間当たりのメッセージ数。                                          |
|                                  | <b>Number of queue blocks per second</b> 一ワークステーション・ハンドラのキューがブロックした、単位時間当たりの回数。これによって、ワークステーション・ハンドラが過負荷状態になった頻度がわかります。 |

## IBM WebSphere MQ モニタの設定

IBM WebSphere MQ モニタを使用するには、まず IBM WebSphere MQ クライアントをインストールし、イベントの監視を行うように MQ サーバ環境を設定し、[IBM WebSphere MQ - 測定値の追加] ダイアログ・ボックスを使用して監視する測定値を選択する必要があります。

---

**注：** LoadRunner IBM WebSphere MQ モニタがサポートするのは、Windows プラットフォーム上の MQ サーバだけです。

---

### IBM WebSphere MQ サーバへの接続

IBM WebSphere MQ モニタは（コントローラ・マシンにインストールされた MQ Client Connection を介して）IBM WebSphere MQ サーバに接続します。MQ クライアント環境では、MQ はクライアント・マシンで実行されません。クライアント・マシンは MQ サーバのインスタンスに接続し、サーバのリソースをクライアント・マシンのローカル・リソースであるかのように使用します。

IBM WebSphere MQ モニタによって、Windows 監視用に IBM MQ サーバ（バージョン 5.2）を実行しているマシンのリソースの使用状況が表示されます。

**モニタを設定する前に、次のことを行います。**

IBM WebSphere MQ Client Connection（バージョン 5.21 のみ）がコントローラ・マシンにインストールされていることを確認します。

---

**注：** IBM WebSphere MQ サーバとクライアントの詳細については、IBM MQSeries Web サイト（<http://www-3.ibm.com/software/ts/mqseries/library/manuals/index.htm>）を参照してください。

---

## サーバ環境の設定（イベント監視）

LoadRunner MQ モニタでは、2 つの標準 MQSeries キューからだけイベント・メッセージを取得します。

- ▶ SYSTEM.ADMIN.PERFM.EVENT — 「queue depth high」などのパフォーマンス・イベント
- ▶ SYSTEM.ADMIN.CHANNEL.EVENT — 「channel stopped」などのチャンネル・イベント

キュー・マネージャ（および、多くの場合、対応するオブジェクト）のイベントを有効にする必要があります。パフォーマンス・イベントは MQ サーバ上でキューの属性を設定して有効にします。チャンネル・イベントは標準設定で有効になっており、無効にできません。

---

**注：** IBM WebSphere MQ モニタは、キュー・マネージャが再起動された後、キュー・マネージャからデータを取得しません。

---

キュー・マネージャのパフォーマンス・イベントを有効にするには、次の手順で行います。

- 1 MQSC コマンド、ALTER QMGR PERFM EV (ENABLED) を使用します。
- 2 次のキューの属性を設定します。

| 測定値                             | 設定するイベント属性                                                                                                                                                                                                                                          |
|---------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Event - Queue Depth High</b> | <ul style="list-style-type: none"> <li>• QDEPTHHI(integer) - integer はメッセージ数の上限に対する割合を表す値で、範囲は 0 ~ 100。</li> <li>• QDPHIEV(action) - action は「ENABLED」、「DISABLED」のいずれかで、それぞれイベントの生成を有効または無効にします。</li> </ul>                                         |
| <b>Event - Queue Depth Low</b>  | <p>キューのイベントを有効にするには、次のキューの属性を設定します。</p> <ul style="list-style-type: none"> <li>• QDEPTHHO(integer) - integer はメッセージ数の上限に対する割合を表す値で、範囲は 0 ~ 100。</li> <li>• QDPLOEV(action) - action は「ENABLED」、「DISABLED」のいずれかで、それぞれイベントの生成を有効または無効にします。</li> </ul> |

| 測定値                                        | 設定するイベント属性                                                                                                                                                                                                                                                                                                            |
|--------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Event - Queue Full</b>                  | <ul style="list-style-type: none"> <li>• QDEPTHHI(integer) - integer はメッセージ数の上限に対する割合を表す値で、範囲は 0 ~ 100。</li> <li>• QDPMAXEV(action) - action は「ENABLED」、「DISABLED」のいずれかで、それぞれイベントの生成を有効または無効にします。</li> </ul>                                                                                                          |
| <b>Event - Queue Service Interval High</b> | <ul style="list-style-type: none"> <li>• QSVCINT(integer) - integer はミリ秒単位の値で、範囲は 0 ~ 999,999,999。<br/>注：この値は Queue Service Interval OK と共有されます。</li> <li>• QSVCIEV(type) - type は「HIGH」、「OK」、「NONE」いずれかで、それぞれ、Queue Service Interval High イベントを有効に、Queue Service Interval OK イベントを有効に、またはイベント生成を無効にします。</li> </ul>   |
| <b>Event - Queue Service Interval OK</b>   | <ul style="list-style-type: none"> <li>• QSVCINT(integer) - integer はミリ秒単位の値で、範囲は 0 ~ 999,999,999。<br/>注：この値は Queue Service Interval High と共有されます。</li> <li>• QSVCIEV(type) - type は「HIGH」、「OK」、「NONE」いずれかで、それぞれ、Queue Service Interval High イベントを有効に、Queue Service Interval OK イベントを有効に、またはイベント生成を無効にします。</li> </ul> |

---

注：MQ Server のエラー・メッセージ (MQRC\_ で始まる) が発生した場合、IBM MQSeries Web サイトの「Reason Codes」のセクション (<http://www-3.ibm.com/software/ts/mqseries/library/manuals/mqw20/AMQ43M32.HTM#HDRMQS> CRN) を参照してください。

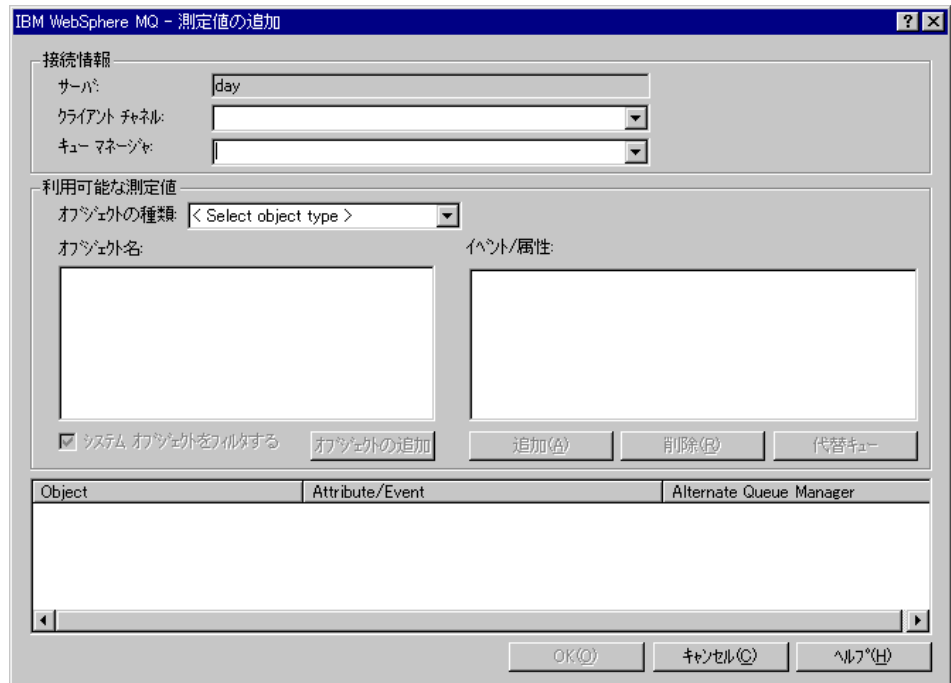
---

## IBM WebSphere MQ モニタの設定

MQ Client をコントローラにインストールし、イベントを監視するようにサーバ環境を設定すれば、測定対象のリソースを指定できます。

IBM WebSphere MQ モニタを設定するには、次の手順で行います。

- 1 グラフ・ツリーで [IBM WebSphere MQ] 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。
- 2 対象のグラフを右クリックして、[測定値の追加] を選択するか、[モニタ] > [測定値の追加] を選択します。
- 3 [IBM WebSphere MQ] ダイアログ・ボックスの [監視するサーバマシン] 項目で [追加] をクリックして、監視対象のマシンのサーバ名または IP アドレスを入力します。サーバ名の形式は、<マシン名> : <ポート番号> です。マシンを実行するプラットフォームを選択して、[OK] をクリックします。
- 4 [IBM WebSphere MQ] ダイアログ・ボックスの [リソース測定場所] セクションで [追加] をクリックして、監視対象の測定値を選択します。[IBM WebSphere MQ - 測定値の追加] ダイアログ・ボックスが開きます。



- 5 [接続情報] セクションの [クライアントチャネル] ボックスに、クライアントが MQ サーバへの接続に使用するチャンネルの名前を入力します。

MQ サーバのインスタンスで特定のチャンネルを設定することも、標準設定の「SYSTEM.DEF.SVRCONN」チャンネルを使用することもできます。クライアント・チャンネルが未定義の場合、MQ サーバはクライアント接続を通してアクセスできません (MQ モニタは、監視対象のキュー・マネージャに接続できないので、動作しません)。

---

**注:** テキスト・ボックスへのユーザ入力 は 48 文字に制限されています。

---

- 6 監視対象となるキュー・マネージャの名前を [キューマネージャ] ボックスに入力します。

このモニタによる監視の対象は、接続されているキュー・マネージャに限定されません。中央のキュー・マネージャに複数のキュー・マネージャから書き込むように設定して、監視を集中化できます (これはイベントの場合だけであり、ポーリング対象オブジェクト属性には適用されません)。すべてのイベントに、そのソースを示すキュー・マネージャ属性が含まれています。

---

**注:** キュー・マネージャには、一度に 1 つのコントローラまたは監視アプリケーションだけがアクセスできます。

---

- 7 [利用可能な測定値] のセクションで、オブジェクトの種類を選択します。

それまでに追加したオブジェクトの中から、選択したオブジェクトの種類に属するオブジェクトのリストが [オブジェクト名] リストに表示されます。選択したオブジェクトの種類に対応する属性またはイベントのリストが [イベント / 属性] リストに表示されます。

監視対象オブジェクト、選択したイベントまたは属性、および代替キュー・マネージャが監視対象オブジェクト表示枠に表示されます。

- 8 標準設定では、ユーザ定義オブジェクトが [オブジェクト名] リストに表示されます。すべてのオブジェクトを表示するには、[システム オブジェクトをフィルタする] チェック・ボックスをクリアします。フィルタ設定は、

< LoadRunner のインストール先フォルダ > \dat\monitors\mqseries.cfg ファイルで変更できます。

- 9 オブジェクトを選択するか、[オブジェクト名] リストに新規オブジェクトを追加します。新規オブジェクト名を追加するには、[オブジェクトの追加] ボタンをクリックします。[オブジェクトの追加] ダイアログ・ボックスで、監視対象オブジェクトの名前を入力して、[OK] をクリックします。ダイアログ・ボックスが閉じて、オブジェクトの名前が [オブジェクト名] リストに表示されます。
- 10 [イベント/属性] ボックスで、測定する属性またはイベントを選択します。属性またはイベントのリストは選択したオブジェクトの種類に適用されます。  
使用可能なパフォーマンス・カウンタの一覧については、609 ページを参照してください。

---

**注：**キューのイベントを有効にするには、そのキューの属性を設定しなければなりません。詳細については、603 ページ「サーバ環境の設定（イベント監視）」を参照してください。

---

- 11 監視の対象となるイベントのソースがリモートのキュー・マネージャ（[IBM WebSphere MQ - 測定値の追加] ダイアログ・ボックスの [キュー・マネージャ] フィールドで指定したもの以外）の場合は、[代替キュー] ボタンをクリックします。[代替キュー・マネージャ] ダイアログ・ボックスにキュー・マネージャの名前を入力して、[OK] をクリックします。

---

**注：**代替キュー・マネージャを追加すると、そのキュー・マネージャが、以降追加するイベントの標準設定のキュー・マネージャになります。接続しているキュー・マネージャに戻るには、[代替キュー・マネージャ] ダイアログ・ボックスにその名前を入力します。

---

- 12 [追加] をクリックして、オブジェクト測定値を監視対象オブジェクトのリストに追加します。オブジェクトの名前、そのイベントと属性、および代替キュー・マネージャが監視対象オブジェクト表示枠に表示されます。

- 13 監視対象オブジェクトのイベントまたは属性を削除するには、監視対象オブジェクト表示枠でオブジェクト測定値を選択して、**[削除]**をクリックします。監視対象オブジェクトのリストからそのエントリが削除されます。
- 14 必要なカウンタをすべてリストに追加したら、**[OK]**をクリックします。**[IBM WebSphere MQ]** ダイアログ・ボックスが開いて、監視対象サーバ・マシンの名前、選択したリソース測定値のリスト、および各測定値の説明が表示されます。



- 15 **[IBM WebSphere MQ]** 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。



## IBM WebSphere MQ パフォーマンス・カウンタ

次の表に、使用可能な IBM WebSphere MQ モニタの測定項目を示します。

### キュー・パフォーマンス・カウンタ

| 測定値                                                            | 説明                                                                      |
|----------------------------------------------------------------|-------------------------------------------------------------------------|
| <b>Event - Queue Depth High (events per second)</b>            | キュー・デプスが設定された最大デプスに達したときに発行されるイベント。                                     |
| <b>Event - Queue Depth Low (events per second)</b>             | キュー・デプスが設定された最小デプスに達したときに発行されるイベント。                                     |
| <b>Event - Queue Full (events per second)</b>                  | キューがいっぱいのときにメッセージをキューに置こうとしたときに発行されるイベント。                               |
| <b>Event - Queue Service Interval High (events per second)</b> | タイムアウトしきい値以内にメッセージがキューにまったく置かれなかったか、またはキューからまったく取得されなかったときに発行されるイベント。   |
| <b>Event - Queue Service Interval OK (events per second)</b>   | タイムアウトしきい値以内にメッセージがキューに置かれたか、またはキューから取得されたときに発行されるイベント。                 |
| <b>Status - Current Depth</b>                                  | ローカル・キューにあるメッセージの現在の数。この測定項目は、監視されているキュー・マネージャのローカル・キューにのみ適用されます。       |
| <b>Status - Open Input Count</b>                               | 開いている入力ハンドルの現在の数。入力ハンドルは、アプリケーションがメッセージをキューに置ける (put) ようにするために開かれます。    |
| <b>Status - Open Output Count</b>                              | 開いている出力ハンドルの現在の数。出力ハンドルは、アプリケーションがメッセージをキューから取得 (get) できるようにするために開かれます。 |

## チャンネル・パフォーマンス・カウンタ

| 測定値                                                        | 説明                                                                                                                          |
|------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| <b>Event - Channel Activated (events per second)</b>       | アクティブになるまで待機しているもののキュー・マネージャのチャンネル・スロットが不足しているためにアクティブになることが禁止されているチャンネルが、突然チャンネル・スロットが使用できるようになったためにアクティブになったときに生成されるイベント。 |
| <b>Event - Channel Not Activated (events per second)</b>   | キュー・マネージャのチャンネル・スロットが不足しているためにアクティブになることが禁止されているチャンネルが、アクティブになろうとしているときに生成されるイベント。                                          |
| <b>Event - Channel Started (events per second)</b>         | チャンネルの開始時に生成されるイベント。                                                                                                        |
| <b>Event - Channel Stopped (events per second)</b>         | チャンネルの停止時に生成されるイベント（停止原因とは無関係）。                                                                                             |
| <b>Event - Channel Stopped by User (events per second)</b> | チャンネルがユーザによって停止されたときに生成されるイベント。                                                                                             |
| <b>Status - Channel State</b>                              | チャンネルの現在の状態。チャンネルは STOPPED（アクティブでない状態）から RUNNING（完全にアクティブな状態）にいたるまでいくつかの状態を経過します。チャンネル状態の範囲は 0（STOPPED）から 6（RUNNING）までです。   |
| <b>Status - Messages Transferred</b>                       | チャンネルを経由して送信されたメッセージの数。チャンネルを経由して発生しているトラフィックがなければ、この測定項目は 0 になります。キュー・マネージャが起動してからチャンネルがまだ開始されていない場合、この測定項目は使用できません。       |
| <b>Status - Buffer Received</b>                            | チャンネルを経由して受信されたバッファの数。チャンネルを経由して発生しているトラフィックがなければ、この測定項目は 0 になります。キュー・マネージャが起動してからチャンネルがまだ開始されていない場合、この測定項目は使用できません。        |

| 測定値                            | 説明                                                                                                                 |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------|
| <b>Status - Buffer Sent</b>    | チャンネルを経由して送信されたバッファの数。チャンネルを経由して発生しているトラフィックがなければ、この測定項目は0になります。キュー・マネージャが起動してからチャンネルがまだ開始されていない場合、この測定項目は使用できません。 |
| <b>Status - Bytes Received</b> | チャンネルを経由して受信されたバイト数。チャンネルを経由して発生しているトラフィックがなければ、この測定項目は0と表示されます。キュー・マネージャが起動してからチャンネルがまだ開始されていない場合、この測定項目は使用できません。 |
| <b>Status - Bytes Sent</b>     | チャンネルを経由して送信されたバイト数。チャンネルを経由して発生しているトラフィックがなければ、この測定項目は0と表示されます。キュー・マネージャが起動してからチャンネルがまだ開始されていない場合、この測定項目は使用できません。 |



# 第 32 章

## オンライン・モニタのトラブルシューティング

LoadRunner モニタを使って、実行時のシナリオのパフォーマンスを表示できます。

以降の項では、オンライン・モニタに関連するヒントと既知の問題について説明します。

- ▶ サーバ・リソース・モニタのトラブルシューティング
- ▶ ネットワーク遅延モニタのトラブルシューティング
- ▶ ネットワークについて考慮すべき事項

### サーバ・リソース・モニタのトラブルシューティング

サーバ・マシンのリソースを監視するには、そのマシンに接続しなければなりません。LoadRunner が指定のサーバを見つけられず、監視が正常に行われない場合、そのサーバが利用可能かどうか確認します。コントローラ・マシンのコマンド・ラインに `ping <サーバ名>` と入力して、`ping` を実行します。

マシンにアクセスできることを確認したら、モニタのトラブルシューティングに関する追加情報がないか、この表で確認してください。

| 問題                                         | 解決策                                                                                                                                                                            |
|--------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 異なるドメインの Windows マシンを監視できない、またはアクセスが拒否される。 | リモート・マシンの管理者権限を取得するには、コマンド・プロンプトで次のコマンドを入力します。<br><code>%net use ¥ &lt;マシン名&gt; ¥user:[ &lt;ドメイン&gt; ¥ &lt;リモート・マシン・ユーザ名&gt; ]</code><br>パスワードの入力を求められたら、リモート・マシンのパスワードを入力します。 |

| 問題                                                                                                                 | 解決策                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|--------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Windows NT/2000 マシンを監視できない (次のようなエラーメッセージが表示される: 「computer_name not found」または「Cannot connect to the host」)。</p> | <p>監視対象の Windows NT/2000 マシンは、管理者権限を持つユーザだけが監視できる設定になっています。管理者以外のユーザが監視できるようにするには、特定のファイルとレジストリ・エントリに対する読み取り権限を許可しなければなりません (Microsoft tech-note number Q158438)。手順は次のとおりです。</p> <p><b>a.</b> エクスプローラまたはファイル・マネージャを使って、次への読み取りアクセス権限をユーザに付与します。</p> <pre>%windir%\system32\PERFCxx.DAT %windir%\system32\PERFHxx.DAT</pre> <p>xxx には、システムの基本言語の ID が入ります (英語の場合は 009 など)。これらのファイルは、なくなっていたり、壊れていたりする可能性があります。そのような場合は、これらのファイルをインストール CD から展開します。</p> <p><b>b.</b> REGEDT32 を使って、以下に対する読み取り権限をユーザに付与します。</p> <pre>HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Perflib</pre> <p>およびそのキーのすべてのサブキー。</p> <p><b>c.</b> REGEDT32 を使って、以下に対する読み取り以上の権限をユーザに付与します。</p> <pre>HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\SecurePipeServers\winreg</pre> |
| <p>Windows 2000 の一部のカウンタを NT マシンから監視できない。</p>                                                                      | <p>Windows 2000 マシンでコントローラを実行します。</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <p>Windows の一部の標準設定のカウンタでエラーが発生している。</p>                                                                           | <p>問題が生じているカウンタを削除して、[測定値の追加] ダイアログ・ボックスを使って適切なカウンタを追加します。</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <p>監視対象のマシンで SQL Server (バージョン 6.5) のパフォーマンス・カウンタを取得できない。</p>                                                      | <p>SQL Server 6.5 に不具合があります。この不具合を回避するために、監視対象マシンの次のレジストリ・キーに対する読み取り権限を与えます (regedt32 を使用)。</p> <pre>HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\MSSQLServer</pre> <p>(Microsoft tech-note number Q170394)</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

| 問題                                                   | 解決策                                                                                                                                                                                                        |
|------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 選択されている測定値がグラフに表示されない。                               | 表示ファイルと <code>online.exe</code> が登録されていることを確認します。完全インストールを行わずにモニタの DLL を登録するには、 <code>&lt; LoadRunner のインストール先フォルダ &gt; %bin</code> にある <code>set_mon.bat</code> バッチ・ファイルを実行します。                           |
| Windows マシンを監視しているときに、グラフに測定値が表示されない。                | Windows に組み込まれているパフォーマンス・モニタを確認します。パフォーマンス・モニタが機能していない場合は、通信の設定に問題がある可能性があります。                                                                                                                             |
| UNIX マシンを監視しているときに、グラフに測定値が表示されない。                   | UNIX マシンで <code>rstatd</code> が実行されていることを確認します（第 20 章「システム・リソースの監視」を参照してください）。                                                                                                                             |
| 次の Web サーバのいずれかを監視できない。MS IIS, MS ASP または ColdFusion | 上記の問題（「Windows マシンを監視できない」）を参照してください。                                                                                                                                                                      |
| WebLogic (JMX) サーバを監視できない。                           | <code>&lt; LoadRunner のインストール先フォルダ &gt;</code><br><code>%dat%monitors%WebLogicMon.ini</code> ファイルを開き、次を検索します。[WebLogicMonitor]JVM=javaw.exejavaw.exe を <code>java.exe</code> に変更します。トレース情報が含まれるウィンドウが開きます。 |

## ネットワーク遅延モニタのトラブルシューティング

LoadRunner が送信元または送信先マシンを見つけられず、監視が正常に行われなかった場合、マシンから指定のマシンが利用できるかどうかを確認します。「ping」を実行します。コマンド・ラインのプロンプトで、次を入力します。

```
ping server_name
```

ネットワーク・パス全体を確認するには、トレースルート・ユーティリティを使ってパスが有効であることを検証します。

Windows の場合、`tracert <サーバ名>` と入力します。

UNIX の場合、`traceroute <サーバ名>` と入力します。

マシンにアクセスでき、ネットワーク・パスも有効であるにも関わらず、監視が正常に行われない場合、次の手順に従います。

1) TCP プロトコルを使用している場合、送信元マシンで **< LoadRunner のインストール先フォルダ > %bin%\webtrace.exe** を実行して、問題がコントローラと、ネットワーク遅延モニタの基盤になる WebTrace テクノロジーのどちらに關係しているかを判断します。UDP または ICMP プロトコルを使用している場合、これらのプロトコルは WebTrace テクノロジーを基盤にしていなため、問題は WebTrace ではなくコントローラに關係しています。

2) **webtrace.exe** の実行結果が返されたら、問題はコントローラに關係しています。送信元マシンが UNIX マシンでないことを確認して、マーキュリー・インタラクティブのカスタマー・サポートに次の情報を報告してください。

- ▶ コントローラ・マシンの temp ディレクトリにあるコントローラ・ログ・ファイル **drv\_log.txt**
- ▶ 送信元のマシンにある **traceroute\_server** ログ・ファイル。LoadRunner 7.02 では、この情報は **< LoadRunner のインストール・フォルダ > %dat%\最新の mdrv log** にありました。LoadRunner 7.5 では、この情報は **< LoadRunner のインストール先フォルダ > folder%\bin%\traceroute\_server.log** にあります。
- ▶ path ディレクトリの **TRS\_debug.txt** および **WT\_debug.txt** ファイルにあるデバッグ情報。これらのファイルは、**< LoadRunner のインストール先フォルダ > %dat%\mdrv.dat** ファイルの [monitors\_server] セクションに次の行を追加し、ネットワーク・モニタを再起動すると作成されます。  
ExtCmdLine=-traceroute\_debug path



3) **webtrace.exe** の実行結果が返されなかった場合、問題は、ネットワーク遅延モニタの基盤になっている **WebTrace** テクノロジーに関係しています。送信元マシンで次の手順に実行します。

- ▶ `WINNT\system32\drivers` ディレクトリに **packet.sys** ファイル (Webtrace ドライバ) があることを確認します。
- ▶ ドライバ (「Cloud」や「Sniffer」など) がネットワーク・カード・ドライバの上からインストールされていないかどうかを確認します。インストールされている場合は、そのドライバを削除して再度 **WebTrace** を実行します。
- ▶ 対象マシンに対する管理者権限を持っていることを確認します。
- ▶ `ipconfig /all` を使って、ネットワーク・カードに IP アドレスが 1 つだけ割り当てられていることを確認します。1 つのカードに複数の IP アドレスが割り当てられていると (IP スプーフィング), **WebTrace** はこれらのアドレスを処理できません。
- ▶ インストールされているネットワーク・カードの数を調べます。`webtrace -devlist` を実行して、使用可能なネットワーク・カードの一覧を取得します。
- ▶ リストに複数のカードがある場合、`webtrace -dev <デバイス名> <送信先>` を実行します。<デバイス名>には、リストに表示されたネットワーク・カードの名前を 1 つ指定します。**WebTrace** が間違ったカードにバインドされていることがわかった場合には、`webtrace set_device <デバイス名>` を使ってレジストリ・キーを設定し、標準設定のカードの代わりに指定のカードを使用するように **WebTrace** に指示します。
- ▶ ネットワーク・カードが Ethernet に対応していることを確認します。
- ▶ マーキュリー・インタラクティブのカスタマー・サポートに、`webtrace.exe -debug` (たとえば、`webtrace.exe -debug www.merc-int.com`) と、マシンで `ipconfig /all` を実行したときの出力を報告してください。

## ネットワークについて考慮すべき事項

ネットワーク上の異常な遅延に気付いたら、以下の項を参照してパフォーマンスの向上を図ります。

- ▶ ネットワークの帯域幅の使用率
- ▶ Ethernet バス接続ネットワーク
- ▶ WAN または負荷の高い LAN での作業

### ネットワークの帯域幅の使用率

ほとんどの負荷テストのシナリオでは、ネットワーク・カードはパフォーマンスにほとんど影響しません。ネットワーク・カードは、物理ネットワーク層の帯域幅を処理できるように作られています。パケットは、IEEE 803.x の標準規格に従う速度で Ethernet 上を伝送されます。ネットワークがボトルネックになる場合、問題はネットワーク・カードのブランドではなく、物理層 (Ethernet, FDDI, ATM, Ethernet Token-Ring など) の帯域幅の制限です。

つまり、T10 回線上ではなく、回線を DS3 (45 Mbps) または T100 (100 Mbps) にアップグレードしてから負荷テストを行います。

ネットワークのアップグレードが必要かどうかを判断するときに役立ついくつかのヒントを以下に示します。

- 1) 仮想ユーザ・ロード・ジェネレータ上でパフォーマンス・モニタを実行します。仮想ユーザの数を増やしながらネットワークのバイト転送率を検査して、飽和点に達しているかどうかを確認します。飽和点に達している場合、さらに多くの仮想ユーザを実行するには、ネットワークをアップグレードします。ネットワークをアップグレードせずに実行すると、仮想ユーザのパフォーマンスが低下します。ネットワーク環境のパフォーマンスは指数的に低下します。
- 2) サーバ・マシン上でパフォーマンス・モニタを実行します。いくつかのロード・ジェネレータ・マシン上で多数の仮想ユーザを実行します。カーネルの使用状況とネットワークの転送率を見て、飽和点に達しているかどうかを確認します。必要なユーザ負荷より少ない仮想ユーザ数で飽和点に達した場合には、ネットワークをアップグレードします。
- 3) どのネットワークにも、ネットワーク管理者によって設定された固有の MTU (Maximum Transmission Unit) があります。MTU とは、ネットワークで送信できる最大物理パケット・サイズ (バイト単位) です。MTU より大きいメッセージは、送信前に複数の小さなパケットに分割されます。

クライアントとサーバが大きなデータ・セットをやり取りしている場合、帯域幅の使用率を向上させるために MTU を大きくするようにネットワーク管理者に指示します。マシンの MTU は、マシンとメッセージの最終的な送信先の間にあるすべてのネットワークの中で最小の MTU と同じ大きさにするのが理想的です。

どれかの MTU より大きいメッセージを送信すると、そのメッセージは断片化され、送信速度が低下します。MTU が大きすぎると、意図しない劣化が生じる可能性があります。試行錯誤を繰り返すことが、最適な MTU を見つける唯一の確実な方法ですが、いくつかの有用なガイドラインがあります。たとえば、ほとんどの Ethernet ネットワークの MTU は 1500 に設定されています。

設定した MTU によってパフォーマンスが低下する場合は、ネットワークをアップグレードするか MTU を小さくしてパフォーマンスを向上させます。

### Ethernet バス接続ネットワーク

次のガイドラインは、Ethernet バス接続ネットワークに適用されます。

通信しているアクティブなマシンが 2 台だけあるネットワークでは、帯域幅の最大使用率は 90% になります。

通信しているアクティブなマシンが 3 台あるネットワークでは、帯域幅の最大使用率は約 85% になります。

ネットワーク上のアクティブなマシンの数が増えるにつれ、全体的な帯域幅の使用率は減ります。

### WAN または負荷の高い LAN での作業

WAN または負荷の高い LAN で LoadRunner を使用すると、ネットワークの問題を示す LoadRunner の異常な振る舞いに気付くことがあります。出力ウィンドウには、再送、パケット喪失、メッセージの不一致に関するメッセージが表示される場合があります。これは、コントローラからのメッセージの一部が LoadRunner エージェントに届いていないためです。この問題を解決するには、ネットワーク・トラフィックを減らすか、ネットワークの帯域幅を大きくします。

次の手順で、ネットワーク・トラフィックを削減できることがあります。

- ▶ [実行環境の設定] ボタンをクリックし、[一般] ノードの下の [ログ] ノードを選択します。[ログを有効にする] チェック・ボックスの選択を解除します。
- ▶ すべてのユーザを実行前に初期化します。初期化が完了してからユーザを実行します。



# 第 5 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。

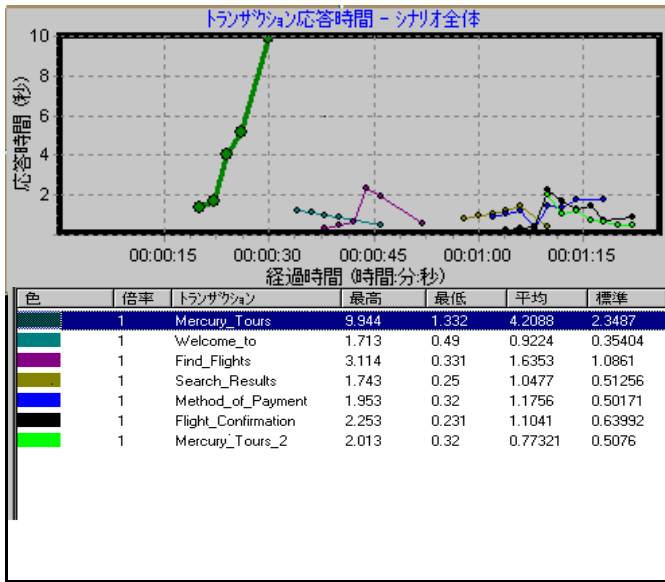
### オンライン・モニタ・グラフ

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？

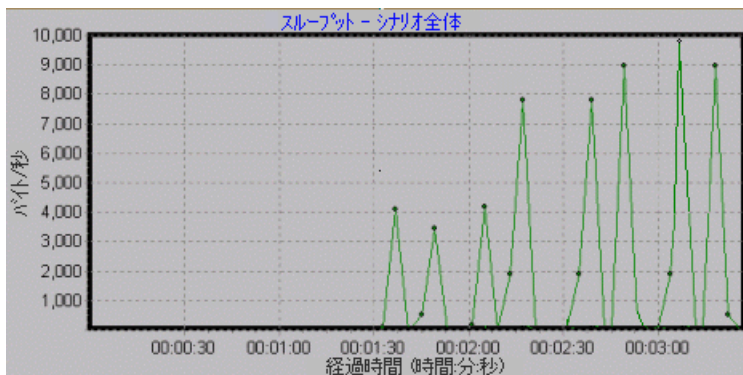
**回答:** [トランザクション応答時間] グラフには、各トランザクションが完了するのに要した時間が表示されます。以下のグラフは、ログイン・トランザクション以外のトランザクション応答時間が速いことを示しています。最初のログインでは、長い時間かかっていませんが、次のログインから極端に遅くなっていることを示します。これは、データベースが同時に2つ以上のログインを

処理できず、その原因はおそらくデータベースのクエリー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。

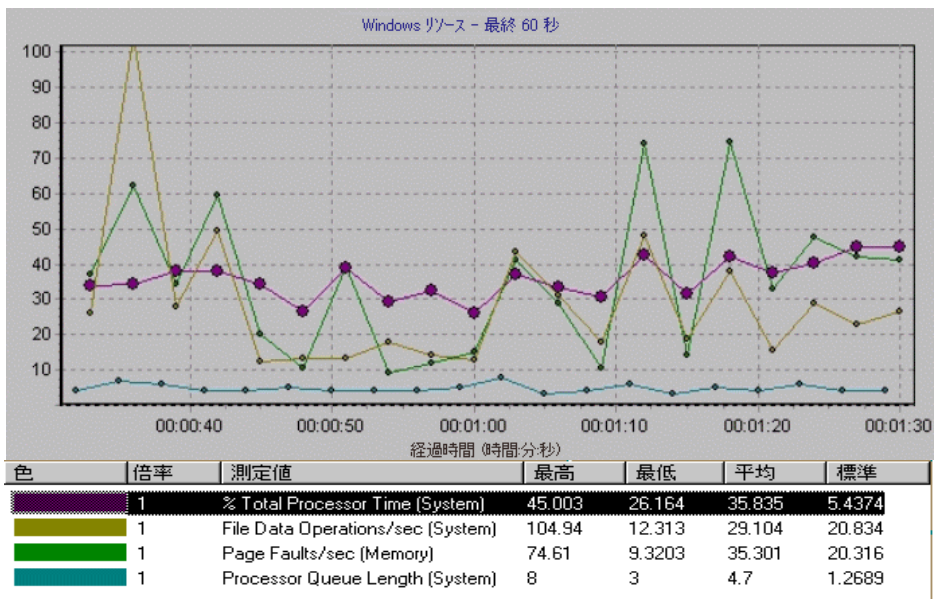




上のグラフでは、スループットが時間の経過とユーザ数の増加に伴って増えており、帯域幅が十分なことを示しています。ユーザ数が増加してもグラフに変化があまり見られない場合は、要求しているデータの量に対して、帯域幅によって制限が課せられていると考えられます。

**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

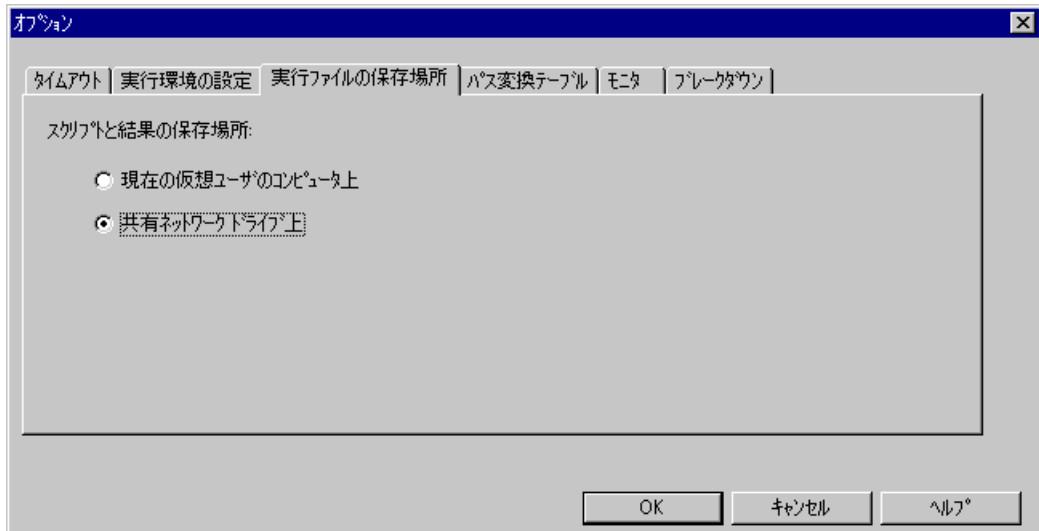
シナリオを実行すると、LoadRunner は参加している仮想ユーザから実行時のデータを収集します。標準では、LoadRunner は、各仮想ユーザ・マシンの一時ファイル内にデータを格納します。シナリオの実行後、データは総合的な結果ディレクトリで照合されます。

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。

[設計] ビューのシナリオ・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。

たとえば、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。たとえば、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。

|                 |                                        |
|-----------------|----------------------------------------|
| <b>hostname</b> | コントローラを実行しているマシンの名前                    |
| <b>win</b>      | コントローラを Windows マシンで実行している             |
| <b>unix</b>     | コントローラを UNIX マシンで実行している                |
| <b>all</b>      | 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。たとえば、ディレクトリ **scripts** がネットワーク・ドライブ **r**（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。たとえば、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**

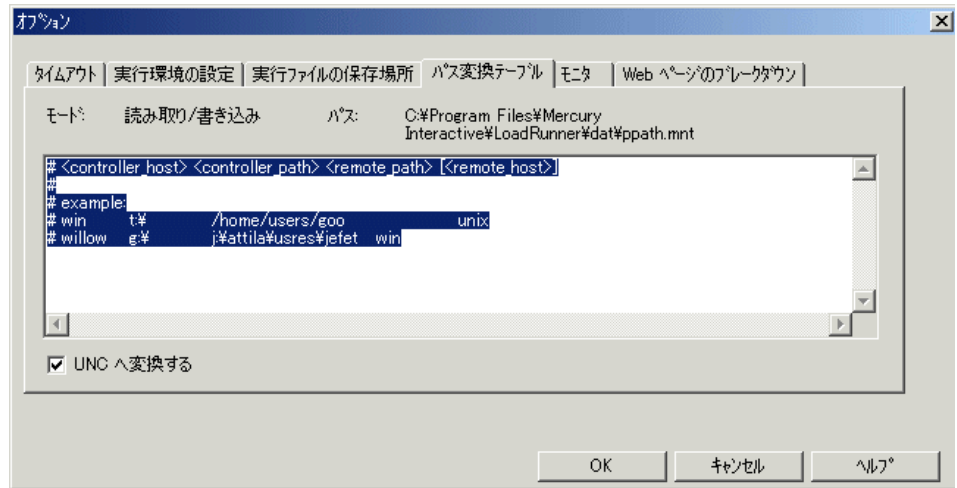
リモート・ロード・ジェネレータの名前または種類。たとえば、すべてのリモートマシンが UNIX ワークステーションの場合、**remote\_host** フィールドに **unix** と指定します。**remote\_host** フィールドのオプションは、上記の **controller\_host** フィールドのオプションと同じです。**remote\_host** パラメータは省略可能です。

## 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。

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:\oadtest\ | 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。

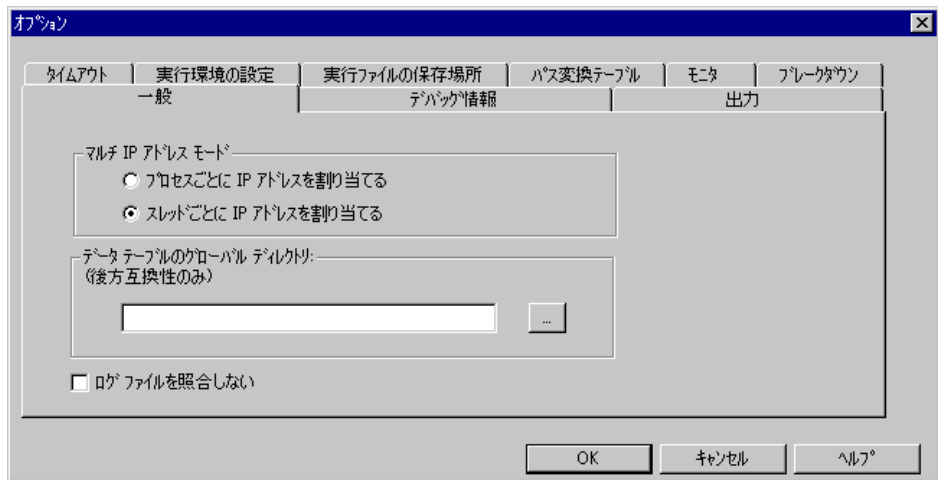
エキスパート・モードを終了する場合も、同じ手順を繰り返します。

## 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] をチェックします。
- 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 – 一般設定】 タブについて

[一般設定] 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。を使用している場合にのみ表示されます。

[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。

[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 ファイルを参照しない]：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] チェック・ボックスを選択します。
- 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。を使用している場合にのみ表示されます。

[**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。たとえば、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。

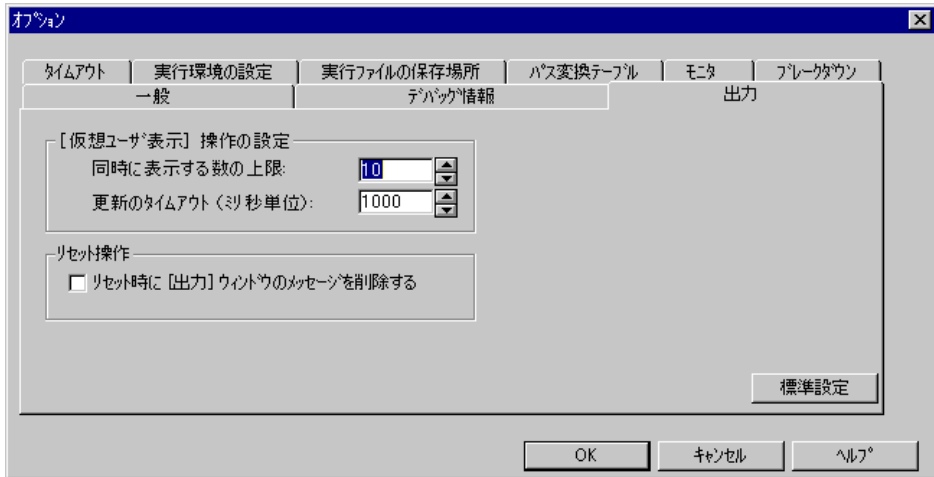
[**一時ファイルを保存する**] : エージェントとコントローラは複数の一時ファイルを作成します。一時ファイルには、仮想ユーザに送信されるパラメータ・ファイル、出力コンパイル・ファイル、構成設定ファイルなどの情報が含まれます。エージェント・ファイルは、エージェント・マシンの **TMP** ディレクトリまたは **TEMP** ディレクトリ内にある **br** フォルダに格納されています。コントローラ・ファイルは、コントローラ・マシンの **TMP** ディレクトリまたは **TEMP** ディレクトリ内にある **lrr** フォルダに格納されています。シナリオの終了時に、これらのファイルはすべて自動的に削除されます。[**一時ファイルを保存する**] 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、仮想ユーザー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ー・ログの最大数を指定します。
- 3 [更新のタイムアウト] ボックスに、LoadRunner がユーザー・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。を使用している場合にのみ表示されます。

[[仮想ユーザの表示] 操作の設定] :

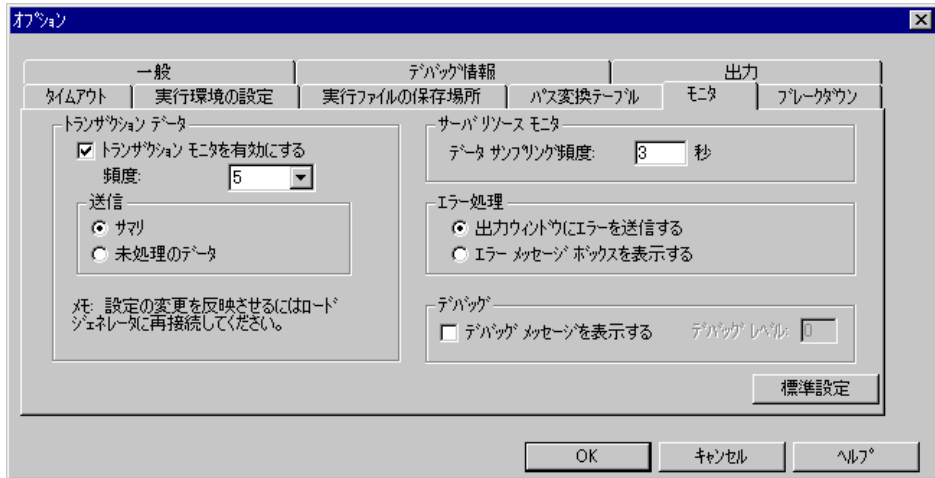
- ▶ [同時に表示する数の上限] : 同時に表示できる仮想ユーザ・ログの最大数と、マシンで Run-Time Viewer を開いてコントローラが表示するアクティブな UNIX, GUI, RTE または Web の仮想ユーザの最大数を指定します。標準設定の値は 10 です。
- ▶ [更新のタイムアウト (ミリ秒)] : 仮想ユーザ・ログを更新する頻度を指定します。標準設定は 1000 ミリ秒ごとです。

[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・ジェネレータの情報

名前:

プラットフォーム:

一時ディレクトリ:

ロード・ジェネレータのシナリオ参加を有効にする

| ファイアウォール | WAN エミュレーション | 端末サービス   |
|----------|--------------|----------|
| ステータス    | 実行ファイルの保存場所  | UNIX 環境  |
|          |              | 実行時コマンド  |
|          |              | 仮想ユーザの制限 |

ログイン

名前:

ログイン名に小文字を使用する

ローカルユーザ:

シェル環境設定

RSH を使用しない

標準設定のシェル:

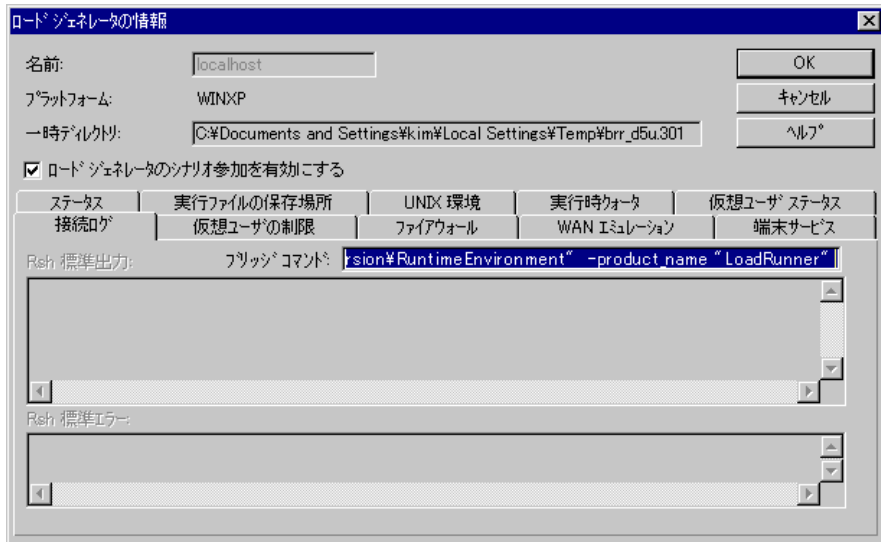
初期化コマンド:

※ この値は仮想ユーザを実行中、またはファイアウォール越しでの監視中のホストには無効です。

標準設定

## 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。

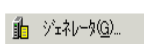


[Rsh 標準出力] : コントローラ が選択された UNIX ロード・ジェネレータに接続したときの rsh の標準出力を表示します。

[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、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 の標準出力を表示します。

[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。

- ▶ 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

## 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・スクリプトとしてすべてのリモート・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。

---

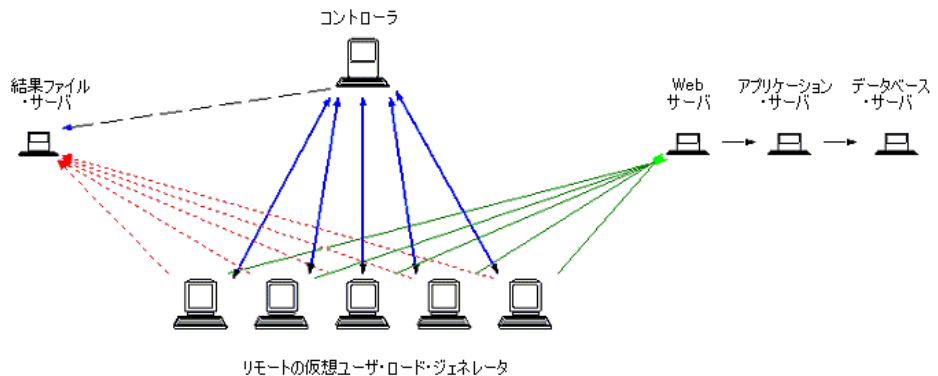
スタンドアロン・モードで仮想ユーザを実行する方法の詳細については、仮想ユーザ・スクリプトの作成について書かれたマニュアルを参照してください。

## 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間の通信は双方向（送信と受信）ですが、仮想ユーザとファイル・サーバとの間の通信は片方向（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。

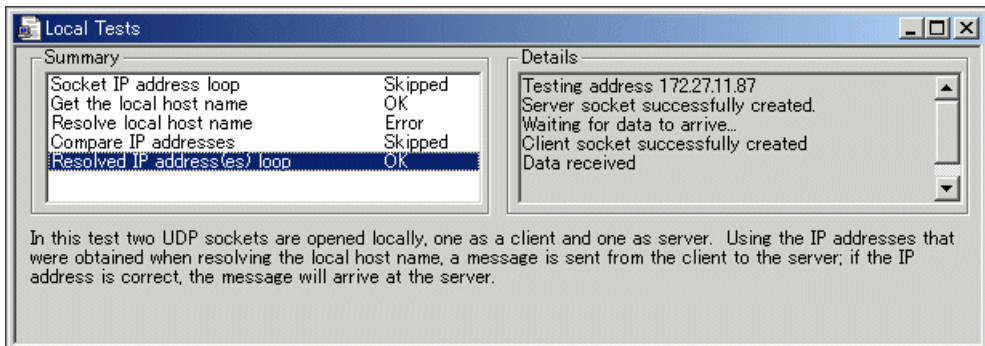
## ロード・ジェネレータとの通信の失敗

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。以下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。



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。たとえば、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、次の手順で行います。

- 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_agent_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、手順 3 で説明した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。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。

- ▶ 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は、次のメッセージが表示されます。  
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: MWAR-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] オプションをチェックします。この後、通常どおり接続を行います。
- 3 エージェント・デーモンを終了するには、< 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 が終了します。正常に終了すると、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、以下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。以下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。たとえば、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。結果ディレクトリ・ファイルのパーミッションを確認し、必要があれば変更します。

## 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 13 章「シナリオの実行」を参照してください。

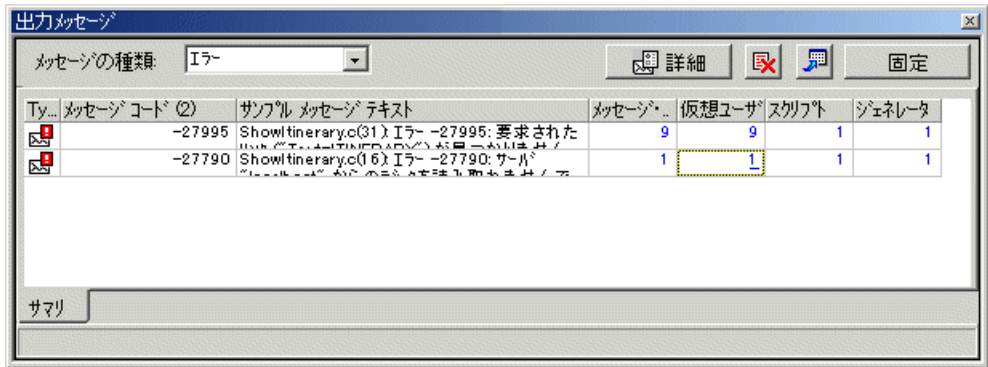
The screenshot shows the '仮想ユーザ(20)' window in LoadRunner. The table displays the status of 20 virtual users. The columns are ID, ステータス (Status), スクリプト (Script), トランザクション (Transaction), and 経過時間 (Elapsed Time). The status for users 1, 3, 7, and 10 is '完了-失敗' (Completed - Failed), while users 2, 4, 5, 6, 8, and 9 are '実行中' (Running). All transactions are on 'localhost' and the script is 'test\_10'. The elapsed times range from 00:00:07 to 00:00:11. On the right side, there are control buttons: 実行(R) (Execute), 徐々に停止(S) (Stop Gradually), 停止(S) (Stop), リセット(E) (Reset), ヘルプ(H) (Help), and 閉じる(C) (Close).

| ID | ステータス                  | スクリプト   | トランザクション  | 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。バッチ・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 14 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、『**LoadRunner 仮想ユーザ・スクリプトの作成**』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。たとえば、シナリオ・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。

The screenshot displays the 'Analysis Summary' window in LoadRunner. It provides a detailed overview of the test execution, including session information, overall statistics, and a breakdown of transactions.

**アナリシス サマリ** 期間: 04/12/2002 17:23:45 - 04/12/2002 17:23:45

シナリオ名: C:\unzipped\LR75-TrainingSetup-01b\web\_project\solutions\\_Lab7.lrs  
 セッションの結果: H:\web\_project\solutions\lab7\_25users\lab7\_25users.lrr  
 継続時間: 1分、43秒

**統計サマリ**

- 最大実行仮想ユーザ数: 25
- 合計スループット (バイト): 20,033,487
- 平均スループット (バイト/秒): 192,630
- 合計ヒット数: 5,800
- 秒ごとの平均ヒット数: 55.769 [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: 成功の合計: 250 失敗の合計: 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 |
|----------------------------------------|--------|--------|--------|-------|----------|-----|----|
| <a href="#">BookFlight_Transaction</a> | 20.289 | 21.762 | 38.966 | 3.07  | 22.776   | 100 | 0  |
| <a href="#">purchase_flight</a>        | 6.039  | 6.089  | 6.229  | 0.044 | 6.13     | 100 | 0  |

グラフとレポートの詳細については、『**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・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。
- ▶ ログ・レベルを上げるには、[スタート] > [プログラム] > [LoadRunner] > [Advanced Settings] > [Agent Settings] を選択し (または、< LoadRunner のインストール先フォルダ > ¥launch\_service¥dat¥br\_lrch\_server.cfg ファイルをテキスト・エディタで開きます)、[Log] セクションで AgentExtended を 1 に設定します。

**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。

ログ・レベルを上げるには、[スタート] > [プログラム] > [LoadRunner] > [Advanced Settings] > [Agent Settings] を選択するか、テキスト・エディタで < LoadRunner のインストール先フォルダ > ¥launch\_service¥dat¥br\_inch\_server.cfg ファイルを開きます。[Log] セクションで、AgentExtended を 1 に設定します。

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、以下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 15 章「ファイアウォールに関する作業」で説明したとおり、[エージェントの設定] 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 15 章「ファイアウォールに関する作業」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なりません。

**Error -10343 : Communication error: -59981 : Failed to connect to remote host - <MI\_Listener\_name > .**

[エージェントの設定] 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 .**

[エージェントの設定] 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 .**

[エージェントの設定] の [プロキシ名] 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 アドレスではありません。

**Error -10343 : Communication error: -59945 : Client failed to connect to a PROXY Server with the following settings:(-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).**

[プロキシ名] 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= < proxy\_server\_name > )(-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).**

[エージェントの設定] 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、[エージェントの設定] 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、[エージェントの設定] 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する] 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 H「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 : sslv3 alert handshake failure.**

**Error -10343 : Communication error: -59907 : SSL connect error : sslv3 alert bad certificate.**

**Error -10343 : Communication error: -59907 : SSL connect error : sslv3 alert certificate expired.**

これらのエラーは、[サーバ証明書を確認する] 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 H「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、[エージェントの設定] で [クライアント証明書の所有者] 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 H 「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.**

これらのエラーは、[サーバ証明書を確認する] 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.**

これらのエラーは、[サーバ証明書を確認する] 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 H 「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.**

これらのエラーは、[MI リスナーの設定] で [クライアント証明書の確認] 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 [クライアント証明書の所有者] 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.**

これらのエラーは、[MI リスナーの設定] で [クライアント証明書の確認] 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 [クライアント証明書の所有者] 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 H「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.**

これらのエラーは、[MI リスナーの設定] で [クライアント認証の確認] が [True] に設定されているのに対し、[エージェントの設定] で [クライアント認証の所有者] 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。

### 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 >** `¥data¥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 [エージェントの設定] 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 [エージェントの設定] 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)。



# 付録 E

---

## サーバ・モニタ・カウンタでの作業

システム・リソース、Microsoft IIS、Microsoft ASP、ColdFusion および SQL サーバ・モニタを設定すると、監視対象のサーバで測定できる標準設定のカウンタのリストが表示されます。次で説明する手順に従って、カウンタを追加したり、既存のカウンタを削除することによって、新しい標準設定のカウンタのリストを作成できます。

さらに、サーバでの最初の負荷テストで、サーバのパフォーマンスを調べたり、ボトルネックの原因を特定したりするのに特に便利なカウンタがあります。

この付録では、以下の項目について説明します。

- ▶ モニタの標準設定のカウンタの変更
- ▶ 負荷テストに役立つカウンタ

### モニタの標準設定のカウンタの変更

< LoadRunner のインストール先フォルダ > \dat ディレクトリにある **res\_mon.dft** ファイルを編集して、システム・リソース、Microsoft IIS、Microsoft ASP または SQL サーバ・モニタの標準設定のカウンタを変更できます。

**標準設定のカウンタを変更するには、次の手順で行います。**

- 1 新しいシナリオを開いて、[実行] タブをクリックします。
- 2 モニタごとに、測定対象のカウンタを選択します。
- 3 シナリオを保存し、そのシナリオの **.jrs** ファイルをエディタで開きます。
- 4 **res\_mon.dft** ファイルに、選択した各カウンタの [MonItemPlus] セクションをコピーします。
- 5 ファイルに追加した新しいカウンタの数を数え、その数で **ListCount** パラメータを更新します。

## 負荷テストに役立つカウンタ

カウンタの中には、サーバでの最初の負荷テストで、サーバのパフォーマンスを調べたり、ボトルネックの原因を特定したりするのに特に便利なカウンタもあります。

Web サーバ・パフォーマンスの監視に役立つカウンタのリストを次に示します。

| オブジェクト              | カウンタ                          |
|---------------------|-------------------------------|
| Web Service         | Maximum Connections           |
| Web Service         | Bytes Total/sec               |
| Web Service         | Current NonAnonymous Users    |
| Web Service         | Current Connections           |
| Web Service         | Not Found Errors              |
| Active Server Pages | Requests/sec                  |
| Active Server Pages | Errors/sec                    |
| Active Server Pages | Requests Rejected             |
| Active Server Pages | Request Not Found             |
| Active Server Pages | Memory Allocated              |
| Active Server Pages | Requests Queued               |
| Active Server Pages | Errors During Script Run Time |
| Memory              | Page Faults/sec               |
| Server              | Total Bytes/sec               |
| Process             | Private Bytes/Inetinfo        |

SQL サーバ・パフォーマンスの監視に役立つカウンタのリストを次に示します。

| オブジェクト          | カウンタ                  |
|-----------------|-----------------------|
| SQLServer       | User Connections      |
| SQLServer       | Cache Hit Ratio       |
| SQLServer       | Net-Network Reads/sec |
| SQLServer       | I/O-Lazy Writes/sec   |
| SQLServer-Locks | Total Blocking Locks  |
| PhysicalDisk    | Disk Queue Length     |

Web および SQL サーバ・パフォーマンスの両方の監視に役立つカウンタのリストを次に示します。

| オブジェクト       | カウンタ                   |
|--------------|------------------------|
| Processor    | % Total Processor Time |
| PhysicalDisk | % Disk Time            |
| Memory       | Available Bytes        |
| Memory       | Pool Nonpaged Bytes    |
| Memory       | Pages/sec              |
| Memory       | Committed Bytes        |
| System       | Total Interrupts/sec   |
| Object       | Threads                |
| Process      | Private Bytes:_Total   |

---

注：「% Disk Time」カウンタについては、コマンド・プロンプトで `diskperf -y` ユーティリティを実行してから、マシンを再起動する必要があります。

---





# 付録 F

---

## 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、以下の項目について説明します。

- ▶ 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 : Java Vuser, Javascript Vuser, VB Vuser, VB Script Vuser
- ▶ E ビジネス : FTP, Palm, SOAP, Web (HTTP/HTML) プロトコル, WinSock/Web Dual プロトコル
- ▶ ERP : Oracle NCA, Siebel-Web
- ▶ メール・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
- ▶ 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、680 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。

以下に、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。

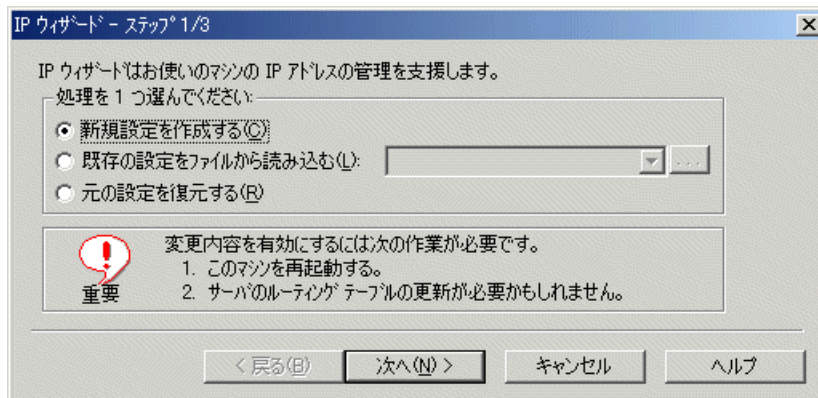
- 1 対象のロード・ジェネレータ・マシン上で IP ウィザードを実行して、指定した数の IP アドレスを追加します。UNIX ロード・ジェネレータ・マシンでは、手作業で新しい IP アドレスを設定します。
- 2 マシンを再起動します。
- 3 必要であれば、サーバのルーティング・テーブルを新しいアドレスで更新します。
- 4 コントローラからこの機能を有効にします。詳細については、682 ページ「コントローラでの複数 IP アドレス機能の有効化」を参照してください。

## 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。

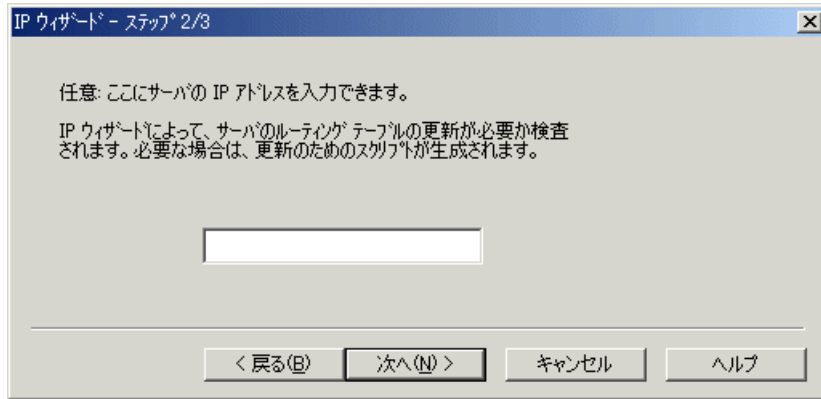
ロード・ジェネレータ・マシンに新しい IP アドレスを追加するには、次の手順で行います。

- 1 LoadRunner プログラム・グループから [IP Wizard] を起動します。

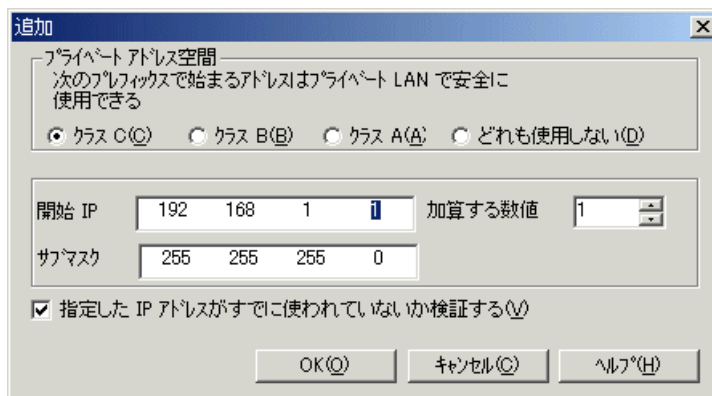


- 2 IP アドレスを設定した既存のファイルがある場合、[既存の設定をファイルから読み込む] を選択してからファイルを選びます。
- 3 新しい設定を定義する場合は、[新規設定を作成する] を選択します。
- 4 [次へ] をクリックして次のステップに進みます。複数のネットワーク・カードがある場合は、指定した IP アドレスを使うカードを選び、[次へ] をクリックします。

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。



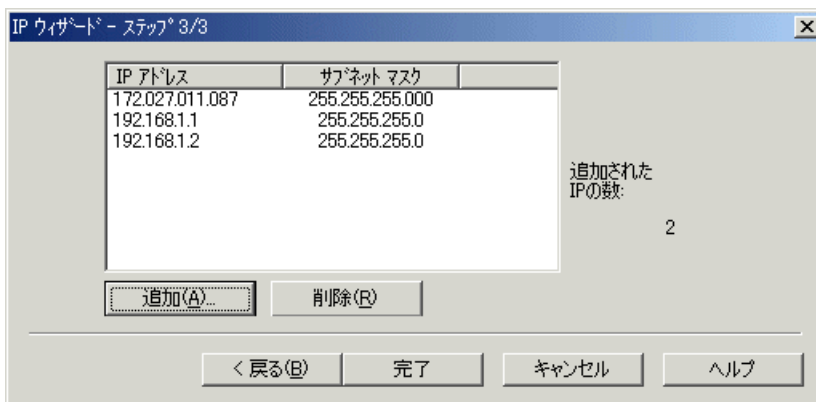
- 5 アドレスの追加直後にサーバのルーティング・テーブルを検査するには、サーバの IP アドレスを入力します。詳細については、681 ページ「ルーティング・テーブルの更新」を参照してください。
- 6 [次へ] をクリックするとマシンの IP アドレスのリストが表示されます。[追加] をクリックしてアドレスの範囲を定義します。



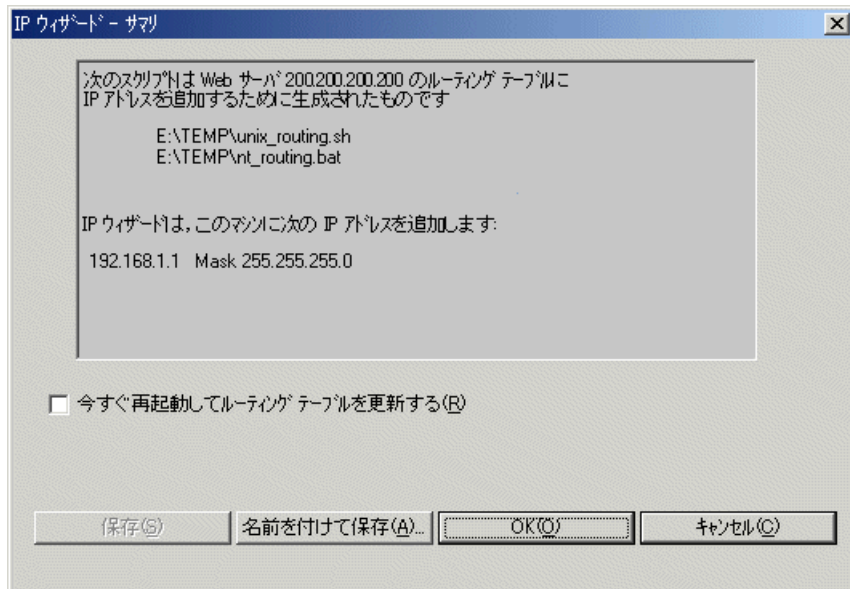
IP アドレスは、**netid** と **hostid** という 2 つの要素で構成されています。アドレスのうちどこまでが **netid** でどこからが **hostid** であるかは、サブマスクによって決まります。

- 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。
- 9 [OK] をクリックして先に進みます。

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・ファイルの使用方法については、681 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

**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
```

- 2 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。**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。

- 3 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。

## 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以下に示します。

```
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 スーパ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 G

---

## 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、以下の項目について説明します。

- ▶ コマンド・ラインからのコントローラの起動
- ▶ TestDirector 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。

たとえば、起動時に TestDirector 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。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。

## TestDirector の引数

TestDirector 引数は、LoadRunner と TestDirector の統合を定義します。LoadRunner と TestDirector の統合に関する詳細については、第 12 章「TestDirector を使ったシナリオ管理」を参照してください。

|                        |                                                                                                               |
|------------------------|---------------------------------------------------------------------------------------------------------------|
| <b>ConnectToTD</b>     | コントローラを起動時に TestDirector に接続するかどうかを指定します (0/1 または ON/OFF)。                                                    |
| <b>TDServer</b>        | TestDirector サーバ名。TestDirector 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。                                                     |
| <b>TDDb</b>            | TestDirector データベース名。次の形式を使用します。<br>" <ドメイン名> . <プロジェクト名> "                                                   |
| <b>UserName</b>        | TestDirector に接続するユーザ名。                                                                                       |
| <b>Password</b>        | ユーザ名に対応するパスワード。                                                                                               |
| <b>TestPath</b>        | TestDirector データベース内のシナリオのパス。次に例を示します。<br>"[TD]¥Subject¥LoadRunner¥Scenario1"<br>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。 |
| <b>TestId</b>          | テスト ID (TestDirector にだけ使用される)。                                                                               |
| <b>ResultCleanName</b> | <b>ResultCycle</b> と併せて使用します。たとえば、「Res1」のように指定します。                                                            |
| <b>ResultCycle</b>     | TestDirector サイクル。たとえば、「LR_60_SP1_247」のように指定します。                                                              |

注： **ResultCycle** 引数と **ResultCleanName** 引数は、TestDirector 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。

|                        |                                                                                                                                                                                                       |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>TestPath</b>        | シナリオのパス。<br>例：<br><b>C:¥LoadRunner¥scenario¥Scenario.lrs</b><br>この引数は、TestDirector 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。<br><b>"[TD]¥Subject¥LoadRunner¥Scenario1"</b><br>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。 |
| <b>Run</b>             | シナリオを実行し、すべての出力メッセージを <b>res_dir¥output.txt</b> に出力し、コントローラを終了します。                                                                                                                                    |
| <b>InvokeAnalysis</b>  | 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。                                                                                                              |
| <b>ResultName</b>      | 結果の完全パス。たとえば、「C:¥Temp¥Res_01」のように指定します。                                                                                                                                                               |
| <b>ResultCleanName</b> | 結果の名前。たとえば、「Res_01」のように指定します。                                                                                                                                                                         |
| <b>ResultLocation</b>  | 結果ディレクトリ。たとえば、「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 H

---

## 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、以下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 15 章「ファイアウォールに関する作業」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 15 章「ファイアウォールに関する作業」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。

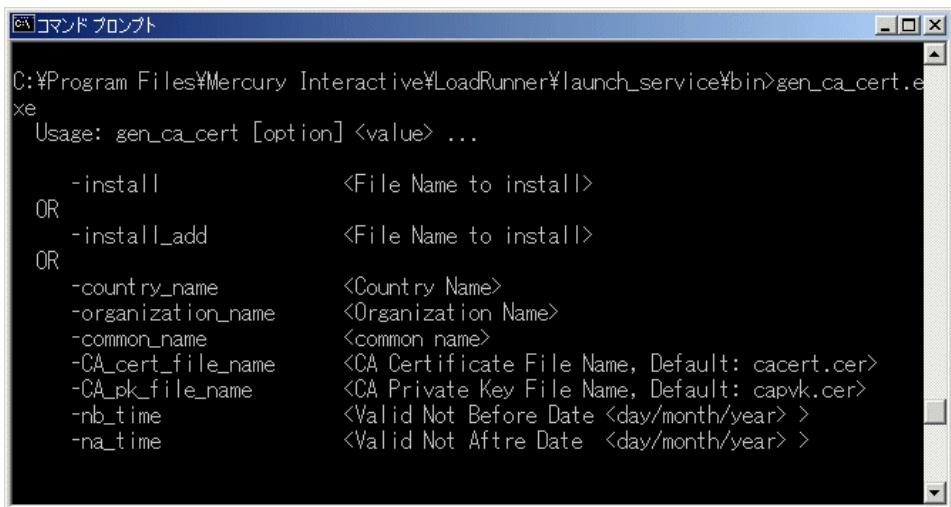
LoadRunner のすべてのインストールに、SSL CA リストと SSL 証明書が含まれています。この証明書は、すべての LoadRunner のインストールで同じです。したがって、第三者が同じ証明書を入手できます。したがって、安全性を高めたい場合は、認証局を独自に作成し、リストに含め、対象とするマシンに対応する証明書を発行します。

## 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。



```
コマンド プロンプト
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.cer>
 -CA_pk_file_name <CA Private Key File Name, Default: capvk.cer>
 -nb_time <Valid Not Before Date <day/month/year> >
 -na_time <Valid Not After Date <day/month/year> >
```



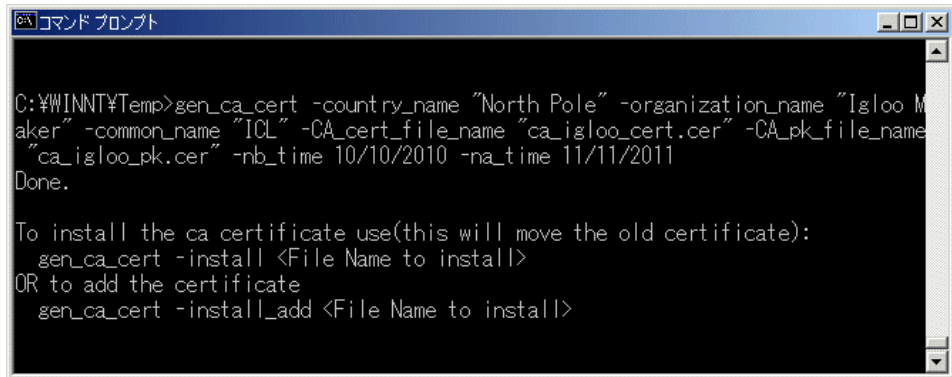
- 2 `gen_ca_cert` コマンドにオプションを少なくとも1つ指定して実行すると、新しい認証局が作成されます：

`-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 (`ca_cert.cer`) と CA 秘密鍵 (`ca_pk.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。

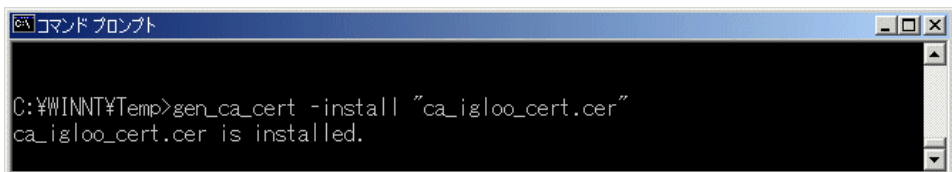


```
C:\WINNT\Temp>gen_ca_cert -country_name "North Pole" -organization_name "Igloo M
aker" -common_name "ICL" -CA_cert_file_name "ca_igloo_cert.cer" -CA_pk_file_name
"ca_igloo_pk.cer" -nb_time 10/10/2010 -na_time 11/11/2011
Done.

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。

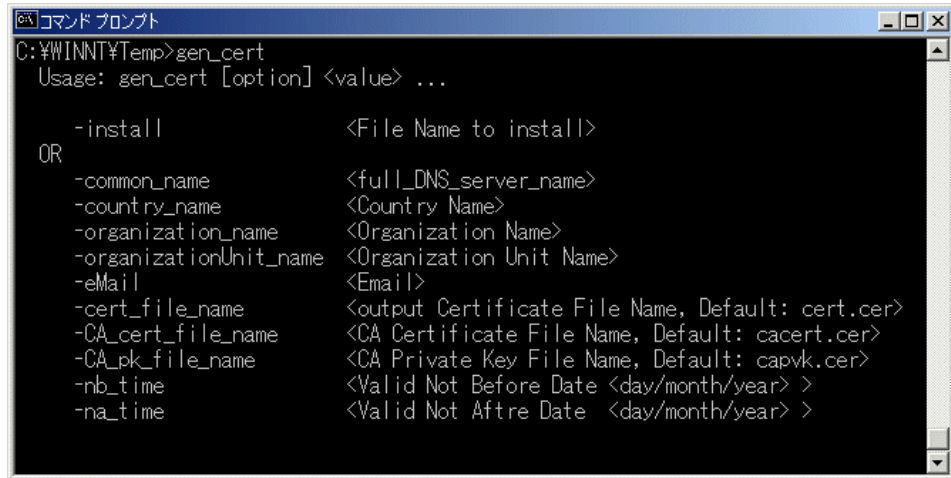


```
C:\WINNT\Temp>gen_ca_cert -install "ca_igloo_cert.cer"
ca_igloo_cert.cer is installed.
```

- 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.cer>
 -CA_cert_file_name <CA Certificate File Name, Default: cacert.cer>
 -CA_pk_file_name <CA Private Key File Name, Default: capvk.cer>
 -nb_time <Valid Not Before Date <day/month/year> >
 -na_time <Valid Not Aftre Date <day/month/year> >

```

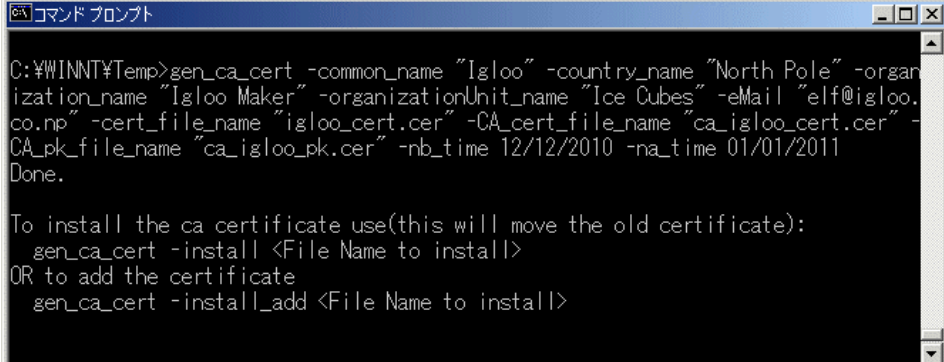
- 2 `gen_ca_cert` コマンドに、次のオプションの少なくとも1つを指定して実行し、新しいデジタル証明書を作成します：
- `-country_name` <国名>、`-organization_name` <組織名>、`-organization_unit_name` <部署名>、`-eMail` <電子メール/アドレス>および`-common_name` <マシンの名前、完全名、または IP アドレス>

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.cer** および **capvk.cer** 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.cer** 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:\$WINDIR\System32>gen_ca_cert -common_name "Igloo" -country_name "North Pole" -organization_name "Igloo Maker" -organizationUnit_name "Ice Cubes" -eMail "elf@igloo.co.np" -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

## A

Acrobat Reader xi  
Antara FlameThrower  
    モニタ 339  
Apache  
    モニタ 372  
Application Deployment ソリューション・モニ  
    タ 583–593  
    Citrix MetaFrame XP モニタ 584  
Ariba  
    モニタ 392  
ASP  
    モニタ 433  
ATG Dynamo  
    モニタ 397

## B

BEA WebLogic  
    モニタ 448  
BroadVision  
    モニタ 402

## C

CA 687  
Check Point FireWall-1  
    モニタ 366  
Citrix MetaFrame XP モニタ 584  
ColdFusion  
    モニタ 411  
controller\_host 629  
controller\_path 630

## D

DB2  
    モニタ 478

## E

## EJB

    モニタ 542  
ERP/CRM サーバ・リソース・モニタ 521–540  
    SAP Portal モニタ 528  
    SAP モニタ 522  
    Siebel Server Manager モニタ 536  
    Siebel モニタ 532  
Ethernet バス接続ネットワーク 619

## F

Fujitsu INTERSTAGE  
    モニタ 414

## G

GUI 仮想ユーザ, 定義 9

## H

hme0 デバイス 680  
hostid, IP アドレス・コンポーネント 677  
Hostinfo ニューティリティ 648  
hosts ファイル 649  
HP での IP アドレスの設定 681  
HTTP  
    秒ごとの応答数グラフ 319

## I

IBM WebSphere MQ モニタ 602  
IIS モニタ 375  
iPlanet/Netscape  
    モニタ 378  
iPlanet (SNMP)  
    モニタ 382  
iPlanet (NAS)  
    ダイアログ・ボックス 423  
    モニタ 417  
IP アドレス  
    hostid 677

## LoadRunner コントローラ・ユーザーズ・ガイド

HP での設定 681  
IP ウィザード 676  
Linux での設定 680  
netid 677  
Solaris での設定 680  
クラス 678  
コントローラからの有効化 682  
サブマスク 677  
複数の設定 673–682  
ロード・ジェネレータへの追加 675  
ロード・ジェネレータ・マシン 673

## J

J2EE  
    モニタ 562  
Java パフォーマンス・モニタ 541–582  
    EJB モニタ 542  
    J2EE モニタ 562

## L

Linux  
    IP アドレスの設定 680  
LoadRunner  
    LoadRunner を使った作業 6  
    アプリケーション・テスト 3  
    概要 3–13  
    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 25  
    テスト工程, 「テスト工程」 参照  
lr\_user\_data\_point 311

## M

Media Player クライアント  
    モニタ 520  
Microsoft  
    ASP モニタ 433  
    IIS モニタ 375  
MI リスナー 252  
MQ モニタ 602

## N

netid, IP アドレス・コンポーネント 677

## O

Oracle

    モニタ 494  
    ユーザ定義クエリー 499  
Oracle9iAS HTTP  
    モニタ 435

## R

RealPlayer  
    クライアント・モニタ 519  
    サーバ・モニタ 516  
remote\_host 630  
remote\_path 630  
rsh  
    UNIX で使用せずに実行 651  
    UNIX ネットワーク・モニタの接続 357  
    コントローラとの接続の確認 651  
rstatd の起動 336  
rstatd プロセス  
    起動 336  
    リソース・モニタ 336  
RTE 仮想ユーザ・スクリプト  
    コントローラ内の 12

## S

SAP  
    モニタ 522  
SAP Portal  
    モニタ 528  
Siebel  
    モニタ 532  
Siebel Server Manager  
    モニタ 536  
SilverStream  
    モニタ 440  
SiteScope  
    モニタ 351  
SNMP  
    リソース・モニタ 337  
Solaris  
    IP アドレスの設定 680  
SQL Server  
    モニタ 501  
SSL  
    MI リスナーの設定 253  
    エージェントの設定 250  
    概要 687  
Sybase

モニタ 504

## T

TCP/IP の設定 648

TestDirector

TestDirector へのシナリオの保存 193

TestDirector への保存 195

仮想ユーザ・スクリプトの追加 196

コマンド・ライン引数 685

シナリオを開く 192

スクリプトの管理 187

接続 188

切断 190

統合 187–198

TestDirector からの切断 190

TestDirector プロジェクトからシナリオを開く  
ダイアログ・ボックス 192

TestDirector プロジェクトへシナリオを保存ダ  
イアログ・ボックス 193

TestDirector への接続ダイアログ・ボックス  
188

Tuxedo

Monitor 設定ダイアログ・ボックス 597  
モニタ 596

TUXEDO 測定値の追加ダイアログ・ボックス  
597

## U

UNIX

rsh 651

rsh を使用しない 651

rstatd の起動 336

シェル 651

リソース・モニタ 333

ロード・ジェネレータへの接続 642

UNIX ProTune エージェント 244

## V

Vuser, 『仮想ユーザ』 参照

## W

WAN エミュレーション 95

IP の除外 99

構成設定 96

詳細オプション 98

セットアップ 95

定義済みのプロファイルの設定 96

停止と再開 101

WebLogic

(JMX) モニタ 448

モニタ 443

WebSphere

(EPM) モニタ 465

モニタ 454

Web アプリケーション・サーバ・リソース・  
モニタ 391–475

Ariba モニタ 392

ATG Dynamo モニタ 397

BroadVision モニタ 402

ColdFusion モニタ 411

Fujitsu INTERSTAGE モニタ 414

iPlanet (NAS) 417

Microsoft ASP 433

Oracle9iAS HTTP モニタ 435

SilverStream モニタ 440

WebLogic (JMX) モニタ 448

WebLogic モニタ 443

WebSphere (EPM) モニタ 465

WebSphere モニタ 454

Web サーバ・リソース・モニタ 371–381

Apache モニタ 372

iPlanet/Netscape モニタ 378

iPlanet (SNMP) モニタ 382

Microsoft IIS モニタ 375

Web ページ・ブレイクダウンの有効化 315

Web リソース・モニタ 317–323

Windows

Media サーバ・モニタ 514

Windows リソース

モニタ 327

## あ

アプリケーション

構成設定 21

使用モデル 22

分析 20–22

## い

一時停止

仮想ユーザ 212

モニタ 286

### インストール

LoadRunner, 『LoadRunner インストール・ガイド』 参照

### え

#### エージェント

サマリ・ウィンドウ 234

定義 7

デーモン 652

#### エキスパート・モード 633–643

UNIX ロード・ジェネレータへの接続 642

一般設定 634

出力の設定 638

デバッグ設定 635

モニタ設定 640

#### エラー—仮想ユーザのステータス

実行中の仮想ユーザ・グラフ 310

シナリオ・グループ表示枠 222

#### エラー処理 280

#### エラーの統計グラフ 310

### お

#### 応答時間の測定

GUI 26

エンドツーエンド 25

サーバ 26

ネットワークとサーバ 25

ミドルウェアとサーバ 26

#### オプション・ダイアログ・ボックス

Web ページのブレイクダウン・タブ 315

一般タブ 634

実行環境の設定タブ 170

実行ファイルの保存場所タブ 176

タイムアウト・タブ 172

デバッグ情報タブ 636

パス変換テーブル 631

#### オンライン・グラフ 281

2 つのグラフの結合 284

Y 軸の目盛り 283

エクスポート 292

オフラインでのデータ表示 292

解析 623–626

グラフを開く 274

更新率 282

サンプリング頻度 279

時間 282

設定 281

測定値の倍率の変更 288

データ・ポイント 311

トランザクション・データ 279

表示のカスタマイズ 277

棒グラフ値の種類 283

リモート監視 293

オンライン・グラフの解析 623–626

オンライン・サポート xii

オンラインでの Web サーバ・リソースの監視

プロキシ・サーバの利用 389

オンライン・トランザクションの監視

グラフ 312

設定 313

トランザクションの追加 314

オンライン文書 xi

オンライン・モニタ 269–272

一時停止 286

エラー処理 280

オフラインでのデータ表示 292

オンライン・グラフの解析 623

起動 272

グラフ 285

グラフ作成時間 283

グラフの設定 281

線の色 287

線の表示 / 非表示 289

測定値の設定 287

デバッグ 280

表示オプション 283

標準設定のカウンタの変更 669

### か

カウンタの変更, 標準設定 669

カウンタ, 負荷テスト 670

#### 仮想ユーザ

エラーのある仮想ユーザ・グラフ 310

グループ・リスト 54–61

RTE 仮想ユーザ 12

GUI 仮想ユーザ 9

一時停止 212

エラー, 警告, 通知メッセージ 225

仮想ユーザ・ウィンドウ 46

仮想ユーザ情報ダイアログ・ボックス 104

仮想ユーザ・スクリプト・ログ 213



- 仮想ユーザ・ログ 230
  - 監視 222
  - グループに追加 69, 70
  - 最大ユーザ負荷のエミュレート 159
  - 実行 211
  - 実行数の決定 27
  - シナリオ・グループ表示枠のステータス 222
  - シナリオ実行中のその他の仮想ユーザのアクティブ化 214
  - 種類 8
  - 設定 64
  - 定義 5
  - 停止 211
  - 番号の再割り当て 212
  - 表示 221–234
  - ロード 212
  - 仮想ユーザ・ウィンドウ 64
  - 仮想ユーザ・グループ
    - 一時停止 205
    - 仮想ユーザの追加 69
    - 仮想ユーザを追加 70
    - 削除 60
    - 作成 54–61
    - 実行 205
    - スケジュール 152
    - 停止 206
    - 変更 60
    - 有効化 / 無効化 59
    - ロード 204
  - 仮想ユーザ数の引き上げ 659
  - 仮想ユーザ・スクリプト 5
    - TestDirector からの追加 196
    - 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) 114
    - グループのための選択 69 (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 134 (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 134
    - コマンド・ライン・オプション 103
    - 削除 (ゴール指向シナリオ) 141
    - 削除 (パーセントモード) 120
    - 作成 16
    - シナリオの選択 37
    - スクリプトの詳細の変更 102, 104
    - 設定 102–105
    - 追加 (ゴール指向シナリオ) 137
    - 追加 (パーセントモード) 116
    - 定義 5
    - ファイル 104
    - 編集 104
    - 有効化 / 無効化 (ゴール指向シナリオ) 141
    - 有効化 / 無効化 (マニュアル・シナリオ) 120
    - 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) 114
  - 仮想ユーザの実行 / 停止ダイアログ・ボックス 214
  - 仮想ユーザのステータス
    - 実行時間グラフ 310
  - 仮想ユーザの追加ダイアログ・ボックス 57
  - 仮想ユーザの表示 221–234
    - エージェント・サマリ 234
    - 概要 221
    - 出力メッセージ・ウィンドウ 225
  - 仮想ユーザを追加ダイアログ・ボックス 69
  - 関数リファレンス xi
  - 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 222
  - 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 222
- く
- グラフ, 「オンライン・グラフ」 参照
  - グラフ作成時間 283
  - グラフの設定ダイアログ・ボックス 281
  - グラフを重ねて表示ダイアログ・ボックス 284
  - グループ情報ダイアログ・ボックス 61
- け
- 継続時間 151
  - 結果 180
    - TestDirector プロジェクトの場所 195
    - 格納場所の指定 180
    - 照合 184
    - ディレクトリ内のファイル構造 183
    - デバッグ対象ファイル 654
    - 名前 180

結果ディレクトリの設定ダイアログ・ボックス

TestDirector プロジェクト 195  
ローカルまたはリモート 181

### こ

更新ボタン 103  
ゴール指向シナリオ 123–141  
    ゴールの定義 128  
    シナリオ・タイプの選択 36  
    スクリプトへの目標の百分率の割り当て 134  
    スクリプトへのロード・ジェネレータの割り当て 134  
    デザイン・タブ 126  
コマンド・ライン  
    オプション, 仮想ユーザ・スクリプト 103  
    引数 683  
コンテキスト・センシティブ・ヘルプ xii  
コントローラ 38  
    概要 38  
    起動 36  
    クイック・ツアー 35–47  
    コマンド・ラインからの実行 683  
    シナリオ・ファイルの管理 42  
    定義 5  
コントローラ・ウィンドウ  
    アイコン・バー 41  
    実行ビュー 40  
    出力ウィンドウ 40  
    ステータス・バー 38  
    タイトル・バー 38  
    ツールバー 38  
    デザイン・ビュー 40  
    メニュー・バー 38  
コントローラの起動 36

### さ

サーバの複製 261  
サーバ・モニタ 261  
    測定値の追加と削除 262  
    測定頻度の設定 263  
    プロパティの設定 257  
サーバのルーティング・テーブル 681  
作成

仮想ユーザ・グループ 54–61  
仮想ユーザ・スクリプト 16  
ゴール指向シナリオ 123–141  
パーセントモードでのマニュアル・シナリオ 109–122  
マニュアル・シナリオ 51–107

サポート情報 xii  
サマリ情報ダイアログ・ボックス 186  
サンプリング頻度 279

### し

システム・リソース・モニタ 325–352  
    Antara FlameThrower モニタ 339  
    SiteScope Monitor 351  
    SNMP リソース・モニタ 337  
    UNIX リソース・モニタ 333  
    Windows リソース・モニタ 327  
実行—仮想ユーザのステータス  
    実行中の仮想ユーザ・グラフ 310  
    シナリオ・グループ表示枠 222  
実行環境の設定  
    共有 72  
    グループ 71  
    コントローラ内の 139  
    コントローラ内の設定 71, 102  
    スクリプト 71  
実行完了—仮想ユーザのステータス  
    実行中の仮想ユーザ・グラフ 310  
実行時間グラフ 309–314  
    解析 623  
実行メモ・ダイアログ・ボックス 233  
シナリオ  
    TestDirector から開く 192  
    TestDirector への保存 193  
    概要 52  
    仮想ユーザ・グループ・モードへの切り替え 122  
    結果ディレクトリ 183  
    結果の照合 184  
    ゴール指向シナリオの作成 123–141  
    サマリ情報 186  
    実行 201–213  
    実行の概要 45–47  
    実行の準備 179–186  
    シナリオ開始ダイアログ・ボックス 145

- シナリオ・スクリプト表示枠 126, 128
  - シナリオのゴール・ウィンドウ 126, 128
  - シナリオのゴールの定義 128
  - シナリオの新規作成 42
  - シナリオ・ファイルの管理 42-44, 45
  - 出力メッセージの表示 225
  - 新規シナリオ・ダイアログ・ボックス 52
  - スケジュール 149
  - 設定 169-178
  - 定義 5
  - 定義されたゴール指向シナリオ 36
  - 定義されたマニュアル・シナリオ 36
  - 閉じる 44
  - パーセントモードでのマニュアル・シナリオの作成 109-122
  - パーセントモードへの切り替え 110
  - 開く 43
  - 保存 44
  - マニュアル・シナリオの作成 51-107
  - シナリオゴールの編集ダイアログ・ボックス 128
  - シナリオ設定タブ 128
  - 負荷の動作タブ 128
  - プレビューのロード 128
  - シナリオの切り替え
    - 仮想ユーザ・グループ・モードへの 122
    - パーセントモードへの 110
  - シナリオの結果の照合 184
  - シナリオの実行 201-213
    - アクティブな仮想ユーザの監視 222
    - 概要 45-47, 201
    - 仮想ユーザ・グループの一時停止 205
    - 仮想ユーザ・グループの実行 205
    - 仮想ユーザ・グループの停止 206
    - 仮想ユーザ・グループのロード 204
    - 仮想ユーザのロード 212
    - 継続時間の制限 149
    - 個々の仮想ユーザ・グループの制御 204
    - 個々の仮想ユーザの制御 211
    - その他の仮想ユーザのアクティブ化 214
    - 遅延 145, 149
  - 無人でのシナリオの実行 203
  - メッセージ 225
  - ランデブーからの手作業による仮想ユーザの解放 213
  - シナリオの設定
    - 結果の格納場所の指定 180
    - 実行環境の設定 170
    - 実行ファイルの格納場所 175
    - タイムアウト時間 172
    - パス変換 178
  - 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 222
  - 出力ウィンドウ
    - デバッグ情報 656
    - 表示 40
  - 出力ファイル 657
  - 出力メッセージ・ウィンドウ 225-230
    - 更新 227
    - 消去 229
    - メッセージの詳細の表示 228
    - メッセージの並べ替え 229
    - メッセージのファイルへの保存 228
    - メッセージのフィルタ処理 228
    - ログ情報のドリル・ダウン 226
  - 準備完了-仮想ユーザのステータス
    - 実行中の仮想ユーザ・グラフ 310
    - シナリオ・グループ表示枠 222
  - 詳細ボタン 228
  - 初期化 211
  - 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 222
  - 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 222
  - 新規監視対象サーバ・プロパティ・ダイアログ・ボックス 258, 261
  - 新規グラフを開くダイアログ・ボックス 275
  - 新規シナリオ・ダイアログ・ボックス 36
  - 新規ロードジェネレータの追加ダイアログ・ボックス 74
- 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 138
  - スクリプト・パス, 相対 106
  - スクリプトを追加ダイアログ・ボックス 137

- スケジュールの編集 146
- スケジュール・ビルダ 143–157
  - 仮想ユーザ・グループ 152
  - シナリオの実行 149
  - スケジュールの削除 148
  - スケジュールの作成 147
  - スケジュールの選択 146, 146–157
  - スケジュールの名前変更 148
  - スケジュールの変更 147
  - スケジュールビルダ・ダイアログ・ボックス 147
- スケジュール・ビルダ・ダイアログ・ボックス
  - 継続時間タブ 151
  - ランプ・アップ・タブ 150
  - ランプ・ダウン・タブ 152
- 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 153
  - 継続時間タブ 155
  - ランプ・アップ・タブ 154
  - ランプ・ダウン・タブ 156
- ステータス・バー 38
- ストリーミング・メディア・モニタ 513–519
  - Media Player Client モニタ 520
  - RealPlayer クライアント・モニタ 519
  - RealPlayer サーバ・モニタ 516
  - Windows Media サーバ・モニタ 514
- スループット・グラフ 318

## せ

### 接続

- TestDirector への 188
- データベースへ 653

接続グラフ 322

接続ログ・タブ 642

### 設定

- 一般 634
- 仮想ユーザ 64
- シナリオ 169–178
- 出力 638
- スクリプト 102–105
- スクリプト (ゴール指向シナリオ) 137–141
- スクリプト (パーセントモード) 116–120

- 測定頻度 263
- タイムアウト時間 172
- デバッグ 635
- モニタ 640
- ロード・ジェネレータ 52, 74–78, 79–101
- ロード・ジェネレータの設定 79–101
- 設定, 測定頻度 263

## そ

- 相対スクリプト・パス 106
- 測定値の削除 262
- 測定値の設定ダイアログ・ボックス
  - 詳細タブ 287
  - 設定タブ 287
- 測定値の追加 262
- 測定値の表示 / 非表示
  - オンライン・モニタ 286
  - トランザクション・モニタ 289

## た

- ターミナル・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
- 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 222

## 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 222

## つ

ツールバー 38

## て

### 停止

- 仮想ユーザ 211
- 仮想ユーザ・グループ 206
- データベース
  - 接続 653
  - データベース・サーバ・リソース・モニタ 477–511

DB2 モニタ 478  
 Oracle モニタ 494  
 SQL Server モニタ 501  
 Sybase モニタ 504  
 データ・ポイント・グラフ (オンライン) 311  
 デジタル証明書  
 MI リスナーの設定 253  
 エージェントの設定 250  
 概要 687  
 テスト工程  
 仮想ユーザ・スクリプトの作成 16  
 シナリオの監視 18  
 シナリオの作成 16  
 シナリオの実行 17  
 テスト結果の分析 18  
 テストの計画 16  
 テスト目標  
 一般的目標の要約 31–33  
 定義 23–24  
 デバッグ  
 情報設定 635  
 レベル 280

## と

トラブルシューティング  
 コントローラ 645–667  
 ネットワークについて考慮すべき事項  
 618  
 ファイアウォール 660  
 モニタ 613–619  
 トランザクション 5  
 失敗 655  
 ダイアログ・ボックス 224  
 定義 5  
 定義対象の決定 27  
 トランザクション応答時間グラフ 312  
 秒ごとのトランザクション (失敗, 中  
 止) グラフ 312  
 秒ごとのトランザクション (成功) グ  
 ラフ 312  
 トランザクション・データ 279  
 トランザクション・モニタ 309–314

## ね

ネットワーク  
 遅延時間グラフ 363

定義済みのパスのネットワーク監視設  
 定ダイアログ・ボックス 358  
 ネットワーク遅延時間ビュー 283  
 ブレークダウン・ダイアログ・ボック  
 ス 364  
 ネットワーク・セグメントの遅延の表示 364  
 ネットワーク遅延監視先マシンの追加ダイア  
 ログ・ボックス 358  
 ネットワーク・モニタ 353–364  
 UNIX 355  
 概要 353  
 設定 358  
 ファイアウォール越しの監視 362  
 ボトルネックの特定 354

## 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 122  
 仮想ユーザの総数に対する割合をスク  
 リプトに割り当てる 114  
 仮想ユーザの総数の定義 112  
 シナリオの作成 109–122  
 シナリオをパーセントモードに切り替  
 え 110  
 スクリプトへのロード・ジェネレータ  
 の割り当て 114  
 デザイン・タブ 111  
 ハードウェア  
 通信の確認 647  
 テスト対象の選択 28–30  
 パケット 354  
 パス変換  
 シナリオの設定 178  
 スクリプト・パス 106  
 定義 627  
 パス変換テーブルの使用 629  
 パス変換テーブルの編集 631  
 ファイル位置のデバッグ 654  
 例 632  
 パフォーマンス測定の範囲 25  
 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 212

## ひ

秒ごとにダウンロードされたページ・グラフ

321

秒ごとの SSL 接続グラフ 322  
秒ごとの再試行数グラフ 322  
秒ごとの接続数グラフ 322  
秒ごとのトランザクションの総合計 (成功)  
グラフ 312  
秒ごとのヒット数グラフ 318

## ふ

ファイアウォール  
MI リスナーのインストール 252  
インストールの設定 238  
仮想ユーザの実行 235–266  
越えての監視 235, 266  
コントローラの設定 254  
トラブルシューティング 264, 660  
ネットワークの監視 362  
ファイアウォール越しに動作するエー  
ジェントの設定 249  
ファイアウォール・サーバ・モニタ  
365–369  
ファイアウォール越しの監視 235, 235–266  
インストール 256  
測定値の追加と削除 262  
頻度の設定, 測定値 263  
プロパティの設定 257  
ファイアウォール越しの監視機能のインス  
トール 256  
ファイアウォール越しの実行 235–266  
ファイル, 仮想ユーザ・スクリプト 104  
負荷テスト, 定義 3  
負荷テストの計画 19–33  
負荷分散 136  
プロキシ・サーバ 389

## 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 222

## ま

マニュアル xii  
マニュアル・シナリオ  
作成 51–107  
定義 36  
デザイン・タブ 55

パーセントモード 109–122  
マルチ IP アドレス 635  
コントローラへの接続 650  
有効化 635

## み

ミドルウェア  
応答時間の測定 26  
システム構成 21  
ミドルウェア・パフォーマンス・モニタ  
595–611  
IBM WebSphere MQ モニタ 602  
Tuxedo モニタ 596

## 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)  
141  
仮想ユーザ・スクリプト (マニユ  
アル・シナリオ) 120

## も

モニタ  
Application Deployment ソリューション  
583–593  
ERP/CRM サーバ・リソース 521–540  
Java パフォーマンス 541–582  
Web アプリケーション・サーバ・リ  
ソース 391–475  
Web サーバ・リソース 371–381  
Web リソース 317–323  
オンライン 269–272  
システム・リソース 325–352  
実行時間 310  
ストーリーミング・メディア 513–519  
データベース・サーバ・リソース  
477–511  
トランザクション 312  
ネットワーク 353–364  
ファイアウォール・サーバ 365–369  
ミドルウェアのパフォーマンス  
595–611  
モニタ設定ダイアログ・ボックス 258

## ゆ

- ユーザ定義クエリー
  - Oracle モニタ 499
- ユーザ定義データ・ポイント・グラフ 311

## ら

- ランタイム・ビューア
  - コントローラから再生の表示 211
- ランデブー 159–166
  - 解放ポリシーの設定 163
  - 概要 159
  - 仮想ユーザのステータス 222
  - 仮想ユーザを無効にする 164
  - 仮想ユーザを有効にする 164
  - 情報 166
  - 属性 160
  - 属性の設定 161
  - タイムアウト・ポリシーの設定 163
  - 定義 5
  - 手作業による仮想ユーザの解放 213
- ランデブー情報ダイアログ・ボックス 166
- ランプ・アップ 150
- ランプ・ダウン 152

## り

- リスト 16
  - 仮想ユーザ・グループのリスト 54–61
  - スクリプト・リスト 102–105
  - スクリプト・リスト (ゴール指向シナリオ) 137–141
  - スクリプト・リスト (パーセントモード) 116–120
  - ランデブー・リスト 159–166
  - ロード・ジェネレータ・リスト 74–78
- リモート・エージェント・ディスパッチャ (プロセス) 7
- リモート・パフォーマンス監視 293–307
  - オンライン・グラフの表示 300
  - グラフ設定の構成 303
  - グラフの更新 305
  - グラフの測定値の構成 306
  - グラフの倍率設定 303
  - モニタのインストール 295
  - モニタへの接続 298
  - ユーザ設定の構成 295

## る

- ルーティング・テーブル 681

## れ

- レジストリの変更 659

## ろ

- ロード・ジェネレータ 5
  - IP アドレスの追加 675
  - 構成設定 52, 74–78
  - 選択 69
  - 選択 (パーセントモード) 114
  - 属性の設定 79–101
  - 追加 77
  - 定義 5
  - 分散 136
  - 変更 77
  - マルチ IP アドレス 635
  - 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 98
  - WAN エミュレーション・タブ 96
  - 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 79
  - 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 641
  - 仮想ユーザの制限 85
  - コントローラとの通信の確認 648
  - 実行ファイル 81
  - 初期化クォータ 84
  - ターミナル・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

## LoadRunner コントローラ・ユーザーズ・ガイド

- 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



## Host Resolution Functions Copyright Agreement

Copyright (c) 1980, 1983, 1985, 1987, 1988, 1989, 1990, 1993

The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

This product includes software developed by the University of California, Berkeley and its contributors.

4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 1993 by Digital Equipment Corporation.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies, and that the name of Digital Equipment Corporation not be used in advertising or publicity pertaining to distribution of the document or software without specific, written prior permission.

THE SOFTWARE IS PROVIDED "AS IS" AND DIGITAL EQUIPMENT CORP. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL DIGITAL EQUIPMENT CORPORATION BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 1996 by Internet Software Consortium.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.



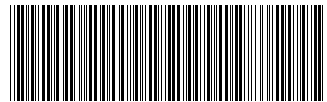




マーキュリー・インタラクティブ・ジャパン株式会社  
〒105-0003  
東京都港区西新橋 2-38-5 西新橋 MF ビル 7 階

電話 : (03) 5402-9300  
ファックス : (03) 5425-2288

Web: <http://www.mercury.co.jp>  
カスタマー・サポート : <http://www.mercury.co.jp/support>



\*LRCTRUG7. 8JP/ 01\*