HP QuickTest Professional

ソフトウェア・バージョン:9.5

ユーザーズ・ガイド 第 2 巻

製造部品番号:T6511-99016 文書発行日:2008 年 1 月(英語版) ソフトウェア・リリース日:2008 年 1 月(英語版)



利用条件

保証

HPの製品およびサービスの保証は、かかる製品およびサービスに付属する明示的な保証の声明 において定められている保証に限ります。本文書の内容は、追加の保証を構成するものではあり ません。HPは、本文書に技術的な間違いまたは編集上の間違い、あるいは欠落があった場合で も責任を負わないものとします。

本文書に含まれる情報は、事前の予告なく変更されることがあります。

制限事項

本コンピュータ・ソフトウェアは,機密性があります。これらを所有,使用,または複製するに は,HPからの有効なライセンスが必要です。FAR 12.211 および 12.212 に従って,商用コン ピュータ ソフトウェア,コンピュータ ソフトウェアのドキュメント,および商用アイテムの技 術データは,HPの標準商用ライセンス条件に基づいて米国政府にライセンスされています。

サードパーティ Web サイト

HPは、補足情報の検索に役立つ外部サードパーティWebサイトへのリンクを提供します。サイトの内容と利用の可否は予告なしに変更される場合があります。HPは、サイトの内容または利用の可否について、いかなる表明も保証も行いません。

著作権

© 1992 - 2008 Hewlett-Packard Development Company, L.P.

商標

Adobe® および Acrobat® は, Adobe Systems Incorporated の商標です。

Intel®, Pentium® および Intel® XeonTMは、米国およびその他の国における Intel Corporation また はその子会社の商標または登録商標です。

JavaTMは, Sun Microsystems, Inc. の米国商標です。

Microsoft[®], Windows[®], Windows NT[®] および Windows XP[®] は, Microsoft Corporation の米国登録 商標です。

Oracle®は、カリフォルニア州レッドウッド市の Oracle Corporation の米国登録商標です。

Unix[®]は, The Open Group の登録商標です。

SlickEdit® は, SlickEdit Inc. の登録商標です。

文書の更新

本書のタイトル・ベージには、次の識別情報が含まれています。

- ソフトウェアのバージョンを示すソフトウェア・バージョン番号
- 文書が更新されるたびに更新される文書発行日
- 本バージョンのソフトウェアをリリースした日付を示す、ソフトウェア・リリース日付 最新のアップデートまたは文書の最新版を使用していることを確認するには、 http://h50146.www5.hp.com/doc/manual/openview/を参照します。

サポート

HP Software Support Web $\forall 1 \land \neg d$, **support.openview.hp.com** $h \land \neg d$, $h \land \neg d$, h

HP Software のオンライン・サポートでは、対話型の技術支援ツールに効率的にアクセスできます。サポートをご利用のお客様は、サポート・サイトを使うことで次のような利点があります。

- 参照したいナレッジ文書の検索
- エンハンスメント要求およびサポート・ケースの登録とトラッキング
- ソフトウェア・パッチのダウンロード
- サポート契約の管理
- HP サポートの連絡先の検索
- 利用可能なサービスに関する情報の参照
- ソフトウェアの他のお客様とのディスカッションに参加
- ソフトウェアのトレーニングの調査および登録

ほとんどのサポート・エリアは, HP Passport ユーザとしての登録およびサインインが必要です。 また多くは,サポート契約も必要です。アクセス・レベルの詳細情報については, www.hp.com/managementsoftware/access_level を参照してください。

HP Passport ID の登録を申請するには,

www.managementsoftware.hp.com/passport-registration.html (英語サイト) にアクセスしてくだ さい。

目次

この目次は,**『HP QuickTest Professional ユーザーズ・ガイド』**の第1巻と第2 巻両方の章を含みます。

本書について	xix
本ガイドの構成	
対象読者	xxii
QuickTest Professional オンライン・ドキュメント	xxiii
その他のオンライン・リソース	xxv
表記規則	xxvi

第 I 部: QuickTest Professional の概要(第 1 巻)

第1章:はじめに	29
QuickTest を使用したテスト	31
テスト・プロセスについて	32
エキスパート・ビューでのプログラミング	38
関数と関数ライブラリについて	38
Quality Center を使ったテスト・プロセスの管理	39
Business Process Testing について	40
必要なアクセス許可の設定	41
サンプル・サイトの使用方法	42
ライセンス情報の変更	42
QuickTest ソフトウェアの更新	43

第2章: QuickTestの概要	45
QuickTest の開始	46
QuickTest のウィンドウ	48
キーワード・ビュー	52
エキスパート・ビュー	53
関数ライブラリ	54
スタート・ページ	55
ActiveScreen	57
情報表示枠	58
使用可能なキーワード表示枠	59
テスト・フロー表示枠	60
リソース表示枠	61
欠落リソース表示枠	62
プロセス・ガイダンス表示枠	63
データ・テーブル	64
デバッグ・ビューア表示枠	64
QuickTest コマンドの使用方法	65
QuickTest Professional プログラム・フォルダの参照	90
製品情報の表示	94

第 II 部:テスト・オブジェクトを使用した作業(第1巻)

第3章:テスト・オブジェクト・モデルについて	101
テスト・オブジェクト・モデルについて	101
テスト・オブジェクト・モデルの概念の適用	
オブジェクト・スパイを使用したオブジェクトのプロパティと	
メソッドの表示	111

第4章:テスト・オブジェクトを使用した作業	117
オブジェクトを使用した作業について	118
オブジェクト・リポジトリの種類について	119
[オブジェクト リポジトリ] ウィンドウについて	126
テスト・オブジェクト・プロパティの表示と変更	136
リポジトリ・パラメータ値の割り当て	158
ローカルまたは共有オブジェクト・リポジトリへの	
テスト・オブジェクトの追加	163
新規テスト・オブジェクトの定義	176
オブジェクト・リポジトリ内のオブジェクトのコピー,	
貼り付け, および移動	178
オブジェクト・リポジトリからのオブジェクトの削除	181
オブジェクトの検索	182
実行セッション中のテスト・オブジェクトでの作業	190
共有オブジェクト・リポジトリの関連付けの管理	191
共有オブジェクト・リポジトリへのローカル・オブジェクトの	
エクスポート	195
第5音・ナブジェクトの認識の設定	100
第5章:オブジェクトの認識の設定 オブジェクトの認識の設定について	199 199
第5章:オブジェクトの認識の設定	199 199 201
第5章:オブジェクトの認識の設定 オブジェクトの認識の設定について [オブジェクトの認識]ダイアログ・ボックスについて	199 199 201 214
第5章:オブジェクトの認識の設定 オブジェクトの認識の設定について	199 199 201 214 224
第5章:オブジェクトの認識の設定 オブジェクトの認識の設定について	199 201 214 224
第5章:オブジェクトの認識の設定 オブジェクトの認識の設定について	199 201 214 224 227
第5章:オブジェクトの認識の設定 オブジェクトの認識の設定について [オブジェクトの認識]ダイアログ・ボックスについて スマート認識の設定 ユーザ定義のテスト・オブジェクト・クラスの割り当て 第6章:オブジェクト・リポジトリの管理 オブジェクト・リポジトリの管理について	199 201 214 224 227 228
第5章:オブジェクトの認識の設定 オブジェクトの認識の設定について	199 201 214 224 227 228 230
第5章:オブジェクトの認識の設定 オブジェクトの認識の設定について	199 201 214 224 227 228 230 237
 第5章:オブジェクトの認識の設定… オブジェクトの認識の設定について [オブジェクトの認識]ダイアログ・ボックスについて スマート認識の設定 ユーザ定義のテスト・オブジェクト・クラスの割り当て 第6章:オブジェクト・リポジトリの管理 オブジェクト・リポジトリの管理について オブジェクト・リポジトリを使った作業 共有オブジェクト・リポジトリでのオブジェクトの管理 	199 201 214 224 227 228 230 237 242
第5章:オブジェクトの認識の設定… オブジェクトの認識の設定について	199 201 214 224 227 228 230 237 242 242
第5章:オブジェクトの認識の設定… オブジェクトの認識の設定について	199 201 214 224 228 230 237 242 242 248 254
第5章:オブジェクトの認識の設定 オブジェクトの認識の設定について	199 201 214 224 227 228 230 237 242 248 254 259
第5章:オブジェクトの認識の設定… オブジェクトの認識の設定について	199 201 214 224 227 228 230 237 242 248 254 259 260
第5章:オブジェクトの認識の設定… オブジェクトの認識の設定について	199 201 214 224 224 227 228 230 237 242 248 254 259 260 261

第7章:共有オブジェクト・リポジトリの結合	
共有オブジェクト・リポジトリの結合について	
オブジェクト・リポジトリ結合ツールについて	
オブジェクト・リポジトリ結合ツールのコマンドの使用方法	
標準設定の定義	
2つのオブジェクト・リポジトリの結合	
ローカル・オブジェクト・リポジトリからの	
共有オブジェクト・リポジトリの更新	
結合の統計情報の表示	
オブジェクトの矛盾について	
オブジェクトの矛盾の解決	
ターゲット・リポジトリ表示枠に対するフィルタの設定	
特定のオブジェクトの検索	
ターゲット・オブジェクト・リポジトリの保存	
第8章:共有オブジェクト・リポジトリの比較	
共有オブジェクト・リポジトリの比較について	310
オブジェクト・リポジトリ比較ツールについて	
オブジェクト・リポジトリ比較ツールのコマンドの使用	
オブジェクトの相違点について	
色の設定の変更	
オブジェクト・リポジトリの比較	
比較の統計情報の表示	
リポジトリ表示枠に対するフィルタの処理	
オブジェクト・リポジトリ・ビューの同期	
特定のオブジェクトの検索	

第 Ⅲ 部:テストの設計(第1巻)

第9章:テストの作成 ー 概要	333
テストの作成について	333
使用する方法論の決定 – キーワード駆動型または記録	335
テストについて	337
テストの拡張	338
QuickTest での相対パスの使用	340
テストの管理	344
第 10 章 : キーワード駆動型の方法論を使用したテストの作成	351
キーワード駆動型の方法論について	352
キーワード駆動型の方法論の使用	354
キーワード駆動型の方法論の実装のサンプル	363

第11章:記録メカニズムを使用したテストの作成	.373
テストの記録について	.374
てんトの記録	.3/5
記録モートの選択	.380
ActiveScreen を使うた作業	. 300
第 12 章 : キーワード・ビューを使った作業	.395
キーワード・ビューを使った作業について	.396
キーワード・ビューについて	.397
QuickTest のオブジェクト階層について	.402
テストへの標準ステップの追加	.404
テストへのその他のタイプのステップの追加	.419
ステップの部分の変更	.422
コメントの使用	.422
アクションのステップの管理	.423
キーワード・ビューでのキーボード・コマンドの使用	.426
キーワート・ヒューの表示オフションの定義	.427
キーリート・ビューでの人ナッノ要素の表示ノロハナイ	.432
キーリート・ビューのフレークホイントを使った作業	.433
第 13 章 : 高度なアクション機能を使用した作業	.435
高度なアクション機能を使用した作業について	.436
既存のアクションへの呼び出しの挿入	.436
アクション・パラメータの設定	.444
アクション・パラメータの使用	.448
アクションの呼び出しのプロパティの設定	.453
アクション情報の共有	.458
エキスパート・ビューのアクションの構文について	.461
アクションの終了	.463
第 14 音・アクションを使った作業	465
アクションを使った作業について	.466
グローバル・データ・シートとアクション・データ・シートの使用	.469
[テスト フロー] 表示枠の使用	.471
キーワード・ビューのアクション・ツールバーの使用	.475
新規アクションの作成	.477
アクションを使った作業についてのガイドライン	.479
アクションのプロパティの設定	.481
アクションのネスト	.492
アクションの分割	.494
アクション名の変更	.496
テストからのアクションの削除	100
	.433

第 IV 部:テストの拡張(第1巻)

第 15 章 : チェックポイントについて	507
チェックポイントについて	507
新しいチェックポイントのテストへの追加	508
既存のチェックポイントのテストへの追加	510
チェックポイントの種類について	513
筆 16 音・ビットマップの検査	519
ビットマップの検査について	519
ビットマップの検査	520
ビットマップ・チェックポイントの修正	530
第 17 音・オブジェクトのプロパティの値の検査	535
オブジェクトのプロパティ値の検査について	
標準チェックポイントの作成	
「チェックポイントのプロパティ」ダイアログ・ボックスについて	538
「画像チェックポイントのプロパティー	
ダイアログ・ボックスについて	543
チェックポイントの変更	545
第 18 音・テーブルの絵本	5/7
第 18 章 : テーブルの検査 テーブルの検査について	547 547
第 18 章 : テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成	547 547 548
第 18 章 : テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 「テーブル チェックポイントのプロパティ]	547 547 548
第 18 章 : テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて	547 547 548 552
第 18 章: テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査	547 547 548 552 554
第 18 章: テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査 テーブル・プロパティの検査	547 547 548 552 554 554
第 18 章: テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査 テーブル・プロパティの検査 テーブル・チェックポイントの変更	547 547 548 552 554 564 566
 第18章:テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査 テーブル・プロパティの検査 テーブル・チェックポイントの変更 第19章:テキストの検査 	547 547 548 552 554 564 566 569
 第18章:テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査 テーブル・プロパティの検査 テーブル・チェックポイントの変更 第19章:テキストの検査 テキストの検査 	547 547 548 552 554 564 566 569 569
 第 18 章: テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査 テーブル・プロパティの検査 テーブル・チェックポイントの変更 第 19 章: テキストの検査 テキストの検査について テキストの検査について テキストの検査 	547 547 548 552 554 564 566 569 569 569 569
 第 18 章: テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査 テーブル・プロパティの検査 テーブル・チェックポイントの変更 第 19 章: テキストの検査 テキストの検査について テキストの検査について テキストの検査について テキストの作成 テキスト領域チェックポイントの作成 	547 547 548 552 554 564 566 569 569 569 571 572
 第18章:テーブルの検査 テーブルの検査について デーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査 テーブル・プロパティの検査 テーブル・チェックポイントの変更 第19章:テキストの検査 テキストの検査について テキスト・チェックポイントの作成 テキスト領域チェックポイントのプロパティ] と 	547 547 548 552 554 564 566 569 569 569 569 572
第 18 章: テーブルの検査 テーブルの検査について デーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて デーブル内容の検査 テーブル・プロパティの検査 テーブル・チェックポイントの変更 第 19 章: テキストの検査 テキストの検査について テキストの検査について テキスト・チェックポイントの作成 テキスト領域チェックポイントのプロパティ] と [テキスト領域チェックポイントのプロパティ]	547 547 548 552 554 564 566 569 569 571 572
 第18章:テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査 テーブル・プロパティの検査 テーブル・チェックポイントの変更 第19章:テキストの検査 テキストの検査について 	547 547 554 554 564 566 569 569 571 572
第 18 章: テーブルの検査 テーブルの検査について テーブル・チェックポイントの作成 [テーブル チェックポイントのプロパティ] ダイアログ・ボックスについて テーブル内容の検査 テーブル・プロパティの検査 テーブル・チェックポイントの変更 第 19 章: テキストの検査 テキストの検査について テキストの検査について テキスト・チェックポイントの作成 テキスト領域チェックポイントの作成 [テキスト チェックポイントのの作成 [テキスト領域チェックポイントのプロパティ] と [テキスト領域チェックポイントのプロパティ] ダイアログ・ボックスについて テキストまたはテキスト領域チェックポイントの変更	547 547 554 554 564 566 569 579 571 572 575 586

第 20 章 : データベースの検査	591
データベースの検査について	591
データベースの検査の作成	592
[データベース チェックポイントのプロパティ]	
ダイアログ・ボックスについて	597
データベース・チェックポイントの変更	606
第 21 章 : XML の検査	609
XML の検査について	610
XML チェックポイントの作成	612
XML テスト・オブジェクト操作チェックポイントの XML 階層の更新	
(WebService テスト・オブジェクトの場合のみ)	631
XML チェックポイントの変更	639
XML チェックポイントの結果の確認	639
XML オブジェクトとメソッドの使用によるテスト内容の拡張	640
第 22 章 : 値のパラメータ化	641
値のパラメータ化について	641
ステップおよびチェックポイントの値のパラメータ化	643
テスト・パラメータとアクション入力パラメータの使用	651
データ・テーブル・パラメータの使用	655
環境変数パラメータの使用	661
乱数パラメータの使用	671
パラメータ化したテストの例	673
データ・ドライバによるテストのパラメータ化	678
第 23 章 : 値の出力	685
値の出力について	685
出力値の作成	686
プロパティ値の出力	692
出力のタイプと設定の指定	698
テキスト値の出力	703
テーブル値の出力	712
データベース値の出力	726
ア ア マ ス に の 山 力	730
XML テスト・オブジェクト操作出力値ステップの XML 陛屆の再新	100
(WebService テスト・オブジェクトの提合のみ)	7/3
(Webbervice) ストー オンシェントの場合の(ア)	743
それでもとして、それので、こと、ここで、ここで、ここで、ここで、ここで、ここで、ここで、ここで、ここで	141
第 24 章 : 値の設定	751
値の設定について	751
定数およびパラメータ値の設定	752
正規表現の使用について	757
正規表現の定義	759

第 25 章 : プログラミング・ロジックを含むステップの追加	
プログラミング・ロジックを含むステップの追加について	768
ステップ・ジェネレータを使用したステップの挿入	769
条件ステートメントの使用	
ループ・ステートメントの使用	794
テストに対する「With」ステートメントの生成	797
メッセージの生成	
コメントの追加	
テストの同期化	

第 V 部: 関数およびその他のプログラミング・タスクの定義(第 2 巻)

第 26 章:エキスパート・ビューおよび関数ライブラリ・ウィンド	ウを
使用した作業	817
エキスパート・ビューおよび関数ライブラリ・ウィンドウを	
使用した作業について	818
エキスパート・ビューの理解と使用	819
エキスパート・ビューおよび関数ライブラリ内での操作	831
VBScript の基本的な構文の理解	841
プログラム的記述の使用	850
プログラムによるアプリケーションの実行と終了	862
コメント,フロー制御,そのほかの	
VBScript ステートメントの使用	863
テスト・オブジェクトのプロパティ値の取得と設定	872
実行環境オブジェクトのプロパティおよびメソッドへのアクセス…	873
DOS コマンドの実行	875
Windows API を使用したテストおよび関数ライブラリの拡張	875
実行セッション中に報告するステップの選択	879

第27章:エキスパート・ビューおよび

関数ライフラリ・ウィンドウのカスタマイズ	
[エキスパート ビュー] および関数ライブラリ・ウィンドウの	
カスタマイズについて	
エディタの動作のカスタマイズ	
エレメントの見映えのカスタマイズ	
編集コマンドのカスタマイズ	

第28章:ユーザ定義関数および関数ライブラリを使用した作業……891
ユーザ定義関数および関数ライブラリの使い方について……891
関数ライブラリの管理……893
関連付けられている関数ライブラリを使用した作業……905
関数定義ジェネレータの使用方法……909
ユーザ定義関数のテスト・オブジェクト・メソッドとしての登録……924
ユーザ定義関数の使い方のヒント……931
テストからの外部定義された関数の実行……933

第 VI 部: テストの実行と分析(第2巻)

第 29 章 : テストの実行	937
テストの実行について	
テスト全体の実行	938
テストの一部の実行	
オプション・ステップの使用	
テスト・バッチの実行	
· · · · · · · · · · · · · · · · · · ·	
「第 30 章 : 実行セッション結果の表示	951
実行セッション結果の表示について	952
[テスト結果] ウィンドウ	953
実行セッションの結果の表示	
テスト実行結果の削除	
実行セッション中に検出された不具合の送信	
テスト結果での WinRunner テスト・ステップの表示	
テスト結果の表示のカスタマイズ	
第 31 章 : 実行セッション結果の分析	
テスト結果に含まれるスマート認識情報の分析	
チェックポイント結果の分析	
パラメータ化された値と出力値の結果の表示	

第 VII 部:テストの保守およびデバッグ(第2巻)

第 32 章 : テストと関数ライブラリのデバッグ	
テストと関数ライブラリのデバッグについて	
デバッグ・セッションの速度調節	1039
シングル・ステップ・コマンドの使用	1040
[ステップまで実行] コマンドおよび	
[ステップからデバッグ] コマンドの使用	1043
実行セッションの一時停止	
ブレークポイントの使用	
デバッグ・ビューアの使用	1049
実行エラーの処理	
アクションまたは関数のデバッグの練習	
第 33 章 : テストの保守	1057
テストが失敗する理由	1057
メンテスナンス実行ウィザードを使ったテストの実行.	1060
更新モード・オプションを使ったテストの更新	

第 VIII 部: QuickTest IDE を使用した作業(第2巻)

第 34 章 : QuickTest ウィンドウのレイアウト	1087
複数のドキュメントを使った作業	1097
第 35 章 : リソースの管理	1101
[リソース] 表示枠について	1101
第 36 章 : テストへのキーワードの追加	1107
[使用可能なキーワード]表示枠について	1107
第 37 章 : 欠落リソースの処理	1111
欠落リソースの処理について	1112
欠落アクションの処理	1114
欠落している環境変数ファイルの処理	1119
欠落している関数ライブラリの処理	1120
欠落している共有オブジェクト・リポジトリの処理	1122
欠落している回復シナリオの処理	1123
未割り当ての共有オブジェクト・リポジトリ・パラメータ値の	
処理	1126

第 38 章 : データ・テーブルを使った作業	1127
データ・テーブルを使った作業について	1127
グローバル・シートおよびアクション・シートを使った作業	1130
データ・テーブルの保存	1131
データ・テーブルの編集	1132
Quality Center でのデータ・テーブル・ファイルの使用	1141
データベースからのデータのインポート	1142
データ・テーブルでの数式の使用	1145
データ・テーブルのスクリプト・メソッドの使用	1149

第 39 草 : フロセス・カイダンスを使った作業	
[プロセス ガイダンス] 表示枠	
プロセス・ガイダンスの開始	1154
利用できるプロセスのリストの管理	

第 IX 部: QuickTest の設定(第 2 巻)

第 40 章 : グローバル・テスト・オプションの設定	1161
グローバル・テスト・オプションの設定について	1161
[オプション] ダイアログ・ボックスの使用	1162
テストの一般オプションの設定	1164
テストのフォルダ・オプションの設定	1169
ActiveScreen オプションの設定	1172
テストの実行オプションの設定	1181
	4407
第41 早:個別のナストのオノション設定	
個別のナストのオフションの設定について	
[テストの設定] タイアロク・ホックスの使用	
テストのためのプロパティの定義	1191
テストのための実行設定の定義	1195
テストのためのリソース設定の定義	1199
テストのパラメータの定義	
テストのための環境設定の定義	
テストのための回復シナリオ設定の定義	
第42 草:実行セッション中のテスト・オフションの設定	
実行セッション中のテスト・オフションの設定について	
テスト・オプションの設定	1222
テスト・オプションの取得	1224
テスト実行の制御	
テスト実行設定の追加と削除	

第 X 部: 高度なテスト機能を使用した作業(第 2 巻)

第 43 草:仮想オフシェクトの字省	1229
仮想オブジェクトの学習について	1229
仮想オブジェクトについて	1231
仮想オブジェクト・マネージャについて	1232
仮想オブジェクトの定義	1233
仮想オブジェクト定義の削除と無効化	1238
	1011
第44 早:回復シナリオの正義と使用	
回復シナリオの定我と使用について	
回復シナリオを使用するダイミングの決定	
回復シナリオの定我	1245
回復シナリオ・ワイサートについて	
回復シナリオの官理	1275
テストへの回復ンナリオの関連付け	
ノロクラムによる回復メカニスムの制御	
第 45 章 : QuickTest Script Editor を使った作業	
QuickTest Script Editor について	
[QuickTest Script Editor] ウィンドウについて	1289
[QuickTest Script Editor] ウィンドウのカスタマイズ	1290
[テスト]表示枠について	
[リソース] 表示枠について	1295
[リソース] 表示枠について表示領域について	
[リソース] 表示枠について 表示領域について テストを使った作業	
[リソース]表示枠について 表示領域について テストを使った作業 関数ライブラリを使った作業	1295 1298 1300 1304
[リソース] 表示枠について	1295 1298 1300 1304
[リソース] 表示枠について 表示領域について テストを使った作業 関数ライブラリを使った作業	1295 1298 1300 1304 1311
[リソース] 表示枠について 表示領域について テストを使った作業	1295 1298 1300 1304 1311 1312
[リソース] 表示枠について 表示領域について テストを使った作業	1295 1298 1300 1304 1311 1312 1313
[リソース] 表示枠について 表示領域について テストを使った作業 関数ライブラリを使った作業	1295 1298 1300 1304 1311 1312 1313
 [リソース]表示枠について 表示領域について テストを使った作業 関数ライブラリを使った作業 第46章:QuickTest操作の自動化について QuickTest 操作の自動化について QuickTest オートメーション・スクリプトを使用する条件 オートメーション・スクリプトの設計と実行に使用する プログラミング言語と開発環境の選択 	1295 1298 1300 1304 1311 1312 1313 1314
 [リソース]表示枠について 表示領域について テストを使った作業 関数ライブラリを使った作業 第46章:QuickTest操作の自動化について QuickTest 操作の自動化について QuickTest オートメーション・スクリプトを使用する条件 オートメーション・スクリプトの設計と実行に使用する プログラミング言語と開発環境の選択 QuickTest オートメーション・スクリプトの基本要素の学習 	1295 1298 1300 1304 1311 1312 1313 1314 1314
 [リソース]表示枠について 表示領域について テストを使った作業 関数ライブラリを使った作業 第46章:QuickTest操作のすートメーション QuickTest操作の自動化について QuickTestオートメーション・スクリプトを使用する条件 オートメーション・スクリプトの設計と実行に使用する プログラミング言語と開発環境の選択 QuickTestオートメーション・スクリプトの基本要素の学習… オートメーション・スクリプトの生成 	1295 1298 1300 1304 1311 1312 1313 1314 1316 1317

第 XI部:その他の HP 製品を使用した作業(第2巻)

第 47 章 : Quality Center を使用した作業	1321
Quality Center を使用した作業について	1322
Quality Center との接続と切断	1323
QuickTest の Quality Center との統合	1332
Quality Center プロジェクトへのテストの保存	1333
Quality Center プロジェクトからテストを開く	1334
テンプレート・テストを使用した作業	1339
Quality Center プロジェクトに格納されているテストの	
QuickTest からの実行	1346
QuickTest でのテストのバージョン管理	1348
Quality Center テストの実行に関する設定	1359
笠 49 音 , Pusinasa Brasses Testing た佐田」 た佐孝	1267
第40 早. Business Flocess Testing を使用した作業	1367
Business Process Testing での得知について	1368
Business Process Testing のテスト方法について	1372
	1072
第 49 章 : WinRunner を使用した作業	1379
WinRunner を使用した作業について	1379
WinRunner テストの呼び出し	1380
WinRunner 関数の呼び出し	1384
第 50 章 : HP のパフォーマンス・テストおよび	
Business Availability Center 製品を使用した作業	1389
HP のパフォーマンス・テストおよび	
Business Availability Center 製品を使用した作業について	1390
QuickTest のパフォーマンス・テストおよび	
Business Availability Center の使用	1391
LoadRunner または Business Process Monitor で使用する	
QuickTest テストの設計	1392
LoadRunner または Business Process Monitor での	
テストの挿入と実行	1393
トランザクションの測定	1395
サイレント・テスト・ランナーの使用	1400

第 XII 部:付録(第2巻)

付録 A: FAQ(よくある質問)	1407
テストの作成	1407
エキスパート・ビューでのプログラミング	1409
動的なコンテンツを使った作業	1411
Web に関する高度な問題	1413
標準 Windows 環境	1415
テストの保守	1417
ローカライズされたアプリケーションのテスト	1419
QuickTest のパフォーマンスの向上	1420
付録 B: カスタムのプロセス・ガイダンス・パッケージの作成	1425
プロセス・ガイダンス・パッケージについて	1425
パッケージ設定ファイルについて	1426
データ・ファイルの作成	1428
QuickTest でのカスタムのプロセス・ガイダンス・パッケージの	
インストール	1429
索引	I-1

本書について

QuickTest Professional ユーザーズ・ガイドへようこそ。本ガイドでは、 QuickTest を使用してアプリケーションをテストする方法について説明します。 テストの作成・デバッグ・実行の方法と、テスト・プロセス中に検出された不 具合の報告の方法を順を追って説明します。

本章では、次の項目について説明します。

- ▶ 本ガイドの構成(xx ページ)
- ➤ 対象読者(xxiiページ)
- ▶ QuickTest Professional オンライン・ドキュメント (xxiii ページ)
- ➤ その他のオンライン・リソース (xxv ページ)
- ▶ 表記規則 (xxvi ページ)

ようこそ

本ガイドの構成

QuickTest Professional の印刷版は2つの巻で構成されています。QuickTest Professional のインストールに含まれている本ガイドの PDF 版およびコンテキ スト・センシティブ・ヘルプ版では、両巻の情報が1つのファイルに納められ ています。

本ガイドは、以下の部で構成されています。

第1巻

第1部 QuickTest Professional の概要

QuickTestの概要と、テスト・プロセスの主要な手順を説明します。

第 || 部 テスト・オブジェクトを使用した作業

テスト・オブジェクト・モデルを紹介し、QuickTest によってアプリケーション のオブジェクトが識別される方法について説明します。オブジェクトの使用方 法,オブジェクトの識別の設定方法,およびスマート認識定義の作成方法につ いて説明します。また,オブジェクト・リポジトリを管理,マージ,および比 較する方法についても説明します。

第Ⅲ部 テストの設計

テストを計画および作成する方法,アクションを使用する方法について説明し ます。

第 Ⅳ 部 テストの拡張

チェックポイント,パラメータ,および出力値を挿入し,正規表現を使用する 方法について説明します。

第2巻

第 V 部 関数およびその他のプログラミング・タスクの定義

QuickTest で,エキスパート・ビューを使用してテストを拡張する方法,エキスパート・ビューと関数ライブラリ・ウィンドウをカスタマイズする方法,およびユーザ定義関数と関数ライブラリを使用する方法について説明します。

第 VI 部 テストの実行と分析

テストを実行し、結果を分析する方法について説明します。

第 VII 部 テストの保守およびデバッグ

実行セッションを制御して、テスト・スクリプトおよび関数ライブラリ内の不 具合を特定し切り分ける方法について説明します。

第 VIII 部 QuickTest IDE を使用した作業

QuickTest のレイアウトを変更する方法,テスト・リソースを管理する方法,お よびプロセス・ガイダンスを使用する方法について説明します。

第 IX 部 QuickTest の設定

グローバルおよびローカルの QuickTest テスト・オプションを変更する方法, および実行セッション中にテスト・オプションを設定する方法について説明し ます。

第 X 部 高度なテスト機能を使用した作業

仮想オブジェクトおよび回復シナリオを使用する方法について説明します。また、より強力なスクリプトを作成するための複数のプログラミング・テクニック、および QuickTest の操作を自動化する方法について説明します。

第 XI 部 その他の HP 製品を使用した作業

テストを実行し、HP の企業向け Microsoft Windows アプリケーション機能テス ト・ツールである WinRunner でコンパイルされたモジュールの関数を呼び出す 方法について説明します。また、この部では、QuickTest を Business Process Testing と組み合わせて使用する方法や QuickTest を, HP の品質集中管理ソ リューションである Quality Center(以前の TestDirector)と連携させる方法につ いても説明します。さらに、この部では HP パフォーマンス・テスト製品やア プリケーション管理製品で使用できるように QuickTest テストを設計するため の注意事項についても説明します。

第 XII 部 付録

よくある質問に関する情報を提供し、カスタマイズされたプロセス・ガイダン ス・パッケージを作成する方法について説明します。

対象読者

本ガイドは、すべてのレベルの QuickTest Professional ユーザを対象としていま す。読者は、機能テストの概念およびプロセスについてある程度理解してい て、テストするアプリケーションの側面を把握している必要があります。

QuickTest Professional オンライン・ドキュメント

QuickTest Professional には、次のオンライン・ドキュメントがあります。

「最初にお読みください」では, QuickTest に関する最新ニュースと最新情報を 入手できます。[スタート] > [プログラム] > [QuickTest Professional] > [Readme] を選択してください。

『QuickTest Professional インストール・ガイド』では、QuickTest のインストールおよびセット・アップ方法を説明します。[ヘルプ] > [印刷用ドキュメント] > [HP QuickTest Professional インストール ガイド] を選択してください。

『QuickTest Professional チュートリアル』では QuickTest の基本スキルを学び アプリケーション向けのテストを設計する方法を説明します。**[ヘルプ]** > **[HP QuickTest Professional チュートリアル**]を選択します。

「**製品の機能紹介ムービー**」(英語版)では,選択された QuickTest 機能の使用 方法について,概要および手順ごとに説明します。[**ヘルプ**]> [**製品の機能 紹介ムービー**]を選択します。

「印刷用ドキュメント」では、Adobe の PDF (Portable Document Format) で全ド キュメントを表示します。オンライン文書は Adobe Reader を使って読んだり印 刷したりできます。Adobe Reader は、Adobe の Web サイト (<u>http://www.adobe.com/jp/</u>) からダウンロードできます。[ヘルプ] > [印刷用 ドキュメント] を選択します。

『QuickTest Professional ヘルプ』には次の文書が含まれています。

- ▶ 『QuickTest Professional の新情報』では、最新バージョンの QuickTest で サポートされている環境、新機能および強化された点について説明します。
- ▶ 『QuickTest Professional ユーザーズ・ガイド』では、QuickTest を使用し てアプリケーションをテストする方法を説明します。
- ▶ 『QuickTest Professional for Business Process Testing ユーザーズ・ガイ ド』では、QuickTest を使用して、ビジネス・プロセス・テスティングで使 用する資産の作成および管理を行う方法を手順ごとに説明します。
- ➤ 『QuickTest Professional アドイン・ガイド』では、サポート対象の環境で QuickTest のアドインを使って作業する方法について説明し、環境に固有の 情報をアドインごとに示しています。

- ▶ 『QuickTest Object Model Reference』(英語版)では、QuickTest テスト・ オブジェクトの説明、各オブジェクトに関連したメソッドおよびプロパティ の一覧、メソッドおよびプロパティの構文情報と使用例を示します。
- ▶ 『QuickTest Advanced References』(英語版)には、次の QuickTest COM および XML リファレンスに関するマニュアルが含まれています。
 - 『QuickTest Automation』(英語版)では、オートメーション・オブジェクト、メソッドとプロパティの構文情報、詳細な情報、および例を入手できます。また、QuickTestのオートメーション・スクリプトを記述する際の詳しい概要も含まれます。オートメーション・オブジェクト・モデルは、QuickTestのほぼすべての機能を制御することを可能にするオブジェクト、メソッド、プロパティを提供することによって、テスト管理の自動化を支援します。
 - 『QuickTest Test Results Schema』(英語版)では、テスト結果のカス タマイズに必要な情報を提供する、テスト結果の XML スキーマについ て説明します。
 - 『QuickTest Test Object Schema』(英語版)では、各種環境でテスト・ オブジェクトのサポートを拡張するのに必要な情報を提供する、テスト・オブジェクトの XML スキーマについて説明しています。
 - 『QuickTest Object Repository Schema』(英語版)では、XML にエクスポートされたオブジェクト・リポジトリ・ファイルを編集するのに必要な情報を提供する、オブジェクト・リポジトリの XML スキーマについて説明しています。
 - 『QuickTest Object Repository Automation』(英語版)では、QuickTest の外部から QuickTest のオブジェクト・リポジトリやその内容を操作する のに必要な情報を提供する、オブジェクト・リポジトリ・オートメー ション・オブジェクト・モデルについて説明します。
- ▶ 『VBScript Reference』(英語版) には、VBScript, Script Runtime, および Windows Script Host を含む Microsoft VBScript のマニュアルが含まれています。

QuickTest Professional ヘルプにアクセスするには、[ヘルプ] > [QuickTest Professional ヘルプ] を選択します。選択した QuickTest ウィンドウおよびダ イアログ・ボックスをクリックして F1 キーを押すことで QuickTest Professional ヘルプにアクセスすることもできます。また、QuickTest テスト・オブジェク ト、メソッド、またはプロパティの上にカーソルを置いて F1 キーを押すこと で、それらの説明、構文、および例を参照できます。

その他のオンライン・リソース

Mercury Tours サンプル Web サイト(英語版)は、本書で説明する多くの例の 基礎となります。この Web サイトの URL は <u>http://newtours.demoaut.com</u> です。 [スタート] > [プログラム] > [QuickTest Professional] > [Sample Applications] > [Mercury Tours Web サイト] を選択します。

ナレッジ・ベースは、Mercury カスタマー・サポート Web サイトの Knowledge Base ページを直接開きます。[**ヘルプ**] > [**ナレッジベース**] を選択します。 この Web サイトの URL は <u>http://support.openview.hp.com/</u>です。

カスタマー・サポート Web サイトからは,HP Software サポート Web サイトに アクセスします。このサイトでは,サポート要求を送信できます。また,英語 版のサイトでは,ナレッジ・ベースの参照,独自の項目の追加,ユーザ・ディ スカッション・フォーラムへの書き込みや検索,パッチや更新された文書のダ ウンロードなどを行うこともできます。[ヘルプ] > [カスタマー サポート Web サイト] を選択します。この Web サイトの URL は http://support.openview.hp.com/ です。

ほとんどのサポート・ページでは, HP Passport ユーザとして登録してログイン することを求められます。また,多くはサポート契約が必要です。

アクセス・レベルの詳細については、次を参照してください。 http://h20230.www2.hp.com/new_access_levels.jsp

HP Passport ユーザ ID の登録は,次の場所で行います。 http://h20229.www2.hp.com/passport-registration.html

フィードバックの送信:製品チームに,QuickTest Professional に関してオン ラインでフィードバック行うことができます。[ヘルプ]>[フィードバック の送信]を選択します。

HP Software Web サイトからは、HP Software Web サイトにアクセスします。 このサイトでは、HP Software 製品に関する最新情報を提供します。新しいソフ トウェアのリリース、セミナー、展示会、カスタマー・サポートなどの情報も 含まれています。[**ヘルプ**] > [**HP Software Web サイト**]を選択します。こ の Web サイトの URL は <u>www.hp.com/jp/hpsoftware</u> です。

表記規則

本書は、次の表記規則に従います。

- [UI 要素] および 関 アクションを実行するインタフェース要素の名前,ファ イル名やパス,および強調表示が必要なその他の項目は このスタイルで示します。例:[保存] ボタンをクリック します。"また、メソッド名または関数名も示します。
 例:wait_window ステートメントには次のパラメータが あります。
- **引数** メソッド,プロパティ,または関数の引数および書名は このスタイルで示します。例:『**HP ユーザーズ・ガイ ド**』を参照してください。
- <置換する値> 実際の値と置換するファイル・パスや URL アドレスの一 部は大括弧で囲みます。例:< MyProduct のインス トール・フォルダ> ¥bin)。
- Example
 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。例:編集ボックスに

 「Hello」と入力します。
- CTRL+C キーボードのキーはこの形式で示します。例:ENTER キーを押します。
- [] 半角の大括弧は、省略可能な引数を囲みます。
- { } 引数に割り当てる値の候補は、中括弧で囲んで示します。 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。プログラム例での3つの点は、プログラム行が意図的に削除されていることを示します。
 - 2 つの値のうちの1 つを選択しなければならない場合, これらの値を垂直バーで区切ります。

第V部

関数およびその他のプログラミング・タスクの 定義



エキスパート・ビューおよび関数ライブラリ・ ウィンドウを使用した作業

QuickTest のテストは, Microsoft のプログラミング言語である VBScript で記述 されたステートメントによって構成されています。エキスパート・ビューは, VBScript に慣れているテスト担当者がキーワード・ビューの代わりに使える機 能です。QuickTest では VBScript を使用して関数ライブラリも作成できます。

本章では、エキスパート・ビューでの作業方法を説明し、VBScript について簡 単に紹介し、いくつかの簡単なプログラミング・テクニックを使ってテストお よび関数ライブラリを拡張する方法を示します。

本章では、次の項目について説明します。

- ➤ エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業について(818ページ)
- ▶ エキスパート・ビューの理解と使用(819ページ)
- ▶ エキスパート・ビューおよび関数ライブラリ内での操作(831ページ)
- ➤ VBScript の基本的な構文の理解(841 ページ)
- ▶ プログラム的記述の使用(850ページ)
- ▶ プログラムによるアプリケーションの実行と終了(862ページ)
- ➤ コメント、フロー制御、そのほかの VBScript ステートメントの使用(863 ページ)
- ▶ テスト・オブジェクトのプロパティ値の取得と設定(872ページ)
- 実行環境オブジェクトのプロパティおよびメソッドへのアクセス(873ページ)
- ▶ DOS コマンドの実行(875ページ)
- ▶ Windows API を使用したテストおよび関数ライブラリの拡張(875ページ)

▶ 実行セッション中に報告するステップの選択(879ページ)

エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業について

エキスパート・ビューでは、アクションを VBScript として表示できます。 VBScript に慣れていれば、プログラミングを通じてステートメントの追加と更 新を行い、テストおよび関数ライブラリを拡張できます。これにより、テスト の機能と柔軟性を向上させることができます。また、[関数ライブラリ]ウィ ンドウを使用して、関数ライブラリの作成や関数ライブラリを使った作業がで きます。

VBScript での作業の詳細については, QuickTest の [**ヘルプ**] メニューから VBScript に関するマニュアルを参照してください ([**ヘルプ**] > [QuickTest Professional **ヘルプ**] > [VBScript リファレンス])。

オブジェクトの操作や,アプリケーションからの情報を取得するステートメン トを追加できます。たとえば,オブジェクトが存在するかどうかを検査するス テップの追加や,メソッドの戻り値の取得ができます。

手作業でまたはステップ・ジェネレータを使用して、テストまたは関数ライブ ラリにステップを追加することができます。ステップ・ジェネレータの使用法 の詳細については、769ページ「ステップ・ジェネレータを使用したステップ の挿入」を参照してください。

エキスパート・ビューに表示されたテストまたは関数ライブラリはいつでも印 刷できます。印刷出力には追加の情報を含めることもできます。エキスパー ト・ビューでの印刷の詳細については、349ページ「テストの印刷」を参照し てください。関数ライブラリの印刷の詳細については、903ページ「関数ライ ブラリの印刷」を参照してください。

エキスパート・ビューの理解と使用

VBScript ステートメントの作業を行うには、キーワード・ビューの代わりに、 エキスパート・ビューでテストの作業を行うことができます。それぞれの ビューを切り替えることが可能です。ビューを切り替えるには、QuickTest ウィ ンドウのテスト表示枠の最下部にある[エキスパート ビュー] タブまたは [キーワードビュー] タブを選択します。

エキスパート・ビューには,キーワード・ビューと同じステップおよびオブ ジェクトが表示されますが,表示される形式が異なります。

- ▶ キーワード・ビューでは、各ステップの情報とともに、オブジェクトの階層が アイコン形式のテーブルとして表示されます。詳細については、第12章 「キーワード・ビューを使った作業」を参照してください。
- ➤ エキスパート・ビューでは、各ステップが VBScrip の行として表示されます。 オブジェクト・ベースのステップでは、VBScript のステートメントがオブジェ クト階層を定義します。

次の図は、同じオブジェクト階層をエキスパート・ビューで表示した場合と キーワード・ビューで表示した場合を示します。

🚓 Action1 💽							
1: Browser("Welcome: Mercury	/Tours").Pr	agei	("Welcom	e: Mercury	Tours").We	bEdit(<mark>"userName"</mark>).Set "mercury"
	iート ビュー						
項目	操作		値	創り当て	コメント	注釈	
✓ € Action1				817-2-0		12.03	
🚽 🗸 🏈 Welcome: Mercury Tours 🚽							
👻 🔂 Welcome: Mercury Tours 🗧]					
userName	Set		"mercury"			"userName" edit bo	
II I I I I I I I I I I I I I I I I I I	パートビュー	7					

エキスパート・ビューの VBScript の各行は、テストの各ステップに相当しま す。上の例は、ユーザがエディット・ボックスに「mercury」という名前を挿 入するテストのステップを表しています。ステップの階層では、サイト名、 ページ名、ページ内のオブジェクトのタイプと名前、オブジェクトに対して実 行されたメソッド名を確認できます。 次の表は、同じステップの各部がキーワード・ビューとエキスパート・ビュー のそれでどのように表現されるかを示します。

キーワード・ビュー	エキスパート・ ビュー	説明
項目	Browser ("Welcome:Merc ury Tours")	ブラウザ・テスト・オブジェ クトの名前は Welcome:Mercury Tours。
項目 👻 🚰 Welcome: Mercury Tours	Page ("Welcome:Merc ury Tours")	現在のページの名前は Welcome:Mercury Tours。
項目 userName	WebEdit ("userName")	オブジェクトのタイプは WebEdit。操作の対象となる エディット・ボックスの名前 は userName。
操作 Set	Set	対象エディット・ボックスで 実行されるメソッドは Set 。
值 "mercury"	"mercury"	[username] エディット・ ボックスに挿入される値は mercury。

エキスパート・ビューでは、オブジェクトの記述は、オブジェクトのタイプに 続いて、括弧内に表示されます。オブジェクトのリポジトリに格納されている すべてのオブジェクトにとって、オブジェクトの記述としては、オブジェクト の名前で十分です。次の例では、オブジェクトのタイプは Browser で、オブ ジェクトの名前は「Welcome: Mercury Tours」です。

Browser ("Welcome:Mercury Tours")

ヒント: テスト・オブジェクトおよびメソッドの名前は大文字小文字が区別されません。

オブジェクトの階層では、オブジェクトはピリオドで区切られます。次の例では、Browser と Page は同じ階層構造内の2つの別々のオブジェクトです。

Browser("Welcome:Mercury Tours").Page("Welcome:Mercury Tours")

オブジェクトに対して実行される操作(メソッド)は、常にステートメントの 末尾に、操作に関係する値が後ろに続く形式で表示されます。次の例では、 Set メソッドを使って、「userName」エディット・ボックスに「mercury」と いう文字列を挿入しています。

Browser("Welcome:Mercury Tours").Page("Welcome: Mercury Tours"). WebEdit("userName").Set "mercury"

QuickTest はアプリケーション内のオブジェクトに基づいてアプリケーションを 処理します。テストに追加したステップは、アプリケーション内のオブジェク トに対して実行される操作に対応します。

QuickTest 内のオブジェクトは、環境によって分けられます。QuickTest の標準 設定では、標準のWindows環境のオブジェクトがサポートされます。追加の環 境を使用するには、QuickTestを開いたときに、アドイン・マネージャで該当す る QuickTest アドインをロードします。

ほとんどのオブジェクトには、対応するメソッドがあります。たとえば、Back メソッドは Browser オブジェクトに関連付けられています。

オブジェクトと、それに関連付けられているメソッドとプロパティの完全なリ ストを参照するには、[ヘルプ] > [QuickTest Professional ヘルプ] を選択 し、[**目次**] タブで [QuickTest Professional Object Model Reference] を開 きます。

メソッドを使用して操作を行うステップを追加する方法の詳細については, 824ページ「エキスパート・ビューまたは関数ライブラリでのステートメント の生成」を参照してください。

VBScript を使用する方法の詳細については、841 ページ「VBScript の基本的な 構文の理解」を参照してください。

チェックポイント・ステートメントおよび出力ステートメントについて

QuickTest で、ページ、テキスト文字列、テーブル、およびそのほかのオブジェ クトを対象としたチェックポイントおよび出力値を作成できます。キーワー ド・ビューでチェックポイントまたは出力値を作成すると、QuickTest によっ て、対応する VBScript 行がエキスパート・ビューに作成されます。チェックポ イントの実行には Check メソッドが使用され、出力値ステップの実行には Output メソッドが使用されます。

たとえば,次のステートメントでは,QuickTest によって「New York」という 文字列が検査されます。

Browser("Mercury Tours").Page("Flight Confirmation").Check Checkpoint("New York")

キーワード・ビューでは、対応するステップが次のように表示されます。

	操作	値	注釈
🔁 Flight Confirmation: Mercury	Check	CheckPoint("New York")	"Flight Confirmation: Mercury" Web page

注:

- ▶ チェックポイントの詳細は、対応する [チェックポイントのプロパティ] ダイアログ・ボックスで設定され、検査対象のオブジェクトとともに保存されます。出力値の詳細は、対応する [出力値のプロパティ] ダイアログ・ボックスで設定され、値を出力する元となるオブジェクトとともに保存されます。エキスパート・ビューに表示されるステートメントは、格納されている情報への参照です。したがって、エキスパート・ビューに手作業でチェックポイント・ステートメントおよび出力値ステートメントを挿入することはできません。また、エキスパート・ビューから別のテストに Checkpoint ステートメントおよび Output ステートメントをコピーすることもできません。
- ▶ チェックポイントの挿入と変更の詳細については、第15章「チェックポイントについて」を参照してください。出力値の挿入と変更の詳細については、第23章「値の出力」を参照してください。

第26章・エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

パラメータ指定について

QuickTest を使って、テストの値をパラメータ化することにより、テストを拡張 できます。「パラメータ」とは、外部のデータ・ソースまたはジェネレータか ら値が割り当てられる変数です。

キーワード・ビューでパラメータを作成すると、対応する VBScript 行がエキスパート・ビューに作成されます。

たとえば、メソッド引数の値としてデータ・テーブル・パラメータとして定義 した場合、次の構文を使用してデータ・テーブルから値が取得されます。

Object_Hierarchy.Method DataTable (parameterID, sheetID)

項目	詳細
Object_Hierarchy	テスト・オブジェクトの階層定義。ピリオドで区切られた1つ 以上のオブジェクトの並びから成ります。
Method	パラメータ化されたオブジェクトを対象に QuickTest によって 実行されるメソッドの名前。
DataTable	データテーブルを表す予約済みオブジェクト。
parameterID	値の取得先となるデータ・テーブル内のカラムの名前。
sheetID	値が格納されているシートの名前。パラメータがグローバル・ パラメータである場合は、dtGlobalSheet が sheetID となります。

たとえば, Mercury Tours サイト用にテストを作成しているときに, 目的地として「San Fransisco」を選択するとします。エキスパート・ビューで, 次のステートメントがテストに挿入されます。

Browser("Welcome: Mercury").Page("Find a Flight:").WebList("toPort"). Select "San Francisco" ここで、目的地の値をパラメータ化し、データ・テーブル内に「**Destination**」 カラムを作成したとします。前のステートメントが次のように変更されます。

Browser("Welcome: Mercury").Page("Find a Flight:").WebList("toPort"). Select DataTable("Destination",dtGlobalSheet)

ここで,Select はメソッド名,DataTable はデータ・テーブルを表すオブジェ クト,Destination はデータ・テーブルのカラムの名前,dtGlobalSheet はデー タ・テーブルのシートの名前です。

キーワード・ビューでは、このステップが次のように表示されます。

👻 👏 Welcome: Mercury Tours			
Welcome: Mercury Tours			
👻 🎝 Find a Flight: Mercury			
	Select	DataTable("Destination", dtGlobalSheet)	"toPort" list からく'Destination' データテーブルナ

パラメータ値の使用方法と定義方法の詳細については,第22章「値のパラ メータ化」を参照してください。

エキスパート・ビューまたは関数ライブラリでのステートメントの生成

ステートメントを生成するには、次のいずれかの方法を使用します。

- ➤ ステップ・ジェネレータを使用して、メソッドおよび関数を使用するステップ を追加できます。詳細については、769ページ「ステップ・ジェネレータを使 用したステップの挿入」を参照してください。
- ➤ メソッドを使用して操作を行う VBScript ステートメントを手作業で挿入できます。QuickTest は、ステートメントで使用するテスト・オブジェクト、メソッド、プロパティ、またはコレクションを選択し、エキスパート・ビューまたは関数ライブラリでの入力に合わせて対応する構文が表示されるステートメント補完(IntelliSense)機能を備えています。詳細については、825ページ「オブジェクトを対象としたステートメントの生成」を参照してください。
- ▶ エキスパート・ビューまたは関数ライブラリで VBScript のキーワードを入力し 始めると、「VBScript 構文を自動的に拡張する」オプションが有効になってい れば、対応する構文またはブロックがスクリプトに追加されます。詳細につい ては、830ページ「VBScript 構文の自動補完」を参照してください。
第26章・エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

オブジェクトを対象としたステートメントの生成

エキスパート・ビューまたは関数ライブラリで入力すると, IntelliSense (QuickTest のステートメント補完機能)により,ステートメントで使用するテ スト・オブジェクト,メソッド,プロパティ,またはコレクションをドロップ ダウン・リストからの選択し,対応する構文を表示できます。

ヒント: テスト・オブジェクトのステートメントを生成しやすくするため,関数ライブラリ・ドキュメントでも以下のように IntelliSense がサポートされますが,関数に完全なオブジェクト階層ステートメントを含めることは一般にお勧めしません。関数は,ほかのオブジェクトでも使用できるように汎用的に作成することをお勧めします。

[ステートメントの自動補完を行う]オプションは標準で有効になっています。 このオプションは,[エディタオプション]ダイアログ・ボックスで設定と解 除ができます。詳細については,第27章「エキスパート・ビューおよび関数 ライブラリ・ウィンドウのカスタマイズ」を参照してください。

[ステートメントの自動補完を行う] オプションが有効になっていると、次の ようになります。

- ▶ Page(というように、オブジェクトに続いて開き括弧「(」を入力すると、その オブジェクトのタイプに一致する、オブジェクト・リポジトリ内の全テスト・ オブジェクトのリストが表示されます。オブジェクト・リポジトリに、当該タ イプに一致するオブジェクトが1つだけある場合には、そのオブジェクトの名 前が開き括弧の後ろに、引用符で囲まれた状態で自動的に挿入されます。
- ➤ ステートメントの中でテスト・オブジェクトの後にピリオドを入力すると、入 力したオブジェクトの後に追加できるテスト・オブジェクト、メソッド、プロ パティ、コレクションおよび登録済みの関数のリストが表示されます。
- ▶ メソッドまたはプロパティの名前を入力すると、使用可能なメソッドおよびプロパティのリストが表示されます。CTRL + SPACE キーを押すと、選択肢が1つしかなければキーワードが自動的に補完されます。選択肢が複数ある場合は、入力したテキストに一致する(アルファベット順で)最初のメソッドまたはプロパティが強調表示されます。

- ➤ メソッドまたはプロパティを入力すると、メソッドまたはプロパティの構文が、その必須引数および任意引数とともに表示されます。メソッドまたはプロパティを使用するステップを追加するとき、メソッドまたはプロパティの必須引数に対して値を定義する必要があります。
- ➤ CTRL+SPACE を押すと、追加が可能な対応するテスト・オブジェクト、メソッド、プロパティ、コレクション、VBScript 関数、ユーザ定義関数、VBScript 定数、およびユーティリティ・オブジェクトのリストが表示されます。このリストは、オブジェクト・リポジトリにまだ追加されていないオブジェクトを入力した場合でも表示されます。テストに関数が含まれている場合、またはテストが関数ライブラリと関連付けられている場合は、その関数もリストに表示されます。
- ➤ Object プロパティをステートメントの中で使用すると、オブジェクトのデータが ActiveScreen、または開いているアプリケーションから現在利用できる場合には、アプリケーションの任意の実行環境オブジェクトのネイティブ・メソッドが表示されます。Object プロパティの詳細については、873ページ「実行環境オブジェクトのプロパティおよびメソッドへのアクセス」を参照してください。

エキスパート・ビューまたは関数ライブラリでステートメントの自動補完機能 を使用してステートメントを生成するには,次の手順を実行します。

- 1 [ステートメントの自動補完を行う] オプションが選択されていることを確認 します([ツール] > [表示オプション] > [一般] タブ)。
- 2 次の手順のいずれかを実行します。
 - ▶ 関数ライブラリで作業している場合は、手順4に進みます。
 - ▶ エキスパート・ビューで作業をしている場合は、オブジェクトに続いて開き 括弧「(」を入力します。

	Action1
1:	Browser("Welcome: Mercury Tours").Page(
4)	

第26章・エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

オブジェクト・リポジトリに、当該タイプに一致するオブジェクトが1つだけある場合には、そのオブジェクトの名前が開き括弧の後ろに、引用符で囲まれた状態でQuickTestによって自動的に挿入されます。当該タイプのオブジェクトがオブジェクト・リポジトリに複数存在する場合には、QuickTestによってそれらがリストに表示されます。



- 3 リスト内のオブジェクトをダブルクリックするか、矢印キーを使ってオブジェクトを選択し、ENTERキーを押します。QuickTestによって、オブジェクトがステートメントに挿入されます。
- 4 次のいずれかを実行します。
 - ➤ エキスパート・ビューで作業をしている場合は、メソッドを実行する対象となるオブジェクトの後に、ピリオド(.)を入力します。



▶ 関数ライブラリで作業している場合は次の例のようにオブジェクトの階層全体を入力します。

Browser("Welcome: Mercury Tours").Page("Book a Flight: Mercury).WebEdit("username").

5 オブジェクト記述の後ろには、("username"). のようにピリオド(.) を入力し ます。QuickTest によって、オブジェクトに対して利用できるメソッドおよびプ ロパティのリストが表示されます。

	CaptureBitmap 📃	
S (CheckProperty	
S (ChildObjects	
S (Click	l
	Drag	l
S [Drop	
r 🗗	Exist	l
	FireEvent	
S (GetROProperty	
	GetTOProperties 🗾	

ヒント: ピリオドの後,またはメソッド名またはプロパティ名の入力を始めて から CTRL+SPACE キーを押すか, [**編集**] > [**詳細設定**] > [**単語入力候補**] を 選択します。入力したテキストに一致するメソッドまたはプロパティが1つの みの場合は,メソッド名またはプロパティ名が自動的に補完されます。入力し たテキストに一致するメソッドまたはプロパティが複数ある場合には,入力し たテキストに一致する (アルファベット順で)最初のメソッドまたはプロパ ティが強調表示されます。 第26章・エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

6 リスト内のメソッド、またはプロパティをダブルクリックするか、矢印キーを 使ってメソッドまたはプロパティを選択し、ENTER キーを押します。QuickTest によって、メソッドまたはプロパティがステートメントに挿入されます。次に 示すエキスパート・ビューの例のように、メソッドまたはプロパティに引数が 含まれている場合は、ツールチップにそのメソッドまたはプロパティの構文が 表示されます。



ステートメントの完了のツールチップ

上の例では, Set メソッドには Text という引数が1つあります。この引数名 は, ボックスの中に挿入するテキストを表します。

ヒント:また,引数を含む任意のメソッドまたは関数にカーソルを置き,CTRL + SHIFT + SPACE キーを押すか,[**編集**] > [**詳細**] > [**引数詳細**] を選択する と,その項目に対応するステートメント補完(引数の構文の)ツールチップが 表示されます。

7 表示される構文に従って、メソッドの後ろにメソッド引数を入力します。

*	Action1	
1:	Browser("Welcome: Mercu	ury Tours").Page("Welcome: Mercury Tours").WebEdit("userName"). Set "mercury"
14 F FI	<u> </u>	<u>01-1 Ea-</u> /

注:エキスパート・ビューでステップを追加した後,追加した新しいステップ をキーワード・ビューに表示することができます。エキスパート・ビューに追 加したステートメントに構文エラーがある場合,[キーワードビュー]を選択 したときにそれらのエラーが情報表示枠に表示されます。詳細については, 847ページ「VBScript構文エラーの処理方法」を参照してください。 QuickTest メソッドの詳細と例については,『**HP QuickTest Professional Object Model Reference**』(英語版)を参照してください。

VBScriptの構文の詳細については、841ページ「VBScriptの基本的な構文の理解」を参照してください。

VBScript 構文の自動補完

エキスパート・ビューまたは関数ライブラリで VBScript のキーワードを入力し 始めたとき、[VBScript 構文を自動的に拡張する] オプションが有効になって いれば、キーワードの最初の2文字が QuickTest によって自動的に認識され、 対応する VBScript 構文またはブロックがスクリプトに追加されます。たとえ ば、行頭で「if」という文字に続いてスペースを入力すると、自動的に次の構 文が入力されます。

If Then

End If

[VBScript 構文を自動的に拡張する] オプションは標準で有効になっていま す。このオプションは, [エディタオプション] ダイアログ・ボックスで設定 と解除ができます。詳細については, 882 ページ「エディタの動作のカスタマ イズ」を参照してください。

入力した2文字が複数のキーワードの最初の2文字に一致する場合,[キーワードの選択]ダイアログ・ボックスが表示されるので,使用したいキーワードを 選択できます。たとえば,「pr」という文字に続いてスペースを入力すると, private および property というキーワードを含んだ[キーワードの選択]ダイア ログ・ボックスが表示されます。



第 26 章 • エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

そこでリストからキーワードを選択して [**OK**] をクリックします。対応する VBScript 構文またはブロックがスクリプトに自動的に挿入されます。

VBScriptの構文の詳細については、841ページ「VBScriptの基本的な構文の理解」を参照してください。

エキスパート・ビューおよび関数ライブラリ内での操作

[移動] ダイアログ・ボックスまたはブックマークを使って、エキスパート・ ビューまたは関数ライブラリ内の特定の行に移動できます。また、エキスパー ト・ビューまたは関数ライブラリ内の特定の文字列を検索し、必要があれば、 その文字列を別の文字列で置換することが可能です。これらのオプションを使 えば、長いアクションや関数のいくつものセクションの間を移動するのが容易 になります。

注: テストが対象の場合,エキスパート・ビューにはアクションが1つのみ表示されます。この項で説明する操作機能は,テスト全体ではなく,現在選択されているアクションを対象とするものです。

[移動] ダイアログ・ボックスの使用

[移動] ダイアログ・ボックスを使用して、アクション内または関数ライブラ リ内の特定の行に移動できます。

ヒント:標準設定では、エキスパート・ビューおよび関数ライブラリに行番号 が表示されます。行番号が表示されない場合は、[**ツール**] > [表示オプショ ン] > [一般] タブの [行番号を表示する] オプションを選択します。エディ タ・オプションの詳細については、第 27 章「エキスパート・ビューおよび関 数ライブラリ・ウィンドウのカスタマイズ」を参照してください。

[移動] ダイアログ・ボックスを使ってエキスパート・ビューまたは関数ライ ブラリ内の特定の行に移動するには、次の手順を実行します。

- 1 [エキスパートビュー] タブをクリックするか, 関数ライブラリを有効にします。
- **2**[編集]>[移動]を選択します。[移動]ダイアログ・ボックスが表示されます。

移動		×
行番号(<u>L)</u> :	40	
ОК	キャンセル	<u>ر الا ال</u>

3 [行番号] ボックスに移動先の行番号を入力し, [OK] をクリックします。指 定した行の先頭にカーソルが移動します。

ブックマークの使用

ブックマークを使って、アクションまたは関数ライブラリの中の重要なセク ションにマークを付けることで、さまざまな部分の間をより簡単に移動できま す。テストの中では、ブックマークは特定のアクション内でのみ使用できま す。アクション間を移動するときには維持されず、テストまたは関数ライブラ リとともに保存されません。

ブックマークを割り当てると、エキスパート・ビューまたは関数ライブラリ内 で選択した行の左に、アイコンが付加されます。[ブックマーク]ダイアログ・ ボックスの [移動] ボタンを使用して、ブックマークの付いている行に移動で きます。 ブックマークの見え方はテストでも関数ライブラリでも同じです。次の例では テスト内のアクションに2つのブックマークを追加しています。



ブックマークを設定するには、次の手順を実行します。

- 1 [**エキスパート ビュー**] タブをクリックするか,関数ライブラリをアクティブ にします。
- 2 ブックマークを設定する対象となる行をクリックします。

3 [編集] > [ブックマーク] を選びます。[ブックマーク] ダイアログ・ボック スが表示されます。

ブックマーク	×
ブックマーク名:	
	追加(<u>A</u>)
名前 行 内容	削除(<u>D)</u>
	移動(<u>G</u>)
[]]]	ヘルプ

- 4 [**ブックマーク名**] フィールドに,一意の名前を入力し, [**追加**] をクリックし ます。ブックマークが,対象となる行の行番号とそのテキスト内容とともに [ブックマーク] ダイアログ・ボックスに追加されます。また,エキスパート・ ビューまたは関数ライブラリ内で選択した行の左に,ブックマーク・アイコン ↓が付加されます。
- 5 ブックマークを削除するには、対象ブックマークを選択し、[**削除**]をクリックします。

特定のブックマークに移動するには、次の手順を実行します。

- 1 [**エキスパート ビュー**] タブをクリックするか, 関数ライブラリをアクティブ にします。
- 2 [編集] > [ブックマーク] を選択します。[ブックマーク] ダイアログ・ボッ クスが表示されます。
- **3** リストからブックマークを選択して [移動] ボタンをクリックします。QuickTest が現在のアクションまたは関数ライブラリの該当する行に移動します。

テキスト文字列の検索

エキスパート・ビュー内の現在のアクションの中または関数ライブラリ内で, 検索する文字列を指定できます。また,ページ・チェックポイントの[HTML ソースの編集]ダイアログ・ボックスおよび[HTMLタグの編集]ダイアロ グ・ボックスのほか,[スクリプトに "With"を追加]ダイアログ・ボックスの 中の文字列も検索できます。リテラル・テキストを検索することも,正規表現 を使用した高度な検索を行うこともできます。また,ほかのオプションを使用 して検索結果の絞込みを行うことも可能です。

[スクリプトに"With"を追加]ダイアログ・ボックスの詳細については,797 ページ「テストに対する「With」ステートメントの生成」を参照してください。 ページ・チェックポイントの詳細については,『HP QuickTest Professional アド イン・ガイド』のページ・チェックポイントに関する項を参照してください。

文字列を検索するには、次の手順を実行します。

- 1 エキスパート・ビューまたは関数ライブラリで、次のいずれかを実行します。
- ▶ [検索] ボタンをクリックします。

βĄ

▶ [編集] > [検索] を選択します。

ヒント:エキスパート・ビューでは、次のいずれかも実行できます。

[**編集**] > [**詳細設定**] > [スクリプトに "With" を追加] を選択してから CTRL + F キーを押します。

[ページチェックポイントのプロパティ]ダイアログ・ボックスで,[HTML ソースの編集]または[HTMLタグの編集]を選択してから右クリックし,表 示されたダイアログ・ボックスで[検索]を選択します。

[検索] ダイアログ・ボックスが開きます。

検索		×
検索する文字列(E):	•) 次を検索(N)
 ○ 大文字と小文字を区別する(C) ○ 完全一致の単語を検索する(M) ○ 正規表現(B) ○ 先頭または末尾で折り返す(W) ○ 選択範囲のみに制限する(<u>T</u>) ○ 末尾にカーソルを置く(P) 	検索する方向 ① 上へ(U) ④ 下へ(D)	キャンセル ヘルプ(出)

- 2 [検索する文字列] ボックスに,検索する文字列を入力します。
- 3 指定する文字列の中で正規表現を使用したい場合には、矢印ボタン ♪ をクリックして正規表現を選択します。リストから正規表現を選択すると、その表現が [検索する文字列] ボックス内のカーソルの位置に自動的に挿入されます。詳細については、840ページ「[検索] および [置換] ダイアログ・ボックスにおける正規表現の使用」を参照してください。

第 26 章 • エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

- 4 次の任意のオプションを選択して検索結果の絞込みを行うことも可能です。
 - ▶ [大文字と小文字を区別する]:検索の際に大文字と小文字を区別します。 [大文字と小文字を区別する]を選択した場合、大文字小文字が、[検索する 文字列]ボックスに入力した文字列と正確に一致する対象のみが QuickTest によって検索されます。
 - ▶ [完全一致の単語を検索する]:単語の一部ではなく単語全体が一致する文字 列を検索します。
 - ▶ [正規表現]:指定した文字列が正規表現として処理されます。リストから正 規表現を選択した場合には、このオプションが自動的に選択されます。
 - ▶ [先頭または末尾で折り返す]:検索の方向に応じて、検索がアクション、ダイアログ・ボックス、または関数ライブラリの先頭または末尾に達したときに、それらの先頭または末尾から検索を続けます。
 - ▶ [選択範囲のみに制限する]: アクション、ダイアログ・ボックス、または関数ライブラリの中で選択されているテキストの範囲内に限定して検索を行います。
 - ▶ [末尾にカーソルを置く]:検索対象文字列が見つかったときに、その文字列 を強調表示して、文字列の末尾にカーソルを移動します。
- 5 アクション、ダイアログ・ボックス、または関数ライブラリ内の現在のカーソ ルの位置からどちらの方向に向かって検索を行うか指定します。[上へ]また は[下へ]のいずれかを選択できます。
- 6 現在のアクション、ダイアログ・ボックスの中、あるいはアクティブな関数ラ イブラリの中で、検索文字列の次の出現を強調表示するには、「次を検索」を クリックします。

テキスト文字列の置換

エキスパート・ビュー内の現在のアクションまたは現在の関数ライブラリの中 で検索する文字列,およびそれらの文字列を置き換えるための文字列を指定で きます。[HTML ソースの編集]ダイアログ・ボックスおよび[HTML タグの 編集]ダイアログ・ボックス内の文字列を検索して置換することも可能です。 リテラル・テキストを検索して置換することも,正規表現を使用した高度な処 理を行うこともできます。また,ほかのオプションを使用して検索と置換の処 理を詳細に設定することも可能です。

文字列を置換するには、次の手順を実行します。

- 1 エキスパート・ビューまたは関数ライブラリで、次のいずれかを実行します。
- ▶ [置換] ボタンをクリックします。

Â.

▶ [編集] > [置換] を選択します。

ヒント: (テストの場合のみ) [ページチェックポイントのプロパティ] ダイア ログ・ボックスで, [HTML ソースの編集] または [HTML タグの編集] を選 択してから右クリックし,表示されたダイアログ・ボックスで [置換] を選択 します。

[置換] ダイアログ・ボックスが表示されます。

置換	×
検索する文字列(E):	
	次を検索(<u>N</u>)
置換後の文字列(P):	
□ 大文字と小文字を区別する(C)	すべて置換(A)
□ 大文字と小文字の区別を保持する[V]	キャンセル
完全一致の単語を検索する(M) 🥫 検索する方向 ——	
□ 正規表現(E) C 上へ(U)	
□ 先頭または末尾で折り返す(W) ○ 下へ(D)	
□ 選択範囲のみに制限する(1)	
□ 末尾にカーソルを置く(P)	

- 2 [検索する文字列] ボックスに,検索する文字列を入力します。
- 3 [置換後の文字列] ボックスに,見つかったテキストを置き換える文字列を入 力します。
- 4 [検索する文字列] または [置換後の文字列] に指定する文字列の中で正規表現を使用したい場合には、矢印ボタン ▶をクリックして正規表現を選択します。リストから正規表現を選択すると、その表現が [検索する文字列] または[置換後の文字列] ボックス内のカーソルの位置に自動的に挿入されます。詳細については、840ページ「[検索] および [置換] ダイアログ・ボックスにおける正規表現の使用」を参照してください。

第 26 章 • エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

- 5 次の任意のオプションを選択して検索結果の絞込みを行うことも可能です。
 - ▶ [大文字と小文字を区別する]:検索の際に大文字と小文字を区別します。 [大文字と小文字を区別する]を選択した場合、大文字小文字が、[検索する 文字列]ボックスに入力した文字列と正確に一致する対象のみが QuickTest によって検索されます。
 - ▶ [大文字と小文字の区別を保持する]: [検索する文字列] に指定した文字列 について、全部小文字、全部大文字、先頭のみ大文字、大文字小文字入り混 じりのそれぞれを調べます。[置換後の文字列] に指定した文字列は、大文 字小文字が入り混じっている場合を除き、見つかった文字列と同じ大文字小 文字に変換されます。大文字小文字が入り混じっている場合、[置換後の文 字列] の文字列が変換されずにそのまま使用されます。
 - ▶ [完全一致の単語を検索する]:単語の一部ではなく単語全体が一致する文字 列を検索します。
 - ▶ [正規表現]:指定した文字列が正規表現として処理されます。リストから正 規表現を選択した場合には、このオプションが自動的に選択されます。
 - ▶ [先頭または末尾で折り返す]:検索の方向に応じて、検索がアクション、ダイアログ・ボックス、または関数ライブラリの先頭または末尾に達したときに、それらの先頭または末尾から検索を続けます。
 - ▶ [選択範囲のみに制限する]: アクション、ダイアログ・ボックス、または関数ライブラリの中で選択されているテキストの範囲内に限定して検索を行います。
 - ▶ [末尾にカーソルを置く]:検索対象文字列が見つかったときに、その文字列 を強調表示して、文字列の末尾にカーソルを移動します。
 - ▶ [検索する方向]:検索する方向を指定します。
 - ▶ [上へ]:現在のテキストからアクション、ダイアログ・ボックス、また は関数ライブラリの先頭までに限定して検索を行います。
 - ▶ [下へ]:現在のテキストからアクション、ダイアログ・ボックス、また は関数ライブラリの末尾までに限定して検索を行います。
- 6 現在のアクションまたはダイアログ・ボックスの中、あるいはアクティブな関数ライブラリの中で、検索テキスト文字列の次の出現を強調表示するには、 [次を検索]をクリックします。

7 強調表示されている文字列を [置換後の文字列] ボックスの文字列で置き換えるには [置換] をクリックします。現在のアクションまたはダイアログ・ボックス、あるいはアクティブな関数ライブラリにおいて、 [検索する文字列] ボックスに指定した文字列のすべての出現を [置換後の文字列] ボックスの文字列で置き換えるには、[すべて置換] をクリックします。

[検索] および [置換] ダイアログ・ボックスにおける正規表現の使用

[検索する文字列] および [置換後の文字列] の文字列の中で正規表現を使用 して検索を拡張できます。正規表現の概要ついては,757 ページ「正規表現の 使用について」を参照してください。[検索] および [置換] ダイアログ・ ボックスで使用できる表現と QuickTest のほかの場所で使用できる表現に違い があることに注意してください。

選択できる正規表現を表示するには、[検索]または[置換]ダイアログ・ ボックスの中で矢印ボタン♪をクリックします。



定義済みの正規表現のリストから選択できます。また,タグ付きの表現も使用 できます。文字列の検索に正規表現を使用するとき,見つかった文字列に応じ て表現を変えたい場合があります。 **第 26 章 •** エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

たとえば、(save¥:n)¥1 という正規表現を指定して検索をすると、save という 文字列の後に任意の数字があり、その直後に save と、先に見つかった任意の 数字があるという文字列が見つかります(つまり、たとえば save6save6 とい う文字列が一致し、save6save7 という文字列は一致しません)。

また、タグ付き表現を使用して、見つかった文字列の一部を置換文字列に取り 込むことができます。たとえば、save(¥:n)を検索して、open¥1 で置換するこ とができます。この場合、saveの後に任の数字がある文字列が検索され、そ れが open と、見つかった数字で置き換えられます。

検索文字列の中でタグ付き表現を示すための括弧「()」を挿入するには,正規 表現のリストから [表記のタグ付け] を選択します。

使用するタグ表現を、「¥」に1から9のいずれかのタグ・グループ番号が続い た形式で指定するには、[タグ付け表記の一致]を選択してから対応するタグ・ グループ番号を選択します(タグ付き表現の番号を調べるには、検索文字列の 開き括弧「(」を数えます。最初の(最も左側にある)タグ付き表現は「¥1」で 最後のは「¥9」です)。

VBScript の基本的な構文の理解

VBScript は簡単に覚えられる強力なスクリプト言語です。VBScript を使用すれば、これまでにプログラミングの経験がなくても、オブジェクトを対象とした 簡単な作業から複雑な作業まで実行できます。

この項では、VBScript ステートメントを使用して QuickTest テストまたは関数 ライブラリを拡張するための基本的なガイドラインをいくつか示します。 VBScript の使用方法の詳細については、QuickTest の [**ヘルプ**] メニュー([**ヘ** $\boldsymbol{\mu}$ **プ**] > [**QuickTest Professional ヘルプ**] > [**VBScript リファレンス**])か ら VBScript のドキュメントを参照してください。

VBScript の各ステートメントには個別の構文規則があります。これらの規則に 従わないと、問題のあるステップを実行したときにエラーが生成されます。加 えて、エキスパート・ビューからキーワード・ビューに切り替えるときに、情 報表示枠のドキュメント内に構文エラーがあれば、QuickTest によってそれらが リスト表示されます。構文エラーを修正してなくさない限り、キーワード・ ビューに切り替えることはできません。詳細については、847ページ 「VBScript 構文エラーの処理方法」を参照してください。

2

ヒント:現在のドキュメントの構文は、[**構文チェック**] ボタンをクリックす るか、[**ツール**] > [**構文チェック**] を選択することにより、いつでも確認で きます。テストが開いている場合は、すべてのアクションの構文が確認されま す。関数ライブラリが開いている場合は、ライブラリ・スクリプトの構文が確 認されます。

エキスパート・ビューまたは関数ライブラリで作業をするときは、次に示す VBScriptの一般構文規則とガイドラインに留意してください。

▶ 大文字と小文字の区別:標準では VBScript は変数,オブジェクト,メソッドの 名前,および定数などの大文字と小文字を区別しません。

たとえば, 次の2つのステートメントは VBScript においては同じです。

Browser("Mercury").Page("Find a Flight:").WebList("toDay").Select "31" browser("mercury").page("find a flight:").weblist("today").select "31"

▶ テキスト文字列:値をテキスト文字列として入力するとき、文字列の前後に引用符を追加しなければなりません。たとえば、先に示したスクリプト行では、Webサイト、Webページ、およびエディット・ボックスの名前はすべて引用符で囲まれたテキスト文字列です。

31の値が引用符で囲まれているのは、それが数値ではなく、数字を表すテキス ト文字列だからです。

次の例では、プロパティ名(最初の引数)のみがテキスト文字列で、引用符で 囲まれています。2番目の引数(プロパティの値)は変数なので引用符で囲ま れていません。3番目の引数(タイムアウト値を指定)は数値なので、これも 引用符が不要です。

Browser("Mercury").Page("Find a Flight:").WaitProperty("items count", Total_Items, 2000)

- ▶ 変数:文字列,整数,配列,オブジェクトを格納するための変数を指定できます。変数を使用することで、スクリプトが読みやすくなり柔軟性が高くなります。詳細については、843ページ「変数の使用」を参照してください。
- ▶ 括弧:望む結果を得てエラーを避けるには、ステートメントの中で括弧を正しく使用することが重要です。詳細については、844ページ「括弧の使用」を参照してください。

第 26 章 • エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

- ➤ インデント:スクリプトをステートメントの論理構造およびネストに合わせて インデントをしたりインデントを解除したりできます。詳細については,846 ページ「VBScript テキストの書式設定」を参照してください。
- ➤ コメント:ステートメントにコメントを追加するには、単一引用符(')を独立の行の先頭で使用するか、ステートメントの末尾で使用します。スクリプトをわかりやすくして保守しやすいように、可能な限りのコメントを追加することをお勧めします。詳細については、846ページ「VBScript テキストの書式設定」および864ページ「コメントの挿入」を参照してください。
- ▶ スペース:スペースを追加することでスクリプトをわかりやすくすることができます。これらのスペースは VBScript によって無視されます。

個々の VBScript を使用してテストまたは関数ライブラリを拡張する方法の詳細 については,863ページ「コメント,フロー制御,そのほかの VBScript ステー トメントの使用」を参照してください。

変数の使用

テストまたは関数ライブラリの中でテスト・オブジェクトや単純な値を格納す るための変数を指定することができます。テスト・オブジェクトを変数に格納 する場合,ほかのステートメントの中でオブジェクト階層全体を指定すること の代わりに変数を使用できます。変数をこのように使用すれば,ステートメン トが読みやすくなり,保守もしやすくなります。

オブジェクトを格納する変数を指定するには、Set ステートメントを次の構文 で使用します。

Set ObjectVar = ObjectHierarchy

次の例では、Set ステートメントを使用して UserEditBox 変数に、username エディット・ボックスの Browser > Page > WebEdit オブジェクト階層全体を格 納するよう指定しています。その後、UserEditBox 変数を対象に Set メソッド を使用して、username エディット・ボックスに John という値を入力してい ます。

Set UserEditBox = Browser("Mercury Tours").Page("Mercury Tours").

WebEdit("username")

UserEditBox.Set "John"

注:単純な値文字列や数字などを格納する変数の指定には Set ステートメント を使用しないでください。次の例は,単純な値のための変数を定義する方法を 示します。

MyVar = Browser("Mercury Tours").Page("Mercury Tours"). WebEdit("username").GetTOProperty("type")

Dim ステートメントを使用して、文字列、整数、配列など、ほかの型の変数を 宣言できます。このステートメントは必須ではありませんが、テストまたは関 数ライブラリの構造を強化するために使用できます。次の例では、Dim ステー トメントを使用して passengers 変数を宣言し、現在のアクションまたは関数 ライブラリのほかのステートメントの中で使用できるようにしています。

Dim passengers

```
passengers = Browser("Mercury Tours").Page("Find Flights").
```

WebEdit("numpassengers").GetROProperty("value")

括弧の使用

VBScript でプログラミングをするとき、ステートメント内での括弧「()」の使用・不使用に関する規則に従うことが重要です。

値を返すメソッドを呼び出し、返された値を使用する場合には、メソッドの引数を括弧で囲む必要があります。

たとえば、値を変数に返す場合、メソッドを If ステートメントの中で使用する 場合、あるいは、Call キーワードを使用してアクションまたは関数を呼び出す 場合などにメソッド引数を括弧で囲みます。チェックポイントの戻り値を取得 したい場合にも、チェックポイント名を括弧で囲む必要があります。

ヒント: テストまたは関数ライブラリの中でステップを実行しているときに **Expected end of statement** エラー・メッセージを受け取った場合には、ス テップのメソッドの引数を括弧で囲む必要があるかもしれません。 第26章・エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

次に、括弧を使用する場合の例および使用しない場合の例をいくつか示します。

次の例では、ChildItem メソッドが値を変数に返すため、メソッドを括弧で囲 む必要があります。

Set WebEditObj = Browser("Mercury Tours").Page("Method of Payment"). WebTable("FirstName").ChildItem (8, 2, "WebEdit", 0) WebEditObj.Set "Example"

次の例では、Call を使用しているため、メソッドの引数を括弧で囲む必要があります。

Call RunAction("BookFlight", onelteration)

または

Call MyFunction("Hello World")

•••

•••

次の例では、If ステートメントの中でメソッドを使用しているため、 WaitProperty メソッドの引数を括弧で囲む必要があります。

If Browser("index").Page("index").Link("All kind of"). WaitProperty("attribute/readyState", "complete", 4) Then Browser("index").Page("index").Link("All kind of").Click End If

次の例では、Check メソッドがチェックポイントの値を返すため、メソッドの 引数を括弧で囲む必要があります。

a = Browser("MyBrowser").Page("MyPage").Check (CheckPoint("MyProperty"))

次の例では、Click メソッドが値を返さないため、メソッドの引数を括弧で囲む 必要はありません。

Browser("Mercury Tours").Page("Method of Payment").WebTable("FirstName"). Click 3,4

VBScript テキストの書式設定

エキスパート・ビューまたは関数ライブラリで作業をするときは、コメントや インデントについて VBScript の慣例に従うことが重要です。

コメントを使用してスクリプトのセクションを説明するようにします。これにより読みやすさが向上し、テストおよび関数ライブラリの保守や更新が容易になります。詳細については、864ページ「コメントの挿入」を参照してください。

インデントはステートメントの論理構造およびネストを反映するために使用します。

➤ コメントの追加:ステートメントにコメントを追加するには、単一引用符(') を独立の行の先頭に追加するか、ステートメントの末尾に追加します。

ヒント:

- ➤ ステートメントをコメントにするには、ステートメントの任意の場所をク リックして、「コメントブロック」ボタンをクリックします。
 - ▶ 選択したテキストのブロックをコメントにするには、[コメント ブロック] ボタンをクリックするか、[編集] > [詳細設定] > [コメント ブロック] を選択します。ブロック内の各行の先頭には単一引用符が付きます。
- ➤ コメントの削除:ステートメントからコメントを削除するには、独立の行の先 頭またはステートメントの末尾の単一引用符(')を削除します。



ヒント:選択したテキストのブロックまたは行のコメントを解除するには, [ブロックのコメント解除]ボタンをクリックするか,[編集]>[詳細設定] >[ブロックのコメント解除]を選択します。

➤ ステートメントのインデント:ステートメントをインデントするには、テキストを選択して、[編集] > [詳細設定] > [インデント] を選択するか、TABキーを押します。882ページ「エディタの動作のカスタマイズ」で説明されているように、[エディタオプション] ダイアログ・ボックスで選択されているタブ間隔に従ってテキストがインデントされます。

注: [エディタオプション] ダイアログ・ボックスの [タブキーを押して選択 されたテキストでインデントを行う] チェック・ボックスが選択されている必 要があります。選択されていない場合, TAB キーを押すと選択したテキストが 削除されます。

➤ ステートメントのインデント解除:ステートメントのインデントを解除するには、[編集] > [詳細設定] > [インデント解除] を選択するか、ステートメントの先頭のスペースを削除します。

VBScriptの書式設定の詳細については、QuickTestの[**ヘルプ**] メニュー([**ヘ** ルプ] > [QuickTest Professional **ヘルプ**] > [VBScript リファレンス]) か ら VBScript のドキュメントを参照してください。

VBScript 構文エラーの処理方法

エキスパート・ビューで[キーワードビュー]タブを選択すると,更新された 情報が QuickTest によってキーワード・ビューに表示されます。新規または更 新された VBScript ステートメントに構文エラーがある場合,ステータス・バー の右側に「エラー」というテキストが赤で点滅し,ステータス・バーに,スク リプトの構文エラーの情報について情報表示枠を確認する必要があることを示 すエラー・メッセージが表示されます。QuickTest は、すべての構文エラーが修 正されるまで、キーワード・ビューにドキュメントを表示できません。

[VBScript リファレンス] には個々の VBScript エラーの説明が表示されます。
 詳細については、「ヘルプ] > [QuickTest Professional ヘルプ] >
 [VBScript リファレンス] > [VBScript] > [リファレンス] > [エラー] >
 [VBScript 構文エラー] を選択してください。



ヒント:

- ▶ 現在のドキュメントの構文は、「構文チェック」ボタンをクリックするか、 [ツール] > [構文チェック] を選択することで、いつでも確認できます。 テストが開いている場合は、すべてのアクションの構文が確認されます。関 数ライブラリが開いている場合は、ライブラリ・スクリプトの構文が確認されます。
- ▶ 『Microsoft VBScript 言語リファレンス』では、VBScript の構文エラーを次の ように定義しています。「ある VBScript ステートメントの構造が VBScript ス クリプト言語の1つ以上の文法規則に違反した結果として生じるエラー」。 VBScript での作業の詳細については、QuickTest の [ヘルプ] メニューから VBScript のリファレンスを参照してください([ヘルプ] > [QuickTest Professional ヘルプ] > [VBScript リファレンス])。

情報表示枠に、ドキュメントの中で見つかった構文エラーの一覧が表示される ので、各構文エラーの場所を調べて修正することができます。

情報				• ‡ X
詳細	項目	アクション	行	
 Expected end of statemer 	t RegExpression	Action1	2	
 Expected statement 	RegExpression	Action1	4	
 Expected statement 	RegExpression	Action1	6	
 Expected ')' 	RegExpression	Action1	7	
 Expected end of statemer 	t RegExpression	Action1	8	
 Expected ')' 	RegExpression	Action1	9	

情報表示枠には、各構文エラーについて次の情報が表示されます。

表示枠内の要素	詳細
[詳細]	構文エラーの詳細。たとえば,条件ブロックを lf ステートメントで 開始したけれども End lf ステートメントで終了しなかった場合, [詳細] には Expected 'End lf' と表示されます。
	注 :状況によっては,QuickTest がエラーを正確に特定できずに, 次のようにいくつかの候補を表示することがあります。 Expected 'End Sub', or 'End Function', or 'End Property' この場合,示された行のステートメントを調べてどちらが該当する かを明らかにします。
[項目]	問題のステートメントが含まれているテストまたは関数ライブラリ の名前。
[アクション]	問題のステートメントが含まれているアクションの名前。このカラ ムは、アプリケーション領域を通じてビジネス・コンポーネントに 関連付けられている関数ライブラリには適用されません。
[行]	構文エラーが含まれている行。行の番号は各アクションまたは関数 ライブラリの先頭から数えられます。

情報表示枠の使用

- ▶ 構文エラーの詳細の上にポインタを移動すると、現在不正である構文が表示されます。
- ▶ 特定の構文エラーを含む行に移動するには、情報表示枠内で構文エラーをダブ ルクリックします。
- ▶ 情報表示枠内のカラムのカラム・ヘッダをドラッグしてサイズを変更し、情報 を読みやすくすることができます。
- ▶ 情報表示枠内の詳細情報を昇順または降順でソートするには、カラム・ヘッダ をクリックします。
- ▶ 情報表示枠内のエラーで F1 キーを押すと、VBScript 構文エラーに関する情報 が表示されます。

プログラム的記述の使用

QuickTest は、アプリケーション内のオブジェクトを学習すると、適切なテス ト・オブジェクトをオブジェクト・リポジトリに追加します。オブジェクトが オブジェクト・リポジトリに追加されたら、エキスパート・ビューでステート メントを追加することで、そのオブジェクトに対して追加のメソッドを実行で きます。ステートメントを追加するには、通常、各オブジェクトの名前(大文 字小文字は区別されない)をそのオブジェクトの階層にオブジェクト記述とし て入力した後で、適切なメソッドを追加します。

たとえば,次に示すステートメントでは,「username」はエディット・ボック スの名前です。このエディット・ボックスは「Mercury Tours」という名前を持 つページ上にあり,このページは「Mercury Tours」という名前を持つブラウザ にあるものです。

Browser("Mercury Tours").Page("Mercury Tours").WebEdit("username")

オブジェクト・リポジトリ内の各オブジェクトは一意の名前を持っているの で、指定する必要があるのはこの名前だけです。実行セッションの実行中、 QuickTest はオブジェクト・リポジトリの中で名前と親オブジェクトに基づいて オブジェクトを検索し、格納されているテスト・オブジェクトの記述を使っ て、アプリケーション内のオブジェクトを識別します。

QuickTest に対して、オブジェクト・リポジトリまたはオブジェクト名を参照せずに、オブジェクトに対するメソッドを実行するように指示できます。これを行うためには、QuickTest に、メソッドの実行対象としたいオブジェクトを識別するために使えるプロパティと値のリストを提供します。

そのようなプログラム的記述は、オブジェクト・リポジトリに格納されていな いオブジェクトに対する操作を行う場合に、非常に便利なことがあります。プ ログラム的記述は、何らかの共通するプロパティを持つ複数のオブジェクトを 対象に同じ操作を行う場合や、実行セッション中に動的に決まる記述に適合す るプロパティを持つ1つのオブジェクトに対する操作を行う場合にも使えます。

[テスト結果]では、実行セッション中にプログラム的記述または ChildObjects メソッドを使ってテスト・オブジェクトが動的に作成されたことを示す大括弧 がテスト・オブジェクト名に付けられます。



たとえば、入力した人名情報に基づいて、雇用主のリストを表示し、リストから選択した雇用主に履歴書を送れるようにするWebサイトのテストをするものとしましょう。テストでは、リストに表示されたすべての雇用主を選択したいのに、テストを設計するときには、ページにいくつのチェック・ボックスが表示されるか分からず、もちろん各チェック・ボックスの正確なオブジェクト記述も知ることができません。こうした状況で、プログラム的記述を使うことでSet "ON"メソッドを、「HTML TAG = input、TYPE = check box」という記述に適合するすべてのオブジェクトを対象に実行することができます。

プログラム的記述には2つのタイプがあります。

- ▶ 静的: VBScript ステートメントの中に、オブジェクトを記述するプロパティと 値のセットを直接指定します。
- ▶ 動的:プロパティと値のコレクションを記述オブジェクトに追加してから、ス テートメントにその記述オブジェクトの名前を入力します。

オブジェクト記述に対する要求が基本的なものであれば,**静的**記述を使ってス テートメントにプログラム的記述を直接入力するほうが簡単でしょう。しか し,ほとんどの場合は**動的**記述を使ったほうが機能,効率,および柔軟性が向 上します。

ステートメントへのプログラム的記述の直接入力

テスト・ステートメントにオブジェクトを直接記述するには、オブジェクトの 論理名を指定する代わりに、オブジェクトを記述する property:=value のペアを 指定します。

一般的な構文は次のとおりです。

TestObject("PropertyName1:=PropertyValue1", "...",

"PropertyNameX:=PropertyValueX")

TestObjec: テスト・オブジェクト・クラスです。

PropertyName:=PropertyValue: テスト・オブジェクトのプロパティとその 値です。property:=valueの各ペアは,カンマと二重引用符で区切る必要があり ます。 実行セッション中に取得するプロパティ値に基づいてオブジェクトを検索する 場合には、プロパティ値として変数名を入力できます。例を次に示します。

MyVar="some text string"

Browser("Hello").Page("Hello").Webtable("table").Webedit("name:=" & MyVar)

注: QuickTest はプログラム的記述のプロパティ値をすべて正規表現として評価 します。したがって,正規表現において特別な意味を持つ文字(*,?,+など) を含んだ値を入力するには、¥(円記号)を使用して,その特殊文字をリテラ ルな文字として扱うように指示します。正規表現の詳細については,757ペー ジ「正規表現の使用について」を参照してください。 **第 26 章 •** エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

次に示すステートメントでは、Mercury Tours ページに、author という名前と、 3 というインデックスを持つ WebEdit テスト・オブジェクトを指定します。実 行セッション中、QuickTest は一致するプロパティ値を持つ WebEdit オブジェク トを検索し、「Mark Twain」というテキストを入力します。

Browser("Mercury Tours").Page("Mercury Tours").WebEdit("Name:=Author", "Index:=3").Set "Mark Twain"

注: テスト・オブジェクト階層の特定のポイントからプログラム的記述を使用 する場合には、同じステートメント内では、そのポイント以降は必ずプログラ ム的記述を使用する必要があります。プログラム的記述を使用して階層内のオ ブジェクトを指定した後に、オブジェクト・リポジトリでの名前を使用してテ スト・オブジェクトを指定すると、当該オブジェクトは QuickTest によって識 別されません。

たとえば, 次の例ではテスト・オブジェクト階層全体を通してプログラム的記 述を使用しているので, このステートメントは使用できます。

Browser("Title:=Mercury Tours").Page("Title:=Mercury Tours"). WebEdit("Name:=Author", "Index:=3").Set "Mark Twain"

次の例も、特定のポイントから(Page オブジェクト記述以降)プログラム的記 述を使用しているので、ステートメントを使用できます。

Browser("Mercury Tours").Page("Title:=Mercury Tours"). WebEdit("Name:=Author", "Index:=3").Set "Mark Twain"

しかし,次の例では,Browser および Page オブジェクトについてはプログラム 的記述を使用しているものの,WebEdit テスト・オブジェクトについてはオブ ジェクト・リポジトリでの名前を使おうとしているので,このステートメント は使用できません。

Browser("Title:=Mercury Tours").Page("Title:=Mercury Tours"). WebEdit("Author").Set "Mark Twain"

QuickTest は、WebEdit オブジェクトをその名前を使用して特定しようとします が、親オブジェクトがプログラム的記述を使用して指定されているため、リポ ジトリの中で当該オブジェクトを見つけることができません。

テスト・オブジェクトの使用方法の詳細については,第4章「テスト・オブ ジェクトを使用した作業」を参照してください。 1回のテストまたは1つの関数ライブラリで同じプログラム的記述を複数回使 用するために、作成したオブジェクトを変数に割り当てることができます。

たとえは、次のように入力する代わりに、

Window("Text:=Myfile.txt - Notepad").Move 50, 50

Window("Text:=Myfile.txt - Notepad").WinEdit("AttachedText:=Find what:"). Set "hello"

Window("Text:=Myfile.txt - Notepad").WinButton("Caption:=Find next").Click

次のように入力できます。

Set MyWin = Window("Text:=Myfile.txt - Notepad") MyWin.Move 50, 50 MyWin.WinEdit("AttachedText:=Find what:").Set "hello" MyWin.WinButton("Caption:=Find next").Click

さらに別の方法として、With ステートメントを使うこともできます。

With Window("Text:=Myfile.txt - Notepad") .Move 50, 50 .WinEdit("AttachedText:=Find what:").Set "hello" .WinButton("Caption:=Find next").Click End With

With ステートメントの詳細については,870ページ「With ステートメント」を 参照してください。 第26章・エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

プログラム的記述のための Description オブジェクトの使用

Description オブジェクトを使用して、Property オブジェクトのセットを格納した Properties コレクションを返すことができます。Property オブジェクトは、プロ パティ名とプロパティ値で構成されます。返された Properties コレクションを、 ステートメント内でオブジェクト名の代わりに使用できます(各プロパティ・オ ブジェクトには、プロパティ名とプロパティ値のペアが含まれています)。

注:標準設定では, Properties コレクションに追加された Property オブジェクトの値はすべて正規表現として処理されます。したがって,正規表現において特別な意味を持つ文字(*,?,+など)を含んだ値を入力するには,¥(円記号)を使用して,その特殊文字をリテラルな文字として扱うように指示します。正規表現の詳細については,757ページ「正規表現の使用について」を参照してください。

コレクション内の特定の Property オブジェクトの値をリテラル値として指定す るには, RegularExpression プロパティを False に設定します。詳細について は, 『HP QuickTest Professional Object Model Reference』(英語版)の Utility の 項を参照してください。

Properties コレクションを作成するには、次の構文を使って **Description.Create** ステートメントを入力します。

Set MyDescription = Description.Create()

Properties オブジェクト(たとえば上の例の MyDescription)を作成すれば、実行セッション中に Properties オブジェクトのプロパティと値の追加、編集、削除、取得を行うステートメントを入力できます。これにより、オブジェクト記述にどのプロパティをいくつ含めるかを、実行セッション中に動的に決めることができます。

Properties コレクションに Property オブジェクト(プロパティと値)のセット を設定したら,テスト・ステートメントの中で,この Properties オブジェクト をオブジェクト名の代わりに指定できます。

たとえは、次のように入力する代わりに、

Window("Error").WinButton("text:=OK", "width:=50").Click

次のように入力できます。

Set MyDescription = Description.Create() MyDescription("text").Value = "OK" MyDescription("width").Value = 50 Window("Error").WinButton(MyDescription).Click

注: テスト・オブジェクト階層の特定のポイントからプログラム的記述を使用 する場合には、同じステートメント内では、そのポイント以降は必ずプログラ ム的記述を使用する必要があります。プログラム的記述を使用して階層内のオ ブジェクトを記述した後に、オブジェクト・リポジトリでの名前を使用してテ スト・オブジェクトを指定すると、当該オブジェクトは QuickTest によって識 別されません。

たとえば, Browser(Desc1).Page(Desc1).Link(desc3) ではテスト・オブジェク ト階層全体を通してプログラム的記述を使用しているので,このステートメン トは使用できます。

Browser("Index").Page(Desc1).Link(desc3) も、特定のポイントから(Page オブ ジェクト記述以降) プログラム的記述を使用しているので、このステートメン トは使用できます。

しかし、Browser(Desc1).Page(Desc1).Link("Example1") の場合,Browser および Page オブジェクトにプログラム的記述を使用する一方で,Link テスト・オブジェクトにオブジェクト・リポジトリでの名前を使用しているので,このステートメントは使用できません(QuickTest によって,オブジェクト名に基づいてLink オブジェクトが検索されますが,プログラム的記述を使用して親オブジェクトを指定しているため,リポジトリの中でオブジェクトが見つかりません)。

第 26 章•エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

Properties オブジェクトを使っているときには、プロパティや値の代わりに変数名を使うことで、実行セッション中に取得したプロパティや値に基づくオブジェクト記述を生成できます。

複数のオブジェクトでプログラム的記述を使いたい場合には、テスト内に複数の Properties オブジェクトを作成することもできます。

Description および **Properties** オブジェクト,および関連するメソッドの詳細に ついては,『**HP QuickTest Professional Object Model Reference**』(英語版)を参 照してください。

子オブジェクトの検索

ChildObjects メソッドを使って,指定された親オブジェクトの中にあるすべて のオブジェクト,あるいは特定のプログラム的記述に適合する子オブジェクト のみを取得できます。この子オブジェクトのサブセットを取得するには, Description オブジェクトを使って,まず記述オブジェクトを作成してから,子 オブジェクト・コレクションに適合するプロパティと値の集合を追加します。

注: ChildObjects 記述引数のためのプログラム的記述を作成するには, Description オブジェクトを使用します。property:=value 構文を使ってプログラ ム的記述を引数に直接入力することはできません。

記述オブジェクトの中に記述を「構築」したら,次の構文を使ってその記述に 適合する子オブジェクトを取得します。

Set MySubSet=TestObject.ChildObjects(MyDescription)

たとえば, 次のステートメントは QuickTest に対して, Itinerary Web ページ上の すべてのチェックボックスを選択するよう指示します。

Set MyDescription = Description.Create() MyDescription("html tag").Value = "INPUT" MyDescription("type").Value = "checkbox"

Set Checkboxes = Browser("Itinerary").Page("Itinerary").ChildObjects(MyDescription) NoOfChildObjs = Checkboxes.Count For Counter=0 to NoOfChildObjs-1 Checkboxes(Counter).Set "ON"

Next

[テスト結果]では、実行セッション中に ChildObjects メソッドまたはプログラ ム的記述を使ってテスト・オブジェクトが動的に作成されたことを示す大括弧 がテスト・オブジェクト名に付けられます。



ChildObjects メソッドの詳細については,『**HP QuickTest Professional Object Model Reference**』(英語版)を参照してください。

プログラム的記述での Index プロパティの使用

Index プロパティは、オブジェクトを一意に識別するためのテスト・オブジェ クト・プロパティとして役立つ場合があります。Index テスト・オブジェク ト・プロパティを使うと、オブジェクトはソース・コード内に出現する順序 (最初の出現は0)に基づいて識別されます。

Index プロパティ値は、各オブジェクトに固有の値です。つまり、インデック ス値3を使って WebEdit テスト・オブジェクトを記述すると、QuickTest によっ てページ内の4番目の WebEdit オブジェクトが検索されます。

これに対し、インデックス値3を使って WebElement オブジェクトを記述する と、QuickTest は、タイプには関係なく、そのページの4番目の Web オブジェ クトが検索されます。これは、WebElement オブジェクトがすべての Web オブ ジェクトに適用されるためです。 たとえば、次のオブジェクトが含まれるページがあるとします。

- ➤ Apple という名前の画像
- ► UserName という名前の画像
- ▶ UserName という名前の WebEdit オブジェクト
- ▶ Password という名前の画像
- ▶ Password という名前の WebEdit オブジェクト

次の記述は、前述のリストの3番目の項目を表します。ページ内で UserName という名前を持つ最初の WebEdit オブジェクトだからです。

WebEdit("Name:=UserName", "Index:=0")

一方,次の記述は,前述のリストの2番目の項目を表します。ページ内で UserName という名前を持つ最初の任意のタイプ(WebElement)のオブジェク トだからです。

WebElement("Name:=UserName", "Index:=0")

注:オブジェクトが1つのみの場合,index=0を指定しても取得されません。 この場合,Index プロパティをオブジェクト記述に含めるべきではありません。

プログラム的記述によるチェックの実行

プログラム的記述またはユーザ定義関数を使って,指定したオブジェクト・プ ロパティの実行時の値とそのプロパティの期待値を比較できます。

プログラム的記述によるチェックは、通常のチェックポイントを適用できない 場合(プロパティをチェックするオブジェクトがオブジェクト・リポジトリに 格納されていない場合など)に便利です。チェックの結果は、テスト結果レ ポートに書き込むことができます。

たとえば、Web ボタンの実行時の値をチェックするとします。GetROProperty または Exist メソッドを使って、オブジェクトの実行時の値を取得したり、実 行セッションのその時点でオブジェクトが存在するかどうかを確認したりでき ます。 以下の例は、プログラム的記述を使って **Continue** Web ボタンが実行セッション中に無効になっているかどうかをチェックする方法を示します。

GetROProperty メソッドを使用する方法

ActualDisabledVal =

Browser(micClass:="Browser").Page(micClass:="Page").WebButton (alt:=Continue").GetROProperty("disabled")

Exist メソッドを使用する方法

While Not Browser(micClass:="Browser").Page(micClass:="Page").WebButton (alt:=Continue").Exist(30) Wend

Report.ReportEvent ステートメントを追加することにより、チェック結果をテスト結果に送信するように QuickTest に指示できます。

If ActualDisabledVal = True Then

Reporter.ReportEvent micPass, "CheckContinueButton = PASS", "The

Continue button is disabled, as expected."

Else

Reporter.ReportEvent micFail, "CheckContinueButton = FAIL", "The Continue button is enabled, even though it should be disabled."
また,アプリケーションが期待どおりに機能しているかどうかをチェックする ユーザ定義関数を作成して使用することもできます。次の例は,オブジェクト が無効になっているかどうかをチェックし,オブジェクトが無効になっている 場合は **True** を返す関数を示します。

'@Description Checks whether the specified test object is disabled '@Documentation Check whether the <Test object name> <test object type> is enabled.

Public Function VerifyDisabled (obj)

Dim enable_property

'テスト・オブジェクトから disabled プロパティを取得する

enable_property = obj.GetROProperty("disabled")

If enable_property = 1 Then ' 値が True (1) ならオブジェクトは無効

Reporter.ReportEvent micPass, "VerifyDisabled Succeeded", "The test object is disabled, as expected."

VerifyDisabled = True

Else

Reporter.ReportEvent micFail, "VerifyDisabled Failed", "The test object is enabled, although it should be disabled."

VerifyDisabled = False

End If

End Function

注: GetROProperty メソッドの使用法の詳細については,874 ページ「実行環 境オブジェクト・プロパティの取得」を参照してください。While...Wend ス テートメントの使用法の詳細については,868 ページ「While...Wend ステート メント」を参照してください。特定のテスト・オブジェクト,メソッド,およ びプロパティの詳細については,『HP QuickTest Professional Object Model Reference』(英語版)を参照してください。

プログラムによるアプリケーションの実行と終了

QuickTest では, [記録と実行環境設定] ダイアログ・ボックスを使って, テストの実行開始時にブラウザまたはアプリケーションを起動するように指定したり, テスト対象アプリケーションを手作業で起動したりできます。また, テスト対象アプリケーションを起動または終了するステートメントをテストに挿入することもできます。

指定した場所から任意のアプリケーションを実行するには、これはテストに複数のアプリケーションが含まれていて、[記録と実行環境設定] ダイアログ・ ボックスで [開かれている Window ベースのアプリケーションすべてでテスト を記録して実行する] チェック・ボックスを選択している場合に特に便利で す。アプリケーションを指定して、サポートされている任意のパラメータを渡 したり、ファイル名を指定して、関連付けられているアプリケーションが起動 しそのファイルを開くようにできます。

ほとんどのアプリケーションは、Close メソッドを使って閉じることができま す。また、SystemUtil ステートメントを使ってアプリケーションを閉じること もできます。詳細については、『HP QuickTest Professional Object Model Reference』を参照してください。

たとえば,次に示すステートメントでは,type.txt というファイルを標準のテ キスト編集アプリケーション(「メモ帳」など)で開き,happy days と入力し てから,ショートカット・キーを使ってファイルを保存し,アプリケーション を終了しています。

SystemUtil.Run "C:¥type.txt", "","" Window("Text:=type.txt - Notepad").Type "happy days" Window("Text:=type.txt - Notepad").Type micAltDwn & "F" & micAltUp Window("Text:=type.txt - Notepad").Type micLShiftDwn & "S" & micLShiftUp Window("Text:=type.txt - Notepad").Close

注:

- ▶ 実行するアプリケーションを [記録と実行環境設定] ダイアログ・ボックス を使用して指定した場合, QuickTest はテストに SystemUtil.Run ステートメ ントを追加しません。
- ► InvokeApplication メソッドは、実行可能ファイルのみを開くことができ、 主に下位互換性を維持するために使用します。

詳細については,『**HP QuickTest Professional Object Model Reference**』(英語版)を参照してください。

コメント,フロー制御,そのほかの VBScript ステートメントの使用

QuickTest では、テストまたは関数ライブラリの論理フローを制御する条件文を 追加することで、テストまたは関数ライブラリに意思決定機能を組み込むこと ができます。さらに、QuickTest からテスト結果に送信するメッセージをテスト 内に定義できます。テストおよび関数ライブラリの読みやすさを向上させるた めに、コメントを追加することもできます。

キーワード・ビューでこれらのプログラミングの概念を使用する方法の詳細に ついては、第25章「プログラミング・ロジックを含むステップの追加」を参 照してください。

注:「VBScript リファレンス」([ヘルプ] > [QuickTest Professional ヘル プ] メニューから利用できます)には、VBScript, Script Runtime, および Windows Script Host などを含む Microsoft VBScript に関するマニュアルが含まれ ます。

コメントの挿入

コメントとは、アポストロフィ(')の後に書かれる、テスト・スクリプトの行または行の一部です。テストを実行しても、QuickTestではコメントは処理されません。読みやすさを向上させ、テストおよび関数ライブラリの更新を容易にするため、テスト・スクリプトのセクションを説明するコメントを使用することをお勧めします。

次の例では、コメントを使ってその下のステートメントの目的を説明しています。

'文字列 "mercury" を "username" エディット・ボックスに設定する Browser("Mercury Tours").Page("Mercury Tours").WebEdit("username"). Set "mercury"

標準設定では、コメントはエキスパート・ビューおよび関数ライブラリに緑色 で表示されます。コメントの表示は[エディタオプション]ダイアログ・ボッ クスでカスタマイズできます。詳細については、885ページ「エレメントの見 映えのカスタマイズ」を参照してください。

ヒント:

- ➤ テキストのブロックをコメントにするには、「編集]> [詳細設定]> [コ メント ブロック] を選択するか、「コメント ブロック] ボタンをクリックし ます。
- Ŕ
- ➤ コメントを解除するには、[編集] > [詳細設定] > [ブロックのコメント 解除] を選択するか、[ブロックのコメント解除] ボタンをクリックします。

注: コメント行を追加するには,VBScript の Rem ステートメントを使用する 方法もあります。詳細については,『Microsoft VBScript 言語リファレンス』 ([**ヘルプ**] > [QuickTest Professional ヘルプ] > [VBScript リファレンス] > [VBScript] を選択)を参照してください。 第26章・エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

計算の実行

数値演算子を使って、簡単な計算を実行するステートメントを作成できます。 たとえば、Web サイトで2つのテキスト・ボックスに表示された値を掛け算す るには、乗法演算子を使用します。VBScript では、次の数値演算子が使用でき ます。

演算子	詳細
+	加法
-	減法
-	否定(負の数値)
*	乗法
1	除法
۸	指数

次の例では,乗客1人あたりの荷物の重さの上限が100ポンドの場合の荷物の総重量を計算するために,乗法演算子を使用しています。

'GetROProperty メソッドを使って、エディット・ボックスから乗客数を取得 する

passenger = Browser ("Mercury_Tours").Page ("Find_Flights"). WebEdit("numPassengers").GetROProperty("value")

' 乗客数に 100 を乗じる

weight = passenger * 100

'メッセージ・ボックスに重量の上限を挿入する

msgbox(" この団体の荷物重量の上限は "& weight &" ポンドです。)

For...Next ステートメント

For...Next ループは、1 つ以上のステートメントを指定した回数だけ実行するように QuickTest に指示します。構文は次のとおりです。

For counter = start to end [Step step]

statement

Next

項目	詳細	
counter	反復の回数を表すカウンタとして使用する変数	
start	カウンタの開始値	
end	カウンタの終了値	
step	各ループの終わりに増分する値。 標準設定値 = 1。 省略可能	
statement	ループ中に実行する1つ以上のステートメント	

次の例では、QuickTest で For ステートメントを使って乗客数の階乗値を計算しています。

passengers = Browser("Mercury Tours").Page("Find Flights").

WebEdit("numPassengers").GetROProperty("value")

total = 1

For i=1 To passengers

total = total * i

NextMsgBox "!"& passengers & "=" & total

For...Each ステートメント

For...Each ループは,配列またはオブジェクト・コレクションの各要素に対して1つ以上のステートメントを実行するようにQuickTestに指示します。この 関数の構文は次のとおりです。

For Each item In array statement

Next

項目	詳細	
item	配列の要素を表す変数	
array	配列の名前	
statement	ループ中に実行する1つ以上のステートメント	

次の例では、For...Each ループを使用して配列の各要素の値を表示しています。

```
MyArray = Array("one","two","three","four","five")
```

For Each element In MyArray

msgbox element

Next

Do…Loop ステートメント

Do...Loop ステートメントは,条件が真である間,または条件が真になるまで, 1 つ以上のステートメントを実行するように QuickTest に指示します。この関数 の構文は次のとおりです。

Do [{while} {until} condition] statement

Loop

項目	詳細	
condition	満たされるべき条件	
statement	ループ中に実行する1つ以上のステートメント	

次の例では、QuickTest で Do...Loop ステートメントを使って乗客数の階乗値を 計算しています。

```
passengers = Browser("Mercury Tours").Page("Find Flights").
WebEdit("numPassengers").GetROProperty("value")
total = 1
i = 1
Do while i <= passengers
total = total * i
i = i + 1
Loop
MsgBox "!"& passengers & "=" & total
```

While...Wend ステートメント

While...Wend ステートメントは,条件が真である間,1つまたは複数のステートメントを実行するようにQuickTestに指示します。この関数の構文は次のとおりです。

While condition statement

Slatenne

Wend

項目	詳細	
condition	満たされるべき条件	
statement	ループ中に実行する1つ以上のステートメント	

次の例では、QuickTest で While ステートメントを使って、乗客数が 10 人未満 である間、ループを実行します。QuickTest によって、ループが 1 回実行される たびに、乗客数が 1 ずつ増えます。

```
passengers = Browser("Mercury Tours").Page("Find Flights").
```

```
WebEdit("numpassengers").GetROProperty("value")
```

```
While passengers < 10
```

```
passengers = passengers + 1
```

Wend

msgbox("この団体の人数は "& passengers & "人です。

If...Then...Else ステートメント

If...Then...Else ステートメントは、特定の条件に基づいて1つまたは複数のス テートメントを実行するように QuickTest に指示します。条件が満たされない 場合は、次の Elseif 条件または Else ステートメントが試されることになりま す。この関数の構文は次のとおりです。

If condition Then statement Elself condition2 Then statement Else statement End If

項目	詳細
condition	満たされるべき条件
statement	実行されるステートメント

次の例では、乗客数が4名未満の場合に、QuickTest によってブラウザが閉じられます。

```
passengers = Browser("Mercury Tours").Page("Find Flights").
```

WebEdit("numpassengers").GetROProperty("value")

If (passengers < 4) Then

Browser("Mercury Tours").Close

Else

Browser("Mercury Tours").Page("Find Flights").Image("continue").Click 69,5 End If 次の例では、lf, Elself, および Else ステートメントを使用して値が 1, 2, またはそれ以外の値に等しいかどうかを調べています。

value = 2
If value = 1 Then
 msgbox "one"
Elself value = 2 Then
 msgbox "two"
Else
 msgbox "not one or two"
End If

With ステートメント

With ステートメントで、同じ親階層を持つ連続するステートメントをグループ 化することによって、スクリプトが短くなり、読み書きと編集がしやすくなり ます。

注:With ステートメントをスクリプトに適用しても,実行セッション自体には 影響しません。エキスパート・ビューにどのように表示されるかにのみ影響し ます。

With ステートメントの構文は、次のとおりです。

With object statements End With

項目	詳細	
object	オブジェクトを返すオブジェクトまたは関数	
statements	オブジェクトに対して実行する1つ以上のステートメント	

たとえば、次のようなスクリプトがあったとします。

Window("Flight Reservation").WinComboBox("Fly From:").Select "London" Window("Flight Reservation").WinComboBox("Fly To:").Select "Los Angeles" Window("Flight Reservation").WinButton("FLIGHT").Click

Window("Flight Reservation").Dialog("Flights Table").WinList("From"). Select "19097 LON "

Window("Flight Reservation").Dialog("Flights Table").WinButton("OK").Click

これは、次のスクリプトで置き換えることができます。

With Window("Flight Reservation")
.WinComboBox("Fly From:").Select "London"
.WinComboBox("Fly To:").Select "Los Angeles"
.WinButton("FLIGHT").Click
With .Dialog("Flights Table")
.WinList("From").Select "19097 LON "
.WinButton("OK").Click
End With 'Dialog("Flights Table")
End With 'Dialog("Flights Table")

エキスパート・ビューで With ステートメントを入力しても,キーワード・ ビューにはまったく影響しません。

注:With ステートメントは手作業で入力することもできますが,With ステー トメントを自動生成させたり,既存のテストに基づいて With ステートメント を生成させたりもできます。詳細については,797ページ「テストに対する 「With」ステートメントの生成」を参照してください。

テスト・オブジェクトのプロパティ値の取得と設定

テスト・オブジェクトのプロパティは、各オブジェクトについて QuickTest に よって定義されている一連のプロパティです。テスト・オブジェクトのプロパ ティ値の設定と取得ができます。また、テスト・オブジェクト・プロパティの 値を実行環境オブジェクトから取得することもできます。

テストを実行すると、QuickTest では、テスト・オブジェクト・リポジトリに格 納されているテスト・オブジェクトの一時的なインスタンスが生成されます。 テストまたは関数ライブラリ内の GetTOProperty、GetTOProperties、および SetTOProperty メソッドを使って、テスト・オブジェクトのテスト・オブジェ クト・プロパティ値の設定と取得ができます。

GetTOProperty および GetTOProperties メソッドを使って, QuickTest がオブ ジェクトの識別に使う,特定のプロパティ値またはすべてのプロパティと値を 取得できます。

SetTOProperty メソッドを使って, QuickTest がオブジェクトを識別するために 使うプロパティ値を変更できます。

注: QuickTest は実行セッション中にテスト・オブジェクトの一時的なインスタ ンスを参照するため、SetTOProperty メソッドを使用して行ったすべての変更 は実行セッション中にのみ有効で、テスト・オブジェクト・リポジトリに格納 されている値には影響を与えません。

たとえば,次に示すステートメントは, [Submit] ボタンの名前値を「my button」に設定し,次に値「my button」を取得して ButtonName 変数に代入 しています。

Browser("QA Home Page").Page("QA Home Page"). WebButton("Submit").SetTOProperty "Name", "my button"

ButtonName=Browser("QA Home Page").Page("QA Home Page"). WebButton("Submit").GetTOProperty("Name")

テスト・オブジェクト・プロパティの現在の値をアプリケーションの実行環境 オブジェクトから取得するには,GetROPropertyメソッドを使います。 たとえば、次のようにして、実行セッション時にリンクのターゲット値を取得 できます。

link_href = Browser("HP Technologies").Page("HP Technologies"). Link("Jobs").GetROProperty("href")

ヒント: アプリケーション内にあるオブジェクトのテスト・オブジェクト・プ ロパティが不明の場合は、オブジェクト・スパイを使うことでそれらを表示で きます。オブジェクト・スパイの詳細については、第3章「テスト・オブジェ クト・モデルについて」を参照してください。

各オブジェクトでサポートされるテスト・オブジェクト・プロパティのリスト と説明,および GetROProperty, GetTOProperty, GetTOProperties, SetTOProperty メソッドの詳細については,『HP QuickTest Professional Object Model Reference』(英語版)を参照してください。

実行環境オブジェクトのプロパティおよびメソッドへのアクセス

特定のテスト・オブジェクトで利用可能なテスト・オブジェクト・メソッドま たはプロパティによって必要な機能が提供されない場合, Object プロパティを 使用することで, アプリケーションの任意の実行環境オブジェクトのネイティ ブ・メソッドおよびプロパティにアクセスできます。

ステートメント完了機能をオブジェクトのプロパティに対して使用すれば、オ ブジェクトに対して利用可能なネイティブ・メソッドとプロパティのリストを 表示できます。ステートメントの自動補完機能の詳細については、824ページ 「エキスパート・ビューまたは関数ライブラリでのステートメントの生成」を 参照してください。

ヒント:オブジェクトがWebオブジェクトである場合は、プログラム的記述の中で「attribute/ <プロパティ>」の形式を使用してネイティブのプロパティにもアクセスできます。詳細については、874ページ「Webオブジェクトのユーザ定義プロパティへのアクセス」を参照してください。

実行環境オブジェクト・プロパティの取得

Object プロパティを使用することで、任意の実行環境オブジェクトのネイティ ブ・プロパティにアクセスできます。たとえば、ActiveX カレンダーの内部 Day プロパティについて現在の値を取得するには、次のようにします。

Dim MyDay Set MyDay= Browser("index").Page("Untitled").ActiveX("MSCAL.Calendar.7").Object.Day

Object プロパティの詳細については, **『HP QuickTest Professional Object Model Reference』**(英語版)を参照してください。

実行環境オブジェクト・メソッドのアクティブ化

Object プロパティを使用することで、任意の実行環境オブジェクトの内部メ ソッドをアクティブ化できます。たとえば次のようにすることで、エディッ ト・ボックスのネイティブ focus メソッドを呼び出せます。

Dim MyWebEdit

Set MyWebEdit=Browser("Mercury Tours").Page("Mercury Tours"). WebEdit("username").Object

MyWebEdit.focus

Object プロパティの詳細については,『**HP** QuickTest Professional Object Model Reference』(英語版)を参照してください。

Web オブジェクトのユーザ定義プロパティへのアクセス

attribute/ **くプロパティ名>**表記を使って,Web オブジェクトのネイティブ・プ ロパティにアクセスし、プログラム的記述を使ってこれらのプロパティに基づ いてオブジェクトを識別することができます。

たとえば、ページ内の2か所に同じ企業ロゴ画像を持つ Web ページがあるとします。

 第 26 章 • エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業

ユーザ定義プロパティの LogoID を次のように記述に含めることで、クリック する画像をプログラム的記述を使って識別できます。

Browser("Mercury Tours").Page("Find Flights").Image("src:=logo.gif", "attribute/LogoID:=123").Click 68, 12

プログラム的記述の詳細については,850ページ「プログラム的記述の使用」 を参照してください。

DOS コマンドの実行

QuickTest テストまたは関数ライブラリの中で,VBScript Windows Scripting Host Shell オブジェクト (WSCript.shell) を使って,標準の DOS コマンドを実行でき ます。たとえば,次のステートメントを使うことにより,DOS コマンド・ウィ ンドウを開いてパスを C:¥ に変更し,DIR コマンドを実行できます。

Dim oShell Set oShell = CreateObject ("WSCript.shell") oShell.run "cmd /K CD C:¥ & Dir" Set oShell = Nothing

詳細については, 『Microsoft VBScript 言語リファレンス』([**ヘルプ**] > [QuickTest Professional **ヘルプ**] > [VBScript リファレンス] > [VBScript] を選択)を参照してください。

Windows API を使用したテストおよび関数ライブラリの拡張

Windows API を使用してテスト機能を拡張し、テストおよび関数ライブラリの 使い勝手と柔軟性を高めることができます。Windows オペレーティング・シス テムは、Windows での操作を制御、管理するために使用できる多数の関数を備 えています。これらの関数を使用することで追加機能を利用できます。

Windows API は次の URL から参照できる Microsoft MSDN Web サイトに文書資料があります。

http://msdn2.microsoft.com/en-us/library/Aa383750

個々の API 関数のリファレンスについては次を参照してください。 http://msdn2.microsoft.com/en-us/library/Aa383749

Windows API 関数を使用するには、次の手順を実行します。

- 1 MSDN において、テストまたは関数ライブラリの中で使用する関数を探します。
- 2 資料を読んで必要なパラメータと戻り値を把握します。
- 3 API 関数の場所を書き留めます。API 関数は Windows の DLL に含まれていま す。要求する関数が含まれている DLL の名前は通常,関数の説明の「Import Library」のセクションに記載してある名前と同じです。たとえば,説明文書の 中で User32.lib と記載してあれば,関数は User32.dll という名前の DLL に含 まれています。この DLL は通常, System32 ライブラリに含まれています。
- 4 QuickTest Extern オブジェクトを使用して外部関数を宣言します。詳細については、『HP QuickTest Professional Object Model Reference』(英語版)を参照してください。

次の例では, user32.dll にある GetForegroundWindow という関数への呼び 出しを宣言しています。

extern.declare micHwnd, "GetForegroundWindow", "user32.dll", "GetForegroundWindow"

5 hwnd = extern.GetForegroundWindows()のように, 宣言した関数を呼び出し, 必要な引数を渡します。

この例では,前面にあるウィンドウのハンドラが取得されます。このようにす ることで,前面のウィンドウがオブジェクト・リポジトリにない場合や,あら かじめ知ることができない場合(たとえば,動的なタイトルを持つウィンドウ など)にテストまたは関数ライブラリを拡張できます。このハンドルを,次の ように、ウィンドウのプログラム的記述の一部として使用することも可能です。

Window("HWND:="&hWnd).Close

状況によっては、あらかじめ定義されている定数値を関数の引数として使用し なければならない場合があります。そうした定数は、テストまたは関数内で定 義されていないため、呼び出し先の関数に渡すためにはそれらの値を調べる必 要があります。こうした定数の値は通常、使用する関数に対応するヘッダー・ ファイルに宣言されています。ヘッダー・ファイルに関する記述も、各関数の 説明文書の「Header」セクションに記載されています。使用しているコン ピュータに Microsoft Visual Studio がインストールされていれば、ヘッダー・ ファイルは通常、X:¥Program Files¥Microsoft Visual Studio¥VC98¥Include の下 に格納されています。 たとえば、GetWindow API 関数では、指定されたウィンドウとハンドルを取得す るウィンドウとの関係を表す数値を受け取ることが想定されています。MSDN の 説明文書には次の定数が記載されています。GW_CHILD, GW_ENABLEDPOPUP, GW_HWNDFIRST, GW_HWNDLAST, GW_HWNDNEXT, GW_HWNDPREV and GW_HWNDPREV.GetWindow の説明文書に記載されている WINUSER.H ファイ ルを開くと、次のフラグ値が設定されているのがわかります。

/*

* GetWindow() Constants

*/

#define GW_HWNDFIRST0 #define GW_HWNDLAST 1 #define GW_HWNDNEXT2 #define GW_HWNDPREV 3 #define GW_OWNER 4 #define GW_CHILD 5 #define GW_ENABLEDPOPUP 6 #define GW_MAX 6

例

次の例では、「メモ帳」アプリケーションの特定のメニュー項目を取得してい ます。

' 定数値 const MF BYPOSITION = 1024 'API 関数の宣言 Extern.Declare micHwnd,"GetMenu","user32.dll","GetMenu",micHwnd Extern.Declare micInteger,"GetMenuItemCount","user32.dll","GetMenuItemCount",micHwnd Extern.Declare micHwnd,"GetSubMenu","user32.dll","GetSubMenu",micHwnd,micInteger Extern.Declare micInteger, "GetMenuString", "user32.dll", "GetMenuString", micHwnd, micInteger, micString+micByRef,micInteger,micInteger ' Notepad.exe hwin = Window("Notepad").GetROProperty ("hwnd")' ウィンドウのハンドルを 取得 MsgBox hwin 'API 関数を使用する men hwnd = Extern.GetMenu(hwin)' ウィンドウのメイン・メニューのハンドル を取得 MsgBox men_hwnd item cnt = Extern.GetMenuItemCount(men hwnd) MsgBox item cnt hSubm = Extern.GetSubMenu(men_hwnd,0)

MsgBox hSubm

rc = Extern.GetMenuString(hSubm,0,value,64 ,MF_BYPOSITION)

MsgBox value

実行セッション中に報告するステップの選択

Report.Filter メソッドを使って、テスト結果にどのステップあるいはどのス テップのタイプを含めるかを決めることができます。ステートメントの後のス テップの報告の有効化と完全な無効化を行うことができます。あるいは、以降 の失敗したステップ、もしくは失敗および警告のステップだけをレポートに含 めるように指定することができます。また、Report.Filter メソッドを使って、 現在のレポート・モードを取得することもできます。

次のレポート・モードが使用できます。

モード	詳細
0 または rfEnableAll	すべてのイベントがテスト結果に表示されます。 標準設定値。
1 または rfEnableErrorsAndWarnings	ステータスが警告または失敗のイベントだけがテ スト結果に表示されます。
2 または rfEnableErrorsOnly	ステータスが失敗のイベントだけがテスト結果に 表示されます。
3 または rfDisableAll	テスト結果にはイベントは表示されません。

▶ 以降のステップの報告を行わないようにするには、次のステートメントを入力します。

Reporter.Filter = rfDisableAll

- ▶ 以降のステップの報告を再び行うようにするには、次のように入力します。
 Reporter.Filter = rfEnableAll
- ▶ 以降の失敗したステップだけをテスト結果に含めるには、次のように入力します。
 Reporter.Filter = rfEnableErrorsOnly

▶ 以降の失敗または警告のステップだけをテスト結果に含めるには、次のように入力します。

Reporter.Filter = rfEnableErrorsAndWarnings

▶ 現在のレポート・モードを取得するには、次のように入力します。

MyVar=Reporter.Filter

詳細については,**『HP QuickTest Professional Object Model Reference』**(英語版)を参照してください。



エキスパート・ビューおよび関数ライブラリ・ ウィンドウのカスタマイズ

[エキスパートビュー] でのテストの表示方法および, 関数ライブラリ・ウィンドウでの関数の表示方法をカスタマイズできます。変更したすべての内容は, [エキスパートビュー] およびすべての関数ライブラリ・ウィンドウにグローバルに適用されます。

本章では、次の項目について説明します。

- ▶ [エキスパートビュー]および関数ライブラリ・ウィンドウのカスタマイズ について(881ページ)
- ▶ エディタの動作のカスタマイズ(882ページ)
- ▶ エレメントの見映えのカスタマイズ(885ページ)
- ▶ 編集コマンドのカスタマイズ(888ページ)

[エキスパート ビュー] および関数ライブラリ・ウィンドウのカス タマイズについて

QuickTest には, 強力でカスタマイズ可能なエディタが用意されています。これ を使用して, [エキスパート ビュー] および関数ライブラリ・ウィンドウの各 部を変更できます。

[エディタオプション] ダイアログ・ボックスでは, [エキスパートビュー] および関数ライブラリ・ウィンドウでの,スクリプトや関数ライブラリの表示 方法を変更できます。また,スクリプトや関数ライブラリに表示されるテキス トのフォントや文字の大きさを変更したり,コメント,文字列,QuickTestの予 約語,演算子,数字などスクリプトのエレメントごとに色を変えたりできま す。たとえば,すべての文字列を赤で表示することもできます。 第27章・エキスパート・ビューおよび関数ライブラリ・ウィンドウのカスタマイズ

QuickTest には,カーソルの移動,文字の削除,クリップボードを使った情報の 切り取り,コピー,貼り付けを行うことができる,標準設定のキーボード・ ショートカットのリストがあります。これらのショートカットは,任意の ショートカットに置き換えることができます。たとえば,Line start コマンド を標準設定の HOME から ALT + HOME へ変更できます。

[印刷]ダイアログ・ボックスのオプションを使用して、スクリプトまたは関数ライブラリの印刷方法を変更することもできます。詳細については、349ページ「テストの印刷」および903ページ「関数ライブラリの印刷」を参照してください。

[エキスパートビュー]の使用方法の詳細については,第26章「エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業」を参照してください。関数ライブラリを使用した作業の詳細については,第28章「ユーザ 定義関数および関数ライブラリを使用した作業」を参照してください。

エディタの動作のカスタマイズ

[エキスパートビュー] および関数ライブラリ・ウィンドウにおけるスクリプト や関数ライブラリの表示方法をカスタマイズできます。たとえば、文字記号を 表示または非表示にしたり、行番号を表示するよう選択したりできます。たと えば、文字記号を表示または非表示にしたり、行番号を表示するよう選択した りできます。[エキスパートビュー]の使用方法の詳細については、第26章 「エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業」を 参照してください。関数ライブラリを使った作業の詳細については、第28章 「ユーザ定義関数および関数ライブラリを使用した作業」を参照してください。

エディタの動作をカスタマイズするには、次の手順を実行します。

- [エキスパートビュー]または関数ライブラリ・ウィンドウがアクティブなときに、[ツール]> [表示オプション]を選択します。[エディタオプション] ダイアログ・ボックスが開きます。
- **2**[**一般**] タブをクリックします。

エディタ オプション	×
一般 フォントと色 キーの連結	
▼ 行番号を表示する(B)	
▼ インデントを自動的に行う(A)	
▶ タブ キーを押して選択されたテキストでインデントを行う①	
▼ ステートメントの自動補完を行う⑤)	
□ 現在の行の周囲を囲う(R)	
🥅 すべての文字を表示する(出)	
□ ダブ文字(1)	
□ 改行文字(₩)	
□ スペース(Q)	
▶ VBScript 構文を自動的に拡張する②	
タフ スペース(12): 4 🔄 文字	

3 次のオプションから選択します。

オプション	詳細
[行番号を表示する]	スクリプトまたは関数の各行の左に行番号が表示され ます。
[インデントを自動的に行う]	インデントを設定した行の後に続く行が,自動的に前の 行と同じ位置から開始されます。キーボードの HOME キーを押すと,カーソルが左マージンに戻ります。

オプション	詳細
[タブ キーを押して選択さ れたテキストでインデント を行う]	TAB キーを押すと,選択されたテキストがインデント されます。このオプションが有効でない場合は,Tab キーを押すと,選択されたテキストがタブ文字1つで 置換されます。
[ステートメントの自動補 完を行う]	 このオプションが選択されている場合は、次を入力すると、それぞれ下記の内容が表示されます。 テスト・オブジェクトの後にドットを入力:入力したオブジェクトの後ろに追加できる使用可能なテスト・オブジェクトとメソッドのリストが表示されます。 オブジェクトの後で開き括弧(を入力:QuickTestによって、オブジェクト・リポジトリにこの種類のすべてのテスト・オブジェクトのリストが表示されます。オブジェクト・リポジトリに、当該タイプに一致するオブジェクトが1つだけある場合には、そのオブジェクトの名前が開き括弧の後ろに、引用符で囲まれた状態でQuickTestによって自動的に挿入されます。 メソッドを入力:特定の必須および任意の引数を含むメソッドの構文が表示されます。
	 ➤ Object プロパティ: ActiveScreen または開いているアプリケーションでオブジェクト・データが現在使用可能な場合は、アプリケーション内の任意の実行時オブジェクトのネイティブ・メソッドとプロパティが表示されます。 ステートメント補完 (IntelliSense) 機能の使用の詳細については、824ページ「エキスパート・ビューまたは関数ライブラリでのステートメントの生成」を参照してください。
[現在の行の周囲を囲う]	テスト内で現在カーソルがある行の周りに枠が表示さ れます。
[すべての文字を表示する]	すべてのタブ,改行,空白文字などの記号が表示され ます。また,対応するチェックボックスを選択/クリ アすれば,これらの文字の一部だけを表示させること もできます。

オプション	詳細
[VBScript 構文を自動的に 拡張する]	キーワードの最初の2文字を自動的に認識し,該当す るキーワードを入力するとそれに対応する VBScript 構文またはブロックをスクリプトに追加します。
	たとえば,エキスパート・ビューの行頭で「if」とい う文字に続いてスペースを入力すると,次の構文が自 動的に入力されます。
	If Then End If
[タブ文字を使用する]	キーボードの TAB キーを使用したときに、タブ文字が 挿入されます。このオプションを選択していなけれ ば、TAB キーを使用したときに代わりに指定された数 のスペース文字が挿入されます。

4 [OK] をクリックし、変更を適用してダイアログ・ボックスを閉じます。

エレメントの見映えのカスタマイズ

QuickTest のテストおよび関数ライブラリには、コメント、文字列、QuickTest やVBScriptの予約語、演算子、数字など、さまざまなエレメントが含まれてい ます。QuickTest のテストおよび関数ライブラリのそれぞれのエレメントは、異 なる色で表示できます。すべてのエレメントに使用されるフォントのスタイル や文字の大きさを指定することもできます。各エレメントに対して、独自にカ スタマイズした配色を作成できます。たとえば、すべてのコメントを青の文字 と黄色の背景で表示できます。 エレメントに対してフォントと色を設定するには、次の手順を実行します。

- [エキスパートビュー]または関数ライブラリ・ウィンドウがアクティブなときに、[ツール]> [表示オプション]を選択します。[エディタオプション] ダイアログ・ボックスが開きます。
- **2 [フォントと色]** タブをクリックします。

エディタ オプション				×
一般 フォントと色 キーの連結				
フォント名(E): Microsoft Sans Serif	サイズ(ℤ): 10	•		
構文の色	 			_
要索(E): コント 演算子 左亲白 数値 選択テキスト 標準設定 文字列 プレビュー	前景(<u>D</u>):	•	スタイル ○ 通常(W) ○ 太字(Q) ○ 科体(Q) ○ 下線(U)	
Private Sub Example() a = 12 + b sTemp = "Temp String" ' <i>Commented text</i> End Sub				
	ОК	キャンセ	<u>الہ ا</u>	,7

3 [フォント] 領域で、すべてのエレメントの表示に使用する [フォント名] と [サイズ] を選択します。このエディタでは Unicode フォントである Microsoft Sans Serif フォントが標準で使用されます。 **注**: Unicode 環境でテストを行う場合は, Unicode 対応のフォントを選択しなけ ればなりません。Unicode 対応のフォントを選択しないと, テストまたは関数ラ イブラリ内のエレメントが, [エキスパート ビュー] または関数ライブラリ・ ウィンドウに正しく表示されない場合があります。ただし, テストまたは関数 ライブラリは選択したフォントに関係なく, 指定前と変わりなく実行されます。 Unicode 対応でない環境で作業をしている場合は, Courier などの固定幅フォン トを使用して, 文字の揃えがよくなるようにします。

- **4**[**要素**] リストから,エレメントを選択します。
- 5 文字の色と背景の色を選択します。
- 6 エレメントのフォント・スタイルを選択します([通常],[太字],[斜体], [下線])変更を適用した場合の例が、ダイアログ・ボックス下部のプレビュー 表示枠に表示されます。
- 7 変更するエレメントごとに、手順4から6を繰り返します。
- **8**[**OK**] をクリックし、変更を適用してダイアログ・ボックスを閉じます。

編集コマンドのカスタマイズ

編集に使用する標準のキーボード・ショートカットをカスタマイズできます。 QuickTest には、カーソルの移動、文字の削除、クリップボードを使った情報の 切り取り、コピー、貼り付けを行うためのキーボード・ショートカットがあり ます。これらのショートカットは、任意のショートカットに置き換えることが できます。たとえば、Line end コマンドを標準設定の END から ALT + END へ 変更できます。

注:標準のQuickTestメニュー・ショートカット・キーは、ユーザが定義する キーの割り当てに優先します。たとえば、貼り付けコマンドのキーの組み合わ せをCTRL+Pと設定した場合でも、[印刷]ダイアログ・ボックスを開くための 標準のショートカット・キー([ファイル]> [印刷]オプションに対応)が 優先されます。QuickTestのメニュー・ショートカット・キーの一覧は、69 ページ「QuickTest コマンドの実行」を参照してください。

編集コマンドをカスタマイズするには、次の手順を実行します。

- [エキスパートビュー]または関数ライブラリ・ウィンドウがアクティブなときに、[ツール]> [表示オプション]を選択します。[エディタオプション] ダイアログ・ボックスが開きます。
- 2 [キーの連結] タブをクリックします。

エディタ オプション	×
一般「フォントと色「キーの連結」	
コマンド: 1 ページ下に移動 1 ページ下に移動 1 ページ上に移動 1 イトレに移動 1 イトレに移動 1 イテレに移動 1 イテレに移動 1 イテレに移動 1 イテレに移動 1 イテレに移動 1 イテレに移動 カーソル位置で削除 カーツルの方面 カージル前の削除 アーレの方見 ケレッジボードから貼り付け 1ピー レードキュメントの先頭 ドキュメントの先頭 ドキュメントの未尾 やの直し 右 石に戻す 行の先頭 行の先頭 行の先頭 行の先頭 ケの半語 小文字表記の弾語 切り取り 前の単語 挿入しまきの切り替え オッ字表記の弾根	使用するキー:

- 3 [コマンド] リストからコマンドを選択します。
- 4 [新規ショートカット キーを押す] ボックスをクリックして, 選択されている コマンドに使用するキーを押します。たとえば, CTRL+4 を入力するには, CTRL キーを押しながら数字の4 を押します。

5 [追加] をクリックします。

注:指定したキーの組み合わせがサポートされていない場合,あるいは別のコ マンドに対してすでに定義されている場合は,ショートカット・キー・ボック スの下にその旨を示すメッセージが表示されます。

- 6 ほかに追加するコマンドがあれば手順3~5を繰り返します。
- 7 リストからキー指定を削除するには、[コマンド] リスト内のコマンドを選択 して [使用するキー] リスト内のキー(またはキーの組み合わせ)を強調表示 し、[削除] をクリックします。
- 8 [OK] をクリックし、変更を適用してダイアログ・ボックスを閉じます。

第 28 章

ユーザ定義関数および関数ライブラリを使用し た作業

QuickTest テスト・オブジェクト・モデルでサポートされているテスト・オブ ジェクト,メソッド,および組み込み関数に加え,VBScript 関数,サブルーチ ン,モジュールなどが含まれる独自の関数ライブラリを定義して,その関数を テストから呼び出せます。

本章では、次の項目について説明します。

- ▶ ユーザ定義関数および関数ライブラリの使い方について(891ページ)
- ▶ 関数ライブラリの管理(893ページ)
- ▶ 関連付けられている関数ライブラリを使用した作業(905ページ)
- ▶ 関数定義ジェネレータの使用方法(909ページ)
- ▶ ユーザ定義関数のテスト・オブジェクト・メソッドとしての登録(924ページ)
- ▶ ユーザ定義関数の使い方のヒント(931ページ)
- ▶ テストからの外部定義された関数の実行(933ページ)

ユーザ定義関数および関数ライブラリの使い方について

テスト内で何度も使用する必要があるコード・セグメントがある場合は、ユー ザ定義関数を作成するとよい場合があります。ユーザ定義関数とは、何らかの 処理(またはプログラミングが必要な一連のステップ)を1つのキーワード (または操作)にカプセル化したものです。ユーザ定義関数を使用することに よって、テストの簡潔になり、設計、理解、保守が容易になります。QuickTest エンジニアが関連するキーワード(または操作)をアクションに挿入すること で、そのアクションからユーザ定義関数を呼び出すことができます。 ユーザ定義関数を、QuickTest テスト・オブジェクトのメソッドとして登録でき ます。登録したメソッドは、実行セッションの間だけ既存のテスト・オブジェ クト・メソッドの機能をオーバーライドしたり、テスト・オブジェクト・クラ スの新しいメソッドとして登録したりできます。ユーザ定義関数の登録の詳細 については、909ページ「関数定義ジェネレータの使用方法」および 924ペー ジ「ユーザ定義関数のテスト・オブジェクト・メソッドとしての登録」を参照 してください。

注: ユーザ定義関数を作成するときは,組み込みの関数と同じ名前(たとえば,GetLastError,MsgBox, Print など)を指定しないようにします。組み込み関数は,ユーザ定義関数に優先します。したがって,組み込み関数と同じ名前のユーザ定義関数を呼び出しても,組み込み関数が代わりに呼び出されます。組み込み関数のリストについては,ステップ・ジェネレータ([**挿入**] > [ステップジェネレータ])の「組み込み関数」リストを参照してください。

QuickTest では、ユーザ定義関数を定義し、関数ライブラリ(標準設定では.qfl ファイルとして保存)または直接テスト内のアクションに格納できます。関数 ライブラリとは、VBscript 関数、サブルーチン、モジュールなどが含まれる Visual Basic スクリプトのことです。また、QuickTest では、既存の関数ライブ ラリ(.vbs ファイルまたは.txt ファイルなど)を変更、デバッグすることもで きます。VBScript の使用の詳細については、847 ページ「VBScript 構文エラー の処理方法」および 841 ページ「VBScript の基本的な構文の理解」を参照して ください。

関数を関数ライブラリに格納し、その関数ライブラリをテストに関連付けれ ば、その関数ライブラリのパブリック関数をテストで呼び出すことができま す。詳細については、905ページ「関連付けられている関数ライブラリを使用 した作業」を参照してください。関連付けられている関数ライブラリに格納さ れている関数は、ステップ・ジェネレータ、およびキーワード・ビューの[**操 作**] カラムからアクセスできます。また、[エキスパートビュー]では手動で 入力できます。

関数をテスト・アクションに格納した場合、その関数はそのアクション内から のみ呼び出すことができます。ほかのアクションまたはテストから呼び出すこ とはできません。これは、特定のアクションの外側から関数を使用できないよ うにする場合に役に立ちます。

892

また、プライベート関数を定義して関数ライブラリに格納することができま す。プライベート関数は、同じ関数ライブラリ内のほかの関数からのみ呼び出 せる関数です。これは、パブリック関数の中でコード・セグメントを再利用す る場合に便利です。

関数は、手作業で定義することも、関数定義ジェネレータを使って定義するこ ともできます。関数定義ジェネレータは、関数の基本的な定義を自動的に作成 します。関数を手作業で定義する場合でも、関数定義ジェネレータを使用すれ ば、ヘッダ情報の追加、テスト・オブジェクトへの関数の登録、テスト・オブ ジェクトの標準メソッドとしての関数の設定を行うために必要な構文を表示で きます。詳細については、909ページ「関数定義ジェネレータの使用方法」を 参照してください。

関数ライブラリの管理

QuickTest で関数ライブラリを作成し、その関数をテストのアクションから呼び 出すことができます。関数ライブラリとは、VBscript 関数、サブルーチン、モ ジュールなどが含まれる独立した QuickTest 文書のことです。各関数ライブラリ は別々のウィンドウで開くため、同時に1つまたは複数の関数ライブラリを開 いて作業できます。関数ライブラリの編集が終了したら、関数ライブラリは閉 じて、QuickTest セッションは開いたままにしておけます。また、開いているす べての関数ライブラリを同時に閉じることもできます。

ユーザ定義関数を関数ライブラリで実装し、その関数ライブラリをテストに関 連付けることによって、ほかのユーザでも、テストに直接コードを追加せず に、複雑な操作(テスト・ステップへの if/then ステートメントとループの追加 や、ユーティリティ・オブジェクトを使った作業など)を実行する関数を選択 できるようになります。さらに、再利用可能な関数の実装および使用により、 時間とリソースを節約できます。

QuickTest には、あらゆる関数ライブラリを(外部エディタで作成された関数ラ イブラリでも)編集およびデバッグできるツールがあります。たとえば、 QuickTest では関数の構文をチェックできます。関数ライブラリ・ウィンドウに は、[エキスパート ビュー]が備えているものと同様の編集機能があります。 [エキスパート ビュー]で使用できるオプションの詳細については、第26章 「エキスパート・ビューおよび関数ライブラリ・ウィンドウを使用した作業」 を参照してください。 注:QuickTest では、テストを開くと、Quality Center プロジェクトに保存されて いる外部リソースのローカル・コピーが作成されます。したがって、別のユー ザが関数ライブラリなど Quality Center プロジェクトに保存されている外部リ ソースを変更した場合、または、外部エディタ(QuickTest ではないもの)を使 用してリソースを変更した場合、その変更はテストを閉じて再度開くまでテス トコンポーネントに実装されません。これに対して、関数ライブラリなどファ イル・システムに保存されている外部リソースに適用した変更は、ただちに実 装されます。これらのファイルは直接アクセスされ、テストを開いたときに ローカル・コピーとして保存されないためです。

関数ライブラリの作成

新しい関数ライブラリは、いつでも作成できます。

QuickTest で新しい関数ライブラリを作成するには、次の手順を実行します。 次の手順のいずれかを実行します。

- ▶ [ファイル] > [新規作成] > [関数ライブラリ] を選択します。
- ▶ [新規作成] ボタンの下向き矢印をクリックし, [関数ライブラリ] を選択します。

新しい関数ライブラリが開きます。

これで、関数ライブラリに内容を追加し、保存できます。関数ライブラリに内容を追加すると、[エキスパートビュー]の内容に適用されるのと同じ書式設定が適用されます。書式設定は、必要に応じて変更できます。詳細については、881ページ「エキスパート・ビューおよび関数ライブラリ・ウィンドウのカスタマイズ」参照してください。

関数ライブラリを開く

QuickTest では、すでに別の文書が開いていても、ファイル・システムまたは Quality Center プロジェクトに保存されている関数ライブラリを開くことができ ます。関数ライブラリは、当該ファイルに対する読み取り許可または読み書き の許可がある場合にのみ開けます。

関数ライブラリを編集モードで開くか読み取り専用モードで開くかを選択できます。

- ➤ 編集モード:関数ライブラリを表示,変更できます。あるコンピュータで関数 ライブラリが開いている間は、ほかのユーザは、そのファイルを読み取り専用 モードで表示できますが、変更はできません。
- ▶ 読み取り専用モード:関数ライブラリを表示することはできますが、変更はできません。標準設定では、現在ほかのコンピュータで開いている関数ライブラリを開くと、読み取り専用モードになります。関数ライブラリを表示する一方で、ほかのユーザが変更できるようにもしたい場合は、関数ライブラリを読み取り専用モードで開けます。

ヒント:自分の文書の関数から別の関数ライブラリの関数定義へ直接移動する こともできます。詳細については、900ページ「関数ライブラリの特定関数へ の移動」を参照してください。

既存の関数ライブラリを開くには、次の手順を実行します。

次の手順のいずれかを実行します。

- ▶ [ファイル] > [開く] > [関数ライブラリ] を選択します。
- ▶ [開く] ボタンの下向き矢印をクリックし, [関数ライブラリ] を選択します。

ヒント:

- ▶ 最近作成した、または開いた関数ライブラリは、[ファイル]メニューに表示される最近使用したファイルから選択できます。
- ▶ 開いているテストに関数ライブラリが関連付けられている場合, [リソース] > [関連付けのあるライブラリ]からその関数ライブラリを選択できます (Quality Center プロジェクトに格納されている関数ライブラリを選択する と, 関連付けられている関数ライブラリを開くために QuickTest が当該プロ ジェクトに接続されている必要があります)。

関数ライブラリを開く〕ダイアログ・ボックスが開きます。

関数ライブラリを開く		<u>? ×</u>
ファイルの場所型:	Contraction Tests	▼ 🖶 🖆 🎫
■ ライブラリ1.qfl		
ファイル名(<u>N</u>):	ライブラリ1.qfl	(IIII)
ファイルの種類(工):	関数ライブラリ (*.qfl;*.vbs;*.txt)	 キャンセル
□ 読み取り専用=	モードで開く	

注: Quality Center に接続している場合,表示されるダイアログ・ボックスは, ファイル・システムの場合の標準ダイアログ・ボックスと異なります。[**開く**] ダイアログ・ボックスで [**ファイル システム**] ボタンおよび [**Quality Center**] ボタンをクリックすることで,ダイアログ・ボックスの2つのバー ジョンを切り替えることができます。
ヒント: 関数ライブラリを読み取り専用モードで開くには, [読み取り専用 モードで開く] チェック・ボックスを選択します。

関数ライブラリを探して選択し, [**開く**] をクリックします。指定した関数ライ ブラリが新しいウィンドウで開きます。これで, 関数ライブラリの内容を表示, 変更できるようになります。詳細については, 900ページ「関数ライブラリの編 集」および 902ページ「関数ライブラリのデバッグ」を参照してください。

関数ライブラリの保存

QuickTest で関数ライブラリを作成または編集したら, Quality Center プロジェク トまたはファイル・システムに関数ライブラリを保存できます。

ヒント:

- ▶ 関数ライブラリに変更を加えると、関数ライブラリが保存されるまで、タイトル・バーにアスタリスク(*)が表示されます。
- ▶ 開いている文書をすべて保存するには、[ファイル] > [すべて保存] を選 択します。まだ保存されていない新規ファイルについては、保存先を指定す るように求められます。
- ▶ 複数の文書を保存するには、「ウィンドウ] > [ウィンドウ] を選択します。 [ウィンドウ] ダイアログ・ボックスで、保存する文書を選択し、「保存] ボ タンをクリックします。まだ保存されていない新規ファイルについては、保 存先を指定するように求められます。
- ▶ アクティブな関数ライブラリを別の名前または別のパスで保存するには、
 [ファイル]> [名前を付けて保存] を選択します。

関数ライブラリを保存するには、次の手順を実行します。

1 保存する対象となる関数ライブラリがアクティブな文書であることを確認します(フォーカスを対象に移動するには、対象関数ライブラリのタブをクリックします)。

H

2 次の手順のいずれかを実行します。

- ▶ [保存] ボタンをクリックします。
 - ▶ [**ファイル**] > [保存] を選択します。
 - ▶ 関数ライブラリ文書のタブを右クリックし、[保存]を選択します。

この関数ライブラリを以前に保存したことがあれば、変更された関数ライブラ リが保存されます。この関数ライブラリを初めて保存する場合は、[関数ライ ブラリを保存]ダイアログ・ボックスが開きます。

関数ライブラリを保存					? ×
保存する場所①:	🚞 Tests	•	* 🖻		
ੑੑਜ਼ ライブラリ1.qfl					
ファイル名(N):	ライブラリ2.qfl		(保存(5)	
ファイルの種類(工):	関数ライブラリ(*.qfl;*.vbs;*.txt)		•	キャンセル	

3 Quality Center プロジェクトまたはファイル・システムに関数ライブラリを保存 します。

注: Quality Center に接続している場合,表示されるダイアログ・ボックスは, ファイル・システムの場合の標準ダイアログ・ボックスと異なります。[保存] ダイアログ・ボックスで [ファイル システム] ボタンおよび [Quality Center] ボタンをクリックすることで,ダイアログ・ボックスの2つのバー ジョンを切り替えることができます。関数ライブラリを Quality Center プロジェ クトに保存するには, [テスト計画ツリー] ボックスで関数ライブラリの保存 先フォルダを選択します。[添付名] ボックスに関数ライブラリ名を入力し, [OK] をクリックします。 QuickTest 関数ライブラリは, 拡張子 .qfl を付けて保存されます(ただし, .vbs や .txt などの別の拡張子を指定した場合, または, 拡張子を完全に削除した場合を除く)。

開いている QuickTest 文書間でのフォーカスの移動

テストが開いている間,複数の関数ライブラリを開き,開いているすべての文 書間でフォーカスを移動することができます。

開いている QuickTest 文書間でフォーカスを移動するには,次の手順を実行し ます。

次の手順のいずれかを実行します。

- ▶ 文書表示枠で、必要な文書のタブをクリックします。
- **ヒント**: スペースが足りず,すべてのタブが表示されない場合は,文書表示枠の左右のスクロール矢印を使用して,必要な文書のタブを表示します。
 - ▶ キーボードの CTRL キーを押しながら TAB キーを押し、開いている文書間を切り替えます。
 - ▶ [**ウィンドウ**] メニューから必要な文書を選択します。
 - ▶ [ウィンドウ] > [ウィンドウ] を選択し, [ウィンドウ] ダイアログ・ボック スで必要な文書を選択して, [切り替え] ボタンをクリックします。

注: [リソース] > [関連付けのある関数ライブラリ] を選択して, リストか ら必要な関数ライブラリを選択することもできます。この操作では, テストに 関連付けられている, 閉じられている関数ライブラリも開きます。

関数ライブラリの特定関数への移動

関数の呼び出しを挿入した後,ソース文書内の当該関数の定義へ直接移動でき ます。関数の定義は、同じ文書(テストまたは関数ライブラリ),またはテス トに関連付けられている別の関数ライブラリに置くことができます。関数の定 義が含まれる文書がすでに開いている場合は、そのウィンドウがアクティブに なります(そこにフォーカスが移動します)。文書が閉じている場合は、その 文書が開きます。

関数の定義に移動するには、次の手順を実行します。

- 1 [エキスパート ビュー] または関数ライブラリで,該当する関数が含まれているステップをクリックします。
- 2 次の手順のいずれかを実行します。
 - ▶ [編集] > [詳細設定] > [関数定義に移動] を選択します。
 - ➤ ステップを右クリックして、ショートカット・メニューの [関数定義に移動] を選択します。

QuickTest 該当する文書がアクティブになり(関数の定義が別の関数ライブラリにある場合),関数の定義の先頭にカーソルが置かれます。

関数ライブラリの編集

[エキスパート ビュー]の QuickTest 編集機能を使用して, 関数ライブラリをい つでも編集できます。

文書間で関数(またはその一部分)のドラッグ・アンド・ドロップが可能です (それには, [最小化] ボタン (QuickTest ウィンドウの [最小化/最大化] ボタ ンの下にあります)をクリックすることで,タブ付きの文書を別々の文書表示 枠に分ける必要があります)。 手動で、またはステップ・ジェネレータを使用して、関数ライブラリにステッ プを追加できます。ステップ・ジェネレータを使用すれば、**予約オブジェクト** (ユーティリティ・オブジェクトなど、機能拡張のために提供されるオブジェ クト)、VBScript 関数 (MsgBox など)、ユーティリティ・ステートメント (Wait など)、同じ関数ライブラリに定義されているユーザ定義関数を含むス テップを追加できます。IntelliSense は、アクションに定義されているすべての 関数、または関連付けられている関数ライブラリに定義されているパブリック 関数で使用できます。

注: 関数ライブラリでは, IntelliSense を使用してテスト・オブジェクトの名前 またはコレクションを表示できません。これは, 関数ライブラリがオブジェク ト・リポジトリに接続されていないためです。



構文をチェックするように QuickTest に指定するには, [**構文チェック**] ボタン をクリックするか, [**ツール**] > [**構文チェック**] を選択します。

ヒント:

- ➤ VBScript の使用の詳細については、841 ページ「VBScript の基本的な構文の 理解」を参照してください。
- ▶ テストに関連付けられているすべての関数ライブラリの構文をチェックする には、[テストの設定]ダイアログ・ボックス([ファイル] > [設定])の [リソース]タブにある [構文チェック]ボタンをクリックします。詳細に ついては、1199ページ「テストのためのリソース設定の定義」を参照して ください。

読み取り専用の関数ライブラリの編集

関数ライブラリを読み取り専用モードで開き、その後でその関数ライブラリに 変更を加えることにした場合、その関数ライブラリを編集可能ファイルに変換 できます。ただし、ほかのユーザがその関数ライブラリをロックしていない場 合に限ります。関数ライブラリを開くときに使用できるオプションの詳細につ いては、895ページ「関数ライブラリを開く」を参照してください。

注:デバッグ・セッション中は,すべての文書(テストおよび関数ライブラリ など)が読み取り専用になります。デバッグ・セッション中に文書を編集する には,まず,デバッグ・セッションを停止する必要があります。

読み取り専用の関数ライブラリを編集するには,次の手順を実行します。

[ファイル]>[編集可能にする]を選択するか,[編集可能にする]ボタンを クリックします。これで,関数ライブラリを編集できます。

関数ライブラリのデバッグ

関数ライブラリをデバッグする前に、関数ライブラリをテストに関連付けて、 ライブラリ内の関数への呼び出しを少なくとも1つ挿入しておく必要がありま す。たとえば、[デバッグビューア]を使用して、関数ライブラリのオブジェク トまたは変数の現在の値を表示、設定、変更できます。関数(ユーザ定義関数 を含む)のステップ・イントゥ、ブレークポイントの設定、ブレークポイント での停止、式の表示などが可能です。デバッグは特定のステップから開始した り、特定のステップで一時停止するように指定したりできます。詳細について は、1037ページ「テストと関数ライブラリのデバッグ」を参照してください。

注: デバッグ・セッション中は, すべての文書が読み取り専用になり, 編集で きません。デバッグ・セッション中に文書を編集するには, まず, デバッグ・ セッションを停止する必要があります。



関数ライブラリの印刷

0

関数ライブラリは、いつでも印刷できます。印刷出力には追加の情報を含める こともできます。

関数ライブラリを印刷するには、次の手順を実行します。

1 [印刷] ボタンをクリックするか, [ファイル] > [印刷] を選択します。[印 刷] ダイアログ・ボックスが開きます。

印刷	
プリンタ: DocuPrint C2250	
□ 選択した部分(L)	設定(S)
🗐 ヘッダに文書名を含める(I)	
🥅 ヘッダに日付を含める(<u>D</u>)	
□ ページ番号(G)	キャンセル
🖂 🔟 🚍 行ごとに行番号を表示する	ヘルプ
部数(N): 1 🛨	

- 2 印刷オプションを指定します。
 - ▶ [プリンタ]:印刷ジョブの送信先となるプリンタが表示されます。プリンタ を変更するには、[設定]ボタンをクリックします。
 - ▶ [選択した部分]: 関数ライブラリ内の現在選択されている(強調表示されている) テキストのみ印刷されます。
 - ▶ [ヘッダに文書名を含める]:印刷出力の上部に関数ライブラリの名前が挿入 されます。
 - ▶ [ヘッダに日付を含める]:印刷出力の上部にその日の日付が挿入されます。 日付書式は Windows の地域のオプションに基づきます。
 - ▶ [ページ番号]:印刷出力の下部にページ番号が挿入されます(例: page 1 of 3)。
 - ► [X 行ごとに行番号を表示する]:指定どおりに、スクリプト行の左側に行番号が表示されます。
 - ▶ [部数]:ドキュメントを印刷する回数を指定します。

- 3 別のプリンタに印刷する場合やプリンタの設定を変更する場合は、[設定]を クリックして「プリンタの設定]ダイアログ・ボックスを表示します。
- 4 [印刷] をクリックすると, 選択内容に従って印刷されます。

関数ライブラリを閉じる

個々の関数ライブラリを閉じたり,複数の関数ライブラリが開いている場合 は、その一部または全部を同時に閉じたりすることができます。いずれかの関 数ライブラリが保存されていないと、保存するよう QuickTest に求められます。

個々の関数ライブラリを閉じるには、次の手順を実行します。

次の手順のいずれかを実行します。

- ▶ 保存する関数ライブラリがアクティブな文書であることを確認し(当該関数 ライブラリにフォーカスを移動するには、関数ライブラリのタブをクリック します)、[ファイル] > [閉じる]を選択します。
- ▶ 関数ライブラリ文書のタブを右クリックし, [閉じる] を選択します。
- ▶ 関数ライブラリ・ウィンドウの右上角にある [閉じる] ボタンをクリックします。
- ▶ [ウィンドウ] > [ウィンドウ] を選択します。[ウィンドウ] ダイアログ・ ボックスで、閉じる関数ライブラリが選択されていない場合は選択し、 [ウィンドウを閉じる] ボタンをクリックします。

複数の関数ライブラリを閉じるには、次の手順を実行します。

[**ウィンドウ**] > [**ウィンドウ**] を選択します。[ウィンドウ] ダイアログ・ ボックスで,閉じる関数ライブラリを選択し,[**ウィンドウを閉じる**] ボタン をクリックします。

開いているすべての関数ライブラリを閉じるには、次の手順を実行します。

[ファイル] > [全関数ライブラリを閉じる] を選択するか, [ウィンドウ] > [全関数ライブラリを閉じる] を選択します。

関連付けられている関数ライブラリを使用した作業

QuickTest では、関数、サブルーチン、モジュールなどが含まれる関数ライブラ リを作成し、そのファイルをテストに関連付けることができます。これによっ て、QuickTest エンジニアは、関連付けられている関数ライブラリ内のパブリッ ク関数やサブルーチンへの呼び出しを、当該テストに挿入することができます (関数ライブラリに格納されているパブリック関数は、関連付けられている任 意のテストから呼び出せるのに対し、プライベート関数は同じ関数ライブラリ 内からのみ呼び出せます)。

テストがステップ内で使用されている関数にアクセスできなくなると(たとえば、関連付けられている関数ライブラリからその関数が削除された場合など), キーワード・ビュー内でステップの横に 2 アイコンが表示されます。そのテ ストを実行すると、存在しない関数を使用しているステップに達したときにエ ラーが発生します。

注:標準の VBScript 構文で書かれたあらゆるテキスト・ファイルを関数ライブ ラリとして使えます。

[テストの設定] ダイアログ・ボックス([**ファイル**] > [**設定**] > [**リソー ス**] タブ)で、新規のすべてのテストに関連付ける標準の関数ライブラリを指 定できます。テストが作成されると、標準関数ライブラリのリストがテストに 統合されます。したがって、[テストの設定] ダイアログ・ボックスの標準関 数ライブラリ・リストに変更を加えても、既存のテストには影響しません。

既存のテストに関連付けられている関数ライブラリのリストは[テストの設 定]ダイアログ・ボックスで編集できます。詳細については,1199ページ「テ ストのためのリソース設定の定義」を参照してください。

注:

- ▶ 関連関数ライブラリの中にある関数のほかに、任意の関数ライブラリ(または VBScript ファイル)に含まれている関数を任意のアクションから ExecuteFile 関数を使って直接呼び出すことができます。関連関数ライブラリに ExecuteFile 関数を挿入することもできます。詳細については、933 ページ「テストからの外部定義された関数の実行」を参照してください。
- ExecuteFile ステートメントを使用して呼び出されるファイルおよび当該 ファイルに含まれている関数をデバッグすることはできません。また、 ExecuteFile ステートメントを含んでいるテストをデバッグする場合、実行 マーカが正しく表示されないことがあります。

Quality Center での関連付けられている関数ライブラリを使用した作業

関数ライブラリがファイル・システムまたは Quality Center プロジェクトに格納 されているかどうかに関係なく、テストに関数ライブラリを関連付けることが できます。ただし、ビジネス・プロセス・テストで関数ライブラリを使用する 計画がある場合は、関数ライブラリを Quality Center プロジェクトに保存する必 要があります。

Quality Center および関連関数ライブラリを使用するときには, [テストの設定] ダイアログ・ボックスの [リソース] タブで関連ファイルを指定する前に, 関 連関数ライブラリを Quality Center プロジェクトに添付ファイルとして保存する 必要があります。Quality Center プロジェクトには新規または既存の関数ライブ ラリを追加できます。

ファイル・システムから Quality Center プロジェクトに既存の関数ライブラリを 追加する場合,実際にはそのファイルのコピーがプロジェクトに追加されま す。したがって,これらの関数ライブラリのどちらか一方(ファイル・システ ム内または Quality Center プロジェクト内)を後で変更した場合,もう一方の関 数ライブラリは影響を受けません。

テストとの関数ライブラリの関連付け

[リソース] 表示枠または現在アクティブな関数ライブラリから,開いている テストに関数ライブラリを関連付けることができます。

また,関連付けられている関数リストを使用して,現在開いているテストに関 数ライブラリを関連付けることもできます。詳細については,908ページ「関 数ライブラリの関連付けの変更」を参照してください。

[リソース] 表示枠を使ってテストに関数ライブラリを関連付けるには, 次の 手順を実行します。

- 1 [リソース] 表示枠で, ツリー内の [**関連付けられた関数ライブラリ**] ノード を右クリックし, [**関数ライブラリの関連付け**]を選択します。[添付を開く] ダイアログ・ボックスが開きます。
- 2 関連付ける関数ライブラリを参照して選択します。
- 3 [**開く**] をクリックします。関数ライブラリがテストに関連付けられ、ツリー 内の [**関連付けられた関数ライブラリ**] ノードに関数ライブラリ・リンクとし て表示されます。

開いている関数ライブラリをテストに関連付けるには,次の手順を実行します。

- 1 関数ライブラリを関連付ける対象となるテストが QuickTest で開かれていることを確認します。
- 2 QuickTest で関数ライブラリを作成するか開きます(次の手順に進む前に、テストに関連付ける関数ライブラリがアクティブな文書であることを確認します。対象関数ライブラリにフォーカスを移動するには、関数ライブラリのタブをクリックします)。詳細については、893ページ「関数ライブラリの管理」を参照してください。
- 3 関数ライブラリを、添付ファイルとして Quality Center プロジェクト、またはファ イル・システムに保存します。詳細については、897ページ「関数ライブラリの 保存」を参照してください。
- 4 QuickTest の中で、「ファイル]>「ライブラリ'くFunction Library>'を'Test' に関連付ける]を選択するか、関数ライブラリの中で右クリックして「ライブ ラリ'くFunction Library>'を'Test'に関連付ける]を選択します。 QuickTest によって、開いているテストに関数ライブラリが関連付けられます。

関数ライブラリの関連付けの変更

テストに関連付けられている関数ライブラリのリストを変更できます。リスト に関数ライブラリを追加したり、リストから削除したりできます。また、関数 ライブラリの優先順位を変更することもできます。

テストへの関数ライブラリの関連付けを変更するには,次の手順を実行します。

- 1 [テストの設定] ダイアログ・ボックスで、[**リソース**] タブをクリックします。
- 2 関連付けられている関数ライブラリ・リストで、[関数ライブラリの追加]ボ タンをクリックします。QuickTest に参照ボタンが表示されます。このボタンを 使用して、ファイル・システム内の関数ライブラリを参照できます。Quality Center プロジェクトに接続している場合は、ファイル・パスに [QualityCenter] が追加され、Quality Center プロジェクトとファイル・システムのいずれかで関 数ライブラリを参照できます。

ただし, QuickTest が Quality Center のプロジェクト・フォルダを検索するのは, 対応する Quality Center プロジェクトに接続しているときだけです。

3 テストに関連付ける関数ライブラリを選択し、[開く] または [OK] をクリックします(関数ライブラリをファイル・システムまたは Quality Center プロジェクトのどちらから選択するかに応じて異なります)。

関連付けられている関数ライブラリを削除するには、次の手順を実行します。

▶ [リソース] 表示枠で、関数ライブラリを右クリックして [関数ライブラリを リストから削除] を選択するか、関数ライブラリを選択して DELETE キーを押 します。

+

「テストの設定」ダイアログ・ボックスの[リソース]タブの関連付けられた
 関数ライブラリのリストで、削除する関数ライブラリを選択し、[削除]ボタンをクリックします。関連付けられている関数ライブラリの優先順位付けを行うには、上向き矢印および下向き矢印を使用します。

詳細については、1199ページ「テストのためのリソース設定の定義」を参照し てください。

関数定義ジェネレータの使用方法

QuickTest の関数定義ジェネレータを使用すれば、新規ユーザ定義関数のための 定義を生成し、その定義にヘッダ情報を追加できます。その後、必要に応じ て、その関数をテスト・オブジェクトに登録できます。必要な情報を入力する と、関数定義ジェネレータが基本的な関数定義を自動的に作成します。関数定 義を定義した後、それを関数ライブラリに挿入してテストに関連付けることが できます。また、定義はエキスパート・ビュー内のテスト・スクリプトに直接 挿入することもできます。最後に、内容(コード)を追加して関数を完成させ ます。

注: 関数をエキスパート・ビューに直接挿入した場合, テストは特定のアクションのどこからでもその関数にアクセスできるようになります。

テスト・オブジェクトに関数を登録すると、関数は、そのテスト・オブジェクトによって呼び出すことができ、そのテスト・オブジェクトの使用可能な操作のリストに表示されます。

テスト・オブジェクトに関数を登録しないと、この関数はグローバル操作となり、ステップ・ジェネレータの [操作] ボックスの操作一覧に、キーワード・ビューの [操作] カラムに、また IntelliSense の使用時に表示されます。関数を登録すると、その関数が登録されているテスト・オブジェクトが選択された場合にステップ・ジェネレータまたはキーワード・ビューに表示される標準の操作として定義できます。

最後に、ステップ・ジェネレータまたはキーワード・ビューで操作にカーソル を位置付けたときと、IntelliSenseの使用時に表示されるツールチップを定義す ることによって、ユーザ定義関数に説明を付けることができます。また、ユー ザ定義関数を含むステップが実際に何をするかについて説明した文章を追加す ることもできます。この文章は、ステップ・ジェネレータの[ステップについ てのコメント] ボックスと、キーワード・ビューの [注釈] カラムに表示され ます。

関数定義ジェネレータに情報を入力するにつれて、でき上がりつつある関数定義 が[**プレビュー**]領域に表示されます。関数の定義が完了したら、アクティブ な QuickTest 文書に定義を挿入します。定義を関数ライブラリに挿入した場合、 関連付けられているテストは関数にアクセスできます。関数を[エキスパート ビュー]内のテストに直接挿入した場合、関数は、特定のアクション内からのみ 呼び出すことができます。最後に、関数の内容(コード)を追加します。

次の項では,関数定義ジェネレータで関数を作成するときに実行する手順の概 要を説明します。

関数定義ジェネレータを使用するには、次の手順を実行します。

- 1 911ページ「関数定義ジェネレータの起動」の説明に従って、関数定義ジェネ レータを開きます。
- 2 913ページ「関数の定義」の説明に従って、関数を定義します。
- 3 914 ページ「関数ジェネレータを使用した関数の登録」の説明に従って、必要 に応じて関数をテスト・オブジェクトに登録します。

標準設定では、テスト・オブジェクトに登録されない関数は、ステップ・ジェ ネレータの[**関数**]カテゴリまたはキーワード・ビューの[**操作**]項目を選択 することによって呼び出せる、あるいは IntelliSense の使用時に呼び出せるグ ローバル関数として自動的に定義されます。関数をテスト・オブジェクトに登 録した場合は、その関数(操作)をそのテスト・オブジェクトの標準設定の操 作として定義することもできます。

- 4 918ページ「関数の引数の指定」の説明に従って、関数に引数を追加します。
- 5 ヘッダ情報を追加して関数を文書化します。詳細については,919ページ「関数への説明の追加」を参照してください。

- 6 関数の仕上げの前に, 921 ページ「関数のプレビュー」の説明に従って, その 関数をプレビューします。
- **7** 921 ページ「別のユーザ定義関数の生成」の説明に従って、必要に応じて別の 関数定義を生成します。
- 8 922 ページ「ユーザ定義関数の仕上げ」の説明に従って、アクティブな文書に 各関数を挿入し、それらの関数に内容を追加して、各関数を仕上げます。

注:この項で説明する手順では、前述の手順を実行しているものと想定しています。

関数定義ジェネレータの起動

QuickTest から関数定義ジェネレータを開きます。

関数定義ジェネレータを開くには,次の手順を実行します。

1 関数定義を挿入する対象となる関数ライブラリまたはテストがアクティブな文書であることを確認します(対象文書にフォーカスを移動するには、文書のタブをクリックします)。これは、関数定義の完了後、関数定義ジェネレータが現在アクティブな文書に関数を挿入するためです。

2 [挿入] > [関数定義ジェネレータ] を選択するか, [関数定義ジェネレータ] ボタンをクリックします。関数定義ジェネレータが開きます。

黎 、関数定義ジェネレータ		×
関数定義 名前(<u>N</u>):	<u>引数:</u> 名前	+ × ↑ ↓
タイプ(I): Function マ 対象(P): Public マ		
「テストオブジェクトに登録する(B) テストオブジェクト(B): 「標準設定操(たとして登録する(D))	▶ 操作[]]:	
注意加情報服 詳細(E): ドキュメント(M):)
プレビュー Public Function 'TODO: add function body here End Function		
, 「別の関数定義を挿入する(S)		

関数定義ジェネレータを開いた後、で説明するように、新しい関数の定義を開 始できます。

関数の定義

関数定義ジェネレータを開いたら、関数の定義を開始できます。

┌関数定義	i
名前(N):	VerifyProperty
タイプ(工):	Function
対象(P):	Public 💌

たとえば、指定されたプロパティの値を確認する関数を定義する場合、関連付け られている任意のテストから呼び出せるように、その関数に VerifyProperty とい う名前を付けてパブリック関数として定義できます(プライベートとして定義し た関数は、同じ関数ライブラリ内にある別の場所からのみ呼び出せます。プライ ベート関数は、テスト・オブジェクトに登録できません)。

関数を定義するには、次の手順を実行します。

 【名前】ボックスに、新しい関数の名前を入力します。ステップ・ジェネレー タまたはキーワード・ビューから簡単に選択できるように、何をする操作なの かがはっきりとわかる名前を付けてください。関数名には、英字以外の文字を 含めることはできません。また、関数名は英字で始まらなければならず、ス ペースや次の文字を含めてはなりません。
 !@#\$%^&*()+=[]¥{}|;':"",/<>?

注:ユーザ定義関数に、組み込みの関数と同じ名前(たとえば、GetLastError, MsgBox, Print など)を付けないようにします。組み込み関数は、ユーザ定義 関数に優先します。したがって、組み込み関数と同じ名前のユーザ定義関数を 呼び出しても、組み込み関数が代わりに呼び出されます。組み込み関数のリス トについては、ステップ・ジェネレータ([挿入] > [ステップジェネレー タ])の「組み込み関数」リストを参照してください。

- 2 関数またはサブルーチンのどちらを定義するのかに応じて、[タイプ] リストから [Function] または [Sub] を選択します。
- 3 [対象] リストから,関数の適用範囲として, [Public] (この関数ライブラリ に関連付けられている任意のテストコンポーネントからこの関数を呼び出せる ようにする場合)または [Private] (同じ関数ライブラリ内の別の場所からの みこの関数を呼び出せるようにする場合)を選択します。標準設定では,適用 範囲は [Public] に設定されています (パブリック関数のみ,テスト・オブ ジェクトに登録することができます)。

注: ユーザ定義関数を手作業で作成し,範囲を [Public] とも [Private] とも 定義しなかった場合,その関数は標準設定でパブリック関数として扱われます。

パブリック関数を定義した後、で説明するように、関数を登録できます。また、プライベート関数を定義した場合、あるいは関数を登録しない場合は、引き続き関数に引数を指定できます。詳細については、918ページ「関数の引数の指定」を参照してください。

関数ジェネレータを使用した関数の登録

パブリック関数をテスト・オブジェクトに登録すると、その関数(操作)をテ スト・オブジェクトに対して実行できるようになります。関数をテスト・オブ ジェクトに登録するときには、既存の操作の機能をオーバーライドすること も、そのテスト・オブジェクトに対する新しい操作として登録することもでき ます。

関数をテスト・オブジェクトに登録すると、ステップ・ジェネレータでそのテ スト・オブジェクトを選択したときに、およびキーワード・ビューの中で[**項**] リストからそのテスト・オブジェクトを選択したときに[操作] リストに、 また IntelliSense に、およびステップ・ジェネレータの一般的な[操作] リスト に、関数が操作として表示されます。関数をテスト・オブジェクトに登録する と、その関数はそのテスト・オブジェクトからしか呼び出せなくなります。 テスト・オブジェクトに関数を登録することを選択した場合,関数定義ジェネ レータの右上角にある[引数]領域に,1番目の引数として自動的に test_object 引数が追加されます。また,関数定義ジェネレータは,関数定義 のすぐ後ろに,適切な引数値を持つ RegisterUserFunc ステートメントを自動 的に追加します。

関数をテスト・オブジェクトに登録するときに、任意でその関数をテスト・オ ブジェクトの標準設定の操作として定義することもできます。定義した場合 は、QuickTest エンジニアまたは各分野のエキスパートが「項目]リスト内の関 連付けられているテスト・オブジェクトを選択したときに、標準設定で「操 作]カラムにその関数が表示されます。また、IntelliSense でも関数を選択でき るようになります。関数をテスト・オブジェクトの標準設定の関数として定義 すると、RegisterUserFunc ステートメントの4番目の引数の値として True が 指定されます。

関数を特定のテスト・オブジェクトに登録しなかった場合,その関数は自動的 にグローバル関数として定義されます。グローバル関数は,ステップ・ジェネ レータの [**関数**] カテゴリ,またはキーワード・ビューの [**操作**] 項目を選択 して呼び出します。グローバル関数のリストは,ステップ・ジェネレータで [**関数**] カテゴリを選択した場合は [**操作**] ボックスに,キーワード・ビュー の [**項目**] リストから [**操作**] 項目を選択した場合は [**操作**] リストに,また IntelliSense の使用時に,アルファベット順に表示されます。

実行時に、QuickTest によって、指定された関数がまずテストの中で検索され、 次に[リソース]タブに表示されている順番で関数ライブラリの中で検索され ます。指定したテストまたは関数ライブラリ内に関数名の一致する関数が複数 見つかった場合は、テストまたは関数ライブラリ内で最後に検出された関数が 使用されます。QuickTest によって2つの異なる関数ライブラリで同じ名前の2 つの関数が見つかった場合、優先順位の高い方の関数ライブラリの関数が使用 されます。混乱を避けるために、1つのテストに関連付けられているリソースの 中では、それぞれの関数に一意の名前を付けることを推奨します。 **ヒント**: この時点で関数を登録しなかった場合は後ほど手作業でその関数の後 に次の例のように RegisterUserFunc ステートメントを付け加えて登録するこ とができます。

RegisterUserFunc "WebEdit", "MySet", "MySetFunc"

この例では、ユーザ定義関数 MySetFunc を使用して、MySet メソッド(操作) を WebEdit テスト・オブジェクトに追加しています。追加後、QuickTest エンジ ニアキーワード・ビューで[項目]リストから WebEdit テスト・オブジェクト を選択すると、[操作]リストに、WebEdit テスト・オブジェクトの登録されて いるほかの操作、およびあらかじめ用意されている操作とともに MySet 操作が 表示されます。

また,関数をほかのテスト・オブジェクトに登録することもできます。それに は,関数コードを関数ライブラリに保存するときに,RegisterUserFuncステー トメントを複製し(コピーして貼り付けて),必要に応じて引数値を変更します。

この関数を標準設定の関数として定義するには RegisterUserFunc ステートメントの4番目の引数の値を True に指定します。例を次に示します。 RegisterUserFunc "WebEdit", "MySet", "MySetFunc", True

注:登録した関数もグローバル関数も、テストに関連付けられているテスト・ スクリプトまたは関数ライブラリに追加された後にのみ、テストから呼び出せ ます。 関数をテスト・オブジェクトに登録するには、次の手順を実行します。

1 [テストオブジェクトに登録する] チェック・ボックスを選択します。この領域内のオプションが使用可能になり、関数定義ジェネレータの右上角にある [引数]領域内の引数のリストに test_object という新しい引数が自動的に追加されます(test_object 引数は、関数を登録する対象となるテスト・オブジェクトを受け取ります)。

関数定義		
名前(N): VerifyProperty		
タイプ(I): Function 💌	test_object リファレンス	
対象(P): Public ▼		
┌ ▽ テスト オブジェクトに登録する(匠)		
テストオブジェクト(B): Link 💌 操作(D): VerifyProperty 💌		
□ 標準設定操作として登録する(D)		

注: [テストオブジェクトに登録する] チェック・ボックスをクリアすると, 標準の test_object 引数が [引数] 領域から自動的に削除されます(名前を変更 しなかった場合)。

- 2 使用可能なオブジェクトのリストから Test object を選択します。たとえば、 サンプルの VerifyProperty 関数は、Link テスト・オブジェクトに登録することになるでしょう。
- 3 テスト・オブジェクトに追加またはオーバーライドする操作を指定します。
 - ➤ 新しい操作を定義するには、「操作」ボックスに新しい操作の名前を入力します。たとえば、サンプルの VerifyProperty 関数の場合は、新しい VerifyProperty 操作を定義します。
 - ▶ 既存の操作の標準機能をオーバーライドするには、[操作] ボックスで使用 可能な操作のリストから操作を選択します。

4 QuickTest エンジニアまたは各分野のエキスパートが、関連付けられている項目を選択したときに関数を標準設定の操作として「操作」カラムに表示されるようにするには、「標準設定操作として登録する」チェック・ボックスを選択します。

たとえば、VerifyProperty 操作を Link テスト・オブジェクトの標準設定の操作 として定義した場合は、RegisterUserFunc ステートメントの4番目の引数に値 として True が定義されます。その構文は次のようになります。

RegisterUserFunc "Link", "VerifyProperty", "VerifyProperty", True

テスト・オブジェクト登録情報を指定した後、関数のその他の引数を指定します。

関数の引数の指定

基本的な関数定義を行い、テスト・オブジェクト登録情報を指定したら、必要 に応じて関数の引数を指定できます。



たとえば、914ページ「関数ジェネレータを使用した関数の登録」で説明した 例のように、テスト・オブジェクトに関数を登録することにした場合は、1番 目の引数の test_object に加えて、prop_name(チェック対象プロパティの名 前)および expected_value(プロパティの期待値)という引数を割り当てるこ とができます。関数が正しく動作するためには、要求された引数を定義する必 要があります。

引数は任意の順序でリスト表示できます。ただし、関数をテスト・オブジェクトに登録する場合は、常に先頭の引数がテスト・オブジェクトを受け取らなけ ればなりません。

関数の引数を定義するには、次の手順を実行します。

[**引数**]領域で,関数の引数を指定します。必要に応じて,引数はいくつでも 追加できます。わかりやすくするために,それぞれの引数には,どのような値 を入力する必要があるかを示す名前を付けてください。

▶ 引数を追加するには, ● をクリックして引数の名前を入力します。引数には、その引数にどのような値を入力する必要があるかをはっきりと示す名前を付けてください。引数名には、英字以外の文字を含めることはできません。また、引数名は英字で始まらなければならず、スペースや以下の文字を含めてはなりません。
 ! @ #\$ % ^ & * () + = [] ¥ { } | ; '; : "", / <>?

標準設定では、「成功モード」は「値」に設定されます。この設定では、 QuickTest が引数として値を関数に渡します。引数値を参照によって渡す場 合は、「成功モード」ボックスで「リファレンス」を選択します。

- ▶ 引数を削除するには、その引数を選択して ≥ をクリックします。これで、 その引数が関数定義ジェネレータから削除されます。
- ▶ 引数の順序を設定するには、矢印 1 と を使用します。引数の順序は関数コードのわかりやすさに影響するだけですが、パブリック関数を登録する場合は例外です。その場合は、先頭の引数がテスト・オブジェクトを受け取る必要があります。

関数への説明の追加

関数定義ジェネレータでは、ユーザ定義関数にヘッダ情報を追加できます。カー ソルが操作の上に置かれたときにツールチップとして表示される説明を追加でき ます。QuickTest エンジニアは、このツールチップの情報に基づいて、使用可能 な操作のリストからどの操作を選択するべきかを判断できます(説明文は可能な 限り短く簡潔にしておくことをお勧めします)。

さらに、当該の関数を使用するステップが正確に何をするかを指定する注釈を 追加できます。テスト・オブジェクト名、テスト・オブジェクト・タイプ、お よびテキスト内の任意の引数値を含めることができます。また、必要に応じて テキストを手作業で入力することもできます。ここで追加したテキストは、ス テップ・ジェネレータのステップについてのコメント]ボックスと、キーワー ド・ビューの[注釈]カラムに表示されます。したがって、明確でわかりやす い文章にしなければなりません。 たとえば、検索エンジンから「HP」へのリンクをチェックする場合、関数定義 ジェネレータを使用して次の注釈を定義します。

'@Documentation Check if the <Test object name> <Test object type> <prop_name> value matches the expected value: <expected_value>.

┌追加情報――	
言羊糸田(<u>E)</u> :	Checks whether a property value matches the actual value.
ドキュメント(<u>M</u>):	ieck whether the <test name="" object=""> <test object="" type=""></test></test>

キーワード・ビューで引数の値を選択した後はこの注釈はたとえば次のように 表示されます。

Check if the "Management Software" link "text" value matches the expected value: "Business Technology Optimization (BTO) Software".

ヒント:キーワード・ビューで任意のカラム・ヘッダを右クリックして, [**ド** キュメントのみ]オプションを選択すると,ステップのリストを表示または印 刷することができます。このオプションを選択すると,QuickTest に [注釈]カ ラムだけが表示されるようになります。[編集] > [**ドキュメントをクリップ** ボードにコピー]を選択して,任意のアプリケーションに注釈を貼り付けるこ ともできます。したがって,ステップに関してこのカラムに表示される文章は, 手動テスト用の指示としても使用できる明確な文章でなければなりません。

関数に説明を付けるには、次の手順を実行します。

 [詳細] ボックスに、ツールチップとして表示されるテキストを入力します。 ツールチップは、ステップ・ジェネレータの[操作] リスト、キーワード・ ビューの[操作] カラムおよび IntelliSense でカーソルが関数名の上に置かれた ときに表示されます。

たとえばサンプルの VerifyProperty 関数の場合は次のように入力できます。 Checks whether a property value matches the actual value.

2 [ドキュメント] ボックスに,ステップ・ジェネレータの [ステップについて のコメント] ボックスまたはキーワード・ビューの [注釈] カラムに表示させ るテキストを入力します。[注釈] のテキストには引数を含めることができま す。それには, ▶ をクリックして使用する引数を選択します。 [テストオブジェクトに登録する] チェック・ボックスがオンになっている場合は, ▶ をクリックすることで,表示されたリストから, [Test object name] 項目または [Test object type] 項目,あるいはその両方を [注釈] カラムに追加することもできます。これらのテスト・オブジェクト項目や引数項目を [注釈] テキストに含めると,それらの項目は,対応するテスト・オブジェクト名,テスト・オブジェクト・タイプ,または引数値と動的に置き換えられます。

関数のプレビュー

関数を定義するにつれて、その関数のコードが [**プレビュー**] 領域に読み取り 専用形式で表示されます。ここで関数の内容を確認して、必要なら関数定義 ジェネレータのさまざまな領域で変更を加えることができます。

たとえば、サンプルの VerifyProperty 関数の場合は、[**プレビュー**] 領域に次のようなコードが表示されます。



コードを確認したら(アクティブな文書に関数を挿入する前に),別の関数定 義を生成するか,または定義した関数のコードを仕上げるかを選択できます。

別のユーザ定義関数の生成

コードをプレビューしたら(アクティブな文書に関数を挿入する前に),さらにほかの関数定義を生成するかどうかを選択できます。

注: ほかの関数を定義しない場合は、次の項に進みます。

ほかのユーザ定義関数を生成するには、次の手順を実行します。

1 [別の関数定義を挿入する] チェック・ボックスを選択して、[挿入] をクリッ クします。アクティブな文書に関数定義が挿入され、関数定義ジェネレータか らのデータが削除されます。関数定義ジェネレータは開いたままとなります。 2 913 ページ「関数の定義」で説明するように、新しい関数を定義します。

ユーザ定義関数の仕上げ

コードをプレビューしたら,関数をアクティブな文書に挿入します。関数を関 数ライブラリに挿入した場合,関数ライブラリに関連付けられているテスト は,関数にアクセスできます。関数をテスト([エキスパート ビュー]内)に 直接挿入した場合,テストには,特定のアクション内の任意の場所からの関数 呼び出しを含めることができます。

コードを必要な場所に挿入した後,関数を仕上げることができます。たとえば,VerifyProperty 関数の場合,関数ライブラリまたはテストに次のコードが挿入されます。

⁽@Description Checks whether a property matches its expected value
⁽@Documentation Check whether the <Test object name> <Test object type>
<prop_name> value matches the expected value: <expected_value>.
Public Function VerifyProperty (test_object, prop_name, expected_value)

'TODO: add function body here End Function RegisterUserFunc "Link", "VerifyProperty", "VerifyProperty"

ヒント: RegisterUserFunc ステートメント(最後の行)が VerifyProperty 関数 を Link テスト・オブジェクトに登録します。関数を複数のテスト・オブジェク トに登録するには、この行をコピーして各テスト・オブジェクト用に複製し、 必要に応じて引数値を変更します。 関数を仕上げるには、その内容を追加します(TODO コメントを置き換えます)。たとえば、関数の中でプロパティの期待値が特定のテスト・オブジェクトの実際のプロパティ値と一致しているかどうかを検証するには、関数の本体に次のコードを付け加えます。

Dim actual_value

' Get the actual property value

actual_value = obj.GetROProperty(prop_name)

' Compare the actual value to the expected value

If actual_value = expected_value Then

Reporter.ReportEvent micPass, "VerifyProperty Succeeded", "The " & prop_name & " expected value: " & expected_value & " matches the actual value"

VerifyProperty = True

Else

Reporter.ReportEvent micFail, "VerifyProperty Failed", "The " & prop_name & " expected value: " & expected_value & " does not match the actual value: " & actual_value

VerifyProperty = False

End If

ユーザ定義関数を仕上げるには、次の手順を実行します。

1 [OK] をクリックします。アクティブな文書に関数定義が挿入され、関数定義 ジェネレータが閉じます。

注: アクション内で関数を直接定義すると、その関数はそのアクションの中でのみ呼び出せます。

2 関数ライブラリまたはテストの中で,必要に応じて TODO 行を置き換えて,関 数コードに内容を追加します。 **ヒント**:実行セッション後に関数をテスト結果ツリー([テスト結果] ウィン ドウ)に表示するには,Reporter.ReportEvent ステートメントを関数コードに 追加します(前出の例のように)。

ユーザ定義関数の中で標準設定のテスト・オブジェクト・メソッドを使用して いる場合,このステップは実行セッション後に[テスト結果]ウィンドウに表 示されます。ただし,Reporter.ReportEventステートメントを関数コードに追 加して追加情報を提供したり,必要に応じてテストのステータスを変更したり できます。

3 コードを関数ライブラリに挿入した場合は、その関数ライブラリをテストに関 連付けて、そのユーザ定義関数にアクセスできるようにする必要があります。 また、コードの構文を確認して、テストがその関数にアクセスできることと、 QuickTest エンジニアがその関数を表示し、使用できることを確認する必要もあ ります。詳細については、905ページ「関連付けられている関数ライブラリを 使用した作業」を参照してください。

ユーザ定義関数のテスト・オブジェクト・メソッドとしての登録

914 ページ「関数ジェネレータを使用した関数の登録」で説明した,QuickTest 関数定義ジェネレータを使用した関数の登録に加え,RegisterUserFuncステー トメントを使って,テスト・オブジェクトに新規メソッドを追加したり,実行 セッション中に既存のテスト・オブジェクト・メソッドの振る舞いを変更した りできます。

関数をテスト・オブジェクトに登録するときに、必要に応じてその関数をテスト・オブジェクトの標準設定の操作として定義することができます。標準設定の操作は、その関数が登録されているテスト・オブジェクトが選択されたときに標準でステップ・ジェネレータ、またはキーワード・ビューの[操作]カラムに表示されます。

関数をテスト・オブジェクトに登録しないと、この関数はグローバル関数にな ります。グローバル関数は、ステップ・ジェネレータの[**関数**]カテゴリ、ま たはキーワード・ビューの[**操作**]項目を選択して、または IntelliSense の使用 時に呼び出します。UnregisterUserFunc ステートメントを使用すれば、新規メ ソッドを無効にしたり、既存のメソッドを QuickTest の元々の振る舞いに戻し たりできます。

メソッドを登録するには、まずテストまたは関連関数ライブラリ内に関数を定 義します。次に、関数の末尾に RegisterUserFunc ステートメントを挿入して、 テスト・オブジェクト・クラス、使用する関数、および関数を呼び出すメソッ ド名を指定します。テスト・オブジェクト・クラスに新しいメソッドを追加す ることも、既存のメソッド名を使って、指定したメソッドの機能を(一時的 に)オーバーライドすることもできます。

登録したメソッドは、メソッドを登録したテストまたは関数ライブラリにのみ 適用されます。また、QuickTest は、各実行セッションの開始時にすべての登録 関数を消去します。

ユーザ定義関数の準備

ユーザ定義関数の使用範囲をローカル・アクションに限定するには、関数をテ ストに直接書き込みます。ユーザ定義関数を多数のアクションやテストで使え るようにするは、関数を関連関数ライブラリに格納します(推奨)。ローカル のアクションと関連関数ライブラリの中に同じ名前の関数が存在する場合, QuickTest ではアクション内で定義されている関数が使用されます。

登録されているメソッドを含んだステートメントを実行すると、ステートメン トによって対象テスト・オブジェクトが最初の引数としてメソッドに送られま す。したがって、ユーザ定義関数には少なくとも1つの引数がなければなりま せん。ユーザ定義関数は任意の数の引数を取ることができます。また、テス ト・オブジェクト引数のみを取ることもできます。関数が既存のメソッドを オーバーライドする場合、その構文はオーバーライド対象の関数と正確に同じ でなければなりません。つまり、最初の引数はテスト・オブジェクトで、残り の引数はすべて元のメソッド引数と一致することになります。

ヒント: parent テスト・オブジェクト・プロパティを使用して, 関数の最初の 引数で表されるオブジェクトの親を取得できます。

例: ParentObj = obj.GetROProperty("parent")

自分で関数を書く場合には、標準のVBScript ステートメントに加え、QuickTest の任意の予約済みオブジェクト、メソッド、関数、および関数の最初の引数と して渡されるテスト・オブジェクトに関連付けられている任意のメソッドが使 えます。

たとえば、エディット・ボックスに新しい値を設定する前に現在の値を[テス ト結果]に報告したいとします。その場合は、標準のQuickTest Set メソッド を、エディット・ボックスの現在の値を取得し、その値を[テスト結果]に報 告してからエディット・ボックスに新しい値を設定する関数でオーバーライド します。 その関数は次のようになります。

Function MyFuncWithParam (obj, x)
 dim y
 y = obj.GetROProperty("value")
 Reporter.ReportEvent micDone, "previous value", y
 MyFuncWithParam=obj.Set (x)
End Function

注:この関数では戻り値を定義して、テストから呼び出されるたびに、**Set** メ ソッドの引数の値を返すようにしています。

ユーザ定義テスト・オブジェクト・メソッドの登録

RegisterUserFunc ステートメントを使って QuickTest に対して,テストの実行中,あるいはメソッドの登録を解除するまでの間,ユーザ定義関数を指定されたテスト・オブジェクト・クラスのメソッドとして使うように指示できます。

注:メソッドを登録する(そしてアクションが終っても登録を解除しない)外 部アクションを呼び出した場合,登録したメソッドは,そのアクションを呼び 出したテストでそれ以降も使用できます。

ユーザ定義関数をテスト・オブジェクト・メソッドとして登録するには, 次の 構文を使います。

項目	詳細
TOClass	任意のテスト・オブジェクト・クラス。 注 : QuickTest の予約済みオブジェクト(たとえば DataTable , Environment, Reporter など)のメソッドと してユーザ定義メソッドを登録することはできません。
MethodName	登録するメソッドの名前(QuickTest上で,たとえばキー ワード・ビューや IntelliSense で表示されます)。特定のテス ト・オブジェクト・クラスにすでに関連付けられているメ ソッドの名前を指定すると,ここで指定するユーザ定義関数 が既存のメソッドをオーバーライドします。新しい名前を指 定すると,オブジェクトがサポートするメソッドのリストに そのメソッドが追加されます。
FunctionName	テストから呼び出すユーザ定義関数の名前。この関数はテスト または任意の関連付けられている関数ライブラリに置けます。
SetAsDefault	登録する関数を,テスト・オブジェクトの標準設定のメソッドとして使用するかどうかを示します。 キーワード・ビューまたはステップ・ジェネレータでテスト・オブジェクトを選択すると,[操作]カラム(キーワード・ビュー)または[操作]ボックス(ステップ・ジェネレータ)に標準設定のメソッドが自動的に表示されます。

RegisterUserFunc TOClass, MethodName, FunctionName, SetAsDefault

ヒント:登録する関数が関数ライブラリ内で定義されている場合は,関数ライ ブラリに RegisterUserFunc ステートメントを含めることをお勧めします。こ れにより,当該関数ライブラリを使用する任意のテストでメソッドをすぐに利 用できるようになります。 たとえば, Find Flights Web ページに [Country] エディット・ボックスが含ま れており,標準設定でこのボックスに「USA」という値が含まれているとしま す。次の例では,エディット・ボックスの標準設定の値を新しい値が入力され る前に取得するために, MySet 関数を使用するように Set メソッドを登録して います。

Function MySet (obj, x)

dim y y = obj.GetROProperty("value") Reporter.ReportEvent micDone, "previous value", y MySet=obj.Set(x)

End Function

RegisterUserFunc "WebEdit", "Set", "MySet"

Browser("MercuryTours").Page("FindFlights").WebEdit("Country").Set "Canada"

詳細と例については, **『HP QuickTest Professional Object Model Reference』**(英 語版)を参照してください。

ユーザ定義テスト・オブジェクト・メソッドの登録解除

RegisterUserFunc ステートメントを使ってメソッドを登録すると、そのメソッドはテストの終わりまで、または登録解除されるまで指定のテスト・オブジェクトのメソッドとして認識されます。このメソッドが QuickTest のメソッドをオーバーライドしている場合、このメソッドの登録を解除すると、メソッドは通常の動作に戻ります。ほかのメソッドの登録を解除すると、テスト・オブジェクトによってサポートされているメソッドのリストから、それらを削除することになります。

メソッドの登録解除は、再利用可能なアクションに QuickTest のメソッドを オーバーライドする登録メソッドが含まれている場合に特に重要です。たとえ ば、呼び出し先のアクションの中で直接定義されている関数を使うメソッドの 登録を解除しない場合、登録したメソッドが後のアクションで呼び出される と、呼び出し元のテストが失敗します。これは、関数の定義を見つけることが できないことが原因です。 この場合,登録した関数が関数ライブラリ内で定義されていれば,呼び出し元 のテストは正常に動作できます(関数ライブラリが呼び出し元のテストに関連 付けられている場合)。しかし,呼び出し元のテストの作者は,呼び出し先の アクションに登録された関数が含まれていることに気付かずに,後のアクショ ンで QuickTest の通常の動作を想定して,登録されたメソッドを使う可能性が あります。そして,予期しない結果が生じることも考えられます。

ユーザ定義メソッドの登録を解除するには、次の構文を使用します。

UnRegisterUserFunc TOClass, MethodName

項目	詳細
TOClass	メソッドが登録されているテスト・オブジェクト・クラス。
MethodName	登録を解除するメソッド。

たとえば, Find Flights Web ページに [Country] エディット・ボックスが含ま れており,標準設定でこのボックスに「USA」という値が含まれているとしま す。次の例では,エディット・ボックスの標準設定の値を新しい値が入力され る前に取得するために, MySet 関数を使用するように Set メソッドを登録して います。[Country] エディット・ボックスのための WebEdit.Set ステートメン トの中で,登録されたメソッドを使用した後,Set メソッドを標準の機能に戻 すために UnRegisterUserFunc ステートメントを使用しています。

```
Function MySet (obj, x)
dim y
y = obj.GetROProperty("value")
```

Reporter.ReportEvent micDone, "previous value", y

MySet=obj.Set(x)

End Function

```
RegisterUserFunc "WebEdit", "Set", "MySet"
Browser("MercuryTours").Page("FindFlights").WebEdit("Country").Set "Canada"
UnRegisterUserFunc "WebEdit", "Set"
```

ユーザ定義関数の使い方のヒント

ユーザ定義関数を使用するときには、次のヒントとガイドラインを考慮してく ださい。

- ▶ 関数定義ジェネレータを使って関数を定義し、さまざまなオプションを試してみれば、必要な構文について詳しい知識を得られます。
- ▶ 関数は、登録されると、テスト・オブジェクト・クラス全体に適用されます。 特定のテスト・オブジェクトに限定してメソッドを登録することはできません。
- ➤ その他のテスト・オブジェクトから関数を呼び出す場合は、RegisterUserFunc 行をコピーして、別の関数の直後に貼り付け、適切な引数値を指定します。
- ➤ 登録する関数が関数ライブラリ内で定義されている場合は、関数ライブラリに RegisterUserFunc ステートメントを含めることをお勧めします。これにより、 当該関数ライブラリを使用する任意のテストでメソッドをすぐに利用できるようになります。
- ▶ QuickTest は各実行セッションの開始時にすべての登録メソッドを消去します。
- ▶ [ステップから実行]や [ステップからデバッグ] などの部分的な実行または デバッグ・オプションを使用して、(関数ライブラリの中でなく)テスト・ス テップ内のメソッド登録の後の位置からテストの実行を開始した場合、 QuickTest はメソッド登録を認識しません。これは、登録が現在の実行セッショ ンの開始よりも前に行われるためです。
- ▶ テストに関連付けられている関数ライブラリ内で Option Explicit ステートメン トを使用するには、ステートメントをテストに関連付けられているすべての関 数ライブラリに含める必要があります。関連付けられている関数ライブラリの 一部にのみ Option Explicit ステートメントを含めた場合、すべての関数ライブ ラリ内のすべての Option Explicit ステートメントが無視されます。Option Explicit ステートメントは、制限なしにアクション・スクリプトの中で直接使用 することができます。
- ▶ 各関数ライブラリのグローバル・スコープにある変数は、一意でなければなりません。2つの関連付けられている関数ライブラリにおいて、Dim ステートメントを使用してグローバル・スコープ内で同じ変数を定義している場合、または同じ名前を持つ2つの定数を定義している場合、2番目の定義によって構文エラーが発生します。グローバル・スコープにおいて同じ名前を持つ2つ以上の変数を使用する必要がある場合、(関数ライブラリは逆順で読み込まれるため)最後の関数ライブラリにのみDim ステートメントを挿入します。

- ▶ 標準設定では、ユーザ定義関数を使用するステップは、実行セッション後に [テスト結果] ウィンドウのテスト結果ツリーに表示されません。関数がテスト 結果ツリーに表示されるようにするには、Reporter.ReportEvent ステートメン トを関数コードに付け加える必要があります。たとえば、必要に応じて追加情 報を提供したり、テストのステータスを変更したりすることが考えられます。
- ▶ 使用されている関数を関連付けられている関数ライブラリから削除すると、その関数を使用しているテスト・ステップは アイコンで示されます。その後、そのテストの実行セッションで、存在しない関数を使用しているステップに達すると、エラーが発生します。
- ▶ テストが参照する関数ライブラリをほかのユーザが変更した場合、または、 QuickTest エンジニアが外部エディタ(QuickTest 以外)を使用して関数ライブ ラリを変更した場合、変更は、テストを再度開くまで反映されません。
- ▶ 同じ名前を持つ2つ以上の関数がテスト・スクリプトまたは関数ライブラリに存在する場合、必ず最後の関数が呼び出されま(QuickTestは、関数ライブラリを検索する前に、テスト・スクリプト内で関数を検索します)。混乱を避けるために、1つのテストに関連付けられているリソースの中では、それぞれの関数に一意の名前を付けてください。
- ▶ 再利用可能なアクションの中でメソッドを登録した場合には、アクションの終わりにそのメソッドの登録を解除することを(そして必要があれば次のアクションの初めに再登録することを)強くお勧めします。そうすることで、そのアクションを呼び出すテストがメソッドの登録による影響を受けないようにします。
- ▶ 先にメソッドを登録解除しなくても、一度登録したメソッドが異なるユーザ定 義関数を使うように登録しなおすことが可能です。ただし、このメソッドの登 録を解除すると、QuickTestの本来の動作に戻り(あるいは、新規のメソッド だった場合には完全に消去され)、それ以前に登録されていたものには戻りま せん。

たとえば、次のステートメントを入力するとします。

RegisterUserFunc "Link", "Click", "MyClick" RegisterUserFunc "Link", "Click", "MyClick2" UnRegisterUserFunc "Link", "Click"

UnRegisterUserFunc ステートメントを実行した後, Click メソッドは MyClick2 関数で定義されている機能を使うのを止め, QuickTest の元の Click 機能に戻 り, MyClick 関数で定義されている機能には戻りません。
▶ VBScript を使った関数やサブルーチンの作成の詳細については、QuickTest の [ヘルプ] メニューから VBScript に関するマニュアルを参照してください([ヘ ルプ] > [QuickTest Professional ヘルプ] > [VBScript リファレンス])。

テストからの外部定義された関数の実行

関数ライブラリ(VBScript ファイル)をテストに関連付けず,なおかつテスト 内のアクションまたはほかの関数ライブラリから関数,サブルーチン,クラス などを呼び出せるようにするには,アクションに ExecuteFile ステートメント を挿入します。

テストを実行すると, ExecuteFile ステートメントは関数ライブラリ内のすべ てのグローバル・コードを実行して,そのファイル内のすべての定義がアク ションのスクリプトのグローバル・スコープから利用できるようにします。

注: ExecuteFile ステートメントを使用して呼び出されるファイルおよび当該 ファイルに含まれている関数をデバッグすることはできません。また, ExecuteFile ステートメントを含んでいるテストをデバッグする場合,実行 マーカが正しく表示されないことがあります。

ヒント:作成するすべてのアクションに同じ ExecuteFile ステートメントを含めたければ、ステートメントをアクション・テンプレートに追加します。詳細については、503ページ「アクション・テンプレートの作成」を参照してください。

外部定義関数を実行するには、次の手順を実行します。

- 標準の VBScript 構文を使って VBScript ファイルを作成します。詳細については、『Microsoft VBScript 言語リファレンス』([ヘルプ] > [QuickTest Professional ヘルプ] > [VBScript リファレンス] > [VBScript])を参照 してください。
- 2 作成したファイルを、テストを実行するコンピュータからアクセスできる任意のフォルダに保存します。

3 次の構文を使用して、テストのアクションに ExecuteFile ステートメントを追加します。

ExecuteFile FileName

- ここで FileName は、作成した VBScript ファイルの絶対パスまたは相対パスです。
- 4 必要に応じて、アクション内で、指定した VBScript ファイルにある関数やサブ ルーチンを使用します。

注:

- ExecuteFile ステートメントでは、VBScript の ExecuteGlobal ステートメントが利用されます。詳細については、『Microsoft VBScript 言語リファレンス』([ヘルプ] > [QuickTest Professional ヘルプ] > [VBScript リファレンス] > [VBScript] を選択)を参照してください。
- ➤ ExecuteFile ステートメントをアクション内で実行するとき、ファイル内の 関数は現在のアクションからのみ呼び出せます。テスト全体で VBScript ファイル内の関数を利用できるようにするには、[テストの設定] ダイアロ グ・ボックスの [リソース] タブの関連関数ライブラリのリストにファイル 名を追加します。詳細については、905 ページ「関連付けられている関数ラ イブラリを使用した作業」を参照してください。

第 VI 部

テストの実行と分析

第 29 章

テストの実行

テストの作成後は、それを実行してアプリケーションの動作を検査できます。

本章では、次の項目について説明します。

- ▶ テストの実行について(937ページ)
- ▶ テスト全体の実行(938ページ)
- ▶ テストの一部の実行(943ページ)
- ▶ オプション・ステップの使用(946ページ)
- ▶ テスト・バッチの実行(948ページ)

テストの実行について

テストを実行すると、それに含まれるステップが QuickTest によって実行され ます。テストのパラメータを設定している場合は、その値を入力するよう求め るメッセージが表示されます。セッションの実行が完了すると、QuickTest に よって結果の詳細を示すレポートが表示されます。結果の表示の詳細について は、第 30章「実行セッション結果の表示」を参照してください。

テストにグローバルなデータ・テーブル・パラメータが含まれている場合は、 データ・テーブルのデータ行ごとに1回ずつテストが実行されます。テスト に、現在のアクションのデータ・シートに対応するデータ・テーブル・パラ メータが格納されている場合は、そのアクション・データ・シートのデータ行 ごとに1回ずつ、そのアクションが実行されます。また、テスト全体または特 定のアクションについて、最初の反復だけを実行するのか、すべての反復を実 行するのかを指定することもできます。あるいは、データ・セットの指定した 範囲について反復を実行するように指定することも可能です。詳細について は、第41章「個別のテストのオプション設定」および第14章「アクションを 使った作業」を参照してください。 テスト全体を先頭から実行することも、一部分だけを実行することも可能で す。特定のステップを「オプション」に指定すれば、そのステップが成功しな かった場合に、実行が停止するのではなくステップがバイパスされるようにす ることができます。テストを更新して、テスト・オブジェクト記述、チェック ポイントの期待値、ActiveScreenの画像や値を変更できます。

動的な要素を持つオブジェクトについてテストを実行することもできます。詳 細については、第4章「テスト・オブジェクトを使用した作業」を参照してく ださい。

一連のテストを設定して順次実行するには、QuickTest Test Batch Runner を使用 します。詳細については、948ページ「テスト・バッチの実行」を参照してく ださい。

WinRunner ユーザの方へ: Quick Test テストの実行中に, WinRunner テストを 実行したり, WinRunner のコンパイル済みモジュールから関数を呼び出したり できます。詳細については, 第 49 章「WinRunner を使用した作業」を参照して ください。

テスト全体の実行

QuickTest では、テストの実行を開始するステップを特に指定しないかぎり、常 に最初のステップから開始されます。ステップまたはアクションを指定して、 そこから(またはそこまで)テストを実行するには、[ステップから実行]オ プションまたは [ステップまで実行]オプションを使用します。これらの機能 は、テストを最初からまたは最後まで実行せずに、テストの特定のセクション を検査する場合に役立ちます。詳細については、943ページ「テストの一部の 実行」を参照してください。

テストを開始するには、[実行]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、結果の場所を指定し、定義されたテストのパラメータ の値を入力します。 テストを実行するには、次の手順を実行します。

1 テストをまだ開いていなければ、[ファイル] > [開く] > [テスト] を選択 するか、[開く] ボタンをクリックして、テストを開きます。

ヒント:最近使用したテストを開く場合は, [**ファイル**] メニューの最近使用 したファイルのリストから選択することもできます。

2 ツールバーの [実行] ボタンをクリックするか, [オートメーション] > [実行] を選択します。[実行] ダイアログ・ボックスが開きます。

実行
結果保管場所 入力パラメータ
実行結果の書き込み先:
● 新規実行結果フォルダN
¥Program Files¥HP¥QuickTest Professiona¥Tests¥UG¥Res1
◎ 一時実行結果フォルダ(古い一時結果(こ上書き)[<u>[</u>]
 OK キャンセル ヘルプ

- 3 その実行セッションにおける結果の場所(および,必要に応じて入力パラメー タの値)を指定します。詳細については、941ページ「[結果保管場所] タブに ついて」および942ページ「[入力パラメータ] タブについて」を参照してく ださい。
- 4 [OK] をクリックします。[実行] ダイアログ・ボックスが閉じ,実行セッションが開始されます。標準では,実行セッションが終了すると,[テスト結果] ウィンドウが開きます。実行セッション結果の表示の詳細については,第 30 章「実行セッション結果の表示」を参照してください。

注: [オプション] ダイアログ・ボックスの [実行] タブで [実行セッション 終了時に結果を表示する] チェックボックスをクリアすると,実行セッション の最後で [テスト結果] ウィンドウは開きません。[オプション] ダイアログ・ ボックスの詳細については,第40章 「グローバル・テスト・オプションの設 定」を参照してください。

ヒント:実行セッションを中断する場合は、次のいずれかを行います。

- ▶ デバッグ・ツールバーの [一時停止] ボタン をクリックするか, [デバッ グ] > [一時停止] を選択します。実行が停止します。一時停止した実行 セッションの実行を再開するには, [実行] ボタンをクリックするか, また は [オートメーション] > [実行] を選択します。
 - ▶ [停止] ボタンをクリックするか, [オートメーション] > [停止] を選択しま す。実行セッションが停止され, [テスト結果] ウィンドウが表示されます。

ファイル操作を実行すると(たとえば,別のテストを開いたり,新規のテスト を作成したりするなど),実行セッションは中断します。

Ш

[結果保管場所] タブについて

[結果保管場所] タブでは、実行セッション結果を保存する場所を指定できます。

実行	×
結果保管場所 入力パラメータ	
実行結果の書き込み先:	
● 新規実行結果フォルダ№) ¥Program Files¥HP¥QuickTest Professiona¥Tests¥UG¥Res1	<u></u>
◎ 一時実行結果フォルダ(古い一時結果に上書き)[[]	
OK キャンセル ヘル	,7

次のオプションのいずれかを選択します。

▶ [新規実行結果フォルダ]: このオプションには、結果を保存する標準パスの フォルダの名前が表示されます。標準設定では、QuickTest テストの結果はテスト・フォルダに格納されます。

標準設定を受け入れるか、テキスト・ボックスにパスを入力するか、参照ボタンをクリックして別のフォルダを選択して、新しいパスを指定します。このフォルダは、新規であるか、空であるか、QuickTest テスト・ファイルまたはコンポーネント・ファイルのみが含まれているフォルダでなければなりません。

注: Quality Center プロジェクトからテストを実行する場合, [新規実行結果 フォルダ] オプションは表示されず,代わりに [プロジェクト名], [実行名], [テスト セット] および [インスタンス] の各オプションが表示されます。詳 細については,1346ページ「Quality Center プロジェクトに格納されているテス トの QuickTest からの実行」を参照してください。 ▶ [一時実行結果フォルダ]:一時フォルダに実行結果を保存します。このオプションによって、このフォルダに保存されている結果はすべて上書きされます。

注: QuickTest では、すべてのテストの一時的な結果が <**システム・ドライブ**> **¥Documents and Settings¥** <**ユーザ名**> **¥** Local Settings¥Temp¥ TempResults に格納されます。[**一時実行結果フォル ダ**] オプションのテキスト・ボックスに表示されるパスは、読み取り専用のた め、変更できません。既存の結果フォルダに結果を保存すると、実行セッショ ン開始時に、フォルダの内容が削除されます。

[入力パラメータ] タブについて

[入力パラメータ] タブでは、実行セッション中に使用する入力パラメータの 実行時の値を指定できます。

実	行			×
ſ	結果保管場所	入力パラメータ		
	入力パラメータ	,		
	名前	タイプ	値	記述
	UserName	文字列	Administrator	
	Destination	文字列	Los Angeles	
		ОК	キャンセル	

[入力パラメータ] タブには,テストに定義された入力パラメータ(**[ファイ** ル] > [**設定**] > [**パラメータ**] タブを使用)が表示されます。 実行セッション中に使用するパラメータの値を設定するには、特定のパラメー タの [**値**] フィールドをクリックして、値を入力するか、ドロップダウン・リ ストから値を選択します。値を入力しなかった場合、実行セッション中は [テ ストの設定] ダイアログ・ボックスの標準設定の値が使用されます。

注:アクションの範囲内でテストの一部を実行する場合は([オートメーショ ン] > [ステップから実行] オプションまたは [オートメーション] > [現在 のアクションの実行] オプションを使用), [実行] ダイアログ・ボックスの [入力パラメータ] タブに, テスト・パラメータではなくアクションのパラ メータを指定する必要があります。

テストのパラメータの設定の詳細については,1204ページ「テストのパラメー タの定義」を参照してください。パラメータの使用方法の詳細については,第 22章「値のパラメータ化」を参照してください。

テストの一部の実行

テスト内の選択した部分を実行するには, [ステップから実行]オプションを 使用します。これにより, アプリケーションの特定のセクションだけを検査し たり, テストの特定の部分がスムーズに実行されるかどうかを確認したりでき ます。

注:また,テストをテストの最初から選択したステップまでデバッグ・モード で実行するには,[**デバッグ**] > [**ステップまで実行**] オプションを使用しま す。詳細については,1043 ページ「[ステップまで実行] コマンドおよび [ス テップからデバッグ] コマンドの使用」を参照してください。 エキスパート・ビューから実行している場合は,選択したステップから現在の アクションの最後までテストを実行できます。キーワード・ビューから実行し ている場合は,テストの最後まで実行できます。

- ▶ 選択したステップからアクションの最後までテストを実行するには、エキスパート・ビューのアクション・ビューから、[ステップから実行]オプションを使用します。このモードで[ステップから実行]を使用すると、すべての反復が無視されます。ただし、アクションに他のアクションがネストされている場合、ネストされているアクションが指定の回数だけ反復されます。
- ▶ 選択したステップからテストの最後までテストを実行するには、キーワード・ ビューから、[ステップから実行]オプションを使用します(選択したステッ プが再利用可能なアクションの一部でない場合)。このモードで[ステップから実行]を使用すると、すべての反復が実行されます。最初の反復は、選択したステップからテストの最後まで実行され、それ以降のすべての反復は、テストの最初から実行されます。

テスト内の単一のアクションを実行するには, [現在のアクションの実行]オ プションを使用します。[現在のアクションの実行]を使用すると,すべての 反復が無視されます。ただし,アクションに他のアクションがネストされてい る場合,ネストされているアクションが指定の回数だけ反復されます。

ヒント:

- ▶ テストの反復を1回だけ実行するには、[テストの設定]ダイアログ・ボッ クスの[実行]タブで、[反復なしで実行する]を選択します。
- ▶ テスト内の特定のポイントまでテストを実行するには、ブレークポイントを 挿入します。その場合、テストは選択したステップまたはアクションからブ レークポイントまで実行されます。ブレークポイントの詳細については、 1047ページ「ブレークポイントの設定」を参照してください。

アクションの詳細については、第14章「アクションを使った作業」を参照してください。

選択したステップからアクション全体,あるいはテストまたはアクションを実 行するには,次の手順を実行します。

- アプリケーションを開き、実行するアクションまたはステップの場所を表示します。
- 2 次のいずれかの方法で、テストの実行を開始するアクションまたはステップを選択します。
 - ▶ キーワード・ビューで、ステップまたはアクションの行を選択して強調表示します。
 - ▶ エキスパート・ビューで特定の行にカーソルを置きます。

選択したステップまたはアクションが、それより前にあるステップに依存して いないことを確認してください。

- 3 [オートメーション] > [ステップから実行] または [現在のアクションの実行] を選択するか,右クリックして [ステップから実行] を選択します。
- 4 941ページ「[結果保管場所] タブについて」および 942ページ「[入力パラメータ] タブについて」で説明されている手順に従い、[実行] ダイアログ・ボック スで、実行セッション結果の保存先と使用する入力パラメータを選択します。

注: アクションの範囲内でテストの一部を実行する場合は, [実行] ダイアロ グ・ボックスの [入力パラメータ] タブに, テスト・パラメータではなくアク ションのパラメータを指定する必要があります。

5 [OK] をクリックします。[実行] ダイアログ・ボックスが閉じ,実行セッションが開始されます。

標準では、実行セッションが終了すると、[テスト結果]ウィンドウが開きます。実行セッション結果の表示の詳細については、第30章「実行セッション 結果の表示」を参照してください。

[テスト結果] サマリには, [ステップから実行] または [現在のアクションの実行] オプションを使用してテストが実行されたことを示す注記が表示されます。

注: [オプション] ダイアログ・ボックスの [実行] タブで [実行セッション 終了時に結果を表示する] チェックボックスをクリアすると,実行セッション の最後で [テスト結果] ウィンドウは開きません。[オプション] ダイアログ・ ボックスの詳細については,第40章 「グローバル・テスト・オプションの設 定」を参照してください。

オプション・ステップの使用

オプション・ステップは、実行セッションが正常に終了するために必ずしも必要ではないステップです。たとえば、テストの記録中に、テスト対象アプリ ケーションによってログイン・ウィンドウにユーザ名とパスワードを入力する よう求められる場合があります。しかし、テストの実行時には、ユーザ名とパ スワードの入力を要求されません。これは、以前に入力した情報がアプリケー ションに保持されているためです。この場合、ログイン情報の入力について記 録されたステップは必要ではないので、オプション・ステップとして指定する 必要があります。

実行セッション中、オプション・ステップのオブジェクトが存在しない場合, QuickTest はこのステップを無視し、テストの実行を続行します。実行セッショ ンが完了すると、実行されなかったステップに関するメッセージが表示されま すが、このステップが原因で実行が失敗することはありません。

ただし、実行セッション中、QuickTest がオブジェクト・リポジトリ内でオプ ション・ステップのオブジェクトを見つけられない場合(たとえば、オブジェ クト名がテスト内で変更されてもオブジェクト・リポジトリ内では変更されて いない場合や、オブジェクトがオブジェクト・リポジトリから削除された場 合)、必要なオブジェクトを一覧表示したエラー・メッセージが表示され、実 行は失敗します。

記録中,QuickTestは、特定のダイアログ・ボックスを開くステップをオプションとして自動的に印を付けます(特定のダイアログ・ボックスの一覧は,948ページ「標準設定のオプション・ステップ」を参照してください)。また、ステップをオプションとして手動で指定できます。

注:ダイアログ・ボックスをバイパスするその他の方法としては,条件ステー トメントや回復シナリオを使って,ボタンのクリック,ENTER キーの押下,ス テップへのログイン情報の入力をステップの中で自動的に行う方法もありま す。詳細については,第44章「回復シナリオの定義と使用」を参照してくだ さい。

オプション・ステップの設定

キーワード・ビューでオプション・ステップを設定するには,右クリックして [オプションステップ]を選択します。[オプションステップ] アイコン ? が,選択したステップの横に追加されます。

👻 🎝 Welcome: Mercury Tours			
	Set	"Nicole"	"userName" edit box(こ "Nicole" を入力する。
🕐 👷 password	SetSecure	"47ce40d2145e	暗号化された文字列 "47ce40d2145eb49088fa85c9c214
	Click		"Sign-In" image をクリックする。
🕨 🎦 Find a Flight: Mercury	Check	CheckPoint("Fin	"Find a Flight: Mercury" Web page が Web コンテンツ アクセ

オプション・ステップをエキスパート・ビューで追加するには, VBScript ス テートメントの先頭に OptionalStep を追加します。例を次に示します。

OptionalStep.Browser("Browser").Dialog("AutoComplete").WinButton("Yes"). Click

エキスパート・ビューでの作業の詳細については,第26章「エキスパート・ ビューおよび関数ライブラリ・ウィンドウを使用した作業」を参照してください。

標準設定のオプション・ステップ

標準設定では、次のダイアログ・ボックスまたはメッセージ・ボックスがオプ ション・ステップと見なされます。

ダイアログ・ボックス / メッセージ・ボックスのタイトル・バー
[オートコンプリート]
[ファイルのダウンロード]
[Internet Explorer]
[Netscape]
[ネットワーク パスワードの入力]
[エラー]
[セキュリティの警告]
[セキュリティ情報]
[セキュリティ警告]
Username and Password Required

テスト・バッチの実行

複数のテストを連続して実行するには, Test Batch Runner を使用します。各テ ストの結果は,それぞれの標準の場所に保存されます。

Test Batch Runner を使用して,同じ一連のテストを後で簡単に実行できるよう に,テストのリストを作成して,.mtbファイルとして保存しておくことができ ます。また,バッチを実行する際に,テストをバッチ・リストに追加したり, バッチ・リストから除外したりできます。

注:

- ▶ Test Batch Runner でテストを実行できるようにするには、「オプション」ダ イアログ・ボックスの[実行]タブで、「他の HP 製品でテストおよびコン ポーネントを実行可能にする]チェック・ボックスを選択する必要がありま す。詳細については、第40章「グローバル・テスト・オプションの設定」 を参照してください。
- ➤ Test Batch Runner は、ファイル・システムに含まれるテストだけを使用できます。バッチ実行で Quality Center に保存されたテストを含める場合は、まずファイル・システムにテストを保存しなければなりません。
- ▶ [停止] ボタンをクリックすると、いつでもバッチ実行を終了できます。

テスト・バッチを設定して実行するには、次の手順を実行します。

- 1 [スタート] メニューから, [プログラム] > [QuickTest Professional] > [Tools] > [Test Batch Runner] を選択します。[Test Batch Runner] ダイア ログ・ボックスが開きます。
- 2 [追加] ボタンをクリックするか,または [バッチ] > [追加] を選択します。 [テストを開く] ダイアログ・ボックスが開きます。

3 テスト・バッチ・リストに追加するテストを選択し, [**開く**] をクリックしま す。選択したテストがバッチ・リストに追加されます。

☰無題 - Test Batch Runner
ファイル(E) バッチ(B) 表示(V) ヘルプ(H)
D 🗳 🖬 3= × 🐉 🗉 🖇
<u>ታ</u> スト
☑ C¥Program Files¥HP¥QuickTest Professional¥Tests¥MercuryTours
準備完了 [] [] []

4 リストに追加するそれぞれのテストについて、手順3を繰り返します。標準設 定では、選択した各テストはリストの最後に追加されます。

リストの最後以外の位置にテストを挿入するには、テストを追加する位置の1 つ下のテストを選択します。テストを追加すると、選択したテストの上にテストが挿入されます。

テストをリストから削除するには, [**削除**] ボタンをクリックするか, [**バッ チ**] > [**削除**] を選択します。

テストをリストに追加した後で、次回のバッチ実行中にそのテストを実行しな いようにするには、テスト名の横にあるチェック・ボックスをクリアします。

- 5 バッチ・リストを保存する場合は、「上書き保存」ボタンをクリックするか、
 [ファイル] > [上書き保存]を選択し、リストの名前を入力します。ファイルの拡張子は.mtb です。
 - 6 テスト・バッチを実行する準備ができたら、[実行] ボタンをクリックするか、 [バッチ] > [実行] を選択します。QuickTest をまだ開いていない場合には自動的に開き、テストが順次実行されます。バッチの実行が完了したら、各テストの結果が保存される標準のフォルダ(<テスト・フォルダ> ¥res#¥report)でテスト結果を参照できます。

テスト結果の詳細については,第30章「実行セッション結果の表示」を参照 してください。

 \times

影

第 30 章

実行セッション結果の表示

テストの実行が終わったら,実行セッション中に発生した重要なイベントに関 するレポートを表示できます。

本章では、次の項目について説明します。

- ▶ 実行セッション結果の表示について(952ページ)
- ▶ [テスト結果] ウィンドウ(953ページ)
- ▶ 実行セッションの結果の表示(960ページ)
- ▶ テスト実行結果の削除(981ページ)
- ▶ 実行セッション中に検出された不具合の送信(989ページ)
- ▶ Quality Center プロジェクトへの自動による不具合の送信(990ページ)
- ▶ 実行セッション中に検出された不具合の送信(989ページ)

実行セッション結果の表示について

実行セッションが終了すると、実行セッションの結果を[テスト結果]ウィン ドウで見ることができます。標準設定では、テストの実行が終了すると、[テ スト結果]ウィンドウが自動的に開きます。この動作を変更するには、[オプ ション]ダイアログ・ボックスの[実行]タブで[**実行セッションの終了時に** 結果を表示する]チェック・ボックスをクリアします。

[テスト結果] ウィンドウには,実行セッション中に実行されたステップの詳細が表示されます。データ・テーブル・パラメータを含まないテストの場合は,[テスト結果]ウィンドウには単一のテスト反復が表示されます。

テストにデータ・テーブル・パラメータが含まれていて、テストの設定が複数 の反復を実行するように設定されている合場合、[テスト結果] ウィンドウに は、テスト実行の反復ごとの詳細が表示されます。結果はテストのアクション 別にグループ化されます。

注: [テストの設定] ダイアログ・ボックスの [実行] タブの中で,テストを1 つまたはすべての反復で実行するように設定します。詳細については,1195 ページ「テストのための実行設定の定義」

テストの実行が終わったら,[テスト結果]ウィンドウに,実行セッションの あらゆる側面が表示されます。これには次のものがあります。

- ▶ 高レベルの結果概要レポート(テストの成功/失敗のステータス)
- ▶ すべてのテスト実行に使用されたデータ
- ▶ アプリケーション・エラーの発生場所を正確に示す、ステップの展開可能なツ リー
- ▶ テスト内で不具合が発生した正確な位置
- ▶ 特定のステップでのアプリケーションの状態の静止画像

- ▶ 特定のステップまたはテスト全体のアプリケーションの状態のムービー・クリップ
- ▶ テストの各段階で、成功または失敗した各ステップとチェックポイントの詳細 な説明

注: [テスト結果] ウィンドウには,結果をツリー階層で最大 300 レベルまで 表示できます。結果が 300 レベル以上ネストされている場合は, results.xml ファイルを手動で開いてレポート全体を表示できます。

[テスト結果] ウィンドウ

実行セッションが終わったら, [テスト結果] ウィンドウで結果を参照します。 標準設定では,実行セッションが完了すると, [テスト結果] ウィンドウが開 きます。標準設定の変更方法の詳細については,1181ページ「テストの実行オ プションの設定」を参照してください。

[テスト結果] ウィンドウの左側の表示枠には,実行結果ツリーが表示されま す。[テスト結果] ウィンドウの右側の表示枠には,実行結果ツリーで選択し たステップの詳細が表示されます。選択したステップの詳細には,テストのサ マリ,ステップの詳細,アプリケーションの静止画像,またはアプリケーショ ンのムービーが含まれます。

[テスト結果] ウィンドウは, [スタート] メニューからスタンドアロンのアプ リケーションとして開くことができます。[テスト結果] ウィンドウを開くに は, [スタート] > [プログラム] > [QuickTest Professional] > [Test Results Viewer] を選択します。 次の例では、3つの反復を含むテストの結果を示します。



前の例に示すテストには、実行結果ツリーに示されているように3つの反復が 含まれます。テストの結果は、テストのアクションごとにまとめられています。 [テスト結果] ウィンドウには、次の主な要素があります。

- ▶ テスト結果タイトル・バー:テストの名前を表示します。
- ▶ メニュー・バー:使用可能なコマンドのメニューが表示されます。
- ▶ 実行結果ツールバー: テスト結果を表示するボタンが含まれています(この ツールバーを表示するには、[表示] > [テスト結果ツールバー]を選択しま す)。詳細については、958ページ「実行結果ツールバー」を参照してください。
- ▶ 実行結果ツリー:実行結果ツリーにテスト結果が視覚的に表示されます。実行結果ツリーは、[テスト結果]ウィンドウの左側の表示枠内にあります。詳細については、955ページ「実行結果ツリー」を参照してください。
- ▶ [結果の詳細] タブ:実行結果ツリーで選択したノードの詳細が表示されます。 [結果の詳細] タブは,[テスト結果] ウィンドウの右側の表示枠内にありま す。詳細については,956ページ「実行結果詳細」を参照してください。
- ➤ [画面レコーダ] タブ: テスト結果に関連付けられている記録されたムービーが表示されます。[画面レコーダ] タブは, [テスト結果] ウィンドウの右側の表示枠内にあります。詳細については, 970ページ「アプリケーションの静止画像およびムービーのキャプチャと表示」を参照してください。
- ➤ ステータス・バー:現在選択されているコマンドのステータスが表示されます (このステータス・バーを表示するには, [表示] > [ステータス バー] を選択 します)。

[テスト結果]ウィンドウの外観は変更できます。詳細については,959ページ 「[テスト結果]ウィンドウの外観の変更」を参照してください。

実行結果ツリー

[テスト結果] ウィンドウの左側の表示枠には,**実行結果ツリー**が表示されます。 実行結果ツリーには,テスト結果が次のように図示されます。

- ▶ ✓ 成功したステップを示します。テストにチェックポイントが含まれていない場合、アイコンは表示されないことに注意してください。
- ▶ ★ 失敗したステップを示します。ステップが失敗すると、そのすべての親ス テップ(ルートのアクションまたはテストまで)が失敗となります。
- ▶ ! 警告を示します。そのステップは成功しなかったけれども、アクションまたはテストが失敗の原因ではなかったことを意味します。

- ▶ 1 ※ チェックポイントのオブジェクトが見つからない場合など、予期せず失敗したステップを示します。
- ▶ ↓ 失敗したために無視された省略可能なステップを示します。これによって テストが失敗することはありません。
- ▶ スマート認識メカニズムによってオブジェクトが正しく見つかったことを示します。
- ▶ ₩回復シナリオがアクティブになったことを示します。
- ▶ 🛺 実行セッションが,終了する前に停止したことを示します。
- ▶ [password].SetSecure テスト・オブジェクト名が角括弧で囲まれている場合, このテスト・オブジェクトは実行セッション中に動的に作成されたことを示し ます。動的テスト・オブジェクトは、プログラミングによる記述、または ChildObjects メソッドによって返されるオブジェクトを使用して作成されます。 また、動的テスト・オブジェクトは、オブジェクト・リポジトリに保存されま せん。
- ▶ Ⅲ 実行時データ・テーブルを表示します。これは、データ・テーブル・パラメータまたはテストの実行中に取得されたデータ・テーブルの出力値を含むテストを実行するときに使用する値を示すテーブルです。
- ▶ シンテナンス・モードでの更新結果を表示します。これは、失敗したステップでメンテナンス実行ウィザードによって行われたアクションと、そのアクションの詳細を示すテーブルです。メンテナンス実行モードで実行されたテストのみが表示されます。メンテナンス実行モードの詳細については、第33章「テストの保守」を参照してください。

ツリーに表示される詳細のレベルを変更するには,実行結果ツリーの分岐を折 りたたむか,展開します。

実行結果詳細

標準設定では,[テスト結果]ウィンドウが開くと,テストのサマリがウィン ドウの右側の表示枠の[**結果の詳細**]タブに表示されます。

[テスト結果] ウィンドウの右側の表示枠には, [結果の詳細] および [画面レ **コーダ**] というタブがあります。実行結果ツリーの最上位ノードを選択する と, [結果の詳細] タブにテストの結果のサマリが表示されます。ツリーで分 岐やステップを選択すると, [結果の詳細] タブにステップの詳細が表示され ます。[結果の詳細] タブには, 強調表示されているステップのアプリケー ションの静止画像も表示される場合があります。 [画面レコーダ] タブには、テスト結果に関連付けられたムービーが表示され ます。テスト結果に関連付けられたムービーがなければ、[画面レコーダ] タ ブには「**結果に関連付けられたムービーがありません。**」というメッセージが 表示されます。

アプリケーションの静止画像およびムービーの表示の詳細については,970 ページ「アプリケーションの静止画像およびムービーのキャプチャと表示」を 参照してください。

テスト結果ツリーの最上位ノードを選択すると,[結果の詳細]タブにはテストの名前,結果名,実行セッションの開始日時と終了日時,反復の回数,および反復が成功したか失敗したかが表示されます。反復にチェックポイントが含まれている場合,起こりうる結果は[**成功**]または[**失敗**]です。反復に チェックポイントが含まれていない場合,起こりうる結果は[**完了**]または [**失敗**]です。

また, Web サービス・アドインがインストールされていて実行セッション中に ロードされた場合, Web サービスの実行ツールキットが [結果の詳細] タブに 表示されます。テストに Web サービス・ステップが含まれていなくても,実行 ツールキットは表示されます。

テストがメンテナンス実行モードで実行された場合,[結果の詳細]タブには [メンテナンスサマリ]が表示されます。[メンテナンスサマリ]には,テス ト内で更新および追加されたオブジェクトの数が表示されます。また,テスト 内で更新およびコメントに変換されたステップの数も表示されます。[オブ ジェクトリポジトリ変更レポート]には、メンテナンス実行ウィザードがオブ ジェクト・リポジトリに加えた特定の変更と次の項が表示されます。

- ▶ [オブジェクトの追加]:メンテナンス実行ウィザードによってオブジェクト・ リポジトリに追加されたオブジェクトが表示されます。
- ▶ [説明が変更されたオブジェクト]:メンテナンス実行ウィザードによって実行 されたオブジェクト・リポジトリへの変更が示されます。

メンテナンス実行モードの詳細については,1057ページ「テストの保守」を参 照してください。 Quality Center からテストが実行された場合,テスト・セットとテスト・インス タンスの名前も表示されます。

test1 結果サマリ

テスト: test1 結果名: 実行 3-25 12-8-15 タイム ゾーン: 東京 (標準時) 実行開始: 2008/03/25 - 12:08:21 実行終了: 2008/03/25 - 12:08:36 テスト セット: default テストのインスタンス: 1

注: テスト・セットとは、特定のテスト目標を達成するために選択されたテス トで構成されたグループです。たとえば、アプリケーションのユーザ・インタ フェースや、ストレス下でのアプリケーションのパフォーマンスをテストする テスト・セットを作成できます。テスト・セットの定義は、Quality Center のテ スト実行モードで行います。詳細については、Quality Center のマニュアルを参 照してください。

実行結果ツールバー

実行結果ツールバーには、テスト結果を表示するためのボタンがあります。



[テスト結果] ウィンドウの外観の変更

標準設定では、[テスト結果] ウィンドウの外観(ルック・アンド・フィール) は QuickTest ウィンドウと同じで、Microsoft Office 2003 テーマが使用されます。 [テスト結果] ウィンドウの外観は、必要に応じて変更できます。

[テスト結果] ウィンドウの外観を変更するには、次の手順を実行します。

[テスト結果] ウィンドウで [**表示**] > [**ウィンドウのテーマ**] を選択し,使 用可能なテーマのリストからウィンドウの外観を選択します。たとえば, Microsoft Office 2000 または Microsoft Windows XP のテーマを適用できます。

注: コンピュータが Windows XP テーマを使用するように設定されている場合 にのみ, [テスト結果] ウィンドウに Microsoft Windows XP テーマを適用でき ます。

ヒント: QuickTest のメイン・ウィンドウに使用するテーマも変更できます。詳細については、52ページ「QuickTest ウィンドウの外観の変更」を参照してください。

実行セッションの結果の表示

標準設定では,実行セッションの最後に結果が[テスト結果]ウィンドウに表示されます(標準設定は,[オプション]ダイアログ・ボックスで変更できます。詳細については,1181ページ「テストの実行オプションの設定」を参照してください)。

また,現在のテストの以前の実行結果や,ほかのテストの結果を表示すること もできます。さらに,テスト結果を画面にプレビュー表示してから標準の Windows プリンタで印刷したり,HTMLファイルにエクスポートしたりもでき ます。

詳細については、以下を参照してください。

- ▶ 961 ページ「特定の実行を表示するためのテスト結果の表示」
- ▶ 965ページ「[テスト結果] ウィンドウでの作業」
- ▶ 970 ページ「Quality Center からのテスト実行の結果の表示」
- ▶ 970ページ「アプリケーションの静止画像およびムービーのキャプチャと表示」
- ▶ 975 ページ「結果ステップの検索」
- ▶ 976ページ「テスト結果の印刷」
- ▶ 977 ページ「テスト結果のプレビュー」
- ▶ 979ページ「テスト結果のエクスポート」

特定の実行を表示するためのテスト結果の表示

現在のテストに関して保存されている結果を表示できます。また,ほかのテストに関して保存されている結果も表示できます。

次の操作を実行して開いた[テスト結果を開く]ダイアログ・ボックスで、表示するテスト結果を選択して開きます。

- ▶ [テスト結果] ウィンドウの中で, [**ファイル**] > [**開く**] を選択します。
- ▶ 現在のテストに関していくつか結果がある場合、または何も結果がない場合でも、QuickTest ウィンドウの [結果] ボタン をクリックするか、[オートメーション] > [結果] を選択します。

テスト結果	を聞く <u>×</u>
テストを対	麗択:
F¥Quic	<test professiona#tests#checkpoint.<="" th=""></test>
利用可能	ちなテスト結果:
Res1 Res2 Res3 Res4 Res5 Res6 Res7	
	ファイルを開く(E) 更新(B)
開く	0) キャンセル ヘルフ

現在のテストに関する実行セッションの結果が一覧表示されます。結果セットの1つを表示するには、該当のセットを選択して[**開く**]をクリックします。

ヒント:指定したテストのパスを変更した後に結果リストを更新するには, [**更新**]をクリックします。



ほかのテストの実行結果を表示するには、ファイル・システムでテストまたは テスト結果ファイルを検索します。Quality Center に接続されている場合は、 Quality Center で QuickTest テスト(Quality Center に保存されている場合)を検 索することもできます。

ファイル・システムでの結果の検索

標準設定では、ファイル・システムに保存されている QuickTest テストの結果 はテスト・フォルダに格納されます。テストを実行するとき、[実行]ダイアロ グ・ボックスの[結果保管場所]タブを使用して結果の保存場所を指定するこ ともできます。結果ファイルの保存場所を指定すると、ファイル・システム内 で結果ファイルの場所を見つけやすくなります。詳細については、941ページ [[結果保管場所] タブについて」を参照してください。

ファイル・システムではテストまたは結果ファイルごとに結果を検索できます。

ファイル・システムの中でテストごとに結果を検索するには,次の手順を実行 します。

- 1 [テスト結果を開く] ダイアログ・ボックスで、テストの結果ファイルが含まれているフォルダのパスを入力するか、[参照] ボタンをクリックして [テストを開く] ダイアログ・ボックスを開きます。
- 2 表示する結果が含まれているテストを探して強調表示し、[**開く**] をクリック します。
- 3 [テスト結果を開く] ダイアログ・ボックスで、表示するテスト結果セットを 強調表示し、[**開く**]をクリックします。[テスト結果] ウィンドウに選択した 結果が表示されます。

ファイル・システムで結果ファイルを検索するには、次の手順を実行します。

- 1 [テスト結果を開く] ダイアログ・ボックスの中で, [**ファイルを開く**] ボタン をクリックして [結果ファイルの選択] ダイアログ・ボックスを開きます。
- 2 テストの結果ファイルが格納されているフォルダを参照します。
- 3 表示するテスト結果(.xml)ファイルを選択して強調表示してから,[開く] をクリックします。[テスト結果]ウィンドウに選択した結果が表示されます。

注:

- ▶ 標準設定では、テストの結果ファイルは<テスト>¥<結果名>¥Report に 保存されます。
- ➤ Quality Center バージョン 6.5 およびそれ以前のバージョンの結果ファイル は、.qtp という拡張子付きで保存されます。標準設定では、[結果ファイル の選択]ダイアログ・ボックスでは拡張子が.xmlの結果ファイルのみが表示されます。[結果ファイルの選択]ダイアログ・ボックスで拡張子が.qtp の結果ファイルが表示されるようにするには、[ファイルの種類]ボックス で「Test Results (*.qtp)」を選択します。

Quality Center に保存されている結果の検索

QuickTest テストが Quality Center に保存されている場合,結果は Quality Center の テスト・フォルダに保存されます。テスト結果の保存場所は変更できません。

Quality Center に保存されたテスト結果を検索するには、次の手順を実行します。

 I [テスト結果] ウィンドウで、「ツール] > [Quality Center への接続] を選択 するか、[Quality Center への接続] ボタン をクリックして Quality Center プロ ジェクトに接続します。 2 [テスト結果を開く] ダイアログ・ボックスで、QuickTest テストの結果ファイ ルが含まれているフォルダのパスを入力するか、[参照] ボタンをクリックし て [Quality Center プロジェクトからテストを開く] ダイアログ・ボックスを開 きます。

@Quality Center にテストを保存				_ 🗆 🗙
カテゴリ:Subject 🖵 🕞			7	ァイル システム
Subject BPT Resources Compiled Modules Gompleted BPT Tutorial Gruses Flight Application (BPT Demo) Flight Reservation Itinerary Mercury Tours Site Profiling Itests	<u></u>		72-87	作成済み
テスト名: MercuryTours				ОК
テストの種類 QuickTest テスト	•	🔽 ActiveScree	≥n ファイルの保存	ネー開じる

- 3 [テストの種類] リストで [QuickTest テスト] を選択します。
- 4 表示するテスト結果が含まれているテストを見つけて強調表示し、[OK] をク リックします。
- 5 [テスト結果を開く] ダイアログ・ボックスで、表示するテスト結果セットを 強調表示し、[**開く**] をクリックします。[テスト結果] ウィンドウに選択した テスト結果が表示されます。

Quality Center での作業の詳細については,第47章「Quality Center を使用した 作業」を参照してください。

[テスト結果] ウィンドウでの作業

[テスト結果] ウィンドウには、実行の視覚的なサマリとテキスト・サマリ、および実行の各ステップの詳細が表示されます。

実行の結果を表示するには、次の手順を実行します。

1 [テスト結果] ウィンドウをまだ開いていない場合は、[結果] ボタンをクリックするか、[オートメーション] > [結果] を選択します。

ヒント: [テスト結果] ウィンドウは, [スタート] メニューからスタンドアロ ンのアプリケーションとして開くことができます。[テスト結果] ウィンドウ を開くには, [スタート] > [プログラム] > [QuickTest Professional] > [Test Results Viewer] を選択します。

- ▶ 現在のテストに関するテスト結果が存在する場合、これらの結果は[テスト 結果]ウィンドウに表示されます。[テスト結果]ウィンドウの詳細につい ては、953ページ「[テスト結果]ウィンドウ」を参照してください。
- ▶ 現在のテストに関するテスト結果がいくつか存在する場合,あるいは現在の テストに関するテスト結果が存在しない場合,[テスト結果を開く]ダイア ログ・ボックスが表示されます。任意のテストについてテスト結果を選択す るか,ファイル・システムに存在するテスト結果(results.xml)ファイルを 検索できます。選択した結果を[テスト結果]ウィンドウに表示するには, [開く]をクリックします。テスト結果の表示の詳細については、961ページ 「特定の実行を表示するためのテスト結果の表示」を参照してください。

注: Quality Center バージョン 6.5 およびそれ以前のバージョンの結果ファイル は,.qtp という拡張子付きで保存されます。

- 2 ツリーに表示される詳細のレベルを選択するには、実行結果ツリーの分岐を折 りたたむか、展開します。
 - ▶ 分岐を閉じるには、その分岐を選択して分岐のアイコンの左にある折りたたみ(-)記号をクリックするか、キーボードのテンキー部分にあるマイナス・キー(-)を押します。結果ツリーでは、その分岐の詳細が消えて、折りたたみ記号が展開記号(+)に変わります。
 - ▶ 実行結果ツリーのすべての分岐を折りたたむには、「表示」> [すべて折りたたみ]を選択するか、または分岐を右クリックして [すべて折りたたみ] を選択します。
 - ▶ 分岐を展開するには、その分岐を選択して分岐アイコンの左にある展開 (+) 記号をクリックするか、キーボードのテンキー部分にあるプラス・ キー(+)を押します。ツリーの分岐の詳細が表示され、展開(+) 記号が 折りたたみ(-) 記号に変わります。

[テスト結果] ウィンドウを初めて開いたときには、ツリーは1レベルずつ展開します。ツリーが以前に展開されていた場合は、その前の状態に戻ります。

- ▶ 1つの分岐と、その分岐の下にあるすべての分岐を展開するには、その分岐を 選択し、キーボードのテンキーにあるアスタリスク・キー(*)を押します。
- ▶ 実行結果ツリーの分岐をすべて展開するには、「表示]> 「すべて展開」を 選択するか、分岐を右クリックして 「すべて開く」を選択するか、またはツ リーの最上位レベルを選択してキーボードのテンキーにあるアスタリスク・ キー(*)を押します。
- 3 個々の反復、アクション、またはステップの結果を表示できます。実行結果ツリーでステップを選択した場合、[テスト結果]ウィンドウの右側に選択したステップの詳細が表示されます。[オプション]ダイアログ・ボックスの[実行]タブでの設定に応じて、[テスト結果]ウィンドウの右側は2つの表示枠に分割されます。下の表示枠には、選択したステップの静止画像(選択したステップによっては別のデータ)が表示されます。また、右側の表示枠には、アプリケーションのムービーが表示されることもあります。詳細については、970ページ「アプリケーションの静止画像およびムービーのキャプチャと表示」および1181ページ「テストの実行オプションの設定」を参照してください。

結果は、次のいずれかになります。

- ➤ チェックポイントを含む反復,アクション、およびステップは、[テスト結果] ウィンドウの右側に「成功」または「失敗」と表示され、ツリー・ウィンドウでは ✓ や ★ アイコンで見分けることができます。
- ▶ チェックポイントは含まれない、実行が成功した反復、アクション、ステップは、[テスト結果]ウィンドウの右側に「完了」と表示されます。
- ▶ 成功しなかったが、テストの実行停止には至らなかったステップは、[テスト結果] ウィンドウの右側に「警告」と表示され、! や! ◎ アイコンで示されます。

注:「**警告**」と表示されたステップのあるテスト,反復,またはアクションで も,「**成功**」や「**完了**」と表示される場合があります。

- 7
- 4 [テスト結果] ウィンドウに表示される情報を絞り込むには、「フィルタ] ボタ ンをクリックするか、[表示] > [フィルタ] を選択します。[フィルタ] ダイ アログ・ボックスが開きます。

71NA
反復 ————————————————————————————————————
● すべて(A)
○ 範囲の指定(): 1 📑 ~ 1 🚍
ステータス
▼ 失敗(E) ▼ 警告(W) ▼ 成功(E) ▼ 完了(D)
内容
● すべて(L)
○ アクションのみ表示する(<u>6)</u>

標準のフィルタ・オプションは、上の画像のように表示されます。[フィルタ] ダイアログ・ボックスには、次のオプションが含まれます。

[**反復**]領域:

- ▶ [すべて]: すべての反復のテスト結果を表示します。
- ▶ [範囲の指定 X ~ Y]: 指定した範囲のテスト反復のテスト結果を表示します。
 「ステータス〕領域:
- ▶ [失敗]:失敗したステップのテスト結果を表示します。
- ▶ [警告]:ステータスが警告のステップ(成功はしなかったが,テストが失敗 する原因にはならなかったステップ)に関するテスト結果を表示します。
- ▶ [**成功**]: 成功したステップのテスト結果を表示します。
- ▶ [完了]: ステータスが完了のステップ(ステップの実行に成功したが,成功,失敗,警告のステータスを受け取らなかったステップ)に関するテスト結果を表示します。

[**内容**] 領域:

- ▶ [すべて]: テストのすべてのノードからすべてのステップを表示します。
- ▶ [アクションのみ表示する]: テスト内のアクション・ノードを表示します (アクション・ノードの特定のステップではありません)。
- 5 [テスト結果] 内で特定のステップを検索するには, [検索] ボタン をクリック するか, [ツール] > [検索] を選択します。詳細については, 「結果ステップ の検索」を参照してください。



6 実行結果ツリー内で以前に選択したノード間を移動するには、[前のノードに 移動]または [次のノードに移動] ボタン をクリックします。



3

φĄ

- 7 ほかの実行セッションの結果を表示するには、[開く] ボタン をクリックする か、[ファイル] > [開く] を選択します。詳細については、961 ページ「特定 の実行を表示するためのテスト結果の表示」を参照してください。
- 8 テスト結果を印刷するには、[印刷] ボタン をクリックするか、[ファイル] > [印刷] を選択します。詳細については、976ページ「テスト結果の印刷」を参 照してください。(実行結果は、印刷する前にプレビューできます。詳細につい ては、977ページ「テスト結果のプレビュー」を参照してください)。
注: Quality Center がインストールされている場合は,検出された不具合を Quality Center プロジェクトに追加できます。詳細については,989ページ「実 行セッション中に検出された不具合の送信」を参照してください。

- 9 実行結果を HTML ファイルにエクスポートするには、「ファイル]> [HTML ファイルにエクスポート]を選択します。詳細については、979ページ「テスト結果のエクスポート」を参照してください。
- **10** [ファイル] > [終了] を選択し, [テスト結果] ウィンドウを閉じます。

注:エキスパート・ビューで Reporter.Filter ステートメントを使用し, 選択し たステップの保存を有効または無効にしたり,「**失敗**」と「**警告**」ステータス のステップだけを保存したりできます。実行セッション情報の保存の詳細につ いては, 879ページ「実行セッション中に報告するステップの選択」,または **『HP QuickTest Professional Object Model Reference』**(英語版)を参照してくだ さい。Reporter.Filter ステートメントは,上記で説明した[フィルタ]ダイア ログ・ボックスとは異なります。Reporter.Filter ステートメントは[テスト結 果]に保存するステップを指定するのに対し,[フィルタ]ダイアログ・ボッ クスは,表示するステップを決定するものです。

Quality Center からのテスト実行の結果の表示

Quality Center から QuickTest テストが含まれるテスト・セットを実行すると, ホスト・コンピュータで Quality Center サーバが開き,そのコンピュータからテ ストを実行します。すべての実行結果は,実行されたテストの標準設定の場所 に保存されます。

QuickTest のテスト実行の結果は, Quality Center から表示できます。結果にアプ リケーションのムービーが含まれている場合, Quality Center で表示できます。 実行結果には、953ページ「[テスト結果] ウィンドウ」で説明した情報のほか に、次のフィールドが含まれます。

- ▶ [**テスト セット**]: テスト・セットの場所を指定します。
- ▶ [テストのインスタンス]: テスト・セット内のテストのインスタンス番号を指定します。たとえば、テスト・セットに同じテストが2回含まれている場合、 テスト・インスタンス1とテスト・インスタンス2の結果を表示できます。

アプリケーションの静止画像およびムービーのキャプチャと表示

Quality Center では、実行セッション中にアプリケーションの静止画像とムー ビーをキャプチャできます。キャプチャされたファイルは、[テスト結果] ウィンドウで表示できます。[テスト結果] ウィンドウの右側の表示枠には、 [**結果の詳細**] および [**画面レコーダ**] というタブがあります。これらのタブ では、アプリケーションの静止画像とテキストの詳細、またはムービーを表示 できます。

アプリケーションの静止画像の表示

標準設定では、QuickTest は失敗したステップに対してアプリケーションの静止 画像を保存します。実行結果ツリーで失敗したステップを選択して[結果の詳 細]タブを選択すると、[テスト結果]ウィンドウの右下の表示枠に、実行結 果ツリーで強調表示したステップに対応するアプリケーションの画面キャプ チャが表示されます。

🚰 Create Order [Res6] - テスト結果		1 ×
: ファイル(E) 表示(⊻) ツール(工) ヘルブ(H)		
े 😅 🗇 🗑 🍳 👭 🔝 @ @ ← → ?		
	標準チェックポイント "Fly From:": 失敗 日時: 2008/03/15 - 0.18:19 チェックオイントのタイムアウト: 10 秒間待機しました (最大許容待機時間: 10 秒)。 詳細 Fly From: 結果 プロパティ名 プロパティ値 selection test San Francisco	Ā
		-
		Ī
	★ Flight Reservation	
	Flight Reservation PIP	
	Flight Schedule:	
	Date of Flight: Fly From: Fly To:	
	01/20/10 San Francisco 🔽 Zurich 💌 Flights	
	Order Information:	
	Flight No: Departure Time: Arrival Time: Airline:	
	Name:	
	Tickets: 1	-
		-
「[[1] とこち畑オレ 人口学校主子されます	進備完了 💦 Quality Center 廿一次 http://cinderella/achin Quality Center 70%	7

強調表示したステップにエラーがない場合,右側の表示枠には結果詳細が表示 され,画面キャプチャは表示されません。

QuickTest が静止画像を保存する条件をカスタマイズするには, [オプション] ダイアログ・ボックスの [実行] タブの [**静止画像キャプチャをテスト結果へ** 保存] リストで, [常に], [エラー発生時], または [エラーおよび警告発生 時] を選択します。詳細については, 1181 ページ「テストの実行オプション の設定」を参照してください。

実行セッションのムービーの表示

QuickTest では、実行セッション中にアプリケーションのムービーを保存できま す。これは、アプリケーションがテスト条件下でどのように動作するかの確認 や、テストのデバッグに役立ちます。ムービー全体、または特定の部分を選択 して、ビューに表示できます。実行結果ツリーでステップを選択して「**画面レ コーダ**]タブをクリックすると、[テスト結果]ウィンドウの右下の表示枠に、 実行結果ツリーで強調表示したステップに対応するムービーのフレームが表示 されます。

QuickTest がムービーを保存する条件をカスタマイズするには, [オプション] ダイアログ・ボックスの [実行] タブの [ムービーをテスト結果へ保存] リス トで, [常に], [エラー発生時], または [エラーおよび警告発生時] を選択し ます。詳細については, 1181ページ「テストの実行オプションの設定」を参照 してください。



[画面レコーダ] タブの最上部にコントロールが表示されます。ここで再生, 一時停止,停止,ムービーの最初のフレームに移動,ムービーの最後のフレー ムに移動,ボリューム操作ができます。また,スライダ・バーをドラッグし て、ムービーをスクロールすることもできます。



ヒント:

- ▶ [テスト結果] ウィンドウの右側の表示枠をダブルクリックすると, [画面レ コーダ] が拡張して実行結果ツリーが隠れます。再度ダブルクリックする と, [画面レコーダ] は元のサイズに戻り,実行結果ツリーが表示されます。 [画面レコーダ] が拡張している場合, [画面レコーダ] の最上部にある再生 コントロールは最後のマウス操作,つまり [画面レコーダ] のどこかをク リックしてから約3秒後に自動的に隠れます。再度マウスを動かすと,再び 表示されます。
- ➤ [画面レコーダ] では、デスクトップ全体のムービーが保存されます。実行 セッション中は QuickTest を最小化して、ムービーのキャプチャ中に QuickTest ウィンドウがアプリケーションの上に部分的に重ならないように できます。実行セッション中に QuickTest を最小化する方法の詳細について は、1168ページ「QuickTest ウィンドウ・レイアウトのカスタマイズ」を参 照してください。

テスト結果からのムービーの削除

テスト結果から保存されているムービーを削除できます。これにより、テスト 結果ファイルのサイズが小さくなります。テスト結果からムービーを削除する には、[**ファイル**] > [**結果からムービーを削除**]を選択します。

キャプチャしたムービー・ファイルのエクスポート

キャプチャした 画面レコーダ・ムービーをファイルにエクスポートできます。 このファイルは .fbr ファイルとして保存されます。.fbr ファイルは HP Micro Recorder で表示できます (974 ページ「HP Micro Player での 画面レコーダ・ ムービー・ファイルの表示」で説明されています)。Quality Center で .fbr ファ イルを不具合に添付することもできます。Quality Center の QuickTest アドイン をインストールしている Quality Center ユーザは, Quality Center からムービー を表示できます。

画面レコーダ・ムービーをエクスポートするには、次の手順を実行します。

- 1 [ファイル] > [ムービーをファイルにエクスポート] を選択します。[名前を 付けて保存] ダイアログ・ボックスが開きます。必要に応じて、標準設定の保存 先フォルダおよびファイル名を変更できます。標準設定では、ファイルは<テス ト名>[<実行結果名>]と命名され、テスト結果フォルダに保存されます。
- 2 [保存] をクリックしてエクスポートされたファイル (.fbr) を保存し, ダイア ログ・ボックスを閉じます。

HP Micro Player での 画面レコーダ・ムービー・ファイルの表示

画面レコーダを使用して実行セッションのムービーをキャプチャした場合, ムービーはテスト結果フォルダに.fbrファイルとして保存されます。.fbrファ イルはファイル・システム内の任意の場所にエクスポートできます(974ペー ジ「キャプチャしたムービー・ファイルのエクスポート」で説明されていま す)。これらの.fbrファイルは, QuickTestの[テスト結果]ウィンドウを開か ずに, HP Micro Player を使用して表示することもできます。

HP Micro Player で 画面レコーダ・ムービーを再生するには、次の手順を実行 します。

- 1 次の手順のいずれかを実行します。
 - ▶ Windows エクスプローラ内で任意の **.fbr** ファイルをダブルクリックします。
 - ▶ [スタート] > [プログラム] > [QuickTest Professional] > [Tools] > [HP Micro Player] を選択し、HP Micro Player で [ファイル] > [開く] を選択して任意の.fbr ファイルを選択します。

HP Micro Player でムービーが開き,再生が始まります。

2 ウィンドウの最上部にあるコントロールを使用すると、ムービー内の特定の場所にアクセスしたり、音量の設定を変更したりできます。

結果ステップの検索

[検索] ダイアログ・ボックスでは、エラーや警告など特定のステップを、テ スト結果から検索することができます。たとえば、成功と完了など、ステータ スの組み合わせを選択して検索できます。

検索		×
結果ステップの検索対象ス	テータス:	次を検索(N)
□ 天敗止] ▼ 警告(₩)	┌検索方向───┐	キャンセル
✓ 成功(P)	○ 上へ(凹)	
☑ 完了(0)	● 下へ(D)	

次のオプションを使用できます。

オプション	説明
失敗	失敗したステップを [テスト結果] の中で検索します。
警告	警告が発行されたステップを検索します。
成功	成功したステップを [テスト結果] の中で検索します。
完了	実行を完了したステップを検索します。
検索方向	[テスト結果]のステップを先頭(上へ)に向かって検索するか,末尾(下へ)に向かって検索するかを指定します。

テスト結果の印刷

[テスト結果] ウィンドウから,テスト結果を印刷できます。印刷するレポートのタイプを選択できます。また,ユーザ定義のレポートの作成や印刷も行えます。

テスト結果を印刷するには、次の手順を実行します。

1 [印刷] ボタン をクリックするか、[ファイル] > [印刷] を選択します。[印
 刷] ダイアログ・ボックスが開きます。

印刷				J	×
ED刷範囲	_	-印刷部数-			
● すべて(<u>A</u>)		部数(<u>C</u>):	L L	1	
○ 選択した部分(E)					
● 簡略(<u>H</u>)					
○ 詳細(□)					
〇 ユーザ定義 XSL(U):					
[印刷(P)		キャンセル		ヘルプ	

- 2 [印刷範囲] オプションを選択します。
 - ▶ [**すべて**]: テスト全体の結果を印刷します。
 - ▶ [選択した部分]:実行結果ツリーで選択した分岐のテスト結果情報を印刷します。
- 3 [印刷部数] で印刷するテスト結果の部数を指定します。

- 4 [印刷形式] オプションを選択します。
 - ▶ [簡略]:実行結果ツリーの各項目のサマリ行(使用可能な場合)を印刷しま す。このオプションは、手順2で[すべて]を選択した場合にのみ使用でき ます。
 - ▶ [詳細]:手順2の選択に従って、実行結果ツリーの各項目、または選択した 分岐の利用可能な情報をすべて印刷します。
 - ▶ [ユーザ定義 XSL]: ユーザ定義の.xsl ファイルを参照したり選択したりで きます。印刷するレポートに含める情報やその表示形式を指定するユーザ定 義の.xsl ファイルを作成できます。詳細については、995ページ「テスト結 果の表示のカスタマイズ」を参照してください。

注:[**印刷形式**] オプションは, QuickTest バージョン 8.0 以降で作成したテスト結果にのみ使用できます。

5 [印刷] をクリックして, 選択したテスト結果情報を標準の Windows プリンタ に出力します。

テスト結果のプレビュー

テスト結果は、印刷する前に画面に表示できます。表示する情報の形式や範囲 を選択できます。また、情報をユーザ定義形式で表示できます。

注:[**印刷プレビュー**]オプションは,QuickTest バージョン 8.0 以降で作成したテスト結果にのみ使用できます。

テスト結果をプレビューするには、次の手順を実行します。

1 [ファイル] > [印刷プレビュー] を選択します。[印刷プレビュー] ダイアロ グ・ボックスが開きます。

印刷ブレビュー
- ED刷範囲
 ● すべて(A)
○ 選択した部分(E)
印刷形式
● 簡略(<u>H</u>)
○ 詳細(2)
○ ユーザ定義 XSL(U):
フレビュービー キャンセル ヘルプ

- 2 [印刷範囲] オプションを選択します。
 - ▶ [すべて]: テスト全体のテスト結果をプレビューします。
 - ▶ [選択した部分]: テスト結果ツリーで選択した分岐のテスト結果情報をプレビューします。
- 3[印刷形式]オプションを選択します。
 - ▶ [簡略]:実行結果ツリーの各項目のサマリ行(使用可能な場合)を印刷しま す。このオプションは、手順2で[すべて]を選択した場合にのみ使用でき ます。
 - ▶ [詳細]:手順2の選択に従って、実行結果ツリーの各項目、または選択した 分岐の利用可能な情報をすべて印刷します。
 - ▶ [ユーザ定義 XSL]: ユーザ定義の.xsl ファイルを参照したり選択したりで きます。プレビューに含める情報やその表示形式を指定するユーザ定義の .xsl ファイルを作成できます。詳細については、995ページ「テスト結果の 表示のカスタマイズ」を参照してください。

4 [**プレビュー**]をクリックし、画面にテスト結果のプレビューを表示します。



ヒント: プレビューに表示されない情報がある場合は(たとえば,チェックポイント名が長すぎてディスプレイに表示されないなど),[印刷プレビュー]ウィンドウの[ページ設定]ボタンをクリックして,ページの向きを [縦]から [横] に変更します。

テスト結果のエクスポート

テスト結果の詳細を HTML ファイルにエクスポートできます。これにより, QuickTest 環境を使用できなくても、テスト結果を表示できます。たとえば、テ スト結果を収めたファイルを、QuickTest をインストールしていない第三者に電 子メールで送信できます。エクスポートするレポートのタイプを選択できま す。また、ユーザ定義のレポートの作成やエクスポートも行えます。

テスト結果をエクスポートするには、次の手順を実行します。

1 [**ファイル**] > [**HTML ファイルにエクスポート**] を選択します。[HTML ファ イルにエクスポート] ダイアログ・ボックスが開きます。

HTML ファイルにエクスポート			×
┌ エクスポート範囲			
● すべて(<u>A</u>)			
○ 選択した部分(E)			
「エクスポート形式			
● 簡略(<u>H</u>)			
○詳細(□)			
○ ユーザ定義 XSL(U): □			
<u> ぼクスポート区</u> …	キャン	211	ヘルプ

- 2 [エクスポート範囲] オプションを選択します。
 - ▶ [すべて]: テスト全体の結果をエクスポートします。
 - ▶ [選択した部分]: テスト結果ツリーで選択した分岐のテスト結果情報をエク スポートします。

- 3 [エクスポート形式] オプションを選択します。
 - ▶ [簡略]:実行結果ツリーの各項目のサマリ行(使用可能な場合)をエクス ポートします。このオプションは、手順2で[すべて]を選択した場合にの み使用できます。
 - ▶ [詳細]:手順2の選択に従って、実行結果ツリーの各項目、または選択した 分岐の利用可能な情報をすべてエクスポートします。
 - ▶ [ユーザ定義 XSL]: ユーザ定義の.xsl ファイルを参照したり選択したりで きます。エクスポートするレポートに含める情報やその表示形式を指定する ユーザ定義の.xsl ファイルの作成が可能です。詳細については,995ページ 「テスト結果の表示のカスタマイズ」を参照してください。

注: [**エクスポート形式**] オプションは,QuickTest 8.0 以降で作成したテスト結果にのみ使用できます。

- 4 [エクスポート] をクリックします。[名前を付けて保存] ダイアログ・ボック スが開きます。必要に応じて、標準設定の保存先フォルダおよびファイル名を 変更できます。標準設定では、ファイルは<テスト名> [<実行結果名>] と いう名前が付けられ、テスト結果フォルダに保存されます。
- 5 [保存] をクリックして HTML ファイルを保存し、ダイアログ・ボックスを閉 じます。

テスト実行結果の削除

テスト結果削除ツールを使い,定義した条件に従って,システムから不要な結 果や古いテスト結果を削除できます。これによって貴重なディスク・スペース を解放できます。

このツールは Windows スタイルのユーザ・インタフェースで使用することも, Windows コマンド・ラインを使用してバックグラウンドで(非対話的に)実行 して,指定する条件に合う結果を直接削除できます。

テスト結果削除ツールを使った結果の削除

テスト結果削除ツールを使用して,ファイル・システムまたは Quality Center プロジェクト内の特定の場所にある,すべてのテスト結果の一覧を表示できます。その後不要なテスト結果を削除できます。

削除対象のテスト結果をより簡単に特定できるように,テスト結果削除ツール では,テスト結果を名前,日付,サイズなどで並べ替えることができます。

テスト結果削除ツールを使ってテスト結果を削除するには、次の手順を実行します。

 【スタート】メニューから、【スタート】>【プログラム】>【QuickTest Professional】>【Tools】>【Test Results Deletion Tool】を選択します。 【テスト結果削除ツール】ウィンドウが開きます。

しまたたい こうしょう しってい しょうしん しょう しょうしん しょうしん しょうしん しょうしん しょうしん しょうしん しょうしん しょう しょう しょう しょう しょう しょう しょう しょう しょう しょう	ソール					IX
テストまたはフォルダ:					参照(<u>B</u>)	▼
テスト結果:	🔲 サブフォルダで検出され	たテスト結果を含める	0			
名前	日付	時間	サイズ (KB) パ	2		
•						▶
肖·『除(D)	更新旧)		₽ 接続(<u>C</u>)	閉じる(0)	1ルプ田)

2 [テストまたはフォルダ] ボックスでは、テスト結果を削除するフォルダまた は特定のテストを指定します。ファイル・システムのフル・パスまたは、 Quality Center のフル・パスを指定できます。

また、次のようにしてテストまたはフォルダを参照できます。

- ▶ 特定のテストに移動するには、「参照」ボタンをクリックするか、「参照」ボタンの右に表示される矢印をクリックして、「テスト」を選択します。
- ▶ 特定のフォルダに移動するには、[参照] ボタンの右に表示される矢印をク リックして、[フォルダ] を選択します。

注: Quality Center データベースからテスト結果を削除するには、テスト・パス を参照または入力する前に、[**接続**] をクリックし、Quality Center に接続しま す。Quality Center のテスト・パスを [Quality Center] Subject¥ <フォルダ名> ¥ <テスト名>という形式で指定します。詳細については、1323 ページ 「Quality Center との接続と切断」を参照してください。

3 指定したフォルダのサブフォルダに含まれるすべてのテスト結果を表示する場合, [サブフォルダで検出されたテスト結果を含める] を選択します。

注: [**サブフォルダで検出されたテスト結果を含める**] チェック・ボックスは, ファイル・システムのフォルダにのみ使用できます。Quality Center のテストが 対象の場合にはサポートされません。

指定したテストまたはフォルダのテスト結果が、それぞれの記述的な情報と一 緒に[テスト結果] ボックスに表示されます。[テスト結果] ボックスのカラ ム・タイトルをクリックすると、当該カラムのエントリに基づいてテスト結果 の順番が並べ替えられます。並び順を逆にするには、同じカラム・タイトルを もう一度クリックします。

[テスト結果] ウィンドウのステータス・バーには,表示されたテスト結果に 関する情報が表示されます。情報には選択した結果の数,指定した場所にある 結果の総数,ファイル・サイズなどが含まれます。

- 4 削除するテスト結果を選択します。Windowsの標準的な選択操作の方法を使用 して、削除するテスト結果を複数選択できます。
- 5 [削除] をクリックします。選択したテスト結果が、システムと Quality Center データベースから削除されます。

ヒント: [**更新**] をクリックすれば、いつでも [テスト結果] ボックスに表示 されるテスト結果の一覧を更新できます。

Windows コマンド・ラインを使用した結果の削除

Windows コマンド・ラインを使用して、テスト結果削除ツールに対して、指定 の条件に従ってテスト結果を削除するよう指示できます。たとえば、特定の日 付より古いテスト結果や、最小ファイル・サイズよりも大きいテスト結果を常 に削除するようにできます。

コマンド・ラインからテスト結果削除ツールを実行するには,次の手順を実行 します。

Windows のコマンド・プロンプトを開き,

< QuickTest インストール先パス> ¥bin¥TestResultsDeletionTool.exe と入力します。次にスペースを入力してから、使用するコマンド・ライン・オ プションを入力します。

注:-Silent コマンド・ライン・オプションを使用してテスト結果削除ツールを 実行すると,指定した条件に合うテスト結果がすべて削除されます。このオプ ションを指定しない場合,[テスト結果削除ツール]ウィンドウが開きます。

コマンド・ライン・オプション

コマンド・ライン・オプションを使用して,削除するテスト結果の条件を指定 できます。以下に各コマンド・ライン・オプションについて説明します。

注:スペースを含むコマンド・ライン・オプションを追加する場合,オプション を引用符で囲んで指定する必要があります。たとえば,次のように指定します。 TestResultsDeletionTool.exe -Test "F:¥Tests¥Keep¥web objects"

-Domain < Quality Center ドメイン名>

接続先となる Quality Center ドメイン名を指定します。このオプションは, -Server, -Project, -User, -Password の各オプションと組み合わせて使う必要 があります。

-FromDate <結果作成日>

指定の日付より後に作成されたテスト結果を削除します。この日付またはそれ よりも前に作成されたテスト結果は削除されません。日付の形式は MM/DD/YYYYです。

次の例では、2005年11月1日より後に作成された結果がすべて削除されます。

TestResultsDeletionTool.exe -Silent -Test "C:¥tests¥test1" -FromDate "11/1/2005"

-Log <ログ・ファイルのパス>

指定したフォルダまたはテスト内の各テスト結果ファイルに関するエントリを 含むログ・ファイルを作成します。ログ・ファイルは、削除されたテスト結果 と、削除されなかった場合にはその理由を示します。たとえば、ファイル・サ イズが指定の最小ファイル・サイズよりも小さいためにテスト結果が削除され ないことが考えられます。

ファイル・パスと名前を指定することも、標準設定のパスと名前を使用することも可能です。ファイル名を指定しない場合は、標準設定のログ・ファイル名は、テスト結果削除ツールがあるフォルダ内の TestResultsDeletionTool.logとなります。

次の例では、ログ・ファイルは C:¥temp¥Log.txt に作られます。

TestResultsDeletionTool.exe -Silent -Log "C:¥temp¥Log.txt" -Test "C:¥tests¥test1"

次の例では、ログ・ファイルは **TestResultsDeletionTool.log** という名前で、 テスト結果削除ツールがあるフォルダ内に作られます。

TestResultsDeletionTool.exe -Silent -Log -Test "C:¥tests¥test1"

-MinSize <最小ファイル・サイズ>

指定した最小ファイル・サイズより大きいか同じサイズのテスト結果を削除し ます。サイズをバイト単位で指定します。

注:-MinSize オプションは、ファイル・システムのテスト結果に対してのみ使用 できます。Quality Center のテストが対象の場合はサポートされません。

The following example deletes all results larger than or equal to 10000 bytes. ファイル・サイズが 10,000 バイトより小さいテスト結果は削除されません。

TestResultsDeletionTool.exe -Silent -Test "C:¥tests¥test1" -MinSize "10000"

-Name <結果ファイル名>

削除する結果ファイルの名前を指定します。指定された名前のテスト結果のみ が削除されます。

正規表現を使用して,削除する結果ファイルの条件を指定できます。正規表現 および正規表現構文の詳細については,757ページ「正規表現の使用について」 を参照してください。

次の例では,Res1 という名前の結果が削除されます。

TestResultsDeletionTool.exe -Silent -Test "C:¥tests¥test1" -Name "Res1"

次の例では, Res で始まる 4 文字の名前を持つ結果がすべて削除されます(た とえば, 名前が Res1 や ResD ならば削除されます。ResDD ならば削除されま せん)。

TestResultsDeletionTool.exe -Silent -Test "C:¥tests¥test1" -Name "Res."

-Password < Quality Center パスワード>

Quality Center ユーザ名のパスワードを指定します。このオプションは, -Domain, -Server, -Project, -User オプションと組み合わせて使う必要があります。

次の例では, http://QCServer/qcbin にあるサーバを使用して, Default Quality Center ドメインに接続します。プロジェクト名は Quality Center_Demo, ユー ザ名は Admin, パスワードは PassAdmin です。

TestResultsDeletionTool.exe -Domain "Default" -Server "http://QCServer/qcbin" -Project "Quality Center_Demo" -User "Admin" -Password "PassAdmin"

-Project < Quality Center プロジェクト名>

接続先となる Quality Center プロジェクトの名前を指定します。このオプション は,-Domain,-Server,-User,-Password の各オプションと組み合わせて使 う必要があります。

-Recursive

指定したシステム・フォルダおよびサブフォルダのテスト結果をすべて削除し ます。-Recursive オプションを使用する場合は,-Test オプションに,(特定の テストのパスではなく)削除するテスト結果があるフォルダのパスが含まれて いる必要があります。

次の例では, F:¥Tests フォルダとそのすべてのサブフォルダの結果をすべて削 除します。

TestResultsDeletionTool.exe -Test "F:¥Tests" -Recursive

注:-Recursive オプションは,ファイル・システムのフォルダにのみ使用でき ます。Quality Center のテストが対象の場合はサポートされません。

-Server < Quality Center サーバのパス>

接続する Quality Center サーバの完全パスを指定します。このオプションは, -Domain, - Project, -User, -Password の各オプションと組み合わせて使う必要 があります。

-Silent

テスト結果削除ツールに,ユーザ・インタフェースなしで,バックグラウンド で(非対話的に)実行するよう指示します。

次の例では、テスト結果削除ツールを非対話的に実行して、**C:¥tests¥test1** にある結果をすべて削除します。

TestResultsDeletionTool.exe -Silent -Test "C:¥tests¥test1"

-Test <テストまたはフォルダのパス>

テスト結果削除ツールがテスト結果を削除するテストまたはテスト・パスを設 定します。テスト名,パス,ファイル・システムのパス,または Quality Center の完全パスを指定できます。

このオプションは必ず-Silent オプションと組み合わせて使用します。

注: Quality Center に接続する場合は, -Domain, -Server, -Project, -User, お よび -Password オプションを使用する必要があります。

次の例では, **F:¥Tests¥Keep¥webobjects** フォルダにある結果のリストを使用 して,テスト結果削除ツールを開きます。

TestResultsDeletionTool.exe -Test "F:¥Tests¥Keep¥webobjects"

次の例では, Quality Center の **Tests¥webobjects** テストの結果をすべて削除します。

TestResultsDeletionTool.exe -Domain "Default" -Server "http://QCServer/qcbin" -Project "Quality Center_Demo592" -User "Admin" -Password "PassAdmin" -Test "Subject¥Tests¥webobjects"

ヒント:-Test オプションと -Recursive オプションを組み合わせて,指定した ファイル・システム・フォルダとそのすべてのサブフォルダのテスト結果をす べて削除できます。

-UntilDate <テスト結果作成日>

指定した日付より前に作成されたテスト結果を削除します。この日付またはそ れよりも後に作成された結果は削除されません。日付の形式は MM/DD/YYYY です。

このオプションは必ず-Silent オプションと組み合わせて使用します。

次の例では、2005年11月1日よりも前に作成された結果がすべて削除されます。

TestResultsDeletionTool.exe -Silent -Test "C:¥tests¥test1" -UntilDate "11/1/2005"

-User < Quality Center ユーザ名>

接続先となる Quality Center プロジェクトのユーザ名を指定します。このオプ ションは, -Domain, - Server, - Project, - -Password の各オプションと組み合わせ て使う必要があります。

このオプションは必ず-Silent オプションと組み合わせて使用します。

実行セッション中に検出された不具合の送信

テストで失敗した各ステップの不具合を自動的に Quality Center プロジェクトに 送信するよう, QuickTest に指示できます。また, [QuickTest テスト結果] ウィ ンドウから Quality Center に直接手作業で特定のステップの不具合を送信できま す。これらのオプションは, Quality Center プロジェクトに接続している場合に のみ使用できます。

Quality Center および QuickTest を参照してください。第 47 章「Quality Center を 使用した作業」Quality Center の詳細については,『**HP Quality Center ユーザー** ズ・ガイド』(英語版)を参照してください。

Quality Center プロジェクトへの手動による不具合の送信

実行セッションの結果の表示中は、検出された不具合を、[テスト結果]ウィンドウから Quality Center プロジェクトに直接送信できます。

不具合を Quality Center に直接送信するには、次の手順を実行します。

- Quality Center クライアントがコンピュータにインストールされていることを確認します(ブラウザに Quality Center サーバの URL を入力し、ログイン画面が表示されることを確認します)。
- 2 [ツール] > [Quality Center への接続] を選択するか, [Quality Center への 接続] ボタン をクリックして, Quality Center プロジェクトに接続します。 Quality Center への接続の詳細については, 第 47 章 「Quality Center を使用した 作業」を参照してください。

注: 次のステップに進む前に Quality Center プロジェクトに接続していない場合は, QuickTest によってステップを続行する前に接続するかどうかを尋ねる メッセージが表示されます。

3 [ツール] > [不具合の追加] を選択するか, [不具合の追加] ボタン をクリックして, 指定した Quality Center プロジェクトの [不具合追加] ダイアログ・ボックスを開きます。

4 必要に応じて不具合の情報を変更できます。テストおよびチェックポイント (存在する場合)に関する基本情報は詳細領域に含まれています。

λプション
一般 フォルダ ActiveScreen 実行 Windows アプリケーション Web
実行モード ・ 標準 (実行矢印を表示)(N) ステップ実行ごとの遅延時間(D): 0 主 ミリ秒 ・ 高速(E)
 □ 失敗したステップごと(2 Quality Center (2不具合を送信する(8)) □ 実行セッション終了時に結果を表示する(2) □ 他の HP 製品でテストおよびコンポーネントを実行可能にする(20)
▼ 静止画像キャブチャをテスト結果へ保存(1): エラー発生時 ▼
· · · · · · · · · · · · · · · · · · ·
© エラーと警告の発生前(U) 2048 - KBまでのムービーセグメントを保存
○ 実行全体のムービーを保存(団)
詳細 クリックして 画面レコーダの 環境を設定

- 5 [送信] をクリックし, Quality Center プロジェクトに不具合情報を追加します。
- 6 [閉じる] をクリックし、ダイアログ・ボックスを閉じます。

Quality Center プロジェクトへの自動による不具合の送信

テストで失敗した各ステップの不具合を自動的に [Quality Center への接続] ダ イアログ・ボックス ([**ファイル**] > [**Quality Center への接続**] で表示され ます) で指定した Quality Center プロジェクトへ送信するよう, QuickTest に指 示できます。

不具合が自動的に Quality Center に送信されるようにするには、次の手順を実行します。

1 [ツール] > [オプション] を選択するか, [オプション] ボタン をクリックし ます。[オプション] ダイアログ・ボックスが開きます。 2 [実行] タブをクリックします。

オプション	×
一般 フォルダ ActiveScreen 実行 Windows アプリケーション Web	
実行モード で標準 (実行矢印を表示)(1) ステップ実行ごとの遅延時間(10): この まりわ この 高速(E)	
▼ 失敗したステップごとに Quality Center (ご不具合を送信する(S)	
▶ 実行セッション終了時に結果を表示する(少)	
□ 他の HP製品でテストおよびコンポーネントを実行可能にする(W)	
▼ 静止画像キャプチャをテスト結果へ保存(1): エラー発生時 ▼	
□ ムービーをテスト結果へ保存(M): エラー発生時 🔽	
○ エラーと警告の発生前(U) 2048 금 KBまでのムービーセグメントを保存	
実行全体のムービーを保存(E)	
詳細(A) クリックして画面レコーダの環境を設定	

- 3 [失敗したステップごとに Quality Center に不具合を送信する] チェック・ ボックスを選択します。
- **4** [**OK**] をクリックして [オプション] ダイアログ・ボックスを閉じます。

次に、不具合ごとに Quality Center に送信される情報の例を示します。

This defect was added automatically by QuickTest Professional
Standard Checkpoint "Flight Details_4" failed
Test name: Recording Test location: C:\Program Files\Mercury Interactive\QuickTest Professional \Tests\Tutorial\Recording on BINDER Action name: Action1
Operating system : Windows 2000 Host: BINDER
Additional Information: Verification type: String Content. Settings: Exact match - ON; Ignore space - ON; Match case - OFF. Results: Checked 28 cells; Succeeded: 27; Failed: 1

テスト結果での WinRunner テスト・ステップの表示

QuickTest テストに WinRunner テストの呼び出しが含まれていて, コンピュータ に WinRunner 7.6 以降がインストールされている場合, [QuickTest テスト結果] ウィンドウで WinRunner ステップの詳細な結果を参照できます。

注: Quality Center 8.0 以降を使って作成された結果の WinRunner 7.6 テスト・ス テップは, WinRunner 7.6 に WR76P44 - Support WR/QTP integration をイン ストールしないと表示できません (このパッチを WinRunner 8.0 以降にインス トールする必要はありません)。このパッチの詳細については,カスタマー・ サポートに問い合わせてください。 QuickTest テスト結果の左側の表示枠には,通常 WinRunner の結果に含まれている WinRunner イベントごとのノードが表示されます。WinRunner テストのイベントまたは関数呼び出しに対応するノードを選択すると,右側の表示枠に呼び出された WinRunner テストまたは関数のサマリと,選択したイベントの詳細が表示されます。



WinRunner テストの開始と終了は,結果ツリーにテスト実行 27 アイコン付き で表示されます。WinRunner イベントは,WinRunner 77 アイコンで表示されま す。WinRunner 関数の呼び出しは,「アイコンで表示されます。

WinRunner テストのステップを選択すると、右上の表示枠に WinRunner テスト の結果のサマリが表示されます。サマリには、テストの開始と終了時間、総実 行時間、オペレータ名、テスト中に実行されたチェックポイントの結果のサマ リが含まれています。 右下の表示枠には次の情報が表示されます。

オプション	説明
[イベント名]	選択したステップの名前です。
[結果]	ステップのステータス(成功または失敗)です。
[行番号]	WinRunner テスト内のステップの行番号です。
[イベント時間]	イベントが実行された時間です。
	選択したステップに関する追加情報が表示されます。 その後にはそのステップの WinRunner の詳細へのリン クが表示されます。 たとえば、プッシュ・ボタンの enabled プロパティを検査する GUI チェックポイントへのリンクをクリックすると、次に示すような [WinRunner] ダイアログ・ボックスが表示されます。 第 Property "enabled" for list "Fly From:" 「************************************

QuickTest からの WinRunner テストおよび関数の実行についての詳細は,第49章「WinRunner を使用した作業」を参照してください。

テスト結果の表示のカスタマイズ

QuickTest 実行セッションの結果はそれぞれ1つの.xmlファイル(results.xml と呼びます)に保存されます。この.xmlファイルには、ディスプレイのそれ ぞれのテスト結果ノードに関する情報が保存されます。これらのノードの情報 は、[テスト結果]ウィンドウの右上の表示枠に表示される.htmファイルを動 的に作成するのに使用されます。

実行結果ツリーの各ノードは results.xml ファイルの要素です。また、テスト 結果に表示される異なるタイプの情報を表す要素もあります。.xml ファイルの テスト結果情報に基づき、XSL を使用して必要な情報をカスタマイズされた形 式で表示できます([QuickTest テスト結果] ウィンドウから印刷する場合、テ スト結果を独自にカスタマイズした結果ビューアで表示する場合、テスト結果 を HTML ファイルにエクスポートする場合のいずれにおいても可能)。

次の図は, .xml ファイル内のいくつかの要素と、それらがテスト結果の中で表 す項目の対応関係を示します。



ヒント: [テスト結果] ウィンドウの外観(ルック・アンド・フィール) は変 更できます。詳細については,959ページ「[テスト結果] ウィンドウの外観の 変更」を参照してください。

XSL には、どのテスト結果情報を表示するか、それをどこにどのように表示、 印刷、エクスポートするかを指定するための手段が用意されています。また、 .xsl ファイルが参照する .css ファイルを修正して、レポートの外観(フォン ト、色など)を変更できます。

たとえば、results.xml ファイルで、ある要素タグにはアクション名が、もう1 つの要素タグには実行セッションが開始される時間についての情報が含まれて いるとします。XSLを使用すれば、カスタマイズしたテスト結果ビューアに、 アクション名をページの特定の位置に緑色の太字で表示し、時間の情報はまっ たく表示しないよう指示できます。

QuickTest が提供する既存の.xsl ファイルと.css ファイルに変更を加えるほう が,カスタマイズしたファイルを最初から作成するよりも簡単です。このファ イルは< QuickTest のインストール・フォルダ> ¥dat にあり,名前は次のと おりです。

- > PShort.xsl: [印刷] ダイアログ・ボックスまたは [HTML ファイルにエクス ポート] ダイアログ・ボックスで [簡略] オプションを選択した場合に印刷ま たは HTML ファイルにエクスポートされるテスト結果レポートの内容を指定し ます。
- PDetails.xsl:[印刷]ダイアログ・ボックスまたは[HTML ファイルにエクス ポート]ダイアログ・ボックスで[詳細]オプションを選択した場合に印刷ま たはHTML ファイルにエクスポートされるテスト結果レポートの内容を指定し ます。
- > PSelection.xsl: [印刷] ダイアログ・ボックスまたは [HTML ファイルにエク スポート] ダイアログ・ボックスで [選択範囲] オプションを選択した場合に 印刷または HTML ファイルにエクスポートされるテスト結果レポートの内容を 指定します。
- ▶ PResults.css: テスト結果の印刷プレビューの外観を指定します。このファイルは3つの.xsl ファイルすべてによって参照されています。

カスタマイズした.xsl ファイルを使用したテスト結果の印刷については,976 ページ「テスト結果の印刷」を参照してください。

カスタマイズした **.xsl** ファイルを使用した HTML ファイルへのテスト結果のエ クスポートの詳細については,979ページ「テスト結果のエクスポート」を参 照してください。

XML スキーマの構造の詳細,およびカスタマイズしたテスト結果レポートで 使用できる要素と属性の説明については,XML レポート・ヘルプ([ヘルプ] > [Quality Center ヘルプ] > [QuickTest Advanced References] > [QuickTest Test Results Schema])を参照してください。

第30章・実行セッション結果の表示

第 31 章

実行セッション結果の分析

実行セッション中に発生した重要なイベントに関するレポートを使用して、実 行セッションの結果を分析できます。

本章では、次の項目について説明します。

- ▶ テスト結果に含まれるスマート認識情報の分析(999ページ)
- ▶ チェックポイント結果の分析(1002ページ)
- ▶ パラメータ化された値と出力値の結果の表示(1025ページ)

テスト結果に含まれるスマート認識情報の分析

記録された記述では QuickTest が指定されたオブジェクトを1度で認識できず, しかもスマート認識定義が定義されている(そして有効である)場合には, QuickTest はスマート認識メカニズムを使ってオブジェクトの認識を試みます。 次の例で2つのシナリオを説明します。

スマート認識ー記録されている記述に一致するオブジェクトがない場合

QuickTest において,記録された記述に一致するオブジェクトが見つからず,ス マート認識によってオブジェクトが見つかった場合,テスト結果は警告ステー タスとなり,次の情報が含まれます。

結果ツリー	結果の詳細
見つからなかったオブジェクトに対 する記述不一致アイコン。 例を次に示 します。	オブジェクト(たとえば userName WebEdit オブジェクト)が見つからなかったことを示 します。
! 🚾 "userName"- 記述の不一致	

結果ツリー	結果の詳細
見つからなかったオブジェクトに対 するスマート認識アイコン。例を次に 示します。 ◆ "userName"- スマート認識	スマート認識メカニズムがオブジェクトを見 つけるのに成功したことと、オブジェクトを 見つけるために使われたプロパティについて の情報を示します。この情報に基づいて、記 録されたテスト・オブジェクト記述を変更 し、QuickTest が以降の実行セッションでそ の記述を使ってオブジェクトを見つけられる ようにできます。
実際に実行されたステップ。例を次に 示します。 「 userName.Set "Mercury"	実行されたステップの詳細を示す通常の結果。

スマート認識メカニズムの詳細については,第5章「オブジェクトの認識の設 定」を参照してください。

次の図に、記録されている記述プロパティ値の1つが変更された後で、スマート認識を使って userName WebEdit オブジェクトを認識したテストの結果を示します。



スマート認識ー記録された記述に複数のオブジェクトが一致する場合

記録されている記述に一致する複数のオブジェクトが見つかった後で, QuickTest がスマート認識を使って1つのオブジェクトを見つけた場合, QuickTest はテスト結果の中にスマート認識情報を示します。その場合でもス テップは合格となります。ほとんどの場合,スマート認識を使用しなくても、 テスト・オブジェクトの記述と序数識別子の組み合わせでオブジェクトを認識 できたはずだからです。

そのような場合、テスト結果には次の情報が示されます。

結果ツリー	結果の詳細
見つからなかったオブジェクトに対 するスマート認識アイコン。例を次に 示します。 ◆ "home"- スマート認識	スマート認識メカニズムがオブジェクトを見 つけるのに成功したことと、オブジェクトを 見つけるために使われたプロパティについて の情報を示します。この情報に基づいて、そ のオブジェクトのための一意のオブジェクト 記述を作成し、QuickTest がその記述を使っ て以降の実行セッションでオブジェクトを見 つけられるようにできます。
実際に実行されたステップ。例を次に 示します。 剥 home.Click	実行されたステップの詳細を示す通常の結果。

次の図に,記録された記述が変更された結果,複数のオブジェクトが適合する ようになった後で,スマート認識を使って Home オブジェクトを一意に認識し たテストの結果を示します。



スマート認識メカニズムでうまくオブジェクトを認識できない場合,テストは 失敗し,[テスト結果]に通常の失敗ステップが表示されます。

チェックポイント結果の分析

テストにチェックポイントを追加すると, Web ページ, テキスト文字列, オブ ジェクトのプロパティ, テーブルなどの期待値を, アプリケーション内のこう した要素の値と比較できます。これによって, アプリケーションが期待どおり に機能していることを確認できます。

テストを実行すると、QuickTest によって、チェックポイントの期待結果と現在の結果が比較されます。この2つの結果が一致しないと、チェックポイントは失敗となり、テストは失敗します。チェックポイントの結果は[テスト結果]ウィンドウで見ることができます。

チェックポイントの結果を表示するには、次の手順を実行します。

- 1 [テスト結果] ウィンドウでテストのテスト結果を表示します。詳細について は、960ページ「実行セッションの結果の表示」を参照してください。
- 2 [テスト結果] ウィンドウの左側の表示枠で、実行結果ツリーの分岐を展開して、結果を表示するチェックポイントの分岐をクリックします。チェックポイントの結果が[テスト結果] ウィンドウに表示されます。

注:標準設定では、[テスト結果] ウィンドウの右下には、選択したチェックポ イントに失敗のステータスが含まれる場合のみ、このチェックポイントに関する 情報が表示されます。ステップの画像を保存する条件を、[オプション] ダイア ログ・ボックスの[実行] タブで変更できます。詳細については、1181ページ 「テストの実行オプションの設定」を参照してください。

[テスト結果] ウィンドウと使用可能なオプションについての情報は,選択し たチェックポイントのタイプによって決定されます。詳細については,以下を 参照してください。

- ▶ 1004 ページ「標準チェックポイント結果の分析」
- ▶ 1005ページ「テーブル・チェックポイントとデータベース・チェックポイントの結果の分析」
- ▶ 1007 ページ「ビットマップ・チェックポイント結果の分析」
- ▶ 1008ページ「テキスト・チェックポイントまたはテキスト領域チェックポイントの結果の分析」
- ▶ 1010 ページ「XML チェックポイント結果の分析」
- ▶ 1020ページ「アクセシビリティ・チェックポイント結果の分析」
- 3 [ファイル] > [終了] を選択して, [テスト結果] ウィンドウを閉じます。

チェックポイントの詳細については,第15章「チェックポイントについて」 を参照してください。

標準チェックポイント結果の分析

標準チェックポイントをテストに追加すると、オブジェクト・プロパティの期 待値を、実行セッション中のオブジェクトの現在値と比較できます。この2つ の結果が一致しないと、そのチェックポイントは失敗となります。標準チェッ クポイントの詳細については、535ページ「オブジェクトのプロパティの値の 検査」を参照してください。

標準チェックポイントの詳細な結果は[テスト結果]ウィンドウで見ることが できます。チェックポイントの結果の表示については、「チェック・ポイント 結果の表示」を参照してください。


右上の表示枠には選択したチェックポイントの詳細結果が表示されます。ここ にはチェックポイントのステータス(成功または失敗),チェックポイントが 実行された日時,使用されたチェックポイント・タイムアウトの間隔(必要に 応じて)などが表示されます。検査対象オブジェクトのプロパティ値も表示さ れます。また,期待プロパティ値と実際のプロパティ値に違いがあればそれら もすべて表示されます。

右下の表示枠には、使用可能ならばチェックポイント・ステップの画像キャプ チャが表示されます。

前の例では、失敗したチェックポイントの詳細によって、期待結果と現在の結 果が一致しないことが示されています。フライトの出発地の期待値は「Paris」 ですが、実際の値は「Frankfurt」です。

テーブル・チェックポイントとデータベース・チェックポイントの結果の分析

テストにテーブル・チェックポイントを追加することで,アプリケーションの テーブル内のセルに,指定したテキスト文字列が表示されているかどうか検査 できます。テストにデータベース・チェックポイントを追加して,アプリケー ションからアクセスするデータベースの内容を検査できます。

テーブル・チェックポイントとデータベース・チェックポイントの結果の表示 は似ています。テストを実行すると、QuickTestにより、チェックポイントの期 待結果と、実行セッションでの実際の結果が比較されます。この2つの結果が 一致しないと、そのチェックポイントは失敗となります。

テーブル・チェックポイントおよびデータベース・チェックポイントの詳細に ついては、第18章「テーブルの検査」および第20章「データベースの検査」 を参照してください。 テーブル・チェックポイントまたはデータベース・チェックポイントの詳細な 結果は[テスト結果]ウィンドウで見ることができます。チェックポイントの 結果の表示については,1002ページ「チェックポイント結果の分析」を参照し てください。



右上の表示枠にはチェックポイント・ステップの結果が表示されます。ここに はチェックポイントのステータス(成功または失敗),チェックポイントが実 行された日時,チェックポイントに指定した検証設定,チェックポイントで成 功/失敗した個々のテーブル・セルまたはデータベース・レコードの数が表示 されます。

右下の表示枠には、チェックポイントで検査されたテーブル・セルまたはデー タベース・レコードが表示されます。検査されたセルの値やレコードは黒で、 検査されなかったセルの値やレコードは灰色で表示されます。チェックポイン トが失敗となったセルやレコードは、失敗 ★アイコンで示されます。

右下の表示枠にある [次の不一致] ボタンをクリックして、チェックポイント が失敗となった次のテーブル・セルまたはデータベース・レコードを強調表示 できます。

66



ビットマップ・チェックポイント結果の分析

テストにビットマップ・チェックポイントを追加して、キャプチャしたビット マップを照合することによって、アプリケーション内の要素の外観を検査でき ます。テストを実行すると、QuickTestにより、チェックポイントの期待結果 と、実行セッションによる実際の結果が比較されます。この2つの結果が一致 しないと、そのチェックポイントは失敗となります。チェックポイントの詳細 については、第16章「ビットマップの検査」を参照してください。

ビットマップ・チェックポイントの詳細な結果は[テスト結果]ウィンドウで 見ることができます。チェックポイントの結果の表示については、チェックポ イントの結果の表示については、「チェック・ポイント結果の表示」を参照し てください。



右上の表示枠にはチェックポイント・ステップの結果が表示されます。ここに はチェックポイントのステータス(成功または失敗),チェックポイントが実 行された日時,使用されたチェックポイント・タイムアウトの間隔(必要に応 じて)などが表示されます。

右下の表示枠には,実行セッション中に比較された期待ビットマップと実際の ビットマップが表示されます。

テキスト・チェックポイントまたはテキスト領域チェックポイントの結 果の分析

テストにテキスト・チェックポイントまたはテキスト領域チェックポイントを 追加することで、テキスト文字列がアプリケーション内の適切な場所に表示さ れていることを検査します。テストを実行すると、QuickTest により、チェック ポイントの期待結果と、実行セッションでの実際の結果が比較されます。この 2つの結果が一致しないと、そのチェックポイントは失敗となります。テキス トおよびテキスト領域チェックポイントの詳細については、第19章「テキス トの検査」を参照してください。 テキスト・チェックポイントまたはテキスト領域チェックポイントの詳細な結 果は[テスト結果]ウィンドウで見ることができます。チェックポイントの結 果の表示については、1002ページ「チェックポイント結果の分析」を参照して ください。



右上の表示枠にはチェックポイント・ステップの結果が表示されます。ここに はチェックポイントのステータス(成功または失敗),チェックポイントが実 行された日時,使用されたチェックポイント・タイムアウトの間隔(必要に応 じて)などが表示されます。また,検査された期待テキストと実際のテキス ト,およびチェックポイントに指定した検証設定が表示されます。

XML チェックポイント結果の分析

XML チェックポイントをテストに追加することで,XML ドキュメントまたは ファイル内のデータおよび構造に予定外の変更が生じていないかどうかを検証 できます。テストを実行すると,QuickTest により,チェックポイントの期待結 果と,実行セッションでの実際の結果が比較されます。この2つの結果が一致 しないと,そのチェックポイントは失敗となります。XML チェックポイントの 詳細については,第21章「XML の検査」を参照してください。

XML チェックポイントの結果の要約は [テスト結果] ウィンドウで見ること ができます。チェックポイントの結果の表示については,1002 ページ「チェッ クポイント結果の分析」を参照してください。



右上の表示枠には、チェックポイント・ステップの結果が表示されます。

右下の表示枠には,妥当性検証(適切な場合)の詳細とチェックポイント結果 のサマリが表示されます。スキーマ検証が失敗した場合は,失敗の理由も表示 されます。 チェックポイントが失敗すると、右下の表示枠の [XML チェックポイント結果 の表示] をクリックしてチェックポイントで実行される各検査を表示できます。[XML チェックポイント結果] ウィンドウが開き、そのチェックポイントの失敗の詳細が表示されます。

注:標準設定では、チェックポイントが成功した場合、[XML チェックポイント結果の表示] ボタンは利用できません。チェックポイントが成功した場合で もチェックポイントの詳細な結果を閲覧したい場合は、[ツール] > [オプ ション]を選択し、[実行] タブを選択します。[静止画像キャプチャをテスト 結果へ保存] オプションで [常に] を選択します。

[XML チェックポイント結果] ウィンドウについて

[テスト結果] ウィンドウの [XML チェックポイント結果の表示] ボタンをク リックすると, [XML チェックポイント結果] ウィンドウに XML ファイルの 階層構造が表示されます。

期待 XML ツリー表示枠には、期待結果として XML チェックポイントに格納さ れている要素、属性、および値が表示されます。

実際 XML ツリー表示枠には、実際の結果が(実行セッション中に XML ドキュ メントが実際に表示されたとおりに)表示されます。

[チェックポイントサマリ]表示枠には,期待結果の表示枠で選択されている 項目に対して実行された検査の結果情報が表示されます。 [XML チェックポイント結果] ウィンドウを開くと, [チェックポイント サマ リ] 表示枠には期待結果の表示枠で最初に検査が行われた項目についての結果 のサマリが表示されます。



[XML チェックポイント結果] ウィンドウの操作

[XML チェックポイント結果] ウィンドウには、メニューとツールバーが表示 され、これらを使用して XML チェックポイントの結果に関するさまざまなコ ンポーネントを操作できます。

次のコマンドまたはツールバー・ボタンを使用して,XML チェックポイントの結果を操作できます。

● チェックポイントのサマリを表示:XML ツリー内の要素を選択して [チェックポイントのサマリを表示] ボタンをクリックするか, [表示] > [チェックポイントのサマリ] を選択します。チェックポイント・サマリ表示枠が [XML チェックポイント結果] ウィンドウの下部に表示されます。チェックポイント・サマリ表示枠には,要素のどの部分の検査で成功または失敗したかに関する詳細が表示されます。

次は、ある XML ファイル内の cities という要素についてチェックポイント・サ マリを表示した場合の例です。

図E XML チェックボイント結果 コーイル(E) まーへの AUTが(E)	
期待 XMI ツリー	■ 実際 XMI ツリー
 ● ● ③ manyosyu ● ● ③ poem ● ● ③ poem ● ● ③ mkana ● ● ◎ mkana ● ● ● mean ● ● ● Mkana ● ● ● ● Mkana ● ● ● ● Mkana ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	 □- ③ manyosyu □- ④ volume □- ④ poem □- ④ poen □- ④ mkana □- ④ mkana □- ④ poet □- ④ poet □- ⑤ yoni □- ⑨ test(\u00fd)\u00fd)\u00fd)\u00fd) □- ⑤ yoni □- ⑨ 数田津(記報乗)世むと月待て(ば離もかなひぬ今 □- ⑤ image □- ⑨ mage/m0008.jpg □- ⑨ 数田津(こさたつ)で、船を出そうと月を待っている
チェックボイント サマリ	
チェック ステータ ス 期待 属性チェック 成功 詳細	値 実際 値 については、属性テーブルを参照してくださ
次のタイブの子数 <any 6<br="" 成功="">Child></any>	6

▶ 属性詳細の表示:XML ツリーで、属性の検査が行われた要素を選択します。 [属性詳細の表示] ボタン をクリックするか、[表示] > [属性の詳細] を選択 します。[XML チェックポイント結果] ウィンドウの下部の、[期待属性値] 表示枠と[実際属性値] 表示枠にそれぞれ、属性検査の詳細が表示されます。

次は,XML Web ページまたはフレーム内の<Action>要素に関する属性の詳細が表示されている例です。[期待属性値]表示枠には,各属性について名前,期待値,および属性検査の結果ステータスが表示されます。

[実際属性値]表示枠には、属性の名前および検査実行時に検出された実際の 値が表示されます。



- ど 次のチェックを検索: [表示] > [次のチェックを検索] を選択するか, [次の チェックを検索] ボタン をクリックすると, XML ツリー内で次に検査が行わ れた項目に直接ジャンプします。
- ▶ 前のチェックを検索: [表示] > [前のチェックを検索] を選択するか, [前の チェックを検索] ボタン をクリックすると, XML ツリー内で検査が行われた 1 つ前の項目に直接ジャンプします。

?

- ▶ 次のエラーを検索: [表示] > [次のエラーを検索] を選択するか, [次のエラーを検索] ボタンをクリックすると, XML ツリー内の次のエラーに直接 ジャンプします。
- ▶ 前のエラーを検索: [表示] > [次のエラーを検索] を選択するか, [前のエラーを検索] ボタン をクリックすると, XML ツリー内の前のエラーに直接 ジャンプします。
- ▶ ツリーの同時スクロール: [表示] > [ツリーの同時スクロール] を選択する か, [ツリーの同時スクロール] ボタン をクリックすると, [期待 XML ツリー] と [実際 XML ツリー] の両方の XML ツリーを同期させながらスクロールでき ます。このオプションが選択されていると, [期待 XML ツリー] と [実際 XML ツリー] は, ツリー構造内をたどっていくときに同時にスクロールしま す。このオプションが選択されていないと, 一度に1つのツリーだけスクロー ルできます。
 - > ヘルプトピック: [ヘルプ] > [目次と索引] を選択するか, [ヘルプトピッ ク] ボタン をクリックすると, [XML チェックポイント結果] ウィンドウのヘ ルプが表示されます。

サンプルの XML チェックポイントの結果の検証

以下では XML チェックポイントのサンプル・シナリオを4つ紹介します。そ れぞれの例では、実際の XML ドキュメントに発生した変更を示し、XML チェックポイントの結果で見つかった問題の原因を特定する方法について説明 し、該当する [XML チェックポイント結果] ウィンドウを示します。

シナリオ1:

次の例では airline という要素タグが airlines に変更されています。XML チェッ クポイントによってタグ構造に対するこの変更が検出されています。airline 要 素の子要素の検査も,親レベルの要素に不一致があったために失敗しています。

問題のあった要素の詳細を閲覧するには, [期待 XML ツリー]の中で airline タ グを選択し, [表示]> [チェックポイント サマリ]を選択します。[XML チェックポイント結果]ウィンドウの下側の表示枠にチェックポイント・サマ リが表示されます。 「指定された要素が見つかりません」というテキストは、XML ドキュメント内の airline 要素タグが変更されたことを示しています。

謳XML チェックボイント結果	
ファイル(E) 表示(V) ヘルプ(H)	
期待 XML ツリー	実際 XML ツリー
	 StSetting StSetting Sector of a contemponent Sector of
チェックポイント サマリ	<u> </u>
チェック タス 期待値	実際値
属性チェック 失敗 詳細について ルを参照して	は、属性テーブ ださい。
次のタイプの子数 <mark>失敗</mark> 0 <任意の子>	<指定された要素が見 つかりません。>
	V

シナリオ 2:

次の例は, orders 要素タグに関連付けられている属性が, 期待値 orders1 から orders2 という新しい値に変化しています。

問題のあった属性の詳細を表示するには、問題のあった要素を[期待 XML ツ リー]の中で選択し、[表示]>[**属性の詳細**]を選択します。[期待属性値] 表示枠と[実際属性値]表示枠が[XML チェックポイント結果]ウィンドウ の下部に表示されます。

[期待属性値]表示枠と[実際属性値]表示枠を使用して、問題の発生した属 性を特定し、不一致が生じている値を確認できます。



シナリオ3:

次の例は、total 要素の実際の値が実行時に変更され、チェックポイントが失敗 しています。

問題のあった値の詳細を閲覧するには,問題のあった要素を[期待 XML ツ リー]の中で選択し, [表示]> [チェックポイント サマリ]を選択します。 [XML チェックポイント結果]ウィンドウの下側の表示枠にチェックポイン ト・サマリが表示されます。 [チェックポイント サマリ]表示枠を使用して, total 要素の期待値と実際の値 とを比較できます。



シナリオ4:

次は, total 要素の値がパラメータ化され, その値の内容によってこの反復で チェックポイントが失敗した例です。

値アイコン図にシャープ記号が付いて表示され図,値がパラメータ化されていることを示しています。

問題のあった値の詳細を閲覧するには,問題のあった要素[期待 XML ツリー] の中で選択し,[**表示**] > [**チェックポイント サマリ**]を選択します。[XML チェックポイント結果] ウィンドウの下側の表示枠にチェックポイント・サマ リが表示されます。チェックポイントの結果を分析する手順は,値がパラメー タ化されていても同じです。 [チェックポイント サマリ]表示枠を使用して, total 要素の期待値と実際の値 とを比較できます。



アクセシビリティ・チェックポイント結果の分析

テストにアクセシビリティ・チェックポイントを含めると、検査した各アクセ シビリティ・オプションの結果が[テスト結果]ウィンドウに表示されます。

実行結果ツリーには、各チェックポイントで検査された各アクセシビリティ・ オプションの個々のステップが表示されます。たとえば、すべてのアクセシビ リティ・オプションを選択した場合、アクセシビリティ・チェックポイントの 実行結果ツリーは、次のように表示されます。



テスト結果の詳細には、W3Cの「Web Content Accessibility Guidelines」に準拠 していない可能性のある Web サイトの部分を特定するのに役立つ情報が含まれ ています。各検査について提供される情報は、W3C 要件に基づいています。

注: アクセシビリティ・チェックポイントに関する W3C の「Web Content Accessibility Guidelines」の一部については,次の項で引用または要約します。 ただし,ここに記載する情報は網羅的なものではありません。Web サイトが W3C の「Web Content Accessibility Guidelines」に準拠しているかどうかを検査 する場合は,<u>http://www.w3.org/TR/WAI-WEBCONTENT/</u>のドキュメント全体を 参照してください。

アクセシビリティ・チェックポイントの詳細については,『**HP QuickTest Professional アドイン・ガイド**』の Web オブジェクトのテストの項を参照して ください。

ActiveX オブジェクトのチェック

W3C の「Web Content Accessibility Guidelines」のガイドライン6では、新しい テクノロジがサポートされていない場合や無効になっている場合でも、その ページに確実にアクセスできることを要求しています。[ActiveX オブジェクト のチェック]を選択した場合、選択したページまたはフレームに ActiveX オブ ジェクト (マルチメディアを含む)が含まれているかどうか検査されます。 ActiveX オブジェクトが含まれていない場合、チェックポイントは成功となり ます。ページまたはフレームに ActiveX オブジェクトが含まれている場合は、 警告と ActiveX オブジェクトのリストが結果に表示されるので、ActiveX がサ ポートされていないブラウザでのこれらのページのアクセシビリティを検査で きます。例を次に示します。

ActiveX オブジェクトのチェック	
オブジェクト タグ	オブジェクト名
OBJECT	MSXML3
OBJECT	MQC

Alt プロパティのチェック

W3Cの「Web Content Accessibility Guidelines」のガイドライン 1.1 では、すべて の非テキスト要素に代替テキストを用意することを要求しています。 [Alt プロ パティのチェック]では、このガイドラインで Alt プロパティが必要とされる オブジェクトに実際にこの属性があるかどうか検査されます。選択したフレー ムまたはページに該当オブジェクトがない場合、またはすべての該当オブジェ クトに必要な属性がある場合、チェックポイントは成功となります。プロパ ティが必要とされる1つ以上のオブジェクトにこのプロパティがない場合、テ ストは失敗となります。テスト結果の詳細には、属性が欠けているオブジェク トのリストが表示されます。例を次に示します。

Alt ブロパティのチェック

オブジェクト タグ	オブジェクト名	Alt 値
IMG	Mercury Tours	Mercury Tours
IMG	html	[なし]
IMG	boxad1	[なし]
IMG	banner2	[なし]
IMG	Featured Destination: Aruba	Featured Destination: Aruba

キャプチャされたページまたはフレームが [テスト結果] ウィンドウ内の右下 に表示されるので, [Alt プロパティのチェック] リストに表示されたオブジェ クトを確認できます。

アプレット・オブジェクトのチェック

[アプレットのチェック]を利用すると、検査対象のページまたはフレームで Java アプレットまたは Java アプリケーションを検索することで、新しいテクノ ロジがサポートされていないか無効になっている場合でも、ページに確実にア クセスできる(W3Cの「Web Content Accessibility Guidelines」のガイドライン 6)かどうかを検査できます。ページまたはフレームに Java アプレットまたは Java アプリケーションが含まれていない場合、チェックポイントは成功となり ます。それ以外の場合は、警告と Java アプレットおよび Java アプリケーショ ンのリストが検査結果として表示されます。例を次に示します。

アブレット オブジェクトのチェック	
オブジェクト タグ	オブジェクト名
APPLET	VApplet.class

フレーム・タイトルのチェック

W3C の「Web Content Accessibility Guidelines」のガイドライン 12.1 では,フ レームの識別およびナビゲーションを容易にするように,各フレームにタイト ルを付けることを要求しています。[フレームタイトルのチェック]を選択す ると,Frame オブジェクトと Page オブジェクトに TITLE タグがあるかどうか検 査されます。選択したページまたはフレーム,およびその中のすべてのフレー ムにタイトルがある場合,チェックポイントは成功となります。選択したペー ジまたは1つ以上のフレームにタグがない場合,テストは失敗となります。テ スト結果の詳細には、タグが欠けているオブジェクトのリストが表示されま す。たとえば,次のような場合です。

フレーム タイトルの チェック			
オブジェクト クラ ス	オブジェクト タ グ	オブジェクト名	タイトルの値
Frame	IFRAME	theFrame	[なし]
Page		Test.com	Test.com

キャプチャされたページまたはフレームが [テスト結果] ウィンドウ内の右下 に表示されるので, [フレーム タイトルのチェック] リストに表示されたフ レームを確認できます。

マルチメディア・リンクのチェック

W3C の「Web Content Accessibility Guidelines」のガイドライン 1.3 および 1.4 で は、マルチメディア表現の視覚的なトラックには、その表現と同期した音声に よる説明を付けることを要求しています。また、ガイドライン 6 では、新しい テクノロジがサポートされていない場合や無効になっている場合でも、その ページに確実にアクセスできることを要求しています。[Multimedia Links Check]は、マルチメディア・オブジェクトへのリンクを識別するので、必要 に応じて代替リンクがあることを確認できます。ページまたはフレームにマル チメディア・リンクが含まれない場合、チェックポイントは成功となります。 それ以外の場合は、警告とマルチメディア・リンクのリストが検査結果として 表示されます。

サーバ側の画像のチェック

W3C の「Web Content Accessibility Guidelines」のガイドライン1.2 では、サーバ 側の画像マップのアクティブな領域ごとに、テキストによる代替リンクを用意 することを要求しています。ガイドライン9.1 では、使用可能な幾何形状で領 域を定義できない場合を除き、サーバ側の画像マップではなく、クライアント 側の画像マップを用意することを推奨しています。[サーバ側画像のチェック] を選択した場合、選択したページまたはフレームにサーバ側の画像が含まれて いるかどうか検査されます。サーバ側の画像が含まれていない場合、チェック ポイントは成功となります。ページまたはフレームにサーバ側の画像が含まれ ている場合、警告とサーバ側の画像のリストが検査の結果として表示されるの で、それぞれがガイドライン要件を満たしているかどうかを確認できます。例 を次に示します。

サーバ側画像のチェック	
オブジェクト クラス	オブジェクト名
Image	Navigation

テーブルのチェック

W3C の「Web Content Accessibility Guidelines」のガイドライン5では、アクセ ス可能なブラウザやその他のユーザ・エージェントによる変換に必要なマーク アップを確実にテーブルに含めることを要求しています。ガイドラインでは、 テーブルの主な使用目的は真の表形式データを表示するためであり、テーブル を線形化した場合でも意味が通る場合を除いて、レイアウト目的でテーブルを 使用することは避けるべきであることが強調されています。TH、TD、THEAD、 TFOOT、TBODY、COL、および COLGROUP が推奨タグです。これらのタグ を使用すれば、ユーザはユーザ・エージェントを利用して、音響手段、音声出 力、または点字画面を介して、テーブル・セル間をナビゲートしたり、ヘッダ やその他のテーブル・セル情報にアクセスしたりできます。 [Tables Check] では、選択したページまたはフレームにテーブルが含まれているかどうか検査されます。サーバ側の画像が含まれていない場合、チェックポイントは成功となります。ページまたはフレームにテーブルが含まれている場合、警告とテーブルのタグ構造の視覚的表現が検査の結果として表示されます。例を次に示します。

Table objects check		
Object Class	Object Name	Table Structure
WebTable	Table 1	

パラメータ化された値と出力値の結果の表示

パラメータ化された値に関する情報と[テスト結果]ウィンドウの出力値ス テップの結果を表示できます。また、実行時データ・テーブルの内容を表示す ることもできます。

[テスト結果] ウィンドウでのパラメータ化された値の表示

「**パラメータ**」とは、外部のデータ・ソースまたはジェネレータから値が割り 当てられる変数です。[テスト結果]ウィンドウにテストで定義されたパラ メータの値を表示できます。

パラメータ化された値を表示するには、次の手順を実行します。

- 1 テストのテスト結果を [テスト結果] ウィンドウに表示します。詳細について は、960ページ「実行セッションの結果の表示」を参照してください。
- 2 [テスト結果] ウィンドウの左側の表示枠で、テスト結果ツリーの分岐を展開 して、パラメータ化された値を含むテストまたはアクションの分岐をクリック します。

テストの場合もコンポーネントの場合も、入力パラメータの名前と値は、右側 の表示枠の下部に表示されます。

■ InputParameter [TempResults_3] - : : ファイル(E) 表示(公) ツール(T) ヘル	テスト結果 ブ(<u>H</u>)	
 ○ T Q 果 函 Q L テスト InputParameter サマリ 実行時データテーブル InputParameter 反復 1 (行 Action サマリ Action サマリ Action サマリ Action サマリ Mercury Interactive デ Mercury Interactive デ Select a Flight 	 → I? Action1 結果サマリ アクション:Action1 実行開始: 4/7/2006 - 19:24:02 実行終了: 4/7/2006 - 19:24:11 結果: 完了 	
	ステータス	
	成功	0
	失敗	0
	警告	0
	入力パラメータ UserName	値 Mercury
[F1] キーを押すと、ヘルプが表示されます。	準備完了	

前の例は, 値「Mercury」を使ってアクション用に定義された入力パラメータ「UserName」を示します。

テストでのパラメータの定義と使用については,第22章「値のパラメータ化」 を参照してください。

[テスト結果] ウィンドウでの出力値の結果の表示

「出力値」とは、実行セッション中に1つ以上の値がキャプチャされるステップです。キャプチャされた値は、実行内の別の時点で使用されます。実行中、値の1つが入力として必要とされた場合に、QuickTestによって、その値が指定された場所から取得されます。

XML 出力値のステップの結果を表示するには、次の手順を実行します。

- 1 テストのテスト結果を [テスト結果] ウィンドウに表示します。詳細について は、960ページ「実行セッションの結果の表示」を参照してください。
- *****#
- 2 [テスト結果] ウィンドウの左側の表示枠で,実行結果ツリーの分岐を展開し て,結果を表示したい出力値の分岐をクリックします。出力値の結果が[テス ト結果] ウィンドウに表示されます。

☆テスト [Res2] - テスト結果					_ 🗆 X
: ファイル(E) 表示(⊻) ツール(I) ヘルプ(Ŀ	<u>H</u>)				1
🗄 😅 🐨 🍳 🤱 🔝 🍳 🔍 📥 🔿	?				
 □-××× ⇒ テスト テスト サマリ Ⅲ 実行時データ テーブル □-×× □-×× □-×× □-× □-× □-× 	標準出力)値 "sign	-in here	ə": 完了	Ā
ਜ਼ੇ- Ø StartUp Browser ⊟- ≭ Ø Action1	日時: 2006/04,	/24 - 14:18:45			_
in the second s	詳細				_
	sign-in here 結果				
I I	プロパティ 名	キャブチャさ れた値	タイプ	名前	
	html tag	В	DataTable	signin_here_html_tag_ou	t
	L				
			準備完	7	/h

右側の表示枠には、ステータス、出力値のステップが実行された日時を含む選択された出力値の詳細な結果が表示されます。また、実行セッション中にキャプチャされた値、そのタイプおよび名前など、出力値の詳細も表示されます。

出力値の詳細については、第23章「値の出力」を参照してください。

XML 出力値のステップの結果の表示の詳細については,1029 ページ「XML 出力値の結果の分析」を参照してください。

実行時データ・テーブルの表示

テストをデータ・テーブル・パラメータまたはデータ・テーブル出力値ステッ プで実行すると、[実行時データテーブル]には、使用されたパラメータ化さ れた値と、実行中にデータ・テーブルに保存された出力値が表示されます。実 行時データ・テーブルの内容は、[テスト結果]ウィンドウで見ることができ ます。

実行時データ・テーブルを表示するには、次の手順を実行します。

- 1 テストのテスト結果を [テスト結果] ウィンドウに表示します。詳細について は、960ページ「実行セッションの結果の表示」を参照してください。
- 2 [テスト結果] ウィンドウの左側の表示枠で、「実行時データ テーブル」を強調 表示します。

🚰 テスト [Res1] - テスト結果							_ 🗆 🗙
[:] ファイル(<u>E</u>) 表示(⊻) ツール(<u>T</u>) -	ヘルプ(日)						
े 😅 🗐 🗑 🍳 😫 🔝 🔍 🔍	← →	?					
□ × 💐 テスト テスト サマリ □ <u>Ⅲ</u> 実行時データテーブル	A3) Seat	le				
□ □ 10 テスト 反復 1 (行 1)		p Item	В	С	D	Е	F▲
田…★ 3 テスト 反1度 2 (行 2)	1	Denver					
	2	New York					
	3	Seattle					
	4						
	5						
	6						
	7						-
		Global 🗡 Acti	on1/				F
[F1] キーを押すと、ヘルプが表示されます。				準備完	7		

前の例では,実行時データ・テーブルには,パラメータ化されたフライトの出 発地の値が含まれています。

実行時データ・テーブルの詳細については,第38章「データ・テーブルを 使った作業」を参照してください。

XML 出力値の結果の分析

アプリケーションで使用している XML ドキュメントのテストに要素または属 性値を出力できます。出力値の詳細については,730ページ「XML 値の出力」 を参照してください。

XML 出力値の結果の要約は [テスト結果] ウィンドウで見ることができます。 チェックポイントの結果の表示については,1002ページ「チェックポイント結 果の分析」を参照してください。

(溜テスト [Res1] - テスト結果		
÷ ファイル(E) 表示(⊻) ツール(工) ^	√レプ(円)	
🗄 😅 🐨 🌏 🞗 🔝 🔍 🔍 🕯	← → ?	
 □- 「「」 テスト テスト サマリ □- 「「」 実行時データテーブル □- 「」 テスト 反復 1 (行 1) □- ● Action1 サマリ □- ■ testxml 	XML 出力値 "test.xml": 完了 日時: 2006/04/24-14:29:35	*
🦾 🧱 Output "tests	詳細 XML出力値	
		Y
	XML 出力值	
	XML 出力値のタイブ:ファイル	
	XML 出力値の結果を表示	
		Y

右表示枠には、出力値の結果の要約が表示されます。 [XML 出力値の結果を表示] をクリックして [XML 出力値の結果] ウィンドウを開き、詳細な結果を 表示できます。 **注**:標準設定では、[XML 出力値の結果を表示]はエラーが発生した場合のみ 使用できます。すべての実行後の出力値の詳細な結果を見る場合は、[ツール] > [オプション]を選択し、[実行]タブを選択します。[Save still image captures to results] オプションで[常に]を選択します。

XML 出力値の結果の詳細については,1030ページ「[XML 出力値の結果] ウィンドウについて」を参照してください。

[XML 出力値の結果] ウィンドウについて

[テスト結果] ウィンドウの [XML 出力値の結果の表示] ボタンをクリックすると, [XML 出力値の結果] ウィンドウに XML ファイルの階層構造が表示されます。

データ・テーブル名表示枠には,XML 出力値の設定(データ・テーブル出力 値の出力用に選択した XML の構造とデータ・テーブル・カラム名)が表示さ れます。

出力値表示枠には、実際の XML ツリー(XML ドキュメントやファイルの実際の見え方や実行中に出力された実際の値)が表示されます。

評XML 出力値の結果 _ 🗆 X ファイル(E) 表示(V) ヘルプ(H) **0** 🔐 🖓 🕼 🎹 ? データテーブル名 出力値 **_**... manyosyu **.**... 🐵 manyosyu 🚊 🗠 🙁 volume 🚊 🗠 🐼 volume 🚊 🗠 🥵 poem 🚊 🗠 🥵 poem 🗄 -- 🙁 pno 🗄 ··· 🙁 pno 🕅 8 🗄 ··· 🐵 mkana 🚊 – 🐵 mkana . ⊨... **⊗** poet . ⊨... **⊗** poet 🔤 🔣 test(ぬかたのおおきみ) 🔤 🔽 DataTable ("value_poet_out1", dtG 🗄 -- 🐼 yomi 🗄 -- 🐼 yomi 🦾 🗹 🕅 DataTable (‴value_yomi_out", dtG - …… 図 熟田津に船乗りせむと月待て(image (₩ Ė… ⟨⊠> image 🛄 image/m0008.jpg 🛄 image/m0008.jpg 🗄 -- 🙁 mean . ⊡… <⊠> mean 🛄 熟田津(にきたつ)で、船を出そうと月を得 🦾 🔟 熟田津(にきたつ)で、船を出そ • F その他の詳細 チェック ステータス 期待值 実際値 値 ステップの実行完了

[その他の詳細] 表示枠には、選択された項目の結果情報が表示されます。

[XML 出力値の結果] ウィンドウの操作

[XML 出力値の結果] ウィンドウには、メニューとツールバーが表示され、これらを使用して XML 出力値の結果に関するさまざまなコンポーネントを操作できます。

次のコマンドまたはツールバー・ボタンを使用して,XML 出力値の結果を操作できます。

■ 出力値のサマリを表示:XML ツリー内の要素を選択して [出力値のサマリを 表示]ボタンをクリックするか,[表示]> [出力値のサマリ]を選択します。 [その他の詳細]表示枠が [XML 出力値の結果]ウィンドウの下部に表示され ます。この表示枠には,選択された要素の出力値,属性,値に関する詳細が表 示されます。

Image: State	
ファイル(E) 表示(V) ヘルプ(H)	
データテーブル名	出力値
□ 《型》manyosyu □ 《型》poem □ 《型》poem □ 《型》pno □ 《型》mkana □ 《型》mkana □ 《型》mkana □ 《型》mkana □ 《型》mkana □ 《型》mkana □ 《型》mkana □ 《型》poet □ 《型》mi □ 《型 image □ 《型 image/m0008.jpg □ 《型 mean □ 《型 mean □ 《型 ME注(zきたつ)で、船を出そうと月を得	 □ 《型> manyosyu □ 《型> poem □ 《型> pno □ 《型> mkana □ 《□> mkana □ 《□> mkana □ 《□> poet □ 《□> image □ 《□> mean □ 《□> mean □ (□> 熱田津(にきたつ)で、船を出そ
その他の詳細	
チェック ステータス 値 ステップの実行完了	期待值実際值
	_

r

▶ 属性詳細の表示:XML ツリーで、属性が値として出力された要素を選択します。[属性の詳細]ボタンをクリックするか、[表示] > [属性の詳細]を選択します。[XML 出力値の結果]ウィンドウの下部の、期待属性値表示枠と実際属性値表示枠にそれぞれ、属性の出力値の詳細が表示されます。

[期待属性値]表示枠には、各属性名と期待値、または出力値名が表示されま す。[実際属性値]表示枠には、属性の名前と各属性の実行セッション中の実 際値が表示されます。



- ▶ 次の出力値を検索: [表示] > [次の出力値を検索] を選択するか, [次の出力 値を検索] ボタン をクリックすると, XML ツリー内の次の出力値に直接ジャンプします。
- ▶ 前の出力値を検索: [表示] > [前の出力値を検索] を選択するか, [前の出力 値を検索] ボタン をクリックすると, XML ツリー内の1つ前の出力値に直接 ジャンプします。

- ▶ 前のエラーを検索: [表示] > [次のエラーを検索] を選択するか, [前のエラーを検索] ボタン をクリックすると, XML ツリー内の前のエラーに直接 ジャンプします。
- ▶ ツリーの同時スクロール: [表示] > [ツリーの同時スクロール] を選択する か, [ツリーの同時スクロール] ボタン をクリックすると, [データ テーブル 名] ツリーと [出力値] ツリーのスクロールを同期できます。

このオプションが選択されていると、[**データ テーブル名**] と [**出力値**] は、 ツリー構造内をたどっていくときに同時にスクロールします。このオプション が選択されていないと、一度に1つのツリーだけスクロールできます。

▶ ヘルプトピック: [ヘルプ] > [目次と素引] を選択するか, [ヘルプトピック] ボタン をクリックすると, [XML 出力値の結果] ウィンドウのヘルプが表示されます。

第 VII 部

テストの保守およびデバッグ

第 32 章

テストと関数ライブラリのデバッグ

実行セッションを制御しデバッグすることで、テスト、関数、および登録され たユーザ関数内の不具合を特定して処理できます。

本章では、次の項目について説明します。

- ▶ テストと関数ライブラリのデバッグについて(1038ページ)
- ▶ デバッグ・セッションの速度調節(1039ページ)
- ▶ シングル・ステップ・コマンドの使用(1040ページ)
- ► [ステップまで実行] コマンドおよび [ステップからデバッグ] コマンドの 使用(1043ページ)
- ▶ 実行セッションの一時停止(1046ページ)
- ▶ ブレークポイントの使用(1046ページ)
- ➤ デバッグ・ビューアの使用(1049ページ)
- ▶ 実行エラーの処理(1052ページ)
- ▶ アクションまたは関数のデバッグの練習(1054ページ)

テストと関数ライブラリのデバッグについて

テストまたは関数ライブラリ(登録済みユーザ関数を含む)の作成が終わった ら、構文やロジックにエラーがなく、スムーズに実行されることを確認する必 要があります。関数ライブラリをデバッグするには、まず関数ライブラリをテ ストに関連付け、次にそのテストから関数ライブラリをデバッグする必要があ ります。

テストまたは関数ライブラリ内の不具合を検出し,切り分けるには,[**一時停** 止] コマンド,および特定のステップに対するステップ・イントゥ,ステッ プ・オーバー,ステップ・アウトが可能な各種ステップ・コマンドを使用して 実行セッションを制御します。

[ステップからデバッグ] コマンドを使用すれば,テスト内の特定のポイント でデバッグ・セッションを開始できます。また,[ステップまで実行] コマン ドを使用して,テスト内の特定のポイントで実行を一時停止できます。ブレー クポイントの設定後,テストまたは関数ライブラリのさまざまな部分をデバッ グする際にブレークポイントを有効にしたり無効にしたりできます。

テストまたは関数ライブラリの実行がブレークポイントで停止したら,[デ バッグビューア]を使って、VBScriptのオブジェクトや変数の値を確認した り、変更したりできます。また、実行セッション中に、QuickTestによって実行 エラーのメッセージが表示された場合、エラー・メッセージの[**デバッグ**] ボ タンをクリックし、実行を一時停止して、テストまたは関数ライブラリをデ バッグすることもできます。

さらに,選択したステップから最後までテストまたは関数ライブラリを実行す るには, [ステップから実行] コマンドを使用します。これにより,アプリ ケーションの特定のセクションだけを検査したり,テストまたは関数ライブラ リの特定の部分がスムーズに実行されるかどうかを確認したりできます。詳細 については,943ページ「テストの一部の実行」を参照してください。

ヒント:画面レコーダを使って、テスト中のアプリケーションのムービーを キャプチャできます。詳細については、970ページ「アプリケーションの静止 画像およびムービーのキャプチャと表示」を参照してください。

注:

- ▶ テストおよび関数ライブラリは、デバッグ・モードで実行されているときは 読み取り専用です。内容の変更は、デバッグ・セッションの停止後(一時停 止時ではなく)にできます。必要に応じて、セッションの停止後に関数ライ ブラリを編集可能にできます([ファイル] > [編集可能にする])。詳細に ついては、902ページ「読み取り専用の関数ライブラリの編集」を参照して ください。変更後、テストおよび関数ライブラリのデバッグを続行できます。
- ➤ ファイル操作を実行すると(たとえば、別のテストを開いたり、新規のテストを作成したりするなど)、デバッグ・セッションは停止します。
- ➤ ExecuteFile ステートメントを使用して呼び出されたファイルまたは当該 ファイル内に含まれている関数はデバッグできません。また, ExecuteFile ステートメントを含んでいるテストをデバッグする場合,実行マーカが正し く表示されないことがあります。
- QuickTest では、テストを開くと、Quality Center プロジェクトに保存した外部リソースのローカル・コピーが作成されます。したがって、関数ライブラリなど、Quality Center プロジェクトに保存されている外部リソースに適用した変更は、テストを閉じて再度開くまでテストに反映されません(外部リソースとは、外部エディタで作成された関数ライブラリなど、QuickTest を使って作成されていないリソースです)。これに対して、関数ライブラリなどファイル・システムに保存されている外部リソースに適用した変更は、ただちに実装されます。これらのファイルは直接アクセスされ、テストを開いたときにローカル・コピーとして保存されないためです。

デバッグ・セッションの速度調節

実行セッション中,通常 QuickTest はステップを高速で実行します。テストまた は関数ライブラリのデバッグ中は,必要に応じて実行を一時停止したり別のタ スクを実行したりできるように,ステップをもっとゆっくり実行したいときが あります。各ステップ間で QuickTest が一時停止する時間(ミリ秒)を指定でき ます。指定するには,[オプション]ダイアログ・ボックス([ツール] > [オ プション])の[実行]タブにある [ステップ実行ごとの遅延時間]オプション を変更します。[実行] タブのオプションの詳細については,1181ページ「テス トの実行オプションの設定」を参照してください。

シングル・ステップ・コマンドの使用

[ステップイントゥ] コマンド, [ステップアウト] コマンド, および [ス テップオーバー] コマンドを使用して, テストまたは関数ライブラリのステッ プを1つだけ実行できます。

ヒント:デバッグ・ツールバーを表示するには, [表示] > [ツールバー] > [デバッグ] を選択します。

[ステップ イントゥ]

アクティブなテストまたは関数ライブラリの現在の行だけを実行するには、 [デバッグ] > [ステップイントゥ] を選択するか、[ステップイントゥ] ボ タンをクリックするか、または F11 キーを押します。アクティブなテストまた は関数ライブラリの現在の行が、ほかのアクションまたは関数を呼び出してい る場合には、呼び出し先のアクションまたは関数が [QuickTest] ウィンドウに 表示され、呼び出し先のアクションまたは関数の最初の行でテストまたは関数 ライブラリが一時停止します。

[ステップ アウト]

[ステップイントゥ]を使用して、アクションまたはユーザ定義関数に入った 後にだけ、[デバッグ] > [ステップアウト]を選択するか、[ステップアウ ト] ボタンをクリックするか、または SHIFT キーを押しながら F11 キーを押し ます。[ステップアウト] によって、呼び出し先のアクションまたはユーザ定 義関数が最後まで実行された後、呼び出し元のテストまたは関数ライブラリに 戻り、実行セッションが一時停止します。

[ステップ オーバー]

アクティブなテストまたは関数ライブラリの現在のステップだけを実行するに は、[**デバッグ**] > [**ステップオーバー**]を選択するか、[**ステップオーバー**] ボタンをクリックするか、または F10 キーを押します。現在のステップで、別 のアクションやユーザ定義関数が呼び出されている場合は、呼び出し先のアク ションまたは関数の全体が実行されますが、呼び出し先のアクションまたは関 数スクリプトは QuickTest ウィンドウに表示されません。



¢≣

Ç≣
シングル・ステップ・コマンドの使用 – 例

サンプルのテストまたは関数ライブラリを作成し, [ステップイントゥ], [ス テップアウト], [ステップオーバー] コマンドを使ってそれを実行するには, 次の手順に従います。

サンプルのテストまたは関数ライブラリを作成するには、次の手順を実行します。

- 新しいテストを開くには、[ファイル] > [新規作成] > [テスト] を選択します。(テストのほかに)新しい関数ライブラリを開くには、[ファイル] > [新規作成] > [関数ライブラリ] を選択します。
- 2 新しいテストを作成した場合は, [エキスパート ビュー] タブをクリックして エキスパート・ビューを表示します。
- 3 テストまたは関数ライブラリで、次の行を正確に入力します。

public Function myfunc() msgbox "one" msgbox "two" msgbox "three" End Function

myfunc myfunc myfunc

- 4 次の手順のいずれかを実行します。
 - ▶ テストを作成した場合は次の手順に進みます。次の手順では、「ステップイントゥ」コマンド、「ステップアウト」コマンド、「ステップオーバー」コマンドを使用してテストまたは関数ライブラリを実行する方法を説明します。
 - ▶ 関数ライブラリを作成したら、SampleFL.qfl という名前でファイル・シス テムまたは Quality Center プロジェクトに保存します(詳細については、897 ページ「関数ライブラリの保存」を参照してください)。

次に, [ファイル] > [ライブラリ '< 関数ライブラリ名>' を '< テスト名>' に関連付ける] を選択して, 関数ライブラリをテストに関連付けます。

[ステップイントゥ] コマンド, [ステップアウト] コマンド, [ステップオー バー] コマンドを使用してテストまたは関数ライブラリを実行するには, 次の 手順を実行します。

- テストまたは SampleFL.qfl 関数ライブラリをまだ開いていない場合はこれを 開くか、テストまたは SampleFL.qfl 関数ライブラリのタブをクリックして、 そこにフォーカスを移動します。
- 2 F9 キー([**ブレークポイントの設定 / 解除**])を押して,テストまたは関数ライ ブラリの7 行目(最初の myfunc 関数の最初の呼び出し)にブレークポイント を追加します。ブレークポイント記号が左マージンに表示されます。詳細につ いては,1047 ページ「ブレークポイントの設定」を参照してください。
 - 3 テストを実行します。テストまたは関数ライブラリはブレークポイントで一時 停止します。
 - **4** F11 キー([ステップイントゥ])を押します。実行矢印が、関数の最初の行 (msgbox "one")をポイントします。
 - 5 F11 キー([ステップイントゥ])をもう一度押します。メッセージ・ボックス に、「one」というテキストが表示されます。
 - 6 [OK] をクリックし、メッセージ・ボックスを閉じます。実行矢印が、関数の 次の行へ移動します。
 - 7 実行矢印が関数から移動してスクリプトの8行目(2回目の myfunc 関数呼び出し)をポイントするまで,F11キー([ステップイントゥ])を押し続けます。
 - 8 F11 キー([ステップイントゥ])を押して、もう一度関数を入力します。実行 矢印によって、関数の最初の msgbox 行がポイントされます。
 - 9 SHIFT + F11 キー([ステップアウト])を押します。3つのメッセージ・ボックスが開きます。3つのうち最後のメッセージ・ボックスを閉じるまで、実行矢印は関数の最初の行をポイントし続けます。3つめのメッセージ・ボックスを閉じると、実行矢印はテストの最後の行へ移動します。
 - 10 F10 キー([ステップオーバー])を押します。もう一度,3つのメッセージ・ ボックスが開きます。実行矢印はテストの最後の行に残ります。

[ステップまで実行] コマンドおよび [ステップからデバッグ] コ マンドの使用

デバッグ中は、ステップに対するステップ・イントゥ、ステップ・アウト、ス テップ・オーバのほかに、[**ステップまで実行**] コマンドと [**ステップからデ バッグ**] コマンドを使って、特定のステップに達するまでテストまたはアク ション (関連付けられた関数ライブラリを含む)を実行するように、または特 定のステップからデバッグを開始するように QuickTest に指示できます。

[ステップまで実行]

テストまたはアクション(エキスパート・ビューのみ)の最初から,あるいは テストまたはアクションの現在の位置から実行するように指定できます。ま た,特定のステップで停止するように指示することもできます。これは,ス テップに一時的なブレークポイントを追加するのに似ています。たとえば,テ ストまたはアクションおよび関連付けられている関数ライブラリをデバッグ・ モードで1ステップずつ実行しているとき,連続する4ステップを実行してか ら5番目のステップで停止したい場合などがあります。

このオプションは、テストまたはアクションの編集またはデバッグ中に使用できます。

特定のステップまで実行するように指定するには、次の手順を実行します。

- ▶ 実行を停止するステップにカーソルを挿入し、[デバッグ] > [ステップまで実行] を選択するか、CTRL キーを押しながら F10 キーを押します。
- ▶ 実行を停止するステップを右クリックし、ショートカット・メニューから [ステップまで実行]を選択します。

注: テストの編集中に [**ステップまで実行**] オプションを使用すると, [実行] ダイアログ・ボックスが開きます。このダイアログ・ボックスでは, デバッグ 実行セッションの結果の格納場所と入力パラメータ値を指定できます。詳細に ついては, 次の「[ステップからデバッグ]」の手順2を参照してください。

[ステップからデバッグ]

テストまたはアクションの最初から実行を開始するのではなく、特定のステッ プからデバッグ・セッションを開始するように指定できます。特定のステップ からデバッグを開始する場合は、その前に、デバッグを開始する位置に合わせ てアプリケーションが開いていることを確認してください。テストまたはアク ションの編集中にテストまたはアクション内の特定のステップからデバッグを 開始できます。

特定のステップから実行するように指定するには、次の手順を実行します。

- 1 デバッグを開始するステップを選択します。
 - ▶ 実行を開始するステップにカーソルを挿入し、[デバッグ] > [ステップからデバッグ] を選択します。または、
 - ▶ 実行を開始するステップを右クリックし、コンテキスト・メニューの [ス テップからデバッグ] を選択します。

[実行] ダイアログ・ボックスが表示されます。

実行	×
結果保管場所 入力パラメータ	
実行結果の書き込み先:	
○ 新規実行結果フォルダ(№)	±11
 ● 一時実行結果フォルダ(古い一時結果に上書き)[<u>T</u>] □:¥DOCUME~1¥ADMINI~1¥LOCALS~1¥Temp¥TempResults 	
	パ

注: QuickTest テストが(ファイル・システムではなく)Quality Center プロジェ クトに保存されている場合は、[実行] ダイアログ・ボックスの [結果保管場 所] タブに、プロジェクトに関する追加オプションが表示されます。詳細につ いては、1346 ページ「Quality Center プロジェクトに格納されているテストの QuickTest からの実行」を参照してください。

2 必要に応じて、デバッグ実行セッションに結果の格納場所と入力パラメータの 値を指定します。標準設定では、[一時実行結果フォルダ]オプションが選択 されます。

[実行] ダイアログ・ボックスのタブの詳細については,941ページ「[結果保 管場所] タブについて」および942ページ「[入力パラメータ] タブについて」 を参照してください。

3 [OK] をクリックします。[実行] ダイアログ・ボックスが閉じ,デバッグ実行 セッションが開始されます。[ステップイントゥ], [ステップオーバー], [ス テップアウト] などの任意の QuickTest デバッグ・オプションを使用できます。 標準では,実行セッションが終了すると,[テスト結果]ウィンドウが開きま す。実行結果の表示の詳細については,第 30 章「実行セッション結果の表示」

を参照してください。

注: [オプション] ダイアログ・ボックスの [実行] タブで [**実行セッション** 終**7時に結果を表示する**] チェック・ボックスをクリアすると,実行セッションの最後で [テスト結果] ウィンドウは開きません。[オプション] ダイアロ グ・ボックスの詳細については,第40章 「グローバル・テスト・オプション の設定」を参照してください。

実行セッションの一時停止

Ш

実行セッションを一時的に停止するには、[**デバッグ**] > [**一時停止**] を選択 するか、[**一時停止**] ボタンをクリックします。一時停止されたテストまたは 関数ライブラリは、それまでに解釈されたすべてのステップが実行された後に 実行を停止します。

ー時停止した実行を再開するには、[実行]ボタンをクリックするか、[オート メーション]>[実行]を選択するか、またはF5キーを押します。一時停止 されたところから、実行が再開されます。

ヒント:実行セッションを停止するには、[**停止**] ボタンをクリックするか、 [**オートメーション**] > [**停止**] を選択します。実行セッションの最後に結果 を表示しないように選択していなければ([**ツール**] > [**オプション**] > [**実** 行] タブ)、実行セッションの停止後、[テスト結果] ウィンドウが開きます。

ブレークポイントの使用

ブレークポイントを使用すると、テストまたは関数ライブラリ内のあらかじめ定 義した位置で実行セッションを一時停止するよう QuickTest に指示できます。ブ レークポイントに達すると、ステップを実行する前に実行が一時停止されます。 ブレークポイントまでの実行の結果を検証し、必要であれば変更を加えた上で、 ブレークポイントからテストまたは関数ライブラリの実行を再開できます。

ブレークポイントは次のような目的に使用できます。

- ▶ 実行セッションを一時停止し、アプリケーションの状態を点検する。
- ▶ ステップ・コマンドを使ってテストまたは関数ライブラリをステップ実行する 開始位置を指定する。

ブレークポイントを設定し、そのブレークポイントを一時的に有効にしたり無 効にしたりできます。ブレークポイントの使用が終了したら、テストまたは関 数ライブラリからブレークポイントを削除できます。

注: ブレークポイントは,現在の QuickTest セッションにのみ適用され,テストまたは関数ライブラリには保存されません。

ブレークポイントの設定

ブレークポイントを設定すると、実行セッションをテストまたは関数ライブラ リ内のあらかじめ定義した位置で実行セッションを一時停止できます。ブレー クポイントは、選択したステップの横の左マージンに赤い丸のアイコンで示さ れます。

ブレークポイントを設定するには、次の手順を実行します。

次の手順のいずれかを実行します。

1

- ▶ テストまたは関数ライブラリ内で、実行を停止するステップの左マージンを クリックします。または、
- ▶ ステップをクリックし、次の作業を行います。
 - ► [**ブレークポイントの設定 / 解除**] ボタンをクリックします。
 - ▶ [デバッグ] > [ブレークポイントの設定 / 解除] を選択します。
- ブレークポイント記号がテストまたは関数ライブラリの左マージンに表示され ます。

ヒント: [**ブレークポイントの有効化/無効化**] オプションを使用して,ステッ プにブレークポイントを追加することもできます。詳細については,1048ペー ジ「ブレークポイントの有効化と無効化」を参照してください。

ブレークポイントの有効化と無効化

ブレークポイントを一時的に無効にすることによって、デバッグ・セッション 中に既存のブレークポイントを無視するように指定できます。指定した場合、 テストまたは関数ライブラリの実行時、QuickTest はブレークポイントが含まれ るステップで停止せず、そのステップを実行します。ブレークポイントを再度 有効にすると、次の実行時にそこで一時停止します。この機能は、テストまた は関数ライブラリに多くのステップがあり、その特定部分をデバッグする必要 がある場合に特に役立ちます。

ブレークポイントの有効化および無効化は、個別に、またはすべて一度に行う ことができます。たとえば、テストまたは関数ライブラリ全体にわたってさま ざまなステップにブレークポイントを追加し、なおかつ当面はドキュメントの 特定の部分のみデバッグしたいとします。この場合、テストまたは関数ライブ ラリ内のすべてのブレークポイントを無効にし、それから特定ステップのブ レークポイントのみ有効にします。ドキュメントの対象セクションのデバッグ が済んだら、有効にしたブレークポイントを無効にし、(デバッグしたいセク ションの)次のブレークポイントのセットを有効にします。ブレークポイント は無効にされていても削除はされていないため、任意のブレークポイントを必 要に応じて探して有効にできます。

有効になっているブレークポイントは、選択したステップの左マージンに赤い 丸のアイコン
■で示されます。

無効になっているブレークポイントは、選択したステップの横の左マージンに 白抜きの丸のアイコン〇で示されます。

特定のブレークポイントを有効または無効にするには、次の手順を実行します。

- 1 有効または無効にするブレークポイントが含まれている行をクリックします。
- 2 [デバッグ] > [ブレークポイントの有効化/無効化] を選択するか, CTRL+F9 キーを押します。ブレークポイントが有効または無効になります(それまでの 状態に応じて異なります)。

すべてのブレークポイントを有効または無効にするには、次の手順を実行します。

1

1

[デバッグ] > [すべてのブレークポイントの有効化/無効化] を選択するか, [全ブレークポイントの有効化/無効化] ボタンをクリックします。少なくとも 1 つのブレークポイントが有効になっている場合は,テストまたは関数ライブラ リ内のすべてのブレークポイントが無効になります。逆に,すべてのブレーク ポイントが無効になっている場合は,すべてのブレークポイントが有効になり ます。

ブレークポイントの削除

現在のテストまたは関数ライブラリに定義されている1つのブレークポイン ト,またはすべてのブレークポイントを削除できます。

1つのブレークポイントを削除するには、次の手順を実行します。

次の手順のいずれかを実行します。

- ▶ ブレークポイントをクリックします。
- ▶ テストまたは関数ライブラリ内のブレークポイント記号のある行をクリックし、次の操作を行います。
 - ▶ [**ブレークポイントの設定 / 解除**] ボタンをクリックします。
 - ▶ [**デバッグ**] > [**ブレークポイントの設定 / 解除**] を選択します。

QuickTest ウィンドウの左マージンから、ブレークポイント記号が削除されます。

すべてのブレークポイントを削除するには、次の手順を実行します。

「すべてのブレークポイントを削除]ボタンをクリックするか、「デバッグ]> 「すべてのブレークポイントを削除]を選択します。QuickTest ウィンドウの左 マージンから、すべてのブレークポイント記号が削除されます。

デバッグ・ビューアの使用

テストまたは関数ライブラリがブレークポイントで停止した場合,あるいはス テップが失敗して**デバッグ・**オプションを選択した場合に,デバッグ・ビュー ア・ウィンドウを使用してテストまたは関数ライブラリ内のオブジェクトや変 数の現在の値を表示,設定,または変更できます。 4

デバッグ・ビューア・ウィンドウを開くには、次の手順を実行します。

[表示] > [デバッグ ビューア] を選択するか, [デバッグ ビューア] ボタン をクリックします。デバッグ・ビューア・ウィンドウが開きます。



デバッグ・ビューアのタブには、現在のアクションのメイン・スクリプトまた は選択したサブルーチン内にある変数やオブジェクトの値が表示されます。 [**コンテキスト**] ボックスでは、アクションのメイン・スクリプト(VBScript: global code) と、アクションのサブルーチンおよび関数のいずれかを選択でき ます。

[ウォッチ] タブ

変数あるいは VBScript オブジェクトを [ウォッチ] タブに追加することによっ て、テストまたは関数ライブラリ内の変数あるいは VBScript オブジェクトの現 在の値を表示できます。テストまたは関数ライブラリ内の後続のステップを引 き続き実行すると、自動的に [ウォッチ] タブが更新され、値が変更されたオ ブジェクトまたは変数の現在の値が表示されます。テストまたは関数ライブラ リがブレークポイントで一時停止している場合は、変数の値を手動で変更する こともできます。

[ウォッチ] タブに式を追加するには、次の手順を実行します。

次の手順のいずれかを実行します。

- ▶ 式をクリックして、[デバッグ] > [ウォッチに追加] を選択します。
- ▶ 式をクリックして, CTRL + T キーを押します。
- ▶ 式を右クリックして、ショートカット・メニューから [ウォッチに追加] を 選択します。
- ▶ [ウォッチ] タブで、[名前] カラムにオブジェクトまたは変数の名前を貼り 付けるか、または入力します。ENTER キーを押すと、[値] カラムに現在の 値が表示されます。

注:エキスパート・ビューまたは関数ライブラリから [ウォッチ] タブに式を 追加できます。

[変数] タブ

[変数] タブには、テストまたは関数ライブラリが停止または一時停止するポイ ントまでの、現在のアクションまたは関数のすべての変数の現在の値が自動的 に表示されます。たとえば、関数をステップ実行している場合、各ステップを 実行すると、[変数] タブのグリッドにステップの変数の現在の値が追加されま す。後続のステップを引き続き実行すると、変数に変更があった場合、[変数] タブに表示されている変数の値が自動的に更新されます。ブレークポイントで 一時停止している間は、変数の値を手作業で変更することもできます。

[コマンド] タブ

[コマンド] タブは、テストまたは関数ライブラリに含まれている変数または VBScript オブジェクトの現在の値を設定または変更するために、1 行のスクリ プトを実行する場合に使用します。実行を継続すると、設定した値が QuickTest によって使用されます。

実行エラーの処理

実行セッション中に表示できる[実行エラー]メッセージ・ボックスには、2 つのタイプがあります。1つは、問題が VBScript の純粋な構文エラーである場 合に表示されます。構文実行エラー・メッセージが表示された場合は、メッ セージ・ボックスの [OK] をクリックし、ステップ内のエラーを解決します。

もう1つのメッセージ・ボックスは、さまざまな状況で表示され、次のように エラーに関する情報と発生したエラーを処理するためのいくつかのボタンを提示します。



- ▶ [停止]: テスト実行を停止します。実行後に実行結果を表示するように OuickTest が設定されていれば、実行結果が表示されます。
- ▶ [再試行]: QuickTest によってステップの実行が再試行されます。ステップが成功すると、実行が再開されます。
- ▶ [無視]: QuickTest によって、エラーが発生したステップが無視され、その次の ステップから実行が再開されます。
- ▶ [デバッグ]: QuickTest によって実行が中断され、テストおよびテストから呼び 出された関数が含まれる関連する関数ライブラリをデバッグできます。

本章で説明している任意のデバッグ操作を実行できます。デバッグが完了した ら、テストまたは関数ライブラリが停止したステップから実行セッションを再 開したり、ステップ・コマンドを使って残りの実行セッションを制御したりで きます。

▶ [ヘルプ]:表示されたエラー・メッセージに関する QuickTest のトラブル シューティング用ヘルプが表示されます。ヘルプ・トピックを読んだら、メッ セージ・ボックスで別のボタンを選択できます。

メッセージ・ボックスには、アプリケーションの意図的な変更がエラーの原因と 考えられ、テスト内の複数のステップやリポジトリ内のオブジェクトを更新する 必要がある場合は、メンテナンス・モードの使用を検討するように推奨するメッ セージも表示されます。詳細については、1060ページ「メンテスナンス実行 ウィザードを使ったテストの実行」を参照してください。

アクションまたは関数のデバッグの練習

テストまたは関数ライブラリのほかの部分で使用される変数を定義するアク ションまたは関数を作成するとします。アクションまたは関数にブレークポイ ントを追加することで、テストまたは関数ライブラリの実行中に変数の値がど のように変化するかを確認できます。ブレークポイントで一時停止している間 に、いずれかの変数の値を変更することによって、テストまたは関数ライブラ リでその新しい値がどのように処理されるかを調べることもできます。

手順1:新しいアクションまたは関数を作成する

テストを開いて新しいアクションを挿入します。または、新しい関数ライブラ リを開いて、SetVariablesという新しい関数を作成します。アクションの挿入 の詳細については、第14章「アクションを使った作業」を参照してください。 関数での作業の詳細については、第28章「ユーザ定義関数および関数ライブ ラリを使用した作業」を参照してください。

エキスパート・ビューまたは関数ライブラリで,次のように VBScript コードを 入力します。

Dim a a="hello" b="me" MsgBox a

Function Library
Function Set∀ariables
Dim a a="hello" b="me" MsgBox a
EndFunction

エキスパート・ビューの詳細については,第26章「エキスパート・ビューお よび関数ライブラリ・ウィンドウを使用した作業」を参照してください。

注:エキスパート・ビューで作業している場合は、手順4に進みます。関数ラ イブラリで作業している場合は、手順2と手順3に進みます。

手順2:(関数ライブラリの場合のみ)関数ライブラリをテストに関連付ける

- 1 関数ライブラリにフォーカスがあることを確認します(フォーカスがない場合は、関数ライブラリのタブをクリックするか、[ウィンドウ]メニューから関数ライブラリを選択して、関数ライブラリをアクティブにします)。
- 2 [ファイル]>[ライブラリ'く関数ライブラリ名>'を'くテスト名>'に関連付ける]を選択するか、右クリックして [ライブラリ'く関数ライブラリ名>' を'くテスト 名>'に関連付ける]を選択します。関数ライブラリがテストに 関連付けられます。

手順3:(関数ライブラリの場合のみ)関数の呼び出しを追加する

関数の呼び出しを追加するには、新しいステップを挿入し、次のように入力し ます。

SetVariables

手順4:ブレークポイントを追加する

b="me" および MsgBox a というテキストが含まれている行にブレークポイント を追加します。ブレークポイントの追加の詳細については,1047 ページ「ブ レークポイントの設定」を参照してください。

手順5:テストの実行を開始する

テストを実行します。テストまたは関数ライブラリは、対象のステップ(スク リプトの行)を実行する前の最初のブレークポイントで停止します。

手順6:デバッグ・ビューア・ウィンドウで変数の値を検査する

- 1 [表示] > [デバッグ ビューア] を選択して、デバッグ・ビューア・ウィンド ウを開きます(まだ開いていない場合)。デバッグ・ビューア・ウィンドウで [ウォッチ] タブを選択します。
- 2 ドキュメント表示枠で変数 a を選択し、[デバッグ] > [ウォッチに追加] を 選択します。[ウォッチ] タブに変数 a が追加されます。ブレークポイントに より、変数 a の値が初期化された後に停止したため、[値] カラムには、a の値 が現在 hello であることが示されます。
- 3 ドキュメント表示枠で変数 b を選択し、[デバッグ] > [ウォッチに追加] を 選択します。[ウォッチ] タブに変数 b が追加されます。変数 b が宣言される 前にテストが停止したため、[値] カラムには、「この変数は宣言されていません。'b'」と表示されます。

4 デバッグ・ビューア・ウィンドウで、「変数」タブを選択します。テストを使って作業している場合は、aがこの時点までに初期化された唯一の変数であるため、変数 a(値は hello)のみ表示されます。関数ライブラリで作業している場合は、SetVariables(値は Empty)とa(値は hello)の両方が表示されます。 変数 bが宣言される前にテストが停止したため、変数 b は表示されません。

手順7:次のブレークポイントで変数の値を検査する

[**実行**] ボタンをクリックして,テストの実行を続行します。テストは次のブレークポイントで停止します。[ウォッチ] タブと [変数] タブの変数 **a** と **b** の値が両方とも更新されています。

手順8:[コマンド]タブを使って変数の値を変更する

デバッグ・ビューア・ウィンドウで, [コマンド] タブを選択します。

コマンド・プロンプトで a="This is the new value of a" と入力し, キーボードの ENTER キーを押します。[実行] ボタンをクリックして, テストの実行を続行し ます。表示されたメッセージ・ボックスに, 「a」の新しい値が示されます。

第 33 章

テストの保守

QuickTest には、テスト対象のアプリケーションが変更されたときにテストを保 守できるツールが用意されています。たとえば、アプリケーションのオブジェ クトのプロパティや記述が変更されたり、オブジェクトが存在しなくなったり する場合があります。アプリケーションの変更に基づいて、テストのチェック ポイントの期待値を更新する必要が生じる場合もあります。本章では、 QuickTest のツールを使ってテストを更新および保守する方法を説明します。

本章では、次の項目について説明します。

- ▶ テストが失敗する理由(1057ページ)
- ▶ メンテスナンス実行ウィザードを使ったテストの実行(1060ページ)
- ▶ 更新モード・オプションを使ったテストの更新(1076ページ)

テストが失敗する理由

テスト内で予期しない状況が発生すると、テストは失敗します。多くの場合、 これはテスト中のアプリケーションが正常に機能していないことが原因です。 QuickTest は、アプリケーションの修正方法を理解するのに役立つテスト結果を 提供します。

場合によってはテストの作成後にテスト中のアプリケーションが変更されたために、テストが失敗する場合もあります。その場合は、QuickTestのテストを更新して、それらの変更を反映する必要があります。QuickTestには、これらの問題のいくつかを特定して解決するのに役立つツールが用意されています。

オブジェクトの変更

QuickTest は、テストのステップを実行するときに、そのステップで参照されて いるオブジェクトをそのテストに関連付けられたオブジェクト・リポジトリ内 で検索します。QuickTest は、リポジトリ内のオブジェクトの記述を使ってアプ リケーション内のそのオブジェクトを認識しようとします。

QuickTest は、いくつかの理由でアプリケーション内のオブジェクトを認識できない場合があります。

オブジェクトがアプリケーション内に存在しない

オブジェクト・リポジトリ内のオブジェクトの記述と一致するオブジェクトが アプリケーション内に見つかりません。メンテスナンス・モードを使用して、 テストで使用するオブジェクトを認識できます。

親オブジェクトが変更されている

オブジェクト・リポジトリ内のオブジェクトと一致し,同じ階層を持つオブ ジェクトがアプリケーション内に見つかりません。メンテスナンス・モードを 使用して,テストで使用するオブジェクトを認識できます。

オブジェクト記述プロパティの値が変更されている

オブジェクト・リポジトリ内のオブジェクトとほぼ同じであり、同じ記述プロ パティ値を持つオブジェクトがアプリケーション内に見つかりません。メンテ スナンス・モードを使用して、テストで使用するオブジェクトを認識できます。

オブジェクトがオブジェクト・リポジトリ内に存在しない

QuickTest は、アプリケーション内のオブジェクトを認識しようとする前に、関連付けられたオブジェクト・リポジトリ内でテストが参照するオブジェクトを 検索します。関連付けられたどのオブジェクト・リポジトリにもオブジェクト が見つからない場合は、[実行エラー]ダイアログ・ボックスが表示され、オ ブジェクトがオブジェクト・リポジトリ内に存在しないことが通知されます。 見つからないオブジェクトをオブジェクト・リポジトリに手動で追加するか、 ステップを変更する必要があります。

オブジェクトの記述セットを変更する必要がある

QuickTest は、一連のプロパティを使ってアプリケーション内のオブジェクトを 認識します。オブジェクト・リポジトリ内のオブジェクトの一連の認識プロパ ティを使ってアプリケーション内のオブジェクトと一致する一意の記述が得ら れない場合、QuickTest はオブジェクトを見つけることができません。更新モー ドを使ってテスト内のオブジェクトの一連の認識プロパティを更新することに より、[オブジェクトリポジトリ]ダイアログ・ボックスで定義された一連の プロパティと照合できます。

チェックポイントの変更

テスト中のアプリケーションに予期しない状況が発生すると、チェックポイントは失敗します。多くの場合、これはアプリケーションが正常に機能していないことが原因です。QuickTestは、アプリケーションの修正方法の理解に役立つテスト結果を提供します。

テストの作成後にアプリケーションが変更されたために、チェックポイントが 失敗する場合もあります。その場合は、QuickTest のチェックポイントを更新し て、それらの変更を反映する必要があります。アプリケーションの変更を反映 するには、更新モードを使ってテスト内のチェックポイントを更新します。

たとえば、アプリケーション内に「Name」というラベルのエディット・ボッ クスがあり、その値を「Michael」にする必要があるとします。この場合、エ ディット・ボックスのラベルが「Name」かどうかをチェックするチェックポ イントと、エディット・ボックスの値が「Michael」かどうかをチェックする チェックポイントを作成できます。エディット・ボックスの値をチェックする チェックポイントが(値が「Suzy」だったために)失敗した場合は、アプリ ケーションが正常に機能していないため、QuickTestのテスト結果を使ってアプ リケーションの修正方法を決定します。エディット・ボックスのラベルが 「ID」に変更されている場合は、エディット・ボックスのラベルが「Name」か どうかをチェックするチェックポイントが失敗します。アプリケーションが変 更されているため、テストを更新して、それらの変更を反映する必要がありま す。アプリケーションの変更を反映するには、更新モードを使ってチェックポ イントを更新します。

メンテスナンス実行ウィザードを使ったテストの実行

テスト中のアプリケーションの変更を反映するには、メンテスナンス実行モー ドを使ってテストを更新します。

メンテスナンス実行モードでテストを実行すると、テストが実行された後、ス テップとオブジェクト・リポジトリを更新する手順が順を追って示されます。 この処理は、オブジェクト・リポジトリ内のオブジェクトのプロパティ値とテ スト内のオブジェクトのプロパティ値が一致しないために実行できないステッ プが見つかるたびに実行されます。

メンテスナンス・モードでテストを実行すると、アプリケーション内でオブ ジェクトが見つからなかったために失敗したステップに対してメンテスナンス 実行ウィザードが開きます。ウィザードを使って、テストで使用するアプリ ケーション内のオブジェクトを指定したり、テストの失敗したステップの前に コメントを追加したりできます。テスト中のアプリケーション内のオブジェク トを指定すると、そのオブジェクトと、関連付けられたオブジェクト・リポジ トリ内のオブジェクトが比較されます。

メンテスナンス・モードでは、指定したオブジェクトのプロパティ値と関連付け られたリポジトリ内のオブジェクトのプロパティ値との比較結果に応じて、テス トを更新してアプリケーションの変更を反映するための方法が1つ示されます。 ウィザードの各ポイントで[**リセット**] ボタンをクリックすると、失敗したス テップで使用する別のオブジェクトをアプリケーションから指定できます。

メンテスナンス実行モードが終了すると、テストに対して行われた変更の要約 がメンテスナンス実行ウィザードに表示されます。メインの[テスト結果] ウィンドウにも、テストに対して行われた変更の詳細を示すメンテナンスの要 約が表示されます。これには、更新されたオブジェクト、追加されたオブジェ クト、更新されたステップ、コメントが付けられたステップ、およびオブジェ クト・リポジトリに対する変更の要約が含まれます。

注:

- ▶ メンテスナンス・モードでテストを実行するには、Microsoft Script Debugger がインストールされていることが必要です。インストールされていない場合 は、QuickTest Additional Installation Requirements ユーティリティを使ってイ ンストールできます([スタート] > [プログラム] > [QuickTest Professional] > [Tools] > [Additional Installation Requirements] を 選択します)。
- ▶ メンテスナンス実行モードで実行できるのは、QuickTest が Normal (実行 マーカを表示する) 実行モードを使用するように設定されている場合だけで す。Fast モードで実行することはできません。詳細については、1181 ページ 「テストの実行オプションの設定」を参照してください。
- ➤ ユーザ・インタフェースのないアプリケーション(Web サービスなど)は、 メンテスナンス実行モードでは実行できません。

ヒント:

- ▶ 通常、メンテスナンス実行モードで実行するのは、アプリケーションが変更 されており、QuickTest がオブジェクトを認識できないことがわかっている 場合です。したがって、QuickTest がオブジェクトの表示を待ち始めてから オブジェクトが見つからないと判断するまでの時間を短くしたい場合があり ます。このような場合は、[テストの設定]ダイアログ・ボックスの[実行] タブでオブジェクト同期化タイムアウトを変更できます。詳細については、 1195ページ「テストのための実行設定の定義」を参照してください。
- > メンテスナンス実行モードの終了後、その設定を前の通常実行時の値にリ セットしたい場合があります。また、[オブジェクトリポジトリ]ウィンド ウまたはオブジェクト・リポジトリ・マネージャの[アプリケーションから 更新]オプションを使って、アプリケーション内のオブジェクトから個々の テスト・オブジェクト記述を更新することもできます。詳細については、 145ページ「アプリケーション内のオブジェクトからのテスト・オブジェク ト・プロパティの更新」を参照してください。

テストをメンテスナンス・モードで実行するには、次の手順を実行します。

▶ 実行 • 1 テストを開き, [オートメーション] > [メンテナンス実行モード] を選択す

るか,ツールバー上の [実行] ボタンの横にある下向き矢印をクリックして [メンテナンス実行モード] を選択します。[実行] ダイアログ・ボックスが開 きます。

実行		×
結果保管場所 入力パラメータ		
実行結果の書き込み先:		
・新規実行結果フォルダ(N)		
rogram Files¥HF¥QuickTest Professiona¥Tests¥debug¥Res	4	
○ 一時実行結果フォルダ (古い一時結果(こ上書き)(<u>T)</u> □:¥DOCUME~1¥ADMINI~1¥LOCALS~1¥Temp¥TempResult	5	
	ヘルプ	

注: QuickTest テストを Quality Center プロジェクトに保存した場合は, [実行] ダ イアログ・ボックスの [結果保管場所] タブに, プロジェクト関連の追加オプ ションが表示されます。詳細については, 1346 ページ「Quality Center プロジェク トに格納されているテストの QuickTest からの実行」を参照してください。

- 2 メンテスナンス実行モードのセッションに、結果の場所と(必要な場合は)入 カパラメータの値を指定します。詳細については、941ページ「[結果保管場 所] タブについて」および942ページ「[入力パラメータ] タブについて」を 参照してください。
- 3 [OK] をクリックします。[実行] ダイアログ・ボックスが閉じ、メンテスナン ス実行モードのセッションが開始されます。標準では、実行セッションが終了 すると、[テスト結果] ウィンドウが開きます。実行セッション結果の表示の詳 細については、第 30 章「実行セッション結果の表示」を参照してください。

注: [オプション] ダイアログ・ボックスの [実行] タブで [**実行セッション** 終**7時に結果を表示する**] チェック・ボックスをクリアすると,実行セッショ ンの最後で [テスト結果] ウィンドウが開きません。[オプション] ダイアロ グ・ボックスの詳細については,第40章 「グローバル・テスト・オプション の設定」を参照してください。

テスト内のオブジェクトがアプリケーション内で見つからない場合は、メンテ スナンス実行ウィザードが開き、問題を解決する手順が順を追って示されま す。問題を解決すると、実行が再開されます。

[オブジェクトが見つかりません] 画面

テスト内のオブジェクトがテスト中のアプリケーション内で見つからない場合 は,[オブジェクトが見つかりません]画面が開きます。[オブジェクトが見つ かりません]画面では,見つからなかった**オブジェクト**と QuickTest が実行し ようとした**ステップ**が特定されます。



注:

[**提案**] 表示枠は、実行セッションで以前に見つからなかったオブジェクトを アプリケーション内で見つけることができなかった場合にのみ表示されます。

次の場合には,見つからなかったオブジェクトに対するメンテスナンス実行 ウィザードのオブジェクト指示オプションとコメント追加オプションが無効に なります。

- ▶ テストが読み取り専用モードで開かれている。
- ▶ オブジェクトが関数ライブラリの関数内で使用されている。
- ▶ オブジェクトのメソッドが登録されたユーザ関数として定義されている。

[オブジェクトが見つかりません] 画面では,次のオプションを使って問題を 解決できます。

▶ [オブジェクトをポイント]: [ポイント] ボタンをクリックして、ステップで 使用する必要があるアプリケーション内のオブジェクトを指定します。アプリ ケーションが変更されたことがわかっており、このステップで使用する新しい オブジェクトを特定することで問題を解決できる場合に、このオプションを使 用します。

指定した場所が複数のオブジェクトに関連付けられている場合は,[オブジェ クトの選択]ダイアログ・ボックスが開きます。ツリーから正しいオブジェク トを選択し,[**OK**]をクリックします。

指定したオブジェクトに応じて、次のいずれかの画面が開きます。

- ▶ 1070ページ「[既存のオブジェクトでステップを更新する] 画面」
- ▶ 1073 ページ「[オブジェクトをリポジトリに追加] 画面」
- ▶ 1067 ページ「[オブジェクトプロパティ値を変更] 画面」
- ▶ [コメントを追加]:失敗したステップを修正するためのメモとしてテストにコメントを追加する場合に、このオプションを使用します。

- ▶ [提案]:メンテスナンス・モード実行で以前に見つからなかったオブジェクト をアプリケーション内で見つけることができなかった場合にのみ表示されます。 最初に見つからなかったオブジェクトを別のオブジェクトに置き換えた場合は、 同じオブジェクトに置き換えることを提案するメッセージが表示されます。
 - ▶ [標準で設定]:後続のステップで同じオブジェクトが見つからなかった場合に、見つからなかったオブジェクトを、オブジェクト・リポジトリに追加したオブジェクトで自動的に置換します。これらの後続のステップは開かれません。

[オブジェクトが見つかりません] 画面には,次のナビゲーション・ボタンが あります。

▶ [無視]:テストの現在のステップを無視し、テストの残りの部分に対してメン テスナンス・モードの実行を継続します。これは、問題が QuickTest のテスト ではなく、テスト中のアプリケーションにある場合に使用できます。

注: [無視] を選択するときは、アプリケーションがテストの次のステップを 実行できる状態であることを確認してください。

- ▶ [再試行]:現在のステップを再試行します。
- ▶ [停止]:メンテスナンス実行を停止し, [メンテナンスモードのサマリ] 画面を開きます。
- ▶ [**ヘルプ**]: このヘルプ・トピックを開きます。

[コメントを追加] 画面

[コメントを追加] 画面では、テストの現在のステップの前にコメントを追加 できます。これは、アプリケーション内のオブジェクトの認識によって問題が 解決されない場合や、テストを手動で修正する場合に使用できます。

メンテナンス実行ウィザード - コメントを追加	×
次のステップの前にコメントを追加しようとしています	
አጋን፤ ንኑ:	
WebList "fromPort"	
Browser("Welcome: Mercury Tours"), Page("Find a Flight Mercury"), WebList("fromPort"), Select "N	
	1
TODO:	ſ
· · · · · · · · · · · · · · · · · · ·	l
戻す(E) (存止(S) へルブ	

[コメントの追加] 画面では、テスト内に、後でステップを修正するためのメモ として、「**TODO**」と追加したテキストからなるコメントを作成します。

[オブジェクト プロパティ値を変更] 画面

[オブジェクトプロパティ値を変更] 画面は,指定したオブジェクトとクラス が同じで記述プロパティ値が異なるオブジェクトが,関連付けられたオブジェ クト・リポジトリ内に存在する場合に開きます。

[オブジェクトプロパティ値を変更] 画面では, 関連付けられたオブジェクト・ リポジトリ内のオブジェクトのプロパティ値を更新して, アプリケーション内 で指定したオブジェクトのプロパティ値と一致させることが提案されます。

メンラ	ナンス実行ウィザード - オブジェクト	プロパティ値を変更	E E E E E E E E E E E E E E E E E E E	×
	選択したオブジェ ありますが、記述 ウィザードはオブジェクトのプ	クトは、関連するオブジェク が一致していません。 ロパティを次のように更新で	ト リポジトリ内に	
	オブジェクト: WinEdit "Agent Name:" オブジェクトのプロパティ:			
	プロパティ	出力値:	新規値	
	attached text	username	Agent Name:	
	ウィザードは、'attached text' プロ 正規表現を使用して、記述を修 この問題の解決に次のいずれかん ・ はい。オブジェクトリポジトリをす ・ 'attached text' プロパティを更? ・ ローカルのオブジェクト リポジト ・ 元のオブジェクト プロパティを得	パティを上に示す新しい値に変更 正できます(X): りオプションを選択してください: 更新してステップを再度実行する(新して正規表現を使い、ステップを りに新規オブジェクトを追加し、ス R持してコメントを追加し、次のス	するか、 [usem Agent N]ame:? <u>U)</u> 王再度実行する(P) テップを更新して再度実行する(A) テップに進む(<u>K</u>)	
	戻す(E)		OK(<u>O)</u> 停止(<u>S</u>) ヘルプ	

注:新しいプロパティ値に正規表現が推奨されない場合は,[オブジェクトプ ロパティ値を変更]画面の表の下に,該当するメッセージと推奨される正規表 現は表示されません。[**' <プロパティ名> ' プロパティを更新して正規表現を** 使い、ステップを再度実行する] ラジオ・ボタンも表示されません。

セクション	説明
[オブジェクト]	関連付けられたオブジェクト・リポジトリ内にあ る,アプリケーション内で指定したオブジェクト と同じクラスのオブジェクト。
[オブジェクトのプロパティ]	オブジェクト・リポジトリ内のオブジェクトのプ ロパティ値に対して行われる変更を表示する表。
[プロパティ]	値が変更されるプロパティの名前。
[出力値]	オブジェクト・リポジトリ内のオブジェクトの元 のプロパティ値。
〔新規値〕	アプリケーション内で指定したオブジェクトに基 づく,オブジェクト・リポジトリ内のオブジェク トの新しいプロパティ値。

[オブジェクトプロパティ値を変更] 画面の中央には、次の情報が表示されます。

指定したオブジェクトによっては、[オブジェクトプロパティ値を変更] 画面 に、関連付けられたオブジェクト・リポジトリ内のオブジェクトのプロパティ 値を更新するために正規表現を使用できることを示すメッセージが表示される 場合があります。['**くプロパティ名>'プロパティを更新して正規表現を使い、** ステップを再度実行する] ラジオ・ボタンも表示されます。エディット・ボッ クスで推奨される正規表現を変更することもできます。正規表現の詳細につい ては、757ページ「正規表現の使用について」を参照してください。

注: 複数のプロパティで正規表現を使用できる場合は,最初のプロパティ値の 正規表現だけが示されます。

[オブジェクトプロパティ値を変更]画面には、次のオプションが表示されます。

➤ [はい。オブジェクトリポジトリを更新してステップを再度実行する]:アプリ ケーション内で指定したオブジェクトのプロパティ値と一致するようにオブ ジェクト・リポジトリ内のオブジェクトのプロパティ値を更新して、ステップ を再実行します。新しいプロパティ値は、[新規値]の下に表示されます。

- ▶ ['<プロパティ名>'プロパティを更新して正規表現を使い、ステップを再度実行する]:正規表現を使うようにプロパティ値を更新できる場合にのみ表示されます。オブジェクト・リポジトリ内のオブジェクトのプロパティ値をエディット・ボックスに示された正規表現で更新して、ステップを再実行します。
- ▶ [ローカルのオブジェクト リポジトリに新規オブジェクトを追加し、ステップ を更新して再度実行する]:指定したオブジェクトに応じて、次のいずれかの 画面が開きます。
 - ▶ [既存のオブジェクトでステップを更新する] 画面: この画面は、追加する オブジェクトが関連付けられたオブジェクト・リポジトリ内に存在する場合 に開きます。
 - ► [オブジェクトをリポジトリに追加] 画面: この画面は、追加するオブジェ クトが関連付けられたオブジェクト・リポジトリ内に存在しない場合に開き ます。
- 「元のオブジェクト プロパティを保持してコメントを追加し、次のステップに進む]:オブジェクト・リポジトリ内のオブジェクトの元のオブジェクト・プロパティを保持します。[コメントを追加] 画面が開き、ステップの前にコメントを追加できます。その後、次のステップに進みます。

画面の下部には、[**リセット**] ボタンが表示されます。このボタンをクリックす ると、[オブジェクトが見つかりません] 画面に戻り、アプリケーション内の別 のオブジェクトを指定したり、このステップに対する別の措置を選択したりで きます。

注:

- ▶ 指定したオブジェクトがオブジェクト・リポジトリ内のオブジェクトとは異なる親オブジェクトとプロパティ値を持っている場合は、[オブジェクトプロパティ値を変更] 画面が2回開きます。最初の画面では、オブジェクト・リポジトリ内のオブジェクトの親オブジェクトを更新して、指定したオブジェクトの親オブジェクトと一致させることができます。2つ目の画面では、オブジェクト・リポジトリ内のオブジェクトを更新して、指定したオブジェクトと一致させることができます。
- ▶ メンテスナンス・モードでは、ローカル・オブジェクト・リポジトリのみが 変更されます。新しいオブジェクトを共有オブジェクト・リポジトリに含め るには、オブジェクト・リポジトリ・マネージャを使用します。詳細につい ては、260ページ「結合操作の実行」を参照してください。

[既存のオブジェクトでステップを更新する] 画面

[既存のオブジェクトでステップを更新する] 画面は, [オブジェクトが見つか りません] 画面で指定したオブジェクトが関連付けられたオブジェクト・リポ ジトリ内に存在し, 次の条件を満たす場合に開きます。

▶ 指定したオブジェクトとクラスが同じで記述プロパティ値が異なるオブジェクトが、関連付けられたオブジェクト・リポジトリ内に存在する。

または

▶ [オブジェクトプロパティ値を変更] 画面で [ローカルのオブジェクト リポジ トリに新規オブジェクトを追加し、ステップを更新して再度実行する] を選択 した。 [既存のオブジェクトでステップを更新する] 画面では、テスト内のステップ を更新して、関連付けられたオブジェクト・リポジトリ内にすでに存在するオ ブジェクトを使用することが提案されます。

メンラ	テナンス実行ウィザード - オブジェクトをリボジ	トリに追加		x
こうちょうしょうとしています:				
	オブジェクト:	オブジェクトのプロパティ:		
	WinObject "ワークシート メニュー	プロパティ	値	
	N="	nativeclass	MsoCommandBar	
		regexpwridciass		
	このウィザードでは、オブジェクトをローカル オブジェクト リボジトリに追追加し、 ステップを更新して次の新しいオブジェクトを使います: 元のステップ: Browser("Welcome: Mercury Tours").Page("Welcome: Mercury Tours").WebEdit("userName").Se 新規ステップ: Window(" UserGuide_listxls").WinObject("ワークシート メニュー バー").Set " tutorial"			
	次のオプションの1つを選択してください:			
	◎ オブジェクトを追加し、ステップを更新して再度実行する(A)			
	○ 元のオブジェクトとステップを保持して)	欠のステップに進む(<u>K</u>)		
	戻す(<u>E</u>)		(O) 停止(S) ヘルプ	

[既存のオブジェクトでステップを更新する] 画面の中央には,次のセクションがあります。

セクション	説明
[オブジェクト]	関連付けられたオブジェクト・リポジトリ内にあ る,アプリケーション内で指定したオブジェクト と同じオブジェクト。
[オブジェクトのプロパティ]	テスト・アプリケーション内で指定したオブジェ クトのプロパティおよびプロパティ値。
[元のステップ]	見つからなかったオブジェクトを含む,失敗した 元のステップ。
[新規ステップ]	関連付けられたオブジェクト・リポジトリ内にす でに存在するオブジェクトを参照するようにス テップを更新した場合の,新しいステップ。

[既存のオブジェクトでステップを更新する] 画面には、次のオプションが表示されます。

▶ [ステップを更新して再実行する]:失敗したステップを [新規ステップ] に示された内容で更新し、ステップを再実行します。

注:メンテスナンス・モードでは、テストから元のステップが削除されません。元のステップはコメントに変換され、更新されたステップがその下に追加されます。

▶ [元のステップを保持して次のステップに進んでください]: 元のステップを保持し、テストの残りの部分に対してメンテスナンス・モードの実行を継続します。

画面の下部には、[**リセット**] ボタンが表示されます。このボタンをクリック すると、[オブジェクトが見つかりません] 画面に戻り、アプリケーション内 の別のオブジェクトを指定したり、このステップに対する別の措置を選択した りできます。

[オブジェクトをリポジトリに追加] 画面

[オブジェクトをリポジトリに追加] 画面は,指定したオブジェクトが関連付け られたどのオブジェクト・リポジトリ内にも存在せず,次の条件を満たす場合 に開きます。

▶ 指定したオブジェクトとクラスが同じで記述プロパティ値が異なるオブジェクトが、関連付けられたオブジェクト・リポジトリ内に存在する。

または

▶ [オブジェクトプロパティ値を変更] 画面で [ローカルのオブジェクト リポジ トリに新規オブジェクトを追加し、ステップを更新して再度実行する] を選択 した。

[オブジェクトをリポジトリに追加] 画面では,指定したオブジェクトをオブ ジェクト・リポジトリに追加することが提案されます。

メンテ	ナンス実行ウィザード - オブジェクトをリボジト	りに追加		×
こうちょうとしています:				
	オプジェクト:	オブジェクトのプロパティ:		
	WinEdit "代理店名:"	プロパティ	値	
		nativeclass	Edit	
l				
	このウィザードでは、オブジェクトを ステップを更新して次の新しいオン	ローカル オブジェクト リオ ブジェクトを使います:	ペジトリに追加し、	
	元のステップ: Window("Flight Reservation Dialog"). 新規ステップ: Dialog("ログイン").WinEdit (" 代理店名・"	Dialog("Open Order Dialog").Set "mercury"	").WinEdit("Flight Date Edit").Set	
	次のオプションの1つを選択してください:			
	●オブジェクトを追加し、ステップを更新して再度実行する(A)			
	○ 元のオブジェクトとステップを保持して次のステップに進む(匹)			
	戻す(E)	ОК	② 停止③ ヘルプ	I

[オブジェクトをリポジトリに追加] 画面の中央には、次のセクションがあり ます。

セクション	説明
[オブジェクト]	テスト・アプリケーション内で指定したオブジェ クト。
[オブジェクトのプロパティ]	テスト・アプリケーション内で指定したオブジェ クトのプロパティおよびプロパティ値。
[元のステップ]	見つからなかったオブジェクトを含む,失敗した 元のステップ。
[新規ステップ]	オブジェクト・リポジトリに追加されるオブジェ クトを参照するようにステップを更新した場合の, 新しいステップ。

[オブジェクトをリポジトリに追加] 画面には、次のオプションが表示されます。

- ▶ [オブジェクトを追加し、ステップを更新して再度実行する]:オブジェクト・ リポジトリに新しいオブジェクトを追加し、失敗したステップを [新規ステッ プ] に示された内容で更新して、ステップを再実行します。
- ▶ [元のオブジェクトとステップを保持して次のステップに進む]:元のオブジェ クトを含む元のステップを保持し、テストの残りの部分に対してメンテスナン ス・モードの実行を継続します。

画面の下部には、[**リセット**] ボタンが表示されます。このボタンをクリック すると、[オブジェクトが見つかりません] 画面に戻り、アプリケーション内 の別のオブジェクトを指定したり、このステップに対する別の措置を選択した りできます。

注:

- ▶ メンテスナンス・モードでは、ローカル・オブジェクト・リポジトリのみが 変更されます。新しいオブジェクトを共有オブジェクト・リポジトリに含め るには、オブジェクト・リポジトリ・マネージャを使用します。詳細につい ては、260ページ「結合操作の実行」を参照してください。
- ▶ メンテスナンス・モードでは、テストから元のステップが削除されません。 元のステップはコメントに変換され、更新されたステップがその下に追加されます。

[メンテナンス モードのサマリ] 画面について

メンテスナンス実行モードを終了すると, [メンテナンスモードのサマリ] 画 面が開きます。

メンテナンス実行ウィザード - メンテナンス モードのサマリ	X
メンテナンス実行は正常に完了しました。	
0 オブジェクトはローカルの オブジェクト リボジトリに追加され ました。	
0 オブジェクトの ブロ パティが更新され ました。	
0 ステップが変更され ました。	
1 [コ メント] がテ ス トに追加されました。	
	1

[メンテナンスモードのサマリ] 画面には, ローカル・オブジェクト・リポジ トリに追加されたオブジェクトの数, 更新されたオブジェクト・プロパティの 数, 変更されたステップの数, およびテストに追加されたコメントの数が表示 されます。

[完了]をクリックすると、メンテスナンス実行が終了します。標準設定では、 実行セッションが終了すると、[テスト結果]ウィンドウが開き、実行中に更 新されたステップとオブジェクトに関する詳細が表示されます。実行セッショ ン結果の表示の詳細については、第30章「実行セッション結果の表示」を参 照してください。

注: [オプション] ダイアログ・ボックスの [実行] タブで [実行セッション 終了時に結果を表示する] チェック・ボックスをクリアすると、実行セッショ ンの最後で [テスト結果] ウィンドウが開きません。[オプション] ダイアロ グ・ボックスの詳細については、第40章「グローバル・テスト・オプション の設定」を参照してください。

更新モード・オプションを使ったテストの更新

更新モードでテストを実行すると、QuickTest によってテストが実行され、テスト・オブジェクト記述、ActiveScreenの画像と値、チェックポイントの期待値、あるいはその両方が更新されます。テストを保存すると、更新されたデータが以降の実行に使用されます。

QuickTest によるテストの更新時には,選択した実行オプションに従って,テストの反復が1回だけと,テスト内の各アクションの反復が1回実行されます。 アクションの詳細については,第14章「アクションを使った作業」を参照してください。
注:

- ▶ 更新モードでテストを実行した場合、データ・テーブルのデータや環境変数 など、パラメータ化された値は更新されません。パラメータ化された値および環境変数の詳細については、第22章「値のパラメータ化」を参照してく ださい。更新モードでは、オブジェクト・リポジトリ内の既存のオブジェク ト記述のプロパティ値は変更されません。オブジェクトのプロパティ値をア プリケーションと一致するように修正するには、メンテスナンス実行モード を使用します。詳細については、1060ページ「メンテスナンス実行ウィ ザードを使ったテストの実行」を参照してください。
- ➤ QuickTest によるテストの更新時には、更新対象のオブジェクトが元々は共 有オブジェクト・リポジトリに含まれていた場合でも、更新されたオブジェ クトはローカル・オブジェクト・リポジトリに保存されます。ローカル・オ ブジェクト・リポジトリはどの共有オブジェクト・リポジトリよりも高い優 先順位を持っているため、次回テストを実行したときは、ローカル・オブ ジェクト・リポジトリのオブジェクトが使用されます。

ヒント:更新モードを使ってテストを更新した後は、オブジェクト・リポジト リ・マネージャの[**ローカル リポジトリから更新**]オプションを使ってローカ ル・オブジェクト・リポジトリのオブジェクトを共有オブジェクト・リポジト リに結合できます。詳細については、第6章「オブジェクト・リポジトリの管 理」を参照してください。 ▶ 実行 ▼

1 テストを開き, [**オートメーション**] > [**更新モード**] を選択するか, ツール バー上の [**実行**] ボタンの横にある下向き矢印をクリックして [**更新モード**] を選択します。

[更新] ダイアログ・ボックスが表示されます。

更新
更新オプション 入力パラメータ
▼ テストオブジェクトの記述を更新する(型)
✓ チェックポイントおよび出力値のプロパティを更新する(C)
✓ ActiveScreenの画像と値を更新する(S)

2 更新プロセスの設定を行います。詳細については、1080ページ「[更新オプ ション] タブについて」および 942ページ「[入力パラメータ] タブについて」 を参照してください。

注: 更新セッションの実行結果は、常に一時保存場所に保存されます。

3 [OK] をクリックします。[更新] ダイアログ・ボックスが閉じ, QuickTest に よって更新モードでの実行が開始されます。テストの更新中は,ステータス・ バーで「**更新**」というテキストが点滅します。 テストが実行され、テスト・オブジェクト記述、ActiveScreen 情報、チェック ポイントの期待値(それぞれユーザの選択に応じて)が更新されます。実行 セッションが終了すると、[テスト結果] ウィンドウが開きます。

結果を表示する方法の詳細については,第30章「実行セッション結果の表示」 を参照してください。

注: [オプション] ダイアログ・ボックスの [実行] タブで [実行セッション 終了時に結果を表示する] チェックボックスをクリアした場合は,更新セッ ションの最後に [テスト結果] ウィンドウは開きません。[オプション] ダイ アログ・ボックスの詳細については,第40章 「グローバル・テスト・オプ ションの設定」を参照してください。

更新が終了すると、[テスト結果]ウィンドウには次の情報が表示されます。

▶ チェックポイントの更新された値。

▶ 更新されたテスト・オブジェクト記述。

例を次に示します。

ステップ名: Welcome: Mercury Tours-記述の更新

ステップ 完了

オブジェクト	詳細	結果	時間
Welcome: Mercury Tours-記述の更 新	テストオブジェクトの以前の記 述: Mercury Class = Browser CreationTime = 0 テストオブジェクトの新しい記述: Mercury Class = Browser	完了	2008/03/11 - 11:01:16

[更新オプション] タブについて

[更新オプション] タブでは、テストのどの側面(テスト・オブジェクト記述、 チェックポイントの期待値、ActiveScreenの画像と値、あるいはそのすべて) を更新するかを指定できます。テストを保存すると、更新されたテストの結果 が以降の実行に使用されます。

更新	×
更新オプション 入力パラメータ	
▶ テフトオブジェクトの記述を再新する(0)	
► TAPATOLOGIZE#(93(0))	
☑ チェックポイントおよび出力値のプロパティを更新する(Ը)	
✓ ActiveScreenの画像と値を更新する(S)	

更新する情報のタイプを次から指定できます。

▶ [テストオブジェクトの記述を更新する]: 関連付けられたオブジェクト・リポジトリ内の各オブジェクト・クラスのプロパティ・セットが、[オブジェクトの認識]ダイアログ・ボックスで現在定義されているプロパティに従って更新されます。このオプションを使用して、特定のタイプのオブジェクトの認識に使用するプロパティのセットを変更できます。

注:あるオブジェクト・クラスに対して [オブジェクトの認識] ダイアログ・ ボックスで選択したプロパティ・セットが特定のオブジェクトには適していな い場合,新しいオブジェクト記述が以降の実行で失敗する可能性があります。 そのため,テストを更新する前にオブジェクト・リポジトリのコピーを保存し (可能な場合は,バージョン・コントロールをサポートした Quality Center プロ ジェクトにチェック・インする),必要に応じて以前のバージョンに戻せるよ うにしておくことをお勧めします。 このオプションは、アプリケーション内の認識しやすいプロパティ値(オブ ジェクトのラベルなど)を使用してテストの記録とデバッグを行う場合に特に 便利ですが、言語やオペレーティング・システムに依存する場合があります。 テストをデバッグした後で、[**更新モード**] オプションを使用し、オブジェク ト記述を変更してより汎用的なプロパティ値を使用できます。

たとえば、アプリケーションの英語版を対象にテストを設計したとします。テ スト・オブジェクトは英語版のテスト・オブジェクト・プロパティの値に従っ て認識されますが、その一部は言語に依存する場合があります。次に、同じア プリケーションのフランス語版でも同じテストを使用するとします。

これを行うには、言語に依存しないプロパティを定義します。それらのプロパ ティはオブジェクトの認識に使用されます。たとえば、リンク・オブジェクト の認識に text プロパティ値ではなく target プロパティ値を使用することもでき ます。プロパティを定義したら、アプリケーションの英語版に対して、定義し た新しいプロパティを使用して更新を行います。テスト・オブジェクト記述が 変更され、後でフランス語版に対してもテストを正しく実行できるようになり ます。

ヒント:正常に実行されるテストがあるが、特定のオブジェクトがスマート認識を使って認識される場合は、オブジェクトの認識に使用されるプロパティ・ セットを変更した後で、[テストオブジェクトの記述を更新する] オプション を使って、スマート認識がオブジェクトの認識に使用しているプロパティ・ セットを使用するようにテスト・オブジェクト記述を更新できます。

[テストオブジェクトの記述を更新する]を選択してテストを実行すると、各 ステップに指定されているテスト・オブジェクトが、現在のテスト・オブジェ クト記述に基づいて検出されます。QuickTest で記述に基づいてテスト・オブ ジェクトを検出できない場合は、(スマート認識が有効な場合)テスト・オブ ジェクトの認識にはスマート認識プロパティが使用されます。QuickTest によっ てテスト・オブジェクトが検出された後、[オブジェクトの認識]ダイアログ・ ボックスで定義した必須プロパティと補足プロパティに基づいて、記述が更新 されます。 **注**: 更新プロセス中に認識されないテスト・オブジェクトは更新されません。 その他の実行セッションの場合と同じように,更新モードでも,実行中にオブ ジェクトが見つからない場合は実行セッションが失敗し,失敗に関する情報が テスト結果に追加されます。このような状況では,メンテスナンス・モードを 使ってこれらの問題を解決することをお勧めします。

以前のテスト・オブジェクト記述に使用され, [オブジェクトの認識] ダイア ログ・ボックスで定義されるテスト・オブジェクト・クラスの記述の一部では なくなっているすべてのプロパティは, 値が正規表現としてパラメータ化また は定義されている場合であっても, 新しい記述から削除されます。

同じプロパティがテスト・オブジェクトの新しい記述と以前の記述の両方に出 現する場合は、次のいずれかの処理が行われます。

- ▶ 以前の記述のプロパティ値が正規表現としてパラメータ化または指定され、 新しいプロパティ値と一致する場合、そのプロパティの以前のパラメータ化 された値または正規表現の値は保持されます。たとえば、以前のプロパティ 値が正規表現 button.* として定義され、新しい値が button1 である場合、プ ロパティ値は button.* のままになります。
- ➤ 以前の記述のプロパティ値が新しいプロパティ値と一致しないものの、オブジェクトがスマート認識を使用して検出された場合、そのプロパティ値は新しい定数のプロパティ値に更新されます。たとえば、以前のプロパティ値がbutton.*であり、新しい値が My button である場合、QuickTest でオブジェクトを検出するスマート認識定義が有効になっていれば、My button が新しいプロパティ値になります。この場合、パラメータ化されているものや正規表現が使用されているものは、すべてテスト・オブジェクト記述から削除されます。
- ▶ [チェックポイントおよび出力値のプロパティを更新する]: テストの作成後に アプリケーションで発生した変更を反映するためにチェックポイントの期待値 が更新され、出力値ステップで取得できる項目のリストが更新されます。

たとえば、テストの一部としてテキスト・チェックポイントを定義していて、 テストの作成後にアプリケーション内のテキストが変更されたとします。テス トを更新して、新しいテキストを反映するため、チェックポイントのプロパ ティを更新できます。 出力値オプションは、主に Web サービス・テストで使用される XML 出力値ス テップに適用されます。詳細については、『HP QuickTest Professional アドイン・ ガイド』の Web サービスについて説明している項を参照してください。

注:

- ▶ チェックポイントのプロパティの値がパラメータ化されているか正規表現を 含んでいる場合は、このオプションを使用しても更新されません。
- テストに WinRunner テストへの呼び出しが含まれ、テストと期待結果フォ ルダの両方の書き込み権限を持っている場合は、[チェックポイントおよび 出力値のプロパティを更新する]を選択すると、WinRunner テストのチェッ クポイントの期待値も更新されます。WinRunner テストを更新しない場合 は、WinRunner テストを呼び出す行をコメント・アウトできます。
 WinRunner テストの呼び出しの詳細については、1380ページ「WinRunner テ ストの呼び出し」を参照してください。コメント行の詳細については、806 ページ「コメントの追加」を参照してください。
- ➤ ビットマップ・チェックポイントの作成時に [選択した領域のみを保存する] チェック・ボックスを選択した場合, [更新モード] オプションによって更新されるのは, ビットマップの保存された領域のみです。オリジナルのフル・サイズのオブジェクトは更新されません。チェックポイントのオブジェクトを増やすには, 新しいチェックポイントを作成します。詳細については, 519ページ「ビットマップの検査」を参照してください。
- ► [ActiveScreen の画像と値を更新する]: テストを記録した時点以降にアプリケーションに生じた可能性のある変更をすべて反映する場合や, ActiveScreen が想定どおりに表示されない場合に, ActiveScreen の画像とプロパティ値が更新されます。たとえば、テストを記録してからアプリケーションのダイアログ・ボックスが変更されたとします。その場合、テストを更新して、ActiveScreen のダイアログ・ボックスの外観とそのプロパティを更新できます。

第33章・テストの保守

第 VIII 部

QuickTest IDE を使用した作業



QuickTest ウィンドウのレイアウト

本章では、QuickTest ウィンドウをカスタマイズする方法、および QuickTest ド キュメントを使用する方法について説明します。

本章では、次の項目について説明します。

- ▶ QuickTest ウィンドウのレイアウト変更(1087 ページ)
- ▶ 複数のドキュメントを使った作業(1097ページ)

QuickTest ウィンドウのレイアウト変更

QuickTest ウィンドウのレイアウトは変更できます。たとえば、表示枠の移動と サイズ変更、表示枠の表示と自動非表示の切り替え、タブ表示枠の作成、およ び表示するツールバーの選択が行えます。必要に応じて、標準のレイアウトを 復元することもできます。

また、QuickTest の各種セッション(表示/編集,記録,および実行セッション)の必要に合わせてQuickTest ウィンドウのサイズを変更することも可能です。たとえば、テストを作成または編集するときは、QuickTest を画面全体に表示して、実行セッション中にはQuickTest ウィンドウを最小化できます。詳細については、1168ページ「QuickTest ウィンドウ・レイアウトのカスタマイズ」を参照してください。

QuickTest ウィンドウのレイアウトをカスタマイズまたは復元すると, QuickTest により, すべてのドキュメント・タイプおよびセッション・タイプに 変更が適用されます。

表示枠の移動

QuickTest ウィンドウの表示枠は、ユーザ自身の使いやすい位置に移動できま す。また、表示枠を並べ替えたり、表示枠をタブ形式の表示枠に(またはその 逆に)変更したりもできます。

表示枠をドラッグしている間は、QuickTest ウィンドウ上にマーカが表示されま す。いずれかのマーカの上にカーソルを置くと、そのマーカによって示される 領域がグレー表示され、選択した位置に表示枠を移動した場合のウィンドウの レイアウトを確認できます。

ヒント:ドッキング可能な表示枠がはめ込まれないようにして移動するには, CTRL キーを押したままで目的の場所へドラッグします。

表示枠を移動するには、次の手順を実行します。

 QuickTest ウィンドウで、移動する表示枠のタイトル・バーまたはタブをドラッ グします。目的の表示枠が QuickTest ウィンドウに表示されていない場合は、 [表示]メニューから選択します。 たとえば、左下にあるタブ形式の[データテーブル]表示枠を、新しい表示枠 としてウィンドウの右上に移動できます。表示枠をドラッグすると、アクティ ブな表示枠および QuickTest ウィンドウの各辺にマーカが表示されます。



現在のペイン マーカー ウィンドウ ペイン マーカー

ヒント:

- ▶ タブ形式の表示枠を1つだけ移動するには、タブのラベルをドラッグします。タブ形式の表示枠をドラッグし始めると、そのタブが消え、タイトル・バーが表示されます。
- ▶ タブ形式の表示枠をすべて移動するには、アクティブなタブ形式の表示枠の タイトル・バーをドラッグします。

次のマーカが表示されます。

タイプ	マーカ	詳細
現在表示枠マーカ		 次のことが可能です。 > マウス・ボタンを放したときに選択されていた矢印マーカに応じて,移動した表示枠を,アクティブな表示枠内の上下左右いずれかの側,または中央に,新しい表示枠として配置します。 > 中央のマーカが選択されているときにマウス・ボタンを放して,移動した表示枠を,アクティブなウィンドウ内に新しいタブ形式の表示枠として配置します。 注:中央のマーカは,移動する表示枠を,(ドキュメント表示枠を除く)既存の表示枠内にドラッグしている場合にのみ表示されます。
ウィンドウ 表示枠マーカ		QuickTest ウィンドウの上部に表示枠を配置します。
		QuickTest ウィンドウの右側に表示枠を配置します。
		QuickTest ウィンドウの下部に表示枠を配置します。
	e	QuickTest ウィンドウの左側に表示枠を配置します。



2 次の図のように、[データテーブル]をドラッグし、アクティブな表示枠の右 矢印マーカの上にカーソルを置きます。表示枠の新しい位置を示すグレーの領 域が表示されます。

🗾 Quick Test Professional - [E:¥Tests ¥Reser	veFlight*]					_ 🗆 ×
😽 ファイル(E) 編集(E) 表示(V) 挿入(D)	オートメーション(<u>A</u>)	リソース(B)	デバッグ(<u>D</u>)	ツール①	ウィンドウѠ	
ヘルプ(出)						- 8×
🕴 🔯 新規作成 👻 🕞 🔣 👻 🗒 🎒 🛔) 🔒 🔍 🗄		🖬 🚯 🗄	I 🖉 🔥 🗌	
🔋 🥥 記録 🕨 実行 🔳 停止 🛛 🌄 🖿 訴	😭 🛣 🕸 🎽	P • 🐲 •	🖗 🏟 🗳	È II %≣	(= 📲 🆣)	89
ReserveFlight*						↓ ▷
項目	操作	>	注釈			
🗸 🥩 Action1						
✓ Ø Welcome: Mercury Tours						
Velcome: Mercury Tours						
userName	Set 🚺	rcury"	"userName"	edit box (2 ")	mercury''を入り	J
password	SetSec	hcfd21a	暗号化された	と文字列 "43	6a3bcfd21a64	6e
Sign-In	データラー					×
Find a Flight: Mercury	A3 🔁	eattle				
je tripi ype	A	R	0	D	F	E A
	1 New York					
	2 Portland					
	3 Seattle					-~
H + P H キーワードビュー / エキスパー	4					
/###2	5					
	6					ĥ
	7					H
	8					[
		Action1 /				
				Ale	06	
		Ũ	″ <u>-</u>			
		MERC	U R Y		SI	GN-OFF
		T O U	RS			<u> </u>
					準備完	7

3 マウス・ボタンを放します。[データテーブル] が固定され, グレー表示され ていた領域に新しい表示枠として表示されます。

📖 QuickTest Professional - [E:¥test¥Re	serveFlight*					-	
	ሊወ オートメー	ション(<u>A</u>)	リソース(B) 🗄	デバッグ(<u>D</u>)	ツール①	ウィンドウѠ)
~ ヘルプ(H)							a ×
· • • 新期作成 • • • 閉(• • • • • • • • • • • • • • •	lv es en		a 🔿 İ 🖙			ചെച്	
				<u>s i i i i i i i i i i i i i i i i i i i</u>			
: ● 記録 ▶ 実行 ■ 停止 🍡 3	9 UN I 🖉 3	<u>r</u> 12. 👌	* * 🖗 * 🖗		: II 9≣ (j= (= 🖑	<u>(</u>)
🔬 ReserveFlight*	4 ▷	データラ	ーブル				• ¤ ×
項目	操作 ▲	A A	\3 Se	attle			
🔫 🥔 Action1			Α	В	С	D	
🗸 🗸 🗩 🖉 Welcome: Mercury Tours		1	New York				
🚽 🎝 Welcome: Mercury Tours		2	Portland				
userName	Set	3	Seattle				
password	SetSecure	4	Ī				
🔚 🔚 Sign-In	Click	5					
🚽 🎝 Find a Flight: Mercury		6					
🚽 🖉 tripType	Select	7					
- 🧔 tripType	Select	8					
	Select	9					
FromPort	Select 🔽						
		12					-
	<u>iereae</u> /		Global 🔨 A	ction1 /			
情幸履	+ † X	ActiveScre	een				- 4 ×
「「「「「「」」「「」」「「」」「」」「「」」「」」「」」「」」「」」「」」「	3						
			- ~		10 A	A.	
		6	201			aplan.	0
				AAD	16		, `
		0	· ·				
		MER				SIGN-OF	F V
	Þ	4					
						準備完了	

ヒント:表示枠は,QuickTest ウィンドウ上または画面上の任意の場所にフロー ティング表示枠として表示することもできます。フローティング表示枠の詳細 については,1093ページ「表示枠の表示と非表示」を参照してください。

4 移動する各表示枠についてこの手順を繰り返します。

表示枠の表示と非表示

表示枠を標準の位置に移動した後は、表示枠を常に表示するか、それとも自動 非表示にして必要なときだけ表示するかを選択できます。

表示枠はドッキングかフローティングのいずれかの状態になります。

▶ ドッキング表示枠:ドッキング表示枠は、アプリケーション内の特定の位置に 固定されます。たとえば、マーカが示す位置に表示枠を移動すると、表示枠は その位置にドッキングされます。

ドッキング表示枠は、QuickTest ウィンドウに常に表示しておくか、それとも自動非表示にするかを選択できます。自動非表示にすると、QuickTest ウィンドウの端にサイドタブが表示され、目的のタブの上にカーソルを置いたときだけ該当する表示枠が表示されます。別の表示枠またはサイドタブを選択すると、自動非表示の表示枠は閉じ、サイドタブとして表示されます。

注: 情報表示枠は, 自動非表示に設定されている場合でも, テスト・スクリプ トで構文エラーが検出されると自動的に表示されます。

自動非表示の表示枠を開くと、標準設定では、QuickTest ウィンドウ上での位置 に応じた1つの辺全体がその表示枠の表示に使用されます。たとえば、 QuickTest ウィンドウの右辺に配置したドッキング表示枠は、QuickTest ウィン ドウの右端にサイドタブとして表示され、選択するとQuickTest ウィンドウの 右辺いっぱいに開きます。

ヒント:タブ形式の表示枠をすべて自動非表示にするには、アクティブなタブ 形式の表示枠のタイトル・バーを選択し、右クリックして[**自動非表示**]を選 択します。タブ形式の表示枠はサイドタブのグループとして QuickTest ウィン ドウの端に表示され、サイドタブの上にカーソルを置いたときだけ該当する表 示枠が表示されます。 ▶ フローティング表示枠:フローティング表示枠は、ほかのすべてのウィンドウの手前に表示されます。これらの表示枠は、画面上の任意の位置にドラッグすることができ、QuickTest ウィンドウの外にもドラッグできます。各フローティング表示枠には専用のタイトル・バーが表示されます。

注:フローティング表示枠を自動非表示にしたり,個別のタブ形式の表示枠を 自動非表示したりはできません。

表示枠の表示と非表示を切り替えるには、次の手順を実行します。

QuickTest ウィンドウで,自動非表示にする表示枠を選択し,QuickTest ウィンドウのいずれかの端にサイドタブとして表示します。タイトル・バーには次のボタンが表示される場合があります。

ボタン	詳細
	 [メニュー] ボタンでは次を選択できます。 > [フローティング]:表示枠は、専用のタイトル・バーが付けられ、ほかのすべてのウィンドウおよび表示枠の手前に表示されます。 > [ドッキング]:表示枠は、QuickTest ウィンドウにドッキングされます。 > [自動非表示]:表示枠が、QuickTest ウィンドウ内での位置に応じて、QuickTest ウィンドウの上下左右いずれかの端にサイドタブとして表示されます。 > [非表示]:表示枠を閉じます。
D	[自動非表示] ボタン:表示枠を非表示にします。 表示枠は,QuickTestウィンドウ上での位置に応じて, QuickTestウィンドウの上下左右いずれかの端にサイドタブ として表示されます。 表示枠を表示するには,サイドタブの上にカーソルを置きま す。このボタンは,下記の[ドッキング] ボタンに切り替わ ります。
-9	[自動非表示] ボタン(横向き)は表示枠を QuickTest ウィ ンドウにドッキングし, 非表示になる前に配置されていた場所に戻します。また,ボ タンは,前述の [自動非表示] ボタンに切り替わります。
×	[閉じる] ボタンは、表示枠を閉じます。 表示枠は QuickTest ウィンドウから取り除かれます。表示枠 を再度開くには、[表示] メニューからその表示枠を選択し ます。 ヒント:表示枠を右クリックし、ショートカット・メニュー から [非表示] を選択して閉じることもできます。

ヒント:

- ▶ タブ形式の表示枠をすべて自動非表示にするには、アクティブなタブ形式の 表示枠のタイトル・バーを選択し、右クリックして[自動非表示]を選択し ます。タブ形式の表示枠はサイドタブのグループとして QuickTest ウィンド ウの端に表示され、サイドタブの上にカーソルを置いたときだけ該当する表 示枠が表示されます。
- ▶ 表示枠のタイトル・バーを右クリックし、ショートカット・メニューから [フローティング]を選択すると、表示枠はフローティング形式になり、専 用のタイトル・バーが付けられて、ほかのすべてのウィンドウおよび表示枠 の手前に表示されます。表示枠のタイトル・バーをダブルクリックするか、 タイトル・バーを右クリックして [ドッキング]を選択すると、その表示枠 はドッキングし、QuickTest ウィンドウ内の元の位置に戻ります。

ツールバーの表示と非表示

ツールバーの表示と非表示を切り替えるには、[表示]>[ツールバー]メ ニュー・オプションを使用します。

ツールバーの左側にあるツールバー・ハンドルの上にカーソルを置いて任意の 位置にドラッグすると、そのツールバーはフローティング形式になり、タイト ル・バー付きで表示されます。

メニューのタイトル・バーをダブルクリックすると、メニューがドッキング し、QuickTest ウィンドウ内の元の位置に戻ります。また、[**閉じる**] ボタンを クリックすると、メニューが閉じます。

×

QuickTest ウィンドウの標準のレイアウトの復元

すべてのドキュメント・タイプについて、いつでも QuickTest ウィンドウの標 準のレイアウトを復元できます。

標準のレイアウトを復元するには、次の手順を実行します。

- 1 [ツール] > [オプション] を選択します。[オプション] ダイアログ・ボック スが表示されます。
- 2 [一般] タブで, [**レイアウトの復元**] ボタンをクリックします。すべてのド キュメント・タイプの表示枠とツールバーが,標準のサイズと位置に復元され ます。

[オプション] ダイアログ・ボックスの詳細については,第40章「グローバル・テスト・オプションの設定」を参照してください。

複数のドキュメントを使った作業

QuickTest では、1 つのテストを開いて作業できるのに加え、複数の関数ライブ ラリを同時に開いて操作できます。現在開いているテストに関連付けられてい るかどうかに関係なく、任意の関数ライブラリを開くことができます。

[**ウィンドウ**] メニュー・オプションでは、開いているドキュメント・ウィン ドウの選択とアクティブ化 (フォーカスの移動),開いているドキュメント・ ウィンドウを QuickTest ウィンドウ内に配置する方法の選択、開いているすべ ての関数ライブラリ・ウィンドウの終了を行うことができます。

また, [ウィンドウ] ダイアログ・ボックスを使用して, 開いている QuickTest ドキュメント・ウィンドウを管理することもできます。 [ウィンドウ] ダイアログ・ボックスを使用して複数のドキュメントを操作するには、次の手順を実行します。

1 [**ウィンドウ**] > [**ウィンドウ**] を選択します。[ウィンドウ] ダイアログ・ ボックスが開きます。



[ウィンドウ] ダイアログ・ボックスには、開いているドキュメント・ウィン ドウのリストが表示されます。リストには、開いているテスト/コンポーネン ト/アプリケーション領域や、現在開いているすべての関数ライブラリ・ウィ ンドウなどが含まれます。 2 [ウィンドウ] ダイアログ・ボックスには次のボタンがあり,開いているド キュメントの管理に使用できます。

ボタン	詳細
[切り替え]	QuickTest ウィンドウの中で, 選択したドキュメントに フォーカスを移動する。
[οκ]	[ウィンドウ] ダイアログ・ボックスを閉じる。
[上書き保存]	選択したドキュメントを保存する。
[ウィンドウを閉じる]	選択した関数ライブラリを閉じる。
[重ねて表示]	選択したドキュメントを、部分的に重ねて表示する。
[上下に並べて表示]	選択したドキュメントを,重ならないように横方向に並べ て表示する。
[左右に並べて表示]	選択したドキュメントを,重ならないように縦方向に並べ て表示する。
[最小化]	選択したドキュメントを最小化する。
[ヘルプ]	このダイアログ・ボックスの QuickTest Professional ヘルプ・ トピックを表示する。

3 [OK] をクリックし, [ウィンドウ] ダイアログ・ボックスを閉じます。

第34章・QuickTest ウィンドウのレイアウト

第 35 章

リソースの管理

QuickTest では、テストに関連付けられているリソースを、1 つの表示枠で管理 できます。[リソース]表示枠では、テストにおける関数ライブラリ、回復シ ナリオ、およびオブジェクト・リポジトリの関連付け、削除、開く操作、優先 順位の変更、その他の管理が行えます。

本章では、次の項目について説明します。

▶ [リソース]表示枠について(1101ページ)

[リソース] 表示枠について

テストおよびアクションは、関数ライブラリ、回復シナリオ、およびオブジェ クト・リポジトリなどのリソースに関連付けられています。QuickTest では、テ ストに関連付けられているリソースの多くが、[リソース]表示枠に表示され ます。[リソース]表示枠では、テストのすべてのリソースを、追加、移動、 および管理できます。



[リソース]表示枠を表示するには, [リソース ウィンドウ] ボタンをクリック するか,または [表示] > [リソース] を選択します。



[リソース]表示枠に,現在のテストのリソースが表示されます。関数ライブ ラリと回復シナリオはリソースの種類に応じてグループ化されます。オブジェ クト・リポジトリは,アクションに応じてグループ化されます。

[リソース]表示枠はツリー構造で表示されます。ツリー内のノードを右ク リックすると、そのリソースのショートカット・メニューが開きます。オプ ションによって、リソースのルート・ノードのショートカット・メニューから アクセスできるものと、特定のリソースのショートカット・メニューからアク セスできるものがあります。

[関連付けられた関数ライブラリ]

[**関連付けられた関数ライブラリ**]ノードは、現在テストに関連付けられているすべての関数ライブラリを示します。

[**関連付けられた関数ライブラリ**] ルート・ノードのショートカット・メ ニューには, 次のオプションがあります。 ▶ [関数ライブラリの関連付け]: [添付を開く] ダイアログ・ボックスを開き, 関数ライブラリをテストに関連付けることができます。

特定の関数ライブラリのショートカット・メニューには、次のオプションがあ ります。

- ▶ [関数ライブラリを開く]:選択した関数ライブラリを表示領域の関数ライブラ リ・ウィンドウに表示します。関数ライブラリをダブルクリックして開くこと もできます。
- ▶ [関数ライブラリをリストから削除]:選択した関数ライブラリをテストから削除します。
- ▶ [上に移動] または [下に移動]: 選択した関数ライブラリを, 関連付けられている関数ライブラリの優先準位リストで上下に移動します。

詳細については,第28章「ユーザ定義関数および関数ライブラリを使用した 作業」を参照してください。

関連付けられた回復シナリオ

[**関連付けられた回復シナリオ**]ルート・ノードは,現在テストに関連付けられているすべての回復シナリオを示します。

[**関連付けられた回復シナリオ**] ルート・ノードのショートカット・メニュー には,次のオプションがあります。

- ▶ [回復シナリオの関連付け]: [回復シナリオの追加] ダイアログ・ボックスを開きます。[回復シナリオの追加] ダイアログ・ボックスの使用方法の詳細については、1280ページ「テストへの回復シナリオの追加」を参照してください。
 回復シナリオのショートカット・メニューは、次のオプションがあります。
- ➤ [回復シナリオのプロパティ]: [回復シナリオのプロパティ] ダイアログ・ ボックスを開きます。
- ► [回復シナリオをリストから削除]:選択した回復シナリオをテストから削除します。
- ▶ [回復シナリオの無効化] または [回復シナリオの有効化]: 選択した回復シナ リオを無効または有効にします。

▶ [上に移動] または [下に移動]: 選択した回復シナリオを, 関連付けられている回復シナリオの優先準位リストで上下に移動します。

回復シナリオをダブルクリックすると, [回復シナリオマネージャ] ダイアロ グ・ボックスが開きます。

詳細については、第44章「回復シナリオの定義と使用」を参照してください。

アクションごとの関連付けられているリポジトリ

[アクション]は、[アクションごとの関連付けられているリポジトリ]ノード の下で、[内部アクション]と[外部アクション]にグループ化されます。[内 部アクション]と[外部アクション]の下に表示されているのは、アクション に関連付けられているローカルのオブジェクト・リポジトリおよび共有オブ ジェクト・リポジトリです。

注:[リソース表示枠]には、テストとともに格納されたすべてのアクション が、テストによって呼び出されていない場合でも表示されます。アクションを テストとともに格納する方法の詳細については、475ページ「キーワード・ ビューのアクション・ツールバーの使用」を参照してください。

アクションのルート・ノードのショートカット・メニューには,次のオプショ ンがあります。

- ▶ [リポジトリとアクションの関連付け]: [添付を開く] ダイアログ・ボックスを 開き、オブジェクト・リポジトリを選択したアクションに関連付けることがで きます。外部アクションに対しては、このオプションは無効になっています。
- ▶ [アクションのプロパティ]: [アクションのプロパティ] ダイアログ・ボック スを開きます。

アクションのショートカット・メニューには、次のオプションがあります。

▶ [リポジトリを開く]: ローカルのオブジェクト・リポジトリに対して [オブ ジェクト リポジトリ ウィンドウ - ローカル オブジェクト リポジトリ]を開き, 共有オブジェクト・リポジトリに対してオブジェクト・リポジトリ・マネー ジャを開きます。オブジェクト・リポジトリをダブルクリックしても,関連付 けられているリポジトリ・ウィンドウが開きます。

- ▶ [リポジトリをリストから削除]: 選択したオブジェクト・リポジトリをアクションから削除します。
- ▶ [上に移動] または [下に移動]: 選択したオブジェクト・リポジトリを, 関連 付けられているリポジトリの優先準位リストで上下に移動します。

オブジェクト・リポジトリの使用方法の詳細については,第4章「テスト・オ ブジェクトを使用した作業」を参照してください。 第35章・リソースの管理

第 36 章

テストへのキーワードの追加

QuickTest では、1 つの表示枠で利用できるキーワードをテストに表示および追加できます。

本章では、次の項目について説明します。

▶ [使用可能なキーワード]表示枠について(1107ページ)

[使用可能なキーワード] 表示枠について

[使用可能なキーワード] 表示枠には、テストで利用できるキーワードが表示 されます。オブジェクトまたは関数呼び出しを表示したり、テストにドラッ グ・アンド・ドロップしたりできます。オブジェクトをアクションにドラッ グ・アンド・ドロップすると、QuickTest によって、そのオブジェクトの標準操 作を持つステップが挿入されます。関数をテストにドラッグ・アンド・ドロッ プすると、QuickTest によって、その関数の呼び出しが挿入されます。

たとえば、ボタン・オブジェクトをアクションにドラッグ・アンド・ドロップ すると、クリック操作(ボタン・オブジェクトの標準操作)を持つステップが ボタン・オブジェクトを使用して追加されます。

関数をテストにドラッグ・アンド・ドロップすると、コメントおよびその関数 の呼び出しが追加されます。コメントには、関数に対する呼び出しがテストに 追加されたこと、および必要な引数が示されます。その関数の引数をテストに 渡す必要があります。キーワード・ビューでは、関数に必要な引数がツール チップに表示されます。エキスパート・ビューでは、関数に必要な引数が IntelliSense によって表示されます。 ほかの場所からテスト・オブジェクトをドラッグ・アンド・ドロップすること もできます。詳細については、以下を参照してください。

- ▶ 126 ページ「[オブジェクトリポジトリ] ウィンドウについて」
- ▶ 245 ページ「オブジェクト・リポジトリ・マネージャの使用によるテストへの テスト・オブジェクトの追加」
- Ŋ

[使用可能なキーワード]表示枠を表示するには,[使用可能なキーワードウィンドウ]ボタンをクリックするか,または[表示]>[使用可能なキーワード]を選択します。

[使用可能なキーワード] 表示枠には、テストで利用できるキーワードがリ ソースごとにまたはキーワードごとに並べ替えられて表示されます。

リソースで並べ替えられたキーワード

- キーワードをリソースごとに並べ替えて表示するには,[**リソースで並べ替え** る]ボタンをクリックします。キーワードは種類(ライブラリ関数,ローカル 関数,オブジェクト)に応じてグループ化され,次にその種類の特定のリソー スに応じてグループ化されます。
 - ▶ 各関数ライブラリ内の関数はアルファベット順に並んでいます。
 - ▶ 各オブジェクト・リポジトリ内のオブジェクトは、アプリケーション内で表示 されるページまたはウィンドウごとにグループ化され、次にオブジェクトの種 類ごとにグループ化されます。その後、アルファベット順に並べ替えられます。
 - ▶ キーワードを右クリックすると、キーワードのリソースを開いたり、または選択したキーワードをクリップボードにコピーしたりできます。
 - ▶ キーワードをダブルクリックするとキーワードのリソースが開き,選択した キーワードを見つけることができます。

キーワードで並べ替えられたキーワード

キーワードをキーワードごとに並べ替えて表示するには、[キーワードで並べ替 える]ボタンをクリックします。キーワードは、リソースに関係なく種類(ラ イブラリ関数、ローカル関数、オブジェクト)に応じてグループ化されます。

- ▶ すべての利用できる関数は、アルファベット順に並べ替えられます。
- ▶ 利用できるオブジェクトはすべて、アプリケーション内で表示されるページまたはウィンドウに応じてグループ化され、次にオブジェクトの種類に応じてグループ化されます。その後、アルファベット順に並べ替えられます。

注:同じ名前のキーワードが2つある場合は、リソースの優先順位に従って表示されます。

- ▶ キーワードを右クリックすると、キーワードのリソースを開いたり、または選択したキーワードをクリップボードにコピーしたりできます。
- ▶ キーワードをダブルクリックするとキーワードのリソースが開き,選択した キーワードを見つけることができます。

第36章・テストへのキーワードの追加

第 37 章

欠落リソースの処理

欠落リソース(欠落している共有オブジェクト・リポジトリまたは欠落アク ションの呼び出しなど)がテスト内にある場合や,値が定義されていないリポ ジトリ・パラメータがテストで使用されている場合は,[欠落リソース]表示枠 にそのことが示されます。この表示枠に表示されるリソースのいずれかが実行 セッション中に使用できない場合,テストは失敗する場合があります。必要に 応じて,欠落リソースを割り当てるか,またはテストからそれを削除できます。

本章では、次の項目について説明します。

- ▶ 欠落リソースの処理について(1112ページ)
- ▶ 欠落アクションの処理(1114ページ)
- ▶ 欠落している環境変数ファイルの処理(1119ページ)
- ▶ 欠落している関数ライブラリの処理(1120ページ)
- ▶ 欠落している共有オブジェクト・リポジトリの処理(1122ページ)
- ▶ 欠落している回復シナリオの処理(1123ページ)
- ▶ 未割り当ての共有オブジェクト・リポジトリ・パラメータ値の処理(1126 ページ)

欠落リソースの処理について

QuickTest では、テストを開くたびに、そのテストに指定されたリソースが使用 可能かどうかの確認が行われます。

欠落リソースが1つでもあると、まだ開いていなければ、[欠落リソース]表示枠が開きます。[欠落リソース]表示枠には、現在使用できないすべてのリ ソースのリスト、およびリソースを使用できるときに QuickTest によってリ ソースが検索される場所が表示されます。[欠落リソース]表示枠では、これ らを見つけたりテストから削除したりできます。

[欠落リソース]を正常に処理すると、そのリソースは表示枠から削除されます。

欠落リソース	
項目 ▽	詳細
Missing environment variables file: Env_var.xml	L:\QuickTest\Tests\Missing Resources\libraries_Repos\Env_var.xml
Missing Object Repository: Repository_1.tsr	L:\QuickTest\Tests\Missing Resources\libraries_Repos\Repository_1.tsr
🗞 Repository Parameters	Unmapped repository parameters
Missing Recovery Scenario: RS_2	L:\QuickTest\Tests\Missing Resources\libraries_Repos\RS2.qrs
Missing Function Library: Common.txt	Common.txt
Missing Action: Return Home	From test: L:\Quick Test\Tests\actionA

[欠落リソース] 表示枠には、次のタイプの欠落リソースが表示されます。

- ▶ [<欠落アクション名>]:欠落しているアクションがテストに含まれている場合は、QuickTestによって、欠落アクションが含まれているテストを検索するために使用するパスが指定されます。詳細については、1114ページ「欠落アクションの処理」を参照してください。
 - ▶ [欠落している環境変数ファイル]: テストによって外部ファイルからユーザ定義の環境変数が読み込まれようとしたが、このファイルが見つからない場合は、QuickTestによって、欠落している XML ファイルを検索するために使用するパスが指定されます。詳細については、1119ページ「欠落している環境変数ファイルの処理」を参照してください。
 - ► [欠落している関数ライブラリ]: テストが見つからない関数ライブラリに関連 付けられている場合は、QuickTestによって、欠落している関数ライブラリを検 索するために使用するパスが指定されます。詳細については、1120ページ「欠 落している関数ライブラリの処理」を参照してください。
- 「リポジトリパラメータ]: テスト内の少なくとも1つのテスト・オブジェクトのプロパティ値が標準設定値のないリポジトリ・パラメータを使用してパラメータ化されている場合,この一般項目が[欠落リソース]表示枠に追加されます。詳細については、1126ページ「未割り当ての共有オブジェクト・リポジトリ・パラメータ値の処理」を参照してください。

注:欠落リソースが使用されるさまざまな画面(キーワード・ビューやテスト 設定など)において、QuickTest によって、特別なアイコンやテキストを使って リソースが欠落していることが示されます。

[欠落リソース] 表示枠のフィルタ処理

[欠落リソース]表示枠には、すべての欠落リソースを表示することも、欠落 リソースの1つのタイプだけを表示することもできます。

表示された欠落リソースのリストをフィルタ処理するには、次の手順を実行し ます。

[欠落リソース] 表示枠を右クリックして, 次のいずれかを選択します。

- ▶ [すべて]: テスト内のすべての欠落リソースが表示されます。
- ▶ [アクション]: 欠落アクションごとに、欠落アクションが含まれている各テ ストを探すためのパスを指定する1行が表示されます。
- ▶ [環境変数ファイル]: ユーザ定義の環境変数を格納するために使用する外部 XML ファイルが表示されます。
- ▶ [関数ライブラリ]: 欠落している関数ライブラリごとに、関数ライブラリを 検索するために使用するパスを指定する1行が表示されます。

- ▶ [オブジェクトリポジトリ]: 欠落している共有オブジェクト・リポジトリごとに、その共有オブジェクト・リポジトリを探すためのパスを指定する1 行が表示されます。
- ▶ [回復シナリオ]: 欠落している回復シナリオごとに、回復シナリオを検索するために使用するパスを指定する1行が表示されます。
- ▶ [リポジトリパラメータ]: リポジトリ内の少なくとも1つのテスト・オブジェクトに、標準設定値のないリポジトリ・パラメータを使用しているプロパティ値が少なくとも1つあることを示す汎用の行が表示されます。

[欠落リソース]表示枠は,選択したリソース・タイプに従ってフィルタ処理 され,適用されたフィルタを示す次の表示が表示枠の下部に示されます。

× []= オブジェクトリボジトリ

フィルタ表示の左側のアイコン■をクリックすると、フィルタがキャンセルされ、すべての欠落リソースが再び表示されます。

欠落アクションの処理

テストに含まれているアクションの呼び出しで、見つからないものがある場合 は、それらのアクションが[欠落リソース]表示枠に表示されます。

欠			
	項目	詳細	
2)欠落アクション: ReturnHome	From test: L'\QuickTest\Tests\actionA	
			-1
4		•	
5	ActiveScreen 🤽 欠落リソース		

さらに、テストに含まれている特定のアクションに対する呼び出しがテスト・フローに含まれているが、そのアクションが見つからない場合は、QuickTest によってそのアクションが [欠落リソース] 表示枠に表示されます。たとえば、テスト作成時に新規の再利用可能なアクションの呼び出しを挿入したとします。その後、(499ページで説明している) [アクションを削除] ダイアログ・ボックスの [選択したアクションの呼び出しを削除する] を選択して、そのアクションの呼び出しを削除したとします。この場合、アクションの呼び出しを 削除しても、そのアクションはテストに参照されたままであるため、見つからない場合は [欠落リソース] 表示枠に表示されます。 [欠落リソース] 表示枠では,次の操作によって欠落アクションの問題を解決 できます。

- ▶ 欠落アクションの特定
- ▶ 欠落アクションの削除

注: テストが読み取り専用形式で開かれている場合は、テスト内の欠落アクションの表示や割り当てを行うことはできません。

欠落アクションの特定

[欠落リソース]表示枠では、テスト内の欠落しているアクションを探せます。 テストに複数の欠落アクションに対する呼び出しが含まれている場合、別のテ ストで欠落アクションを見つけると、QuickTestによって、同じテスト内で他の 欠落アクションが見つかる場合があります(これは、たとえば、呼び出される アクションが含まれているソース・テストの名前が変更されたり、別のフォル ダに移動されたりした場合に起こることがあります)。

これらのアクションの呼び出しを同時に見つけるように QuickTest に指定できます。また、欠落アクションの呼び出しを個別に処理することもできます。

欠落アクションを見つけるには、次の手順を実行します。

1 [欠落リソース] 表示枠で,検出するアクションを右クリックしてショート カット・メニューから [検索] を選択するか,検出するアクションをダブルク リックします。

欠落リソーλ ×			×
項目		詳細	-
饕️欠落アクション: ReturnHome	検索	From test: L:\QuickTest\Tests\actionA	
	削除		
•	Filter 🔸		-
ActiveScreen 公 欠落リソース			

[アクションの選択]	ダイアログ・	・ボックスが開きます。
------------	--------	-------------

アクションの選択		×
テスト(<u>T</u>):		
Webservice		•
アクション(A):		
,		
アクションの記述:		
		<u> </u>
OK	キャンセル	~JL7

[アクションの選択] ダイアログ・ボックスが開くと, [**テスト**] ボックスに欠 落アクションを含むテストの名前(QuickTest がソース・テストを識別できる場 合)または「**く現在のテスト>**」が表示されます。

注:欠落アクションが別のテストから呼び出されたネストしたアクションであ る場合は、[**検索**] ボタンを使ってそのアクションを参照することはできませ ん。その場合は、外部テストの中から欠落アクションを解決する必要がありま す。たとえば、(テストAの) アクション AAが (テストBの) アクション BB を呼び出し、アクション BBが (テストCの) アクション CC を呼び出すとし ます。ここで、テストAを開いてもアクション CC の呼び出しが見つからない 場合は、テストBを開いてアクション CC を見つけないと、欠落アクションを 解決できません (テストAの内部では解決できません)。 2 見つけ出したいアクションが含まれているテストを探すには、参照ボタンをク リックします。[アクション]ボックスには、選択したテストのすべての再利 用可能なアクションが表示されます。

注:

- ▶ テストを選択すると、[テスト] ボックスの名前が [コピー元のテスト] に 変わります。選択したテストに再利用可能なアクションが含まれている場合 は、それらのアクションが [アクション] ボックスに表示されます。
- ▶ [テスト] または [コピー元のテスト] ボックスには, Quality Center フォル ダまたは相対パスを入力できます。相対パスを入力すると, [オプション] ダイアログ・ボックスの [フォルダ] タブで指定されているフォルダの中 で, QuickTest によってテストが検索されます。詳細については, 1169 ペー ジ「テストのフォルダ・オプションの設定」および 340 ページ「QuickTest での相対パスの使用」を参照してください。
- 3 [アクション] リストで,呼び出すアクションを選択します。アクションを選 択すると,そのタイプ(再利用可能なアクション)および存在する場合は説明 が表示されます。これによって,呼び出すアクションを識別できます。アク ションの説明の詳細については、484ページ「一般的なアクション・プロパ ティの設定」を参照してください。
- 4 [OK] をクリックします。QuickTest によって、変更を反映してテストが更新され、アクションが [欠落リソース] 表示枠から削除されます。

注:同じテストの中で見つかる欠落アクションがほかにもテストに含まれてい る場合,メッセージ・ボックスが開き,それらのアクションも割り当てるかど うか確認を求められます。該当するアクションをすべて割り当てるには,[は い]をクリックします。指定したアクションのみを割り当てるには,[いいえ] をクリックします。

欠落アクションの削除

不要になった欠落アクションは、テストから削除できます。

欠落アクションを削除するには、削除するアクションを [欠落リソース] 表示 枠の中で右クリックし、ショートカット・メニューから [**削除**] を選択します。

次落リンース ×			
項目		詳細	
饕餮欠落アクション:ReturnHome	検索	From test: L:\QuickTest\Tests\actionA	
	肖耶余		
•	Filter 🔸		
🐴 ActiveScreen 🗳 欠落リソース			

確認ダイアログが表示されます。[OK] をクリックして欠落アクションを削除 します。QuickTestによってアクションがテストおよび[欠落リソース]表示枠 から削除されます。

注: テストにほかにも同じテスト内に欠落アクションがある場合,パスが同じ すべてのアクションを削除するかどうかを確認するメッセージ・ボックスが表 示されます。同じパス内の欠落アクションをすべて削除するには,[**はい**]を クリックします。指定したアクションのみを削除するには,[**いいえ**]をク リックします。

欠落している環境変数ファイルの処理

外部の環境変数ファイルを使用するテストを開くと、QuickTest によって、ファ イルがアクセス可能であることが確認されます。外部の環境変数ファイルが見 つからない場合は、テストを開いたときに、見つからない外部環境変数ファイ ルの名前とパスが[欠落リソース]表示枠に表示されます。

欠落	欠落リソース		
	項目	詳細	
xmb	欠落している環境変数ファイル: Env_var_xml	L:\QuickTest\Test\Missing Resources\libraries_Repos\Env_var.xml	
			-
l a l			
U -			-

[欠落リソース]表示枠では、欠落している外部環境変数ファイルを見つける か削除することによって問題を解決できます。

欠落している外部環境変数ファイルを見つけるには、次の手順を実行します。

1 検出する欠落環境変数ファイルを右クリックしてショートカット・メニューから [検索]を選択するか、検出する欠落環境変数ファイルをダブルクリックします。

欠落リソース		×
項目	「詳細	<u> </u>
[●] ② 欠落している環境変数ファイル: Env_var.xml	検索 背膝 目标	_
1 一 ActiveScreen ふ 欠落リソース		

[環境変数ファイルを探す] ダイアログ・ボックスが開きます。

2 テストで使用する環境変数ファイルを参照して [**開く**]をクリックします。選 択した環境変数ファイルがテストで使用され、欠落環境変数ファイルは [欠落 リソース] 表示枠から削除されます。

欠落環境変数ファイルを削除するには、削除する欠落環境変数ファイルを右ク リックし、ショートカット・メニューから [**削除**]を選択します。確認ダイア ログが表示されます。[**OK**]をクリックして欠落環境変数を削除します。欠落 環境変数ファイルが、テストと [欠落リソース]表示枠から削除されます。

欠落リソース		×
項目	詳細	
😻 欠落している環境変数ファイル・Env_var.xml	検索 削除 消除	
•	Filter	
ActiveScreen 必欠落リソース		

欠落している関数ライブラリの処理

関数ライブラリが関連付けられているテストを開くと、QuickTest によって、指定したライブラリがアクセス可能であることが確認されます。関数ライブラリが見つからない場合は、テストを開いたときに、見つからない関数ライブラリの名前とパスが[欠落リソース]表示枠に表示されます。

欠落リソース		×
項目	詳細	
📑 欠落している関数ライブラリ: function1.qfl	Tests¥function1.qfl	
× []= '関数ライブラリ'		
•		
②欠落リソース ActiveScreen		

[欠落リソース]表示枠では、欠落している関数ライブラリを見つけるか削除 することによって問題を解決できます。

欠落している関数ライブラリを見つけるには、次の手順を実行します。

 検出する欠落関数ライブラリを右クリックしてショートカット・メニューから [検索]を選択するか、検出する欠落関数ライブラリをダブルクリックします。

欠落リソース ×		
項目	詳細	
京 欠落している関数ライブラリ: function1 検索 削除	frests≇function1.qtj	
X [] = '関数ライブラリ' Filte	r •	
📀 欠落リソース 📲 ActiveScreen		

[関数ライブラリの検索] ダイアログ・ボックスが開きます。

2 テストに関連付ける関数ライブラリを参照して[開く]をクリックします。選択した関数ライブラリがテストに関連付けられ、欠落している関数ライブラリが「欠落リソース]表示枠から削除されます。

欠落関数ライブラリを削除するには、削除する欠落関数ライブラリを右クリックし、ショートカット・メニューから [**削除**]を選択します。確認ダイアログが表示されます。[OK]をクリックして関数ライブラリを削除します。欠落関数ライブラリが、テストと [欠落リソース]表示枠から削除されます。

次落 リソース ×		
項目	詳細	
 次落している関数ライブラリ: function1 。 検索 削除 	f⊤ist¥function1.qt	
▲ 1 - bass y 1999 Filter ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	>	

注: 関数ライブラリがテストから削除されても、これらの関数に対する呼び出しはテストから削除されません。

欠落している共有オブジェクト・リポジトリの処理

共有オブジェクト・リポジトリをアクションに関連付けると,指定したリポジ トリがアクセス可能かどうかが確認されます。また、テストを開くたびに、関 連付けられたすべての共有オブジェクト・リポジトリがアクセス可能かどうか も確認されます。共有オブジェクト・リポジトリが見つからない場合は、テス トを開いたときに、見つからないリポジトリの名前とパスが[欠落リソース] 表示枠に表示されます。

次》	r落Jソース ×		
	項目 て	詳細	
	欠落しているオブジェクトリポジトリ: test.tsr	C:¥Program Files¥HF¥QuickTest Professiona¥Tests¥Tutorial_ObjectReposotpries	
1	ActiveScreen 🔏 欠落リソース		

たとえば,共有オブジェクト・リポジトリの名前やリポジトリが格納されているフォルダを変更した場合は,その共有オブジェクト・リポジトリをテストに割り当てる必要があります。

欠落しているオブジェクト・リポジトリが表示されている行を右クリックして [**解決**] を選択するか,または欠落しているオブジェクト・リポジトリが表示 されている行をダブルクリックすると,[リポジトリの関連付け] ダイアログ・ ボックスが開きます。[リポジトリの関連付け] ダイアログ・ボックスでは,1 つ以上の共有オブジェクト・リポジトリをテスト内の1つ以上のアクションに 関連付けることができます。また,選択したアクションまたはテスト内のすべ てのアクションからオブジェクト・リポジトリの関連付けを削除することもで きます。

注: [リポジトリの関連付け] ダイアログ・ボックスを使用して,新しいオブ ジェクト・リポジトリをテストに関連付けることによって,欠落しているオブ ジェクト・リポジトリを解決します。欠落しているオブジェクト・リポジトリ はテストに関連付けられたままなので,[欠落リソース]表示枠に表示されま す。欠落オブジェクト・リポジトリを[欠落リソース]表示枠およびテストか ら削除するには,[リポジトリの関連付け]ダイアログ・ボックスの[リポジ トリを削除]機能を使用する必要があります。 詳細については、191ページ「共有オブジェクト・リポジトリの関連付けの管理」を参照してください。

欠落している回復シナリオの処理

回復シナリオが関連付けられているテストを開くと、QuickTest によって、指定 したシナリオがアクセス可能であることが確認されます。回復シナリオが見つ からない場合は、テストを開いたときに、見つからないリポジトリの名前とパ スが[欠落リソース]表示枠に表示されます。

欠落リソース		×
項目	「詳細	
₩ 欠落している回復シナリオ: RS_2	L:\QuickTest\Test\Missing Resources\libraries_Repos\RS2.qrs	
		_
		-1
•		Þ
- ActiveScreen 💰 欠落リソース		

[欠落リソース]表示枠では、次の操作によって欠落している回復シナリオを 解決できます。

- ▶ 欠落回復シナリオの特定
- ▶ 欠落している回復シナリオの削除

欠落回復シナリオの特定

[欠落リソース]表示枠では、テスト内で欠落している回復シナリオを探せま す。テストに複数の欠落回復シナリオが含まれている場合、回復ファイルで欠 落シナリオを見つけると、QuickTestによって、そのファイル内でほかの欠落シ ナリオが見つかる場合があります。これらの欠落回復シナリオを同時に見つけ るようにQuickTestを指定できます。また、欠落シナリオを個別に処理するこ ともできます。

欠落回復シナリオを見つけるには、次の手順を実行します。

1 [欠落リソース] 表示枠で,検出する回復シナリオを右クリックしてショート カット・メニューから [検索] を選択するか,検出する回復シナリオをダブル クリックします。

欠落リソース		x
項目	「詳細	
√☆欠落している回復シナリオ: RS_2 ▲	QuickTest\Test\Missing Resources\libraries_Repos\RS2.qrs 削除 Filter ・	
🐴 ActiveScreen 🤌 欠落リソース		

[回復シナリオの検索]ダイアログ・ボックスが開きます。

回復シナリオの検索	<
□復ファイル:	
シナリオ:	
OK キャンセル ヘルプ	

- 2 [参照] ボタンをクリックして回復ファイルを選択します。[シナリオ] 領域 に,選択した回復ファイルに含まれているすべてのシナリオが表示されます。
 - 3 欠落している回復シナリオを回復シナリオのリストから選択します。[OK] を クリックします。選択した回復シナリオがテストに関連付けられ、欠落してい る回復シナリオが [欠落リソース] 表示枠から削除されます。

....

注:同じ回復ファイルの中で見つかる欠落回復シナリオがほかにもテストに含まれている場合、メッセージ・ボックスが開き、それらの回復シナリオも割り当てるかどうかどうか確認を求められます。欠落している回復シナリオをすべて割り当てるには、[**はい**]をクリックします。指定したシナリオのみを割り当てるには、[**いいえ**]をクリックします。

欠落している回復シナリオの削除

不要になった欠落回復シナリオは、テストから削除できます。

欠落回復シナリオを削除するには、削除する回復シナリオを[欠落リソース] 表示枠の中で右クリックし、ショートカット・メニューから [**削除**] を選択し ます。確認ダイアログが表示されます。[**OK**] をクリックして、回復シナリオ を削除します。欠落回復シナリオが、テストと [欠落リソース] 表示枠から削 除されます。

欠素	創ソース			×
	項目		言¥細	
√	欠落している回復シナリオ: RS_2	検索 削除 	Quick Test\Test\Missing Resources\libraries_Repos\RS2.grs	
-5	ActiveScreen 🔥 欠落リソース			

未割り当ての共有オブジェクト・リポジトリ・パラメータ値の処理

テスト内で使用するすべてのリポジトリ・パラメータに,値が指定されている 必要があります。この値は、パラメータの作成時に指定された標準設定値か、 またはテスト内で指定した値です(リポジトリ・パラメータの詳細については、 248ページ「リポジトリ・パラメータを使用した作業」を参照してください)。

値がないリポジトリ・パラメータを使って値がパラメータ化されたオブジェクト・プロパティを持つオブジェクト・リポジトリを使用するテストを開くと、 [欠落リソース]表示枠に [リポジトリパラメータ] が表示され、この状況が示されます。



たとえば、前の画面の選択内容に応じて名前のプロパティが変化するエディッ ト・ボックスがアプリケーション内にあるとします。リポジトリ・パラメータ を使ってオブジェクト・リポジトリ内の名前プロパティの値をパラメータ化し たにもかかわらず、そのリポジトリ・パラメータの標準設定値が定義されてい ない場合は、パラメータの値を定義する必要があります。この値をデータ・ テーブル・パラメータ、環境変数パラメータ、乱数パラメータ、テスト・パラ メータ、またはアクション・パラメータに割り当てることができます。また、 定数値を定義することなどもできます。

[リポジトリパラメータ]が表示されている行を右クリックして [解決]を選 択するか,または [リポジトリパラメータ]が表示されている行をダブルク リックすると,[リポジトリパラメータの割り当て]ダイアログ・ボックスが 開き,割り当てられていないオブジェクト・リポジトリ・パラメータの値を指 定できます。このダイアログ・ボックスをフィルタ処理して,未割り当てのパ ラメータのみを表示したり,テストまたは特定のアクション内の(値が割り当 て済みおよび未割り当ての)パラメータをすべて表示したりできます。詳細に ついては,158ページ「リポジトリ・パラメータ値の割り当て」を参照してく ださい。

第 38 章

データ・テーブルを使った作業

QuickTest では、データ・テーブルに格納されたデータを使って動作するステップを挿入し、実行できます。

本章では、次の項目について説明します。

- ▶ データ・テーブルを使った作業について(1127ページ)
- ▶ グローバル・シートおよびアクション・シートを使った作業(1130ページ)
- ▶ データ・テーブルの保存(1131ページ)
- ▶ データ・テーブルの編集(1132ページ)
- ▶ Quality Center でのデータ・テーブル・ファイルの使用(1141 ページ)
- ➤ データベースからのデータのインポート(1142ページ)
- ▶ データ・テーブルでの数式の使用(1145ページ)
- ▶ データ・テーブルのスクリプト・メソッドの使用(1149ページ)

データ・テーブルを使った作業について

テストによって使用されるデータは「設計時」データ・テーブルに格納されま す。これはステップを挿入,編集するときに画面下部の[データテーブル]表 示枠に表示されます。

データ・テーブルには、Microsoft Excel のスプレッドシートと同じ機能があり、 セルでデータを格納、使用し、またセル内で数式を計算することもできます。 データ・テーブル内の任意のセルでデータを操作するには、DataTable、 DTSheet および DTParameter ユーティリティ・オブジェクトを使用できます。 これらのオブジェクトの詳細については,『**HP QuickTest Professional Object Model Reference**』(英語版)の「**Utility**」の項を参照してください。

注: データ・テーブルでは,複雑な式またはネストされた式の使用はサポート されていません。

テストにはデータ・テーブル・パラメータと出力値を挿入できます。テストで データ・テーブル・パラメータまたは出力値を使用すると、指定したデータを 使用して複数回実行する「データ駆動」テストまたはアクションを作成できま す。実行の繰り返し、つまり反復のたびに、QuickTest によってデータ・テーブ ルの異なる値が使用されます。

実行セッション中,QuickTestによって実行時データ・テーブルが作成されま す。これは、テストに関連するデータ・テーブルの実使用版です。実行セッ ション中,QuickTestによって[データテーブル]表示枠に実行時データが表 示され、データ・テーブルに変更が加えられる様子を確認できます。 実行セッション終了時に、実行時データ・テーブルが閉じられ、[データテー ブル]表示枠には格納されている設計時データ・テーブルが再度表示されま す。実行セッション中に実行時データ・テーブルに入力されたデータは、テス トとともには保存されません。実行時データ・テーブルの最終的なデータは、 [テスト結果]ウィンドウの[**実行時データ テーブル**]に表示されます。実行 時データ・テーブルの詳細については、1028ページ「実行時データ・テーブル の表示」を参照してください。

ヒント:実行時データ・テーブルの結果データを保存しておく必要がある場合 は、DataTable.Export ステートメントをテストの最後に挿入し、実行時デー タ・テーブルをファイルにエクスポートします。これによって、データ・テー ブルを右クリックして表示されるメニューから [ファイル] > [ファイルから インポート] を選択して、データを設計時データ・テーブルにインポートでき ます。あるいは、テストの最初に DataTable.Import ステートメントを追加し て、前回の実行セッションの最後にエクスポートされた実行時データ・テーブ ルをインポートできます。これらのメソッドの詳細については、

『HP QuickTest Professional Object Model Reference』(英語版)を参照してくだ さい。

グローバル・シートおよびアクション・シートを使った作業

テストを使って作業する際,データ・テーブルには,[Global]と[Action] の2種類のデータ・シートが表示されます。それぞれのシートにアクセスする には,データ・テーブルの下のそれぞれのタブをクリックします。

- ▶ データをテスト内のすべてのアクションで使用できるようにし、また多くのテ スト反復を制御できるようにする場合は、[Global] タブに保存します。
- ➤ データ・テーブル・パラメータのデータを特定のアクションにのみ使用できる ようにし、多くのアクション反復をするようにする場合は、データを [Action] のタブに保存します。

たとえば、サンプルの Mercury Tours Web サイトを対象としたテストを作成す るとします。この場合、ログイン、フライト予約、ログアウトのそれぞれに対 してアクションを1つずつ作成することになるでしょう。ユーザが1度サイト にログインし、5名分のフライトを予約するテストを作成する必要があるとし ます。乗客に関するデータは2つ目のアクションにだけ関係するため、データ はそのアクションに対応するアクション・タブに格納するようにします。

グローバル・シート

グローバル・シートには、テストの反復ごとにパラメータを置換するデータが 格納されています。Arrivals というグローバル・パラメータを作成すると、グ ローバル・シートは次のようになります。

A1	l Paris							
	Arrivals	В	С	D	E	F	G	
1	San Francisco							
2	New York							
3	Paris							
4								
5								
6								-
••\	אוא–חט אווא–םל	1		· · · · · ·				

アクション・シート

新しいアクションをテストに追加するたびに,新しい**アクション・シート**が データ・テーブルに追加されます。アクション・シートのラベルには,対応す るアクションの名前がそのまま自動的に付けられます。アクション・シートに 格納されているデータは,対応するアクションのデータ・テーブル・パラメー タにだけ関連しています。たとえば,テストに次のようなデータ・テーブルが ある場合,**Purchase**アクション内でアクション・パラメータ・ステップを反 復実行するとき,QuickTestによって[Purchase]シートに格納されているデー タが使用されます。

F3	F3							
	Departure	В	С	D	E	F	G	H
1	New York							
2	Paris							
3	Los Angeles							
4								
5								
6								
∎P∑	↓ 「「グローバル 入 Purchase / 」							

グローバル・パラメータおよびアクション・パラメータの作成の詳細について は、第22章「値のパラメータ化」を参照してください。

データ・テーブルの保存

データ・テーブルには、入力した値や数式だけでなく、テスト実行時にデー タ・テーブル・パラメータを置き換えるために QuickTest によって使用される 値が含まれます。テストを保存するとき、QuickTest によって、テストのデー タ・テーブルが自動的に.xls ファイルとして保存されます。

テストを使って作業を行う場合,標準設定ではデータ・テーブルはテストと一緒 に保存されます。データ・テーブルを別の場所に保存し,テスト実行時にこの データ・テーブルを使用することができます。データ・テーブルの名前と場所 は,[テストの設定]ダイアログ・ボックスの[リソース]タブで指定します。

[テストの設定] ダイアログ・ボックスの詳細については,第41章「個別のテ ストのオプション設定」を参照してください。 指定した場所にデータ・テーブルを保存すると、次のような環境で便利です。

- ▶ 同じテストを異なる入力値のセットを使って実行する場合。たとえば、テスト する言語ごとに、異なるデータ・テーブル・ファイルを用意してテストを実行 することで、アプリケーションのローカライズ機能をテストできます。テスト を実行するたびに異なる環境パラメータ・ファイルを使用することで、検査対 象ユーザ・インタフェース文字列を言語ごとに変更することもできます。詳細 については、第22章「値のパラメータ化」を参照してください。
- ▶ 異なるテストに関して、同じ入力情報が必要である場合。たとえば、使用する テストは異なるけれども使用するデータ・テーブル・ファイルが同じである、 同じアプリケーションの Web バージョンと標準 Windows バージョンをテスト できます。

注:外部ファイルをデータ・テーブルとして選択した場合,外部データ・テー ブルのカラム名はテストのパラメータ名と一致し,外部データ・テーブルの シートはテストのアクション名と一致する必要があります。

データ・テーブルの編集

データ・テーブルの情報は、テーブル・セルに直接入力して編集できます。 データ・テーブルは、Microsoft Excel スプレッドシートと同じように使います (数式をセルに挿入するなど)。また、Microsoft Excel、タブ区切りのテキスト・ ファイル (.txt)、または ASCII 形式で保存されたデータをインポートすること もできます。

サポートされている Microsoft Excel のバージョンの詳細については, 『HP QuickTest Professional 最初にお読みください』を参照してください。



A1	A1 Acapulco							
	departure	arrival	С	D	E	F	G	4
1	Acapulco	New York						
2	New York	Paris						
3	London	Frankfurt						
4								
5								•
IIN								

テーブルの各行は、テストまたはアクションの特定の反復中に、QuickTest に よって、パラメータ化された引数に対して送信される値のセットを表します。 [アクションのプロパティ]ダイアログ・ボックスの[実行]タブで選択した 設定に基づき、QuickTest によって、アクションの反復が実行されます。テスト が実行する反復の回数は、グローバル・シートの行数と同じです。

テーブルの各**カラム**は、パラメータ化された特定の引数に対する値のリストです。カラムのヘッダは、パラメータ名です。

また,データ・テーブル・パラメータとともに使用することを前提としていな いカラム(カラムのヘッダにパラメータ名がないカラム)のセルにデータおよ び数式を入力することもできます。

データ・テーブルを使用した作業に関するガイドライン

- ▶ データ・テーブルにデータを追加する場合、データの入力は、上から下の行、 左から右のカラムの順に行う必要があります。行またはカラムに何も入力せず に空けておくことはできません。たとえば、行1にデータが存在する場合、行 2にデータを入力するまで、行3のセルにデータを入力することはできません。 同様に、カラムAにデータが存在する場合、カラムBにデータを入力するまで、カラムCにデータを入力することはできません。
- ➤ データ・テーブルから返される値は常に文字列に変換されます。値を文字列以 外に変換する場合は、Clnt、CLng、CDbl など、VBScriptの変換機能を使用で きます。例を次に示します。
 - Window("Flight Reservation").WinComboBox("Fly From:").Select CInt(DataTable("ItemNumber", dtGlobalSheet))

##

➤ データ・テーブルのセルに内容を追加すると、行の下側のグリッド線がグレーから黒に変わります。[すべての行で実行する]オプションを使ってテストを実行すると([ファイル]> [設定]> [実行]タブ、または[編集] [アクション]> [アクション呼び出しプロパティ]> [実行] タブで定義されます)、下側のグリッド線が黒の行ごとに反復を1回実行します。[すべての行で実行する]オプションを選択したときに、特定の行に対して反復を実行したくない場合、[編集]> [削除]オプション (CTRL+K)を使用してデータ・テーブルからその行全体を削除する必要があります(削除すると、下側のグリッド線が黒からグレーに戻ります)。テーブルを右クリックして表示されるメニューから[編集]> [クリア]オプションを使用するか(または CTRL+X キー)、セルを選択してキーボードの Delete キーを押すと、データはセルから削除されますが、行は削除されず、黒の線も残ります。これは、行にデータが存在しなくても、QuickTest によってこの行が反復実行されることを意味します。

データ・テーブルの仕様

データ・テーブルを使った作業の主な制限事項は次のとおりです。

- ▶ 最大ワークシート・サイズ: 65,536 行, 256 カラム
- ▶ カラム幅:0から255 文字
- ▶ テキストの長さ: 16,383 文字
- ▶ 数式の長さ: 1024 文字
- ➤ 数値の精度:15 桁
- ▶ 整数の最大値: 9.9999999999999998307
- ▶ 負の数の最大値: 9.9999999999999998307
- ▶ 整数の最小値: 1E-307
- ▶ 負の数の最小値: 1E-307
- ▶ ワークブックごとの名前の最大数:使用できるメモリによって制限されます。
- ➤ 最長の名前: 255
- ➤ 最短の数式文字列:255
- ▶ テーブルの最大数(ワークブック):システム・リソース(ウィンドウとメモリ)によって制限されます。
- ▶ データ・テーブルでは、色の使用と書式設定は行えません。

- ▶ データ・テーブルでは、複雑な数式およびネストされた数式はサポートされていません。
- ▶ データ・テーブルでは、コンボ・ボックスやリスト・セル、条件付き書式などの特別なセル書式はサポートされていません。

カラム名の変更

カラム・ヘッダのセルをダブルクリックすると、パラメータのカラム名を変更 できます。[パラメータ名の変更]ダイアログ・ボックスで、新しいパラメー タ名を入力します。このパラメータ名は、テスト内で一意でなければなりませ ん。名前には、英字、数字、カンマ、アンダスコアを使用できます。ただし、 最初の文字は、英字またはアンダスコアである必要があります。

注: テーブル内のカラム名を変更したら,テスト内の対応するパラメータに対して定義されている名前も変更する必要があります。

データ・テーブル・メニュー・コマンドの使用

次のデータ・テーブル・メニュー・コマンドを使用すると、データ・テーブル のデータを編集できます。データ・テーブル・メニューを開くには、セル、行 のヘッダ、またはカラム・ヘッダを右クリックします。

次のメニューを使用できます。

- ▶ [ファイル] メニュー
- ▶ [シート] メニュー
- ▶ [編集] メニュー
- ▶ [データ] メニュー
- ▶ [書式] メニュー

[ファイル] メニュー

[**ファイル**] メニューでは,次のコマンドを使用できます。

[ファイル] のコ マンド	詳細
[ファイルからイ ンポート]	既存の Microsoft Excel ファイルまたはタブ区切りのファイルを データ・テーブルにインポートします。このコマンドは,選択 した Microsoft Excel ファイルのすべてのシートをインポートし ます。既存の Microsoft Excel ファイルからシートを1枚だけイ ンポートするには,下記のように[シート] > [インポート] > [ファイルから] コマンドを使用します。
	 注: インポートするテーブル・ファイルは、テーブルのすべての シートのデータをすべて置き換えます。また、各 Microsoft Excel シートの最初の行は、対応するデータ・テーブル・シー トのカラム・ヘッダに置き換えられます。したがって、 Microsoft Excel シートの最初の行がテストのパラメータ名に正 しく一致しており、ファイルには現在のデータ・テーブルと 少なくとも同数のシートが含まれていなければなりません。 コンボ・ボックスやリスト・セル、条件付き書式などの特別 なセル書式が含まれる Microsoft Excel テーブルをインポート する場合、書式はインポートされず、セルは固定値とともに データ・テーブルに表示されます。
[エクスポート]	指定された Microsoft Excel (.xls) ファイルにテーブルをエクス ポートします。
[印刷]	テーブル全体または選択したシートを印刷します。

[シート] メニュー

[**シート**] メニューでは,次のコマンドを使用できます。

[シート] のコマンド	詳細
[インポート]> [ファイルから]	タブ区切りのテキスト・ファイルまたは既存の Microsoft Excel ファイルからの1シートがテーブルにインポートされます。
	注 : インポートするシートは,テーブルで現在選択されてい るシートのデータをすべて置き換え,Excel シートの最初の行 は,対応するデータ・テーブル・シートのカラム・ヘッダを 置き換えます。したがって,Microsoft Excel シートの最初の 行がテストのパラメータ名に正しく一致していなければなり ません。
[インポート]> [データベースから]	指定されたデータベースから現在のシートにデータをイン ポートします。
[エクスポート]	指定された Microsoft Excel(.xls)ファイルに,データ・テー ブルの現在のシートをエクスポートします。

[編集] メニュー

[編集] メニューでは、次のコマンドを使用できます。

[編集] のコマンド	詳細
[切り取り]	テーブルの選択されている範囲を切り取り,クリップボードに 格納します。
[コピー]	テーブルの選択されている範囲をコピーし,クリップボードに 格納します。
[貼り付け]	クリップボードの内容を,現在のテーブルの選択範囲に貼り付 ける。
[値を貼り付け]	クリップボードに格納されている値を,現在テーブルで選択さ れている範囲に貼り付けます。値に適用されている書式はすべ て無視されます。さらに,数式の計算結果だけが貼り付けられ, 数式は無視されます。
[クリア]	現在選択されている部分から書式または内容を消去します。書 式のみ,内容(数式も含む)のみ,または書式と内容の両方を 消去できます。

[編集] のコマンド	詳細
[挿入]	現在選択されている行の上,または列の左に,新しい行または 列が挿入されます。新しいセルのための場所を空けるために, 挿入部分に隣接するセルがずらされます。このオプションが使 用できるのは,行のヘッダまたはカラム・ヘッダが選択されて いる場合のみです。
[削除]	現在選択されている行とカラムの全体を削除します。削除された セルによってできたスペースを埋めるために,削除部分に隣接す るセルがずらされます。このオプションが使用できるのは,行の ヘッダまたはカラム・ヘッダが選択されている場合のみです。
[右方向へコピー]	選択された範囲内の一番左にあるセルのデータを、その範囲内 でそのセルの右にあるすべてのセルにコピーします。
[縦方向にコピー]	選択された範囲内の一番上にあるセルのデータを、その範囲内 でそのセルの下にあるすべてのセルにコピーします。
[検索]	指定されたテキストを含むセルを検索します。テーブルの行ご とまたはカラムごとの検索,大文字と小文字を区別した検索, セル全体が一致するものだけの検索ができます。また,数式や 値の検索も可能です。
[置換]	指定されたテキストを含むセルを見つけ,これを別のテキスト に置き換えます。テーブルの行ごとまたはカラムごとの検索, 大文字と小文字を区別した検索,セル全体が一致するものだけ の検索ができます。また,数式や値の検索も可能です。見つ かったテキストのすべてのインスタンスを置換することもでき ます。
[移動]	指定されたセルに移動します。移動先のセルがアクティブにな ります。セルのカラムや行の番号を入力する必要があります。

[データ] メニュー

[**デー**タ] メニューでは,次のコマンドを使用できます。

[データ] の コマンド	詳細
[再計算]	テーブル内のすべての数式のセルを再計算します。
[並べ替え]	選択したセルを, 行またはカラムおよびキーに従って, 昇順ま たは降順で並べ替えます。
[オートフィルの リスト]	 オートフィルのリストを作成,編集,または削除します。オートフィルのリストには、月や曜日など、よく使われる一連のテキストが含まれています。オートフィルのリストを使うには、最初の項目をテーブルのセルに入力します。カーソルをセルの右下の角からドラッグすると、オートフィルのリストに従い、QuickTest によって範囲内のセルが自動的に埋められます。 「リスト]:プロジェクトで使用できるリスト。4つの標準設定のリストが含まれています。 「現在のリスト]:選択したリスト。このウィンドウは、新しいリストの作成に使用できます。新しいリストの各項目はセミコロンで区切ります。 「追加]:新しいリストを[リスト] ボックスに追加します。 「削除]:[リスト] ボックスからリストを削除します。 「開く]:[ファイルを開く] ダイアログ・ボックスが開きます。このダイアログ・ボックスで、あらかじめ作成しておいたリストを参照できます。 「保存]:[名前を付けて保存] ダイアログ・ボックスが開きます。このダイアログ・ボックスを使って、新しいリストを保存できます。
[暗号化]	選択したセルのテキストを暗号化します。暗号化されたデータ は復号できません。 また,パスワード・エンコーダを使って,任意のテキスト文字列
	を暗号化することもできます。これは、[エキスパートビュー] でメソッドの引数として暗号化した文字列を入力するのに便利で す。詳細については、418ページ「メソッド引数およびデータ・ テーブル・セルへの暗号化されたパスワードの挿入」を参照して ください。

[書式] メニュー

[書式] メニューでは、次のコマンドを使用できます。

[書式] の コマンド	詳細
[一般]	ー般書式に設定します。一般書式では,小数位は入力したとお りに表示されます。カンマは表示されません。
[通貨]	書式の設定をカンマ区切りで小数点を使わない通貨形式にしま す。QuickTest では,Windowsの [地域のオプション]ダイアロ グ・ボックスで定義された通貨記号が使用されます。
[通貨]	通貨の書式に設定します。カンマ(桁区切り)を使用し、小数 位は2桁表示されます。QuickTest では、Windowsの[地域のオ プション]ダイアログ・ボックスで定義された通貨記号が使用 されます。
[固定]	固定の精度の書式に設定します。カンマ(桁区切り)を使用し, 小数位は表示されません。
[パーセント]	パーセントの書式に設定します。小数位は表示されません。数 字を百分率で表示し、数字の後にパーセント記号(%)が表示 されます。
[分数]	1/2 など、分子 / 分母の形式の分数に書式を設定します。
[指数]	小数位2桁の指数表現の書式に設定します。
[日付(ダイナ ミック)]	[月 / 日 / 年] の日付書式に設定します。
[午前 / 午後 時: 分]	[午前 / 午後 時:分]の時間書式に設定します。
[ユーザ定義数]	ユーザ定義の数値の書式に設定します。このオプションにより, パーセンテージ,通貨,日付,時間などに関して,特別なユー ザ定義書式を設定できます。

ショートカット・キーを使用してデータ・テーブル・メニュー・コマンドを実 行することもできます。詳細については, 69ページ「QuickTest コマンドの実 行」を参照してください。

Quality Center でのデータ・テーブル・ファイルの使用

Quality Center およびデータ・テーブルを使って作業する場合, [テストの設定] ダイアログ・ボックスの [リソース] タブでデータ・テーブル・ファイルを指 定する前に, データ・テーブル・ファイルを添付ファイルとして Quality Center プロジェクトに保存しておく必要があります。

新規または既存のデータ・テーブル・ファイルを Quality Center プロジェクトに 追加できます。ファイル・システムの既存のデータ・テーブルを Quality Center プロジェクトに追加すると,追加したデータ・テーブルは Quality Center プロ ジェクトに含まれていないテストによって使用されるデータ・テーブルをコ ピーしたものとなります。そのため,追加したファイルをプロジェクトに保存 すると,Quality Center データ・テーブルに対して行われた変更は、ファイル・ システムのデータ・テーブル・ファイルに反映されません。また、その逆の場 合も同様です。

Quality Center でデータ・テーブル・ファイルを使用するには、次の手順を実行します。

- 1 新しいデータ・テーブル・ファイルを追加する場合は、**.xls** という拡張子の付いた新しい Microsoft Excel ファイルをファイル・システムに作成します。
- 2 Quality Center で、データ・テーブル・ファイルを添付ファイルとしてプロジェ クトに追加します。
- **3** [テストの設定] ダイアログ・ボックスで, [**リソース**] タブをクリックします。
- 4 [その他] を選択し、参照ボタンをクリックして、データ・テーブル・ファイ ルを検索します。
- 5 テストを作成します。テストを保存すると、QuickTest によって、データ・テー ブル・ファイルが Quality Center プロジェクトに保存されます。

データベースからのデータのインポート

Microsoft Query からクエリを選択するか, SQL ステートメントを手作業で指定 することによって,データベースからデータをインポートできます。

Microsoft Query は, Microsoft Office のカスタム・インストール・オプションで インストールできます。

注: Excel ファイルのインポート([**ファイル**] > [**ファイルからインポート**]) とは異なり, データベースからデータをインポートするときに, データ・テー ブルの既存のデータは置き換えられません。インポートするデータベースに既 存のカラムと同じ名前のカラムがある場合, そのデータベースのカラムはカラ ム名の後に連番が付き, 新しいカラムとして追加されます。たとえば, デー タ・テーブルに「departures」という名前のカラムがすでに存在する場合, 同じ 名前のデータベースのカラムは「departures1」という名前でそのデータ・テー ブルに挿入されます。

データベースからデータをインポートするには、次の手順を実行します。

 データのインポート先にするデータ・テーブル・シートを右クリックし、 [シート] > [インポート] > [データベースから] を選択します。データ ベース・クエリ・ウィザードが起動します。

データベース クエリ ウィザード		×
	データベース接続方法(©): ODBC クエリの定義 「Microsoft Query でクエリを作成する(R)	
Ø	○ SQL ステートメントを手作業で指定する⑤	
QuickTest Professional データベース クエリ ウィザード	□ 最大行数(M): ■ Microsoft Query の使い方を表示する(H)	
< 戻	る(B) 次へ(N) キャンセル (ハルブ	

- 2 対象のデータベースを選択し、[次へ]をクリックします。次のオプションの いずれかを選択することができます。
 - ➤ [Microsoft Query でクエリを作成する]: Microsoft Query が起動し、新しい クエリを作成できるようになります。クエリの定義を完了すると、 QuickTest に戻ります。このオプションは、コンピュータに Microsoft Query がインストール済みである場合にのみ使用できます。
 - ➤ [SQL ステートメントを手作業で指定する]: ウィザードに [SQL ステート メントを指定します。] 画面が表示され,接続文字列と SQL ステートメント を定義できる状態になります。詳細については,手順3を参照してください。
 - ▶ [最大行数]: このチェック・ボックスを選択して、インポートするデータ ベース行の最大数を入力します。最大で 32,000 行を指定できます。
 - ➤ [Microsoft Query の使い方を表示する]: [次へ] をクリックすると, Microsoft Query を開く前に,説明画面が表示されます([Microsoft Query でクエリを作成する] を選択した場合にのみ使用できます)。
- 3 前の手順で [Microsoft Query でクエリを作成する] を選択した場合, Microsoft Query が起動します。データ・ソースを選択し、クエリを定義します。 クエリの作成の詳細については、1144ページ「Microsoft Query でのクエリの作 成」を参照してください。

前の手順で[SQL ステートメントを手作業で指定する]を選択した場合,次の 画面が表示されます。

データベース クエリ ウィザード		×
Contraction Contr	SQL ステートメントを指定します。 接続文字列: 作成で)	
< 戻	る(B) 完了(E) キャンセル ヘルブ	

接続文字列と SQL ステートメントを指定し, [完了] をクリックします。

- ▶ [接続文字列]:接続文字列を入力するか、[作成]をクリックし、[データ ソースの選択]ダイアログ・ボックスを表示します。[データ ソースの選 択]ダイアログ・ボックスで*.dsn ファイルを選択するか、新しい*.dsn ファイルを作成すると、接続文字列がこのボックスに挿入されます。
- ▶ [SQL ステートメント]: SQL ステートメントを入力します。
- 4 数秒間でデータベース・クエリがキャプチャされ、QuickTest ウィンドウに戻ります。データベース・クエリの結果データは、データ・テーブルに表示されます。

Microsoft Query でのクエリの作成

Microsoft Query を使用して、データ・ソースを選択し、データ・ソースを対象 とするクエリを定義することができます。サポートされている Microsoft Query のバージョンの詳細については、『HP QuickTest Professional 最初にお読みくだ さい』を参照してください。

Microsoft Query でデータ・ソースを選択し、クエリを定義するには、次の手順を実行します。

- 1 データベースからのデータのインポート処理中に Microsoft Query が起動した ら、新しいデータ・ソースまたは既存のデータ・ソースを選択します。
- 2 クエリを定義します。
- **3** Query ウィザードの [完了] 画面で, [**終了し、QuickTest へ戻る**] を選択し, [**完了**] をクリックして Microsoft Query を終了します。

あるいは、[Microsoft Query でデータの表示またはクエリの編集を行う] を クリックし、[完了] を クリックします。データを表示または編集したら、[ファ イル] > [終了し、QuickTest へ戻る] を選択し、Microsoft Query を閉じて QuickTest に戻ります。

Microsoft Query での作業の詳細については, Microsoft Query のマニュアルを参照してください。

データ・テーブルでの数式の使用

データ・テーブルでは、Microsoft Excel の数式を使用できます。したがって、 実行セッション中に、実行コンテキストに応じたデータを作成できます。ま た、チェックポイントの一部として数式を使用し、実行中に作成された(動的 に生成された)ページのオブジェクト、または Webページ内やアプリケーショ ンのその他の可変オブジェクトに、任意のコンテキストに応じた期待値が設定 されているかどうかを検査することもできます。

データ・テーブルで数式を使用し、値を比較する場合(通常はチェックポイン トで)、比較する値は同じタイプ(整数、文字列など)である必要があります。 異なる関数を使用して、アプリケーションの異なる場所から値を抽出すると、 値のタイプが異なる場合があります。これらの値は、画面上では同じものに見 えるかもしれませんが、たとえば数値の8.2と文字列の「8.2」は等しくないた め、比較は失敗します。

注: データ・テーブルでは,複雑な式またはネストされた式の使用はサポート されていません。

TEXT および VALUE 関数を使用して, 値のタイプを別のタイプに変換できます。

- ➤ TEXT(value, format) は数値と同等のテキストを指定した形式で返します。たとえば、数式=TEXT(8.2, "0.00") は "8.20" となります。
- ➤ VALUE(string) は文字列と同等の数値を返します。たとえば、=VALUE("\$8.20") は 8.20 となります。

ワークシートの関数の使用法については、Microsoft Excel のマニュアルを参照 してください。

パラメータ化データを作成する数式の使用

パラメータのカラムのセルに、固定値の代わりに数式を入力できます。

たとえば、今日以降の日付値を必要とする WebEdit オブジェクトに対する値を パラメータ化する必要があるとします。[Date] カラムのセルを日付の書式に 設定し、最初の反復用に今日の日付を値に設定するために、Excel の数式 =NOW() を最初の行に入力します。 これによって、次のように残りの行に別の数式を使用して、上の行の日付に1 日足した日付を入力できます。この数式を使用することで、テストをいつ実行 しても、日付は常に有効になります。



パラメータの使用方法の詳細については,第22章「値のパラメータ化」を参照してください。

チェックポイントでの数式の使用

チェックポイントで数式を使用し、実行時に作成された(動的に生成された) ページのオブジェクト、またはWebページ内やアプリケーションの別の可変オ ブジェクトに、任意のコンテキストに応じた期待値が設定されていることを確 認できます。たとえば、ショッピング・カートのWebサイトに、合計金額が表 示されるとします。この場合、表示された合計金額を対象とするテキスト・ チェックポイントを作成してから、データ・テーブル数式を使用して、各反復 で選択された商品のそれぞれの価格を合計し、サイトの合計金額として比較し て正しく計算されているかどうかを検査できます。

チェックポイントでデータ・テーブル数式を使う場合,QuickTest はデータ・ テーブルに2つのカラムを作成します。最初のカラムには、標準のチェックポ イント数式が含まれています。2番目のカラムには、検査対象の値が出力パラ メータの形式で格納されています。数式の結果はBooleanで,TRUE または FALSEです。

A1		=\$B1="337"		
	Total_	Price	Total_	Price_out
1	TRI	JE _	337	
2				

1146

テストの実行中にチェックポイントのカラムで FALSE の結果が出ると,テストが失敗します。

チェックポイントの追加が終了したら,最初のカラムで標準の数式を変更し, 必要な検査を実行できます。

チェックポイントで数式を使うには、次の手順を実行します。

- 1 第15章「チェックポイントについて」の説明に従って、チェックポイントを作成するオブジェクトまたはテキストを選択し、[チェックポイントの挿入]ダイアログ・ボックスを開きます。
- 2 [値の設定] 領域で, [パラメータ] をクリックします。
- 3 [パラメータオプション] ボタンをクリックします。[パラメータオプション]
 ダイアログ・ボックスが表示されます。

パラメータ オブション	×
パラメータのタイプ	
DataTable	
名前(N): userName_name 💽	
データテーブル内の場所―――	
⑦ グローバル シート(G)	
○ 現在のアクション シート (ローカル)(山)	
□ 正規表現(日)	
□ データテーブル数式(E)	
OK キャンセル ヘルプ	

- **4** パラメータ・タイプとして [**Data Table**] を選択し, [**名前**] ボックス・リストからパラメータを選択するか,新しい名前を入力します。
 - ▶ 既存のパラメータを使用するには、リストからパラメータを選択します。
 - ▶ 新しいパラメータを作成するには、標準設定のパラメータ名を使用するか、 わかりやすいパラメータ名を入力します。
- 5 [**データ テーブル数式**] チェック・ボックスを選択して, [**OK**] をクリックし, [パラメータ オプション] ダイアログ・ボックスを閉じます。

注:[**正規表現**]が選択されている場合は,[**データ テーブル数式**]を選択できません。

- 6 第 15 章「チェックポイントについて」の説明に従って、チェックポイントを設 定します。
- 7 [OK] をクリックします。テーブルに2つのカラムが追加され,テストには チェックポイント・ステップが挿入されます。
- 8 最初の(式の)カラムにある値を強調表示して数式を確認し,必要に応じて数 式を変更します。
- 9 複数の反復を実行する場合、テストまたはアクションの反復ごとに、数式のカラムの続く行に適切な数式を追加します。

ヒント:パスワードを暗号化し、その結果生成された文字列をメソッド引数ま たはデータ・テーブル・パラメータ値として使用できます。詳細については、 418ページ「メソッド引数およびデータ・テーブル・セルへの暗号化されたパ スワードの挿入」を参照してください。データ・テーブル・メニューの[暗号 化]オプションを使用して、データ・テーブル列の文字列を暗号化することも できます。詳細については、1139ページ「[データ]メニュー」を参照してく ださい。
データ・テーブルのスクリプト・メソッドの使用

QuickTest には、実行時データ・テーブルに関する情報を取得したり、実行時 データ・テーブルのセルに値を設定したりするためのデータ・テーブル・メ ソッドがいくつか用意されています。これらのステートメントは[エキスパー トビュー]の中で手作業で入力します。エキスパート・ビューでの作業の詳細 については、第26章「エキスパート・ビューおよび関数ライブラリ・ウィン ドウを使用した作業」を参照してください。

プログラミングの観点から見ると、データ・テーブルは、DataTable、Sheet (シート)、Parameter (カラム)の3種類のオブジェクトで構成されています。 各オブジェクトには、値の取得や設定に使用できる、いくつかのメソッドとプ ロパティがあります。

データ・テーブル・メソッドの詳細については,『**HP** QuickTest Professional **Object Model Reference**』(英語版)を参照してください。

第38章・データ・テーブルを使った作業



プロセス・ガイダンスを使った作業

プロセス・ガイダンスは、特定のプロセスを最も効果的に実行する方法に関す る手順と詳細を提供するツールです。プロセス・ガイダンスを使用すると、新 しいプロセスについて学習したり、すでに知っているプロセスを実行するため の望ましい方法について学習したりできます。このため、プロセス・ガイダン スは新しいユーザおよび経験を積んだユーザの両方にとって役に立ちます。

プロセスとは、アクティビティ、つまりサブプロセスの集合のことです。各プ ロセスにおいて、そのプロセスで必要なアクティビティを段階的に案内します。 各プロセスのアクティビティを順にたどり、各アクティビティで説明している タスクを実行することで、特定のプロセスの実行方法に次第に慣れてきます。

QuickTest は、複数のプロセスで構成される組み込みのパッケージを備えていま す。これらのプロセスでは、テストの計画と作成など、最も一般的な QuickTest タスクを実行する方法に関する基本的な情報およびヒントが提示されます。

また、組織独自のカスタム・プロセスを作成し、その組織に関係のある特定の 要件およびベスト・プラクティスについてユーザを案内することもできます。

本章では、次の項目について説明します。

- ▶ [プロセスガイダンス]表示枠(1152ページ)
- ▶ プロセス・ガイダンスの開始(1154ページ)
- ▶ 利用できるプロセスのリストの管理(1155ページ)

[プロセス ガイダンス]表示枠

83

QuickTest では、プロセス・ガイダンスは、[プロセス・ガイダンス・アクティ ビティ] 表示枠と [プロセス・ガイダンスの説明] 表示枠という 2 つの表示枠 に表示されます。

これらの表示枠の表示と非表示を切り替えるには, [表示] > [プロセス ガイ ダンス] を選択するか, または [プロセス ガイダンス ウィンドウ] トグル・ ボタンをクリックします。

マイルビ 編集(E) 表示(Y) 構入(D) オートメーション(A) リソース(B) デバッグ(D) ソール(D) ウインドウ(W) ヘルブ(D) ● & ● 記録 ▶ 実行・ ■ (P止 日) ま、ご ● : ①・ ジ・ ジ な シ 4): ○ ○ ○ ▲ ▲ 70世23.ガイダンスのアウティビティ ・ 4 × ★ -ワード駆動のテスト ● 記録 ▶ 実行・ ■ (P止 日) ま、ご ● : ①・ ジ・ ジ な シ 4): ○ ○ ○ ▲ ▲ 70世23.ガイダンスのアウティビティ ・ 4 × ★ -ワード駆動のテスト ● (P」) 「 「 「 」 」 」 ② ● : ②・ ジ・ ジ な シ 4): ○ ○ ○ ▲ ▲ 70世23.ガイダンスのアウティビティ ・ 4 × ★ -ワード駆動のテスト ● (P」) 「 「 」 小 ジョンの分析 アクションの分析 アクションの分析 アクションの分析 アクションの時間 オブジェクト レオジドリの設定 オブジェクト レオジドリの設定 オブジェクト レオジドリの設定 オブジェクト リオジドリカンドション オブジョクト レオジドリッビデジャ ★ -ワード駆動のテストのブロセン ガイダンスのド ビクヘンは交 ま。 アーレッビアション ク分析 アクションの時間 オグジョント レオジドリーボランド ビグ - トレズ 学習 (オブジュクト リオジドリカ) ビグ - トレズ 学習 (オブジョクト リオジドリ マオージャ ★ 1 ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ●	図QuickTest Professional - [テスト]				
新規作成・ ○ 間、・ □ 部 ○ × □ へ □ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○	·	.(I) オートメーション(<u>A</u>) リソース(<u>R</u>) デバッグ(<u>D</u>) ツール(I) ウィンドウ(<u>W</u>) へ/	ルプ(日)	- #×
● 記録 実行・ 停止 田本 (日本) (日	: 🎦 新規作成 🔹 🗁 開(🔹 🗐 🦪 🥭		124		
2 Dt2次 ガイダンスのアウティビティ ● 4 × フロセス ガイダンスのアウティビティ ● 4 × キーワード駆動のテスト ● 東高 ※パー ● 東西 ※パー ● 東西 <t< td=""><td></td><td></td><td>0 0 00 2</td><td></td><td></td></t<>			0 0 00 2		
キーワード駆動のテスト 第一・キーワード駆動のテスト (2) (3) (4) (4) (5) (5) (6) (7) (7) (7)	プロセス ガイダンスのアクティビティ ・ A ×		プロセス ガイダンスの話	488	- a x
キーワード駆動のテスト 第二 第二 <td></td> <td></td> <td>5 a c// 3 - 5 5 / 6 6</td> <td></td> <td>▲</td>			5 a c// 3 - 5 5 / 6 6		▲
	キーワード駆動のテスト	Action1 「一 戻る」」	キーワード駆動	のテストの紹介	
・ キーワード駆動のテスト QuickTest Professional テスト ・ キーワード駆動のテスト QuickTest Professional テスト ・ テストの二・ズの決定 テスト環境の定義 対象のアカケーションの分析 アウションの計画 ・ オンジェクトリード ・ オンジェクト ・ オンジェクシッシ ・ オンジェクシッシ ・ オンジェクト ・ オンジェクシッシ ・ オンジェクシッシ ・ オー ・ オー ・ オー <td>紹介</td> <td>項目 操作 値</td> <td></td> <td></td> <td></td>	紹介	項目 操作 値			
QuickTest Professional テスト ストの作成を計画とテスト実装の2 段階に分ける 手法です。 デストのニーズの決定 テスト環境の定義 対象のアカリケーションの分析 アクションの計画 計画印留では、対象のアプリケーションを分析し、 テストのニーズを決定します。テストのニーズに 差次11で、必要なリソースを決定します。事 前作業を終えたら、テストの作成を閉覧をできます。 オブジェクトリメジリの設定 オブジェクトリメジリリの設定 オブジェクトリメジリリの設定 オブジェクトリオジリリメジリリ キビオートにて学習 [オブジェクトリオジリリ] ダイアログ ホッ… オブジェクトリオジリリ マネージャ メー INF チークーF ビュー 人 エキスパート ビュー オブジェクトリオジリリ キビオートにて学習 [オブジェクトリオジリリ マネージャ マストの実行経巣できます。 テストの実行経巣でしたするれるま で、テスト サイクルを繰り返します。 INF チークラーデ ビュー 人 エキスパート ビュー オブジェクトリオジリレ マクラーフル マストの実行経巣でしたするれるま で、テスト サイクルを繰り返します。 INF チークラーブル マスト INF チークラーブル マスト INF チークラーブル マスト マークラーブル マスト INF チークラーブル A B INF チークラーブル マスト INF チークラーブル ActiveScreen INF チーク、クリンクと、他の存在したり マスト INF チーク INF チーク	▶ キーワード駆動のテスト		 キーワード駆動のテ ピックへようこそ。キ 	・ストのブロセス カイ: キーワード駆動のテス	ダンスのト いた、テ
テストのニーズの決定 ・ テスト環境の定義 ・ 対象のアカウーションの分析 ・ アクションの計画 ・ オブジェクトリメラクトリの設定 ・ オブジェクトリメラクトリの設定 ・ オブジェクトレスオジョンの分析 ・ アクションの計画 ・ オブジェクトリメラクトリの設定 ・ オブジェクトレスオジョンの ・ オブジェクトリメラクトリカジリ ・ オブジェクトリメンクリカジリ ・ オブジェクトリボジトリ ・ オブジェクトリボジトリ ・ オブジェクトリボジトリ ・ オブジェクトリボジトリ ・ オブジェクトリボジトリ ・ オブジェクトリボジトリ ・ オブジェクトリボジー ・ オブジェクト ・ オブジェクト ・ オブジェクト ・ オー ・ ・ ・ 1 ・ <td>QuickTest Professional テスト</td> <td></td> <td> ストの作成を計画と 手法です。 </td> <td>:テスト実装の 2 段開</td> <td>間に分ける</td>	QuickTest Professional テスト		 ストの作成を計画と 手法です。 	:テスト実装の 2 段開	間に分ける
デスト環境の定義 対象のアウルケーションの分析 アクションの分析 アクションの計画 計画問題では、対象のアウルケーションを分析し、 テストの二・スたまでは、テストのためを開始できま。テストの、 オブジェクトリホジトリの認定 オブジェクトリホジトリの認定 オブジェクトレして学習 [オブジェクトリホジトリ]ダイアログホタ オブジェクトリホジトリ」ダイアログホタ オブジェクトリホジトリ」ダイアログホタ オブジェクトリホジトリ」ダイアログホタ オブジェクトリホジトリマキージャ 計画見留では、対象のアウルケーションを分析し、 ストの実装段階では、対象のアウルケーションを決定し、リンースを作 成します。がな、テストの作成を開始できます。 テストの実装段階では、対象のアウルケーションル 対するテストの実行結果の表示 サレーントして学習 [オブジェクトリホジトリ]ダイアログホタ オブジェクトリホジトリ」ダイアログホタ オブジェクトリホジトリマキージャ (日本ショントして学習) [オブジェクトリホジトリ」ダイアログホタ オブジェクトリホジトリマキージャ エークードビュー 人 エキスパートビュー 要したいてディングを行いたまであります。 エークテーブル マークテーブル マークテーブル マーク、アークリンクションが超いたます。 エーク・ディングでは、このプロセスガイダンストビックでは、Cohビター クロヘルレブリのセンションが超いたいます。 エーク・ディングの構造を使きます。 エーク・ディングの構造を表した。 エーク・アーク、レクマロス、たちままままます。 エーク・ディングの目前に、このトレク・ クロヘルレブリの生くがついたいを開達していための構造を使う エーク、ションパングルンクを使用してこれらの目的などうに、Cohビター クロヘルレブリのたきションが超いたいたいたいたきままままます。 エーク・ディングの構造を使用していたりの目的を発行する たたいたります。 エーク・ディングの目した、このトレク・ クロヘルレブリのたきションがあり、「HP QuicKreat Professional ユージーズ ガイド・ハレンの関連ト ビックへのリンクと、他の役立つリンクが示されていまま。 エーク・ディンの関連ト エーク・ディンの開達した。 エーク・ディンの目的を見たいたります。 エーク・ディングの目的を見たいたります。 エーク・ディングの構成した。 エーク・ディングの目のまた。 エーク・ディングの目的を見たいたります。 エーク・ディングの目のまたり、 エーク・ディングの目のまたり、 エーク・ディングのアングロンクレックのまたり、 エーク・ディングロンクレック・ エーク・ディングロンクレックのまたり、 エーク・ エーク・ エーク・ エーク・ エーク・ エーク・ エーク・ エーク・ </td <td>テストのニーズの決定</td> <td></td> <td>1724 () 0</td> <td></td> <td></td>	テストのニーズの決定		1724 () 0		
対象のアカリケーションの分析 アクションの計画 オブブェクトリオシドリの設定 オブジェクトリオシドリの設定 オブジェクトリオンドリの設定 オブジェクトリオンドリの設定 オブジェクトリオンドリの設定 オブジェクトリオンドリリントして学習 [オブジェクトリオンドリ] ダイアログ ホッ オブジェクトリオンドリ] マイージャ (1) ひょう インタの定義 「日秋シーイントクストロン」 (1) ひょう インタの定義 日日 (1) ひょう インタークンレクション/ (1) ひょう インタのに (1) ひょう インタのに (1) ひょう インタークン) (1) ひょう インタークン (1) ひょう インタークン (1) ひょう インタークン) (1) ひょう インタークン (1) レクローバレク、 (1) レクローバレク、 (1) レクローバレク、 (1) レクローバレク、 (1) レクローバレク、 (1) レクローバレク、 (1) レクローバレクレーズに、 (1) レクローバレクレクローバービー (1) レクローバレクレクローバービー (1) レクローバレクレクローバービー (1) レクローバービー (1) レクローバレクレクローバービー (1) レクローバレクローバービー (1) レクローバービー (1) レクローバレクローバービー (1) レクローバレクレクローバービー (1) レクローバレクローバービー (1) レクローバレクローバービー (1) レクローバービー (1) レクローバービー (1) レクローバービー (1) レクローバービー (1) レクローバービー (1) レクローバービー (1) レクローバー (1) レクローバービー (1) レクローバー (1) レクロー (1) レクローバー (1) レクローバー (1) レクローバー (1) レクローバー (1) レクロー (1) レクロー (1) レクローバー (1) レクロー (1) レクロー (1	テスト環境の定義		計画段階では、対象 テストのニーズを決	&のアブリケーションを 定します。 テストのコ	を分析し、 ニーズに
アクションの計画 アクションの計画 アクションの計画 アクションの計画 アクションの計画 オブジェクトリホジトリカジン オブジュクトリホジト アイクションの計画 アクションの計画 アクションの計画 オブジュクトレスデジ オブジュクトレスデジ アクションの計画 アクションの計画 アクションの計画 オブジュクトリホジト アインジェクト アクション アクション アクション アクラ・ブル マー マー アクション アクション アクラ・ブル マー マー アクション アクション アクラ・ブル マー マー アクション アクション 「日秋ライブラリの定業 A B C アク アク 「日秋ライブラリの定業 A B C アク アク アク 「日秋ライブラリの定業 A B C アク アク アク アク アク 「日秋ライブラリッコンスト アク ア	対象のアプリケーションの分析		基づいて、必要ない	ソースを決定し、リソ・	ースを作
	アクションの計画		前作業を終えたら、	テストの作成を開始1	できます。
オブジェクトリボジトリ オブジェクトリボジトリ ナビゲートして学習 「オブジェクトリボジトリ ダイアロヴ ボッ… オブジェクトリボジトリ ダイアロヴ ボッ… オブジェクトリボジトリ マネージャ 「オブシェクトリボジトリ マネージャ 「オブシェクトリボジトリ マネージャ 「オブシェクトリボジトリ マネージャ 「オブシェクトリボジトリ マネージャ 「オブジェクトリボジトリ マネージャ 「オブジェクトリボジャリ マネージャ 「オブジェクトリボジトリ マネージャ 「オブジェクトリボジトリ マネージャ 「オブジェクトリボジトリ マネージャ 「オブジェクトリ オージッ 「オブジェクトリ オージャ 「オブジェクトリ オージャ 「オブジェクトリ オージャ 「オージョー オージ オージャ 「オージョー オージ オード・トレージの 関連ト ビック ハのリンクな、他の役立つ リンクが示されてい ます。 「オージョー オージー オージー オージー オージー 「オージー オージー オージー 「オージー オージー オージー 「オージー オージー 「オージー	オブジェクトリポジトリの設定		テストの実装段階で	9は、対象のアプリケ、	-2417
the set of the	オブジェクトとオブジェクト リポジトリ		対するテストの実行	で、テストの実行結果の	の表示、必
(オブジェクトリボジトリノオイジャク A1 (オブジェクトリボジトリオネージャ A1 (東京・フトリイクルを築い返します。 (ローク)	ナビゲートして学習	デーカテーブル. ・ A ×	要に応じていいの	プリケーションが期待。	どおりに
	【オフンエクトリボントリ】ダイアロク ホッ… オブジェクト 日ボジトロ マネージャ		実行していることかう で、テスト サイクル?	テスト結果によって示 を繰り返します。	iðnos
田政 A B C C C Dフレゼス カナ49 2人 ドビック ビは、Quick test Frofessional を使用してこわらる各類酸を実行す S T C C S X C C S C C Dフレゼス カナ49 2人 ドビック じは、Quick test Frofessional を使用してこわらる各類酸を実行す S S C Dアレマスカナ49 C C Dアレマス Dアレマス D T C C Dアレマス D T C C D C D C D C D C D C D C D C D C D C D C D C D C D C D C D C D C D C D C C D D C D C D C D C D C D C D C D C D C D	333131.39(313.44.2)			" The second	
	関数ライブラリの定義		このプロセス カイタ Professional を使用	シストビックでは、Q 用してこれらの各段階	uickTest 皆を実行す
		2	る方法を説明します。 クのヘルプロのセク	。各トビックには、「こ パノョンがあり」「HP O	このトピッ DuickTest
[1] 「1] 「1] 「1] 「1] 「1] 「1] 「1] 「1] 「1] 「	🗲 戻る 🛛 次ヘ 🌗	3	Professional 그-1	ザーズ ガイドコヘルラ	の関連ト
			- ニックへのウンシン() ます。	1回の1支立 シワンフルッ	ROTICI
		Activescreen			▼ (書字フ

[プロセス ガイダンス アクティビティ] 表示枠

[プロセスガイダンスアクティビティ]表示枠(左側に表示)には,選択した プロセスの一部であるアクティビティが表示されます。アクティビティは多く の場合グループ化されており,関心のあるサブプロセスに直接移動できます。 前出の例では,キーワード駆動のテスト・プロセスにおけるグループとアク ティビティの一部を示しています。たとえば,「テストのニーズの決定」グ ループには,「テスト環境の定義」,「対象のアプリケーションの分析」,および 「アクションの計画」という3つのアクティビティが含まれています。 [プロセス ガイダンス アクティビティ]表示枠では,次の操作を実行できます。

- ➤ アクティビティをクリックして、そのアクティビティに対応するトピックを [プロセス ガイダンスの説明]表示枠で開く。
- ▶ どのアクティビティが [プロセスガイダンスの詳細] 表示枠に表示されている かを確認する(現在選択されているアクティビティが矢印で示される)。
- ▶ [戻る] ボタンと [次へ] ボタンを使用してアクティビティ間を上下に移動し、 [プロセス ガイダンスの説明] 表示枠に前または次のアクティビティのトピッ クを表示する。
- ▶ 上矢印または下矢印上にカーソルを置いてアクティビティのリストをスクロールする(上矢印は, [プロセスガイダンスアクティビティ] タイトル・バーのすぐ下に,下矢印は, [戻る] ボタンおよび [次へ] ボタンのすぐ上にある)。

[プロセス ガイダンスの説明] 表示枠

[プロセスガイダンスの説明]表示枠(前出の例では右側に表示)には,選択 したアクティビティに関するトピック(説明)が表示されます。

詳細ごとに特定のアクティビティが紹介され,そのアクティビティの実行方法 に関する情報を見つけられる場所へのリンクが提示されます。さらに,多くの 詳細には,ダイアログ・ボックスやほかの関連する機能を開く対話型のリンク が含まれており,説明されている機能に直接アクセスできます。

プロセス・ガイダンスの開始

プロセスは, [スタートページ], [**オートメーション**] メニュー, または [プ ロセス ガイダンス アクティビティ] 表示枠から開けます。

スタート・ページ

スタート・ページの [プロセス ガイダンス リスト] には、利用可能なプロセ スがすべて表示されます。一部のプロセスは、特定の条件のもとでのみ利用で きる場合があります。たとえば、ビジネス・コンポーネント・プロセスのガイ ダンスを利用できるのは、ビジネス・プロセス・テストをサポートする Quality Center プロジェクトに接続されている場合にかぎられます。さらに、一部のプ ロセスは、特定のアドインが読み込まれている場合にのみ表示されます。たと えば、[SAP Gui for Windows のテスト] 組み込みプロセスが表示されるのは、 SAP アドインが読み込まれている場合にかぎられます。

リストから QuickTest プロセスを選択すると、関連する種類のドキュメントが 開きます。たとえば、テスト・ドキュメントを開いた状態で、[**アプリケー ション領域**] プロセスを選択した場合、新しいアプリケーション領域が開き、 選択したプロセスをたどりながらアプリケーション領域を操作できます(ビジ ネス・プロセス・テストがサポートされている Quality Center プロジェクトに接 続されている場合)。

スタート・ページから特定のプロセスを開くには、次の手順を実行します。

- QuickTest で、「スタートページ」タブをクリックしてスタート・ページを表示 します([スタートページ] タブが表示されていない場合は、「表示] > [ス タートページ] を選択してスタート・ページを開きます)。
- 2 [プロセス ガイダンス リスト]で、開くプロセスのリンクをクリックします。 アクティビティのリストが [プロセス ガイダンス アクティビティ]表示枠に 表示され、リストの最初のアクティビティの詳細が [プロセス ガイダンスの説明]表示枠に表示されます。

ヒント: [プロセス ガイダンス リスト] が空の場合は, [ファイル] > [プロ セス ガイダンス管理] を選択し, [プロセス ガイダンス管理] ダイアログ・ ボックスで少なくとも1つのプロセスを選択します。

[オートメーション] メニュー・コマンド

現在開いているドキュメントの種類,または読み込まれている QuickTest アド インで利用できるプロセスを開くには,[**オートメーション**] > [**プロセス ガ イダンス リスト**]を選択し,リストからプロセスを選択します。

[プロセス ガイダンス リスト] が空の場合は,[ファイル] > [プロセス ガイ ダンス管理] を選択し,[プロセス ガイダンス管理] ダイアログ・ボックスで 少なくとも1つのプロセスを選択します。次に,現在のドキュメントを再び開 くか,または新しいドキュメントを開いて[プロセス ガイダンス リスト] メ ニューで利用できるプロセスのリストを更新します

(現在のテスト・ドキュメントまたは読み込まれている QuickTest アドインに関 連のないプロセスを開くには、スタート・ページの [プロセス ガイダンス リ スト] からプロセスを開く必要があります)。

現在開いているテスト・ドキュメントに利用できるプロセスが複数ある場合 は、プロセスのタイトルのドロップ・ダウン・リストで必要なプロセスを選択 することによって、これらのプロセスを行き来できます。

プロセスガイダンスのアクティビティ 🝷 📮 🗴	<
「キーワード駆動のテスト・]
✔ キーワード駆動のテスト	
SAP Gui for Windows のテスト	

利用できるプロセスのリストの管理

プロセスはプロセス・ガイダンス・パッケージに格納されています。QuickTest は、複数のプロセスを含んだ組み込みパッケージを備えています。このパッ ケージは、[プロセスガイダンス管理]ダイアログ・ボックスに標準で表示さ れます。

各組織は,組織,チーム,組織での役割などに固有のプロセスが含まれている 追加のパッケージを提供できます。 QuickTest で利用できるプロセスの一覧を管理するには、[プロセス ガイダンス 管理] ダイアログ・ボックスを使用します。

プロセス ガイダンス管理	x
必要に応じてパッケージを追加または にプロセス ガイダンス リストに表示させ まれているパッケージを選択します。	削除します。 次 るプロセスが含
☑ QuickTest パッケージ (組み込み	追加(A) 削除(<u>R</u>)
<u>ا</u>	選択されたパッ ケージを削除で きません。
プロセス リスト: キーワード駆動のテスト アプリケーション領域 SAP Gui for Windows のテスト ビジネス コンポーネント	
<u>閉じる(O)</u>	

パッケージの利用と除外

QuickTest で利用できるパッケージ群に含まれているパッケージを利用するか除 外するかを選択できます。

パッケージを含めることを選択すると,QuickTest によって,そのパッケージ内 のすべてのプロセスが,スタート・ページの[プロセスガイダンスリスト] に追加されます(現在読み込まれていないQuickTest アドインのプロセスを除 きます)。現在開いているドキュメントの種類および現在読み込まれている QuickTest アドインで利用できるプロセスは,[オートメーション]メニューの [プロセスガイダンスリスト]にも追加され,現在のドキュメントを閉じてか ら再び開くか同じ種類の新しいドキュメントを開くことによってリストを更新 した後に開くことができます。

パッケージ内の個々のプロセスを追加したり除外したりすることはできません。

QuickTest で利用できるパッケージ群を対象にパッケージを追加したり除外したりするには、次の手順を実行します。

- 1 [**ファイル**] > [**プロセス ガイダンス管理**] を選択します。[プロセス ガイダ ンス管理] ダイアログ・ボックスが開きます。
- 含めたいプロセスのパッケージの横にあるチェック・ボックスを選択するか、除 外したいプロセスのパッケージの横にあるチェック・ボックスをクリアします。

3 [閉じる] をクリックします。QuickTest によって,該当するプロセスが [プロ セス ガイダンス リスト] に追加または削除されます。

プロセス・ガイダンス・パッケージの追加

組織に独自のプロセスがある場合は、それらをスタート・ページの [プロセス ガイダンスリスト] に追加できます。これを行うには、該当するパッケージを [プロセスガイダンス管理] ダイアログ・ボックスに追加し、選択することで 表示します。

リストにパッケージを追加するには、次の手順を実行します。

- 1 [**ファイル**] > [**プロセス ガイダンス管理**] を選択します。[プロセス ガイダ ンス管理] ダイアログ・ボックスが開きます。
- プロセスガイダンス管理]ダイアログ・ボックスで、[追加]をクリックします。[ファイルを開く]ダイアログ・ボックスが開きます。
- 3 プロセス・ガイダンス・パッケージ・ファイルを参照して [**開く**] をクリック します。利用可能パッケージのリストにパッケージが追加されます。

第 IX 部

QuickTest の設定



グローバル・テスト・オプションの設定

グローバル・テスト・オプションを設定することで、QuickTest とテストの連携の方法を制御できます。

本章では、次の項目について説明します。

- ▶ グローバル・テスト・オプションの設定について(1161ページ)
- ▶ [オプション] ダイアログ・ボックスの使用(1162ページ)
- ▶ テストの一般オプションの設定(1164ページ)
- ▶ テストのフォルダ・オプションの設定(1169ページ)
- ➤ ActiveScreen オプションの設定(1172ページ)
- ▶ テストの実行オプションの設定(1181ページ)

グローバル・テスト・オプションの設定について

グローバル・テスト・オプションでは、テストの記録および実行方法のほか、 QuickTest の全般的な表示動作を設定します。たとえば、QuickTest の起動時に スタート・ページを表示しないようにしたり、QuickTest がテストの実行時に使 用するタイミングに関する設定を行ったりできます。設定した値は、すべての テストおよび以降のテスト・セッションで有効になります。グローバル・テス ト・オプションの設定は、[オプション] ダイアログ・ボックスを使用して (1162ページで説明)、またはエキスパート・ビューでステートメントを挿入す ることによって行います。

また,QuickTest で現在開いているテストだけで有効なテスト・オプションを設 定することもできます。詳細については,第41章「個別のテストのオプショ ン設定」を参照してください。 2

[オプション] ダイアログ・ボックスの使用

[オプション] ダイアログ・ボックスを使用してグローバル・テスト・オプ ションを変更できます。設定した値は、後続のすべての QuickTest セッション で有効になります。

グローバル・テスト・オプションを設定するには、次の手順を実行します。

 [ツール] > [オプション] を選択するか, [オプション] ツールバー・ボタン をクリックします。[オプション] ダイアログ・ボックスが開きます。このダ イアログ・ボックスは、内容に応じていくつかのタブに分かれています。

オプション
一般 フォルダ ActiveScreen 実行 Windows アプリケーション Web
▼ 起動時にアドインマネージャを表示する(D)
☑ 起動時にスタート ページを表示する①
▼ 起動時にソフトウェアの更新を確認する(C)
□ 記録時の仮想オブジェクト認識を無効にする(V)
▼ テストオブジェクトの名称変更時にテストとコンポーネントのステップを自動的に更新する(U)
□ 記録後、"With" ステートメントを自動生成する(G)
内部で "With" ステートメントを生成するオブジェクト数(E): 2 📑
ウィンドウをポイントしてから① 5 💼 × 0.1 秒経過後にアクティブ化する
この順番でテキスト認識メカニズムを使用する(2): 最初に Windows APL、次に OCR 💌
レイアウトの復元(E) ウィンドウを標準サイズと位置に戻します。
スクリプトの生成(P) グローバル テスティング オブションを使ってオートメーション スクリプトを生成します。
あまた (の)

- 2 使用するタブを選択し、必要に応じてオプションを設定します。各タブで使用 できるオプションの詳細については、次の表を参照してください。
- 3 変更を適用してダイアログ・ボックスを開いたままにしておくには [適用] を クリックします。変更を適用してダイアログ・ボックスを閉じるには [OK] をクリックします。

[オプション] ダイアログ・ボックスには,次のタブがあります。

タブ	内容
[一般]	テストの一般設定のオプション。詳細について は,1164 ページ「テストの一般オプションの設 定」を参照してください。
[フォルダ]	QuickTest が,ダイアログ・ボックスおよびス テートメントの中で相対パスを使って指定されて いるテスト,アクション,またはファイルを検索 するフォルダ(検索パス)を入力するオプショ ン。詳細については,1169ページ「テストのフォ ルダ・オプションの設定」を参照してください。
[ActiveScreen]	記録中に QuickTest が ActiveScreen に保存,表示 する情報を設定するオプション。詳細について は,1172ページ「ActiveScreen オプションの設 定」を参照してください。
[実行]	テストを実行するためのオプション。詳細につい ては,1181 ページ「テストの実行オプションの 設定」を参照してください。
[Windows アプリケーション]	Windows アプリケーションを対象とした QuickTest によるテストの記録および実行の方法 を設定するためのオプション。詳細については, 『HP QuickTest Professional アドイン・ガイド』 で Windows ベースのアプリケーションのテスト に関する項を参照してください。

[オプション] ダイアログ・ボックスには、その時点で読み込まれているアド インに応じて対応するタブが表示される場合があります。詳細については、 『HP QuickTest Professional アドイン・ガイド』の関連する項を参照してくだ さい。

テストの一般オプションの設定

[一般] タブのオプションでは, QuickTest の全般的な表示動作およびその他の 一般的なテスト・オプションを指定します。

オプション <u>×</u>
→般 フォルダ ActiveScreen 実行 Windows アプリケーション Web
✓ 起動時にアドインマネージャを表示する(D)
▼ 起動時にスタート ページを表示する(!)
✓ 起動時にソフトウェアの更新を確認する(C)
□ 記録時の仮想オブジェクト認識を無効にする(V)
▼ テストオブジェクトの名称変更時にテストとコンポーネントのステップを自動的に更新する(U)
□ 記録後、"With" ステートメントを自動生成する(G)
内部で "With" ステートメントを生成するオブジェクト数(E): 2 📑
ウィンドウをポイントしてから① 5 🛨 × 0.1 秒経過後にアクティブ化する
この順番でテキスト認識メカニズムを使用する(X): 最初に Windows APL、次に OCR 👤
レイアウトの復元(<u>R</u>) ウィンドウを標準サイズと位置に戻します。
スクリプトの生成(P) グローバル テスティング オプションを使ってオートメーション スクリプトを生成します。
OK キャンセル 適用(A) ヘルプ

[一般] タブには、次のオプションがあります。

オプション	詳細
[起動時にアドイン マネー ジャを表示する]	QuickTest を起動したときに、アドイン・マネージャ を表示するかどうかを指定します。アドイン・マ ネージャの使用方法の詳細については、 『HP QuickTest Professional アドイン・ガイド』で QuickTest アドインの読み込みに関する項を参照して ください。
[起動時にスタート ページを 表示する]	QuickTest を起動したときに,スタート・ページを表示するかどうかを指定します。
[起動時にソフトウェアの更 新を確認する]	ソフトウェア・アップデートの有無を起動のたびに 自動的に調べるように QuickTest を設定します。詳細 については、43ページ「QuickTest ソフトウェアの更 新」を参照してください。
[起動時の仮想オブジェクト 認識を無効にする]	記録時に、仮想オブジェクト・マネージャに格納さ れている定義済みの仮想オブジェクトを認識するか どうかを指定します。詳細については、第43章「仮 想オブジェクトの学習」を参照してください。
[テスト オブジェクトの名称 変更時にテストとコンポー ネントのステップを自動的 に更新する]	ローカルまたは共有オブジェクト・リポジトリ内の テスト・オブジェクトの名前を変更した場合,テス トおよびコンポーネントのステップを自動的に更新 するかどうかを指定します。詳細については,147 ページ「テスト・オブジェクトの名前の変更」を参 照してください。
[記録後, "With" ステートメ ントを自動生成する]	QuickTest が記録時に自動的に With ステートメントを 生成するよう指示します。詳細については, 797 ペー ジ「テストに対する「With」ステートメントの生成」 を参照してください。
[内部で "With" ステートメ ントを生成するオブジェク ト数]	With ステートメントを適用する対象となるオブジェ クトが最低限連続する数を示します。この設定は, 記録後に QuickTest が自動的に With ステートメント を生成する場合,および既存のアクションのために With ステートメントを生成するよう選択した場合に 使います。標準設定=2詳細については,797ページ 「テストに対する「With」ステートメントの生成」を 参照してください。

オプション	詳細
[ウィンドウをポイントして から X x 0.1 秒経過後にアク ティブ化する]	QuickTest が, (オブジェクト・スパイ, チェックポイ ント, ステップ・ジェネレータ, 回復シナリオ・ ウィザードなどのために) アプリケーション内のオ ブジェクトを指すのに指差しマークを使っていると き, アプリケーション・ウィンドウをアクティブに するまで待機する時間を 10 分の 1 秒単位で指定しま す。標準設定 = 5

オプション	詳細
[この順番でテキスト認識メ カニズムを使用する]	テキスト/テキスト範囲チェックポイントまたは出力 値ステップのテキストをキャプチャするときに, QuickTest によって使用されるテキスト認識メカニズ ムを指定します。
	取り得る値:
	最初にWindows API,次にOCR:(標準)最初に Windows API ベースのメカニズムを使用して,オブ ジェクトからテキストを直接取得するように QuickTest を設定します。テキストを取得できない場 合は(テキストが画像の一部の場合など),QuickTest は,OCR(光学文字認識)メカニズムを使用してテ キストの取得を試みます(対象が日本語,中国語, 韓国語の場合に強くお勧めします)。
	最初に OCR、次に Windows API :最初に OCR メカ ニズムを使用してオブジェクトからテキストを取得 するように QuickTest を設定します。テキストを取得 できなかった場合,QuickTest は Windows API ベース のメカニズムを使用してオブジェクトからテキスト を取得します。
	Windows API のみ使用する:Windows API ベースの メカニズムのみを使用して,オブジェクトからテキ ストを取得するように QuickTest を設定します(OCR メカニズムは使用しません)。
	OCR のみ使用する: OCR メカニズムのみを使用し て、オブジェクトからテキストを取得するように QuickTest を設定します (Windows API ベースのメカ ニズムは使用しません) (Windows Vista で作業すると きは必須です)。
	Windows ベース環境におけるテキスト認識のサポー トの詳細については, 『HP QuickTest Professional 最 初にお読みください』を参照してください。

オプション	詳細
[レイアウトの復元]	表示枠およびツールバーが標準のサイズで標準の位 置に表示されるように,QuickTest ウィンドウの配置 を復元します。
	注: QuickTest は,動作モード(表示/編集,記録, および実行)ごとに最新のウィンドウ・レイアウト を復元します。詳細については,1168ページ 「QuickTest ウィンドウ・レイアウトのカスタマイズ」 を参照してください。
[スクリプトの生成]	現在のグローバル・テスト・オプションを含んだ自動化スクリプトを生成します。詳細については、 1311 ページ「QuickTest 操作のオートメーション」、 または『QuickTest Automation Reference』(英語版) ([ヘルプ] > [QuickTest Professional ヘルプ] > [QuickTest Advanced References] > [QuickTest Automation])を参照してください。

QuickTest ウィンドウ・レイアウトのカスタマイズ

QuickTest は、複数の異なるモード(表示/編集,記録,および実行)で動作し ます。モードの機能に対応するようにQuickTestのレイアウトを変更できます。 たとえば、記録時にはQuickTestを部分的に見えるようにすると便利な場合が あります。これにより、ActiveScreenを表示せずに、テストを記録しながら、 ステップが追加されるのを見ることができます。テストの実行時には、多くの 場合QuickTestを最小化して、テスト実行中にアプリケーションを見えるよう にすると便利です。テストを表示または編集するときには、QuickTest ウィンド ウを最大化して、すべての表示枠を表示すると便利な場合があります。

QuickTest は、モードごとにメイン・ウィンドウおよびすべての表示枠のサイズ と場所を記憶します。QuickTest でいずれかのモードを開始すると、そのモード で最後に使用したレイアウトが復元されます。つまり、QuickTest のメイン・ ウィンドウと各表示枠は、現在のモードの前回のレイアウトに基づいて、最小 化、最大化、またはサイズ変更されます。

QuickTest の各モードのレイアウトを設定するには、次の手順を実行します。

- 1 新規または既存のテストを開きます。
- 2 記録セッションを開始します。
- 31ステップを記録します。

- 4 記録モードのレイアウト設定をすべて行います。
- 5 記録セッションを停止します。
- 6 テストの最初のステップの前にブレークポイントを設定します。これにより、実行セッション中にレイアウトを変更できます。ブレークポイントの設定方法の詳細については、1047ページ「ブレークポイントの設定」を参照してください。
- 7 テストを実行します。
- 8 QuickTest がブレークポイントに達したら、実行モードのレイアウト設定をすべて行います。
- 9 実行セッションを停止します。
- **10** 表示 / 編集モードのレイアウト設定をすべて行います。

これで、これらすべてのモードのレイアウトを設定できました。QuickTest でこ れらのモードのいずれかが開始されるたびに、関連するレイアウトが適用され ます。

テストのフォルダ・オプションの設定

[フォルダ] タブには、ダイアログ・ボックスおよびステップの中で相対パス を使って指定されているテスト、アクション、ファイルを QuickTest が検索す るフォルダ(検索パス)を入力できます。たとえば、すべてのテストが格納さ れているフォルダをフォルダ・リストに追加するとします。後にアクションの コピーをテストに挿入する際は、挿入するアクションを含むテストの名前を [アクションのコピーを挿入] ダイアログ・ボックスに入力するだけで済みま す。QuickTest は [フォルダ] タブに指定されたフォルダの中でテストのパスを 検索します。

注:

- ▶ 標準設定では現在のテストが検索リストに表示されます。これは削除できません。
- ▶ 相対パスまたは絶対パスの詳細については、340ページ「QuickTest での相対 パスの使用」を参照してください。

QuickTest は、フォルダが検索リストに表示されている順に、指定されたテスト、アクション、またはファイルを検索します。同じファイル名が複数のフォルダに存在する場合は、

QuickTest は、最初に検出したファイルを使用します。

オプション
一般 フォルダ ActiveScreen 実行 Windows アプリケーション Web
QuickTest Professional でファイルの検索先フォルダを入力し、優先順位を付けてくださ い。 パスは絶対、相対または Quality Center サブジェクトのいずれかを指定できます。 相対パスは現在の テスト に相対します。
検索リスト: + × 1 ↓
L + Frogram Files+HF+Quick Lest Professiona# Lests
▼ リソースのパス指定に相対パスを使用する(B)
OK キャンセル 適用(品) ヘルプ

[フォルダ] タブには、次のオプションがあります。

オプション	詳細
[検索リスト]	QuickTest がテスト,アクション,またはファイルを検索す るフォルダを示します。ここでフォルダを定義しておけば, ほかのダイアログ・ボックスや呼び出しステートメントの中 でテスト,アクション,またはファイルのフルパスを指定す る必要がなくなります。リスト内の検索パスの順序に応じ て,QuickTest が特定のアクションやファイルを検索する順 序が決まります。
+	新しいフォルダを検索リストに追加します。
	 Quality Center に接続されている状態で Quality Center パス を追加するにはこのボタンをクリックします。QuickTest に [QualityCenter] が追加され, Quality Center のパスを指 定するための参照ボタンが表示されます。 Quality Center に接続されていない場合は SHIFT キーを押 しながらこのボタンをクリックします。QuickTest に [QualityCenter] が追加されます。ここでパスを入力しま す。Quality Center のパス全体を手作業で入力することも できます。入力する場合は [QualityCenter] の後にスペー
	スを追加する必要があります。たとえば、
	[QualityCenter] Subject¥ lests のようにします。 ▶ ただし, QuickTest が Quality Center のプロジェクト・フォ ルダを検索するのは,対応する Quality Center プロジェク トに接続しているときだけです。
×	選択したフォルダを検索リストから削除します。
Ť	選択したフォルダを上方向に移動します。
4	選択したフォルダを下方向に移動します。
[リソースのパス指定 に相対パスを使用す る]	リソースを保存するときに,相対パスを使用することを促す メッセージを表示できます。詳細については,340ページ 「QuickTest での相対パスの使用」を参照してください。

ヒント: テストの中で PathFinder.Locate ステートメントを使えば, [フォル ダ] タブで指定されたフォルダに基づく特定の相対パスに対応する, QuickTest で使われるフル・パスを取得できます。詳細については, 『HP QuickTest Professional Object Model Reference』(英語版)を参照してください。

ActiveScreen オプションの設定

[ActiveScreen] タブでは、テストの記録および実行時に QuickTest が ActiveScreen に保存および表示する情報を指定できます。

オプション ×
一般 フォルダ ActiveScreen 実行 Windows アプリケーション Web
-キャプチャのレベル
- - 部分的
- ·第1ステップのウィンドウまたはダイアログボックスに表示されるすべて のオブジェクトで全プロパティをキャプチャします。同じ親オブジェクトを対 - 象にしたほかのステップで記録されたオブジェクトに関連のあるプロパティ
- ダイナミックな変更があるごとに、すべての Web ページを保存します。 - ActiveScreen ファイルを圧縮形式で保存します。
ユーザ定義レベル… 標準レベル
┌─外観 (Web)
これらの設定を使って、ActiveScreenに表示される Web ページの外観をカスタマイズしてください。
詳細設定
OK キャンセル 適用(公) 【 へルブ

ヒント: ActiveScreen に保存される情報が多いほど,記録後にテストを編集す るのが楽になります。しかし,ActiveScreen に保存される情報が多くなると, 記録に必要な時間とディスク容量も多くなります。これは Visual Basic, ActiveX,.NET Windows Forms の各環境では特に重要です。

テストの記録後にそれらに保存されている情報量を増やすことも減らすことも できます。詳細については,を参照してください。

注: MDI (Multiple Document Interface) アプリケーションの記録時には, ActiveScreen にアクティブでない子フレームの情報は保存されません。

[ActiveScreen] タブには、次のオプションがあります。

オプション	詳細
オブション [キャプチャのレベル]	 詳細 QuickTest が ActiveScreen にデータを格納するべきオ ブジェクトを指定します。 スライダを使用して次のオプションから1つを選択します。 「完全]:ステップごとに、アプリケーションのア クティブなウィンドウ、ダイアログ・ボックス、 Webページのすべてのオブジェクトのすべてのプロパティを取得します。このレベルは、動的な変 更後のWebページを保存し、ActiveScreenファイル を圧縮した形式で保存します。 「部分的]:(標準設定)アプリケーションに対して 実行された最初のステップで、アプリケーションに対して 実行された最初のステップで、アプリケーションのア のアクティブなウィンドウ、ダイアログ・ボック ス、Webページのすべてのオブジェクトのすべて のプロパティ、および、同じウィンドウにおいて 以降のステップで記録されるオブジェクトのすべ てのプロパティを取得します。このレベルは、動 的な変更後のWebページを保存し、ActiveScreen ファイルを圧縮した形式で保存します。 「最低]:各ステップのActiveScreenに、記録された オブジェクトとその親のみのプロパティを保存す るよう設定します。このレベルは、(動的な変更の 前の)全WebページのオリジナルのHTMLソース を保存し、ActiveScreenファイルを圧縮した形式で 保存します。 「なし]:すべてのアプリケーションおよびWeb
	ページを対象とする ActiveScreen ファイルのキャプ チャを無効にします。
[ユーザ定義レベル]	カスタムの ActiveScreen オプションを指定できるよう にします。詳細については、1175ページ 「ActiveScreen キャプチャのユーザ定義設定」を参照し てください。
[標準レベル]	キャプチャ・レベルの設定を既定のレベル(部分的) に戻します。

オプション	詳細
[詳細設定]	ActiveScreen における Web ページの表示動作を指定で きます。詳細については、1179 ページ「Web ページ の外観」を参照してください。

ActiveScreen キャプチャのユーザ定義設定

[ActiveScreen キャプチャのユーザ定義設定] ダイアログ・ボックスでは, QuickTest による ActiveScreen 情報のキャプチャと保存の方法をカスタマイズで きます。

ActiveScreen のユーザ設定を適用すると, [ActiveScreen] タブでのキャプチャ・レベルの設定が, [ActiveScreen キャプチャのユーザ定義設定] ダイアログ・ボックスでのすべての設定で置き換えられます。

[ActiveScreen キャプチャのユーザ定義設定] ダイアログ・ボックスの標準設定 には, [オプション] ダイアログ・ボックスの [ActiveScreen] タブで選択され たキャプチャ・レベルの設定は反映されません。特定の設定だけカスタマイズ する場合は, [**戻した後の値**] オプションを使用してほかのすべての設定に既 定のキャプチャ・レベルの設定が使用されるようにした後で,必要に応じて個 別の設定を変更します。

ActiveScreen キャブチャのユーザ定義設定	×
 ■ ・検 ・	•
 iavapackage / i ・キャブチャのレベル O 完全 	•
詳細	
ユーザ定義設定を元に戻す 戻した後の値: 完全 ▼ 戻す(E)	
OK キャンセル ヘルプ	

注: [ActiveScreen キャプチャのユーザ定義設定] ダイアログ・ボックスには, コンピュータにインストールされている任意の QuickTest アドインに適用され るオプションが含まれている場合もあります。これらのオプションの詳細につ いては,『**HP QuickTest Professional アドイン・ガイド**』の関連する項を参照し てください。

[一般] オプション

ActiveScreen がキャプチャした情報を格納するために QuickTest が使用する圧縮 のタイプを指定できます。

- ▶ [シンプル]: ActiveScreen のキャプチャ・データを標準の非圧縮ファイル形式 (.html, .png など)で保存するよう QuickTest を設定します。
- ▶ [E縮]: ActiveScreen のキャプチャ・データを圧縮ファイル形式(ZIP形式)で 保存するよう QuickTest を設定します。このオプションを使用するとディスク 領域を節約できますが、ActiveScreenへの画像の読み込み時間に影響が出る場 合があります。これは標準設定のオプションです。

[Windows アプリケーション] オプション

ActiveScreen 用にキャプチャする Windows アプリケーションの各オブジェクト について、どのプロパティをキャプチャするかを設定できます。

▶ [完全]:各ステップの ActiveScreen に、アプリケーションの開いているウィン ドウまたはダイアログ・ボックス内のすべてのオブジェクトのすべてのプロパ ティを保存するよう QuickTest を設定します。

このオプションにより, すべてのステップの ActiveScreen で, ウィンドウまた はダイアログ・ボックスの任意のオブジェクトに対してチェックポイントを挿 入したりほかの操作を実行したりできるようになります。

▶ [部分的]:(標準設定)アプリケーションに対して実行された最初のステップで、アプリケーションの開いているウィンドウおよびダイアログ・ボックスのすべてのオブジェクトのすべてのプロパティ、および、同じウィンドウにおいて以降のステップで記録されるオブジェクトのすべてのプロパティを保存するよう QuickTest を設定します。

このオプションにより、ActiveScreen に表示される任意のオブジェクトに対し て、チェックポイントを挿入したり、他の操作を実行したりできるようになる と同時に、記録時間とディスク容量を節約できます。このオプションの場合、 以降のステップで ActiveScreen の情報が一部更新されないことがあります。 ▶ [最小]:各ステップのActiveScreen に、記録されたオブジェクトとその親のみのプロパティを保存するようQuickTestを設定します。

このオプションによって、記録時間が短くなり、必要となるディスク容量は比較的小さくなります。ただし、チェックポイントの挿入やほかの操作の対象にできるのは、記録されたオブジェクトとウィンドウまたはダイアログ・ボックスだけとなります。ActiveScreenに表示されるほかのオブジェクトに対して操作は実行できません。

▶ [なし]: Windows アプリケーションの ActiveScreen ファイルのキャプチャを無 効にします。

このオプションによって、非常に記録時間が短くなり、必要となるディスク容 量は最小限になります。ただし、ActiveScreen で記録後のテスト編集を実行で きません。

[Web] オプション

ActiveScreen 用に Web ページをキャプチャするかどうかを QuickTest に指定す ることができます。

➤ [ActiveScreen のキャプチャを無効にする]: ActiveScreen でのすべてのステップの画面キャプチャを無効にします。

このオプションを選択しない場合,テストの編集を終えた後に[名前を付けて 保存]を選択し,[ActiveScreen ファイルの保存]チェックボックスをクリア すれば,ActiveScreenの情報を削除することもできます。詳細については,346 ページ「テストの保存」を参照してください。

▶ [元の HTML ソースをキャプチャする]: Web ページにおいてスクリプトが実行 される前の最初の状態での HTML ソースをキャプチャします。このオプション の選択を解除すると、HTML ソースに動的な変更が(ページの読み込み時にス クリプトが自動的に実行されるなどして)加えられた後に Web ページの HTML ソースをキャプチャするよう QuickTest が設定されます。

[ユーザ定義設定を元に戻す]

ユーザ定義設定を QuickTest においてあらかじめ定義されているいずれかのレ ベル([完全],[部分的],[最低] または [なし])に戻すには,[戻した後の 値] リストからレベルを選択して [戻す] をクリックします。指定可能なキャ プチャ・レベルの詳細については,1177ページ「[Windows アプリケーション] オプション」を参照してください。

Web ページの外観

[Web ページの外観] ダイアログ・ボックスでは、キャプチャされた Web ページを ActiveScreen にどのように表示するかを QuickTest に設定できます。

Web ページの外観
/ スクリプトの実行
O 無効
■ ActiveXコントロールをロードする ■ 画像をロードする
□ Java アプレットをロードする
ActiveScreen(こページがロードされる際、QuickTest でスクリプトを実行するかどうかを指定します。
OK キャンセル ヘルプ

[Webページの外観]ダイアログ・ボックスには、次のオプションが含まれます。

- ➤ [スクリプトの実行]:次のいずれかのオプションに従って、ActiveScreenに Webページが読み込まれるときにQuickTestでスクリプトを実行するかどうか を指定します。
 - ▶ [有効]: ActiveScreen にページを読み込むたびにスクリプトを実行します。
 - ▶ [自動]:表示されるページに応じて、必要な場合にスクリプトを実行します。
 - ▶ [無効]: ActiveScreen にページを読み込むときに、スクリプトを実行しません。

注:このオプションはページの読み込み時に実行されるスクリプトだけを対象 とします。ActiveScreenの画面で操作を行なったときにスクリプトを動作させ るものではありません。

- ➤ [ActiveX コントロールをロードする]: 各ステップにおいてページが実際にどのようにアプリケーションに表示されるかプレビューできるように、ブラウザ・ページから ActiveScreen に ActiveX コントロールを読み込むよう QuickTest を設定します。このオプションをクリアすると、すべての ActiveX コントロール・オブジェクトについて、ActiveScreen に標準の ActiveX 画像が表示されます。
- ▶ [**画像をロードする**]: ブラウザ・ページから ActiveScreen 表示枠に画像をロードするように QuickTest を設定します。
- ▶ [Java アプレットをロードする]: 各ステップにおいてページが実際にどのようにアプリケーションに表示されるかプレビューできるように、ブラウザ・ページから ActiveScreen に Java アプレットを読み込むよう QuickTest を設定します。このオプションをクリアすると、すべての Java アプレット・オブジェクトについて、ActiveScreen に標準の Java 画像が表示されます。

注:

- ➤ QuickTest では、ActiveX コントロールおよび Java アプレットが ActiveScreen に表示専用モードで読み込まれます。読み込まれた ActiveX オブジェクトま たは Java オブジェクトを操作したり、それらから追加情報を取得したりで きません。ActiveScreen でこれらの項目を操作するには、対応するアドイン を読み込み、ActiveX または Java オブジェクトを対象に直接記録を行う必要 があります。
- ➤ ActiveScreen に読み込まれた ActiveX コントロールおよび Java アプレットの 動作は、アプリケーションでの動作と異なる場合があります。読み込まれた 個々のコントロールまたはアプレットの実装にもよりますが、そのことが原 因で予期しない動作をする場合があります。

テストの実行オプションの設定

[実行] タブのオプションでは,QuickTest がテストを実行うする方法および実行 セッションの結果を [テスト結果] ウィンドウに表示する方法を指定します。

オプション <u>×</u>
一般 フォルダ ActiveScreen 実行 Windows アプリケーション Web
実行モード
▼ 失敗したステップごとに Quality Center に不具合を送信する(S)
▶ 天行ゼッジョン於了時に結果を表示する(型)
□ 他の HP 製品でテストおよびコンポーネントを実行可能にする(W)
▼ 静止画像キャプチャをテスト結果へ保存(I): エラー発生時 ▼
□ ムービーをテスト結果へ保存(M): エラー発生時 🔽
○ エラーと警告の発生前(U) 2048 🗧 KBまでのムービーセグメントを保存
€ 実行全体のムービーを保存(日)
詳細(A) クリックして画面レコーダの環境を設定

[実行] タブには, 次のオプションがあります。

オプション	詳細
[実行モード]	QuickTest に対してテストの実行方法を指定します。
	 ▶ [標準 (実行矢印を表示)]: キーワード・ ビューまたはエキスパート・ビューの左に実行 矢印を表示しながらテストを実行し,実行した 各ステップまたはステートメントがわかるよう にします。テストに複数のアクションが含まれ ている場合,キーワード・ビューの[項目]カ ラムのツリーが展開されてステップが表示さ れ,エキスパート・ビューには現在実行してい るアクションのスクリプトが表示されます。 [ステップ実行ごとの遅延時間]: 連続する各ス テップを実行するまでに QuickTest が待機する 時間をミリ秒単位で指定できます(最大 10000 ミリ秒車で)
	次の説明のとおり,[標準]実行モード・オプ ションには,[高速]オプションよりも多くの システム・リソースが必要です。
	注 :このモードを有効にするには,Microsoft Script Debugger 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,『 HP QuickTest Professional インストール・ガイド』 を参照し てください。
	▶ [高速]:キーワード・ビューまたはエキスパート・ビューの左に実行矢印を表示せずにテストを実行します。項目ツリーの展開表示や、実行している各アクションのスクリプトの表示は行いません。このオプションでは、必要とされるシステム・リソースが少なくて済みます。
	注 :Quality Center からテスト・セットを実行す る場合, 標準 モードが選択されていても,テス トは自動的に 高速 モードで実行されます。

オプション	詳細
[失敗したステップごとに Quality Center に不具合を送信 する]	テストの失敗したステップごとに不具合を自動的 に Quality Center に送信するように QuickTest を設 定します。このオプションは, Quality Center プロ ジェクトに接続されている場合にのみ利用できま す。詳細については, 990 ページ「Quality Center プロジェクトへの自動による不具合の送信」を参 照してください。
[実行セッション終了時に結果 を表示する]	実行セッションの後に結果を自動的に表示するように QuickTest を設定します。
[他の HP 製品でテストおよび コンポーネントを実行可能にす る]	Quality Center や Test Batch Runner などほかの HP 製 品で QuickTest のテストを実行できるようにします。 注:このオプションは WinRunner で QuickTest のテ ストを実行する場合には必要ありません。
[静止画像キャプチャを結果へ 保存]	 テスト結果に表示するために、アプリケーションの静止画像を実行セッション中にキャプチャおよび保存するタイミングをQuickTestに指定します。 リストからオプションをどれか選択します。 「常に]:実行時にすべてのステップの画像をキャプチャします。 [エラー発生時]:失敗したステップの画像のみをキャプチャします。これが標準設定です。 [エラーおよび警告発生時]:失敗または警告を返したステップのみ画像をキャプチャします。 詳細については、970ページ「アプリケーションの静止画像およびムービーのキャプチャと表示」を参照してください。

オプション	詳細
[ムービーをテスト結果へ保存]	 テスト結果に表示するために、アプリケーションのムービーを実行セッション中にキャプチャおよび保存するタイミングをQuickTestに指定します。 標準設定ではこのオプションは無効です。 リストからオプションを選択します。 ▶「當仁」: 実行時にすべてのステップのムービー
	 をキャプチャします。 [エラー発生時]:失敗したステップのムービーのみをキャプチャします。 [エラーおよび警告発生時]:失敗または警告を返したステップのみムービーをキャプチャします。 詳細については、970ページ「アプリケーションの静止画像およびムービーのキャプチャと表示」を参照してください。
[エラーと警告の発生前 KB までのムービー セグメントを 保存] ([エラー発生時] または [エ ラーおよび警告発生時] が [ムービーをテスト結果へ保存] オプションで選択されている場 合のみ有効)	選択すると、QuickTest によってエラー(または警告)ごとにムービー・セグメントが保存されます。 各セグメントには、失敗したステップ(または警告)までの指定したキロバイト数のムービーが含まれます。400 (0.4MB)から2097152 (2GB)までの任意の値を入力できます。テスト実行で複数のセグメントがキャプチャされた場合は、QuickTest によって、すべての関連するムービー・ セグメントで構成されるテスト結果を持つ単一のムービーが保存されます。
[実行全体のムービーを保存] ([エラー発生時]または[エ ラーおよび警告発生時]が [ムービーをテスト結果へ保存] オプションで選択されている場合のみ有効)	選択すると、少なくとも1つのエラー(または警告)が発生したら、QuickTest によって、実行全体のムービーが保存されます。
[詳細] ([ムービーをテスト結果へ保 存]を選択した場合にのみ有 効)。	ムービーのファイル・サイズおよび表示形式に影響する,画面レコーダの詳細オプションを設定します。。
[画面レコーダのオプション] ダイアログ・ボックスについて

[画面レコーダのオプション] ダイアログ・ボックスでは、ムービーのファイル・サイズ、表示設定、および記録パフォーマンスに影響する、画面レコーダのオプションを設定できます。Windowsの機能に影響する設定は、セッションの終了時に元の状態に戻されます。

画面レコーダのオブション
画面レコーダの全ムービー クリップに適用するオブションを選択します。これらの オブションはムービー ファイルのサイズと外観に影響します。
□ サウンドを記録(B)
☑ プレーンな壁紙を設定(<u>S</u>)
☑ ウィンドウをドラッグするときに内容を表示しない(0)
キャプチャ ドライバ 画面レコーダのパフォーマンスを向上させるため、画面レコード キャプチャ ド ライバをインストールすることをお勧めします。
注:この操作を実行するには管理者権限が必要です。詳細については [ヘルプ]をクリックしてください。
「インストールト」
OK キャンセル ヘルプ

[画面レコーダのオプション] ダイアログ・ボックスには, 次のオプションが あります。

オプション	詳細
[サウンドを記録]	アプリケーションのムービーとともにサウンドを保存 するように QuickTest を設定します。
[プレーンな壁紙を設定]	実行セッション中のデスクトップの壁紙を青一色に設 定します。
[ウィンドウをドラッグする ときに内容を表示しない]	実行セッション中にウィンドウがドラッグされたとき に、ウィンドウの輪郭だけを表示して、内容は表示し ないように Windows を設定します。
[インストール] / [アンインストール]	画面レコーダ・キャプチャ・ドライバをインストール またはアンインストールします。画面レコーダ・キャ プチャ・ドライバは、ムービー記録中の画面レコーダ のパフォーマンスを向上させます。
	注 : リモート接続を介して QuickTest を実行する場合 は, 画面レコーダ・キャプチャ・ドライバをインス トールまたはアンインストールすることはできません。

Vista ユーザの方へ:上記のオプションに加えて,Windows Vista の配色テーマ が Aero に設定されている場合は,パフォーマンスを最大限に引き出すために, 実行セッションのムービーをキャプチャしている間は,QuickTest によってテー マが自動的に Vista Basic に設定されます。実行セッションが終了すると,配 色テーマは元の設定に戻されます。

キャプチャしたムービーの使用方法の詳細については,972ページ「実行セッションのムービーの表示」を参照してください。

第 41 章

個別のテストのオプション設定

テストごとにテスト・オプションを設定することで、QuickTest による個々のテ ストの記録および実行を制御できます。

本章では、次の項目について説明します。

- ▶ 個別のテストのオプションの設定について(1188ページ)
- ▶ [テストの設定] ダイアログ・ボックスの使用(1189ページ)
- ▶ テストのためのプロパティの定義(1191ページ)
- ▶ テストのための実行設定の定義(1195ページ)
- ▶ テストのためのリソース設定の定義(1199ページ)
- ▶ テストのパラメータの定義(1204ページ)
- ▶ テストのための環境設定の定義(1207ページ)
- ▶ テストのための回復シナリオ設定の定義(1215ページ)

個別のテストのオプションの設定について

特定のテストの記録および実行方法に影響を与えるテスト・オプションを設定 できます。たとえば、データ・テーブルの特定の行についてだけパラメータ化 されたテストを実行するように、QuickTest に指示することができます。指定し た個々のテスト・オプションは、テストを保存するときに保存されます。

注: すべてのテストおよびコンポーネントに影響を与えるテスト・オプション も設定できます。詳細については,第40章「グローバル・テスト・オプショ ンの設定」を参照してください。

[テストの設定] ダイアログ・ボックスの使用

テストを記録または実行する前に, [テストの設定] ダイアログ・ボックスを 使用して,特定のテストのテスト・オプションを変更できます。

単独のテストのテスト・オプションを設定するには、次の手順を実行します。



1 [**ファイル**] > [設定] を選択するか, [テストの設定] ツールバー・ボタンを クリックします。[テストの設定] ダイアログ・ボックスが開きます。このダ イアログ・ボックスは,内容ごとにタブ・ページに分かれています。

テストの設定		x
プロパティ 実行 リソース	、 パラメータ 環境 Web 回復	
名前:	Test1	
作成者:	Administrator	
作成アプリケーション:	QuickTest Professional 9.5	
最終アプリケーション:	QuickTest Professional 9.5	
場所(上):		
C:¥Program Files¥HP	#QuickTest Professional¥Tests¥Test1	
記述(D):		
		1
関連アドイン:		
🗹 Web		
	変更(<u>M</u>)	
<u></u>	指定C1WCTスF設定ビオードメーンヨノスクリフトを主めします。	
	OK 「たいない」 液田(A) 「「へいつ」	

2 使用するタブを選択し、必要に応じてオプションを設定します。各タブで使用 できるオプションの詳細については、次の表を参照してください。 3 変更を適用してダイアログ・ボックスを開いたままにしておくには [適用] を クリックします。変更を適用してダイアログ・ボックスを閉じるには [OK] をクリックします。

[テストの設定] ダイアログ・ボックスには、次のタブがあります。

タブ	タブの内容
[プロパティ]	テストの説明や関連アドインなど、テストのプロパティを 設定するためのオプション。詳細については、1191ページ 「テストのためのプロパティの定義」を参照してください。
[実行]	テストの実行セッションに関する設定を行うためのオプ ション。詳細については,1195ページ「テストのための実 行設定の定義」を参照してください。
[リソース]	関数ライブラリに格納されている VBScript 関数ライブラリ など,テストに関連付けるリソースを指定するオプショ ン。詳細については,1199ページ「テストのためのリソー ス設定の定義」を参照してください。
[パラメータ]	テストの入力パラメータおよび出力パラメータを指定する ためのオプション。詳細については,1204ページ「テスト のパラメータの定義」を参照してください。
[環境]	既存およびユーザ定義の環境変数の表示のほか,ユーザ定 義環境変数の追加,変更,保存,およびアクティブな外部 環境変数ファイルの選択のためのオプション。詳細につい ては,1207ページ「テストのための環境設定の定義」を参 照してください。
[回復]	実行セッション中に、テスト環境で生じる予期しないイベ ントやエラーから QuickTest が回復する方法を設定するオ プション。詳細については、1215 ページ「テストのための 回復シナリオ設定の定義」を参照してください。

[テストの設定] ダイアログ・ボックスにはこれらのタブのほかに、インストー ルまたは読み込まれている QuickTest アドインがあれば、それらに対応するタブ が含まれることもあります。アドインの詳細については、『HP QuickTest Professional アドイン・ガイド』の関連する項を参照してください。

テストのためのプロパティの定義

[テストの設定] ダイアログ・ボックス([**ファイル**] > [設定])の[プロパ ティ] タブを使用して,テストに関連付けられているアドインなど,テストに 関する一般情報の表示および定義が可能です。また,テスト設定のためのオー トメーション・スクリプトを生成することもできます。

テストの設定		×
プロパティ 実行 リソー	ス パラメータ 環境 Web 回復	
名前·	Testi	
作成者:	Administrator	
作成アプリケーション:	QuickTest Professional 9.5	
最終アプリケーション:	QuickTest Professional 9.5	
場所(L):		
C:¥Program Files¥HI	P¥QuickTest Professional¥Tests¥Test1	1
記述(D):		- 11
関連アドイン:		
Veb Web		
	密 更(M)	11
		-
スクリプトの生成(巴	指定されたテスト設定でオートメーション スクリプトを生成します。	
	OK キャンセル 適用(点) へルプ	

[テストの設定] ダイアログ・ボックスの [プロパティ] タブには,次の項目 があります。

オプション	詳細
[名前]	テストの名前を示します。
[作成者]	テストを作成した人物の Windows ユーザ名を示します。

オプション	詳細
[作成アプリケーション]	テストの作成に使用した QuickTest のバージョンを示します。
[最終アプリケーション]	最後にテストに変更を加えた QuickTest のバージョンを示します。
[場所]	テストのパスおよびファイル名を示します。
[記述]	テストの説明を指定できます。
[関連アドイン]	テストに関連付けられているアドインが表示されます。詳 細については,1192ページ「テストへのアドインの関連付 け」を参照してください。
[変更]	テストに関連付けるアドインを選択できます。詳細につい ては,1193 ページ「関連アドインの変更」を参照してくだ さい。
[スクリプトの生成]	現在のテスト設定を含んだオートメーション・スクリプト を生成します。詳細については、1311 ページ「QuickTest 操 作のオートメーション」、または『QuickTest Automation Reference』(英語版)([ヘルプ] > [QuickTest Professional ヘルプ] > [QuickTest Advanced References] > [QuickTest Automation])を参照してく ださい。

テストへのアドインの関連付け

アドインを選択して読み込むには、QuickTestの起動時に[アドインマネージャ]ダイアログ・ボックスで対象アドインを選択します。必要なアドインが 読み込まれている任意の環境での記録が可能です。

新しいテストを作成すると、現在読み込まれているアドインが自動的にテスト に関連付けられます。

アドインをテストに関連付けると、そのテストを開くたびに、関連付けられて いるアドインが読み込まれているかどうかが QuickTest によってチェックされ ます。

テストを開いたとき、関連付けられているアドインが現在読み込まれていないか、テストに関連付けられていないアドインが読み込まれている場合は、そのことがQuickTestによって通知されます。このチェック処理により、アドインを読み込んでいないことが原因で実行セッションが失敗するのを確実に防止で

きます。また、現在開いているテストで使用するアドインを、必須アドインと して関連アドイン・リストに追加できます。アドインの読み込みと使用方法の 詳細については、『HP QuickTest Professional アドイン・ガイド』を参照してく ださい。

Quality Center では、関連付けられているアドイン・リストを使用して、QuickTest の起動時に読み込むアドインを決定します。Quality Center での作業の詳細については、第47章「Quality Center を使用した作業」を参照してください。

関連アドインの変更

アドインをテストに関連付けたり,関連付けを解除したりするには,[関連ア ドインの変更]ダイアログ・ボックスを使います。

関連7	パインの変更	×
次のリ コンポ ストまえ 除する	は入は現在ロードしているアドインの他、テストま ーネントに関連した全アドインを表示します。 ンチェックボックスを選択して、アドインを使用中 たはコンボーネントに関連付けます。関連付けま 5(こはアドインチェックボックスをクリアします。	たは ¹ のテ ₁ 解
	ActiveX Visual Basic Web	
記述: Tests	s ActiveX controls OK キャンセル ヘルフ	9

このダイアログ・ボックスには、テストに現在関連付けられているすべてのア ドインが、現在 QuickTest に読み込まれているその他のアドインとともに一覧 表示されます。テストに関連付けられているが現在読み込まれていないアドイ ンは、グレーで表示されます。 **注**: このリストには,追加の環境またはコントロールをサポートするために ユーザまたはサードパーティがアドイン拡張を使用して開発したアドインを表 す子ノードが含まれている場合があります。詳細については,関連するアドイ ン拡張の開発者ガイドを参照してください(拡張のセットアップに含まれてい ます)。

テストに関連付けるアドインのチェック・ボックスを選択したり、テストに関 連付けないアドインのチェック・ボックスをクリアしたりできます。[関連ア ドインの変更]ダイアログ・ボックスに子アドインが含まれていて、それを選 択した場合は、その親アドインも自動的に選択されます。親アドインのチェッ ク・ボックスをクリアすると、子アドインのチェック・ボックスもクリアされ ます。

上の例では次のようになっています。

- ▶ Web が読み込まれ、テストに関連付けられている。
- ▶ ActiveX は読み込まれているが、テストに関連付けられていない。
- ▶ Visual Basic はテストに関連付けられているが, 読み込まれていない。

注:現在読み込まれていないアドインをテストに関連付けるには,QuickTest を 再起動して,そのアドインを[アドインマネージャ]から読み込みます。 QuickTest の起動時に[アドインマネージャ]ダイアログ・ボックスが表示さ れない場合には,QuickTest の次回起動時に表示されるように設定できます。そ れには,[オプション]ダイアログ・ボックスの[一般]タブで,[起動時に表 示する]を選択します。[オプション]ダイアログ・ボックスの詳細について は,第40章「グローバル・テスト・オプションの設定」を参照してください。 アドイン・マネージャの詳細については,『HP QuickTest Professional アドイ ン・ガイド』のQuickTest アドインでの作業に関する項を参照してください。 オートメーション・スクリプトを使用してこのリストを取得し, リストに基づいてアドインを読み込めます。オートメーション・スクリプトでの作業に関する詳細については, 『QuickTest Automation』([ヘルプ] > [QuickTest Professional ヘルプ] > [QuickTest Advanced References] > [QuickTest Automation]) を参照してください。

テストのための実行設定の定義

テストを実行すると、アプリケーションについて記録したステップが QuickTest によって実行されます。

[テストの設定] ダイアログ・ボックス([**ファイル**] > [設定])の[実行] タブを使って,実行セッション中にエラーが生じた場合に何をするかを選択 し,オブジェクト同期タイムアウトを設定し,スマート認識メカニズムを無効 にするかどうかを選択できます。

テストの設定
プロパティ 実行 リソース パラメータ 環境 Web 回復
データテーブルの反復 ○ 反復なしで実行する(L) ○ 実行開始行(L): 1 終了行: 1
実行セッション中にエラーが発生した場合(N): メッセージ ボックスをボップアップする ▼ オブジェクト同期化のタイムアウト(I): 20 秒 □ 実行セッション中にスマート認識を無効にする(D)
□ エラー発生時、デスクトップの画像を保存する(S) (HP Business Process Monitor でテストが実行される場合)
OK キャンセル 適用(A) (二ヘルプ)

標準設定では、グローバル・パラメータを持つテストを実行した場合、 QuickTestによって、指定したパラメータに従って、データ・テーブルの行ごと にテストが実行されます。詳細については、659ページ「グローバルまたはア クション・データ・テーブル・パラメータの選択」を参照してください。

[実行] タブを使用して、データ・テーブルの [グローバル] タブの特定の行 に限定して、テストの反復を実行するように QuickTest に指示できます。

注: [テストの設定] ダイアログ・ボックスの [実行] タブは, テスト全体に 適用されます。テストの個々のアクションの実行プロパティを設定するには, 選択したアクションの [アクション呼び出しプロパティ] ダイアログ・ボック スの [実行] タブを使用します。アクションの実行プロパティの詳細について は, 454 ページ「アクションの実行プロパティの設定」を参照してください。

[実行] タブには、次のオプションがあります。

オプション	詳細
[データ テーブルの反	テストの反復の設定を行います。次のいずれかのオプショ
復]	ンを選択します。
	 「反復なしで実行する]: グローバル・データ・テーブルの最初の行だけを使って、テストを1回だけ実行します。 「すべての行で実行する]: グローバル・データ・テーブルのすべての行を使って、テストを反復実行します。 [実行開始行 X 終了行]: グローバル・データ・テーブルの指定した範囲にある行の値を使って、テストを反復実行します。
[実行セッション中に	実行セッション中に発生したエラーに対応する QuickTest の
エラーが発生した場	応答を指定します。詳細については,1197 ページ「エラー
合]	に対する応答の指定」を参照してください。

オプション	詳細
[オブジェクト同期化 のタイムアウト]	QuickTest がテストのステップを実行する前に、オブジェクトの読み込みが終わるのを待機する最長の時間(秒単位)を設定します。
	注: web オフシェクト C作業しているとき, Quick lest は [ブラウザ ナビゲーションのタイムアウト] に設定されてい る時間と [オブジェクト同期化のタイムアウト] に対して 設定されている時間の合計の長さだけ待機します。[ブラウ ザナビゲーションのタイムアウト] オプションの詳細につ いては, 『HP QuickTest Professional アドイン・ガイド』を 参照してください。
[実行セッション中に スマート認識を無効に する]	実行セッション中にスマート認識メカニズムを使わないよ うに QuickTest を設定します。
	注 : このオプションを選択すると, [オブジェクトのプロパ ティ] と [オブジェクトリポジトリ] ダイアログ・ボック スの中の [スマート認識を有効にする] が無効になります。 ただし設定は保存されています。このオプションをクリア すると, [スマート認識を有効にする] は以前の設定 (True または False) に戻ります。
[エラー発生時、デス クトップの画像を保存 する(HP Business Process Monitor で テストが実行される場 合)]	このオプションは, HP Business Availability Center の Business Process Monitor コンポーネントによって実行される テストにのみ適用されます。
	このオプションを選択すると, Business Process Monitor に よって開始されたテストの実行セッションでエラーが生じ たときにデスクトップの画面ショットを取得するよう QuickTest が指示されます。取得した画像は Business Availability Center に保存されます。実行結果は, Business Process Monitor によって Business Availability Center サーバに 転送されます。

エラーに対する応答の指定

標準の設定では、実行セッション中にエラーが生じると、エラーを説明する ポップアップ・メッセージ・ボックスが QuickTest によって表示されます。実 行セッションを続けるか終了するには、メッセージ・ボックスのボタンをク リックします。 「ポップアップ・メッセージ・ボックス」オプションを受け入れるか, [実行 セッション中にエラーが発生した場合] ボックスのリストにある次のいずれか のオプションを選択することで,別の応答方法を指定できます。

- ▶ 次のアクション反復に進む:エラーが発生したときに、QuickTest は次の反復に 進みます。
- ▶ 実行を停止する:エラーが発生したときに、QuickTest はテストを停止します。
- ▶ 次のステップに進む:エラーが発生したときに、QuickTest はテストの次のステップに進みます。

QuickTest では、まずテストに関連付けられている回復シナリオがあればそれが 実行され、回復シナリオでエラーを解消できない場合にのみ、前述の手順で選 択したオプションが実行されます。詳細については、1215ページ「テストのた めの回復シナリオ設定の定義」を参照してください。

注:多数のテストで作業している場合は,QuickTest オートメーション・スクリ プトを使用してテストごとに異なる値を設定する必要がある場合があります。 このオプションを制御するオートメーション・スクリプトの行にアクセスする には,[テストの設定]ダイアログ・ボックスの[プロパティ]タブにある [**スクリプトの生成**]ボタンを使用します。

詳細については、1311 ページ「QuickTest 操作のオートメーション」、または 『QuickTest Automation』([ヘルプ] > [QuickTest Professional ヘルプ] > [QuickTest Advanced References] > [QuickTest Automation]) を参照し てください。

テストのためのリソース設定の定義

[テストの設定] ダイアログ・ボックス([**ファイル**] > [**設定**])の[リソース] タブを使って、VBScript 関数ライブラリやデータ・テーブル・ファイルな ど、特定のファイルをテストに関連付けます。また、現在関連付けられている 関数ライブラリの設定を、以降すべての新規テストの標準設定として設定でき ます。

テストの設定	×
プロパティ 実行 リソース パラメータ 環境 Web 回復 ライブラリー テストと関連付ける関数ライブラリを入力し、優先順位を付けてください。	1
間連付けのある関数ライブラリ: ・ × ↑ ↓ Check_data_validity.qfl ・ verifyproperty.qfl	
標準値(2設定(<u>S</u>) 構文チェック(<u>C</u>) アクセス可能なすべてのファイルで構文をチェックします。 データテーブルー ○ 標準設定の場所 (F2トディレクトリ内)(F)	
○ その他(回):	

注:オブジェクト・リポジトリは、テスト内の個々のアクションに関連付けられます。オブジェクト・リポジトリをアクションに関連付けるには、[アクションのプロパティ]ダイアログ・ボックス([**編集**] > [**アクション**] > [**アクションのプロパティ**])および[リポジトリの関連付け]ダイアログ・ボックス([**リソース**] > [**リポジトリの関連付け**])を使用します。

[テストの設定] ダイアログ・ボックスの [リソース] タブには, 次のオプ ション領域があります。

オプション領域	詳細
[ライブラリ]	テストに関連付けられている関数ライブラリが表示されます。ファ イルの追加,削除,優先順位設定が可能です。また,以降の新規テ ストの標準関数ライブラリを設定できます。詳細については,1201 ページ「関連関数ライブラリの指定」を参照してください。
[標準値に設定]	関数ライブラリの現在のリストを,以降の新規テストに関連付け る標準設定のリストとして設定します。
	注 : [標準値に設定] オプションは,テストが対象の場合にのみ使 用できます。このオプションは,対象テストの設定が全テストの 標準設定値と異なる場合に有効になります。
[構文チェック]	関連付けられた関数ライブラリにテストの正常な実行を妨げる構 文エラーが含まれているかどうかをチェックします。テストを確 定する前に、[構文チェック] ボタンをクリックしてファイルの構 文エラーをチェックしてください。構文エラーが見つかると、情 報表示枠が開き、構文エラーを含んでいるファイルのリストが表 示されます。エラーがなければ、すべての関数ライブラリの構文 が有効であることを知らせる情報ボックスが開きます。
	注 : QuickTest は、アクセスできる関連付けられた関数ライブラリ のみを検査します。たとえば、関連付けられた関数ライブラリが、 現在接続されていない Quality Center プロジェクトに格納されてい る場合、その構文は検査されません。

オプション領域	詳細
[データ テーブ ル]	 テストで使用するデータ・テーブルの場所を指定します。 「標準設定の場所(テストディレクトリ内)]を選択すると、テスト・フォルダの下にあるデータ・テーブルの標準の格納場所に保存されているデータを使用するよう、QuickTestが設定されます。 「その他]を選択すると、データ・テーブルが格納されている指定の場所に保存されているデータを使用するよう、QuickTestが設定されます。データ・テーブルには、任意のMicrosoft Excel(.xls)ファイルを使用できます。 データ・テーブルの格納場所の選択に関する詳細については、1132ページ「データ・テーブルの編集」を参照してください。
	注 : Quality Center に格納されている Microsoft Excel ファイルを データ・テーブルとして指定できます。詳細については、1141 ページ「Quality Center でのデータ・テーブル・ファイルの使用」 を参照してください。

関連関数ライブラリの指定

[リソース] タブの [**関連付けのある関数ライブラリ**] 表示枠は,テストに関 連付けられている関数ライブラリのリストを示します。それらのファイルの中 から,テストで指定した VBScript 関数,サブルーチンなどが検索されます。

テストのステップから呼び出される関数またはサブルーチンが QuickTest に よって検索される順序は、このリスト内の関数ライブラリの順序によって決ま ります。同じ名前の関数またはサブルーチンが2つある場合、QuickTest は最初 に見つかった方を使用します。詳細については、905ページ「関連付けられて いる関数ライブラリを使用した作業」を参照してください。

関連関数ライブラリは、相対パスを使って指定できます。実行セッション中、 QuickTest は現在のテストのディレクトリでファイルを検索し、次に[オプショ ン]ダイアログ・ボックスの[フォルダ]タブに表示されるフォルダで検索を 行います。詳細については、1169ページ「テストのフォルダ・オプションの設 定」および 340ページ「QuickTest での相対パスの使用」を参照してください。 **注**: テストで使用する関数ライブラリがファイル・システムに格納されている 場合,そのテストをほかのユーザまたは HP 製品がほかのコンピュータ上で実 行できるようにするには,パスを相対パスとして指定する必要があります(パ スを一度クリックするとパスが強調表示され,その状態で再度クリックすると 編集モードになります)。以後はほかのユーザも,QuickTest で相対パスを参照 するドライブ文字とフォルダを [オプション] ダイアログ・ボックス ([ツー ル] > [オプション])の [フォルダ] タブで指定することにより,同じテス トを実行できるようになります。詳細については,1169ページ「テストのフォ ルダ・オプションの設定」および 340ページ「QuickTest での相対パスの使用」 を参照してください。

次の関数ライブラリ制御ボタンを使用して,テストに関連付けられている関数 ライブラリの追加,削除,優先順位設定が可能です。

オプ ション	詳細
+	関数ライブラリをテストに関連付けます。関数ライブラリの絶対パスま たは相対パス,およびファイル名を入力するか,参照ボタンを使用して 必要なファイルを指定します。関数ライブラリに構文エラーがある場合, 構文エラーが原因でテストが失敗することを示すメッセージが表示され ます。
	Quality Center のプロジェクト・フォルダ内のファイルを関連付けることも可 能です。詳細については、次に示す 1202 ページ「Quality Center プロジェク ト・フォルダ内の関数ライブラリの関連付け」を参照してください。
×	リストから関連付けられている関数ライブラリを削除します。
1	選択した関数ライブラリの優先順位を上げます。
4	選択した関数ライブラリの優先順位を下げます。

Quality Center プロジェクト・フォルダ内の関数ライブラリの関連付け

Quality Center に接続されている状態で **十** ボタンをクリックすると, QuickTest に [QualityCenter] が追加され, Quality Center のパスを指定するための参照ボタンが表示されます。

Quality Center に接続していないときに Quality Center プロジェクト・フォルダ 内のファイルを追加するには、SHIFT キーを押しながら デボタンをクリックし ます。QuickTest によって [QualityCenter] が追加され、パスが入力できるよう になります。Quality Center パス全体を手作業で入力することもできます。入力 する場合は [QualityCenter] の後にスペースを追加する必要があります。たとえ ば、次のように指定します。[QualityCenter] Subject¥Tests のようにします。

注: QuickTest でテストを実行する際には、対応する Quality Center プロジェクトに接続している場合に限り、Quality Center のプロジェクト・フォルダに格納されている関連付けられている関数ライブラリが使用されます。

Quality Center プロジェクトでの作業の詳細については,第47章「Quality Center を使用した作業」を参照してください。

テストのパラメータの定義

[テストの設定] ダイアログ・ボックス([**ファイル**] > [**設定**])の[パラ メータ] タブでは,テストに値を渡す入力パラメータ,およびテストから外部 のソースに値を渡す出力パラメータを定義します。また,既存のテスト・パラ メータの変更や削除を行うこともできます。

テストのパラメータは,アクションのパラメータに似ています。アクション・ パラメータの詳細については,444ページ「アクション・パラメータの設定」 を参照してください。

テストの設定				×
プロパティ 実行	リソース パラメ	─タ 環境 Web	回復	
入力パラメータ			+ >	<
名前	タイプ	標準設定値	記述	
username	文字列	Nicole	"username" edit box(2	
		1		-
出力パラメータ	L (-0		+ >	<u>(</u>
名前 Status	タイプ 掛/店		lit allowed 2: depied	
Jacus	T T T			
	0	K ++)	ンセル 適用(A)	NI7

[パラメータ] タブには,次の2つのパラメータ・リストがあります。

▶ [入力パラメータ]:実行元または呼び出し元から渡される値をテストで受け取るためのパラメータを指定します。

▶ [出力パラメータ]:実行元または呼び出し元にテストから値を渡すためのパラ メータを指定します。

既存のパラメータを編集するには,該当するリストで対象のパラメータを選択 し,詳細を編集します。

テストの入力パラメータおよび出力パラメータを追加または削除するには,次 のパラメータ制御ボタンを使用します。

オプ ション	詳細
+	該当するパラメータ・リストにパラメータを追加します。新しいパラ メータの名前(大文字と小文字は区別されます)を入力し,パラメータ のタイプを選択してください。テストにおける当該パラメータの用途な ど,パラメータの詳細を入力できます。
	入力パラメータを定義する場合には,指定のパラメータのタイプに対す る標準設定値が自動的に入力されます。パラメータの標準設定値を[標 準設定値]カラムで変更できます。詳細については,次に示す1206ペー ジ「入力パラメータの値の定義」を参照してください。
	テストのパラメータは、アクション・パラメータと同様の方法で定義し ます。パラメータおよびパラメータ・タイプの定義方法の詳細について は、444ページ「アクション・パラメータの設定」を参照してください。
×	選択したパラメータをテストから削除します。

入力パラメータの値の定義

テストを実行したときにパラメータの値として実際に使用される値は、次の表 に示すように、テストを呼び出したアプリケーション(QuickTest または Quality Center)から送信される値です。

ドキュメントの タイプ	呼び出し元	パラメータ値の指定場所
テスト	QuickTest	[実行] ダイアログ・ボックスの [入力 パラメータ] タブ。詳細については, 938 ページ「テスト全体の実行」を参照 してください。
テスト	Quality Center	[テスト実行のプロパティ] ダイアログ・ ボックス(Test Lab モジュール)詳細に ついては,『 HP Quality Center ユーザー ズ・ガイド』を参照してください。

テストの実行時に,QuickTest またはQuality Center から値が渡されない入力パラメータがあった場合,それらのパラメータについては標準設定値が使用されます。

[テストの設定] ダイアログ・ボックスの [パラメータ] タブで新しいパラ メータを定義するとき、パラメータの標準設定値を指定したり、当該パラメー タのタイプに対して QuickTest が割り当てる標準設定値を受け入れたりするこ とができます。

値のタイプ	QuickTest の標準設定値
文字列	空文字列
ブール値	True
日付	現在の日付
数值	0
パスワード	空文字列
任意	空文字列

ステップにおけるテスト・パラメータの使用

テストのパラメータに直接アクセスできるのは、最上位アクションの入力パラ メータをパラメータ化する場合および最上位レベルの出力パラメータの格納場 所を指定する場合のみです。アクション内のステップでテスト・パラメータの 値を使用するには、ステップを含んでいるアクションにテスト・パラメータを 渡す必要があります。詳細については、444ページ「アクション・パラメータ の設定」を参照してください。

あるいは、Parameter ユーティリティ・オブジェクトを使用し、次の形式でエキス パート・ビューにパラメータ名を入力できます:Parameter("ParameterName")。 詳細については、654ページ「[エキスパートビュー]のステップでのアクショ ン・パラメータの使用」を参照してください。

テストのための環境設定の定義

[テストの設定] ダイアログ・ボックス([**ファイル**] > [**設定**])の[環境] タブには,既存の組み込み環境変数およびユーザ定義環境変数が表示されま す。また,このタブでは,ユーザ内部定義環境変数の追加,変更,および削除 が可能なほか,定義されている変数の外部 XML ファイルへの保存,および, ファイルからの変数の取得が可能です。

ユーザ定義変数を外部 XML ファイルにエクスポートすれば,エクスポートした環境変数ファイルをほかの任意のテストで使用できます。

環境変数および環境パラメータの詳細については,661ページ「環境変数パラ メータの使用」を参照してください。

[環境] タブには, [変数のタイプ] に関して次のオプションがあります。

- ▶ 組み込み: QuickTest Professional によって定義される組み込み環境変数および それらの現在の値が表示されます。
- ▶ ユーザ定義:内部および外部のユーザ定義環境変数およびそれらの現在の値が 表示されます。

組み込み環境変数

[組み込み]を選択すると、[環境] タブには QuickTest Professional によって定義される組み込み環境変数のリストが表示されます。

テストの設定		×
「プロパティ 実行 リソース	パラメータ 環境 Web 回復	
変数のタイプ(P): 組み込み	y	
名前	▲	
ActionIteration	現在実行しているアクションの反復を表示	
ControllerHostName	現住美行中のアクンヨンを示しより	
GroupName	シナリオ識別番号	
LocalHostName	ローカルホスト名	
OS	オペレーティング システム	
OSVersion	オペレーティングシステムのバージョン	
ProductDir	フロクラムのインストール先フォルタのバス	
4		
現在の値():		
Microsoft Windows 2003 S	erver	
Γ	OK キャンセル 適用(A) ヘルプ	
L		

組み込み環境変数に関して、次の情報が表示されます。

- ▶ [名前]:各組み込み環境変数の名前。
- ▶ [詳細]:各組み込み環境変数の簡単な説明。
- ▶ [現在の値]:選択した環境変数の現在の値。

ユーザ定義環境変数

[**ユーザ定義**]を選択すると,[環境]タブには,そのテストで利用できるユー ザ定義環境変数のリストが表示されます。

Ŧ	ストの設定					×
	プロパティ 実行	 リソース パ ラ メ	友 環境 ₩	eb 回復	1	
		·			· (
	変数のタイプ(<u>P</u>):	ユーザ定義	•		🛨 🗙 🗹	
	名前	値		5	パプ	
	Email	john.brown@	videac.com	9	本部	
	FirstName	John		9	木部	
	LastName	Brown		<u>9</u>	本部	
	PhoneNum	4126-1177		9	本部	
	Record	555		Ŕ	勺部	
						- -
	エクスポート区)	ユーザ定義の環	境変数を XML)	ファイルにエクスプ	ポートします。	
		,				
	☑ 変数と値を外れ	部ファイルからロート	"する(上)			
		est xml				(I
	Prive, P					1 1
		C	к *	ャンセル	適用(<u>A</u>)	ヘルプ

注:外部環境変数ファイルの変数は、青い文字で表示されます。内部環境変数 は、黒い文字で表示されます。

[環境] タブは、ユーザ定義環境変数に関して次の情報を提供します。

- ▶ [名前]: 各ユーザ定義変数の名前。
- ▶ [値]:各ユーザ定義変数に割り当てられた値。
- ▶ [タイプ]:各ユーザ定義変数のタイプ。内部または外部のいずれかのタイプ。内部環境変数は、その変数が定義されているテスト内においてのみ使用できます。

[環境] タブは、ユーザ定義環境変数に関して次のオプションを提供します。

オプション	詳細
+	新規環境変数を定義してリストに追加できます。詳細につい ては,次に示す「ユーザ定義環境変数の追加」を参照してく ださい。
×	選択した内部環境変数をリストから削除します。
	注 :環境変数の削除をいったん確定すると,[テストの設定] ダイアログ・ボックスで [キャンセル]をクリックしても, 変数は元に戻せません。
	選択した内部環境変数の編集および選択した外部環境変数の プロパティの表示を可能にします。詳細については,1211 ページ「ユーザ定義環境変数の表示および変更」を参照して ください。
[エクスポート]	ユーザ定義環境変数を外部 XML ファイルにエクスポートし, ほかのテストで使用できるようにします。エクスポートした 環境変数ファイルは,任意のテストで使用できます。詳細に ついては,1213ページ「ユーザ定義環境変数のエクスポー トと読み込み」を参照してください。
[変数と値を外部ファ イルからロードする]	テストで使用するよう指定した XML ファイルから,保存されている変数を読み込みます。詳細については,1213 ページ「ユーザ定義環境変数のエクスポートと読み込み」を参照してください。

ユーザ定義環境変数の追加

ユーザ定義内部変数の追加は, [テストの設定] ダイアログ・ボックスの [環 境] タブを使用して行います。内部環境変数は、それが定義されているテスト でだけ使用できます。

ユーザ定義内部環境変数を追加するには、次の手順を実行します。

- 1 [環境] タブの [変数のタイプ] ボックスで [ユーザ定義] を選択します。
- 2 [新規作成] ボタンをクリックします。[新規環境パラメータの追加] ダイアロ グ・ボックスが開きます。

新規環境パラメータの追加
名前(1):
(直(⊻):
0K キャンセル ヘルプ

- 3 変数の定義を次のように入力します。
 - ▶ [名前]:変数の名前を入力します。
 - ▶ [**値**]:変数の値を入力します。
- 4 [OK] をクリックして保存し, [新規環境パラメータの追加] ダイアログ・ ボックスを閉じます。変数が, [テストの設定] ダイアログ・ボックスの [環 境] タブのリストに追加されます (黒色で表示されます)。

ユーザ定義環境変数の表示および変更

ユーザ定義内部変数の編集は, [テストの設定] ダイアログ・ボックスの [環 境] タブを使用して行います。外部ユーザ定義変数のプロパティを表示するこ とも可能です。

内部変数および外部変数の値をコピーして、データ・テーブルなど、QuickTest のほかの場所で使用できます。



ユーザ定義内部環境変数を変更するには、次の手順を実行します。

1 [テストの設定] ダイアログ・ボックスの [環境] タブで、変更する内部変数を ダブルクリックするか、内部変数を選択して [環境変数の表示/編集] ボタンを クリックします。「環境パラメータの編集] ダイアログ・ボックスが開きます。

新規環境/	ラメータの追加
名前(<u>N</u>):	record
値(⊻):	999
OK	**>セル ヘルプ

- 2 変数の値を変更するには、[値] ボックスに異なる値を入力します。
- 3 変数の値をクリップボードにコピーするには,値のテキストを選択し,右ク リックして [**コピー**]を選択します。
- 4 [OK] をクリックして保存し, [環境パラメータの編集] ダイアログ・ボック スを閉じます。[テストの設定] ダイアログ・ボックスの [環境] タブ上で, 変数の値が更新されます。

ユーザ定義の外部環境変数を表示するには、次の手順を実行します。

1 [テストの設定] ダイアログ・ボックスの [環境] タブで、表示する外部変数 をダブルクリックするか、外部変数を選択して [環境変数の表示/編集] ボタ ンをクリックします。[環境パラメータの表示] ダイアログ・ボックスに、選 択した変数の詳細が表示されます。

環境パラメ	ータの表示
名前(N):	FirstName
値(⊻):	John
	閉じる ヘルプ



変数の値が複合的な値の場合(値全体を[**値**]ボックスに表示できない場合), 値の内容を表示するには,[**複雑な値の表示/編集**]ボタンをクリックします。

- 2 変数の値をクリップボードにコピーするには,値のテキストを選択し,右ク リックして [**コピー**]を選択します。
- 3 [閉じる] をクリックし, [環境パラメータの表示] ダイアログ・ボックスを閉 じます。

ユーザ定義環境変数のエクスポートと読み込み

ユーザ定義環境変数は、外部 XML ファイルにエクスポートすることにより、 ほかのテストで使用できます。エクスポートした環境変数は、ファイルから外 部ユーザ定義変数として読み込み、任意のテストで使用できます。

ファイルをファイル・システムに保存した場合,値はテストを実行するたびに 読み込まれます。ファイルを Quality Center プロジェクトに保存した場合,値は テストを最初に読み込むときに読み込まれます。テストの読み込み後に値が変 更されても,新しい値は次にテストを読み込むまでは使用されません。

ユーザ定義環境変数をエクスポートするには、次の手順を実行します。

- 1 [テストの設定] ダイアログ・ボックスの [環境] タブで、 [エクスポート] ボ タンをクリックします。ユーザ定義変数と値の現在のリストを XML ファイル にエクスポートできる [環境変数ファイルの保存] ダイアログ・ボックスが開 きます。
- 2 ファイルの保存先フォルダを選択します。QuickTest が現在 Quality Center に接続されていれば、[Quality Center] ボタンをクリックしてファイルを Quality Center に保存するか、[ファイル システム] ボタンをクリックしてファイルをファイルをファイル・システムに保存できます。
- 3 [ファイル名] ボックスにファイルの名前を入力します。
- 4 [保存] をクリックして、ファイルを保存します。

注: リソースへのパスを保存する場合は,QuickTestによって,[オプション] ダイアログ・ボックスの[フォルダ]タブ([**ツール**] > [**オプション**] > [**フォルダ**])に,パスまたはパスの一部が存在するかどうかが確認されます。 パスが存在する場合は,入力したパスの相対的な部分だけを使用してパスを定 義するように求められます。パスが存在しない場合は,リソースのパスを [フォルダ] タブに追加して,相対的なパスを定義するように求められます。

詳細については,340ページ「QuickTest での相対パスの使用」を参照してください。

ユーザ定義外部環境変数ファイルから変数を読み込むには,次の手順を実行し ます。

- 1 [テストの設定] ダイアログ・ボックスの [環境] タブで, [変数と値を外部 ファイルからロードする] を選択します。
- 2 [ファイル] ボックスで、ファイル名を入力するか、参照ボタンをクリックし てユーザ定義外部変数ファイルを探します。QuickTest が現在 Quality Center に 接続されていれば、[開く] ダイアログ・ボックスの [Quality Center] ボタン をクリックしてファイルを Quality Center の中で探すか、[ファイル システム] ボタンをクリックしてファイルをファイル・システムの中で探せます。

選択したファイルから読み込まれた環境変数は,[テストの設定]ダイアログ・ ボックスの[環境]タブ上に青色で表示されます。

注:環境変数ファイルは相対パスで指定できます。QuickTest は [オプション] ダイアログ・ボックスの [フォルダ] タブに指定されているフォルダ内でその ファイルを検索します。詳細については,1169ページ「テストのフォルダ・オ プションの設定」および 340ページ「QuickTest での相対パスの使用」を参照し てください。

組み込み環境変数とユーザ定義変数の詳細について、および、外部ユーザ定義 環境変数ファイルの作成方法については、661ページ「環境変数パラメータの 使用」を参照してください。

テストのための回復シナリオ設定の定義

[テストの設定] ダイアログ・ボックス([**ファイル**] > [設定])の[回復] タブには、現在のテストに関連付けられているすべての回復のシナリオのリス トが表示されます。また、テストに関連付ける回復シナリオの追加、テストか らのシナリオの削除、実行セッションに対する回復シナリオ適用順序の変更、 および、各シナリオの表示専用サマリの表示が可能です。

対象のテストに対して,特定のシナリオの有効と無効を切り替えることも,回 復メカニズム全体の有効と無効を切り替えることもできます。

テストに対する作業中には,現在のシナリオのリストを以降の新規テストすべてにおいて使用するよう指定することもできます。

回復シナリオの詳細については,第44章「回復シナリオの定義と使用」を参照してください。

テストの設定	×
「プロパティ」実行 「リソース」	パラメータ 環境 Web 回復
シナリオ	+ × 1 ↓ 🖻
シナリオ名	7711
Pop-up handler	C:\Program Files\My Tests\popup.qrs
ActiveX Unsafe	[QualityCenter] Subject\Web.grs
Authorization Failed	[QualityCenter] Subject\Web.grs
File Download	[QualityCenter] Subject\Web.grs
Internet Redirect	[QualityCenter] Subject\Web.grs
✓\' Network Error	[QualityCenter] Subject/Web.grs
2,+0+===2+	
ンチリカ記述	
	<u> </u>
	-
J	
回復シナリオのアクティブ化:	エラー発生時 <u>▼</u>
177.00 (Tr)	
_標準値に設定也)	
(C.	OK キャンセル 適用(A) ヘルプ

[回復] タブには、次のオプションがあります。

オプション領域	詳細
[シナリオ]	テストに関連付けられている各回復シナリオの名前と回復ファイ ルのパスが表示されます。リスト内のシナリオの追加,削除,優 先順位設定が可能なほか,選択したファイルのパスを編集できま す。詳細については,次に示す第41章「関連回復シナリオの指 定」を参照してください。
[シナリオ記述]	[シナリオ] ボックスで選択したシナリオの説明テキストが表示さ れます。
[回復シナリオ のアクティブ 化]	 関連付けられているシナリオを次のように実行するかどうか確認 するよう QuickTest を設定します。 ▶ [各ステップごと]: 回復メカニズムはステップごとに起動され ます。 ▶ [エラー発生時]: 回復メカニズムはエラーの戻り値を返すス テップの後にのみ起動されます。 ▶ [なし]: 回復メカニズムは無効にされます。 注: [各ステップごと] を選択すると,実行セッション中のパ フォーマンスが低下する場合があります。
[標準値に設定]	回復シナリオ・ファイルの現在のリストを,以降の新規テストに 関連付ける標準設定のリストとして設定します。 注:[標準値に設定]オプションは,テストが対象の場合にのみ使 用できます。このオプションは,対象テストの設定が全テストの 標準設定値と異なる場合に有効になります。

注: テストで使用する回復ファイルがファイル・システムに格納されている場 合,そのテストをほかのユーザまたは HP 製品がほかのコンピュータ上で実行 できるようにするには,回復ファイルのパスを相対パスとして指定する必要が あります (パスを一度クリックするとパスが強調表示され,その状態で再度ク リックすると編集モードになります)。以後はほかのユーザも,QuickTest で相 対パスを参照するドライブ文字とフォルダを [オプション] ダイアログ・ボッ クス ([ツール] > [オプション])の [フォルダ] タブで指定することによ り,同じテストを実行できるようになります。詳細については,1169ページ 「テストのフォルダ・オプションの設定」および 340ページ「QuickTest での相 対パスの使用」を参照してください。

関連回復シナリオの指定

各シナリオの横のチェック・ボックスを選択またはクリアすることで,現在の テストに対して当該シナリオを有効または無効にできます。

また、回復シナリオ・ファイル・パスを編集するには、パスを一度クリックし てパスを強調表示し、その状態で再度クリックして編集モードに入ります。た とえば、絶対ファイル・パスをファイルの相対ファイル・パスに変更したい場 合があります。回復シナリオ・ファイルのパスを変更した場合は、テストを実 行する前に、回復シナリオが新しいパスに存在することを確認する必要があり ます。

シナリオは、次のアイコンによって示されます。

アイ コン	祥維
V	対象回復シナリオが,実行セッション中に開いているアプリケーションの特 定のポップアップ・ウィンドウをきっかけとして起動されることを示しま す。
V	対象回復シナリオが,アプリケーション内のオブジェクトのプロパティ値が 特定の値に一致したときに起動されることを示します。
V	対象回復シナリオが,テスト内で正常に実行されないステップがあった場合 に起動されることを示します。

アイ コン	詳細
Vě	対象回復シナリオが,実行セッション中に指定のアプリケーションが失敗し たときに起動されることを示します。
	対象回復シナリオが、そのテストから利用できなくなっていることを示しま す。理由としては、回復ファイルの名前の変更や場所の移動が原因で QuickTest からアクセスできなくなったことなどが考えられます。実行セッ ション中に、関連付けられている回復ファイルが利用できないと、メッセー ジがテスト結果に表示されます。

注:QuickTest にあらかじめ用意されている標準設定の回復シナリオは, QuickTest のインストール・フォルダにインストールされています。[回復] タ ブ内において標準設定の回復シナリオを指定するパスは,パスの中で環境変数 (%ProductDir%)を使用します。そうすることにより,当該回復シナリオに 対応するテストを別のコンピュータや別のHP製品で実行した場合にも, QuickTest でシナリオを発見できるようになります。これら標準設定の回復シナ リオ・ファイルのパスを変更したり,この環境変数をほかの目的に使用したり してはなりません。

次の回復シナリオ・ファイル制御ボタンを使用して,テストに関連付けられて いる回復シナリオ・ファイルの追加,削除,優先順位設定が可能です。

オプショ ン	祥希
+	[回復シナリオの追加] ダイアログ・ボックスが開き,テストに1つ以上の回復シナリオを関連付けることができます。詳細については,1280 ページ「テストへの回復シナリオの追加」を参照してください。
×	選択した回復シナリオをテストから削除します。
1	選択したシナリオをリストの中で上に移動することで優先順位を上げます。

オプショ ン	詳細
4	選択したシナリオをリストの中で下に移動することで優先順位を下げます。
	選択した回復シナリオのプロパティのサマリを表示専用の形式で表示します。詳細については、1283ページ「回復シナリオのプロパティの表示」を参照してください。
第 42 章

実行セッション中のテスト・オプションの設定

実行セッション中にテスト・オプションを設定したり、取得したりすることに よって、QuickTest でのテストの記録と実行の方法を制御できます。

本章では、次の項目について説明します。

- ▶ 実行セッション中のテスト・オプションの設定について(1221ページ)
- ▶ テスト・オプションの設定(1222ページ)
- ▶ テスト・オプションの取得(1224ページ)
- ▶ テスト実行の制御(1225ページ)
- ▶ テスト実行設定の追加と削除(1225ページ)

実行セッション中のテスト・オプションの設定について

QuickTest テスト・オプションは、テストの記録と実行の方法に影響します。た とえば、Webページを読み込むときに、URL アドレスが見つからないと判断す るまでに QuickTest で許容される最長時間を設定できます。

テスト・オプションの値を実行セッション中に設定および取得するには,エキ スパート・ビューで Setting オブジェクトを使います。エキスパート・ビュー での作業の詳細については,第26章「エキスパート・ビューおよび関数ライ ブラリ・ウィンドウを使用した作業」を参照してください。

Setting オブジェクトを使ってテスト・オプションを取得および設定することで、QuickTest でのテストの実行方法を制御できます。

多くのテスト・オプションは, [オプション] ダイアログ・ボックス (グロー バルなテスト・オプション) および [テストの設定] ダイアログ・ボックス (テスト固有の設定) を使っても設定できます。詳細については, 第40章「グ ローバル・テスト・オプションの設定」および第41章「個別のテストのオプ ション設定」を参照してください。

本章では、テスト・スクリプト内から Setting オブジェクトを使って設定でき る QuickTest のテスト・オプションの一部について説明します。Setting オブ ジェクトで使用可能なすべてのメソッドとプロパティの詳細については、 『HP QuickTest Professional Object Model Reference』(英語版)の「Utility」の 項を参照してください。

注: QuickTest オプションは、ほかの操作と同様、自動プログラムを使って外部 アプリケーションから制御することもできます。詳細については、1311 ページ 「QuickTest 操作のオートメーション」、または『QuickTest Automation』(英語 版)([ヘルプ] > [QuickTest Professional ヘルプ] > [QuickTest Advanced References] > [QuickTest Automation])を参照してください。

テスト・オプションの設定

Setting オブジェクトを使って、テスト・オプションの値をテスト・スクリプト 内から設定できます。オプションを設定するには、次の構文を使います。

Setting (テスト・オプション) = 新変数

オプションの中にはグローバルなものと、現在のテストにのみ影響するものが あります。Setting オブジェクトを使用してテスト・オプションを設定すると、 もう一度変更するか、または現在の QuickTest セッションが終了まで設定が有 効です。Setting オブジェクトを使い、特定のテストの特定の部分の設定だけを 変更することもできます。詳細については、1225 ページ「テスト実行の制御」 を参照してください。 Setting オブジェクトを使用して設定できるテスト・オプションの一部は, [オ プション] ダイアログ・ボックス (グローバル・オプション) または [テスト の設定] ダイアログ・ボックス (テスト固有の設定) でも利用できます。 Setting オブジェクトを使用してこれらのオプションを設定すると, 変更は関連 するダイアログ・ボックスに反映されます。ほかのテスト設定は, 関連するダ イアログ・ボックスまたは Setting オブジェクトのうち, いずれかの方法を使 用してのみアクセスできます。

例:[オプション]ダイアログ・ボックスに反映されるオプションを設定する ための Setting オブジェクトの使用

Web Add-in を読み込んだ状態で、次のステートメントを実行します。

Setting("AutomaticLinkRun")=1

QuickTest により, テストの自動的に作成されたチェックポイントが無効になり ます。現在の QuickTest セッションでは, この設定は, 別の Setting ステートメ ントを使うか, または [詳細 Web オプション] ダイアログ・ボックス ([ツー ル] > [オプション] > [Web] タブを選択して [詳細設定] をクリック) で [テストまたはコンポーネントの実行時に自動チェックポイントを無視する] チェックボックスをクリアして, 設定を変更するまで有効です。

例:[テストの設定] ダイアログ・ボックスに反映されるオプションを設定す るための Setting オブジェクトの使用

次のステートメントを実行します。

Setting("WebTimeOut")=50000

QuickTest によって, テストのステップを実行する前に Web ページの読み込み が終わるまで待機する時間の上限が, 自動的に 50000 ミリ秒に変更されます。 この設定は, 現在の QuickTest セッション中に別の Setting ステートメントを使 うか, [テストの設定] ダイアログ・ボックスの [Web] タブで [**ブラウザナ ビゲーションのタイムアウト**] オプションを使って, 設定を変更するまで有効 です。 **注**: Setting オブジェクトを使って行った変更が [オプション] ダイアログ・ ボックスと [テストの設定] ダイアログ・ボックスに反映されている場合で も、同じダイアログ・ボックスでほかの変更を手作業で行って [**適用**] または [**OK**] をクリックしなければ、これらの変更は QuickTest を閉じたときに保存 されません。

テスト・オプションの取得

Setting オブジェクトを使って、テスト・オプションの現在の値を取得すること もできます。

変数に値を格納するには、次の構文を使います。

新変数 = Setting (テスト・オプション)

メッセージ・ボックスに値を表示するには、次の構文を使います。

MsgBox (Setting (テスト・オプション))

例を次に示します。

LinkCheckSet = Setting("AutomaticLinkRun")

これは, AutomaticLinkRun 設定の現在の値を, ユーザ定義変数の LinkCheckSet に代入します。

設定を取得する対象にできるその他のテスト・オプションの例については, 1222 ページ「テスト・オプションの設定」を参照してください。

テスト実行の制御

Setting オブジェクトの取得機能と設定機能を組み合わせることで、グローバル 設定を変更せずに、実行セッションを制御できます。たとえば、 DefaultTimeOut テスト・オプションを、ある特定の Web ページ上にあるオブ ジェクトについてだけ 5 秒に変更するには、テスト・スクリプトの中でその Web ページを開く処理の後に、次のステートメントを挿入します。

' DefaultTimeOut テスト・オプションの元の値を保存 old_delay = Setting ("DefaultTimeOut")

' DefaultTimeOut テスト・オプションの一時値を設定 Setting("DefaultTimeOut")= 5000

Web ページの最後で DefaultTimeOut テスト・オプションを元の値に戻すには, スクリプトの中で次ページにリンクする処理の直前に,次のステートメントを 挿入します。

' DefaultTimeOut テスト・オプションを元の値に戻す Setting("DefaultTimeOut")=old_delay

テスト実行設定の追加と削除

グローバルな設定や固有の設定だけでなく、実行環境の設定も追加、変更、削 除できます。これらの設定は、実行セッション中にのみ適用されます。

新しい実行環境の設定を追加するには、次の構文を使います。

Setting.Add " テスト・オプション ", " 値 "

たとえば,現在のテスト実施者の名前を表示し,その名前をメッセージ・ボッ クスに表示する設定を作成できます。

Setting.Add "Tester Name", "Mark Train" MsgBox Setting("Tester Name") **ヒント**: Setting.Add ステートメントを使用するときに,既存の設定オプション を追加しようとするとエラーが発生します。このエラーを防ぐには,先に Setting.Exists ステートメントを使う必要があります。すべての Setting メソッ ドの詳細については,『**HP QuickTest Professional Object Model Reference**』(英 語版)を参照してください。

すでに初期化されている実行環境設定を変更するには、標準の設定オプション を設定する場合と同じ構文を使います。

Setting (テスト・オプション) = 新しい値

例を次に示します。

Setting("Tester Name")="Alice Wonderlin"

ユーザ定義の実行環境の設定を削除するには、次の構文を使います。

Setting.Remove (テスト・オプション)

例を次に示します。

Setting.Remove ("Tester Name")

ヒント: Setting.Remove ステートメントを使用するときに,存在しない設定オ プションを削除しようとするとエラーが発生します。このエラーを防ぐには, 先に Setting.Exists ステートメントを使う必要があります。すべての Setting メ ソッドの詳細については,『**HP QuickTest Professional Object Model Reference』** (英語版)を参照してください。

第X部

高度なテスト機能を使用した作業

第 43 章

仮想オブジェクトの学習

アプリケーションの任意の領域を「**仮想オブジェクト**」として定義することに より,QuickTest でこの領域がオブジェクトとして認識されるように設定できま す。仮想オブジェクトを利用すると,QuickTest が通常ならば認識しないオブ ジェクトを対象とするテストの作成と実行が行えます。

本章では、次の項目について説明します。

- ▶ 仮想オブジェクトの学習について(1229ページ)
- ▶ 仮想オブジェクトについて(1231ページ)
- ▶ 仮想オブジェクト・マネージャについて(1232ページ)
- ▶ 仮想オブジェクトの定義(1233ページ)
- ▶ 仮想オブジェクト定義の削除と無効化(1238ページ)

仮想オブジェクトの学習について

アプリケーションには、標準のオブジェクトと同様に動作するにもかかわら ず、QuickTest が認識できないオブジェクトが含まれることがあります。これら のオブジェクトを仮想オブジェクトとして定義し、ボタンやチェック・ボック スなどの標準クラスに割り当てることができます。これにより、実行セッショ ン中、QuickTest によって、この仮想オブジェクトに対するユーザのアクション がエミュレートされます。テスト結果には、仮想オブジェクトは標準クラスの オブジェクトとして表示されます。

たとえば、ユーザがクリックするビットマップが含まれる Web ページをテスト するとします。このビットマップには、それぞれ異なるリンク・ページを表示 するいくつかのハイパーリンク領域があります。テストを作成するときに、 Web サイトでは、クリックしたビットマップの座標が検出され、リンク・ペー ジが表示されます。 実行セッション中に QuickTest で必要な座標をクリックできるようにするため, その座標が含まれるビットマップ領域の仮想オブジェクトを定義し,これをボ タン・クラスに割り当てます。テストを実行すると,QuickTest によって,仮想 オブジェクトとして定義した領域のビットマップがクリックされ,対応するリ ンク・ページが Web サイトに表示されます。

仮想オブジェクトの定義は、仮想オブジェクト・ウィザードで行います ([ツール] > [仮想オブジェクト] > [新規仮想オブジェクト])。このウィ ザードでは、仮想オブジェクトの割り当ての対象にする標準のオブジェクト・ クラスを選択するよう指示されます。その後、十字形のポインタで、仮想オブ ジェクトの境界を指定します。次に、この仮想オブジェクトの親とする、テス ト・オブジェクトを選択します。最後に、仮想オブジェクトの名前とコレク ションを指定します。仮想オブジェクトのコレクションとは、仮想オブジェク ト・マネージャに格納されている仮想オブジェクトの集合に名前を付けたもの です。

注: QuickTest では,アナログや低レベル記録用の仮想オブジェクトはサポート されていません。低レベルの記録の詳細については,1407ページ「テストの作 成」を参照してください。

仮想オブジェクトについて

QuickTest では、境界線に基づいて仮想オブジェクトが識別されます。オブジェ クトの境界を指定することで、Webページまたはアプリケーション・ウィンド ウにおけるオブジェクトのサイズと位置を指定します。仮想オブジェクトの親 としてテスト・オブジェクトを割り当てると、仮想オブジェクトの境界座標は その親オブジェクトを基準とする相対座標と認識されます。テストを記録する ときに、QuickTest によって、親オブジェクトに含まれる仮想オブジェクトが認 識され、実行セッション中にこのオブジェクトを識別できるようオブジェク ト・リポジトリにテスト・オブジェクトとして追加されます。QuickTest では、 仮想オブジェクトは、オブジェクト・リポジトリに手作業で追加するときにも テスト・オブジェクトとして認識されます。

注:実行セッション中は、アプリケーション・ウィンドウは記録時と同じサイ ズで同じ位置になければなりません。それ以外の場合、親オブジェクトに対す る仮想オブジェクトとの相対座標が記録時と異なり、それが実行セッションの 正常な実行に影響を与える可能性があります。

仮想オブジェクト・マネージャから削除しなくても、仮想オブジェクトが認識 されないようにすることも可能です。詳細については、1238ページ「仮想オブ ジェクト定義の削除と無効化」を参照してください。

注:

- ▶ 仮想オブジェクトは、テストの記録時と実行時にのみ使用できます。仮想オ ブジェクトへのチェックポイントの挿入や、オブジェクト・スパイによる仮 想オブジェクトのプロパティの表示を行うことはできません。
- ➤ マークを付けた仮想オブジェクトを対象に ActiveScreen で操作を実行するには、そのオブジェクトをあらかじめ記録しておく必要があります。それにより、仮想オブジェクトのプロパティが、オブジェクト・リポジトリのテスト・オブジェクト記述に保存されます。まだ記録していない仮想オブジェクトを対象に ActiveScreen で操作を実行すると、仮想オブジェクトは標準オブジェクトとして扱われます。

仮想オブジェクト・マネージャについて

仮想オブジェクト・マネージャには、お使いのコンピュータで定義されている すべての仮想オブジェクト・コレクションが含まれます。仮想オブジェクト・ マネージャで、仮想オブジェクトとコレクションの定義と削除が行えます。

仮想オブジェクト マネージャ		×
QuickTest Professional	利用可能な仮想オブジェクトのコレクション: ■ ŵ VoCollection0	新規作成(N) 削除(D) ヘルブ(H) 閉じる(C)

[利用可能な仮想オブジェクトのコレクション]:お使いのコンピュータで定義 されている仮想オブジェクト・コレクションと,それぞれのコレクションに含 まれる仮想オブジェクトが表示されます。コレクションの横の[+]と[-]記 号を使用して,コレクションに定義されている仮想オブジェクトを表示したり 非表示にしたりできます。

[新規作成]: [仮想オブジェクトウィザード]を開きます。このウィザードは, 新規あるいは既存のコレクションに新規仮想オブジェクトを定義する手順を示 します。詳細については,1233ページ「仮想オブジェクトの定義」を参照して ください。

[**削除**]:選択した仮想オブジェクトまたは仮想オブジェクト・コレクションを 削除します。詳細については、1238ページ「仮想オブジェクト定義の削除と無 効化」を参照してください。 **注**: [仮想オブジェクトマネージャ] に表示される仮想オブジェクト・コレク ションは、お使いのコンピュータに格納されますが、仮想オブジェクト・ス テップを含むテストと一緒には保存されません。つまり、テスト・ステップの 中で仮想オブジェクトを使用した場合、オブジェクトが実行セッション中に認 識されるのは、適切な仮想オブジェクト定義を含むコンピュータで実行された 場合のみとなります。仮想オブジェクト・コレクション定義を別のコンピュー タにコピーするには

< QuickTest インストール・フォルダ> ¥dat¥VoTemplate フォルダの内容 (またはこのフォルダ内の個々の.vot コレクション・ファイル) を, コピー先 コンピュータの同じフォルダにコピーします。

仮想オブジェクトの定義

仮想オブジェクト・ウィザードを使って、仮想オブジェクトを標準のオブジェ クト・クラスに割り当てたり、仮想オブジェクトの境界と親を指定したり、名 前を割り当てたりできます。また、コレクションに割り当てることによって、 仮想オブジェクトを論理的なグループにまとめることができます。

仮想オブジェクトは、QuickTest Professional によって Click または DblClick メ ソッドが記録されるオブジェクトに対してのみ定義できます。それ以外の仮想 オブジェクトは無視されます。たとえば、WinList オブジェクトを仮想オブ ジェクトとして定義しようとすると、Select 操作は記録されますが、仮想オブ ジェクトは無視されます。 仮想オブジェクトを定義するには、次の手順を実行します。

- 1 QuickTest を記録モード以外のモードで起動している状態で、アプリケーション を開いて、仮想オブジェクトとして定義する領域を含むオブジェクトを表示し ます。
- **2** QuickTest で, [ツール] > [仮想オブジェクト] > [新規仮想オブジェクト] を選択します。あるいは, [仮想オブジェクトマネージャ] で [新規作成] を クリックします。仮想オブジェクト・ウィザードが開きます。

仮想オブジェクト・ウィザード	医相关 하고 나는 나는 방어
QuickTest Professional	仮想オフシェクト ウイザー ドに ようこそ! 記録作業を正しく行うため、テストするアプリケーションを 標準オブジェクトとしていかに記職するか Quick Test に 学習させることができます。 次の場合このウィザードを利用します。 1. 仮想オブジェクトを標準クラスに書的当てる 2. 仮想オブジェクトの境界と親オブジェクトを定義する 3. 仮想オブジェクトの名前を指定する 4. 仮想オブジェクトをコレクションに書の当てる
	< 戻る(四) (次へい) キャンセル ヘルプ

[**次へ**] をクリックします。

3 仮想オブジェクトの割り当て先の標準クラスを選択します。

仮想オブジェクト ウィザード		×
WickTest Professional	標準クラスへの割り当て 仮想オブジェクトに最も相似する標準クラスを選択します。 クラス心: Check box	
	< 戻る(B) 次へ(N) > キャンセル ヘルプ	

list クラスを選択した場合,仮想オブジェクトの行数を指定します。table クラスの場合,行数とカラム数を選択します。[次へ]をクリックします。

4 [オブジェクトのマーク付け] をクリックします。

仮想オブジェクト ウィザード	仮想オブジェクトのマーク付け 1. [オブジェクトのマーク付け] ボタンをクリックします。 この画面は最小化されます) 2. 十字ボインタを使って仮想オブジェクトの輪郭を取ります。 スクロールバーやラベルは仮想オブジェクトに含めないでください。 オブジェクトのマーク付け(M)	×
Professional	幅(W): 98 高さ(H): 66 < 戻る(B) (次へい)ンコートキャンセル ヘルブ	

QuickTest ウィンドウと仮想オブジェクト・ウィザードが最小化されます。十字 形のポインタで、仮想オブジェクトの領域を指定します。マウスの左ボタンを 押しながら矢印キーを使用すると、十字形ポインタで定義した領域を微調整で きます。

[**次へ**] をクリックします。

注:仮想オブジェクトをマーク付けするときは、アプリケーションのほかの仮 想オブジェクトと重ならないようにします。仮想オブジェクトが別の仮想オブ ジェクトと重なっていると、その仮想オブジェクトに対するテストが正しく記 録または実行されないことがあります。

5 仮想オブジェクトの親として割り当てるオブジェクトをオブジェクト・ツリー の中でクリックします。

仮想オブジェクト ウィザード	× オブジェクトの構成 仮想オブジェクトの親を指定してください。 Page : Welcome: Mercury Tours Page : Welcome: Mercury Tours Page : berner? 選択した親オブジェクト: Image : banner2
Professional	選択した親オブジェクト: [Image : banner2 -オブジェクト認識のレベル ○ 親階層の全体(H)
	< 戻る(B) 次へ(N) > キャンセル ヘルプ

仮想オブジェクトの輪郭の座標は、選択した親オブジェクトを基準とする相対 座標です。

- 6 [オブジェクト認識のレベル] ボックスで,仮想オブジェクトの識別方法と割 り当て方法を選択します。
 - ➤ QuickTest で、仮想オブジェクトのすべての出現を識別させる場合、[親の み]を選択します。親階層全体ではなく、直接の親だけを基準にして仮想オ ブジェクトが識別されます。たとえば、Browser("A").Page("B").Image("C") を使用して仮想オブジェクトが定義されている場合、階層が、 Browser("X").Page("Y").Image("C")であっても仮想オブジェクトが識別され ます。
 - ➤ QuickTest で、仮想オブジェクトの特定1回の出現を識別させる場合、「親階層の全体]を選択します。QuickTest では、親階層が完全に一致している場合のみ仮想オブジェクトが識別されます。たとえば、Browser("A").Page("B").Image("C")を使用して仮想オブジェクトが定義されている場合、階層がBrowser("X").Page("B").Image("C")に変わるとこの仮想オブジェクトは識別されません。

[**次へ**] をクリックします。

7 仮想オブジェクトの名前とコレクションを指定します。コレクションのリスト から選択するか, [コレクション名] ボックスに新しい名前を入力して新しい コレクションを作成します。

仮想オブジェクト ウィザード	×
	仮想オブジェクトの保存
	QuickTest は定義した仮想オブジェクトを認識するように設 定されました。
(h)	仮想オブジェクトの名前を入力してください。 名前(M): Check box
QuickTest Professional	仮想オブジェクトを割り当てたコレクションの名前を指定してく ださい。 コレクション名(©): VoCollection0 マ
	別の仮想オブジェクトを定義しますか? ○ はい(Y) ○ いいえ(Q)
	〈 戻る(B) 〒7 キャンセル ヘルプ

8 次の手順のいずれかを実行します。

- ▶ 仮想オブジェクトを仮想オブジェクト・マネージャに追加してウィザードを 閉じるには、[いいえ]を選択し、[完了]をクリックします。
- ▶ 仮想オブジェクトを仮想オブジェクト・マネージャに追加し、さらに別の仮 想オブジェクトを定義するには、[はい]を選択し、[次へ]をクリックしま す。ウィザードが[標準クラスへの割り当て]画面に戻り、次の仮想オブ ジェクトを定義できます。

仮想オブジェクト定義の削除と無効化

仮想オブジェクトをテストに含めないようにするには、記録された仮想オブ ジェクトをテストから削除するか、記録時に認識されないように設定します。

仮想オブジェクトを削除するには、次の手順を実行します。

1 [ツール] > [仮想オブジェクト] > [仮想ユーザ オブジェクト マネージャ] を選択します。仮想オブジェクト・マネージャが起動します。

仮想オブジェクト マネージャ		×
QuickTest Professional	利用可能な仮想オブジェクトのコレクション:	新規作成(N) 削除(D) ヘルプ(H) 閉じる(C)

2 利用可能な仮想オブジェクトのコレクションのリストで、コレクションの横にあるプラス記号をクリックし、削除する仮想オブジェクトを表示します。仮想オブジェクトを選択し、「削除」をクリックします。

コレクション全体を削除するには、コレクションを選択し、[**削除**]をクリックします。

3 [閉じる] をクリックします。

ヒント:新しい仮想オブジェクトを定義するには、仮想オブジェクト・マネージャで[新規作成]をクリックし、仮想オブジェクト・ウィザードを起動します。

記録中に仮想オブジェクトが認識されないようにするには、次の手順を実行します。

- [ツール] > [オプション] を選択するか, [オプション] ツールバー・ボタン をクリックします。[オプション] ダイアログ・ボックスが開きます。
- 2 [一般] タブで, [記録時の仮想オブジェクト認識を無効にする] チェック・ ボックスをオンにします。
- 3 [OK] をクリックします。

2

注:記録中に QuickTest で仮想オブジェクトが認識されるようにするには, [オ プション]ダイアログ・ボックスの [一般] タブの [記録時の仮想オブジェク ト認識を無効にする] チェック・ボックスがクリアされていることを確認しま す。詳細については, 1164ページ「テストの一般オプションの設定」を参照し てください。

第 44 章

回復シナリオの定義と使用

実行セッション中,テスト環境で生じる予期しないイベントやエラーから回復 するよう QuickTest に指示できます。

本章では、次の項目について説明します。

- ▶ 回復シナリオの定義と使用について(1242ページ)
- ▶ 回復シナリオを使用するタイミングの決定(1244ページ)
- ▶ 回復シナリオの定義(1245ページ)
- ▶ 回復シナリオ・ウィザードについて(1249ページ)
- ▶ 回復シナリオの管理(1275ページ)
- ▶ テストへの回復シナリオの関連付け(1280ページ)
- ▶ プログラムによる回復メカニズムの制御(1286ページ)

回復シナリオの定義と使用について

実行セッション中に、予期しないイベント、エラー、およびアプリケーショ ン・クラッシュが発生すると、実行セッションが妨げられ、正しいテスト結果 が得られない可能性があります。これは、テストを無人で実行する場合に特に 問題になります。回復に必要な操作を実行するまで、テストが一時停止状態に なるからです。このような状況を処理するために、QuickTest では回復シナリオ を作成して特定のテストに関連付けることができます。トリガ・イベントが発 生すると、[回復]シナリオによって特定の回復操作が開始されます。どのよ うなときに回復シナリオを使用するかについては、1244ページ「回復シナリオ を使用するタイミングの決定」を参照してください。

回復シナリオ・マネージャでは、「回復シナリオ」を定義するプロセスを案内 するウィザードが使用できます。回復シナリオには、予期しないイベントの定 義と、実行セッションを回復するために必要な操作が含まれています。たとえ ば、「Printer out of paper」というメッセージを検出し、[OK] ボタンをク リックしてメッセージを閉じることによって実行セッションを回復し、テスト を続行するよう QuickTest に指示できます。

回復シナリオは、次の要素で構成されています。

- ▶ トリガ・イベント:実行セッションを中断するイベントです。たとえば、画面 上にポップアップ表示されるウィンドウや、QuickTestの実行エラーなどです。
- **回復操作**:トリガ・イベントによって実行セッションを中断された後も QuickTest でテストを実行し続けることができるようにする操作です。たとえ ば、ポップアップ・ウィンドウの [OK] ボタンをクリックすることや、 Microsoft Windows の再起動などです。
- ▶ 回復後のテスト実行のオプション:回復操作を実行してからの QuickTest の継続方法の指示,および回復操作を実行した場合はテストのどのポイントから QuickTest を継続するかの指示です。たとえば、テストを初めからやり直したり、完全に1つのステップをスキップしたり、テストの次のステップから続けたりすることができます。

回復シナリオは回復シナリオ・ファイルに保存されます。回復シナリオ・ファ イルは回復シナリオの論理的な集合で,特定の独自の要件に従ってグループ化 されています。 実行セッション中に回復シナリオを実行するよう QuickTest に指示するには, まずそのテストに回復シナリオを関連付ける必要があります。テストに関連付 けることができる回復シナリオの数に制限はありません。テストに関連付ける シナリオに優先順位を付けて,必要な順序でトリガ・イベントを認識,処理さ せることができます。詳細については,1280ページ「テストへの回復シナリオ の追加」を参照してください。

回復シナリオを定義した対象のテストを実行して,エラーが発生した場合は, QuickTest によって,エラーの原因である定義済みのトリガ・イベントが検索さ れます。トリガ・イベントが発生した場合,QuickTest によって,対応する回復 操作および回復後操作が実行されます。

テストに Recovery ステートメントを挿入することで,実行セッション中に回復 シナリオを制御し,呼び出すこともできます。詳細については,1286ページ 「プログラムによる回復メカニズムの制御」を参照してください。

注:[テストの設定]ダイアログ・ボックスの[回復]タブにある[**回復シナ** リオのアクティブ化]ボックスで「エラー発生時」を選択した場合,回復メカ ニズムでは、テストの最後のステップで発生するトリガは処理されません。こ のオプションを選択し、かつテストの最後のステップで発生する可能性のある 予期しないイベントやエラーから回復する必要がある場合は、テストの最後に さらにステップを追加することで、予期しないイベントやエラーから回復でき ます。

回復シナリオを使用するタイミングの決定

回復シナリオは、前もって予測できないイベントでの使用、またはテストの特定のステップと同期させることのできないイベントのみを対象としています。 たとえば、プリンタ・エラーを処理するために回復シナリオを定義できます。 そうすれば、実行セッション中にプリンタ・エラーが発生したときに、回復シ ナリオが QuickTest に対して [プリンタ エラー] メッセージ・ボックスの標準 のボタンをクリックするよう指示できます。

この例で回復シナリオを使用するのは、テストでは、このようなエラーを直接 処理できないからです。なぜなら、どの時点でネットワークによってプリン タ・エラーが返されるか知ることができないためです。テストにおいてファイ ルをプリンタに送信するステップの直後に If ステートメントを追加することに よってこのイベントを処理しようとしても、ネットワークによって実際のプリ ンタ・エラーが返される前に、テストは数ステップ進む場合があります。

イベントがテストの特定のポイントで発生することが予測できる場合は、回復 シナリオを使用するのではなく、If ステートメントなどのステップをまたはオ プション・ステップに追加追加して、テストから直接イベントを処理すること をお勧めします。たとえば、実行セッション中に [**保存**] ボタンがクリックさ れるとファイルの上書きメッセージ・ボックスが表示されることがあるという 場合は、このイベントを、メッセージ・ボックスが開くと [OK] をクリック する If ステートメントを追加する、あるいはメッセージ・ボックスで [OK] をクリックするオプション・ステップを追加することで処理できます。

テストから直接イベントを処理すると、回復シナリオよりもより明確にエラー 処理が行えます。これは回復シナリオが一般的な予測しないイベントのセット を処理するよう設計されているためです。またテストから直接イベントを処理 すれば、調整処理のタイミングを制御でき、最小の労力で最大の効果を挙げる ことができます。標準では、回復シナリオ操作が開始されるのは、ステップに よってエラーが返された後にかぎられます。これは、最初にエラーを引き起こ したステップの数ステップ後に発生する可能性があります。別の方法として は、ステップごとにトリガ・イベントを確認することも考えられますが、パ フォーマンスが低下する可能性があります。このため、予測可能なエラーはテ ストで直接処理することをお勧めします。

オプション・ステップの詳細については,946 ページ「オプション・ステップ の使用」を参照してください。If ステートメントなど,プログラミング・ス テートメントの挿入の詳細については,第25章「プログラミング・ロジック を含むステップの追加」を参照してください。

回復シナリオの定義

回復シナリオは、回復シナリオ・ウィザード([回復シナリオマネージャ]ダ イアログ・ボックスからアクセスします)を使用して作成します。回復シナリ オ・ウィザードでは、回復シナリオの各段階を定義するプロセスを案内しま す。作成した回復シナリオは、回復ファイルに保存します。回復ファイルは、 複数の回復シナリオをまとめて整理、格納するのに便利です。

[回復シナリオマネージャ]ダイアログ・ボックスを使用すると、任意の回復 ファイルを選択してそのファイルに格納されているすべての回復シナリオを管 理できます。これにより、選択した回復シナリオを編集したり、特定の回復シ ナリオを特定のテストに関連付けて、指定したトリガ・イベントが発生したと きに回復シナリオを使用するように QuickTest を設定したりできます。

回復ファイルの作成

回復シナリオを格納するには回復ファイルを作成します。新規の回復ファイル を作成することも、既存のファイルを編集することもできます。 回復ファイルを作成するには、次の手順を実行します。

1 [**リソース**] > [**回復シナリオ マネージャ**] を選択します。[回復シナリオ マ ネージャ] ダイアログ・ボックスが開きます。

₩回復シナリオ マネージャ				×
ファイル: <mark>〈無題〉</mark>	新規作成	厭▼	保存	-
シナリオ	<u> 7</u>		×	
, シナリオ記述				
			<u>^</u>	
			Ŧ	
	閉じる		レプ	

2 標準設定では、「回復シナリオマネージャ」ダイアログ・ボックスには新規の 回復ファイルが表示されます。この新規ファイルを使用するか、「開く」ボタ ンをクリックし、既存の回復ファイルを選択します。あるいは、「開く」ボタ ンの横にある矢印をクリックして、最近使用した回復ファイルをリストから選 択します。

次の項で説明するように、回復シナリオ・ウィザードを使用して回復シナリオ を作成し、回復ファイルに保存することができます。

[回復シナリオ マネージャ] ダイアログ・ボックスについて

[回復シナリオマネージャ]ダイアログ・ボックスでは、回復ファイルの作成 と編集、およびこれらのファイルに保存されている回復シナリオの作成と管理 ができます。

[回復シナリオマネージャ]ダイアログ・ボックスには、現在開いている回復 ファイルの名前、回復ファイルに保存したシナリオのリスト、および各シナリ オの説明が表示されます。

♥回復シナリオ マネージャ	×
ファイル: <課題>	新規作成 【保存 ▼
シナリオ	
 シナリオ記述	
	*
	_
	閉じる ヘルプ

[回復シナリオマネージャ] ダイアログ・ボックスには,次のツールバー・ボ タンがあります。

オプション	詳細
新規作成	新規の回復ファイルを作成します。詳細については,1245 ページ「回 復ファイルの作成」を参照してください。
開く▼	既存の回復ファイルを開きます。矢印をクリックして,最近使用した 回復ファイルのリストから回復ファイルを選択することもできます。
保存▼	現在の回復ファイルを保存します。詳細については,1273ページ「回 復ファイルでの回復シナリオの保存」を参照してください。
3	回復シナリオ・ウィザードを開き,新規の回復シナリオを定義します。 詳細については,1249ページ「回復シナリオ・ウィザードについて」 を参照してください。
1	選択した回復シナリオの回復シナリオ・ウィザードを開き,回復シナ リオの設定を変更できます。詳細については,1277 ページ「回復シナ リオの変更」を参照してください。
	選択した回復シナリオのプロパティのサマリを表示専用の形式で表示 します。詳細については、1276ページ「回復シナリオのプロパティの 表示」を参照してください。
2	回復シナリオを,開いている回復ファイルからクリップボードにコ ピーします。これによって,回復シナリオを別の回復ファイルに貼り 付けることができます。詳細については,1279ページ「回復シナリ オ・ファイル間での回復シナリオのコピー」を参照してください。
	回復シナリオを、クリップボードから開いている回復ファイルに貼り 付けます。詳細については、1279ページ「回復シナリオ・ファイル間 での回復シナリオのコピー」を参照してください。
×	回復シナリオを削除します。詳細については,1278 ページ「回復シナ リオの削除」を参照してください。

注: 各回復シナリオには、その種類ごとに異なるアイコンが関連付けられていま す。詳細については、1275ページ「回復シナリオの管理」を参照してください。

回復シナリオ・ウィザードについて

回復シナリオ・ウィザードでは、回復シナリオを作成するプロセスを段階的に 案内します。回復シナリオ・ウィザードには、次の主要なステップが含まれて います。

- ▶ 実行セッションを中断するトリガ・イベントの定義
- ▶ 継続に必要な回復操作の指定
- ▶ 回復後のテスト実行の操作の選択
- ▶ 回復シナリオの名前と説明の指定
- ▶ 回復シナリオを現在のテストに関連付けるか、すべての新規テストに関連付けるかどうかの指定

回復シナリオ・ウィザードを開くには, [回復シナリオマネージャ] ダイアロ グ・ボックス ([**リソース**] > [**回復シナリオマネージャ**])の [**新規シナリ オ**] ボタンをクリックします。



[回復シナリオ ウィザードにようこそ] 画面

[回復シナリオウィザードにようこそ] 画面には、回復シナリオ・ウィザード のさまざまなオプションについての一般的な情報と、回復シナリオの定義に関 連する段階の概要が表示されます。

回復シナリオ ウィザード	
	回復シナリオ ウィザードにようこそ 回復シナリオ ウィザードの手順に沿って、回復シナリオを簡単に定義できます。回復 シナリオでは、テストの実行が中断される予想外のイベントと、その対処方法を定義し ます。 このウィザードを使って回復シナリオを定義すると、聞かれているテスト、および新規テ ストの標準設定に適用できます。
QuickTest Professional ・ようこそ ・トリガ ・回復 ・回復後操作 ・名前 ・完了	このウィザードの使用目的: 1. テストの実行を中断する原因となるトリガ イベントを定義する。 2. テスト実行の再開、続行に必要な回復操作を指定する。 3. 回復後のテスト実行操作を選択する。 4. シナリオについての情報を記述する。 続行するには、『次へ』 をクリックしてください。
	< 戻る(母) (次へい)> 完了 キャンセル ヘルプ

[次へ]をクリックして [トリガイベントの選択] 画面に進みます(1251 ページを参照してください)。

[トリガイベントの選択] 画面

[トリガイベントの選択] 画面では、回復シナリオをトリガするイベントのタ イプと、QuickTest によるイベントの認識方法を定義できます。

回復シナリオ ウィザード		×
レンジェン した した した した での した での した での した での した での した での した での した での の にの にの での の にの での の にの での の にの での での の の の にの での での の の の の にの での での の の の の の の した での の の の の の の の の の の の の の	 トリガイベントの選択 回復操作を開始するきっかけとなるトリガイベントの種類を選択してください。 アッファッブウィンドブ テスト実行中、起動中のアプリケーションでウィンドウが表示された場合 オブジェクトの状態 アプリケーション内オブジェクトのプロパティ値が指定値と一致した場合 (プロ パティ値は、階層内の各オブジェクトで指定可能) テスト実行エラー テスト内のステップが正しく実行されない場合 アブリケーションが実行失敗になった場合 	
	< 戻る(B) 次へ(N) > 完了 キャンセル ヘルプ	

トリガのタイプを選択し、[**次へ**]をクリックします。ウィザードに表示され る次の画面は、以下のどのトリガのタイプを選択したかに応じて異なります。

▶ [ポップアップウィンドウ]: QuickTest によって、ポップアップ・ウィンドウ が検出され、ウィンドウのタイトルとテキストの内容に従ってそのウィンドウ が識別されます。たとえば、実行セッション中にプリンタの用紙切れを示す メッセージ・ボックスが表示される場合があります。QuickTest では、実行セッ ションを続行するために、このウィンドウを検出し、定義済みの回復シナリオ を呼び出すことができます。

このオプションを選択し、[次へ]をクリックして、[ポップアップウィンドウの条件を指定]画面に進みます(1253ページを参照してください)。

▶ [オブジェクトの状態]: QuickTest によって、特定のテストのオブジェクト状態 が検出され、そのプロパティ値とすべての祖先のプロパティ値に従ってオブ ジェクト状態が識別されます。オブジェクトはクラスではなく、プロパティ値 によってのみ識別されます。 たとえば、特定のプロセスが開いている場合にダイアログ・ボックスの特定の ボタンが無効になることがあります。QuickTest では、この問題のプロセスを開 いているときに発生するボタンのオブジェクト・プロパティ状態を検出し、定 義済みの回復シナリオを呼び出して、そのプロセスを閉じて実行セッションを 続行できます。

このオプションを選択し、[次へ]をクリックして、[オブジェクトの選択] 画 面に進みます(1255ページを参照してください)。

▶ [テスト実行エラー]: QuickTest では、テストの実行エラーが検出され、メソッドからの失敗した戻り値によってエラーが識別されます。たとえば、QuickTestでは、メニュー項目は実行セッション中の特定の場所では利用できないため、メソッド引数で指定されたメニュー項目を識別できない場合があります。 QuickTestでは、実行セッションを続行するために、この実行エラーを検出し、定義済みの回復シナリオを呼び出すことができます。

このオプションを選択し、[次へ]をクリックして、[テスト実行エラー] 画面 に進みます(1258ページを参照してください)。

▶ [アプリケーションのクラッシュ]: QuickTest によって、アプリケーション・ク ラッシュが検出され、定義済みのアプリケーションのリストに従って識別され ます。たとえば、実行セッション中にステップを実行したときに、2次的なア プリケーションがクラッシュすることが考えられます。使用中のアプリケー ションの問題ではない可能性のあるこのクラッシュによって実行セッションが 失敗とならないように、QuickTest では、このアプリケーションのクラッシュを 検出し、定義済みの回復シナリオを呼び出して実行セッションを続行すること ができます。

このオプションを選択し、[**次へ**]をクリックして、[回復操作] 画面に進みます(1261 ページを参照してください)。

注:

- ▶ 一連の回復操作は、トリガ・イベントの条件に合致する状態が発生するたび に実行されます。たとえば、特定のオブジェクト状態を定義し、2つのオブ ジェクトがこの状態と一致した場合、指定した状態に各オブジェクトが一致 するたびに1回ずつ、合計2回の回復操作が実行されます。
- ➤ 回復メカニズムでは、テストの最後のステップで発生するトリガは処理されません。テストの最後のステップで発生する可能性のある予期しないイベントやエラーから回復する必要がある場合は、テストの最後にさらにステップを追加することで、予期しないイベントやエラーから回復できます。

[ポップアップ ウィンドウの条件を指定] 画面

[トリガイベントの選択] 画面にある [ポップアップウィンドウ] トリガを選 択した場合(1251ページを参照してください), [ポップアップウィンドウの 条件を指定] 画面が開きます。

回復シナリオ ウィザード	
	オップアップ ウィンドウの条件を指定 指定された回復シナリオで表示されるポップアップ ウィンドウを認識 する条件を指定します。指差しアイコンを使って、ポップアップ ウィン ドウのタイトルおよびテキストの値をキャプチャできます。 ウィンドウの認識条件:
Quick lest Professional ・ようこそ ・トリガ ・回復 ・回復後操作 ・名前 ・完了	 □ 正規表現 □ ウィンドウのテキスト内容: □ 正規表現

次のいずれかを実行してポップアップ・ウィンドウの識別方法を指定します。

- > ポップアップ・ウィンドウの [ウィンドウのタイトル] または [ウィンドウ のテキスト内容],あるいはその両方に基づいてポップアップ・ウィンドウ を識別するかどうかを選択し、ポップアップ・ウィンドウの識別に使用する テキストを入力します。ウィンドウ・タイトルまたはテキスト内容に対して 正規表現を使用するには、該当する [正規表現] チェック・ボックスを選択 し、該当する場所に正規表現を入力します。正規表現の詳細については、 757 ページ「正規表現の使用について」を参照してください。
- ▶ 指差しマークをクリックします。次に、ポップアップ・ウィンドウをクリックし、ウィンドウ・タイトルとウィンドウのテキスト内容をキャプチャします。

注:前者([ウィンドウのタイトル] または [ウィンドウのテキスト内容],あ るいはその両方)を実行した場合,QuickTest は指定したタイトルまたはテキス ト,あるいはその両方を含んだ任意のポップアップ・ウィンドウを識別しま す。後者(指差しマーク)を実行した場合,QuickTest は選択したウィンドウの オブジェクト・プロパティ値に一致するポップアップ・ウィンドウのみを識別 します。

ヒント:ウィンドウのフォーカスを変更したり、ショートカット・メニューを 表示するために右クリックやポインタをオブジェクト上に移動するなどの操作 を実行したりするには、左側の CTRL キーを押しながら操作を行います。選択 対象オブジェクトを含んでいるウィンドウが最小化されている場合は、左 CTRL キーを押しながら、Windows タスク・バー内のアプリケーションを右クリック して、ショートカット・メニューから [元のサイズに戻す] を選択すること で、ウィンドウを表示できます。

[次へ]をクリックして[回復操作]画面に進みます(1261ページを参照して ください)。

[オブジェクトの選択] 画面

[トリガイベントの選択] 画面にある [オブジェクトの状態] トリガを選択する と(1251 ページを参照してください), [オブジェクトの選択] 画面が開きます。

回復シナリオ ウィザード	x
	オブジェクトの選択 指差しアイコンを使って、プロパティを指定する対象となるオブジェ プラー
QuickTest Professional ・ようこそ ・トリガ ・回復 ・回復後操作 ・名前 ・完了	オブジェクトの階層構造:
	(人民る低) 次へ(N) 完了 キャンセル ヘルプ

指差しマークをクリックしてから、プロパティを指定するオブジェクトをク リックします。

ヒント:ウィンドウのフォーカスを変更したり,ショートカット・メニューを 表示するために右クリックやポインタをオブジェクト上に移動するなどの操作 を実行したりするには,左側の CTRL キーを押しながら操作を行います。選択 対象オブジェクトを含んでいるウィンドウが最小化されている場合は,左 CTRL キーを押しながら,Windows タスク・バー内のアプリケーションを右クリック して,ショートカット・メニューから [元のサイズに戻す] を選択すること で,ウィンドウを表示できます。 クリックした場所が複数のオブジェクトに関連付けられている場合は、[オブジェ クトの選択-オブジェクトの状態トリガ]ダイアログ・ボックスが開きます。

オブジェクトの選択 - オブジェクトの状態トリガ
クリックした場所は、複数のオブジェクトと関連付けられています。 操作を行う対象となるオブジェクトを選択してください。
E-S Browser : Welcome: Mercury Tours
🖻 🚽 Page : Welcome: Mercury Tours
🚊 📲 WebTable : Home
📥 🚟 WebTable : SIGN-ON
🖮 🚰 WebTable : Atlanta to Las Vegas
🚊 📲 WebTable : Atlanta to Las Vegas
Image : Featured Destination: Aruba
OK キャンセル ヘルプ

指定するプロパティを含んでいるオブジェクトを選択し, [**OK**] をクリックします。選択したオブジェクトとその親が [オブジェクトの選択] 画面に表示されます。

注: 階層オブジェクトの選択ツリーでは,Webテーブルなど,QuickTest が通常は 学習しないオブジェクト(親以外のオブジェクト)を選択することもできます。

[次へ]をクリックして [オブジェクトのプロパティと値の設定] 画面に進み ます(1257 ページを参照してください)。
[オブジェクトのプロパティと値の設定] 画面

[オブジェクトの選択] 画面でプロパティを指定するオブジェクトを選択する と(1255 ページを参照してください), [オブジェクトのプロパティと値の設 定] 画面が開きます。

回復シナリオ ウィザード					×
	オブジェクトのフロパティと値の。 階層内の各メンバで、オブジェクトを でオブジェクトを選択し、認識で適用 ここで指定される値は、オブジェクト!	設定 プロパティと値の/ 引するプロパティと リポジトリ内のオ:	組み合わせによっ 値の組み合わせ ブジェクトに影響を	て認識します。ツ を指定してください 汲ぼしません。	J
(UP)	オブジェクトの階層ツリー	オブジェクトのプ	ロパティ	追加/削除…	
QuickTest	Browser : Welcome: Me	217	プロパティ	苗	
Professional	Link : Destinatic	ABC	text html tar	Destinations	-
• ようこて					
● 回復					. 💌
● 回復後操作		· · · · · · · · · · · · · · · · · · ·			
● 名前		プロパティ値の編	i集: ∣Destii	nations 	
● 完了			□止規	表現	
			الرحلي لأبر حل		
		元「	キャンセル		

階層の各オブジェクトに対して、[**プロパティ値の編集**] ボックスで、オブ ジェクトの識別に使用するプロパティ値を変更できます。[**追加/削除**] ボタン をクリックし、検査対象のプロパティ値のリストからオブジェクト・プロパ ティを追加または削除することもできます。オブジェクトはクラスではなく、 プロパティ値によってのみ識別されます。

プロパティ値で正規表現を使用する場合は, [**正規表現**] チェック・ボックス を選択します。正規表現の詳細については, 757 ページ「正規表現の使用について」を参照してください。

[次へ]をクリックして[回復操作]画面に進みます(1261ページを参照して ください)。

[テスト実行エラー] 画面

[トリガイベントの選択] 画面にある [**テスト実行エラー**] トリガを選択すると (1251 ページを参照してください), [テスト実行エラーの選択] 画面が開きます。

回復シナリオ ウィザード	
	テスト実行エラーの選択 トリガ イベントとして使用する実行エラーを選択してください。
QuickTest Professional ・ようこそ ・トリガ ・回復 ・回復後操作 ・名前 ・完了	エラー: オブジェクトが見つかりません。 エラー記述: オブジェクトのテスト オブジェクト記述と一致するオブジェクトは、指定された親オブジェ クトにありません。
	< 戻る(B) 次へ(N)> 完了 キャンセル ヘルプ

[**エラー**] リストでは,次のように,トリガ・イベントとして使用するテストの実行エラーを選択します。

- ▶ [任意のエラー]: テスト・オブジェクト・メソッドによって返されるすべての エラー・コード。
- ▶ [リストまたはメニュー内の項目が一意ではありません]: リスト、メニュー、 またはツリーにある複数の項目に、メソッド引数で指定されている名前がある 場合に発生します。
- ▶ [リストまたはメニュー内の項目が見つかりません]:メソッド引数で指定されているリスト、メニュー、またはツリー項目が QuickTest によって識別できない場合に発生します。この原因としては、その項目を現在利用できない、または名前が変更されていることが考えられます。
- ▶ [物理記述に対して複数のオブジェクトが該当します]:アプリケーションの複数のオブジェクトが、ステップで指定されているオブジェクトのテスト・オブジェクト記述で指定されている値と同一のプロパティ値を有する場合に発生します。

- ▶ [オブジェクトが無効になっています]: ステップで指定されているオブジェクトが現在無効となっているため、QuickTest がステップを実行できない場合に発生します。
- ▶ [オブジェクトが見つかりません]:指定された親オブジェクト内に、オブジェ クトのテスト・オブジェクト記述と一致するオブジェクトがない場合に発生し ます。
- ▶ [オブジェクトが非表示になっています]: ステップで指定されたオブジェクトが現在画面上に表示されていないため、QuickTest がステップを実行できない場合に発生します。

[次へ]をクリックして 1261 ページ「[回復操作] 画面」に進みます。

[プロセスの選択] 画面

[トリガイベントの選択] 画面にある [**アプリケーションのクラッシュ**] トリ ガを選択すると(1251ページを参照してください), [プロセスの選択] 画面が 開きます。

回復シナリオ ウィザード			×
	プロセスの選択		
	選択されたプロセスをリストに追加するには、[+] するか、「実行中のプロセス」 リストで1 つまたは をクリックしてください。	ボタンをクリックしてプロセン それ以上のプロセスを選択	ス名を入力 尺し [追加]
(b)	実行中のプロセス	プロセス	+ ×
QuickTest Professional	AcroRd32.exe agentsvr.exe conime.exe csrss.exe		
• ようこそ	Ctfmon.exe EXCEL.EXE		
 ● トリガ ● 回復 	explorer.exe iexplore.exe instinfo.exe		
● 回復後操作	jusched.exe Isass.exe		
● 名前 ● 空フ	Iservnt.exe mdm.exe		
• . . 1	00. JavaService exe		_
	戻る(B) 次へ(N) 次へ(N) 完了	キャンセル	

[実行中のプロセス] リストには,現在実行中のすべてのアプリケーション・ プロセスが表示されます。[プロセス] リストには,クラッシュした場合に回 復シナリオをトリガするアプリケーション・プロセスが表示されます。 +

×

アプリケーション・プロセスを [**プロセス**] リストに入力するか, [実行中の **プロセス**] リストからアプリケーション・プロセスを選択することによって, [**プロセス**] リストにアプリケーション・プロセスを追加できます。

- ▶ [実行中のプロセス] リストからプロセスを追加するには、[実行中のプロセス] リストのプロセスをダブルクリックするか、プロセスを選択して [追加] ボタンをクリックします。Windows で複数選択する際の標準的な方法 (CTRL キーや SHIFT キー)を使用して、複数のプロセスを選択できます。
- ▶ [プロセス] リストにプロセスを直接追加するには、「新規プロセスの追加] ボタンをクリックし、リストに追加するプロセスの名前を入力します。
 - ▶ [プロセス] リストからプロセスを削除するには、プロセスを選択して [プ ロセスの削除] ボタンをクリックします。

ヒント: プロセスの名前を変更するには, [**プロセス**] リストでプロセスを選択し, プロセス名をクリックして編集します。

[次へ]をクリックして[回復操作]画面に進みます(1261ページを参照して ください)。

[回復操作] 画面

[回復操作] 画面では、回復シナリオにおける一連の回復操作を管理できます。 回復操作とは、QuickTest によってトリガ・イベントが認識されたときに順次実 行される操作です。

		_
	回復操作 トリガ イベントが発生した際行う、複数の操作を定義できます。[上に移 動]、「下に移動] ボタンを使って、操作を行う順番を指定してください。	
QuickTest Professional ・ようこそ ・トリガ	回復操作: ♪ ↓ × タイプ 詳細	
 ●回復後操作 ●名前 ●完了 	必ず1つまたはそれ以上の操作を定義してください。定義するには じたへ〕 ボ タンを押してください。 く戻る(B) 次へ(N) > 完了 キャンセル	

少なくとも1つの回復操作を定義する必要があります。回復操作を定義し、それを[**回復操作**]リストに追加するには、[**次へ**]をクリックして、[回復操 作] 画面に進みます(1262 ページを参照してください)。

2 つ以上の回復操作を定義する場合,回復操作を選択し,[**上に移動**]または [**下に移動**]ボタンを使用して,QuickTestによる回復操作の実行順序を変更で きます。回復操作を選択し,[**削除**]ボタンをクリックして,回復シナリオか ら回復操作を削除することもできます。

注: [Microsoft Windows の再起動]回復操作を定義する場合,常に最後の回 復操作としてこの操作が挿入されるため,リストで位置は変更できません。

回復操作を1つでも定義すると、[他の回復操作を追加する] チェック・ボッ クスが表示されます。

- ▶ ほかの回復操作を定義するには、このチェック・ボックスを選択し、[次へ] をクリックします。
- ▶ チェック・ボックスをクリアし、[次へ] をクリックして [回復後のテスト実行オプション] 画面に進みます (1269 ページを参照してください)。

[回復操作] 画面

[回復操作] 画面では,QuickTest によるトリガ・イベント検出後に実行される 操作を指定できます。

回復シナリオ ウィザード	
回復シナリオ ウィザード の で の い に 大 し に 大 し い サ し に ち た し し し し し し し し し し し し し	▲ ■復操作 トリガ イベントが発生した際行う操作を選択してください。 操作の種類: キーボードまたはマウス操作 アプリケーション プロセスの終了 関数の呼び出し Microsoft Windows の再起動 メモ: 再起動の操作を選択する場合は、すべての回復操作が完了して
 ● トリカ ● 回復 ● 回復後操作 ● 名前 ● 完了 	からウィンドウが再起動されるように、リストの最後に指定してください。 「 「 「 、 、 、 、 、 、 、 、 、 、 、 、 、 、 、 、

回復操作のタイプを選択し、[次へ]をクリックします。ウィザードに表示される次の画面は、選択する回復操作のタイプに応じて異なります。

次のタイプの回復操作を定義できます。

▶ [キーボードまたはマウス操作]: QuickTest によって、ウィンドウのボタンをク リックする操作、またはキーボードのキーを押す操作がシミュレートされま す。このオプションを選択し、[次へ] をクリックして、[回復操作−ボタン、 またはキーを押す] 画面に進みます(1264 ページを参照してください)。

- ▶ [アプリケーション プロセスの終了]: QuickTest によって、指定のプロセスが 閉じます。このオプションを選択し、[次へ] をクリックして、[回復操作−プ ロセスの終了] 画面に進みます(1266 ページを参照してください)。
- ➤ [Microsoft Windows の再起動]: QuickTest によって Microsoft Windows が再起 動されます。このオプションを選択し、[次へ] をクリックして、[回復操作] 画面に進みます(1261ページを参照してください)。

注: [Microsoft Windows の再起動] 回復操作を使用する場合,操作を実行す る前に,この回復シナリオと関連付けられているテストをすべて保存する必要 があります。また,再起動時に自動ログインするように,テストを実行するコ ンピュータを設定する必要があります。

[回復操作-ボタン,またはキーを押す] 画面

[回復操作] 画面で [**キーボードまたはマウス操作**] 回復操作を選択した場合 (1262 ページを参照してください), [回復操作-ボタン,またはキーを押す] 画面が開きます。

回復シナリオ ウィザード	×
	回復操作 - ボタン、またはキーを押す 押すボタン、またはキーボードのキーを選択してください。指差しアイコンを使って、ボタ ンのラベルを指定することができます。キーボードで複数のキーの組み合わせを押すこ とも可能です。
QuickTest Professional	キーボード、またはマウス操作を選択してください。 ② [標準設定] ボタン/ENTER キーを押す ③ [キャンセル] ボタン/ESCAPE キーを押す
 ● ようこそ ● トリガ ● 回復 ● 回復後操作 ● 名前 	 ● 押すボタンのラベルを指定する: ● キー、またはキーの組み合わせを押す: □ なし (例: F2、Ctrl+A、Ctrl+Alt+D)
● 完了	〈戻る(B) 次へいひろう 完了 キャンセル ヘルプ

QuickTest によってトリガ・イベントが検出された場合に実行させる,キーボー ドまたはマウスの操作を指定します。

- ▶ [[標準設定] ボタン/ENTER キーを押す]: トリガが発生した場合に、表示されているウィンドウで、標準のボタンをクリックする、または ENTER キーを押すよう QuickTest に指示します。
- ▶ [[キャンセル] ボタン/ESCAPE キーを押す]:トリガが発生した場合に、表示 されているウィンドウで、[キャンセル] ボタンをクリックする、または ESCAPE キーを押すよう QuickTest に指示します。
- ▶ [押すボタンのラベル指定する]:トリガが発生した場合に、表示されている ウィンドウで、指定したラベルの付いたボタンをクリックするよう QuickTest に指示します。このオプションを選択した場合は、指差しマークをクリックし た後、トリガ・ウィンドウの中の任意の場所をクリックします。

ヒント:ウィンドウのフォーカスを変更したり,ショートカット・メニューを 表示するために右クリックやポインタをオブジェクト上に移動するなどの操作 を実行したりするには,左側の CTRL キーを押しながら操作を行います。選択 対象オブジェクトを含んでいるウィンドウが最小化されている場合は,左 CTRL キーを押しながら,Windows タスク・バー内のアプリケーションを右クリック して,ショートカット・メニューから [元のサイズに戻す] を選択すること で,ウィンドウを表示できます。

選択されたウィンドウのボタン・ラベルはすべて、リスト・ボックスに表示さ れます。リストから必要なボタンを選択します。

▶ [キー、またはキーの組み合わせを押す]:トリガが発生した場合に、表示されているウィンドウで、指定したキーボードのキーまたはキーの組み合わせを押すようQuickTestに指示します。このオプションを選択した場合は、エディット・ボックスをクリックした後、指定する単独のキーまたはキーの組み合わせを押します。

[次へ]をクリックします。[回復操作] 画面が再び開き,定義したキーボード またはマウスの回復操作が表示されます。

[回復操作ープロセスの終了] 画面

[回復操作] 画面の [アプリケーション プロセスの終了] 回復操作を選択する と(1262ページを参照してください), [回復操作-プロセスの終了] 画面が開 きます。

回復シナリオ ウィザード	<u>></u>	1
	回復操作 - フロセスの終了 [+] ボタンをクリックしてプロセス名を入力するか、「実行中のプロセス] リストで 1 つまた はそれ以上のプロセスを選択し 「追加」 をクリックしてください。	
QuickTest Professional ・ようこそ ・トリガ ・回復 ・回復後操作 ・名前 ・完了	実行中のプロセス AcroFd32 exe agentsvr.exe conime.exe csrss.exe ctfmon.exe EXCELEXE explore.exe iexplore.exe isslore.exe sass.exe Iservnt.exe mdm.exe	
[<戻る(B) 次へ(N) > 完了 キャンセル へルプ	

[実行中のプロセス] リストには,現在実行中のすべてのアプリケーション・ プロセスが表示されます。[終了するプロセス] リストには,トリガが呼び出 されたときに閉じるアプリケーション・プロセスが表示されます。

- ▶ [実行中のプロセス] リストからプロセスを追加するには、[実行中のプロセス] リストのプロセスをダブルクリックするか、プロセスを選択して [追加] ボタンをクリックします。Windows で複数選択する際の標準的な方法 (CTRL キーや SHIFT キー)を使用して、複数のプロセスを選択できます。
- ▶ [終了するプロセス] リストにプロセスを直接追加するには、[新規プロセスの 追加] ボタンをクリックし、リストに追加するプロセスの名前を入力します。
- ▶ [終了するプロセス] リストからプロセスを削除するには、プロセスを選択して [プロセスの削除] ボタンをクリックします。

+

×

ヒント:プロセスの名前を変更するには、[終了するプロセス]リストでプロ セスを選択し、プロセス名をクリックして編集します。

[次へ]をクリックします。[回復操作] 画面が再び開き,定義した「プロセス を終了する」回復操作が表示されます。

[回復操作ー関数の呼び出し] 画面

[回復操作] 画面で [**関数の呼び出し**] 回復操作を選択すると(1262ページを 参照してください), [回復操作--関数の呼び出し] 画面が開きます。

	以仔のフインフリーファイルおよびファイルは400関数。 してください(後者の場合は、ライブラル ファイルに	を選択するか、新規関数を定義 (保存することになります)。
/ b	関数ライブラリ:	•
49	€ 関数の選択:	T
QuickTest	 新規関数の定義: 	
	関数のプロトタイプ: Function RecoveryFunctio	n1 (ProcessName, ProcessID)
- みりこて		
● トリガ		
● トリガ ● 回復		<u> </u>
 トリガ 回復 回復後操作 		A
 トリガ 回復 回復後操作 名前 	4	×

[**関数ライブラリ**] ボックスで,最近指定した関数ライブラリを選択します。 あるいは,参照ボタンをクリックして,既存の関数ライブラリに移動します。

注: QuickTest では,選択した関数ライブラリが,テストと自動的に関連付けられます。したがって,[テストの設定]ダイアログ・ボックスの[リソース] タブで,それぞれ関数ライブラリとテストを関連付ける必要はありません。 関数ライブラリを選択した後、次のオプションのいずれか1つを選択します。

▶ [**関数の選択**]: 選択した関数ライブラリから既存の関数を選択します。

1251 ページ「[トリガイベントの選択] 画面」で選択したトリガ・タイプのプロトタイプ構文と一致する関数のみが表示されます。

各トリガ・タイプのプロトタイプを次に示します。

テスト実行エラーのトリガ

OnRunStep(

)

[in] Object as Object: 現在のステップのオブジェクト。

[in] Method as String: 現在のステップのメソッド。

[in] Arguments as Array: 実際のメソッドの引数。

[in] Result as Integer: 実際のメソッドの結果。

ポップアップ・ウィンドウとオブジェクトの状態のトリガ OnObject

[in] Object as Object: 検出されたオブジェクト。

アプリケーションのクラッシュのトリガ

OnProcess

)

[in] ProcessName as String: 検出されたプロセスの名前。

[in] ProcessId as Integer: 検出されたプロセスの ID。

▶ [新規関数の定義]:一意の名前を指定し、表示される関数のプロトタイプに 従って [関数名] ボックスで関数を定義することによって、新規の関数を作成 します。新しい関数が、選択した関数ライブラリに追加されます。

注: 複数のシナリオで異なる関数ライブラリから同一の名前を持つ関数を使用 すると,回復プロセスは失敗することがあります。この場合,実行セッション 中に回復の失敗に関する情報が表示されます。

[次へ]をクリックします。[回復操作]画面が再び開き(1261ページを参照してください),定義した関数の操作が表示されます。

[回復後のテスト実行オプション] 画面

[回復操作] 画面の [他の回復操作を追加する] チェック・ボックスをクリア し(1261 ページを参照してください), [次へ] をクリックすると, [回復後の テスト実行オプション] 画面が開きます。回復後のテスト実行のオプションで は, QuickTest によってイベントが識別され, 指定の回復オプションがすべて実 行された後の, 実行セッションの継続方法を指定します。

回復シナリオ ウィザード	
	回復後のテスト実行オブション 回復操作が完了した後に行うテストの実行操作を選択してください。
QuickTest Professional ・ようこそ ・トリガ ・回復 ・回復後操作 ・名前 ・完了	 テストの実行オブション: 現在のステップを繰り返し、統行する。 次のステップに進む。 次のアクションまたは、コンボーネント反復に進む。 次のテスト反復に進む。 現在のテスト実行を再開する。 テスト実行を停止する。
	< 戻る(B) 次へ(N) > 完了 キャンセル ヘルプ

定義した回復操作が実行された後、QuickTest では、次の実行セッションのオプションのいずれか1つを実行できます。

▶ [現在のステップを繰り返し、続行する]

この現在のステップとは、回復シナリオがトリガされた際に QuickTest が実行 中であったステップです。回復シナリオに対して「**エラー発生時**」呼び出しオ プションを使用している場合、通常、エラーを返すステップは、トリガ・イベ ントの発生原因となったステップより1つ以上後のステップです。

したがってほとんどの場合,現在のステップを繰り返しても,トリガ・イベントは再現されません。詳細については,1284ページ「回復シナリオの有効化と無効化」を参照してください。

▶ [次のステップに進む]

回復シナリオがトリガされた際にQuickTest が実行中であったステップをス キップします。アプリケーションに対して操作を実行するステップをスキップ すると、それ以降のステップが失敗する場合があることに注意してください。

▶ [次のアクションまたはコンポーネント反復に進む]

現在のアクションまたはコンポーネントの反復でのステップ実行を停止し,最初(または現在のアクションまたはコンポーネントを追加的に反復する必要がない場合は次のアクションまたはコンポーネント)から次のアクションの反復 を開始します。

▶ [次のテスト反復に進む]

現在のアクションまたはコンポーネントでステップ実行を停止し、最初から次のQuickTest テストまたはビジネス・プロセス・テストの反復を開始します(またはテストを追加的に反復する必要がない場合はテストの実行を停止します)。

▶ [現在のテスト実行を再開する]

ステップの実行を停止し、テストを始めから再実行します。

▶ [テストの実行を停止する]

テストの実行を停止します。

注:回復操作として [Microsoft Windows の再起動]を選択した場合,上記の 最後2つのテスト実行のオプションのみから選択できます。

テスト実行オプションを選択し、[次へ]をクリックして、[名前と記述] 画面 に進みます(1271ページを参照してください)。

[名前と記述] 画面

[回復後のテスト実行オプション] 画面でテスト実行オプションを指定し (1269 ページを参照してください), [次へ] をクリックすると, [名前と記述] 画面が開きます。

[名前と記述] 画面では,回復シナリオを識別するための名前を指定します。 シナリオに関する説明情報を追加することもできます。

回復シナリオ ウィザード		×
	名前と記述 回復シナリオの名前および記述を入力してください。	
(p)	回復ファイル: 〈新規回復シナリオ ファイル〉 シナリオ名: 〈	
QuickTest Professional	記述: 	
 ⇒ よっこそ ● トリガ ● 回復 		
 ● 回復後操作 ● 名前 		
● 完了		
	へルプ	

回復シナリオの名前とテキスト形式の説明を入力し, [**次へ**] をクリックして, [回復シナリオ ウィザードの完了] 画面に進みます(1272 ページを参照してく ださい)。

[回復シナリオ ウィザードの完了] 画面

[名前と記述] 画面で回復シナリオの名前と説明を指定し(1271ページを参照 してください), [次へ] をクリックすると, [回復シナリオ ウィザードの完了] 画面が開きます。

[回復シナリオウィザードの完了] 画面では、定義したシナリオ設定の概要を 確認できます。回復シナリオを自動的に現在のテストに関連付けるかどうか、 または新しいすべてのテストの標準設定に回復シナリオを追加するかどうかを 指定できます。

回復シナリオ ウィザード		×
	回復シナリオ ウィザードの完了 次の回復シナリオのサマリを確認してください。シナリオ定義を変更するには、「戻る」 をクリックし、通切な画面に戻ってください。変更の必要がない場合は、「完了」をク リックし、回復シナリオを保存してください。	
QuickTest Professional ・ようこそ ・トリガ ・回復		
 ● 凹復夜採作 ● 名前 ● 完了 	 □ シナリオを現在のテストに追加する □ シナリオを標準テスト設定に追加する 	
	< 戻る(B) 次へ(N) > 完了 キャンセル ヘルプ	

[シナリオを現在のテストに追加する] チェック・ボックスを選択すると、この回復シナリオを現在のテストに関連付けることができます。[完了] をクリックすると、QuickTestによって、[テストの設定]ダイアログ・ボックスの [回復] タブにある [シナリオ] リストに回復シナリオが追加されます。

この回復シナリオを,新しく作るすべてのテストの標準設定のシナリオにする には,[シナリオを標準テスト設定に追加する]チェック・ボックスを選択し ます。次回テストを作成する際には,[テストの設定]ダイアログ・ボックス の[回復]タブの[シナリオ]リストに,このシナリオが表示されます。 **注**:シナリオは標準シナリオのリストから削除できます。詳細については, 1215ページ「テストのための回復シナリオ設定の定義」を参照してください。

[完了]をクリックすると、回復シナリオの定義は完了します。

回復ファイルでの回復シナリオの保存

回復シナリオ・ウィザードを使用して、回復ファイル内の回復シナリオの作成 または変更を行ったら、その回復ファイルを保存する必要があります。

ヒント:回復ファイルを保存せずに[回復シナリオマネージャ]ダイアログ・ ボックスの[**閉じる**]ボタンをクリックすると,回復ファイルを保存するよう 求められます。[**はい**]をクリックし,下記の手順2に進みます。既存の回復 ファイル内のシナリオの追加または変更を行い,メッセージに対して[**はい**] をクリックすると,回復ファイルとそのシナリオは保存されます。

新規の回復ファイルまたは変更した回復ファイルを保存するには,次の手順を 実行します。

1 [保存] ボタンをクリックします。既存の回復ファイル内のシナリオの追加または変更を行うと、回復ファイルとそのシナリオが保存されます。新規の回復ファイルを使用している場合は、[添付の保存] ダイアログ・ボックスが表示されます。

ヒント: [保存] ボタンの右側の矢印をクリックし, [名前を付けて保存] を選択して,別の名前で回復ファイルを保存することもできます。

2 ファイルの保存先フォルダを選択します。

3 [ファイル名] ボックスにファイルの名前を入力し, [保存] をクリックします。

注:リソースへのパスを保存する場合は,QuickTestによって,[オプション] ダイアログ・ボックスの[フォルダ]タブ([**ツール**] > [**オプション**] > [**フォルダ**])に,パスまたはパスの一部が存在するかどうかが確認されます。 パスが存在する場合は,入力したパスの相対的な部分だけを使用してパスを定 義するように求められます。パスが存在しない場合は,リソースのパスを [フォルダ] タブに追加して,相対的なパスを定義するように求められます。

詳細については、340ページ「QuickTest での相対パスの使用」を参照してくだ さい。

回復ファイルは、ファイル拡張子.qrs 付きで、指定の場所に保存されます。

回復シナリオの管理

回復シナリオを作成したら,回復シナリオ・マネージャを使用してシナリオを 管理できます。

♥回復シナリオ マネージャ					×
ファイル: C:¥Program Files¥HF¥Quic	新規作成		▼ (¥	存	•
シナリオ	20			×	
Cobject id popup 2 popup window test crash test run error					
, シナリオ記述					
		5	ヘルプ		

回復シナリオ・マネージャには、次の回復シナリオ・アイコンがあります。

アイコン	詳細
V	実行セッション中,開いているアプリケーションでウィンドウがポップ アップしたときに回復シナリオがトリガされることを示します。
V	対象回復シナリオが,アプリケーション内のオブジェクトのプロパティ 値が特定の値に一致したときに起動されることを示します。
V	テストのステップが正しく実行されないときに回復シナリオがトリガさ れることを示します。
Vē	実行セッション中に,開いているアプリケーションが失敗したときに回 復シナリオがトリガされることを示します。

回復シナリオ・マネージャでは、次の方法で既存のシナリオを管理できます。

- ▶ 回復シナリオのプロパティの表示
- ▶ 回復シナリオの変更
- ▶ 回復シナリオの削除
- ▶ 回復シナリオ・ファイル間での回復シナリオのコピー

回復シナリオのプロパティの表示

定義済みのすべての回復シナリオのプロパティを表示できます。

回復シナリオのプロパティを表示するには、次の手順を実行します。

1 [シナリオ] ボックスで、プロパティを表示する回復シナリオを選択します。



2 [**プロパティ**] ボタンをクリックします。あるいは, [**シナリオ**] ボックスのシ ナリオをダブルクリックする方法もあります。[回復シナリオのプロパティ] ダイアログ・ボックスが開きます。

回復シナリオのプロパティ	
一般 りガイベント 回復操作 回復後の操作	
回復ファイル· < <p>(無題)</p>	
シナリオ名: Recorvery Steps	
記述	
Stepを繰り返す	
回復シナリオ マネージャで回復シナリオを開き、シナリオのプロパティを	編集します。
J	
	閉じる ヘルプ

[回復シナリオのプロパティ]ダイアログ・ボックスには,選択したシナリオ に関する,次の読み取り専用の情報が表示されます。

- ▶ [一般] タブ:回復シナリオに対して定義されている名前と説明に加えて、シ ナリオが保存されている回復ファイルの名前とパスが表示されます。
- ▶ [トリガイベント] タブ:回復シナリオに対して定義されているトリガ・イベントの設定が表示されます。
- ▶ [回復操作] タブ:回復シナリオに対して定義されている回復操作が表示されます。
- ▶ [回復後の操作] タブ:回復シナリオに対して定義されている回復後の操作が 表示されます。

回復シナリオの変更

既存の回復シナリオの設定を変更できます。

回復シナリオを変更するには、次の手順を実行します。

1 [**シナリオ**] ボックスで,変更するシナリオを選択します。



- 2 [編集] ボタンをクリックします。回復シナリオ・ウィザードが開き,選択した回復シナリオに対して定義した設定が表示されます。
- 3 回復シナリオ・ウィザードを操作して、必要に応じて詳細を変更します。回復 シナリオ・ウィザードのオプションの詳細については、1245ページ「回復シナ リオの定義」を参照してください。

注:変更は、[回復シナリオマネージャ]ダイアログ・ボックスで [**保存**] を クリックするまで保存されません。変更を保存せずに [回復シナリオマネー ジャ]ダイアログ・ボックスの [**閉じる**] ボタンをクリックすると、回復ファ イルを保存するよう求められます。[**はい**] をクリックし、変更を保存します。

回復シナリオの削除

必要のない既存の回復シナリオは削除できます。回復シナリオ・マネージャか ら回復シナリオを削除すると、回復シナリオ・ファイルからは対応する情報も 削除されます。

注:削除した回復シナリオがテストと関連付けられている場合,QuickTest では、実行セッション中はその回復シナリオが無視されます。

回復シナリオを削除するには、次の手順を実行します。

- 1 [シナリオ] ボックスで、削除するシナリオを選択します。
- \mathbf{x}
- 2 [**削除**] ボタンをクリックします。[回復シナリオマネージャ] ダイアログ・ ボックスから,回復シナリオが削除されます。

注:シナリオは, [回復シナリオマネージャ]ダイアログ・ボックスで [**保存**] をクリックするまで実際には削除されません。削除を保存せずに [回復シナリ オマネージャ]ダイアログ・ボックスの [**閉じる**] ボタンをクリックすると, 回復ファイルを保存するよう求められます。[**はい**] をクリックして回復シナ リオ・ファイルを保存し,シナリオを削除します。 回復シナリオ・ファイル間での回復シナリオのコピー

ある回復シナリオ・ファイルから別の回復シナリオ・ファイルに、回復シナリ オをコピーできます。

ある回復シナリオ・ファイルから別の回復シナリオ・ファイルに回復シナリオ をコピーするには、次の手順を実行します。

- 1 [シナリオ] ボックスで、コピーする回復シナリオを選択します。
- 2 [**コピー**] ボタンをクリックします。シナリオがクリップボードにコピーされ ます。
 - 3 [開く] ボタンをクリックして、シナリオのコピー先の回復シナリオ・ファイ ルを選択するか、[新規作成] ボタンをクリックして、シナリオをコピーする 回復シナリオ・ファイルを新規作成します。
- 4 [貼り付け] ボタンをクリックします。シナリオが新規の回復シナリオ・ファ イルにコピーされます。

注:

C.

B

- ▶ 回復シナリオ・ファイルに同じ名前のシナリオがすでに存在する場合、それ をコピーした新しいシナリオで置き換えるかどうかを選択できます。
- > 変更は、「回復シナリオマネージャ」ダイアログ・ボックスで「保存」をク リックするまで保存されません。変更を保存せずに「回復シナリオマネー ジャ」ダイアログ・ボックスの「閉じる」ボタンをクリックすると、回復 ファイルを保存するよう求められます。「はい」をクリックし、変更を保存 します。

テストへの回復シナリオの関連付け

回復シナリオを作成した後は、作成した回復シナリオと選択したテストを関連付け、トリガ・イベントが発生した場合、実行セッション中にQuickTestによって 適切なシナリオが実行されるようにします。シナリオに優先順位を付け、実行 セッション中QuickTestがシナリオを適用する順序を設定できます。テストに関 連付けられている特定のシナリオ、またはすべてのシナリオを無効にすることも できます。すべての新しいテスト用の標準シナリオとして、どの回復シナリオ を使用するかも定義できます。

テストへの回復シナリオの追加

回復シナリオを作成した後は、トリガ・イベントが発生すると、実行セッション中に回復シナリオを実行するよう QuickTest に指示するために、1 つ以上のシナリオをテストと関連付けることができます。[テストの設定] ダイアログ・ボックスの[回復] タブには、現在のテストに関連付けられているすべての回復シナリオが表示されます。

ヒント:トリガ・イベントが発生すると,QuickTestによって,[回復]タブに 表示されている順序で適切な回復シナリオが確認されます。この順序は,1283 ページ「回復シナリオの優先順位の設定」で説明している手順で変更できます。 回復シナリオをテストに追加するには、次の手順を実行します。

1 [ファイル] > [設定] を選択します。[テストの設定] ダイアログ・ボックス が開きます。[回復] タブを選択します。

テス	トの設定		×
E	。 「ロパティ」 実行 「 リソース 」	パラメーター環境 Web 回復	
L.			
	シナリオ	+ × ↑ ↓ 🕾	
	シナリオ名	<u>ファイル</u>	
	₽ 🖓 Pop-up handler	C:\Program Files\My Tests\popup.qrs	
	ActiveX Unsafe	[QualityCenter] Subject\Web.grs	
	Authorization Failed	[QualityCenter] Subject\Web.grs	
	File Download	[QualityCenter] Subject\Web.grs	
	Internet Redirect	[QualityCenter] Subject\Web.grs	
	Metwork Error	[QualityCenter] Subject\Web.grs	
	<u> </u>		
	シナリオ記述		
		-	
		_	
	。 同時なませまのフタニップル。		
	回1度ソナリオのアクティブ16:	I7─発生時 ▼	
	標準値(z設定(D)		
	[[]	OK キャンセル 適用(A)	ヘルプ



2 [追加] ボタンをクリックします。[回復シナリオの追加] ダイアログ・ボック スが開きます。

回復シナリオの追加		×
回復ファイル: 1990		•
シナリオ:		
シナリオの追加	キャンセル	

- 3 [回復ファイル] ボックスで、テストと関連付ける回復シナリオが含まれてい る回復ファイルを選択します。あるいは、参照ボタンをクリックして、選択す る回復ファイルに移動します。[シナリオ] ボックスには、選択したファイル に保存されているシナリオの名前が表示されます。
- 4 [シナリオ] ボックスで、テストと関連付けるシナリオを選択し、[シナリオの 追加] をクリックします。[回復シナリオの追加] ダイアログ・ボックスが閉 じ、選択したシナリオが [回復] タブの [シナリオ] リストに追加されます。

ヒント:パスを1回クリックして強調表示した後,再度パスをクリックして編 集モードに入ることで,回復シナリオ・ファイルのパスを編集できます。たと えば,絶対ファイル・パスをファイルの相対ファイル・パスに変更したい場合 があります。回復シナリオ・ファイルのパスを変更する場合,テストを実行す る前に,回復シナリオが新しいパスの位置に定義されていることを確認する必 要があります。

回復シナリオのプロパティの表示

テストと関連付けられているすべての回復シナリオのプロパティを表示できます。

注: [回復シナリオ マネージャ] ダイアログ・ボックスから,回復シナリオの 設定を変更します。詳細については,1277 ページ「回復シナリオの変更」を参 照してください。

回復シナリオのプロパティを表示するには、次の手順を実行します。

- 1 [シナリオ] ボックスで,プロパティを表示する回復シナリオを選択します。
- 2 [プロパティ] ボタンをクリックします。あるいは、[シナリオ] ボックスのシ ナリオをダブルクリックする方法もあります。[回復シナリオのプロパティ] ダイアログ・ボックスが開き、選択したシナリオの設定に関する読み取り専用 の情報が表示されます。詳細については、1276ページ「回復シナリオのプロパ ティの表示」を参照してください。

回復シナリオの優先順位の設定

実行セッション中に、関連付けられたシナリオが QuickTest によって実行され る順序を指定できます。トリガ・イベントが発生した場合、QuickTest によっ て、[テストの設定] ダイアログ・ボックスの [回復] タブに表示されている 順序に従って該当する回復シナリオが確認されます。

回復シナリオの優先順位を設定するには、次の手順を実行します。

- 1 [シナリオ] ボックスで,優先順位を変更するシナリオを選択します。
- 2 [上に移動] ボタンまたは [下に移動] ボタンをクリックします。選択に従って, 選択したシナリオの優先順位が変更されます。
 - 3 優先順位を変更するシナリオごとに1~2の手順を繰り返します。

テストからの回復シナリオの削除

[テストの設定] ダイアログ・ボックスの[回復] タブを使用して,特定のシ ナリオとテストの間の関連付けを削除できます。テストからシナリオを削除し た後も,まだシナリオ自体は存在していますが,QuickTestによって実行セッ ション中にシナリオが実行されることはありません。





回復シナリオをテストから削除するには、次の手順を実行します。

1 [**シナリオ**] ボックスで,削除するシナリオを選択します。



2 [**削除**] ボタンをクリックします。選択したシナリオは、テストとの関連付け が解除されます。

回復シナリオの有効化と無効化

[テストの設定] ダイアログ・ボックスの [回復] タブでは,特定のシナリオ を有効または無効にしたり,QuickTest による回復シナリオのメカニズムの呼び 出し条件を指定したりできます。特定のシナリオを無効にしても,テストとの 関連付けは残りますが,実行セッション中にそのシナリオがQuickTest によっ て実行されることはありません。そのシナリオは後で有効にできます。

また、回復シナリオを呼び出す条件を指定することもできます。

特定の回復シナリオを有効/無効にするには、次の手順を実行します。

- ▶ シナリオを有効にするには、各シナリオの左側にあるチェック・ボックスを選択します(複数選択も可)。
- ▶ シナリオを無効にするには、各シナリオの左側にあるチェック・ボックスをク リアします。

回復メカニズムを呼び出すか条件を定義するには、次の手順を実行します。

[回復シナリオのアクティブ化] ボックスにある次のオプションのいずれかを 選択します。

- ▶ [各ステップごと]:回復メカニズムはステップごとに起動されます。[各ス テップごと]を選択すると、テストの実行中にパフォーマンスが低下する場 合があります。
- ▶ [エラー発生時]:回復メカニズムはエラーの戻り値を返したステップの後にのみ起動されます。

エラーを返すステップは,多くの場合,例外イベントの発生原因となるス テップとは同じではありません。 たとえば、チェック・ボックスを選択するステップによって、ポップアッ プ・ダイアログ・ボックスが開いたとします。ポップアップ・ダイアログ・ ボックスがトリガ・イベントとして定義されてはいるものの、チェック・ ボックスを選択するステップの実行は成功しているため、QuickTest の処理 は次のステップに進みます。続くいくつかのステップでは、アプリケーショ ンに対する操作の実行を必要としないチェックポイント、関数、その他の条 件ステートメントまたはループ・ステートメントが実行される可能性があり ます。そして、ポップアップ・ダイアログ・ボックスによって実行が妨げら れる操作をアプリケーションに対して実行するようにQuickTestに指示する ステップが登場するには、10個のステートメントを経てようやく、という ことが考えられます。この場合、エラーを返し、回復メカニズムをトリガし てダイアログ・ボックスを閉じるのは、この10番目のステップです。回復 操作が完了した後は、現在のステップではありません。

▶ [**なし**]:回復メカニズムは無効にされます。

ヒント:実行セッション中,テストに関連付けられている特定のシナリオ,またはすべてのシナリオをプログラムの中から有効/無効にすることもできます。 詳細については,1286ページ「プログラムによる回復メカニズムの制御」を参照してください。

すべての新しい テストの標準回復シナリオの設定

[テストの設定] ダイアログ・ボックスの[回復] タブで [標準値に設定] ボ タンをクリックすると、回復シナリオの現在のリストを、すべての新しいテス ト用の標準シナリオに設定できます。設定以降、現在の回復シナリオのリスト に加える変更は現在のテストにのみ影響し、定義した標準設定のリストは変更 されません。

プログラムによる回復メカニズムの制御

Recovery オブジェクトを使用すれば,実行セッション中にプログラムの中で回 復メカニズムを制御できます。たとえば,回復メカニズム全体を有効または無 効にしたり,実行セッションの一部分で特定の回復シナリオを有効または無効 にしたりできます。また,特定の回復シナリオに関するステータス情報を取得 したり,実行セッションの特定の時点で回復メカニズムを明示的に呼び出した りできます。

標準設定では、実行セッション中にエラーが返された場合、QuickTest によって 回復トリガが調べられます。Recovery オブジェクトの Activate メソッドを使用 すると、QuickTest に、実行セッションの特定のステップの後でトリガを調べさ せることができます。たとえば、オブジェクト・プロパティ・チェックポイン トの実行時に、あるプロセスが開いていると、そのチェックポイントが失敗す るとわかっているとします。アプリケーションにおける別の問題である可能性 があるため、こうした開いているプロセスがチェックポイントの成功または失 敗に影響を及ぼさないようにする必要があります。

しかし、チェックポイントの失敗は、実行エラーにはなりません。そのため、 標準設定では、回復メカニズムがオブジェクトの状態によって呼び出されるこ とはありません。オブジェクトのプロパティが特定の状態にあるときに指定の 開いているプロセスを探して閉じる回復シナリオを定義できます。オブジェク トのプロパティの状態は、問題のあるプロセスが開いている場合の値を示しま す。QuickTest に対して、チェックポイントが失敗したときに回復メカニズムを 呼び出させ、問題のプロセスが開いていないか調べさせて、あれば閉じさせ、 失敗したチェックポイントを実行しなおすように指示できます。これにより、 チェックポイントが2回目に実行されるときには、チェックポイントは開かれ ているプロセスによる影響を受けなくなります。

回復オブジェクトとそのメソッドの詳細については,『HP QuickTest Professional Object Model Reference』(英語版)を参照してください。



QuickTest Script Editor を使った作業

QuickTest Script Editor は、複数のテスト・スクリプトおよび関数ライブラリを 同時に開いて編集できるツールです。

本章では、次の項目について説明します。

- ► QuickTest Script Editor について (1288 ページ)
- ▶ [QuickTest Script Editor] ウィンドウについて (1289 ページ)
- ▶ [QuickTest Script Editor] ウィンドウのカスタマイズ (1290 ページ)
- ▶ [テスト]表示枠について(1292ページ)
- ▶ [リソース]表示枠について(1295ページ)
- ▶ 表示領域について(1298ページ)
- ▶ テストを使った作業(1300ページ)
- ▶ 関数ライブラリを使った作業(1304ページ)

QuickTest Script Editor について

QuickTest Script Editor を使用すると、複数のテストおよび関数ライブラリのス クリプトを同時に開いて変更できます。さらに、新しい関数ライブラリを作成 することもできます。ただし、QuickTest Script Editor では、テストのスクリプ トのみを変更できます。新しいテストを作成したり、既存のテスト名、テスト 設定、パラメータ化、またはデータ・テーブル値などの情報を変更したりする ことはできません。

注:

- ➤ QuickTest Script Editor では、QuickTest のテストおよび関数ライブラリでのみ 作業できます。コンポーネントまたはスクリプト・コンポーネントの使用方 法の詳細については、第48章「Business Process Testing を使用した作業」を 参照してください。
- ➤ QuickTest の以前のバージョンで保存したテストをスクリプト・エディタで 開くと、最新のバージョンに更新されます。スクリプト・エディタでテスト を保存したら、以前のバージョンの QuickTest では開くことができません。
- ➤ QuickTest Script Editor によって、UTF-16 識別子が保存する各関数ライブラ リ・ファイル(新規または既存)の開始位置に自動的に追加されます。

詳細については、以下を参照してください。

- ▶ 1300ページ「テストを使った作業」
- ▶ 1304ページ「関数ライブラリを使った作業」

[QuickTest Script Editor] ウィンドウについて

QuickTest Script Editor を開くには, **[スタート**] > **[プログラム**] > **[QuickTest Professional]** > **[Tools**] > **[QuickTest Script Editor**] を選択 します。

[QuickTest Script Editor] ウィンドウの例を次に示します。



[QuickTest Script Editor] ウィンドウには,次の主要な要素があります。

- ▶ [テスト] 表示枠:開いているテストごとにアクションの呼び出しの流れを表示します。
- ▶ [リソース] 表示枠:開いているテスト、そのローカル・アクション、各テスト に関連付けられている関数ライブラリ、および現在開いているすべての関数ラ イブラリのリストを表示します。
- ▶ [表示] 領域:開いているテストと関数ライブラリごとにウィンドウを表示します。

詳細については、以下を参照してください。

- ► 1290 ページ「[QuickTest Script Editor] ウィンドウのカスタマイズ」
- ▶ 1292ページ「[テスト] 表示枠について」
- ▶ 1295ページ「[リソース] 表示枠について」
- ▶ 1298ページ「表示領域について」

[QuickTest Script Editor] ウィンドウのカスタマイズ

[カスタマイズ] ダイアログ・ボックスでは、スクリプト・エディタ・ツール バー、メニュー、およびその他の表示オプションを、ほかの多くの Windows ア プリケーションに似た方法でカスタマイズできます。

[カスタマイズ] ダイアログ・ボックスを開くには、ツールバーまたはメ ニュー・バーで右クリックして [**ユーザ定義**] を選択します。

カスタマイズ		×
コマンド ツールバー キーオ	ボード メニュー オプション	
分類(<u>G</u>):	LAAR(D):	
ファイルド) 編集(E) 表示(A) ツール(T) ウィンドウ(W) ヘルブ(H) 新規メニュー すべてのコマンド	 新規関数ライブラリ(N) テスト(T) 関数ライブラリ(F) 保存(S) すべて保存(V) 印刷(P) 	
言羊糸田:		
2		閉じる

タブをクリックし、要件に応じてスクリプト・エディタをカスタマイズします。

[コマンド] タブ

スクリプト・エディタのツールバーおよびメニューのボタンとコマンドを追加 および移動できます。また、表示されているツールバーおよびメニューから、 ボタンとコマンドを削除することもできます。

[ツールバー] タブ

スクリプト・エディタ・ウィンドウに表示する利用可能なツールバーを選択でき ます。ツールバー・ボタンに対してテキスト・ラベルを表示するかどうかを選択 できます。また,ツールバー表示を標準設定にリセットすることもできます。

[キーボード] タブ

新しいキーボード・ショートカットをツールバーやメニュー・コマンドに割り 当てたり、既存のショートカットを変更および削除したりできます。また、す べてのキーボード・ショートカットを標準設定にリセットすることもできます。

[メニュー] タブ

スクリプト・エディタのウィンドウに表示する利用可能なメニュー,およびコ ンテキスト・メニューに表示するコマンドを選択できます。メニューのアニ メーション設定,および影を表示するかどうかを選択できます。また,表示さ れているメニューを標準設定にリセットすることもできます。

[オプション] タブ

ツールバー・ボタンのツールチップを表示するかどうか,ツールチップにショー トカット・キーを表示するかどうか,およびツールバー・ボタンを大きいアイコ ンとして表示するか,小さいアイコンとして表示するかを選択できます。

[テスト] 表示枠について

[フロー] 表示枠には、現在開いているテストごとにテスト・フロー(アク ション呼び出しのフロー)が表示されます。開いている各テストはツリー内に ノードとして表示され、各ノードには、ローカルのアクション、再利用可能な アクション、および外部アクションに対する呼び出しなど、テストで呼び出さ れたすべてのアクションの階層が含まれます。各テストのアクション呼び出し は、表示領域にある関連するテスト・ウィンドウの[テスト]表示枠にも表示 できます。



[テスト] 表示枠には、次のアイコンが表示されます。

オプション	詳細
<u>F</u>	テスト
<i>i</i>	ローカル・アクションの呼び出し。
3	外部アクションの呼び出し。
<i>6</i>	再利用可能なアクションの呼び出し。
オプション	詳細
------------	--
i 🖗	パスがテストとともに保存されていないアクションの呼び出し。
% .	ループになっているアクションの呼び出し。つまり,テスト・ フロー階層ですでに呼び出されたアクションの呼び出し。

[テスト] 表示枠では、次の操作を実行できます。

- アクションのスクリプトを表示する。アクションをダブルクリックするか、 アクションを右クリックし、[表示]を選択します。表示される各アクションは、テスト・ウィンドウにタブとして表示されます。外部アクションを表示すると、呼び出されたアクションが含まれているテストは、[フロー]表示枠および[リソース]表示枠のツリーに追加され、選択したアクションは、表示領域の新しいテスト・ウィンドウに表示されます。
- 選択したアクションを呼び出すテスト・スクリプトの行を表示する。アクションを右クリックして [アクション呼び出しへ移動] を選択します。アクション呼び出しのスクリプト行が、テスト・ウィンドウの関連するアクション・タブで強調表示されます。
- ▶ テストまたはアクションのプロパティを表示する。テストまたはアクション を右クリックして [プロパティ] を選択します。テストまたはアクションの 名前およびそのパスが表示されます。QuickTest に相対パスとして定義され ている場合は、パスが.¥ <アクションまたは関数ライブラリの名前> と表 示されます。アクションが外部アクションの場合は、[外部] チェック・ ボックスが選択されます。
- ▶ テストを閉じる。テストを右クリックして [閉じる] を選択します。変更内容を保存していない場合は、保存するように促されます。

テストに、存在しないアクションまたは見つからないアクションに対する呼び 出しが含まれている場合も、そのアクションは[フロー]表示枠のツリーに表 示されます。そのアクションを表示しようとすると、アクションを見つけるこ とができないことを通知するエラー・メッセージが表示されます。

ヒント:

▶ [フロー] 表示枠のタイトル・バーで右クリックすると、利用できる表示オ プションを表示して [フロー] 表示枠の表示方法を決定できます。たとえ ば、表示枠を自動的に非表示にしたり、固定したり、閉じたりできます。



▶ [フロービューの切り替え] ツールバー・ボタンをクリックすると, [テスト] 表示枠の表示と非表示を切り替えることができます。

詳細については、1300ページ「テストを使った作業」を参照してください。

[リソース] 表示枠について

[リソース] 表示枠には、現在開いているテストとそのリソース(アクションおよび関連付けられている関数ライブラリ)がすべて表示されます。各テストはツ リー内にノードとして表示され、各ノードには、テストに関連付けられているア クションおよび関数ライブラリが含まれています。現在開いている関数ライブラ リおよびその関数はすべて、表示枠上部の個別のノードにも表示されます。



リソース 表示枠には、次のアイコンか表示	:されます。
----------------------	--------

オプション	詳細
N	開かれている関数ライブラリ
5	関数ライブラリに定義されているパブリック関数
	関数ライブラリに定義されているプライベート関数
<u>I</u>	テスト
<i>i</i>	ローカル・アクション
<i>i</i>	再利用可能なアクション
14	テストに関連付けられている関数ライブラリへのリンク

[リソース] 表示枠では、次の操作を実行できます。

- ▶ 既存の関数ライブラリをテストに関連付ける。関連するテストの [関連付けのある関数ライブラリ] フォルダを右クリックして [既存の関数ライブラリの関連付け] を選択し、関連付ける関数ライブラリ・ファイルを参照します。
- アクティブな関数ライブラリをテストに関連付ける。表示領域のアクティブ なウィンドウが(保存されている)関数ライブラリの場合は、関連するテストの[関連付けのある関数ライブラリ]フォルダを右クリックし、[アク ティブな関数ライブラリの関連付け]を選択します。
- > ローカル・アクションのスクリプトまたは関数ライブラリのコードを表示す る。アクションまたは関数ライブラリをダブルクリックするか、右クリック して [表示] を選択します。表示される各アクションは、テスト・ウィンド ウにタブとして表示され、各関数ライブラリは個別のウィンドウに表示され ます。
- > ローカル・アクションまたは関数ライブラリのプロパティを表示する。アクションまたは関数ライブラリを右クリックして [プロパティ] を選択します。アクションまたは関数ライブラリの名前およびそのパスが表示されます。QuickTest に相対パスとして定義されている場合は、パスが .¥ <アクションまたは関数ライブラリの名前> と表示されます。アクションが再利用可能なアクションの場合は、[再利用可能] チェック・ボックスが選択されます。

- ▶ 関数ライブラリ内の関数の場所を表示する。[開いている関数ライブラリ] フォルダで関数を右クリックし,[関数定義へ移動]を選択します。関数ラ イブラリ・ウィンドウで,関数定義の最初の行が強調表示されます。
- > 関数ライブラリをテストから削除する。関連するテストの [関連付けのある 関数ライブラリ]フォルダで関数ライブラリを右クリックして [関数ライブ ラリを削除]を選択するか,関連するテストの [関連付けのある関数ライブ ラリ]フォルダで関数ライブラリを選択して, DELETE キーを押します。
- ▶ 関数ライブラリを閉じる。[開いている関数ライブラリ] フォルダで関数ラ イブラリを右クリックし,[閉じる]を選択します。変更内容を保存してい ない場合は,保存するように促されます。
- ▶ テストを閉じる。テストを右クリックして [閉じる] を選択します。変更内容を保存していない場合は、保存するように促されます。

ヒント:

<u>1</u>

- ▶ [リソース] 表示枠のタイトル・バーで右クリックすると、利用できる表示 オプションを表示して [リソース] 表示枠の表示方法を決定できます。たと えば、表示枠を自動的に非表示にしたり、固定したり、閉じたりできます。
- ▶ [リソース ビューの切り替え] ツールバー・ボタンをクリックすると、[リ ソース] 表示枠の表示と非表示を切り替えることができます。

詳細については,1300ページ「テストを使った作業」および1304ページ「関数ライブラリを使った作業」を参照してください。

表示領域について

表示領域には、開いているテストまたは関数ライブラリごとに個別のウィンド ウが表示され、各テスト・ウィンドウには、そのテスト内で開いているローカ ル・アクションごとのタブが含まれています。



ヒント: [**ウィンドウ**] メニューのオプションを使用すると、これらのウィンドウを表示領域に配置する方法を決定できます。

表示領域にアクションまたは関数ライブラリを表示するには、[テスト]また は[リソース]表示枠でアクションまたは関数ライブラリをダブルクリックす るか、右クリックして [表示]を選択します。テスト・ウィンドウでは、タブ は開いているローカル・アクションごとに表示されます。[テスト]または [リソース]表示枠で外部アクションへの呼び出しをダブルクリックすると、 呼び出されたアクションが含まれているテストは[テスト]表示枠および[リ ソース]表示枠のツリーに表示され、テストは呼び出されたアクションのタブ とともに表示領域に新しいウィンドウとして表示されます(アクションが含ま れているテストがすでに開いている場合、呼び出されたアクションのタブは、 表示されていない場合はテスト・ウィンドウに追加されます)。

アクションが存在しない場合,または見つからない場合は,そのアクションを 開こうとするとメッセージが表示されます。

開いているローカル・アクションで利用できるすべてのタブが,テスト・ウィ ンドウの下部に表示されるわけではありません。利用できるタブを切り替える には,ウィンドウの下部にある矢印をクリックしてタブを切り替えます。

アクション・タブを選択してからウィンドウ右下の [**アクションを非表示にし ます**] ボタンをクリックすると、ウィンドウからアクションのタブを削除でき ます。アクションは非表示になるだけで終了していないため、加えた変更を保 存するように通知されることはありません。

選択したアクションを呼び出すテスト・スクリプトの行を表示するには、[テ スト]表示枠のツリーでアクションを右クリックし、[**アクション呼び出しへ** 移動]を選択します。アクション呼び出しのスクリプト行が、テスト・ウィン ドウの関連するアクション・タブで強調表示されます。

関数ライブラリ内の関数の場所を表示するには、[リソース]表示枠でツリー の最上位にある[開いている関数ライブラリ]フォルダで関数を右クリック し、[関数定義へ移動]を選択します。関数ライブラリ・ウィンドウで、関数 定義の最初の行が強調表示されます。

[エディタオプション] ダイアログ・ボックスを使用すると([**ツール**] > [**エ ディタオプション**]), テスト・スクリプトと関数ライブラリを, どのように QuickTest Script Editor に表示するかをカスタマイズできます。たとえば, 行番 号を表示するか, またはスクリプトの表示に使用するフォントと色を変更する かを選択できます。[エディタオプション] ダイアログ・ボックスの詳細につ いては, 第27章「エキスパート・ビューおよび関数ライブラリ・ウィンドウ のカスタマイズ」を参照してください。

詳細については,1300 ページ「テストを使った作業」および 1304 ページ「関 数ライブラリを使った作業」を参照してください。



<u>6</u>

テストを使った作業

複数の既存のテストを開き、編集して保存できます。

また,テスト・スクリプトの表示方法をカスタマイズしたり,各テスト内のテ キスト文字列を検索および置換したり,テストを印刷したりできます。詳細に ついては,以下を参照してください。

- ▶ 881ページ「エキスパート・ビューおよび関数ライブラリ・ウィンドウのカス タマイズ」
- ▶ 835 ページ「テキスト文字列の検索」
- ▶ 837 ページ「テキスト文字列の置換」
- ▶ 349ページ「テストの印刷」

テストを開く

ファイル・システムのテスト,および Quality Center プロジェクトに保存されて いるテストを開くことができます。テストは必要な数だけ開くことができま す。テストを開くとツリーに表示され、テスト・ウィンドウの[テスト]表示 枠に、テスト内の最上位レベルのアクションに対する呼び出しがすべて表示さ れます。

ヒント:既存のテストを開くには、ファイル・システム(Windows Explorer) からスクリプト・エディタのウィンドウにドラッグします。最近使用したテス トを開くには、[ファイル]メニューの最近使用したファイルのリストから選 択します。 テストを開くには、次の手順を実行します。

- R
- 必要に応じて、[Quality Center への接続] ボタンをクリックして Quality Center に接続します。Quality Center への接続の詳細については、1324 ページ 「QuickTest の Quality Center への接続」を参照してください。
- 2 次のいずれかの方法でテストを開きます。
 - ▶ [テスト] 表示枠で、開くテストをダブルクリックするか、テストを右ク リックして [表示] を選択します。
 - ▶ [テストを開く] ツールバー・ボタンをクリックするか, [ファイル] > [開 く] > [テスト] を選択します。[テストを開く] ダイアログ・ボックスが 開きます。テストを選択し, [開く] をクリックします。

テスト・スクリプトを表示するだけで変更しない場合は、ダイアログ・ボック ス下部の[読み取り専用モードで開く]チェック・ボックスを選択します。

注:表示領域のアクティブ・ウィンドウに基づいて,[**開**く] ボタンは, [**テ** ストを開く] と [**関数ライブラリを開**く] の間で切り替えられます。[**関数 ライブラリを開く**] ボタンを [**テストを開く**] に変更するには, ボタンの横 にあるドロップダウン矢印をクリックして [**テスト**] を選択するか, 表示領 域でテスト・ウィンドウをクリックします。

表示領域に [<テストのパス>] ウィンドウ開き, [テスト フロー] タブにテ ストのアクション呼び出しが表示されます。テストおよびすべてのアクション は [テスト] 表示枠のツリーに表示され, ローカル・アクションおよび関数ラ イブラリは [リソース] 表示枠のツリーに表示されます。

選択したテストが別のユーザによって開いている場合は、テストがすでに開い ていること、および開いているユーザが通知され、テストは読み取り専用モー ドで開きます。この状況は、QuickTestでテストを開いてから、同じコンピュー タのQuickTest Script Editorで同じテストを開こうとした場合、およびその反対 の操作をした場合にも発生します。さらに、QuickTest Script Editorでテストを 開くと、そのテストはロックされ、そのテストを閉じるまで、ほかのユーザは 変更を加えることができません。

テストの編集

QuickTest Script Editor を使用すると、複数のテスト・スクリプトを同時に編集 できます。テストを編集するには、情報の追加または変更、コピーと貼り付 け、またはほかのテストおよび関数ライブラリから情報のドラッグ・アンド・ ドロップを行います。

注: QuickTest Script Editor でテストを使って作業するときは、新しいテストを 作成したり、既存のテストに新しい名前を付けて保存したりできません。テス ト・スクリプトのみ変更できます。つまり、テストの設定、パラメータ化、 データ・テーブル値などの情報は変更できません。

QuickTest の機能(オブジェクト・リポジトリなど)は、QuickTest Script Editor では利用できないため、正しい構文、形式、およびスペルを使用して、テス ト・スクリプトですべての変更を行う必要があります。

テストを編集するには、次の手順を実行します。

- 1 編集するテスト,および必要に応じて情報のコピー元になるテストを開きま す。また,必要になる可能性がある関数ライブラリも開くことができます。
- 2 必要に応じてテストを編集します。変更内容を保存するまで、アスタリスク
 (*) が編集したテスト・ウィンドウのタイトル・バーに表示されます。

ヒント:



選択したコメントに変換されたテキストをコメントに変換されていないテキストをコメントに変換されていないテキストに変更したり、その反対の操作を行ったりするには、[コメント ブロック] または [ブロックのコメント解除] ツールバー・ボタンを使用するか、または [編集] メニューのオプションを使用します。



選択したテキストをインデントまたはインデント解除するには,[インデント] または[インデント解除]ツールバー・ボタンを使用するか,または[編集] メニューのオプションを使用します。

テストの保存

アクティブなテストを保存することも,開いているすべてのテストと関数ライ ブラリを保存することもできます。

テストを保存するには、次の手順を実行します。

- ▶ [保存] ツールバー・ボタンをクリックするか, [ファイル] > [保存] を選択 してアクティブなテストを保存します。アクティブなテストとは,現在表示領 域でフォーカスが当てられているテスト・ウィンドウのことです。
- ▶ [すべて保存] ツールバー・ボタンをクリックするか, [ファイル] > [すべて保存] を選択して開いているテストおよび関数ライブラリをすべて保存します。

テストを閉じる

テストは, [フロー] 表示枠, [リソース] 表示枠, または表示領域から閉じる ことができます。

テストを閉じるには、次の手順を実行します。

X

[フロー] 表示枠または [リソース] 表示枠で,閉じるテストを右クリックして [**閉じる**] を選択するか,表示領域で閉じるテスト・ウィンドウ上部の [**閉じる**] ボタンをクリックします。テスト・ウィンドウが閉じられ,テストが [フロー] 表示枠および [リソース] 表示枠から削除されます。

注:保存していない変更がある場合は、テストを閉じる前にこれらの変更を保存するように促されます。

関数ライブラリを使った作業

関数ライブラリ・ファイルには、VBScript 関数、サブルーチン、クラス、モジュールなどを含めることが可能で、テストと関連付けて追加の機能を提供できます。QuickTest Script Editor を使用すると、複数の関数ライブラリを開いて編集したり、新しい関数ライブラリを作成したり、関数ライブラリをテストに関連付けたりすることができます。

関数ライブラリ・コードの表示方法をカスタマイズしたり,各関数ライブラリ 内のテキスト文字列を検索および置換したり,関数ライブラリを印刷したりで きます。詳細については,以下を参照してください。

- ▶ 881 ページ「エキスパート・ビューおよび関数ライブラリ・ウィンドウのカス タマイズ」
- ▶ 835ページ「テキスト文字列の検索」
- ▶ 837 ページ「テキスト文字列の置換」
- ▶ 903 ページ「関数ライブラリの印刷」

関数ライブラリを開く

関数ライブラリは、ファイル・システムおよび Quality Center プロジェクトの一 部である関数ライブラリから開くことができます。関数ライブラリは必要な数 だけ開くことができます。QuickTest Script Editor では、.qfl, .vbs, および.txt 関数ライブラリ・ファイルを使用できます。

関数ライブラリを開くと表示領域の関数ライブラリ・ウィンドウに表示され, 関数ライブラリとその関数は,[リソース]表示枠のツリーの最上位にある [**開いている関数ライブラリ**]フォルダに表示されます。関数ライブラリが開 いているテストに関連付けられている場合は,[リソース]表示枠にあるツ リーの[**関連付けのある関数ライブラリ**]フォルダにも関数ライブラリのリン クとしてテストの下に表示されます。

ヒント:既存の関数ライブラリを開くには、ファイル・システム(Windows Explorer)からスクリプト・エディタのウィンドウにドラッグします。最近使用した関数ライブラリを開くには、[ファイル]メニューの最近使用したファ イルのリストから選択します。 関数ライブラリを開くには,次の手順を実行します。

- R
- 必要に応じて、[Quality Center への接続] ボタンをクリックして Quality Center に接続します。Quality Center への接続の詳細については、1324 ページ 「QuickTest の Quality Center への接続」を参照してください。
- 2 次のいずれかの方法で関数ライブラリを開きます。
 - ▶ [リソース] 表示枠で,開く関数ライブラリをダブルクリックするか,関数 ライブラリを右クリックして [表示] を選択します。
 - ▶ [関数ライブラリを開く] ツールバー・ボタンをクリックするか, [ファイ ル] > [開く] > [関数ライブラリ] を選択します。[関数ライブラリを開 く] ダイアログ・ボックスが開きます。関数ライブラリを選択して [開く] をクリックします。

注:表示領域のアクティブ・ウィンドウに基づいて,[開く]ボタンは,[テ ストを開く]と[関数ライブラリを開く]の間で切り替えられます。[テス トを開く]ボタンを[関数ライブラリを開く]に変更するには,ボタンの横 にある矢印をクリックして[関数ライブラリ]を選択するか,表示領域で関 数ライブラリのウィンドウをクリックします。

[<関数ライブラリのパス>] ウィンドウが開き,関数ライブラリが,[リソース] 表示枠のツリーの最上位にある [開いている関数ライブラリ] フォルダに表示されます。

ファイル・システムから別のユーザが開いている関数ライブラリを開くと,そ のユーザによって変更が行われたかどうかが通知され,加えられた変更を許可 するか拒否するかを選択できます。

Quality Center に保存されている関数ライブラリを開き、その関数ライブラリを 別の Quality Center ユーザが開いている場合は、関数ライブラリがロックされて いること、ロックしているユーザ、および関数ライブラリは読み取り専用モー ドで開くことが通知されます。さらに、Quality Center に保存されている関数ラ イブラリを開くと、その関数ライブラリはロックされ、その関数ライブラリを 閉じるまで、ほかのユーザは変更を加えることができません。

関数ライブラリの作成

QuickTest Script Editor では、テストに関連付けることができる新しい関数ライ ブラリを作成できます。

関数ライブラリを作成するには、次の手順を実行します。

- 【関数ライブラリの新規作成】ツールバー・ボタンをクリックし、[ファイル]
 >[新規関数ライブラリ]を選択します。表示領域に関数ライブラリ・ウィンドウが開きます。標準では、関数ライブラリの名前は Library <番頭>です。
- 2 関数ライブラリに必要なコードを入力します。
- 3 [保存] ツールバー・ボタンをクリックするか, [ファイル] > [保存] を選択 して新しい関数ライブラリを保存します。[関数ライブラリを保存] ダイアロ グ・ボックスが開きます。
- **4** 1308 ページ「関数ライブラリの保存」の説明に従って、関数ライブラリを保存 します。

関数ライブラリのテストとの関連付け

アクティブな関数ライブラリ,または任意の既存する関数ライブラリをテスト に関連付けることができます。

関数ライブラリをテストに関連付けるには、次の手順を実行します。

[リソース]表示枠で、関数ライブラリを関連付けるテストの[関連付けのある関数ライブラリ]フォルダを右クリックし、[既存の関数ライブラリの関連付け]を選択します。[関数ライブラリを開く]ダイアログ・ボックスが開きます。

ヒント: アクティブな関数ライブラリを関連付ける場合は,関連するテストの [**関連付けのある関数ライブラリ**]フォルダを右クリックして [**アクティブな 関数ライブラリの関連付け**]を選択します(このオプションは,アクティブな 関数ライブラリを保存していない場合には利用できません)。

2 関連付ける関数ライブラリを参照して選択します。

3 [**開く**] をクリックします。関数ライブラリがテストに関連付けられて、ツ リーの [**関連付けのある関数ライブラリ**] フォルダに関数ライブラリのリンク として表示されます。

関連付けられている関数ライブラリをテストから削除するには、次の手順を実 行します。

[リソース] 表示枠で,関数ライブラリを右クリックして [関数ライブラリを 削除] を選択するか,関数ライブラリを選択して DELETE キーを押します。

関数ライブラリの編集

複数の関数ライブラリの関数コードを編集できます。関数ライブラリを編集す るには、情報の追加または変更、コピーと貼り付け、またはほかの関数ライブ ラリおよびテストから情報のドラッグ・アンド・ドロップを行います。

関数ライブラリを編集するには、次の手順を実行します。

- 必要に応じて、編集する関数ライブラリ、および情報のコピー元の関数ライブラリを開きます。また、必要になる可能性があるテストも開くことができます。
- 2 必要に応じて関数ライブラリを編集します。変更内容を保存するまで、アスタ リスク(*)が編集した関数ライブラリ・ウィンドウのタイトル・バーに表示 されます。

ヒント:



- ▶ 選択したコメントに変換されたテキストをコメントに変換されていないテキストに変更したり、その反対の操作を行ったりするには、[コメントブロック]または[ブロックのコメント解除]ツールバー・ボタンを使用するか、または「編集]メニューのオプションを使用します。
- ▶ 選択したテキストをインデントまたはインデント解除するには、[インデン
 ト]または[インデント解除]ツールバー・ボタンを使用するか、または [編集]メニューのオプションを使用します。

関数ライブラリの保存

アクティブな関数ライブラリを保存したり,関数ライブラリの名前を変更して 別の場所に保存したり,開いているすべての関数ライブラリおよびテストを保 存したりできます。関数ライブラリは,ファイル・システムまたは Quality Center に保存できます。

関数ライブラリを保存するには、次の手順を実行します。

- 1 [保存] ツールバー・ボタンをクリックするか、[ファイル] > [保存] を選択してアクティブな関数ライブラリを保存します。アクティブな関数ライブラリとは、現在表示領域でフォーカスが当てられている関数ライブラリのウィンドウのことです。
 - ▶ [ファイル] > [名前を付けて保存] を選択してアクティブな関数ライブラ リの名前を変更するか,新しい場所に保存します。
 - ▶ [すべて保存] ツールバー・ボタンをクリックするか, [ファイル] > [すべ て保存] を選択して開いている関数ライブラリとテストをすべて保存します。

[関数ライブラリを保存] ダイアログ・ボックスが開きます。

注: Quality Center に接続している場合,表示されるダイアログ・ボックスは, ファイル・システムの場合の標準ダイアログ・ボックスと異なります。これら 2 種類のダイアログ・ボックスは,関連する [関数ライブラリを保存] ダイア ログ・ボックスで, [**ファイル システム**] ボタンおよび [Quality Center] ボ タンをクリックすることで切り替えることができます。

- 2 関数ライブラリの保存先フォルダを選択します。
- 3 関数ライブラリの名前およびファイル拡張子を入力します。QuickTest Script Editor では、.qfl、.vbs、および.txt 関数ライブラリ・ファイルを使用できます。
 - ▶ ファイル・システムにファイルを保存するには、[ファイル名] ボックスに 名前とファイル拡張子を入力して [保存] をクリックします。
 - ➤ Quality Center に接続しているときにファイルを保存するには、[添付名] ボックスに名前とファイル拡張子を入力して[OK]をクリックします。

関数ライブラリを閉じる

関数ライブラリは、[リソース]表示枠または表示領域から閉じることができ ます。

関数ライブラリを閉じるには、次の手順を実行します。

×

[リソース] 表示枠で,閉じる関数ライブラリ([開いている関数ライブラリ] フォルダ内)を右クリックして[閉じる]を選択するか,表示領域で閉じる関 数ライブラリのウィンドウ上部にある[閉じる]ボタンをクリックします。関 数ライブラリのウィンドウが閉じ,関数ライブラリが[リソース]表示枠の [開いている関数ライブラリ]フォルダから削除されます。

注:保存していない変更がある場合は、関数ライブラリを閉じる前にこれらの 変更を保存するように促されます。 第45章・QuickTest Script Editor を使った作業



QuickTest 操作のオートメーション

QuickTest を使用してアプリケーションのテストを自動化するのと同じように, QuickTest Professional のオートメーション・オブジェクト・モデルを使用して QuickTest 操作を自動化できます。QuickTest のオートメーション・オブジェク ト・モデルによって公開されているオブジェクト,メソッド,およびプロパ ティを使用すれば,QuickTest のオプションを設定したりテストを実行したりす るスクリプトを作成して,QuickTest のインタフェースを使用して手動で行う方 法の代わりに,これらを行うプログラムを作成できます。

オートメーション・スクリプトは、同じ作業を複数回実行する場合や、複数の テストを対象に実行する場合、あるいは、特定の環境またはアプリケーション でのニーズに合わせて QuickTest をすばやく設定する場合に特に有用です。

本章では、次の項目について説明します。

- ➤ QuickTest 操作の自動化について(1312ページ)
- ▶ QuickTest オートメーション・スクリプトを使用する条件(1313ページ)
- ▶ オートメーション・スクリプトの設計と実行に使用するプログラミング言語 と開発環境の選択(1314ページ)
- ▶ QuickTest オートメーション・スクリプトの基本要素の学習(1316ページ)
- ▶ オートメーション・スクリプトの生成(1317ページ)
- ▶ QuickTest オートメーション・リファレンスの使用方法(1318 ページ)

QuickTest 操作の自動化について

QuickTest Professional オートメーション・オブジェクト・モデルを使用して QuickTest の操作を自動化するスクリプトを作成できます。QuickTest オートメー ション・オブジェクト・モデルは,別のアプリケーションからQuickTest を制御 できるようにするオブジェクト,メソッド,およびプロパティを提供します。

オートメーションとは

「オートメーション」とは、あるアプリケーション内のソフトウェア・オブ ジェクトを別のアプリケーションからアクセスできるようにする Microsoft 社の 技術です。これらのオブジェクトは、VBScript や VC++ などのスクリプティン グ言語またはプログラミング言語を使用して作成して操作できます。オート メーションを利用することで、アプリケーションの機能をプログラムの中から 制御できるようになります。

「オブジェクト・モデル」とは、システムまたはアプリケーションの実装を構成するソフトウェア・オブジェクト(クラス)を構造化して表したものです。 オブジェクト・モデルは、クラスとインタフェースのセットに加えて、プロパ ティ、メソッド、およびイベント、そしてそれらの関係を定義します。

QuickTest オートメーション・オブジェクト・モデルとは

QuickTest のインタフェースを通じて提供されるほとんどの設定および実行機能 は、オブジェクト、メソッド、およびプロパティを通じて、QuickTest オート メーション・オブジェクト・モデルにおいて何らかの方法で表されます。必ず しも1対1の関係は示せませんが、QuickTest の大半のダイアログ・ボックスに は対応するオートメーション・オブジェクトがあり、ダイアログ・ボックスの 大半のオプションは対応するオブジェクト・プロパティを使用して設定と取得 が可能で、ほとんどのメニュー・コマンドその他の操作は対応するオートメー ション・メソッドがあります。

QuickTest オートメーション・オブジェクト・モデルによって公開されているオ ブジェクト,メソッド,およびプロパティを,ループや条件判断ステートメン トなどの標準のプログラミング要素と組み合わせてスクリプトを設計できます。

オートメーション・スクリプトは、同じ作業を複数回実行する場合や、複数の テストを対象に実行する場合、あるいは、特定の環境またはアプリケーション でのニーズに合わせて QuickTest をすばやく設定する場合に特に有用です。 たとえば、Microsoft Visual Basic からオートメーション・スクリプトを作成およ び実行して、テストに必要なアドインの読み込み、可視モードでの QuickTest の起動、テストを開く操作、[オプション]、[テストの設定]、および[記録と 実行環境設定]ダイアログ・ボックスでの対応する設定、テストの実行、およ びテストの保存ができます。

以後,スクリプトに簡単なループを追加し,1つのスクリプトで上記の操作を 複数のテストに対して実行するようにできます。

また、QuickTest を特定の設定で起動する初期化スクリプトを作成することも可 能です。そうしておけば、テスト担当者全員にこのオートメーション・スクリ プトを使用して QuickTest を起動するように指示し、テスト担当者の全員が必 ず同じ設定で作業を行っていることを保証できます。

QuickTest オートメーション・スクリプトを使用する条件

役に立つ QuickTest オートメーション・スクリプトを作成するには、計画、設 計時間、およびテストが必要です。常に、初期投資と、時間がかかったり煩雑 だったりする作業を自動化することで実現される時間と人的資源の節約とを天 秤にかけなければなりません。

何度も繰り返し実行する必要があったり、定期的に実行する必要のある QuickTest 操作は、QuickTest オートメーション・スクリプトのよい候補です。

次にいくつかの有用な QuickTest オートメーション・スクリプトを示します。

▶ 初期化スクリプト: QuickTest を自動的に起動し、特定の環境でのテストに必要 なオプションおよび設定を指定するスクリプトを作成できます。

- ▶ テストテストの集合に対して繰り返し実施されるスクリプトを記述して、特定の目標を達成できます。例を次に示します。
 - ▶ 値の更新:スクリプトを記述して適切なアドインとともに各テストを開き、 更新されたアプリケーションに対して更新実行モード実行し、すべてのテストの値を更新するときには保存してアプリケーションの更新された値と一致 させることができます。
 - ▶ 既存のテストへの新規オプションの適用: QuickTest の新しいバージョンに アップグレードしたときに,既存のテストに適用したいオプションが新しい バージョンに存在する場合があります。既存のテストをそれぞれ開いて,新 しいオプションのための値を設定し,変更を保存して閉じるというスクリプ トを作成できます。
- ▶ ほかのアプリケーションからの QuickTest の呼び出し: QuickTest オートメーション・スクリプトを実行するオプションやコントロールを持った独自のアプリケーションを設計できます。たとえば、QuickTest に精通していない製品マネージャでも QuickTest の実行予定を立てられるような Web フォームや簡単なWindows インタフェースを作成できます。

オートメーション・スクリプトの設計と実行に使用するプログラミ ング言語と開発環境の選択

オートメーション・スクリプトの作成に使用できるオブジェクト指向プログラ ミング言語がいくつかあります。それぞれの言語に対して,オートメーショ ン・スクリプトの設計と実行に使用できるいくつかの開発環境が提供されてい ます。

オートメーション・スクリプトの作成

QuickTest オートメーション・スクリプトはオートメーションをサポートする任意の言語および開発環境を使用して作成できます。たとえば、次が使用できます。VBScript, JavaScript, Visual Basic, Visual C++, Visual Studio .NET。

タイプ・ライブラリの参照がサポートされている開発環境もあります。**タイ プ・ライブラリ**とは、オブジェクトの記述、インタフェース、その他、オブ ジェクト・モデルの定義を含んでいるバイナリ・ファイルです。 タイプ・ライブラリの参照をサポートする開発環境を選択した場合, Microsoft IntelliSense, 自動ステートメント補完, ステータス・バーのツールチップなど の機能を, スクリプトの作成時に利用することができます。QuickTest オート メーション・オブジェクト・モデルは, QTObjectModel.dll という名前のタイ プ・ライブラリ・ファイルを提供します。このファイルは,

< QuickTest のインストール・フォルダ> ¥bin に格納されています。

タイプ・ライブラリをサポートする環境を選択する場合には、オートメーショ ン・スクリプトの作成および実行を開始する前に、必ず QuickTest タイプ・ラ イブラリを参照するようにします。たとえば、Microsoft Visual Basic を使用して 作業をする場合には、[プロジェクト] > [参照設定] を選択してプロジェク トの [参照設定] ダイアログ・ボックスを開きます。続いて「QuickTest Professional <バージョン> Object Library」(ここで、<バージョン>は現 在インストールされている QuickTest オートメーション・タイプ・ライブラリ のバージョン)を選択します。



オートメーション・スクリプトの実行

オートメーション・スクリプトを実行できるアプリケーションがいくつありま す。Microsoft Windows Script Host を使用すれば、オートメーション・スクリプ トをコマンド・ラインからも実行できます。

たとえば、次のコマンド・ラインを使用してオートメーション・スクリプトを 実行できます。

WScript.exe /E:VBSCRIPT myScript.vbs

QuickTest オートメーション・スクリプトの基本要素の学習

大半のオートメーション・オブジェクト・モデルと同様に, QuickTest オート メーション・オブジェクト・モデルのルート・オブジェクトは, Application オ ブジェクトです。Application オブジェクトは, QuickTest のアプリケーション・ レベルを表します。このオブジェクトを使用して, テスト・オブジェクト (テス ト・ドキュメントを表します), Options オブジェクト ([オプション] ダイアロ グ・ボックスを表します), または Addins コレクション ([アドイン] ダイアロ グ・ボックスのアドインのセットを表します) など, QuickTest の要素を返した り, アドインのロード, QuickTest の起動, テストの起動と保存, および QuickTest の終了などの操作を実行したりできます。

Application オブジェクトによって返されるオブジェクトはそれぞれ他のオブ ジェクトを返したり、オブジェクトに関係する操作を行ったり、オブジェクト に関連付けられているプロパティの取得と設定を行ったりできます。

オートメーション・スクリプトは必ず QuickTest Application オブジェクトの作 成から始まります。オブジェクトを作成しても QuickTest は起動しません。 QuickTest オートメーション・オブジェクト・モデルの他のオブジェクト,メ ソッド,プロパティにアクセスするためのオブジェクトを用意するだけです。

注: 必要があれば、オブジェクトを作成する対象となるリモート QuickTest コ ンピュータ(スクリプトを実行するコンピュータ)を指定することも可能で す。詳細については、オンラインの『QuickTest Automation Object Model Reference』(英語版)の「Running Automation Programs on a Remote Computer」の項を参照してください。

スクリプトの残りの部分は、スクリプトの目的に応じて異なります。QuickTest を 開始する前に、テストに対応するアドインの取得、アドインのロード、QuickTest を可視モードでの開始など、いくつかの操作を行うことができます。

これらの準備を行った後,QuickTest がコンピュータでまだ起動されていなければ,Application.Launchメソッドを使用してQuickTest を起動できます。オートメーション・スクリプトの大半の操作は,Launchメソッドの後に実行します。

オートメーション・プログラムで実行できる操作の詳細については、オンラインの『**HP QuickTest Professional Object Model Reference**』(英語版)を参照してください。このヘルプ・ファイルの詳細については、1318ページ「QuickTest オートメーション・リファレンスの使用方法」を参照してください。

必要な操作を完了したら、あるいは、ロードされているアドインのセットを変 更する場合など、QuickTest をいったん終了して再度起動する操作を実行したい 場合には、Application.Quit メソッドを使用します。

オートメーション・スクリプトの生成

[テストの設定] ダイアログ・ボックスの [プロパティ] タブ, [オプション] ダイアログ・ボックスの [一般] タブ,および, [オブジェクトの認識] ダイ アログ・ボックスのそれぞれに, [**スクリプトの生成**] ボタンがあります。こ のボタンをクリックすると,該当するダイアログ・ボックスの現在の設定を含 んだオートメーション・スクリプト・ファイル (.vbs) が生成されます。

生成されたスクリプトをそのまま実行すれば、スクリプトの生成に使用した QuickTest アプリケーションとまったく同じ設定で QuickTest を起動できます。 また、生成されたファイルから特定の行をコピーして、自分のオートメーショ ン・スクリプトに貼り付けることもできます。

たとえば, [オプション] ダイアログ・ボックスから生成したスクリプトは, 次のようになります。

Dim App 'As Application Set App = CreateObject("QuickTest.Application") App.Launch App.Visible = True App.Options.DisableVORecognition = False App.Options.AutoGenerateWith = False App.Options.WithGenerationLevel = 2 App.Options.TimeToActivateWinAfterPoint = 500

····

App.Options.WindowsApps.NonUniqueListItemRecordMode = "ByName" App.Options.WindowsApps.RecordOwnerDrawnButtonAs = "PushButtons" App.Folders.RemoveAll [スクリプトの作成] ボタン, [オプション], [オブジェクトの認識], [テストの設定] ダイアログ・ボックスで利用できるオプションの詳細については, 第5章「オブジェクトの認識の設定」, 第40章「グローバル・テスト・オプションの設定」, および第41章「個別のテストのオプション設定」を参照してください。

QuickTest オートメーション・リファレンスの使用方法

『QuickTest オートメーション・リファレンス』は、QuickTest オートメーショ ン・オブジェクト・モデルのオブジェクト、メソッド、プロパティに関する詳 細な説明、構文情報、使用例を提供するヘルプ・ファイルです。

『QuickTest オートメーション・リファレンス』は、次の場所から開くことができます。

- ➤ QuickTest プログラム・フォルダ([スタート] > [プログラム] > [QuickTest Professional] > [Documentation] > [QuickTest Automation Reference])
- ► QuickTest のメイン・ヘルプ ([ヘルプ] > [QuickTest Professional ヘルプ] > [QuickTest Advanced References] > [QuickTest Automation])

第 XI 部

その他の HP 製品を使用した作業



Quality Center を使用した作業

アプリケーションの包括的なテストを確実に行うためには、通常、多くのテストを作成して実行する必要があります。HP Quality Center は、品質の集中管理ソリューション(以前の TestDirector)であり、テスト・プロセスの整理と管理に役立ちます。

注:本章に含まれる Quality Center の説明は, Quality Center の現在サポートさ れているすべてのバージョンに適用されます。サポートされている Quality Center のバージョンについては,『**HP QuickTest Professional 最初にお読みくだ さい**』の一覧を参照してください。

本章では、次の項目について説明します。

- ➤ Quality Center を使用した作業について(1322ページ)
- ➤ Quality Center との接続と切断(1323ページ)
- ▶ QuickTest の Quality Center との統合(1332 ページ)
- ▶ Quality Center プロジェクトへのテストの保存(1333 ページ)
- ▶ Quality Center プロジェクトからテストを開く(1334ページ)
- ▶ テンプレート・テストを使用した作業(1339ページ)
- ➤ Quality Center プロジェクトに格納されているテストの QuickTest からの実行 (1346 ページ)
- ▶ QuickTest でのテストのバージョン管理(1348ページ)
- ➤ Quality Center テストの実行に関する設定(1359ページ)

Quality Center を使用した作業について

QuickTest は HP の品質集中管理ソリューションである Quality Center と統合で きます。Quality Center は、アプリケーションの機能のあらゆる側面を網羅する あらゆる種類のテスト(QuickTest テスト、ビジネス・プロセス・テスト・マ ニュアル・テスト、他の HP 製品作成されるテストなど)で構成されるプロ ジェクトの保守を支援します。プロジェクトでは、アプリケーションの特定の テスト要件を満たすように各テストを設計します。また、プロジェクトの目標 を達成するため、プロジェクトで行うテストを個別のグループにまとめます。

Quality Center には、テストのスケジュール設定と実行、結果の収集、結果の分析、テストのバージョン管理などを行うための直感的かつ効率的なメソッドが 用意されています。不具合を追跡するためのシステムも備わっているため、不 具合の検出から解決に至るまで、不具合を厳密に監視することができます。

Quality Center プロジェクトとは、特定のテスト・プロセスに関係するデータを 収集し、格納するためのデータベースです。QuickTest から Quality Center プロ ジェクトにアクセスするには、Quality Center がインストールされているローカ ルまたはリモートの Web サーバに接続する必要があります。QuickTest が Quality Center に接続されている場合は、テストを作成し、このテストを Quality Center プロジェクトに保存できます。テストの実行後には、Quality Center で結 果を確認できます。

Quality Center で QuickTest を使用するには、次のアクセス許可が必要です。

- ➤ Quality Center キャッシュ・フォルダ (Quality Center のクライアント側にあります)の読み取りおよび書き込みの許可
- ➤ Quality Center インストール・フォルダの QuickTest アドインへの完全な読み 取りおよび書き込み許可

Quality Center を使って作業をしている場合, Quality Center プロジェクトに添付 されている外部ファイルをテストに関連付けることができます。すべてのテス トまたは1つのテストに対して,外部ファイルを関連付けることができます。 たとえば,新規テストのための標準リポジトリ・モードとして,共有オブジェ クト・リポジトリ・モードを設定するとします。この場合,Quality Center に格 納されている特定のオブジェクト・リポジトリを使用するようにQuickTest に 指定できます。 すべてのテストで外部ファイルを使用するように指定する方法の詳細について は、第40章「グローバル・テスト・オプションの設定」を参照してください。 1つのテストで外部ファイルを使用するように指定する方法の詳細については、 第41章「個別のテストのオプション設定」を参照してください。

不具合は,発生するたび自動的に,または手作業でQuickTestの[テスト結果] ウィンドウからQuality Center プロジェクトに直接報告できます。Quality Center プロジェクトへの自動または手作業による不具合の報告については,989ペー ジ「実行セッション中に検出された不具合の送信」を参照してください。

Quality Center での作業の詳細については,『**HP** Quality Center User's Guide』 (英語版)を参照してください。QuickTest と Quality Center の統合に関する最新 情報およびヒントについては,「**HP** QuickTest Professional 最初にお読みくださ い」([スタート] > [プログラム] > [QuickTest Professional] > [最初に お読みください]から利用できます)を参照してください。

Quality Center との接続と切断

QuickTest および Quality Center の両方で作業する場合, QuickTest を Quality Center プロジェクトに接続できます。

テスト・プロセスのどの時点でも、Quality Center プロジェクトと QuickTest と を接続したり切断したりできます。ただし、Quality Center から QuickTest テス トを開いているときや、QuickTest で Quality Center の共有リソース(共有オブ ジェクト・リポジトリやデータ・テーブル・ファイルなど)を使用していると きは、Quality Center と QuickTest との接続を切断しないでください。

注: Quality Center の現在サポートされているバージョンには接続できます。 Quality Center のサポートされているバージョンについては,『**HP QuickTest Professional 最初にお読みください**』の一覧を参照してください。詳細について は, 1332 ページ「Quality Center 接続アドイン」を参照してください。

QuickTest の Quality Center への接続

接続プロセスには2つの段階があります。まず,QuickTest をローカルまたはリ モートのQuality Center サーバに接続します。このサーバによって,QuickTest とQuality Center プロジェクト間の接続が処理されます。

次に、ログインし、QuickTestのアクセス対象となるプロジェクトを選択しま す。このプロジェクトには、テスト対象アプリケーションのテストと実行セッ ションの情報が保存されます。プロジェクトは、パスワードで保護されるた め、ユーザ名とパスワードを指定する必要があります。

QuickTest を Quality Center サーバに接続するには、次の手順を実行します。

R

 ファイル]> [Quality Center への接続] を選択するか, [Quality Center への接続] ツールバー・ボタンをクリックします。[Quality Center への接続 -サーバへの接続] ダイアログ・ボックスが表示されます。

🌏 Quality Cent	ter への接続 - サーバへの接続	×
手順 1: サーバに:	接続する	_
サーバ URL:]
	例: http://server/qcbin	
□ 起動時にサ	ーバに再接続する(<u>R</u>)	
	☑ 接続(C) 閉じる(L) ヘルプ	

2 [サーバURL] ボックスに, Quality Center がインストールされている Web サー バの URL アドレスを入力します。

注: ローカル・エリア・ネットワーク(LAN)または広域エリア・ネットワーク(WAN)を介してアクセス可能な Quality Center サーバを選択できます。

3 次回の QuickTest の起動時に, Quality Center サーバに自動的に再接続するには, [起動時にサーバに再接続する] チェック・ボックスを選択します。 4 [接続] をクリックします。

接続手順の第2段階は、接続先の Quality Center サーバのバージョンによって異なります。詳細については次の項を参照してください。

- ▶ 1325 ページ「9.x サーバを使用したプロジェクトへの接続」
- ▶ 1327 ページ「8.2.x サーバを使用したプロジェクトへの接続」

9.x サーバを使用したプロジェクトへの接続

Quality Center 9.x サーバに接続した場合は、サーバにログインし、接続先のド メインとプロジェクトを指定します。

9.x サーバを使用してプロジェクトに接続するには、次の手順を実行します。

1 Quality Center 9.x サーバに接続すると, [Quality Center への接続] ダイアロ グ・ボックスが開きます。

🌏 Quality Cente	er への接続				×
手順 1: サーバは持	轰続する ――――				
サーバ URL:	http://kim8.cool.i	deac.co.jp:8888/	qcbin		
▼ 起動時(さ 手順2:ユーザ情報	ナーバに再接続する() 報を認証する	<u>5</u>) 	×	切断(<u>D</u>)	
ユーザ名(山):	admin				
パスワード(W):			_		
□ 起動時()謬	2証する(工)			1211 (<u>A</u>)	
手順 3: プロジェク	トにログインする ――				
ドメイン(<u>M</u>):					V
プロジェクト(<u>P</u>):					7
□ 起動時につ	クロジェクトにログインマ	∮る(<u>G</u>)	*	ログイン(1)	
		<u>閉じるの</u>		ヘルプ	

Quality Center サーバの名前が読み取り専用形式で [サーバ URL] ボックスに表示されます。

2 [ユーザ名] ボックスに, Quality Center ユーザ名を入力します。

- 3 [パスワード] ボックスに, Quality Center のパスワードを入力します。
- 4 [認証] をクリックし、Quality Center サーバに対してユーザ情報を認証させます。
 ユーザ情報の認証が完了すると、[ユーザ情報を認証する] 領域のエディット・ボックスが読み取り専用形式で表示されます。[認証] ボタンが [ユーザを変更] ボタンに変わります。

ヒント:同じ Quality Center サーバに別のユーザ名でログインするには,[ユー **ザを変更**]をクリックし,新しいユーザ名とパスワードを入力して,再び[**認** 証]をクリックします。

- 5 [**ドメイン**] ボックスで, Quality Center プロジェクトが保存されているドメイ ンを選択します。接続する権限を持っているドメインのみが表示されます。
- 6 [**プロジェクト**] ボックスで,作業対象のプロジェクトを選択します。ユーザ が定義されているプロジェクトのみが表示されます。
- 7 [**ログイン**] をクリックします。
- 8 次回の QuickTest の起動時に, Quality Center サーバに自動的に再接続するには, [起動時にサーバに再接続する] チェック・ボックスを選択します。
- 9 [起動時にサーバに再接続する] チェック・ボックスをオンにすると, [起動時 に認証する] チェック・ボックスが有効になります。次回の QuickTest の起動 時に,ユーザ情報を自動的に認証させるには, [起動時に認証する] チェック・ ボックスを選択します。
- 10 [起動時に認証する] チェック・ボックスをオンにすると、[起動時にプロジェ クトにログインする] チェック・ボックスが有効になります。起動時に選択し たプロジェクトにログインするには、[起動時にプロジェクトにログインする] チェック・ボックスを選択します。
- 11 [閉じる] をクリックし, [Quality Center への接続] ダイアログ・ボックスを閉 じます。ステータス・バーに, QuickTest が Quality Center プロジェクトに接続 されていることを示す Quality Center アイコンが表示されます。



ヒント:現在の Quality Center 接続を表示するには、ステータス・バーの Quality Center アイコンにマウス・ポインタを合わせます。QuickTest が接続さ れている Quality Center サーバ名とプロジェクトがツールチップに表示されま す。[Quality Center への接続] ダイアログ・ボックスを開くには、Quality Center アイコンをダブルクリックします。

8.2.x サーバを使用したプロジェクトへの接続

Quality Center 8.2 Service Pack 1 サーバに接続した場合,接続先となるドメイン およびプロジェクトを指定して,プロジェクトにログインします。

8.2 サーバを使用してプロジェクトに接続するには、次の手順を実行します。

Quality Center 8.2 Service Pack 1 サーバに接続すると、[Quality Center への接続] ダイアログ・ボックスが開きます。

🌏 Quality Cent	er への接続 - ブロジェクトへの接続 🛛 🗙
サーバへの接続	
サーバ URL:	http://venommp.mercury.co.il/qcbin
□ 起動時にサ	ーバに再接続する(S) 🛛 切断(D)
プロジェクトへの接続	続
ドメイン(<u>M</u>):	ACCEPTANCE_TEST
プロジェクト(P):	AT_APM
ユーザ名(<u>U</u>):	
パスワードWD:	
	☑ 接続(N)
□ 起動時にブ	ロジェクトに再接続する(R)
┏ 起動時に再	i接続するためパスワードを(保存する(<u>い</u>)
	閉じる(©) ヘルプ

Quality Center サーバの名前が読み取り専用形式で [サーバ URL] ボックスに表示されます。

- 2 [**ドメイン**] ボックスで, Quality Center プロジェクトが保存されているドメイ ンを選択します。
- 3 [**プロジェクト**] ボックスで,作業対象のプロジェクトを選択します。
- 4 [**ユーザ名**] ボックスに, 選択したプロジェクトを開くためのユーザ名を入力 します。
- 5 [パスワード] ボックスに, 選択したプロジェクトのパスワードを入力します。
- 6 [接続] をクリックして,選択したプロジェクトに QuickTest を接続します。

注:アクセス権を持っていないプロジェクトを選択した場合は、[**接続**]をク リックすると通知が表示されます。

選択したプロジェクトへの接続が確立されると, [**プロジェクトへの接続**]領 域内のフィールドが読み取り専用形式で表示されます。

- 7 次回の QuickTest の起動時に, Quality Center サーバに自動的に再接続するには, [起動時にサーバに再接続する] チェック・ボックスを選択します。
- 8 [起動時にサーバに再接続する] チェック・ボックスをオンにすると, [起動時 にプロジェクトに再接続する] チェック・ボックスが有効になります。起動時 に選択したプロジェクトに自動的に接続するには, [起動時にプロジェクトに 再接続する] チェック・ボックスを選択します。
- 9 [起動時にプロジェクトに再接続する] チェック・ボックスをオンにすると、
 [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・ボックスが
 有効になります。起動時に再接続するためのパスワードを保存するには、この
 [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いと、起動時に QuickTest が Quality Center に接続すると きに、パスワードを入力するよう指示が求められます。
10 [閉じる] をクリックし, [Quality Center への接続 - プロジェクトへの接続] ダ イアログ・ボックスを閉じます。ステータス・バーに, QuickTest が Quality Center プロジェクトに接続されていることを示す Quality Center アイコンが表示 されます。

準備完了

ヒント:現在の Quality Center 接続を表示するには、ステータス・バーの Quality Center アイコンにマウス・ポインタを合わせます。QuickTest が接続さ れている Quality Center サーバ名とプロジェクトがツールチップに表示されま す。[Quality Center への接続]ダイアログ・ボックスを開くには Quality Center アイコンをダブルクリックします。

Quality Center からの QuickTest の切断

Quality Center プロジェクトまたは Quality Center サーバからいつでも QuickTest を切断できます。最初にプロジェクトから切断せずに, QuickTest を Quality Center サーバから切断すると, QuickTest からプロジェクト・データベースへの 接続は自動的に切断されます。

注: Quality Center から切断するときに Quality Center テストまたは共有ファイル(共有オブジェクト・リポジトリやデータ・テーブル・ファイルなど)が開いていると、これらは QuickTest によって閉じられます。

8.2 サーバから QuickTest を切断するには、次の手順を実行します。



ファイル] > [Quality Center への接続] を選択するか, [Quality Center への接続] ツールバー・ボタンをクリックします。[Quality Center への接続 - プロジェクトへの接続] ダイアログ・ボックスが表示されます。

😪 Quality Center への接続 - ブロジェクトへの接続 🛛 🛛 🔀		
サーバへの接続		
サーバ URL:	http://venommp.mercury.co.il/qcbin	
▼ 起動時にサ	ーバに再接続する(S) 🛛 切断(D)	
プロジェクトへの接	続	
ドメイン(M):	ACCEPTANCE_TEST	
プロジェクト(<u>P</u>):	AT_APM	
ユーザ名(山):	rkshuk	
パスワード(<u>W</u>):	****	
	区 切断(N)	
▼ 起動時にプ	ロジェクトに再接続する(<u>R</u>)	
☑ 起動時に再	接続するためパスワードを採存するいの	
	閉じる(C) ヘルプ	

- 2 選択したプロジェクトから QuickTest を切断するには、[プロジェクトへの接続]領域で[切断]をクリックします。
- 3 選択した Quality Center サーバから QuickTest を切断するには、 [サーバへの接続] 領域で [切断] をクリックします。
- 4 [閉じる] をクリックし, [Quality Center への接続 プロジェクトへの接続] ダ イアログ・ボックスを閉じます。

9.x サーバから QuickTest を切断するには、次の手順を実行します。



ファイル] > [Quality Center への接続] を選択するか、[Quality Center への接続] ツールバー・ボタンをクリックします。[Quality Center への接続] ダイアログ・ボックスが表示されます。

💽 Quality Center への接続	X
手順 1: サーバに接続する	
サーバ URL: http://kim8.cool.ideac.co.jp:8888	3/qcbin
✓ 起動時にサーバに再接続する(≦) 手順 2: ユーザ情報を認証する	☑ 切断(D)
ユーザ名(U): jideac	
パスワード()): *********	
▶ 起動時に認証する(1)	🛛 ユーザを変更(<u>A</u>)
手順 3: プロジェクトにログインする	
ドメイン(M): DEFAULT	_
プロジェクト(P): QualityCenter_Demo	7
✓ 起動時にプロジェクトにログインする(G)	🛛 🛛 🖾 🖾
	/ルプ

- 2 選択したプロジェクトから QuickTest を切断するには、[手順 3: プロジェクトに ログインする]領域で[ログアウト]をクリックします。
- 3 選択した Quality Center サーバから QuickTest を切断するには、[手順1:サーバ に接続する]領域で[切断]をクリックします。

ヒント:同じ Quality Center サーバに別のユーザ名でログインするには,[**ユ**-**ザの変更**]をクリックし,新しいユーザ名とパスワードを入力して,再度[認 証]をクリックします。

4 [**閉じる**] をクリックし, [Quality Center への接続] ダイアログ・ボックスを閉 じます。

QuickTest の Quality Center との統合

QuickTest の Quality Center との統合により, Quality Center プロジェクト内の ファイルの格納やアクセスが可能になりました。また, QCUtil オブジェクトを 使用して, Quality Center オープン・テスト・アーキテクチャ API に用意されて いる広範な種類の機能にアクセスできます。

Quality Center 接続アドイン

Quality Center 接続アドインを使用して, QuickTest を Quality Center と統合しま す。このアドインは, [Quality Center への接続] ダイアログ・ボックスを使用 して QuickTest を Quality Center に接続するときに, QuickTest コンピュータに自 動的にインストールされます。このアドインを手動でインストールするには, Quality Center アドインのページ (Quality Center メイン画面から使用できます) から [Mercury Quality Center 接続] を選択します。

現在コンピュータにインストールされている Quality Center 接続アドインのバー ジョンを表示するには、[ヘルプ] > [QuickTest Professional のバージョン 情報]を選択し、[製品情報] ボタンをクリックします。詳細については、94 ページ「製品情報の表示」を参照してください。

Quality Center との統合

最も基礎的なこととして, QuickTest の Quality Center との統合により, QuickTest が Quality Center と接続している場合, Quality Center プロジェクトの QuickTest テストおよび関数ライブラリの格納やアクセスが可能になります。

また、テストおよび関数ライブラリで QCUtil オブジェクトを使用することによ り、Quality Center OTA(オープン・テスト・アーキテクチャ)の全機能に対す るアクセスや使用が可能になります(OTA は以前の TestDirector OTA または TDOTA です)。これにより、Quality Center データベースに不具合を直接報告す るなど、実行セッション中の統合操作を自動化することができます。詳細につ いては、『HP QuickTest Professional Object Model Reference』(英語版)のユー ティリティの項、および『Quality Center オープン・テスト・アーキテクチャ』 のマニュアルを参照してください。

QuickTest 自動スクリプトの TDOTA オブジェクトを使用して, Quality Center OTA にアクセスすることもできます。詳細については,『QuickTest Automation Reference』([ヘルプ] > [QuickTest Professional ヘルプ] > [QuickTest Advanced References] > [QuickTest Automation] を選択) を参照してください。

Quality Center プロジェクトへのテストの保存

H

QuickTest を Quality Center プロジェクトに接続すると、QuickTest で新しいテストを作成し、そのテストをプロジェクトに直接保存することができます。テストを保存するには、テスト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。これにより、各サブジェクトに作成したテストを容易に追跡し、テストの計画と作成の進行状況を素早く確認することができます。

テストを Quality Center プロジェクトに保存するには、次の手順を実行します。

- **1** Quality Center サーバおよびプロジェクトに接続します。詳細については、1324 ページ「QuickTest の Quality Center への接続」を参照してください。
- 2 QuickTest で [保存] をクリックするか, [ファイル] > [保存] を選択して, テストを保存します。[Quality Center にテストを保存] ダイアログ・ボックス が開き, テスト計画ツリーが表示されます。

🧟 Quality Center (こテストを保存 📃 🗖 🖡			
カテゴリ: Subject 🖵 🕞			ファイル システム
⊡ © Subject	· テスト名	ステータス	作成済み
⊕ 🍅 BPT			
🗄 👘 BPT Resources			
⊕ i BPT_Tutorial			
🕂 💼 Compiled Modules			
🕂 💼 Cruises			
🕂 👘 Esti			
🕂 💼 Flight Application (BPT Demo)			
🕂 🧰 Flight Reservation			
🕂 🧰 Itinerary			
🕀 🧰 Mercury Tours Site			
🕂 💼 Profiling			
🕂 🧰 Request_Component			
テスト名: Parameter			ок
テストの種類: QuickTest テスト	ActiveScree	en ファイルの保存	閉じる

[Quality Center にテストを保存] ダイアログ・ボックスは, QuickTest が Quality Center プロジェクトに接続されている場合にだけ開きます。

テストをファイル・システムに直接保存するには, [ファイル システム] ボタ ンをクリックし, [QuickTest テストの保存] ダイアログ・ボックスを開きます。 [QuickTest テストを保存] ダイアログ・ボックスで [Quality Center] ボタン をクリックすると, [Quality Center にテストを保存] ダイアログ・ボックスに 戻ることができます)。

- 3 テスト計画ツリーの中で、該当するサブジェクト・フォルダを選択します。ツリーを展開して下位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。下位レベルを折りたたむには、開いているフォルダをダブルクリックします。
- 4 [テスト名] ボックスにテストの名前を入力します。テストを識別しやすい、 分かりやすい名前を付けます。テストの名前は(パスも含めて)220文字までです。名前の先頭および最後にスペースを使うことはできません。また、次の 文字は使用できません。¥/:*?"<>|%';
- 5 テストと一緒に ActiveScreen ファイルを保存する場合, [ActiveScreen ファイ ルの保存] が選択されていることを確認します。このチェック・ボックスをク リアすると, ActiveScreen ファイルが削除されるため, ActiveScreen オプション を使ったテストの編集ができなくなります。詳細については, 346ページ「テ ストの保存」を参照してください。
- 6 [OK] をクリックし、テストを保存してダイアログ・ボックスを閉じます。
 QuickTest がテストを保存している間は、ステータス・バーのテキストが変わります。

次回 Quality Center 起動時に,新しいテストが Quality Center のテスト計画ツ リーに表示されます。詳細については,『**HP Quality Center User's Guide**』(英 語版)を参照してください。

Quality Center プロジェクトからテストを開く

QuickTest を Quality Center プロジェクトに接続すると, Quality Center プロジェ クトに含まれている QuickTest テストを開くことができます。ファイル・シス テム内の実際の格納場所ではなく, テスト計画ツリーの中でテストを探しま す。[ファイル] メニューの最近使用したテストのリストからテストを開くこ ともできます。 バージョン・コントロール・サポートの Quality Center プロジェクトのテストを開 くと、アイコンはテストのバージョン・コントロール・ステータスを示します。

Quality Center プロジェクトからテストを開くには、次の手順を実行します。

- **1** Quality Center サーバおよびプロジェクトに接続します。詳細については、1324 ページ「QuickTest の Quality Center への接続」を参照してください。
- 2 QuickTest で、「開く] をクリックするか、「ファイル] > [開く] > [テスト] を選択し、テストを開きます。[Quality Center からテストを開く] ダイアログ・ ボックスが開き、テスト計画ツリーが表示されます。

Quality Center からテストを開く			
カテゴリ: Subject 👽			ファイル システム
📋 💼 BPT 💽	テスト名	ステータス	作成済み
庄 👝 BPT Resources 🔤	🌠 Departing And Arriving L	Ready	2005/04/
⊕ BPT_Tutorial	📝 Number Of Passengers	Ready	2005/05/
🚊 🦳 Compiled Modules	Trip Type	Ready	2005/06/
🕂 💼 Cruises			
🕂 💼 Esti			
🕂 🦳 Flight Application (BPT Den			
🚊 📇 Flight Reservation			
Book Flight			
Flight Confirmation			
Flight Cost			
Flight Finder			
Select Flight			
🕂 🧰 Itinerary			
テスト名:			0K
テストの種類: QuickTest テスト	- 読み取り専り	用モード で開く	閉じる

[Quality Center からテストを開く] ダイアログ・ボックスは, QuickTest が Quality Center プロジェクトに接続されている場合にだけ表示されます。

注: Quality Center に接続している状態でテストをファイル・システムから直接 開くには、[ファイル システム] ボタンをクリックし、[テストを開く] ダイア ログ・ボックスを表示します([テストを開く] ダイアログ・ボックスで [Quality Center] ボタンをクリックすると、[Quality Center からテストを開く] ダイアログ・ボックスに戻ることができます)。 3 テスト計画ツリーで該当するサブジェクトをクリックします。ツリーを展開して下位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するテストが [Quality Center からテストを開く] ダイアログ・ボックスの右表示枠に表示されます。

- ▶ テストがバージョン・コントロール・サポート付きの Quality Center プロ ジェクトに格納されると、[テスト名]の横に表示されるアイコンは、テス トのバージョン・コントロール・ステータスを示します。詳細については、 1338ページ「バージョン・コントロール・サポート付きの Quality Center プ ロジェクトからテストを開く」を参照してください。
- ▶ [テスト名] カラムに表示されているのは、選択したサブジェクトに属する テストの名前です。
- ▶ [ステータス] カラムは、各テストが Design 段階かテスト実行のための Ready 段階かを示します。標準設定では、QuickTest から Quality Center プロ ジェクトに保存されたテストは Design と表示されます。ステータスを変更 できるのは Quality Center クライアントだけです。
- ▶ [作成済み] カラムは,各テストが作成された日付を示します。
- 4 [**テスト名**] リストでテストを選択します。テストが読み取り専用の [**テスト** 名] ボックスに表示されます。
- 5 読み取り専用モードでテストを開くには, [読み取り専用モードで開く] チェック・ボックスを選択します。
- 6 [OK] をクリックし、テストを開きます。

QuickTest がテストをダウンロードして開くのに応じて、実行される操作がステータス・バーに表示されます。

テストを開くと、QuickTest タイトル・バーに、「[Quality Center]」という文字 列と、サブジェクトのフルパスおよびテスト名が表示されます。次に例を示し ます。

[Quality Center] Subject¥System¥qa_test1

次の場合、テストは読み取り専用モードで開きます。

- ▶ [読み取り専用モードで開く] を選択した場合
- ▶ 現在(バージョン・コントロール・サポートのプロジェクトの) Quality Center バージョン・コントロール・データベースにチェック・インしている テストを開いた場合
- ▶ 現在 (バージョン・コントロール・サポートのプロジェクトの) ほかのユー ザがチェック・アウトしているテストを開いた場合

詳細については、1338ページ「バージョン・コントロール・サポート付きの Quality Center プロジェクトからテストを開く」を参照してください。

最近使用したテストのリストからテストを開く

[ファイル] メニューの最新ファイル・リストから Quality Center テストを開く こともできます。Quality Center プロジェクトのテストを選択したときに、現在 QuickTest が Quality Center に接続されていないか、テストの正しいプロジェク トに接続されていないと、[Quality Center プロジェクトへの接続] ダイアログ・ ボックスが表示され、正しいサーバ、プロジェクト、および前回当該コン ピュータでテストを開いたユーザの名前が表示されます。

Quality Center プロジェクトへの接続		
サーバ: http://labm1i18n02:8080/gcbi	n	
プロジェクト: QualityCenter_Demo		
ユーザ名(山):	OK.	
パスワード(P):	キャンセル	

また,前回異なる Quality Center ユーザ名を使用して編集したテストを開くこと を選択した場合も,[Quality Center プロジェクトへの接続]ダイアログ・ボッ クスが開きます。表示されているユーザ名を使ってログインすることも、 [**キャンセル**]をクリックして現在のユーザ名でログインすることもできます。

バージョン・コントロール・サポート付きの Quality Center プロジェ クトからテストを開く

[**開く**] ツールバー・ボタンをクリックするか, [**ファイル**] > [**開**く] > [**テ スト**] を選択してバージョン・コントロールがサポートされている Quality Center プロジェクトからテストを開くと, [Quality Center プロジェクトから QuickTest テストを開く] ダイアログ・ボックスには, 選択したサブジェクトの 各テストのバージョン・コントロール・ステータスを示すアイコンが表示され ます。

バージョン・コントロール・サポートの Quality Center プロジェクトからテスト を開くと、テストは現在のバージョン・コントロール・ステータスに応じて、 読み書きモードまたは読み取り専用モードで開きます。

次の表に, バージョン・コントロール・ステータスのアイコンと各ステータス のオープン・モードを示します。

アイコン	説明	オープン・モード
<なし>	このテストは,現在バージョン・コントロー ル・データベースにチェック・インしています。	読み取り専用
a	このテストは,現在ユーザがチェック・アウ トしています。	読み取り/書き込み
	このテストは,現在ほかのユーザがチェッ ク・アウトしています。	読み取り専用
Θ	現在古いバージョンのテストをユーザのコン ピュータで開いています。	現状のまま

バージョン・コントロールの Quality Center プロジェクトに格納されているテストを使った作業の詳細については、1348ページ「QuickTest でのテストのバージョン管理」を参照してください。

テンプレート・テストを使用した作業

テンプレート・テストは, Quality Center で作成されるすべての QuickTest テストの土台となります。テンプレート・テストは,標準のテスト設定を含む QuickTest テストです。たとえば,テンプレート・テストでは,QuickTest アド イン,関連付けられている関数ライブラリ,テストに関連付けられている回復 シナリオを指定します。これらのテスト設定は,QuickTest の[テストの設定] ダイアログ・ボックス([**ファイル**] > [**設定**])で変更できます。

標準のテスト設定以外に、テンプレート・テストには Quality Center で作成され るすべての新規の QuickTest テストに追加したい、任意のコメントやステップ を含めることもできます。たとえば、テンプレート・テストに関連付けられて いるアドインの種類をユーザに伝えるコメントを追加したり、すべてのテスト の開始時に特定の Web ページまたはアプリケーションを開くステップを追加し たりできます。テンプレート・テストに追加したステップやコメントは、その テンプレート・テストを土台として Quality Center で作成されるすべての新規の テストに追加されます。

Quality Center 用の QuickTest Professional アドインをインストールするとき, Quality Center クライアントごとに標準設定のテンプレート・テストがインス トールされます。この標準設定のテンプレート・テストに変更を加えたり,さ まざまなテスト設定が関連付けられたカスタマイズ・テンプレート・テストを 作成したりできます。

テンプレート・テストはすべて(Quality Center クライアントに置かれる標準設定のテンプレート・テストを除き)Quality Center プロジェクトに保存されます。各ユーザのローカル・コンピュータにコピーする必要はありません。これにより、ユーザは必要な場合に各自の標準設定のローカル・テンプレート・テストをカスタマイズできる一方で、グローバルに管理されたテンプレート・テストにもアクセスできます。詳細については、1340ページ「新しいテンプレート・テストを使用した作業」を参照してください。

特定のテンプレート・テストを土台とするテストが Quality Center から実行され ると、テスト内の定義に従って、関連付けられているアドインが QuickTest に よって自動的にロードされ、必要な設定が適用されます。

標準設定のテンプレート・テストを使用した作業

Quality Center 用の QuickTest アドインをインストールすると、サポートされて いるすべての QuickTest バージョンに対応した標準設定のテンプレート・テス トが、コンピュータの < Quality Center 用 QuickTest アドイン・フォルダ> ¥bin¥Templates フォルダ (C:¥Program Files¥HP¥QuickTest Add-in for Quality Center¥bin¥Templates¥Template90 など) にインストールされます。

Quality Center ユーザが Quality Center で QuickTest テストを新規に作成すると, ユーザが別のテンプレート・テストを選択しない限り,インストールされてい る QuickTest バージョンに対応した標準設定のテンプレート・テストが自動的 にテストに関連付けられます。詳細については,1342ページ「Quality Center で の QuickTest テストの作成」を参照してください。

Quality Center 用の QuickTest アドインとともに標準設定でインストールされる テンプレート・テストには、変更を加えることができます。標準設定のテンプ レート・テストはローカルにインストールされるため、テンプレート・テスト に加えた変更は自分のコンピュータで(Quality Center クライアントを使用し て)作成したテストにのみ適用されます。あるいは、次項の説明に従って、テ ンプレート・テストを新規に作成することもできます。

標準設定のテンプレート・テストを Quality Center 内の新しいテストに適用する 方法の詳細については, 1342 ページ「Quality Center での QuickTest テストの作 成」を参照してください。

新しいテンプレート・テストを使用した作業

新しいテンプレート・テストを作成すると、それらは Quality Center プロジェクトに格納され、その Quality Center プロジェクトで作成される新しい QuickTest テストの土台としてすべての Quality Center ユーザが使用できるようになります。

テンプレート・テストを複数作成し、それぞれを特定のテスト目的に使用する ことができます。たとえば、ActiveX コントロールを使用する Web アプリケー ションをテストする QuickTest テスト向けのテンプレート・テストを作成し、 それとは別に、標準の Windows アプリケーションをテストする QuickTest テス ト用のテンプレート・テストを作成できます。この場合、前者のテンプレー ト・テストには ActiveX と Web アドインを関連付けます。後者のテンプレー ト・テストには、QuickTest アドインを一切関連付けずに、テストの対象となる Windows アプリケーションを指定します。このほか、必要に応じてそれぞれの テンプレート・テストのテスト設定に変更を加えることもできます。 作成した各テンプレート・テストは, ActiveX_Web_Addins_Template や Std_Windows_Template_Test のように, 用途を明確に示す分かりやすい名前を 付けて保存できます。そのようにすることで, ユーザは Quality Center で QuickTest テストを作成するときに適切なテンプレート・テストを選択できます。

注:特定の QuickTest アドインの関連付を行うテンプレート・テストを定義す るときは、テストを最終的に実行する QuickTest コンピュータに、そのアドイ ンが実際にインストールされていることを確かめておきます。アドインがイン ストールされていないと、テストを実行したときに、QuickTest が必要なアドイ ンをロードすることができず、テストが失敗することがあります。Quality Center からの QuickTest テストの実行の詳細については、Quality Center のマ ニュアルを参照してください。

テンプレート・テストの新規作成

テンプレート・テストを作成するには、まず、で必要なテスト設定を持つ空の テストを QuickTest 作成します。次に、Quality Center プロジェクトのテスト計 画モジュールで QuickTest テストを参照し、それを「テンプレート・テスト」 として指定します。

QuickTest でテストを保存するときは、用途を明確に示すわかりやすい名前を付けます。たとえば、テンプレート・テストを、ActiveX と Web のアドインを新しいテストに関連付ける目的に使用する場合、ActiveX_Web_Addins_Templateという名前を付けることができます。

ヒント: Quality Center のテスト計画ツリー(テスト計画モジュール)で、テン プレート・テスト用の特別なフォルダを作成することもできます。このように すれば、ほかのユーザが Quality Center で QuickTest テストを新規に作成すると きに、関連するテンプレート・テストをすぐに見つけることができます。

テンプレート・テストを作成するには、次の手順を実行します。

QuickTest での作業:

- 必要なアドインをロードした状態で QuickTest を開きます。QuickTest アドイン のロードの詳細については、『HP QuickTest Professional アドイン・ガイド』の QuickTest アドインのロードの項を参照してください。
- 2 [テストの設定] ダイアログ・ボックス([**ファイル**] > [設定]) で必要な設 定を定義します。詳細については、1189ページ「[テストの設定] ダイアロ グ・ボックスの使用」を参照してください。
- **3** このテンプレート・テストを土台とするすべてのテストにコメントやステップ を追加したい場合は、それらを追加します。
- 4 [保存] ボタンをクリックするか, [ファイル] > [保存] を選択してテストを 保存します。[Quality Center にテストを保存] ダイアログ・ボックスが表示さ れます用途を明確に示すわかりやすい名前を付けて, テストを Quality Center プ ロジェクトに保存します。詳細については, 1333 ページ「Quality Center プロ ジェクトへのテストの保存」を参照してください。

Quality Center での作業:

_
1

H

- 5 Quality Center でプロジェクトを開き,サイドバーの [**テスト計画**] ボタンをク リックしてテスト計画モジュールを開き,手順4で保存したテストを参照します。
- **6** テストを右クリックし, [**テンプレート テストとしてマーク**]を選択します。. テストがテンプレート・テストに変換されます。
- 7 必要に応じて、手順1から6を繰り返して追加のテンプレート・テストを作成 します。

Quality Center での QuickTest テストの作成

Quality Center では、テスト計画モジュールで QuickTest テストを作成します。 QuickTest テストを作成するときは、テンプレート・テストをそのテストに適用 します。QuickTest クライアントに格納されている標準設定のテンプレート・テ ストを洗濯することも、Quality Center プロジェクトに保存されているテンプ レート・テストを選択することもできます。

Quality Center プロジェクトにテンプレート・テストをまったく保存していない 場合や, [テンプレート] ボックスで [**くなし>**] を選択した場合(1334 ペー ジに示した [テストの新規作成] ダイアログ・ボックスにあります), Quality Center では, Quality Center クライアントに「Quality Center 用 QuickTest アド イン」と一緒にインストールされたテンプレート・テストに定義されている設 定が使用されます。詳細については、1340ページ「標準設定のテンプレート・ テストを使用した作業」を参照してください。そうではなく、少なくとも1つ のテンプレート・テストが Quality Center プロジェクトに保存されている場合 は、QuickTest テストの新規作成時にはそのテンプレート・テストを選択できま す。詳細については、1340ページ「新しいテンプレート・テストを使用した作 業」を参照してください。

注: Quality Center で QuickTest テストを作成するときは、テストに関連付ける QuickTest アドインを指定するテンプレート・テストを選択する必要がありま す。アドインを指定するテンプレートを選択しないと、必要な QuickTest アド インが実行セッション中にロードされません。

新しい QuickTest テストでは,選択したテンプレート・テストに定義されてい るすべての設定が使用されます。テストが Quality Center から実行されたとき は,[テストの設定]ダイアログ・ボックスに指定されている設定が QuickTest によって使用され,必要な QuickTest アドインが自動的にロードされます。

注:次の手順は, Quality Center でテンプレート・テストを使用してテストを作 成する方法を示します。この手順は,使用する Quality Center のバージョンに よって異なる場合があります。Quality Center でテストを新規に作成するための 最新の手順については,『HP Quality Center User's Guide』(英語版)を参照し てください。 Quality Center でテンプレート・テストを使用してテストを作成するには、次の手順を実行します。



- 1 Quality Center で、サイドバーの [**テスト計画**] ボタン をクリックして、テスト 計画モジュールを開きます。
- 2 テスト計画ツリーで、フォルダを選択します。
- 3 [テストの新規作成] ボタンをクリックするか, [テスト] > [テストの新規作 成] を選択します。[テストの新規作成] ダイアログ・ボックスが開きます。

テストの新規作成	×
テストのタイ MANUAL	-
テスト名:	
テンプレートなし>	
OK キャンセル ヘルプ	

注: Quality Center アドインまたは QuickTest Professional アドインがコン ピュータにインストールされている場合にのみ, [テンプレート] ボックスが 表示されます。[テンプレート] ボックスが表示されない場合, Quality Center アドインを QuickTest Professional DVD からインストールするか, QuickTest Professional アドインを [その他の Quality Center アドイン] ページ ([Quality Center オプション] ウィンドウ, または [ヘルプ] > [アドインページ] から 開きます) からインストールする必要があります。

- 4 [テストのタイプ] リストから, [QUICKTEST_TEST] を選択します。
- 5 [テスト名] ボックスに、英字(Roman)、数字、アンダスコア(必要ならば) を使用してテストの名前を入力します。テストの名前は(パスも含めて)220 文字までです。名前の先頭および最後にスペースを使うことはできません。ま た、次の文字は使用できません。
 ¥ /:*?"<>|%';
- 6 [**テンプレート**] ボックスの参照ボタンをクリックします。[テストの選択] ダ イアログ・ボックスが開きます。

7 テンプレート・テストが格納されているフォルダを展開します。

Subject BPT BPT Resources BPT_Tutorial Compiled Modules Cruise Cancellation Cruises Fight Application (BPT Demo) Flight Reservation Itinerary Profiling Request_Component



- 8 新しいテストの土台とするテンプレート・テストを選択し, [**追加**] ボタンを クリックします。[テストの選択] ダイアログ・ボックスが閉じ, 選択したテ ンプレート・テストが [**テンプレート**] ボックス([テストの新規作成] ダイ アログ・ボックス内) に表示されます。
- 9 [テストの新規作成] ダイアログ・ボックスで, [OK] をクリックします。テ ンプレート・テストに定義されているテスト設定を持つ,新しいテストが作成 されます。
- 10 新しいテストが, テスト計画ツリー内の選択したサブジェクト・フォルダの下 に表示されます。

注: [必須フィールド] ダイアログ・ボックスが表示された場合は,必要な値 を設定して [OK] をクリックします。詳細については,『HP Quality Center Administrator's Guide』(英語版)を参照してください。

11 テストの作成を続けます。Quality Center でのテストの作成の詳細については, 『**HP Quality Center User's Guide**』(英語版)を参照してください。

Quality Center プロジェクトに格納されているテストの QuickTest からの実行

QuickTest では, Quality Center プロジェクトからテストを実行し,実行結果を プロジェクトに保存できます。実行結果を保存するには,結果を格納する実行 セッションとテスト・セットの名前を指定します。

テストの実行結果を Quality Center プロジェクトに保存するには、次の手順を 実行します。

1 QuickTest で、[実行] ボタンをクリックするか、[オートメーション] > [実 行] を選択します。[実行] ダイアログ・ボックスが表示されます。

Į	
	結果保管場所 入力パラメータ
	実行結果の書き込み先:
	● Quality Center プロジェクト内の新規実行結果(N)
	プロジェクト名: QualityCenter_Demo
	実行名旧): 実行 3-14 16-24-56
	テスト セット(S): default 💌 インスタンス(L): 1 💌
	○ 一時実行結果フォルダ (プロジェクト(こ未(保存)(工)

2 [プロジェクト名] ボックスには現在接続している Quality Center プロジェクト が表示されます。

テストの実行結果を Quality Center プロジェクトに保存するには、標準設定の実行名を受け入れるか、別の名前をボックスに入力します。

3 標準設定のテスト・セットを受け入れるか,別のテスト・セットを参照して, 選択します。 4 テスト・セットにテストのインスタンスが複数ある場合には,結果を保存する テストのインスタンスを [インスタンス] ボックスに指定します。

注:「テスト・セット」とは、特定のテスト目標を達成するために選択された テストで構成されたグループです。たとえば、アプリケーションのユーザ・イ ンタフェースや、負荷をかけた状態のアプリケーションのパフォーマンスをテ ストするテスト・セットを作成できます。テスト・セットの定義は、Quality Center のテスト実行モードで行います。詳細については、Quality Center のマ ニュアルを参照してください。

テストを実行し,前のテスト実行の結果を上書きするには,[**一時実行結果** フォルダ(プロジェクトに未保存)]オプションを選択します。

注: QuickTest では, すべてのテストの一時的なテスト実行結果が %TMP%¥TempResults に格納されます。[一時実行結果フォルダ(プロジェク トに未保存)] オプションのテキスト・ボックスに表示されるパスは読み取り 専用のため,変更できません。

5 [OK] をクリックします。[実行] ダイアログ・ボックスが閉じ, QuickTest に よってテストの実行が開始されます。QuickTest によるテストの実行が進行する と同時に、キーワード・ビューの各ステップが強調表示されます。

テストの実行が停止すると, [オプション] ダイアログ・ボックスの [実行] タブで [**テスト実行終了時に結果を表示する**] チェック・ボックスをクリアし ていないかぎり, [テスト結果] ウィンドウが表示されます。[オプション] ダ イアログ・ボックスの詳細については, 第40章「グローバル・テスト・オプ ションの設定」を参照してください。 テストの実行が停止すると、ステータス・バーに「**アップロードしています**」 と表示されます。アップロード処理が完了すると、[テスト結果]ウィンドウ が表示されます。

注:不具合は,発生するたび自動的に,または手作業でQuickTestの[テスト結果] ウィンドウからQuality Center プロジェクトに直接報告できます。詳細については,989ページ「実行セッション中に検出された不具合の送信」を参照してください。

QuickTest でのテストのバージョン管理

バージョン・コントロールがサポートされている Quality Center プロジェクトに QuickTest を接続すると、以前のバージョンを維持したまま、自動化されたテス ト・スクリプトの更新や変更ができます。これにより、各テストに加えた変更 の追跡、テストのバージョン間の変更の確認、テストの旧バージョンの復元が 容易になります。

バージョン・コントロール・サポート付きのプロジェクトにテストを保存する と、バージョン・コントロール・データベースにテストが追加されます。バー ジョン・コントロール・データベースにテストをチェック・イン、チェック・ アウトすることで、テストのバージョンが管理されます。

最新バージョンのテストは, Quality Center のテスト・リポジトリにあり, Quality Center によってすべてのテストの実行に使用されます。

注:

- ▶ バージョン・コントロールがサポートされている Quality Center プロジェク トを利用するには、バージョン・コントロールのソフトウェアと Quality Center のバージョン・コントロール・アドインをインストールする必要があ ります。詳細については、Quality Center のマニュアルを参照してください。
- ▶ [バージョン] メニュー・オプション群は、バージョン・コントロールがサポートされている Quality Center プロジェクトのデータベースに接続して Quality Center テストを開いている場合にだけ使用できます。

バージョン・コントロール・データベースへのテストの追加

バージョン・コントロールがサポートされている Quality Center プロジェクトに [名前を付けて保存] を使って新規のテストを保存すると、QuickTest によって 自動的にテストがプロジェクトに保存され、バージョン・コントロール・デー タベースにバージョン番号 1.1.1 としてチェック・インされ、その後作業を続 けられるようにチェック・アウトされます。

QuickTest ステータス・バーにはそのつど各操作が表示されます。ただし,既存 のテストに変更を保存しても、そのテストをチェック・インすることにはなり ません。テストを保存して閉じても、チェック・インを選択するまではテスト はチェック・アウトしたままです。詳細については、1352ページ「バージョ ン・コントロール・データベースへのテストのチェック・イン」を参照してく ださい。

バージョン・コントロール・データベースからのテストのチェック・アウト

[**ファイル**] > [**開く**] > [**テスト**] を選択して,現在バージョン・コント ロール・データベースにチェック・インされているテストを開くと,テストは 読み取り専用モードで開きます。 **注**: [Quality Center からテストを開く] ダイアログ・ボックスには、プロジェ クトの各テストのバージョン・コントロール・ステータスを示すアイコンが表 示されます。詳細については、1334ページ「Quality Center プロジェクトからテ ストを開く」を参照してください。

チェック・インしたテストを確認できます。また、テストを実行して結果を表 示することもできます。

テストを変更するには、テストをチェック・アウトする必要があります。テス トをチェック・アウトすると、テストは Quality Center によってユーザの固有の チェック・アウト・ディレクトリ(初めてテストをチェック・アウトするとき に自動的に作成されます)にコピーされ、プロジェクト・データベース内のテ ストがロックされます。これにより、テストに加えた変更を Quality Center プロ ジェクトのほかのユーザに上書きされないようにします。ただし、この場合で も、ほかのユーザはデータベースにチェック・インされているテストの最新 バージョンを実行できます。

テストを保存して閉じることはできますが, Quality Center データベースにテス トを戻すまでテストはロックされています。テストを解放するには, テストを チェック・インするかチェック・アウト操作を取り消します。テストのチェッ ク・インの詳細については, 1352ページ「バージョン・コントロール・データ ベースへのテストのチェック・イン」を参照してください。チェック・アウト の取り消しの詳細については, 1358ページ「チェック・アウト操作の取り消 し」を参照してください。

標準では、チェック・アウト・オプションはテストの最新バージョンにアクセスします。以前のバージョンのテストもチェック・アウトできます。詳細については、1354ページ「[バージョンの履歴]ダイアログ・ボックスの使用方法」を参照してください。

最新バージョンのテストをチェック・アウトするには、次の手順を実行します。

1 チェック・アウトするテストを開きます。詳細については、1334ページ 「Quality Center プロジェクトからテストを開く」を参照してください。

注:開いたテストが現在チェック・インされていることを確認します。自分が すでにチェック・アウトしているテストを開くと,[チェックアウト]オプ ションは無効になります。ほかのユーザによってチェック・アウトされている テストを開くと,[バージョンの履歴]オプションを除くすべての [Quality Center バージョンコントロール]オプション群が無効になります。

2 [ファイル] > [Quality Center バージョンコントロール] > [チェック アウ
 ト] を選択します。[チェック アウト] ダイアログ・ボックスが開き、チェック・アウトされるテスト・バージョンが表示されます。

ቻェック アウト	
₩~-9%eV:	1.1.4
אעאב	
	ОК # +уtл

- 3 [コメント] ボックスに変更の詳細を入力します。
- 4 [OK] をクリックします。読み取り専用のテストが閉じて,書き込み可能テストとして再び自動的に開きます。

5 必要に応じてテストを表示または編集します。

注: テストをチェック・インせずに変更を保存して閉じることができますが, テストをチェック・インするまで, ほかの Quality Center ユーザは加えられた変 更を利用できません。変更を加えたテストをチェック・インしない場合は, チェック・アウトを取り消すことができます。テストのチェック・インの詳細 については, 1352 ページ「バージョン・コントロール・データベースへのテス トのチェック・イン」を参照してください。チェック・アウトの取り消しの詳 細については, 1358 ページ「チェック・アウト操作の取り消し」を参照してく ださい。

バージョン・コントロール・データベースへのテストのチェック・イン

あるテストがチェック・アウトされていても,Quality Center ユーザは以前に チェック・インされたバージョンを実行できます。たとえば、テストのバー ジョン 1.2.3 をチェック・アウトし、いくつか変更を加えてテストを保存する とします。バージョン 1.2.4 (または別の番号を割り当てる)としてテストを バージョン・コントロール・データベースに再びチェック・インするまで、 Quality Center ユーザはバージョン 1.2.3 を引き続き実行できます。

テストの変更が終了し, Quality Center ユーザに新しいバージョンを使用しても らう準備が整ったら,バージョン・コントロール・データベースにテストを チェック・インします。

注: Quality Center データベースにテストをチェック・インしたくない場合は, チェック・アウト操作を取り消すことができます。詳細については,1358ペー ジ「チェック・アウト操作の取り消し」を参照してください。

テストをバージョン・コントロール・データベースにチェック・インして戻す と, Quality Center はチェック・アウト・ディレクトリからテストのコピーを削 除し, Quality Center プロジェクトのほかのユーザがそのバージョンのテストを 利用できるように,データベース内のテストのロックを解除します。 現在開いているテストをチェック・インするには、次の手順を実行します。

現在開いているテストを自分がチェック・アウトしていることを確認します。
 詳細については、1354ページ「テストのバージョン情報の表示」を参照してください。

注:開いているテストが現在チェック・インされていると,[**チェック イン**] オプションは無効になります。ほかのユーザによってチェック・アウトされて いるテストを開くと,[**バージョンの履歴**]オプションを除くすべての [Quality Center **バージョンコントロール**]オプション群が無効になります。

2 [ファイル] > [Quality Center バージョン コントロール] > [チェック イン] を選択します。[チェック イン] ダイアログ・ボックスが表示されます。

ቻェック イン	
∦°−ິ9°∎ິນ:	1.1.2
axon	
	•
🔲 ቻェックアウト	のままにしておく
	OK #+ンセル

3 標準設定の新しいバージョン番号を受け入れて手順7に進むか、参照ボタンを クリックしてユーザ定義のバージョン番号を指定します。参照ボタンをクリッ クすると、[チェックイン]ダイアログ・ボックスが開きます。

ቻェック イン				×
	1 ≑	1	5	
		OK _ ₹דיזל	l	

- 4 バージョン番号を、手作業またはバージョン番号の各要素の横にある上向き矢印と下向き矢印を使って変更します。最初の要素に入力できる数字は1~900です。第2、第3の要素に入力できる数字は1~999です。バージョン・コントロール・データベースにある対象のテストの最新バージョン番号よりも低いバージョン番号は入力できません。
- 5 [OK] をクリックしてバージョン番号を保存し, [チェックイン] ダイアロ グ・ボックスを閉じます。
- 6 テストのチェック・アウト時に変更の詳細を入力した場合、その詳細は [コメント] ボックスに表示されます。コメントの入力や修正はこのボックスで行えます。
- 7 [OK] をクリックし, テストをチェック・インします。テストが閉じて, 読み 取り専用のテストとして再び自動的に開きます。

[バージョンの履歴] ダイアログ・ボックスの使用方法

[バージョンの履歴] ダイアログ・ボックスを使って,現在開いているテスト に関するバージョン情報の表示や,以前のバージョンのテストの表示および取 得が行えます。

テストのバージョン情報の表示

Quality Center バージョン・コントロール・データベースに格納されている任意の開いているテストのバージョン情報を,現在のステータスに関係なく表示できます。

テストの [バージョンの履歴] ダイアログ・ボックスを開くには,テストを開き [ファイル] > [Quality Center バージョン コントロール] > [バージョン の履歴] を選択します。

■ パージョンの履歴				
テストは現在パージョンコントロールで [チェックアウト] 状態にあります。以前のパージョンを [取 チェックアウトを元に戻す] 得] または [チェックアウト するには、まずテストを [チェックイン] 状態にしてくたもい。				
			ቻェック イン	
[[*=**a*]	7~#*			
1.1.3	Admin	7/4/2004	17:22:08	
1.1.2	Admin	7/4/2004	17:21:30	
1.1.1	Admin	7/4/2004	17:21:12	
אַעאַד				
Add checkpoint to Test				
			F	
			•	

[バージョンの履歴] ダイアログ・ボックスには次の情報が表示されます。

[**テスト名**]: 現在開いているテストの名前です。

[テストのステータス]: テストのステータスです。次のいずれかになります。

- ▶ [チェック イン]: このテストは、現在バージョン・コントロール・データ ベースにチェック・インされています。テストは現在読み取り専用形式で開 いています。テストをチェック・アウトして編集できます。
- ▶ [チェックアウト]: このテストは,現在ユーザ自身がチェック・アウトしています。テストは現在読み書き可能な形式で開いています。
- ▶ [<ほかのユーザ>によるチェックアウト]: このテストは,現在ほかの ユーザによってチェック・アウトされています。テストは現在読み取り専用 形式で開いています。示されているユーザがテストをチェック・インするま で,テストのチェック・アウトや編集はできません。

[**開いているテストのバージョン**]:現在,お使いの QuickTest コンピュータで 開いているテストのバージョンです。

[バージョン詳細]:テストのバージョンの詳細です。

- ▶ [**バージョン**]: テストのバージョンの一覧です。
- ▶ [ユーザ]:各バージョンのチェック・インをしたユーザです。
- ▶ [日付]:各バージョンがチェック・インされた日です。[時刻]:各バージョンがチェック・インされた時刻です。

[**コメント**]: 選択したテスト・バージョンがチェック・インされたときに入力 されたコメントです。

以前のバージョンのテストを対象とした作業

以前のバージョンのテストは,読み取り専用モードで表示するか,チェック・ アウトして最新バージョンとしてチェック・インできます。

以前のバージョンのテストを表示するには、次の手順を実行します。

- Quality Center テストを開きます。最新バージョンのテストが開きます。詳細については、1334ページ「Quality Center プロジェクトからテストを開く」を参照してください。
- 2 [ファイル] > [Quality Center バージョン コントロール] > [バージョンの 履歴] を選択します。[バージョンの履歴] ダイアログ・ボックスが表示されます。
- 3 表示するテストのバージョンを [バージョン詳細] リストから選択します。
- 4 [バージョンの取得] ボタンをクリックします。テストがチェック・アウトさ れていないため読み取り専用モードで開く旨を知らせるメッセージが表示され ます。
- 5 [OK] をクリックし, QuickTest メッセージを閉じます。選択したバージョンが 読み取り専用モードで開きます。

ヒント:

- ▶ 現在 QuickTest で開いているバージョン番号を確認するには、[バージョンの履歴] ダイアログ・ボックスの [開いているテストのバージョン] の値を調べます。
- ▶ [バージョンの取得]オプションを使用して読み取り専用モードで以前の バージョンを開いたら、[ファイル] > [Quality Center バージョンコント ロール] > [チェック アウト]を選択して、開いているテストをチェック・ アウトできます。これは、[バージョンの履歴]ダイアログ・ボックスで [チェック アウト]ボタンを使用するのと同等の機能です。

以前のバージョンのテストをチェック・アウトするには、次の手順を実行します。

- Quality Center テストを開きます。最新バージョンのテストが開きます。詳細については、1334ページ「Quality Center プロジェクトからテストを開く」を参照してください。
- ファイル] > [Quality Center バージョン コントロール] > [バージョンの 履歴] を選択します。[バージョンの履歴] ダイアログ・ボックスが表示され ます。
- 3 表示するテストのバージョンを [バージョン詳細] リストから選択します。
- 4 [チェック アウト] ボタンをクリックします。確認メッセージが開きます。
- 5 以前のバージョンのテストをチェック・アウトすることを確認します。
 [チェックアウト]ダイアログ・ボックスが開き、チェック・アウトされるテ スト・バージョンが表示されます。

ቻェック アウト	
∦∿–ອ%ອ⊃:	1.1.4
אַעאַר	4
	OK キャンセル

6[**コメント**]ボックスに変更の詳細を入力します。

- **7** [**OK**] をクリックします。開いていたテストが閉じ,選択したバージョンが書 き込み可能テストとして開きます。
- 8 必要に応じてテストを表示または編集します。
- 9 最新バージョンとしてテストを Quality Center データベースにチェック・インするには、[ファイル] > [Quality Center バージョンコントロール] > [チェック イン] を選択します。変更したテストを Quality Center にアップロードしない場合は、[ファイル] > [Quality Center バージョンコントロール] > [チェック アウト作業を元に戻す] を選択します。

テストのチェック・インの詳細については,1352ページ「バージョン・コント ロール・データベースへのテストのチェック・イン」を参照してください。 チェック・アウトの取り消しの詳細については,1358ページ「チェック・アウ ト操作の取り消し」を参照してください。

チェック・アウト操作の取り消し

テストをチェック・アウトした後に,変更を加えたテストを Quality Center に アップロードしないと決定した場合は,ほかの Quality Center ユーザがチェッ ク・アウトできるように,チェック・アウト操作を取り消す必要があります。

チェック・アウト操作を取り消すには、次の手順を実行します。

- 1 チェック・アウトしたテストをまだ開いていない場合は開きます。
- 2 [ファイル] > [Quality Center バージョン コントロール] > [チェック アウト作業を元に戻す] を選択します。
- 3 [はい] をクリックし、チェック・アウト操作の取り消しを確定します。チェック・アウト操作が取り消されます。チェック・アウトしたテストが閉じ、前回 チェック・インしたバージョンが読み取り専用モードで再び開きます。

Quality Center テストの実行に関する設定

Quality Center データベースに格納されている QuickTest テストの実行は, QuickTest, またはコンピュータにインストールされている Quality Center クラ イアント,あるいはリモートの Quality Center クライアントまたはサーバを介し て行います。Quality Center は,QuickTest テストを実行するときに関連アドイン 一覧を使用して,テストに適したアドインを QuickTest コンピュータにロード します。詳細については,1193ページ「関連アドインの変更」および1339 ページ「テンプレート・テストを使用した作業」を参照してください。

注:

QuickTest コンピュータがログオフまたはロックされている場合, Quality Center から QuickTest テストを実行することはできません。

標準設定では、Quality Center がテストを実行するために QuickTest をアクティ ブにすると、QuickTest は非表示モードで起動して実行されます。これはパ フォーマンスの向上に役立ちます。この設定は、QuickTest リモート・エージェ ントで変更できます。詳細については、1361ページ「QuickTest リモート・ エージェントの設定」を参照してください。

Quality Center テストを QuickTest コンピュータで実行するとき,失敗したス テップに関する不具合を報告するよう指示できます。また,[QuickTest テスト 結果]ウィンドウから Quality Center に手作業で不具合を送信できます。詳細に ついては,989ページ「実行セッション中に検出された不具合の送信」を参照 してください。

QuickTest テストを実行するようにリモートの Quality Center クライアントに指示 する前に, QuickTest アプリケーションを使用する権限を Quality Center に与える 必要があります。リモート・エージェントの設定の表示や修正もできます。

Quality Center に対する QuickTest コンピュータでのテストの実行許可

セキュリティ上の理由により,QuickTest アプリケーションへのリモート・アク セスは許可されていません。(またはほかのリモート・アクセスのクライアン ト)に対してテストを開いて実行する許可を与えるには,[**ほかの HP 製品でテ ストまたはコンポーネントを実行可能にする**]オプションを選択する必要があ ります。 **注**: QuickTest テストを Quality Center からリモート実行する必要があり, QuickTest が Windows XP Service Pack 2, Windows 2003 Serve, または Windows Vista にインストールされている場合は,先に DCOM のアクセス許可を変更し てファイアウォール・ポートを開く必要があります。詳細については,HP Software Knowledge Base にアクセスして,記事番号 **43245** を検索してください。

また、QuickTest テストを Quality Center からリモートで実行する必要があり、 QuickTest が Windows Vista にインストールされている場合、最初に Quality Center に接続する前に、Windows のコントロール・パネルの [ユーザー アカ ウント] を使用して、ユーザー・アカウント・コントロール(UAC)を無効に する必要があります。詳細については、『HP QuickTest Professional インストー ル・ガイド』を参照してください。

リモートの Quality Center クライアントが QuickTest コンピュータでテストを 実行可能にするには,次の手順を実行します。

- 1 QuickTest を開きます。
- 2 [ツール] > [オプション] を選択するか, [オプション] ツールバー・ボタン をクリックします。[オプション] ダイアログ・ボックスが開きます。
- 3 [実行] タブをクリックします。
- 4 [他の HP 製品でテストまたはコンポーネントを実行可能にする] チェック・ ボックスを選択します。

このオプションの詳細については,1181ページ「テストの実行オプションの設 定」を参照してください。

ヒント: Quality Center から QuickTest テストにアクセスするには, Quality Center の QuickTest アドインが Quality Center クライアント・コンピュータにイ ンストールされている必要があります。このアドインの詳細については, QuickTest Professional アドイン画面 (Quality Center のメイン画面からアクセス できます)を参照してください。

QuickTest リモート・エージェントの設定

Quality Center から QuickTest テストまたはビジネス・プロセス・テストを実行 すると, QuickTest コンピュータで QuickTest リモート・エージェントが開きま す。テストが Quality Center などのリモートのアプリケーションによって実行さ れた場合の QuickTest の動作は, QuickTest リモート・エージェントによって決 まります。

いつでも [リモート エージェントの設定] ダイアログ・ボックスを開いて, Quality Center がコンピュータでテストを実行するときに QuickTest アプリケー ションが使用している設定の表示や変更ができます。 [リモートエージェントの設定] ダイアログ・ボックスを開くには,次の手順 を実行します。

🚟 5:11 PM

- 【スタート】> [プログラム] > [QuickTest Professional] > [Tools] > [Remote Agent] を選択します。リモート・エージェントが開き、リモート・ エージェントのアイコンがタスクバー・トレイに表示されます。
 - 2 リモート・エージェント・アイコンを右クリックし, [設定] を選択します。 [リモート エージェントの設定] ダイアログ・ボックスが開きます。

リモート エージェントの設定		
טי		
ログの保管フォルダ:		
クリーンアップ		
テストツールの再起動: 0 📑 回実行後		
テストの処理方法		
🔲 テスト実行前に、開かれている編集中のテストを保存する		
 テスト実行前に、開かれている新規テストを保存する (<quicktest center内)<="" li="" インストールフォルダン¥tests¥quality=""> テスト実行後、新規テストを開く </quicktest>		
☑ QuickTestを非表示モードで実行(D)		
タイムアウト回復		
テスト ツールを再起動するタイミング:		
2700 🕂 秒後 (対象: 操作)		
2700 🕂 秒後 対象: クエリ)		
OK キャンセル ヘルブ田)		

- 3 ダイアログ・ボックスで設定の表示や修正ができます。詳細については、1363 ページ「[リモートエージェントの設定] ダイアログ・ボックスについて」を 参照してください。
- 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。
- 5 **リモート・エージェント・**アイコンを右クリックして [終了] を選択し, リ モート・エージェント・セッションを終了します。

[リモート エージェントの設定] ダイアログ・ボックスについて

[リモートエージェントの設定] ダイアログ・ボックスでは, Quality Center が コンピュータで QuickTest テストまたはビジネス・プロセス・テストを実行す るときに QuickTest が使用する設定の表示や変更ができます。

[リモートエージェントの設定] ダイアログ・ボックスには,次のオプション が含まれています。

オプション	説明
[レベル]	Quality Center が QuickTest テストまたはビジネス・プロセス・テストを実行するときに作成されるログに含まれる内容のレベルです。
	[なし] (標準設定): ログは作成されません。
	[低]: ログには Quality Center と QuickTest の間の通信 エラーがすべて表示されます。
	[中]: ログには Quality Center と QuickTest の間の通信 エラーと, Quality Center と QuickTest の間の通信に関 するほかの重要な操作の情報が含まれます。
	[高]: ログには Quality Center と QuickTest の通信に関 する,すべての利用可能な情報が含まれています。
[ログの保管フォルダ]	ログ・ファイルが格納されているフォルダのパスで す。[レベル]オプションでログの種類を指定した場 合に必要です。

オプション	説明	
[テスト ツールの再起動 X 回実行後]	QuickTest テストの場合, Quality Center がテスト実行 を指定されている回数完了させたら, QuickTest を再 起動します。QuickTest は再起動すると, テスト・ セットの中の次のテストを実行します。	
	ビジネス・プロセス・テストの場合, Quality Center が コンポーネントの反復を指定されている回数完了させ たら, QuickTest を再起動します。ただし, ビジネス・ プロセス・テスト実行の最中に反復の指定されている 回数に達した場合, 再起動は現在のビジネス・プロセ ス・テストの反復が終了してから行われます。	
	このオプションを使用して利用可能なメモリを最大限 にできます。	
	テスト・セット中に QuickTest を再起動しない場合は, 0(標準設定)を入力してください。	
[テスト実行前に、開かれ ている編集中のテストを保 存する]	リモート・エージェントによってテストの実行が開始 されるときに、QuickTest で既存の(命名済みの)テス トまたはビジネス・コンポーネントが開かれている場 合、このオプションにより、開いているテストまたは ビジネス・コンポーネントに加えられたすべての未保 存の変更を保存するよう、QuickTest に指示されます。 注:リモート・エージェントによってテストの実行が 開始されるときに、QuickTest で既存の(命名済みの) 関数ライブラリが開かれている場合、関数ライブラリ は保存されません。	
[テスト実行前に、開かれ ている新規テストを保存す る]	リモート・エージェントによってテストの実行が開始 されるときに、QuickTest で新規の(命名されていな い)テストが開かれている場合,テストにはシーケン シャルのテスト名がつけられ、< QuickTest のイン ストール・フォルダ> ¥Tests¥Quality Center に保存 されます。 注:リモート・エージェントによってテストの実行が 開始されるときに、QuickTest で新規の(命名されて いない)ビジネス・コンポーネントまたは関数ライブ ラリが開かれている場合、関数ライブラリは保存され ません。	
オプション	説明	
------------------------------	--	--
[テスト実行後、新規テス トを開く]	標準設定では、すべてのテストの実行が終わったと き、リモート・エージェントによって最後に実行され たテストは QuickTest で開いたままです。ただし、開 いているテストに共有リソース(共有オブジェクト・ リポジトリやデータ・テーブル・ファイルなど)が関 連付けられている場合、これらのリソースは、テスト が閉じられるまでほかのユーザにロックされていま す。このオプションを選択すれば、Quality Center が実 行する最後のテストを閉じ、代わりに空のテストを開 くことができます。	
[QuickTest を非表示モー ドで実行する]	非表示(サイレント)モードで QuickTest を実行する かどうかを指定します。標準では、このオプションは 選択されています。	
[テスト ツールを再起動す るタイミング]	次に示す項目に指定されている秒数が経過しても応答 がない場合,QuickTestを再起動します。	
	操作 : 開く,実行などの QuickTest 操作です。	
	クエリ :リモートのアプリケーションが実行してアプ リケーションの応答を確認する,標準のステータス・ クエリです (Quality Center の get_status クエリなど)。	
	両方のオプションの標準設定の値は 2700 秒(45分) です。ただし, QuickTest 操作の応答時間が長くなる ことはあっても、クエリの応答時間は通常数秒しかか かりません。したがって、それぞれのオプションに異 なる値を設定するとよいでしょう。	
	注 : QuickTest で変更を保存していない関数ライブラ リが開かれている場合, QuickTest により保存するよ う促すメッセージが表示されます。関数ライブラリが 10 秒以内に保存されなければ, QuickTest は再起動し, 保存していない変更は失われます。	

第47章・Quality Center を使用した作業



Business Process Testing を使用した作業

Business Process Testing のサポートが組み込まれた Quality Center プロジェクト に接続すると, QuickTest で Quality Center ビジネス・プロセス・テストで使用 されるコンポーネントのステップの作成や実装を行えます。

本章では、次の項目について説明します。

- ➤ Business Process Testing での作業について (1367 ページ)
- ▶ Business Process Testing での役割について (1368 ページ)
- ▶ Business Process Testing のテスト方法について(1372ページ)

Business Process Testing での作業について

Business Process Testing では、各分野のエキスパートがテストの実施および自動テ スト環境の向上を目的として、キーワード駆動方式のテストを作成できます。

Business Process Testing は QuickTest を Quality Center と組み合わせて使用し,専用の Business Process Testing ライセンスを購入することによって有効にできます。QuickTest 内から Business Process Testing を使って作業するためには, Business Process Testing のサポートが組み込まれた Quality Center プロジェクトに接続する必要があります。

本項では Business Process Testing モデルの概要を説明します。詳細については, 『HP Business Process Testing User's Guide』(英語版) および『HP QuickTest Professional for Business Process Testing ユーザーズ・ガイド』を参照してくだ さい。

Business Process Testing での役割について

Business Process Testing モデルはロールベースのモデルで,(Quality Center で作 業する)技術者でない各分野のエキスパートが,(QuickTest Professional で作業 する)オートメーション・エンジニアと効率的に協力できるようにします。各 分野のエキスパートは,ビジネス・プロセス,ビジネス・コンポーネント,お よびビジネス・プロセス・テストを定義し文書化します。一方,自動化エンジ ニアは,共有オブジェクト・リポジトリ,関数ライブラリ,および回復シナリ オなどの必要なリソースや設定を定義します。両者が連携することで,各分野 のエキスパートにプログラミングの知識がなくても,ビジネス・プロセス・テ ストを作成し,データ駆動し,文書化し,実行できます。

注:役割の構造と、組織における役割ごとに実行されるタスクは、組織で採用 されている方法論によって、ここに記述されているものとは異なる場合があり ます。これらの役割は柔軟性があり、Business Process Testing を使用する人の能 力と時間に応じても役割は異なります。たとえば、各分野のエキスパートのタ スクおよびオートメーション・エンジニアのタスクを同じ人が実行すうる場合 もあります。特定の組織でどのようなロール(役割)を定義する必要がある か、またどの種類のユーザがBusiness Process Testing のどの作業(タスク)を 実行できるか(ユーザが正しい権限を持っていることが前提となります)につ いて、製品固有の規則や制限はありません。

Business Process Testing モデルには、次のユーザの役割の分類があります。

各分野のエキスパート: 各分野のエキスパートは, アプリケーション・ロジッ クに関する知識, システム全体の深い理解, テスト対象アプリケーションの基 本となる個々の要素とタスクに関する深い理解を有しています。これにより, 各分野のエキスパートはテストが必要な運用シナリオやビジネス・プロセスを 決定したり, 複数のビジネス・プロセスに共通の重要なビジネス・アクティビ ティを特定したりできます。

Quality Center でビジネス・コンポーネント・モジュールを使用して,各分野の エキスパートは,アプリケーションを対象に実行される個別のタスクと,こう したタスクの前後のアプリケーションの条件または状態を記述するビジネス・ コンポーネントを作成します。そして,ビジネス・プロセスを構成する各ビジ ネス・コンポーネントの個々のステップを,手動ステップまたは非自動ステッ プの形式で定義します。 デザイン段階で、各分野のエキスパートは、オートメーション・エンジニアと 作業を行い、コンポーネントの自動化に必要なリソースおよび設定を特定し、 オートメーション・エンジニアがそれらを準備できるようにします。リソース と設定が準備できたら、各分野のエキスパートがそれらをキーワード駆動型の コンポーネントに変換して、手動のステップを自動化します。このプロセスで は、各コンポーネントのアプリケーション領域を選択する必要があります。ア プリケーション領域には、必要なすべてのリソース・ファイルおよび設定が含 まれます。これらは、テスト対象アプリケーションの特定の領域に固有のリ ソース・ファイルおよび設定です。各コンポーネントをアプリケーション領域 に関連付けることで、コンポーネントはこれらのリソースおよび設定にアクセ スできるようになります。

各分野のエキスパートは Quality Center のテスト計画モジュールを使用してビジ ネス・コンポーネントを組み合わせて,連続するコンポーネントのフローで構 成されるビジネス・プロセス・テストを作成します。たとえば,ほとんどのア プリケーションでは,ユーザはアプリケーションの何らかの機能にアクセスす る前にログインする必要があります。各分野のエキスパートは,ログインの手 順を行う1つのビジネス・コンポーネントを作成できます。このコンポーネン ト・プロシージャは,さまざまなビジネス・プロセス・テストで使用できま す。その結果,保守,更新,テスト管理がより容易かつコスト効率の高いもの になります。

各分野のエキスパートは,ビジネス・プロセス・テストに使用される値を設定 し,それらをテスト・セットで実行して,結果を検討します。また,各ビジネ ス・コンポーネントのテスト・ステップの保守も担当します。

コンポーネントを定義する間,各分野のエキスパートはオートメーション・エ ンジニアと協力を続けます。たとえば,コンポーネントに新しい操作(機能) を要求したり,コンポーネントに対して計画されている今後の変更を検討した りすることが考えられます。

オートメーション・エンジニア:オートメーション・エンジニアは,QuickTest Professional のような自動テスト・ツールの使用に関するエキスパートです。 オートメーション・エンジニアは各分野のエキスパートと連携して,さまざま なビジネス・プロセス・テストで必要なリソースを特定します。 その後、オートメーション・エンジニアは、個別のコンポーネントに関連付け られている機能をテストするために必要なリソースおよび設定を準備し、それ らを Quality Center プロジェクト内のアプリケーション領域に格納します。 Quality Center プロジェクトは、特定のアプリケーションを対象にしたビジネ ス・プロセス・テストの作成および実行を担当する各分野のエキスパートが使 用するプロジェクトと同じものです。

各アプリケーション領域は、コンポーネントに必要なすべてのリソースおよび 設定を保存する1つのエンティティとしての役割を果たします。これは、アプ リケーションの特定部分のテストに関連付けられているすべての要素の保守を 集中して行う場所となります。アプリケーション領域には一般に、1つ以上の 共有オブジェクト・リポジトリ、コンポーネントに対して使用できるキーワー ドのリスト、自動化関数(操作)を含む関数ライブラリ、失敗したステップの ための回復シナリオ、そのほかコンポーネントが正しく動作するために必要な 他のリソースおよび設定が含まれています。コンポーネントは、アプリケー ション領域内のリソースおよび設定に結び付いています。したがって、アプリ ケーション領域に変更が加えられると、関連するすべてのコンポーネントが自 動的に更新されます。

オートメーション・エンジニアは QuickTest の仕組みおよび機能を使用して, QuickTest からこれらのリソースを作成します。たとえば, QuickTest では, オートメーション・エンジニアは, アプリケーションの開発が完了する前で も,各種の共有オブジェクト・リポジトリを作成し,テスト対象アプリケー ション内のさまざまなオブジェクトを表すテスト・オブジェクトを含めること ができます。その後オートメーション・エンジニアは,必要に応じてリポジト リ・パラメータなどを追加できます。オートメーション・エンジニアはオブ ジェクト・リポジトリ・マネージャを使用してさまざまなオブジェクト・リポ ジトリを管理したり,Object Repository Merge Tool を使用してリポジトリを結合 したりできます。また,オートメーション・エンジニアは,QuickTest を使用し て,特定のタスクの実行に必要なステップをカプセル化するためにプログラミ ング・ロジックを使用する関数を含む関数ライブラリの作成とデバッグを行う ことができます。

オートメーション・エンジニアによって作成されたリソースを使用して,各分 野のエキスパートはコンポーネント・ステップを自動化し,コンポーネントお よびビジネス・プロセス・テストの作成と保守ができます。

自動化エンジニアは、必要に応じて QuickTest の中でコンポーネントを作成、 デバッグ、および変更することもできます。

Business Process Testing のワークフローについて

Business Process Testing のワークフローはテストのニーズに応じて異なります。 次は一般的なワークフローの例です。



Business Process Testing のテスト方法について

各分野のエキスパートが作成する各シナリオは,ビジネス・プロセス・テスト です。1つのビジネス・プロセス・テストは,連続するコンポーネントのフ ローで構成されています。各コンポーネントによって特定のタスクが実行され ます。また,コンポーネントから後続のコンポーネントにデータが引き渡され ます。

コンポーネントについて

コンポーネントは、保守が簡単で、特定のタスクを実行する再利用可能なスク リプトであり、効果的なビジネス・プロセス・テスト構造を作成できる基礎的 な要素です。コンポーネントは、細分化されたビジネス・プロセスの一部で す。たとえば、ほとんどのアプリケーションのユーザは、何かを行う前にログ インする必要があります。各分野のエキスパートは、アプリケーションへログ インする手続きを行う1つのコンポーネントを作成できます。各コンポーネン トは別のビジネス・プロセスのテストで再利用できます。その結果、保守、更 新、テスト管理がより容易になります。

コンポーネントはステップで構成されています。たとえば,ログイン・コン ポーネントの最初のステップではアプリケーションを開きます。第2ステップ はユーザ名の入力です。第3ステップはパスワードの入力,そして第4ステッ プは [**ログイン**] ボタンのクリックです。

また、コンポーネントには、チェックポイント・ステップや出力値も追加できます。

- ▶「チェックポイント」は、指定したプロパティの現在値とそのプロパティの期待値を比較する確認ポイントです。チェックポイントにより、アプリケーションが正しく機能しているかどうかを判断できます。コンポーネント・ステップに対して、標準チェックポイントおよびビットマップ・チェックポイントを実行できます。詳細については、507ページ「チェックポイントについて」を参照してください。
- ▶「出力値」とは、実行セッション中にコンポーネントの特定のポイントで1つ 以上の値がキャプチャされ、格納されるステップです。その値は、後で実行 セッションの別のポイントで入力値として使用できます。詳細については、 685ページ「値の出力」を参照してください。

サポートされている任意の環境を対象としたステップの追加,特定の項目のパ ラメータ化,および特定のタスクの実行に必要なステップをカプセル化する関 数(操作)の組み込みによるコンポーネントの拡張など,QuickTestの中でコン ポーネントの作成および編集ができます。Quality Centerでは,各分野のエキス パートはコンポーネントを作成し,それらを組み合わせて,アプリケーション が想定どおりに動作することをチェックするために使用するビジネス・プロセ ス・テストを組み立てます。

Quality Center ビジネス・コンポーネント・モジュールでのコンポーネ ントの作成

各分野のエキスパートは、新規のコンポーネントを作成し、Quality Center のビジネス・コンポーネント・モジュールの中でそれを定義できます。



ビジネス・コンポーネント・モジュールには、次が含まれます。

- ▶ [詳細] タブ: コンポーネントの目的またはゴールのサマリ,およびコンポーネントの実行前後のアプリケーションの状態(前提条件および後提条件)を指定します。
- ► [スナップショット] タブ: コンポーネントの目的や操作を視覚的に示すイメージを表示します。

- ▶ 【パラメータ】タブ:ビジネス・コンポーネントの入力コンポーネント・パラメータ値および出力コンポーネント・パラメータ値を指定します。パラメータを実装および使用することで、コンポーネントは、外部ソースからデータを受け取り、ビジネス・プロセス・テスト・フローのほかのコンポーネントにデータを渡すことができるようになります。
- ► [ステップのデザイン] タブ: ビジネス・コンポーネントの手動ステップを作成または表示し、必要に応じて手動ステップを自動化できます。
- ▶ 【オートメーション】 タブ:自動コンポーネントの表示またはアクセスができます。キーワード方式のコンポーネントの場合、キーワード方式のテーブル形式で自動ビジネス・コンポーネントのステップを作成、変更でき、実装されているコンポーネントの各ステップについて分かりやすく書かれた説明が表示されます。
- ▶ [使用対象] タブ: 現在選択されているビジネス・コンポーネントが含まれる ビジネス・プロセス・テストの詳細が示されます。また、テスト計画モジュー ルの関連ビジネス・プロセス・テストへのリンクも示されます。

QuickTest Professional でのコンポーネントの実装

一般に、コンポーネントは各分野のエキスパートが Quality Center の中で作成しますが、QuickTest の中でも作成およびデバッグができます。

QuickTest 内では、ステップを手動で追加することでコンポーネントを作成しま す(オブジェクト・リポジトリにデータが設定されており、必要な操作が利用 可能な場合)。また、サポートされているすべての環境のステップを記録する ことによって、コンポーネントを作成することもできます。特定の項目のパラ メータ化が可能です。また、コンポーネント固有のオプションの表示や設定も 行えます。

QuickTest では、ビジネス・コンポーネントおよびスクリプト・コンポーネント の2種類のコンポーネントの作成および変更が可能です。ビジネス・コンポー ネントは、特定のタスクを実行する1つ以上のステップで構成されている、保 守が容易で再利用可能な単位です。スクリプト・コンポーネントは、プログラ ム・ロジックを含むことのできる自動化されたコンポーネントです。スクリプ ト・コンポーネントは、テスト・アクションおよびビジネス・コンポーネント の両方の機能を併せ持ちます。 たとえば、キーワード・ビュー、エキスパート・ビュー、その他の QuickTest ツールやオプションを使用して、QuickTest の中でスクリプト・コンポーネント の作成、表示、変更、デバッグが行えます。スクリプト・コンポーネントは複 雑なため、QuickTest 内でのみ編集可能です。(必要に応じて、テスト・アク ションをスクリプト・コンポーネントに変換できます。詳細については、

『HP QuickTest Professional for Business Process Testing ユーザーズ・ガイド』の スクリプト・コンポーネントを説明している項,または[アクション変換ツー ル]ウィンドウの [**ヘルプ**] ボタンをクリックしてください)。

Quality Center では,各分野のエキスパートは QuickTest 内で作成されたコン ポーネントを開けます。その後各分野のエキスパートはビジネス・コンポーネ ントを表示および編集できますが,スクリプト・コンポーネントについては詳 細のみを表示できます。

Quality Center テスト計画モジュールでのビジネス・プロセスのテスト の作成

ビジネス・プロセス・テストを作成するには,各分野のエキスパートがビジネ ス・プロセス・テストに適したコンポーネントを選択(ドラッグ・アンド・ド ロップ)し,それらの実行設定を設定します。

注: Quality Center からビジネス・プロセス・テストを実行する場合,テスト実行は QuickTest リモート・エージェントでの設定の影響も受ける可能性があります。QuickTest リモート・エージェントの詳細については,1361ページ「QuickTest リモート・エージェントの設定」を参照してください。

各コンポーネントは,異なるビジネス・プロセス・テストで別々に使用できま す。たとえば,コンポーネントに対してテストごとに異なる入力パラメータ値 を使用したり,異なる反復回数で実行されるよう設定したりできます。

ビジネス・プロセス・テストを作成しているときに、各分野のエキスパート が、ビジネス・プロセス・テストに必要な要素に対してコンポーネントがまだ 定義されていないことに気付いたとします。その場合は、テスト計画モジュー ルからコンポーネント要求を送信できます。

ビジネス・プロセス・テストの実行と結果の分析

QuickTest で [実行] オプションや [デバッグ] オプションを使用して, 個々の コンポーネントの実行やデバッグができます。

Quality Center のテスト計画モジュールからテストを実行して,ビジネス・プロ セスのテストのデバッグが行えます。このモジュールから実行する場合,デ バッグ・モードで実行するコンポーネントを選択できます。(この処理は,コ ンポーネントの開始時に実行を一時停止します。

ビジネス・プロセス・テストがデバッグされ,通常のテスト実行の準備が整ったら,各分野のエキスパートは,他のテストを Quality Center で実行する場合と 同様の方法で,テストのラボ・モジュールからテストを実行します。テストを 実行する前に,各分野のエキスパートは,テストのラボ・モジュールのグリッ ドの[**反復**]カラムを使用して,実行時のパラメータ値と反復を定義できます。

テストのラボ・モジュールから、ビジネス・プロセス・テスト全体の実行結果 を確認できます。結果には、各パラメータの値や、QuickTestによって報告され た個々のステップの結果が含まれています。

[**レポートの表示**]のリンクをクリックして,QuickTest レポート全体を開くことができます。この階層化されたレポートには,ビジネス・プロセス・テスト実行での,それぞれの反復やコンポーネントがすべて含まれています。

コンポーネントとテストの違いについて

すでに QuickTest を使用したアクション・ベースのテストの作成に慣れている 場合は、その手順とコンポーネントの作成や編集の手順が非常に似ていること に気付くでしょう。ただし、コンポーネント・モデルの設計や目的により、コ ンポーネントの作成、編集、実行の点で多少の違いがあります。次のガイドラ インでは、これらの違いの概要を説明します。

- ➤ コンポーネントは単独のエンティティです。複数のアクションを含めたり、別のアクションや別のコンポーネントを呼び出すことはできません。
- ➤ コンポーネントを使って作業をすると、外部ファイルはすべて現在接続している Quality Center プロジェクトに格納されます。
- ▶ キーワード・ビューでのコンポーネントのノードの名前は、保存されているコンポーネントと同じです。ノード名は変更できません。
- ▶ ビジネス・コンポーネントは、エキスパート・ビューではなくキーワード・ ビューで作成します。

- ▶ コンポーネントに直接ではなく、コンポーネントのアプリケーション領域を介してリソースを追加します。
- ➤ コンポーネントは、関数ライブラリで作成されたユーザ定義のキーワードを使用して、プロパティ値の確認およびテスト対象アプリケーションの起動などの操作を実行します。

第48章 • Business Process Testing を使用した作業



WinRunner を使用した作業

QuickTest では、WinRunner のテストを実行したり、コンパイル済みモジュール 内の TSL 関数またはユーザ定義関数を呼び出したりすることもできます。

本章では、次の項目について説明します。

- ▶ WinRunner を使用した作業について(1379ページ)
- ▶ WinRunner テストの呼び出し(1380 ページ)
- ➤ WinRunner 関数の呼び出し(1384 ページ)

WinRunner を使用した作業について

コンピュータに WinRunner 8.2 以上がインストールされている場合には, QuickTest のテストに WinRunner のテストまたは関数への呼び出しを含めること ができます。

WinRunner テストまたは関数への呼び出しを作成したら、呼び出しステートメントをエキスパート・ビューまたはキーワード・ビューで編集して、そのステートメントの引数値を変更できます。

QuickTest を, WinRunner のテストまたはコンパイル済みモジュールが含まれて いる Quality Center プロジェクトに接続すると, Quality Center プロジェクトに 格納されている WinRunner テストまたは関数を呼び出すことができます。

WinRunner テストの呼び出し

テストを実行するために、QuickTestからWinRunnerに接続すると、WinRunner が起動されテストが開かれ、実行されます。WinRunnerテスト実行に関する情報は、QuickTestの[テスト結果]ウィンドウに表示されます。

WinRunner テストへの呼び出しを挿入するには, [WinRunner テストの呼び出し] ダイアログ・ボックスを使用するか, エキスパート・ビューの中で TSLTest.RunTestEx ステートメントを入力します。

注: QuickTest テストへの呼び出しを含む WinRunner テストを呼び出すことはできません。

[WinRunner テストの呼び出し] ダイアログ・ボックスを使用して WinRunner テストへの呼び出しを挿入するには, 次の手順を実行します。

1 [**挿入**] > [WinRunner テストの呼び出し] を選択します。[WinRunner テスト の呼び出し] ダイアログ・ボックスが表示されます。

WinRunner テストの呼び出し	×
テストのパス[[]:	
テストのパラメータ	
名前	
名前(1): (值(1): (值(1): (1): (1): (1): (1): (1): (1): (1):	
✓ WinRunnerを最小化の状態で実行する(M)	
🔲 テスト実行後、WinRunnerを終了する(<u>C</u>)	
OK キャンセル ハ	มว่

2 [テストのパス] ボックスに WinRunner テストのパスを入力するか,参照ボタ ンを使って WinRunner テストを指定します。

Quality Center に接続していると、参照ボタンをクリックしたときに、Quality Center プロジェクトのモジュールを選択するための、[Quality Center プロジェ クトから WinRunner テストを開く]ダイアログ・ボックスが開きます。このダ イアログ・ボックスの詳細については 1334 ページ「Quality Center プロジェクトからテストを開く」を参照してください。

- 3 [テストのパラメータ] ボックスには、WinRunner テストに必要なすべてのテスト・パラメータのリストが表示されます。パラメータの値を入力するには、次の手順を実行します。
 - ▶ [テストのパラメータ] リスト内のパラメータを選択して強調表示にします。 選択した値がリストの下の [名前] ボックスに表示されます。
 - ▶ [**値**] ボックスに新しい値を入力します。

注: QuickTest ランダム・パラメータまたは環境パラメータのパラメータ値や, QuickTest データ・テーブルのパラメータ値を,WinRunner テストのパラメータ として使用することもできます。それにはパラメータの情報を TSLTest.RunTestEx ステートメントに手作業で入力しても行えます。詳細につ いては,1382ページ「WinRunner テストへの QuickTest でパラメータ化された 値の引き渡し」を参照してください。

- 4 WinRunner ウィンドウがテストの実行中に表示されないようにするには、[WinRunner を最小化の状態で実行する]を選択します(このオプションのサポート対象は、WinRunner バージョン 7.6以降です)。
- 5 WinRunner テストを呼び出したステップが完了したときに WinRunner アプリ ケーションを終了させるには、[テスト終了後、WinRunner を実行する]を選 択します(このオプションのサポート対象は, WinRunner バージョン 8.2 以降 です)。
- 6 [OK] をクリックして、ダイアログ・ボックスを閉じます。

WinRunner のテスト・パラメータの詳細については,『**HP WinRunner ユーザー** ズ・ガイド』を参照してください。 QuickTest では, WinRunner テストへの呼び出しが次のように表示されます。

▶ キーワード・ビューには、WinRunnerのRunTestEx ステップが表示されます。 次に例を示します。

and the contraction of the	操作	値
🔤 💏 TSLTest	RunTestEx	"C:¥WinRunner¥Tests¥basic flight",True,0,MyValue

➤ エキスパート・ビューには、VBScript で書かれた TSLTest.RunTestEx ステート メントが表示されます。次に例を示します。

TSLTest.RunTestEx "C:\U00e4WinRunner\U00e4Tests\U00e4basic_flight",TRUE, 0, "MyValue"

RunTestEx メソッドの構文は、次のとおりです。

TSLTest.RunTestEx テストのパス, 最小化実行, アプリケーション終了 [, パラ メータ]

注: QuickTest 6.0 で作成したテストは, RunTest メソッドを使用して WinRunner テストを呼び出している場合があります。このメソッドは,構文が 少し異なります。このメソッドでもテストは正常に実行されます。ただし, WinRunner 7.6 以降を使用している場合には,RunTestEx メソッド(および対 応する引数の構文)を使用するようにテストを更新することをお勧めします。 これらのメソッドの詳細については,『HP QuickTest Professional Object Model Reference』(英語版)を参照してください。

テストの実行後,結果を表示できます。詳細については,1383ページ「結果の 表示」を参照してください。

RunTestEx メソッドの詳細とその使用例については,『**HP QuickTest Professional Object Model Reference**』(英語版)を参照してください。

WinRunner テストへの QuickTest でパラメータ化された値の引き渡し

WinRunner テストに要求されるパラメータに固定の値を設定する代わりに, QuickTest データ・テーブル, ランダム・パラメータまたは環境パラメータに定 義されているパラメータ値を WinRunner に引き渡せます。これらのパラメータ 値を指定するには, TSLTest.RunTestEx ステートメントのパラメータ引数とし て, 適切なステートメントを入力します。 たとえば、Windows ベースのフライト予約アプリケーションを対象に WinRunner テストを実行するとき、そのテストにフライトの乗客数と座席のク ラスをパラメータ化するステートメントが組み込まれていたとします。その場 合、最初のパラメータの値を QuickTest ランダム・パラメータ (1から 100まで の数をランダムに生成)から、また座席のクラスの値を QuickTest データ・ テーブルの Class というカラムから WinRunner テストに渡します。その場合、 QuickTest の TSLTest.RunTestEx ステートメントは、たとえば次のようになり ます。

TSLTest.RunTestEx "D:¥test1", TRUE, FALSE, RandomNumber(1, 100), DataTable("Class", dtGlobalSheet)

RandomNumber メソッド, Environment メソッド,および DataTable メソッド の構文および使用方法の詳細については,『HP QuickTest Professional Object Model Reference』(英語版)の「Utility」のセクションを参照してください。

結果の表示

WinRunner テストへの呼び出しを実行するとき,コンピュータに WinRunner 8.2 以降がインストールされていれば,通常なら WinRunner の結果に含まれるイベ ントごとに,QuickTest の結果にノードが含まれます。WinRunner ステップに対 応するノードを選択すると,右の表示枠にWinRunner テストのサマリと,選択 したステップの詳細が表示されます。

注: 呼び出された WinRunner テストの結果は, WinRunner テストの results フォ ルダから見ることもできます。Quality Center に格納されている WinRunner テス トの場合, Quality Center から WinRunner テストの結果を見ることができます。

詳細については, 992ページ「テスト結果での WinRunner テスト・ステップの 表示」を参照してください。

WinRunner テストの設計と実行の詳細については、WinRunner のマニュアルを 参照してください。

WinRunner 関数の呼び出し

関数を呼び出すために、QuickTestからWinRunnerに接続すると、WinRunnerが 起動され、コンパイル済みモジュールが読み込まれてから、関数が呼び出され ます。これは、QuickTestでWinRunnerのユーザ定義関数を使用する場合に便利 です。

QuickTest から WinRunner 関数を呼び出すには、その関数と、その関数が含まれているコンパイル済みモジュールを指定します。

注: WinRunner 関数から返される値を QuickTest テストの中で取得することはできません。ただし、返された値は、結果の中で見ることができます。

WinRunner のコンパイル済みモジュールのユーザ定義関数を呼び出すには、次の手順を実行します。

1 [**挿入**] > [**WinRunner の呼び出し**] > [**関数**] を選択します。[WinRunner 関数の呼び出し] ダイアログ・ボックスが表示されます。

WinRunner 関数の呼び出	.U		×
モジュール(M): 関数名(E):	_		(.)
引数 ————— 名前	値	৯17	
名前(1):	値[⊻]:		V
 ✓ WinRunnerを最小化 「テスト実行後、WinR 	Cの状態で実行する(E) unnerを終了する(E)		
	OK 3	キャンセル ヘル	"

2 [モジュール] ボックスに, その関数を含んでいるコンパイル済みモジュール のパスを入力するか,参照ボタンを使って指定します。

Quality Center に接続していると、参照ボタンをクリックしたときに、Quality Center プロジェクトのコンパイル済みモジュールを選択するための、[Quality Center プロジェクトから WinRunner テストを開く] ダイアログ・ボックスが開きます。

WinRunner の TSL 関数を呼び出すには、任意のコンパイル済みモジュールのパスを入力します。

- 3 [関数名] ボックスに,指定したコンパイル済みモジュールに定義されている 関数の名前を入力するか,任意の WinRunner TSL 関数を入力します。
- 4 [引数] ボックスの中をクリックします。コンピュータ上で WinRunner が開いている場合,[引数] ボックスには,選択した関数に対して定義されている引数の名前が表示されます。WinRunner が開いていなければ,[引数] ボックスには関数に対する最大 15 個までの引数を表す p1 から p15 までが表示されます。
- 5 次のように, in または inout 引数の値を入力します。
 - ▶ [引数] ボックスで引数を選択して強調表示にします。引数の名前が [名前] ボックスに表示されます。
 - ➤「in」または「inout」の引数タイプを選択した場合、[值] ボックスに新しい 値を入力します。
 - ▶ [**タイプ**] ボックスで,正しい引数のタイプ(in/out/inout)を選択します。

注: QuickTest ランダム・パラメータまたは環境パラメータのパラメータ値や, QuickTest データ・テーブルのパラメータ値を,関数の in 引数または inout 引 数として使用することもできます。それには引数の情報を TSLTest.CallFuncEx ステートメントに手作業で入力します。詳細については,1387ページ 「WinRunner 関数への QuickTest パラメータの引き渡し」を参照してください。

関数パラメータの詳細については,『HP WinRunner ユーザーズ・ガイド』を参照してください

6 関数が実行されている間、WinRunner ウィンドウが表示されないようにするには、[WinRunner を最小化の状態で実行する]を選択します(このオプションのサポート対象は、WinRunner バージョン 8.2 以降です)。

- 7 WinRunner 関数を呼び出したステップが完了したときに WinRunner アプリケーションを終了させるには、[テスト実行後、WinRunner を終了する] を選択します(このオプションのサポート対象は、WinRunner バージョン 7.6 以降です)。
- 8 [OK] をクリックして、ダイアログ・ボックスを閉じます。

QuickTest では,TSL 関数への呼び出しが次のように表示されます。

▶ キーワード・ビューには、WinRunnerの CallFuncEx ステップが表示されます。 例を次に示します。

操作 値 ■ 予告 CallFuncEx "C¥WinRunner¥Tests¥TIStep","TIStep1",TRUE,0,"MyArg1"

➤ エキスパート・ビューには、VBScript で書かれた TSLTest.CallFuncEx ステート メントが表示されます。次に例を示します。

CallFuncEx "C:¥WinRunner¥Tests¥TIStep","TIStep1",TRUE, 0, "MyArg1"

CallFuncEx メソッドの構文は、次のとおりです。

TSLTest.CallFuncEx モジュールのパス, 関数, 最小化実行, アプリケーション 終了[, 引数]

注: QuickTest 6.0 で作成したテストは, CallFunc メソッドを使用して WinRunner テストを呼び出している場合があります。このメソッドは,構文が 少し異なります。このメソッドでもテストは正常に実行されます。ただし, WinRunner 7.6 以降を使用している場合には, CallFuncEx メソッド(および対 応する引数の構文)を使用するようにテストを更新することをお勧めします。 これらのメソッドの詳細については,『HP QuickTest Professional Object Model Reference』(英語版)を参照してください。

テストの実行後,結果を表示できます。詳細については,1387ページ「結果の 表示」を参照してください。

WinRunner 関数,関数引数,および WinRunner のコンパイル済みモジュールの 詳細については,『HP WinRunner ユーザーズ・ガイド』および『HP WinRunner TSL リファレンス・ガイド』を参照してください。

WinRunner 関数への QuickTest パラメータの引き渡し

WinRunner 関数の in 引数や inout 引数に固定した値を設定する代わりに, QuickTest ランダム・パラメータや環境パラメータで定義されたパラメータ値, または QuickTest データ・テーブルで定義されたパラメータ値を WinRunner に 引き渡すように, QuickTest に指示できます。これらのパラメータを指定するに は, TSLTest.CallFuncEx ステートメントの パラメータ引数として,適切なス テートメントを入力します。

たとえば WinRunner を使って、アプリケーションを起動し、そのアプリケー ションにユーザ名とパスワードを入力するユーザ定義関数を作成するとします。

そのためには、ユーザ名とパスワードの値を QuickTest データ・テーブルの FlightUserName 列と FlightPwd 列から WinRunner に引き渡すよう、QuickTest に指示します。その場合、QuickTest の TSLTest.CallFuncEx ステートメントは、 たとえば次のようになります。

TSLTest.CallFuncEx "D:¥flightfuncs", "run_flight", TRUE, FALSE, DataTable("FlightUserName", dtGlobalSheet), DataTable("FlightPwd", dtGlobalSheet)

RandomNumber メソッド, Environment メソッド,および DataTable メソッド の構文および使用方法の詳細については,『HP QuickTest Professional Object Model Reference』(英語版)の「Utility」のセクションを参照してください。

結果の表示

QuickTest から WinRunner 7.6 以降を使用して WinRunner 関数を実行した後,関数呼び出しの結果を表示することができます。QuickTest の[テスト結果] ウィンドウに,WinRunner 関数の開始と WinRunner 関数の結果が表示されます。呼び出した関数に report_msg または tl_step などのイベントが含まれている場合, それらのイベントの結果に関する情報も含まれます。 結果ツリーの [WinRunner Function Results] を選択して強調表示にすると, 関数の戻り値と関数への呼び出しに関する追加情報が表示されます。

🚰 Test1 [Res4] - Test Results	
Eile <u>V</u> iew <u>T</u> ools <u>H</u> elp	
े 🖻 🍠 📅 🌏 🛱 🔍 🔍	← → ?
est1 Summary n-Time Data st1 Iteration 1 (Row 1) Action1 Summary	Step Name: WinRunner Function Results
sun winnumer rest start run (basic_flight) x the bitmap checkpoint (Img1:1) v the property check x stop run (basic_flight)	Object Details Result Time WinRunner Function 5/20/2004 - 9:32:01
my Start WinRunner Function (TLStep) 	Call to WinRunner Function
- MinRunner Function Results	Compiled module: CAtest\comp
	Function name: <i>TLStep</i>
For Help, press F1	Ready

WinRunner 関数およびコンパイル済みモジュールを使用した作業の詳細については,WinRunnerのマニュアルを参照してください。



HP のパフォーマンス・テストおよび Business Availability Center 製品を使用した 作業

QuickTest を使用して,アプリケーションの機能をテストする一連のテストを作成して実行できたら,今度はアプリケーションがどれくらいの負荷を処理できるかを検証したり,アプリケーションが実行する様子を監視したりできます。

HP LoadRunner は、一定の負荷または過負荷状態でのアプリケーションのパフォーマンスをテストします。負荷を生成するために、LoadRunner によって数百人の仮想ユーザが実行されます。これらの仮想ユーザは、一貫性のある再現可能かつ測定可能な負荷を提供し、現実のユーザとまったく同じようにアプリケーションを操作します。

HP Business Availability Center は, エンド・ユーザ体験をリアルタイムで監視で きます。Business Process Monitor は, 監視対象アプリケーションを対象に仮想 ユーザを実行して典型的な操作を実行します。

すでに QuickTest でテストを作成して,それがユーザのアクションをうまく表 現することがわかっている場合は,その QuickTest テストをパフォーマンス・ テストおよびアプリケーション管理の基盤として使うことができます。Silent Test Runner を使用して, QuickTest テストが LoadRunner および Business Process Monitor から正常に実行されることを事前に確認できます。

本章では、次の項目について説明します。

- ➤ HP のパフォーマンス・テストおよび Business Availability Center 製品を使用 した作業について(1390ページ)
- ➤ QuickTest のパフォーマンス・テストおよび Business Availability Center の使用(1391ページ)

- ▶ LoadRunner または Business Process Monitor で使用する QuickTest テストの設計(1392ページ)
- ▶ LoadRunner または Business Process Monitor でのテストの挿入と実行(1393 ページ)
- ▶ トランザクションの測定(1395ページ)
- ▶ サイレント・テスト・ランナーの使用(1400ページ)

HP のパフォーマンス・テストおよび Business Availability Center 製品を使用した作業について

QuickTest では、アプリケーション全般の機能を検査し、アプリケーションのすべての要素があらゆる状況で期待どおりに動作することを確認する複雑なテストを作成できます。

HP のパフォーマンス・テスト製品および HP Business Availability Center 製品で 使用されている実行メカニズムは同じです。つまり, LoadRunner と Business Process Monitor の両方に互換性のあるテストを作成することができ, QuickTest で設計され, デバッグされたテストやテストのセグメントを利用できるという ことです。

たとえば、QuickTest テストを LoadRunner シナリオの特定のポイントに追加し、 それらのポイントでの追加の負荷によってアプリケーションの機能が影響を受 けていないことを確認できます。また、Business Process Monitor を対象として QuickTest テストを実行し、エンド・ユーザ体験をシミュレートし、アプリケー ションが正しく適切なタイミングで実行されることを確認することもできます。

また QuickTest は, LoadRunner および Business Process Monitor と統合専用に設計されているいくつかの機能を提供します。しかし, LoadRunner および Business Process Monitor は,標準的なユーザ操作を同時に行う多数のユーザを 表す仮想ユーザを使用して,テストを実行するように設計されているので, QuickTest とこれらの製品を統合すると,QuickTest のいくつかの機能が利用で きない場合があります。 LoadRunner か Business Process Monitor またはその両方と, QuickTest とで, 1つ のテストを使う場合は, テストを設計するときに, 各製品がサポートするオプ ションの違いを考慮しなければなりません。詳細については, 1392ページ 「LoadRunner または Business Process Monitor で使用する QuickTest テストの設 計」および 1393 ページ「LoadRunner または Business Process Monitor でのテス トの挿入と実行」を参照してください。

QuickTest のパフォーマンス・テストおよび Business Availability Center の使用

Services オブジェクトと関連メソッドを使用して,パフォーマンス・テストお よび Business Availability Center に特に関連のあるステートメントを挿入できま す。これらには,AddWastedTime,EndDistributedTransaction, EndTransaction,GetEnvironmentAttribute,LogMessage,Rendezvous, SetTransaction,SetTransactionStatus,StartDistributedTransaction, StartTransaction,ThinkTime,および UserDataPoint があります。これらのメ ソッドの詳細については、『HP QuickTest Professional Object Model Reference』 の「Services」の項,および LoadRunner または Business Availability Center の マニュアルを参照してください。

🏟 😩

また、[挿入] > [トランザクションの開始] および [挿入] > [トランザク ションの終了] メニュー・オプションまたはツールバー・ボタン を使用して、 StartTransaction および EndTransaction ステートメントを挿入できます。これ らのオプションの詳細については、1395 ページ「トランザクションの測定」を 参照してください。

注: LoadRunner と Business Process Monitor は、トランザクションに含まれた データだけを使用し、トランザクション外のテストのデータは無視します。

LoadRunner または Business Process Monitor で使用する QuickTest テストの設計

LoadRunner または Business Process Monitor で使用する QuickTest テストは、シ ンプルに保ち、対象の操作を限定し、外部アクションの使用や外部ファイルへ の参照は避けるべきです。また、アクションの反復を操作するときは、対応関 係にある StartTransaction ステートメントおよび EndTransaction ステートメン トが同じアクション内に含まれている必要があります。

LoadRunner 向けのテストの設計

LoadRunner で使用するテストを設計するときは、次のガイドラインに従ってください。

- ▶ 外部アクションや、外部データ・テーブル・ファイル、環境変数ファイル、共 有オブジェクト・リポジトリ、関数ライブラリなどの外部リソース(Quality Center に保存されているリソースを含む)への参照を含めないようにします。 これは、LoadRunner が外部のアクションやリソースにアクセスできない場合が あるためです(ただし、リソースがネットワーク上で見つかる場合、QuickTest はそのリソースを使用します)。
- ▶ LoadRunner で有益な情報を提供するためには、すべての QuickTest テストに少なくとも1つのトランザクションを含める必要があります。
- ▶ テスト内の最後の(1つまたは複数の)ステップで、テスト対象のアプリケーションを、実行中のすべての子プロセスも含めて必ず終了するようにします。 これにより、テストの次の反復で再びアプリケーションを開くことができるようになります。

Business Process Monitor 向けのテストの設計

Business Process Monitor で使用するテストを設計するときは、次のガイドラインに従ってください。

- ➤ Business Process Monitor で有益な情報を提供するためには、すべての QuickTest テストに少なくとも1つのトランザクションを含める必要があります。
- ▶ 2つの異なる Business Process Monitor プロファイルで分散トランザクションを測定する場合,StartDistributedTransaction ステートメントを使用するプロファイルを,関連付けられた EndDistributedTransaction ステートメントを使用するプロファイロファイルより前に実行する必要があります。

- ▶ 分散トランザクションを測定する場合、テストを1つの Business Process Monitor のインスタンスに関連付けることを確認します。Business Process Monitor に よってすべてのインスタンスの終了トランザクション名が検索され、複数のイ ンスタンスに含まれている場合に不正な分散トランザクションを終了する場合 があります。
- ▶ 分散トランザクションを2つのBusiness Process Monitorのプロファイルで測定す る場合,指定するタイムアウト値の大きさが十分であり, StartDistributedTransaction ステップを含むプロファイルおよび EndDistributedTransaction ステップを含むプロファイルの前に実行するすべて のプロファイルが,指定したタイムアウト値より短い時間で実行し終えること を確認します。
- Business Process Monitor では、Quality Center に保存されているリソース(共有 オブジェクト・リポジトリ、関数ライブラリ、外部データ・テーブル、外部ア クションなど)を含む外部リソースへのアクセスを必要とする QuickTest Professional テストの実行はサポートされていません。外部リソースを必要とす るテストは、Business Process Monitor 上での実行に失敗する場合があります (ただし、リソースがネットワーク上で見つかる場合、QuickTest はそのリソー スを使用します)。
- ▶ テスト内の最後の(1つまたは複数の)ステップで、テスト対象のアプリケーションを、実行中のすべての子プロセスも含めて必ず終了するようにします。 このクリーンアップ・ステップにより、次のテスト実行で再びアプリケーションを開くことができるようになります。

LoadRunner または Business Process Monitor でのテストの挿入 と実行

LoadRunner または Business Process Monitor で QuickTest テストを挿入し実行す るときは,次のガイドラインに従ってください。

注:サイレント・テスト・ランナーを使用すれば,LoadRunnerまたはBusiness Process Monitor からのテストの実行方法をシミュレートできます。詳細について は、1400ページ「サイレント・テスト・ランナーの使用」を参照してください。

LoadRunner シナリオでのテストの挿入と実行

- ▶ 1 台のコンピュータで同時に実行できる GUI 仮想ユーザは1つまでです(GUI 仮想ユーザとは QuickTest テストを実行する仮想ユーザのことです)。
- ➤ QuickTest テストを LoadRunner シナリオに挿入するには、コントローラの [テ ストを開く] ダイアログ・ボックスでテスト・フォルダまで移動し、[ファイ ルの種類] ボックスで [QuickTest テスト] を選択します (LoadRunner 9.1 以前のバージョンの場合は [Astra テスト] を選択)。これで、フォルダ内の QuickTest テストが表示されます。
- ➤ LoadRunner から QuickTest テストを実行する前に、QuickTest コンピュータ上の QuickTest が終了していることを確認します。
- ➤ QuickTest を使用して記録したテスト(スクリプト)では、トランザクション・ ブレークダウンはサポートされていません。
- ▶ 次のコンピュータ上では QuickTest を実行できません。
 - ➤ ログオフまたはロックされたコンピュータ。これらの場合は、ターミナル・ サーバでの QuickTest の実行を検討してください。
 - ▶ すでに QuickTest テストが実行されているコンピュータ。テストの実行が完了 していることを確かめてから,別の QuickTest テストを開始してください。
- ▶ LoadRunnerの[実行環境設定]ダイアログ・ボックスの設定は、QuickTestテストには関係しません。
- ➤ LoadRunner でテストを実行しているときは、ResultDir QuickTest 環境変数を使用することはできません。

LoadRunner での作業の詳細については, LoadRunner のマニュアルを参照して ください。

Business Process Monitor でのテストの挿入と実行

- ➤ Business Process Monitor で QuickTest テストを実行する前に、使用しているバージョンの Business Process Monitor でどのバージョンの QuickTest がサポートされているのかを確認してください。詳細については、Business Process Monitorのマニュアルを参照してください。
- ➤ Business Process Monitor で一度に実行できる QuickTest テストは1つだけです。 前の QuickTest テストの実行が完了していることを確かめてから,別の QuickTest テストを開始してください。

- ➤ Business Process Monitor から QuickTest テストを実行する前に、QuickTest コン ピュータ上の QuickTest が終了していることを確認します。
- ➤ QuickTest を使用して記録したテストでは、トランザクション・ブレークダウン はサポートされていません。
- QuickTest テストを Business Availability Center にアップロードした後に、テストのローカル・コピーに変更を加えた場合は、変更を加えたテストを Business Process Monitor で実行できるようにするために、ZIP 圧縮したテストを再度 アップロードする必要があります。
- ▶ ログオフされたコンピュータ、ロックされたコンピュータ、または QuickTest を非対話型サービスとして実行しているコンピュータでは、QuickTest はテスト を実行できません。
- ➤ Business Process Monitor でテストを実行しているときは, ResultDir QuickTest 環 境変数を使用することはできません。

Business Availability Center での作業の詳細については, Business Availability Center のマニュアルを参照してください。

トランザクションの測定

トランザクションを定義することで、テストの特定セクションの実行にかかる 時間を測定できます。トランザクションは、測定対象のアプリケーション内の プロセスを表しています。テストには、LoadRunner または Business Process Monitor で使用できるトランザクションを含める必要があります。LoadRunner と Business Process Monitor は、トランザクションに含まれたデータだけを使用 し、トランザクション外のテストのデータは無視します。

トランザクション開始ステートメントとトランザクション終了ステートメント でテストの該当セクションを囲むことで、テスト内のトランザクションを定義 できます。たとえば、飛行機の座席を予約するのにかかる時間や、クライアン トの端末に確認メッセージが表示されるまで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。

テストの実行中, StartTransaction ステップは,時間測定の開始を示します。 時間測定は, EndTransaction ステップに到達するまで継続されます。テスト・ レポートに,トランザクションの実行に要した時間が表示されます。 **注**:トランザクションの開始時にすでに同じ名前のトランザクションが開いて いると、先行していたトランザクションは**失敗**ステータスで終了し、その後新 しいトランザクションが開始されます。

トランザクション内で使用するステートメントの詳細については,

『HP QuickTest Professional Object Model Reference』(英語版)を参照してくだ さい。

テストに追加できるトランザクションの数には上限がありません。トランザク ション内にトランザクションを挿入することも可能です。

次に、トランザクションを含むサンプル・テストの一部を、キーワード・ ビューに表示されるとおりに示します。

トランザクション開始	Services	StartTransaction	"ReserveSeat"	"ReserveSeat"トランザクションを開始する。
	👻 🎝 Find a Flight: Mercury			
	- 🐺 fromPort	Select	"Frankfurt"	"fromPort" listから "Frankfurt" メニュー項目を選択する。
	- 😽 fromMonth	Select	"Dec"	"fromMonth" list から "Dec" メニュー項目を選択する。
	- 😽 fromDay	Select	"29"	"fromDay" listから "29" メニュー項目を選択する。
	- 🔙 toPort	Select	"Acapulco"	"toPort" listから "Acapulco" メニュー項目を選択する。
	- 🔙 toMonth	Select	"Dec"	"toMonth" list から "Dec" メニュー項目を選択する。
	- 😽 toDay	Select	"31"	"toDay" listから "31" メニュー項目を選択する。
	🔊 servClass	Select	"Business"	"servClass" radio button group の中で ["Business"] ラジオ ボタンを
	🛄 findFlights	Click	37,5	"findFlights" image をクリックする。
	👻 🎝 Select a Flight: Mercury			
	- 🚚 reserveFlights	Click	63,12	"reserveFlights" image をクリックする。
トランザクション終了	— 🔄 🕂 🕂 — 🕂 — — — — — — — — — — — — — —	EndTransaction	"ReserveSeat"	"ReserveSeat"トランザクションを終了する。

エキスパート・ビューではテストのサンプル部分は次のように表示されます。

Services.StartTransaction "ReserveSeat" Browser("Welcome: Mercury Tours").Page("Find a Flight: Mercury"). WebList("fromPort").Select "London" Browser("Welcome: Mercury Tours").Page("Find a Flight: Mercury"). WebList("toPort").Select "Frankfurt" Browser("Welcome: Mercury Tours").Page("Find a Flight: Mercury"). WebList("toDay").Select "12" Browser("Welcome: Mercury Tours").Page("Find a Flight: Mercury"). WebRadioGroup("servClass").Select "Business" Browser("Welcome: Mercury Tours").Page("Find a Flight: Mercury"). WebList("airline").Select "Blue Skies Airlines" Browser("Welcome: Mercury Tours").Page("Find a Flight: Mercury"). Image("findFlights").Click 65,12 Browser("Welcome: Mercury Tours").Page("Select a Flight: Mercury"). WebRadioGroup("outFlight").Select "Blue Skies Airlines" Browser("Welcome: Mercury Tours").Page("Select a Flight: Mercury"). WebRadioGroup("inFlight").Select "Blue Skies Airlines" Browser("Welcome: Mercury Tours").Page("Select a Flight: Mercury"). Image("reserveFlights").Click 46.8 Services.EndTransaction "ReserveSeat"

ステップ・ジェネレータまたはエキスパート・ビューを使用して,さまざまな トランザクション関連ステートメントを挿入できます。詳細については, 『HP QuickTest Professional Object Model Reference』(英語版)の「Services」 の項を参照してください。「トランザクション開始」ステップおよび「トラン ザクション終了」ステップは、QuickTest ウィンドウ内のオプションを使用して 入力することもできます。

詳細については、以下を参照してください。

- ▶ 1398ページ「トランザクションの挿入」
- ▶ 1399ページ「トランザクションの終了」

トランザクションの挿入

テストの実行中,「トランザクション開始」は,時間測定の開始を示します。 [トランザクション開始]ダイアログ・ボックスでトランザクションの開始を 定義します。

トランザクションを挿入するには、次の手順を実行します。

- トランザクションの時間測定を開始するステップをクリックします。
 [ActiveScreen] タブに対応するページが表示されます。
- ▲ 2 [トランザクション開始] ボタンをクリックするか, [挿入] > [トランザク ション開始] を選択します。[トランザクション開始] ダイアログ・ボックス が開きます。

トランザクションの開始	×
🤹 名前: 📃	
ステートメントの挿入:	◎ 現在のステップの前追) ○ 現在のステップの後(A)
	OK キャンセル ヘルプ

3 [名前] ボックスに、わかりやすい名前を入力します。

注:トランザクション名にスペースを含めることはできません。

- 4 トランザクションの時間測定を開始する場所を決定します。
 - ▶ 現在のステップの前でトランザクションを終了するには、[現在のステップの前]を選択します。
 - ▶ 現在のステップの後でトランザクションを終了するには、[現在のステップの後]を選択します。
- 5 [OK] をクリックします。Start Transaction ステップがキーワード・ビューに 追加されます。

トランザクションの終了

テストの実行中,「トランザクション終了」で時間測定の終了を示します。[ト ランザクション終了]ダイアログ・ボックスでトランザクションの終了を定義 します。

注:実行セッション中にエラーが発生した場合でもトランザクション内のすべ てのステップを実行するよう QuickTest を設定したい場合があります。これに は、[テストの設定] ダイアログ・ボックス([ファイル] > [設定])の[実 行] タブで、[実行セッション中にエラーが発生した場合] リストから[次の ステップに進む] を選択します。これらの問題に対処するために、回復シナリ オまたはほかのエラー処理ステップを作成することもできます。詳細について は、第44章「回復シナリオの定義と使用」を参照してください。

トランザクションを終了するには、次の手順を実行します。

- トランザクションの時間測定を終了するステップをクリックします。
 [ActiveScreen] タブに対応するページが表示されます。
- 2 [トランザクション終了] ボタンをクリックするか, [挿入] > [トランザク ション終了] を選択します。[トランザクション終了] ダイアログ・ボックス が開きます。

トランザクションの終了
名前(N): ReserveSeat
ステートメントの挿入: 〇 現在のステップの前旧) 〇 現在のステップの後(A)
OK キャンセル ヘルプ

3 [名前] ボックスには、現在のテストでユーザが定義したトランザクション名 のリストが含まれています。終了するトランザクションの名前を選択します。

- 4 トランザクションを終了する場所を決めます。
 - ▶ 現在のステップの前でトランザクションを終了するには、[現在のステップの前]を選択します。
 - ▶ 現在のステップの後でトランザクションを終了するには、[現在のステップの後]を選択します。
- **5** [**OK**] をクリックします。**End Transaction** ステップがキーワード・ビューに 追加されます。

サイレント・テスト・ランナーの使用

サイレント・テスト・ランナーを使用して, LoadRunner および Business Availability Center からの QuickTest テストの実行方法をシミュレートできます。 サイレント・テスト・ランナーを使用してテストを実行すると, QuickTest の ユーザ・インタフェースを開かずにサイレント・テスト・ランナーが起動し, LoadRunner または Business Availability Center から実行したときと同じ速度でテ ストが実行されます。テスト実行の最後では、テスト実行とトランザクション 回数に関する情報を表示できます。

また,サイレント・テスト・ランナーを使用して,QuickTest テストが LoadRunner および Business Availability Center と互換性があることを確かめるこ ともできます。サイレント・テスト・ランナーを使用してテストを実行すると きに,LoadRunner または Business Availability Center でサポートされていない機 能を使用している場合,テストは失敗します。サポートされていない機能の詳 細については,1392ページ「LoadRunner または Business Process Monitor で使用 する QuickTest テストの設計」および 1393ページ「LoadRunner または Business Process Monitor でのテストの挿入と実行」を参照してください。
注:

- ➤ QuickTest がすでに開いている場合や、別のテストが現在実行中の場合は、 サイレント・テスト・ランナーを実行できません。QuickTest を閉じ、その プロセスが終了するまで待ってから、サイレント・テスト・ランナーを使用 してテストを実行する必要があります。
- ▶ 起動できるサイレント・テスト・ランナーのインスタンスは1つだけで、実行を指定できるテストも1つだけです。
- ▶ ネットワーク・パスを指定するには、ネットワーク・ドライブを割り当てる 必要があります。
- ▶ サイレント・テスト・ランナーでテストを実行しているときは、ResultDir QuickTest 環境変数は使用できません。

サイレント・テスト・ランナーを使用して QuickTest テストを実行するには, 次の手順を実行します。

 サイレント・テスト・ランナーを開くには、[スタート] ボタンから [QuickTest Professional] プログラム・グループを開き、[Tools] > [サイ レント テスト ランナー] を選択します。[サイレントテスト ランナー] ダイ アログ・ボックスが開きます。

🤹 サイレント テスト ランナー 📃 🗆 🗙					
	-テスト実行				
	ታスト:	l			参照(<u>B</u>)
		テストを実行(日)			
	ーテスト結果				
	ታスト	実行ログ(L)	トランザクション サマリ(王)		
			[閉じる(<u>C</u>)	ヘルプ(円)

- 2 テストの場所まで移動するために [参照] ボタンをクリックします。[テスト を開く] ダイアログ・ボックスが開き、< QuickTest Professional > ¥Tests フォルダにあるテストが表示されます。
- 3 実行するテストを選択し、[開く] をクリックします。[テストを開く] ダイア ログ・ボックスが閉じ、テスト名が [サイレント テスト ランナー] ダイアロ グ・ボックスの [テスト] ボックスに表示され、[テストを実行] ボタンが有 効になります。

注:以前に実行したテストを選択した場合は, [テスト実行ログ] ボタンと [トランザクション サマリ] ボタンが有効になり,選択したテストの前回の実 行に関する情報を表示できます。テストを初めて実行する場合は, [テスト実 行ログ] ボタンと [トランザクション サマリ] ボタンは無効になっています。

4 [テストの実行] をクリックして、テストを実行します。QuickTest ユーザ・インタフェースを開かずにテストが実行されます。テストの実行中は、「テストを実行しています…」というテキストが [テストを実行] ボタンの横に表示されます。

注:テストの実行を開始した後に,サイレント・テスト・ランナーからテスト の実行を停止することはできません。サイレント・テスト・ランナーを閉じて も,テストの実行は継続されます。テストを終了するには,mdrv.exe プロセ スを終了します。

5 テストの実行が完了すると、テキスト「テストを実行しています…」が「テス ト実行が完了しました」に変わります。サイレント・テスト・ランナーがテス トを実行できなかった場合は、「テストを実行できませんでした」というテキ ストが表示されます。[テスト実行ログ]ボタンが無効であった場合は、有効 になります。トランザクションのあるテストを実行した場合、[トランザク ションサマリ]ボタンが無効であった場合は、そのボタンも有効になります。 ログ・ファイルの表示の詳細については、1403ページ「テスト実行情報の表 示」を参照してください。

テスト実行情報の表示

サイレント・テスト・ランナーでは、テストの実行情報がログ・ファイルに記 録されます。テストごとにテスト実行ログが生成され、トランザクションのあ るテストの場合は追加のトランザクション・サマリが生成されます。

テスト実行ログの表示

テスト実行ログは、< QuickTest Professional > ¥Tests¥ <テスト名>フォル ダに output.txt として保存されます。ログ・ファイルはサイレント・テスト・ ランナーによるテスト実行のたびに保存され、テストを再実行すると上書きさ れます。ログ・ファイルを開くには、[テスト実行ログ]をクリックします。

ログ・ファイルにはテストの実行に関する情報が表示されます。たとえば、 個々の反復、アクション呼び出し、ステップ・トランザクション、失敗したス テップなどに関する情報が表示されます。各行には、メッセージまたはエラー ID が表示されます。ログ・ファイル内のメッセージおよびエラー・コードの詳 細については、Performance Center または Business Availability Center のマニュア ルを参照してください。

トランザクション・サマリの表示

トランザクション・サマリは,

< QuickTest Professional > ¥Tests¥ <テスト名>フォルダに transactions.txt として保存されます。トランザクション・サマリはトランザ クションを含むテストごとに保存され、テストを再実行すると上書きされま す。トランザクション・サマリを開くには、[トランザクションサマリ]をク リックします。トランザクション・サマリには、テスト内の各トランザクショ ンに対応した行が表示されます。各トランザクションについて、ステータス と、総継続時間および浪費時間(秒単位)が表示されます。サイレント・テス ト・ランナーでのトランザクション測定は、テストを LoadRunner または Business Availability Center から実行した場合とまったく同じになります。

注:

- ▶ トランザクション・サマリは、EndTransaction ステートメントで終わるト ランザクションが含まれているテストについてのみ生成されます。トランザ クションが開始されたものの、テストの失敗のために終了しなかった場合 は、そのトランザクションはトランザクション・サマリに記録されません。
- ▶ 分散トランザクション(あるテストで開始し、別のテストで終了するトラン ザクション)は、トランザクション・サマリには報告されず、テスト実行ロ グに記録されます。
- ▶ トランザクション・サマリに記録されているトランザクション情報はすべて、テスト実行ログにも記録されます。

第 XII 部



付録A

FAQ(よくある質問)

本章では、QuickTest の上級ユーザから寄せられることの多いいくつかの質問についてお答えします。質問と回答は次の項に分類されています。

本章では、次の項目について説明します。

- ▶ テストの作成(1407ページ)
- ➤ エキスパート・ビューでのプログラミング(1409ページ)
- ▶ 動的なコンテンツを使った作業(1411ページ)
- ▶ Web に関する高度な問題(1413ページ)
- ▶ 標準 Windows 環境(1415 ページ)
- ▶ テストの保守(1417ページ)
- ▶ ローカライズされたアプリケーションのテスト(1419ページ)
- ▶ QuickTest のパフォーマンスの向上(1420ページ)

テストの作成

➤ QuickTest でサポートされていないオブジェクトまたは環境で記録を実行する には、どのようにすればよいですか。

様々な方法があります:

➤ QuickTest Professional で使用可能な任意の外部アドインをインストールして ロードします。QuickTest では、Java、Oracle、.NET、SAP Solutions、Siebel、 PeopleSoft、ターミナル・エミュレータ、Web サービスなど、数多くの開発 環境がサポートされています。

- ➤ 識別されなかったクラスやユーザ定義のクラスのオブジェクトは、標準の Windows クラスにマップできます。オブジェクトのマッピングの詳細につ いては、224ページ「ユーザ定義のテスト・オブジェクト・クラスの割り当 て」を参照してください。
- > QuickTest が提供しているアドイン拡張機能を使用すると、QuickTest に組み 込まれている各種オブジェクトのサポートを拡張できます。これにより、オ ブジェクトを特定のテスト・オブジェクト・クラスに属するものとして認識 するよう QuickTest に指示でき、テスト・オブジェクトの振る舞いを指定で きるようになります。また、QuickTest が認識する使用可能なテスト・オブ ジェクト・クラスのリストを拡張することもできます。これにより、カスタ ム・オブジェクトの特定の振る舞いを完全にサポートするテストを作成でき ます。
- ▶ テスト・オブジェクトと同じように振る舞うオブジェクトに「仮想オブジェクト」を定義して、通常の記録モードで記録できます。仮想オブジェクトの定義の詳細については、第43章「仮想オブジェクトの学習」を参照してください。
- ➤ 低レベル記録あるいはアナログ・モードで、座標に基づいてクリックとキーボード入力を記録できます。低レベル記録とアナログ記録の詳細については、380ページ「記録モードの選択」を参照してください。

▶ アプリケーションをテストから起動するにはどうすればよいですか。

次のように、**SystemUtil** ステップをテストに追加することによって、テストの 中からアプリケーションを起動できます。

SystemUtil.Run "D:¥My Music¥Breathe.mp3","","D:¥My Music¥Details","open"

また、Windows ベース・アプリケーションの場合は、[記録と実行環境設定] ダイアログ・ボックスの[Windows アプリケーション] タブで、QuickTest が 開くアプリケーションを対象として記録と実行を行うように QuickTest を設定 していることを確認してください。

➤ QuickTest は、どのようにして Web ページでのユーザのプロセスをキャプ チャするのでしょうか。

QuickTest では、Microsoft Internet Explorer ブラウザにフックをかけます。ユー ザが Web ベースのアプリケーションを操作すると、QuickTest によって、ユー ザのアクションが記録されます(記録されたユーザ・アクションを変更する方 法については、『HP QuickTest Professional アドイン・ガイド』を参照してくだ さい)。記録したテストは、ステップを元の順序で実行する QuickTest の機能を 使って実行できます。

エキスパート・ビューでのプログラミング

▶ 関数やサブルーチンを関数ライブラリに保存できますか。

関数は個々のアクション内で定義できます。または、関数が含まれる1つまた は複数の VBScript 関数ライブラリを作成できます。そして、任意のテストから それらを呼び出すことができます。QuickTest 関数ライブラリ・エディタを使用 して、関数ライブラリの作成とデバッグを行うことができます。

関数を QuickTest テスト・オブジェクトのメソッドとして登録することもでき ます。登録したメソッドは、実行セッションの間だけ既存のテスト・オブジェ クト・メソッドの機能をオーバーライドしたり、テスト・オブジェクト・クラ スの新しいメソッドとして登録したりできます。

詳細については,第28章「ユーザ定義関数および関数ライブラリを使用した 作業」および第12章「アプリケーション領域の使用」を参照してください。

関数を再利用可能なアクションとして格納するのではなく関数ライブラリに格納することによって、QuickTestのパフォーマンスの向上を促進できます。

▶ 実行セッション中に情報を入力するにはどうすればよいですか。

VBScript の InputBox 関数を使用すると、ユーザに入力を求めるダイアログ・ ボックスを表示してからテストの実行を続けることができます。ユーザが入力 した値は、その後の実行セッションで使用できます。InputBox 関数の詳細につ いては、『VBScript Reference』(英語版)を参照してください。

次の例は、InputBox 関数を使用してパスワードの入力をユーザに求めます。

Browser("Mercury Tours").Page("Mercury Tours").WebEdit("username").Set "administrator"

Passwd = InputBox ("Enter password", "User Input")

Browser("Mercury Tours").Page("Mercury Tours").WebEdit("password").Set Passwd

➤ Microsoft Access データベースのデータをテストで使用したいと考えています。どうすればよいですか。

エキスパート・ビューでは, ADO および ODBC を使用してデータベースにア クセスできます。次の例は, データベースの "Authors" テーブル内で, ある著 者によって書かれた本を検索するテストです。

Dim MyDB Dim MyEng Set MyEng = CreateObject("DAO.DBEngine.35") Dim Td Dim rs

' 使用するデータベースを指定します。 Set MyDB = MyEng.OpenDatabase("BIBLIO.MDB")

```
'最初の10人の著者の名前を読み取って使用します。
```

Set Td = MyDB.TableDefs("Authors")

Set rs = Td.OpenRecordset

rs.MoveFirst

For i = 1 To 10

Browser("Book Club").Page("Search Books").WebEdit("Author Name").Set rs("Author")

Browser("Book Club").Page("Search Books").WebButton("Search").Click Next

▶ テスト結果をカスタマイズするにはどうすればよいですか。

次のように,**ReportEvent**メソッドを使用することで,メッセージをテスト結 果レポートに送信できます。

Reporter.ReportEvent 1, "Custom Step", "The user-defined step failed"

QuickTest 実行セッションの結果はすべて1つの.xml ファイル(results.xml という名前です)に保存されています。必要に応じて、このファイルに変更を加えることができます。QuickTest テスト結果のスキーマ(QuickTest Professional ヘルプから利用できます)は、テスト結果のカスタマイズに役立ちます。

動的なコンテンツを使った作業

▶ 表示するたびに動的に変化するオブジェクトを対象とした、テストを作成し、 実行するにはどうすればよいでしょうか。

アプリケーション内のオブジェクトで動的コンテンツを持つものは内容が変化 することがあります。正規表現, Description オブジェクト, リポジトリ・パ ラメータ, または SetTOProperty ステップを使用してテストを実行するとき に, それらのオブジェクトが QuickTest によって認識されるように, オブジェ クトの動的記述を作成できます。

▶ 子ウィンドウの有無を検査するには、どうすればよいですか。

1つのウィンドウ内のリンクが別のウィンドウを作成することがあります。

Exist プロパティを使用して、ウィンドウが存在するかどうかを検査できます。 例を次に示します。

If Window("Main").ActiveX("Slider").Exist Then

· · ·

また, ChildObjects メソッドを使用して, テスクトップ上あるいはほかの親オ ブジェクト内のすべての子オブジェクト(または, ある記述と一致する子オブ ジェクトのサブセット)を取得することもできます。

例:

```
Set oDesc = Description.Create
oDesc("Class Name").Value = "Window"
```

```
ser coll = Desktop.ChildObjects(oDesc)
For i = 0 to coll.count -1
    msgbox coll(i).GetROProperty("text")
```

Next

Exist プロパティおよび ChildObjects メソッドの詳細については, 『HP QuickTest Professional Object Model Reference』(英語版)を参照してくだ さい。

➤ QuickTest は、動的に生成される URL や Web ページをどのようにして記録す るのでしょうか。

QuickTest は、リンクがページに表示されると、実際にそのリンクをクリックします。そのため、QuickTest はオブジェクト自体ではなく、ページ上のリンクなど特定のオブジェクトを検索する方法を記録します。たとえば、動的に生成された URL へのリンクが画像である場合、QuickTest は「IMG」HTML タグと、その画像の名前を記録します。これにより、それ以後 QuickTest はこの画像を検索し、その画像をクリックできるようになります。

▶ QuickTest は、タブをどのように処理するのでしょうか。

QuickTest が提供しているいくつかのメソッドを **Browser** テスト・オブジェクトで使用することにより, Web ブラウザのタブを管理できます。

OpenNewTabは,現在のWebブラウザで新しいタブを開きます。

IsSiblingTabは、指定したタブが同じブラウザ・ウィンドウ内の現在のタブ・ オブジェクトの兄弟タブかどうかを示します。

Close は、タブが複数存在する場合は現在のタブを閉じ、ブラウザにタブが1 つしか含まれていない場合はブラウザ・ウィンドウを閉じます。

CloseAllTabs は、ブラウザ内のすべてのタブを閉じ、ブラウザ・ウィンドウ を閉じます。

これらの Browser 関連メソッドの詳細については, 『HP QuickTest Professional Object Model Reference』(英語版)の「Web」セクションを参照してください。

Web に関する高度な問題

➤ QuickTest はクッキーをどのように処理するのでしょうか。

CGI スクリプトなど接続のサーバ側では、クッキーを利用することで、接続の クライアント側に情報を格納したり、そこから情報を取得したりできます。

QuickTest ではユーザごとにメモリにクッキーを格納し、ブラウザは通常どおり にそれらを処理します。

➤ Web ページのクッキーはどこにありますか。

Internet Explorer ブラウザが使用するクッキーには, **.Object** プロパティを使用して, ブラウザの DOM (ドキュメント・オブジェクト・モデル) を通じてアクセスできます。次の例では, クッキーのコレクションがブラウザから返されます。

Browser("Flight reservations").Page("Flight reservations").Object.Cookie

➤ QuickTest は、セッション ID をどのように処理するのでしょうか。

ブラウザでなくサーバが,通常はクッキーによって,またはすべてのリンクに セッション ID を埋め込むことによって,セッション ID を処理します。これ は,QuickTest には影響を与えません。

➤ QuickTest は、サーバのリダイレクトをどのように処理するのでしょうか。

サーバがクライアントをリダイレクトした場合,通常はクライアントはそれに 気付かないため、リダイレクトの間違いが起こることはありません。ほとんど の場合、クライアントはサーバ上の別のスクリプトにリダイレクトされます。 この追加のスクリプトが、以降に表示されるページのHTMLコードを生成しま す。これは、QuickTestにもブラウザにも影響を与えません。

➤ QuickTest は、META タグをどのように処理するのでしょうか。

META タグは、ページの表示に影響を与えません。META タグには通常、ページの作成者、更新頻度、ページの内容説明、およびページの内容を表すキー ワードの情報だけが含まれています。したがって、QuickTest は問題なく META タグを処理できます。

➤ QuickTest は .asp および .jsp に対応していますか。

Active Server Page テクノロジを使用して動的に生成される Web ページには, .asp という拡張子が割り当てられています。Java Server Page テクノロジを使用して 動的に生成される Web ページには, jsp という拡張子が割り当てられています。 これらは完全にサーバ側の技術であるため, QuickTest には影響しません。

▶ QuickTest は, COM に対応していますか。

QuickTest は, COM 標準に準拠しています。

QuickTest は、Web ページに埋め込まれた COM オブジェクトをサポートしており(現在, COM オブジェクトは Microsoft Internet Explorer を使用している場合にだけアクセス可能です), VBScript 内で COM オブジェクトを駆動できます。

➤ QuickTest は, XML に対応していますか。

XML (eXtensible Markup Language) は、Web ドキュメント用に SGML を簡略 化したものです。XML を使えば、Web デザイナーはカスタマイズした独自の タグを作成できます。QuickTest は XML に対応しており、XML タグをオブ ジェクトとして認識します。

また,Webページ,Webフレーム,WebファイルのXMLドキュメントの内容 を検査するXMLチェックポイントを作成できます。QuickTestはXML出力と スキーマ検証もサポートしています。

詳細については,第21章「XMLの検査」,および『**HP** QuickTest Professional Object Model Reference』(英語版)の「Utility」セクションの XMLUtil オブ ジェクトを参照してください。

▶ HTML タグに直接アクセスするにはどうすればよいですか。

QuickTest では, Internet Explorer の DOM (ドキュメント・オブジェクト・モデ ル) に直接アクセスでき, DOM を通じて HTML タグに直接アクセスできます。 DOM へのアクセスは .Object の表記法を使用して実行します。

次のテストは, Internet Explorer ページ内のすべてのタグについて反復処理を行 う方法の例です。反復処理後,このテストは Reporter オブジェクトを使用して タブの内部テキスト(タグに囲まれたテキスト)をテスト結果に出力します。

' すべての要素が内部テキストを持っているわけではないため, On Error オプ ションを使用します。

On Error Resume Next

Set Doc = Browser("CNN Interactive").Page("CNN Interactive").Object

'ページ内のすべてのオブジェクトについてループ処理します。 For Each Element In Doc.all

TagName = Element.TagName ' タグ名を取得します。 InnerText = Element.innerText ' 内部テキストを取得します。

'情報をテスト結果に書き込みます。

Reporter.ReportEvent 0, TagName, InnerText Next

➤ Internet Explorer のドキュメント・オブジェクト・モデルに関する情報はどこ で入手できますか。

Internet Explorer の DOM の詳細については, 次の Web サイトを参照してください。

ドキュメント・オブジェクト

http://msdn2.microsoft.com/en-us/library/ms531073.aspx

その他の DHTML オブジェクト http://msdn2.microsoft.com/en-us/library/ms533054.aspx

DHTML の全般的なリファレンス http://msdn2.microsoft.com/en-us/library/ms533050.aspx

▶ キーボードのキーのコマンド(ショートカット・コマンドなど)を, Type メ ソッドをサポートしていないオブジェクトに送信するにはどうすればよいです か。

Type メソッドをサポートしていないオブジェクトの場合は,Windows Scripting の **SendKeys** メソッドを使用します。詳細については,『Microsoft VBScript Language Reference』を参照してください([**ヘルプ**] > [**QuickTest Professional ヘルプ**] > [**VBScript Reference**] > [**Windows Script Host**] を選択します)。

標準 Windows 環境

▶ 非標準のメニューを対象とした記録はどのように実行すればよいですか。

メニューを記 録するときの QuickTest の振る舞いを変更できます。この振る舞 いを制御するオプションは、[Windows Application 詳細設定オプション] ダイ アログ・ボックスにあります([ツール] > [オプション] > [Windows アプ リケーション] > [詳細設定])。

詳細については,『HP QuickTest Professional アドイン・ガイド』を参照してく ださい。

▶ 応答していないアプリケーションを強制終了するにはどうすればよいですか。

次のいずれかのステップをテストに追加することにより、QuickTest でのテストの実行中に任意の標準のアプリケーションを強制終了できます。

- ► SystemUtil.CloseProcessByName "app.exe"
- SystemUtil.CloseProcessByWndTitle "Some Title"

▶ 実行セッション中にクリップボードを介したコピーや貼り付けはできますか。

Clipboard オブジェクトを使用して, QuickTest の実行セッション中にテキストのコピー, 切り取り, 貼り付けを実行できます。

Clipboard オブジェクトには Visual Basic で使用できる Clipboard オブジェクトと 同じメソッドがあります。

- ► Clear
- ► GetData
- ► GetFormat
- ► GetText
- ➤ SetData
- ► SetText

これらのメソッドの詳細については, http://msdn.microsoft.com/library/en-us/vb98/html/vbobjclipboard.asp?frame=true を参照してください。

Clipboard オブジェクトの使用例を次に示します。

Set MyClipboard = CreateObject("Mercury.Clipboard") MyClipboard.Clear MyClipboard.SetText "TEST" MsgBox MyClipboard.GetText

テストの保守

▶ アプリケーションに変更が加えられた場合、テストをどのように保守すればよいですか。

アプリケーションに変更が加えられた場合のテストの保守方法は、アプリケー ションに加えられた変更の量に応じて異なります。アプリケーション全体を対 象に1つの大きなテストを作成するのではなく、テストを小さなグループに分 けて作成するべき主な理由の1つがこれです。

また、QuickTestのアクションを使用して、よりモジュール化され、より効果的 なテストを設計できます。テストを機能ごとにいくつかのアクションに分割し ます。アプリケーションに変更が加えられたら、特定のアクションだけを変更 するだけでよく、テストのほかの部分は変更せずに済みます。可能なかぎり、 複数のテストにまったく同じスクリプトを作成するのではなく、再利用可能な アクションの呼び出しを挿入するようにします。こうすることで、元の再利用 可能なアクションに変更を加えるだけで、そのアクションを呼び出すすべての テストに変更が自動的に適用されます。詳細については、第13章「高度なア クション機能を使用した作業」を参照してください。

同じテスト・オブジェクトが含まれるテストやアクションが数多くある場合 は、1か所で集中的にオブジェクト情報を更新できるように、共有オブジェク ト・リポジトリを使用することをお勧めします。

[**更新モード**] オプションを使用すると、チェックポイントまたは ActiveScreen の変更された情報を更新したり、アプリケーション内のオブジェクトの識別に 使用されている1組のテスト・オブジェクト・プロパティを変更したりできま す。詳細については、1076ページ「更新モード・オプションを使ったテストの 更新」を参照してください。

オブジェクト・リポジトリに保存されているオブジェクト・プロパティ値とア プリケーション内のオブジェクト・プロパティ値との間で不一致がある場合 は,[メンテナンス実行モード]を使用して不一致の修正に役立てることがで きます。メンテナンス実行モードでテストを実行すると,QuickTestによってテ ストが実行された後,オブジェクト・リポジトリの不一致が原因で実行できな いステップが出現するたびに,ステップとオブジェクト・リポジトリを更新す るためのプロセスがひととおり示されます。詳細については,1060ページ「メ ンテスナンス実行ウィザードを使ったテストの実行」を参照してください。

▶ テストの記録終了後、ActiveScreen 情報を増やしたり減らしたりできますか。

記録後に ActiveScreen に保存された情報が、テストを編集するには不十分な場 合や、ActiveScreen 情報が不要になり、テストのサイズを小さくする場合に、 テストに保存されている ActiveScreen 情報の量を変更する方法はいくつかあり ます。

- ▶ テストによって使用されるディスク容量を減らすには、「名前を付けて保存」 を選択して [ActiveScreen ファイルを保存する] チェック・ボックスをク リアすることで ActiveScreen 情報を削除します。詳細については、346ペー ジ「テストの保存」を参照してください。
- ➤ Windows または Java アプリケーションの記録時に ActiveScreen の情報をすべて保存しないことを選択した場合,次の方法のどれかを使用して, ActiveScreen に格納される情報を増やすことができます。

[オプション] ダイアログ・ボックスの [ActiveScreen] タブで, ActiveScreen のキャプチャ設定が必要な量の情報をキャプチャするように設 定されているか確認します。次に以下のことを行います。

- ▶ [更新モード] を実行し、既存のすべてのステップについて必要な量の情報を ActiveScreen に保存します。[更新モード] オプションの詳細については、1076ページ「更新モード・オプションを使ったテストの更新」を参照してください。
- ▶ ActiveScreen に追加するオブジェクトが含まれるステップを再記録します。

ステップを再記録するには、記録するステップの前のステップを選択し、 テストで選択された位置と一致するようにアプリケーションを配置して から、記録を開始します。あるいは、追加するステップの前のステップ でテストにブレークポイントを設定し、そのブレークポイントまでテス トを実行します。これにより、ステップの記録の開始点に移動します。 ブレークポイントの設定の詳細については、1047ページ「ブレークポイ ントの設定」を参照してください。

Windows アプリケーションで ActiveScreen に保存される情報量の変更に関する 詳細については, 1172 ページ「ActiveScreen オプションの設定」を参照してく ださい。

▶ 古いテストからのテスト結果ファイルを削除するにはどうすればよいですか。

テスト結果削除ツールを使用して,ファイル・システムまたは Quality Center プロジェクト内の特定の場所にある,すべてのテスト結果の一覧を表示できます。その後不要なテスト結果を削除できます。

削除するテスト結果をより簡単に特定できるように,テスト結果削除ツールに よって,テスト結果を名前,日付,サイズなどで並べ替えることができます。

このユーティリティを開くには, [スタート] > [プログラム] > [QuickTest Professional] > [Tools] > [Test Results Deletion Tool] を選択します。

ローカライズされたアプリケーションのテスト

 あるアプリケーションについて、ローカライズされたいくつかのバージョンを テストしています。それぞれのバージョンには、ローカライズされたユーザ・ インタフェース文字列が含まれています。QuickTest で効率的なテストを作成 するには、どのようにすればよいですか。

これらのユーザ・インタフェース文字列は、グローバル環境変数リストにある パラメータを使ってパラメータ化できます。グローバル環境変数リストは、変 数と、それに対応する値のリストで、任意のテストからアクセスできます。詳 細については、第22章「値のパラメータ化」を参照してください。

▶ あるアプリケーションについて、ローカライズされたいくつかのバージョンを テストしています。テストの際に、アプリケーションの言語に応じて異なる データを効率的に入力するにはどうすればよいですか。

テストの反復を1回だけ実行する場合,あるいはアクションまたはテストのす べての反復で変数の値を変えない場合は,環境変数を使用し,テストの実行ご とにアクティブな環境変数ファイルを切り替えます。

テストまたはアクションの反復を複数回実行し、入力データを反復ごとに変更 する場合は、アプリケーションのローカライズ・バーションごとに外部デー タ・テーブルを作成します。テスト対象を別のローカライズされたバージョン に変更するときには、[テストの設定]ダイアログ・ボックスの[リソース] タブでテスト用のデータ・テーブル・ファイルを別のデータ・テーブル・ファ イルに切り替えます。データ・テーブルを使った作業の詳細については、第38 章「データ・テーブルを使った作業」を参照してください。テストのデータ・ テーブル・ファイルの選択に関する詳細については、1199ページ「テストのた めのリソース設定の定義」を参照してください。

QuickTest のパフォーマンスの向上

▶ QuickTest の動作速度を向上させるには、どうすればよいですか。

QuickTest の動作速度を向上させるには、次のいずれかを行います。

- ▶ アドイン・マネージャで、QuickTestの起動時に特定のQuickTestセッション に必要なアドインのみをロードします。こうすることで、実行セッション中 のオブジェクトの学習時のパフォーマンスが向上します。アドインのロード の詳細については、『HP QuickTest Professional アドイン・ガイド』を参照 してください。
- ▶ テスト内のアクションの数を最小限に抑えます。理想的には、テストに含めるアクションの数は数十個程度にします。
- ▶ 関数を再利用可能なアクションとして格納するのではなく関数ライブラリに 格納します。
- ▶ テストを「高速モード」で実行します。それには、[オプション] ダイアロ グ・ボックスの[実行] タブで、[高速] オプションを選択します。これに より、QuickTest は各ステップで実行矢印を表示せずにテストを実行するた め、テストの実行を高速化できます。[オプション] ダイアログ・ボックス の[実行] タブの詳細については、1181ページ「テストの実行オプション の設定」を参照してください。
- ▶ テストを編集する際に ActiveScreen を使用していない場合は、テストの編集中に ActiveScreen を非表示にしておくことで、編集時の応答時間を改善します。これを行うには、[表示] > [ActiveScreen] を選択するか、または[ActiveScreen] ツールバー・ボタンをクリックして、ActiveScreen を非表示に切り替えます。詳細については、第2章「QuickTest の概要」を参照してください。

- ▶ 情報をキャプチャして ActiveScreen に保存する条件とその量を指定します。 キャプチャする情報が多いほど、さまざまな ActiveScreen オプションを使用 してテストにステップを追加するのが容易になります。しかし、キャプチャ する情報が多いと、記録や編集を行うのに時間がかかるようになります。パ フォーマンスを向上させるために選択できる ActiveScreen オプションは、次 のとおりです。
 - ➤ Windows アプリケーションをテストしている場合は、あらゆるステップのActiveScreen 情報をすべて保存する、特定のステップのActiveScreen 情報だけを保存する、ActiveScreen のキャプチャを完全に無効化する、といった選択ができます。この設定は、[オプション]ダイアログ・ボックスの[ActiveScreen] タブで行います。詳細については、1172ページ「ActiveScreen オプションの設定」を参照してください。
 - ➤ Web アプリケーションをテストしている場合は、ActiveScreen でのすべてのステップの画面キャプチャを無効にします。[オプション]ダイアログ・ボックスの[ActiveScreen]タブで、[ユーザ定義レベル]をクリックして [Active Screen キャプチャのユーザ定義設定]ダイアログ・ボックスを開きます。

次に, [Disable Active Screen Capture] オプションを選択します。こ れにより,記録時間を短縮できます。[オプション] ダイアログ・ボック スの [ActiveScreen] タブの詳細については,1172ページ「ActiveScreen オプションの設定」を参照してください。

▶ 新規テストを保存する場合、または [名前を付けて保存] を使用してテ ストを新しい名前で保存する場合は、[上書き保存] または [名前を付け て保存] ダイアログ・ボックスの [ActiveScreen ファイルを保存する] オプションをクリアにすることで、テストでキャプチャされた ActiveScreen ファイルを保存しないように選択します。これは、テスト の設計が完了し、テスト実行のためだけにテストを使用する場合に特に 便利です。ActiveScreen ファイルのないテストは、速く開き、占有する ディスク容量が極めて少なくなります。[オプション] ダイアログ・ボッ クスの [ActiveScreen] タブの詳細については、1172 ページ 「ActiveScreen オプションの設定」を参照してください。 **ヒント**: ActiveScreen ファイルなしでテストを保存した後で ActiveScreen ファ イルを回復する必要が生じた場合は、必要なステップを再び記録するか、「更 新モード]オプションを使用してテストのすべてのステップの画面を再キャプ チャします。詳細については、1076ページ「更新モード・オプションを使った テストの更新」を参照してください。

- ▶ テスト結果としてアプリケーションの画像やムービーをキャプチャし、保存 するタイミングを指定します。[オプション]ダイアログ・ボックスの[実 行]タブで、[静止画像キャプチャをテスト結果へ保存]ボックスからオプ ションを選択するか、[ムービーをテスト結果へ保存]チェック・ボックス を選択して必要な設定を指定します。エラー発生時などの特定の条件を満た した場合にだけ画面キャプチャやムービー・セグメントを保存したり、画像 をまったく保存しないようにすることで、実行時間を短縮し、ディスク容量 を節約できます。[オプション]ダイアログ・ボックスの[実行]タブの詳 細については、1181ページ「テストの実行オプションの設定」を参照して ください。
- ▶ テスト結果レポートを一時フォルダに保存し、テストを実行するたびに以前の実行セッションからの結果を上書きします。詳細については、938ページ「テスト全体の実行」を参照してください。
- ▶ 結果削除ツールを使い、定義した条件に従って、システムから不要なテスト 結果や古いテスト結果を削除できます。これによって貴重なディスク・ス ペースを解放できます。詳細については、981ページ「テスト結果削除ツー ルを使った結果の削除」を参照してください。

▶ QuickTest が使用するディスク容量を減らすには、どうすればよいですか。

QuickTest が使用するディスク容量を減らすには、次のいずれかを行います。

▶ テスト結果としてアプリケーションの画像をキャプチャし、保存するタイミングを指定する。それには、[オプション]ダイアログ・ボックスの[実行] タブにある [静止画像キャプチャをテスト結果へ保存]ボックスでオプションを選択します。特定の条件を満たした場合にだけ画面キャプチャを保存したり、画像を保存しないようにすることで、ディスク容量を節約し、テストの実行時間を短縮できます。[オプション]ダイアログ・ボックスの [ActiveScreen] タブの詳細については、1172ページ「ActiveScreen オプションの設定」を参照してください。

- ▶ 情報をキャプチャして ActiveScreen に保存する条件とその量を指定します。 キャプチャする情報が多いほど、さまざまな ActiveScreen オプションを使用 してテストにステップを追加するのが容易になります。しかし、キャプチャ する情報が多いと、記録や編集を行うのに時間がかかるようになります。パ フォーマンスを向上させるために選択できる ActiveScreen オプションは、次 のとおりです。
 - ➤ Windows アプリケーションをテストしている場合は、あらゆるステップのActiveScreen 情報をすべて保存する、特定のステップのActiveScreen 情報だけを保存する、ActiveScreen のキャプチャを完全に無効化する、といった選択ができます。この設定は、[オプション]ダイアログ・ボックスの[ActiveScreen]タブで行います。詳細については、1172ページ「ActiveScreen オプションの設定」を参照してください。
 - ➤ Web アプリケーションをテストしている場合は、ActiveScreen でのすべてのステップの画面キャプチャを無効にします。[ActiveScreen] タブで、 [ユーザ定義レベル]をクリックして [Active Screen キャプチャのユーザ 定義設定] ダイアログ・ボックスを開きます。次に、[Disable Active Screen Capture] オプションを選択します。これにより、記録時間を短縮できます。[オプション]ダイアログ・ボックスの [ActiveScreen] タブの詳細については、1172ページ「ActiveScreen オプションの設定」を参照してください。
 - ▶ 新規テストを保存する場合、または [名前を付けて保存] を使用してテ ストを新しい名前で保存する場合は、[上書き保存] または [名前を付け て保存] ダイアログ・ボックスの [ActiveScreen ファイルを保存する] オプションをクリアにすることで、テストでキャプチャされた ActiveScreen ファイルを保存しないように選択します。これは、テスト の設計が完了し、テスト実行のためだけにテストを使用する場合に特に 便利です。ActiveScreen ファイルのないテストは、使用されるディスク 容量が極めて少なくなります。

ヒント: ActiveScreen ファイルなしでテストを保存した後で ActiveScreen ファ イルを回復する必要が生じた場合は、必要なステップを再び記録するか、「更 新モード]オプションを使用してテストのすべてのステップの画面を再キャプ チャします。詳細については、1076ページ「更新モード・オプションを使った テストの更新」を参照してください。

▶ 推奨されるテストの長さはありますか。

テストの長さについて公式の制限はありませんが、テストをアクションに分割し、 可能な限り、再利用可能なアクションを使用することを推奨します。アクション に含むステップ数は、200~300までにしてください。数十までが理想的です。詳 細については、第14章「アクションを使った作業」を参照してください。

付録 B

カスタムのプロセス・ガイダンス・パッケージ の作成

本章では、カスタムのプロセス・ガイダンス・パッケージの作成プロセスをひ ととおり説明します。カスタム・パッケージは組織の QuickTest ユーザに配布で きます。QuickTest ユーザは、作業時に QuickTest でカスタム・パッケージのプロ セスを表示し、組織のプロセスと標準に従う上で役立てることができます。

本章では、次の項目について説明します。

- ▶ プロセス・ガイダンス・パッケージについて(1425ページ)
- ▶ パッケージ設定ファイルについて(1426ページ)
- ▶ データ・ファイルの作成(1428ページ)
- ➤ QuickTest でのカスタムのプロセス・ガイダンス・パッケージのインストール (1429 ページ)

プロセス・ガイダンス・パッケージについて

プロセス・ガイダンス・パッケージは,パッケージ設定ファイルとデータ・ファイルの2つの要素で構成されています。

- ➤ パッケージ設定ファイル:この XML ファイルでは、パッケージに含まれる 「プロセス」と、各プロセスの「グループ」および「アクティビティ」の構造 を定義します。
- ▶ データ・ファイル:1組の HTML ファイルで、各 HTML ファイルには単一のア クティビティの内容が含まれています。

パッケージ設定ファイルについて

新しいパッケージを作成するには、最初に、パッケージに含めるプロセスを記述し、各プロセスのグループとアクティビティの構造を設定する、XMLファイルを作成します。この構造は、QuickTestの[プロセスガイダンスのアクティビティ]表示枠で選択されたプロセスの内容のテーブルとして表示されます。

重要:設定ファイルは Configuration.xml という名前で保存します。

2つのプロセスが含まれているパッケージ設定ファイルの例を次に示します。

<?xml version="1.0" encoding="UTF-8" ?>

<ProcessGuidance Name="MyCustomPackage">

<Process Name="My Process" ID="Process1" DocType="test" Addin="web" SortLevel="4" >

<Group Name="New User Overview">

```
<Activity Name="Step 1" Address="Step1.html" />
```

<Activity Name="Step 2" Address="Step2.html" />

</Group>

</Process>

<Process Name="Important Processes" ID="Process2" DocType="test|AA" SortLevel="3">

<Group Name="Getting Started">

<Activity Name="Open" Address="F:¥ProcessData¥open.html" />
<Activity Name="Create" Address="F:¥ProcessData¥create.html" />
<Activity Name="Test" Address="F:¥ProcessData¥test.html" />
<Activity Name="Debug" Address="F:¥ProcessData¥debug.html" />
</Group>

<Group Name="Finish">

<Activity Name="Save" Address="F:\ProcessData\save.html" />
<Activity Name="Close" Address="F:\ProcessData\close.html" />
<Activity Name="Exit" Address="F:\ProcessData\exit.html" />
</Group>

</Process>

</ProcessGuidance>

XML の詳細

本項では、パッケージ設定ファイルで使用できる要素と属性について説明します。

- ➤ < Process > 要素:新しいプロセスを定義します。この要素では次の属性がサ ポートされています。
 - ► Name : QuickTest の [プロセス ガイダンス] 表示枠に表示するプロセスの 名前。
 - ► ID: 一意の識別名。この名前は、同じ名前を持つ2つのプロセスを区別するのに使用されます。
 - DocType: このプロセスが適用可能な QuickTest のドキュメント・タイプを示します。指定した場合は、関連するドキュメント・タイプが開いているときのみ、プロセスを使用できます。

前述の例では、QuickTest ユーザがテスト・ドキュメントを開いている場合 は両方のプロセスを使用できますが、アプリケーション領域ドキュメントが 開いている場合は2番目のプロセスのみ使用できます。

取り得る値:

- ▶ test : テスト・ドキュメント。
- ► AA:アプリケーション領域ドキュメント。
- ▶ BC : ビジネス・コンポーネント・ドキュメント。
- ► SBC : スクリプト・コンポーネント・ドキュメント。
- Addin:このプロセスが適用可能なQuickTestアドインを示します。指定した場合は、関連するアドインがロードされているときのみ、プロセスを使用できます。

前述の例では,Webアドインがロードされている場合にのみ,最初のプロ セスを使用できます。2番目のプロセスは常に表示されます。

アドイン・マネージャに表示されるアドイン名を使用して,アドインの値を 指定します。

➤ SortLevel: プロセス・リスト内でのプロセスの位置を決めます。このリストは、[プロセスガイダンス管理]ダイアログ・ボックス、およびQuickTestの[オートメーション]> [プロセスガイダンスリスト]メニューに表示されます。

- > < Group > 要素: プロセス内の新しいグループを定義します。この要素では次の属性がサポートされています。
 - ► Name:前述の < Process > 要素の Name 属性と同じです。
 - ► ID:前述の < Process > 要素の ID 属性と同じです。
 - ► Addin:前述の < Process > 要素の Addin 属性と同じです。
- ▶ < Activity > 要素:グループ内部のアクティビティを定義します。
 - ▶ Name:前述の < Process > 要素の Name 属性と同じです。
 - ► ID:前述の < Process > 要素の ID 属性と同じです。
 - ➤ Addin:前述の< Process > 要素の Addin 属性と同じです。
 - > Address:該当する HTML データ・ファイルのパス。ファイル・システム または HTTP アドレス上の、ローカル・パスまたはネットワーク・パスにな ります。相対パスを指定した場合は、設定ファイルの場所を基準とする相対 的な場所に解決されます。

データ・ファイルの作成

各データ・ファイルには、単一のプロセス・ガイダンス・アクティビティの HTML コンテンツが含まれています。[プロセス ガイダンスのアクティビティ] 表示枠でアクティビティ・リンクをクリックすると、QuickTest の[プロセス ガイダンスの説明] 表示枠のブラウザ・コントロールに HTML コンテンツが表 示されます。

パッケージ・データ・ファイルには、内容を組織の標準のスタイルで表示する ための.CSS ファイルへの参照を含めることができ、ブラウザで表示できる任意 の内容を含めることができます。

HTML ファイルと、その HTML ファイルが参照しているフォルダやファイルは すべて、ユーザのローカルのハード・ディスク・ドライブ、ファイル・システ ムのネットワーク上の場所、または Web サーバに格納できます。パッケージ設 定ファイル(各 Activity 要素の Address 属性)によって、各アクティビティ の HTML リンクが提供されます。

各アクティビティの HTML ファイルは,その内容が [プロセス ガイダンスの 説明]表示枠で標準のサイズで表示されるときに最小限のスクロールで済むよ うに記述することをお勧めします。 HTML ファイルが長すぎる場合は、QuickTest ユーザが作業時に参照しやすいように、ファイルを複数のプロセス・ガイダンス・アクティビティに分割することもできます。

QuickTest でのカスタムのプロセス・ガイダンス・パッケージのインストール

次の2つの方法で,カスタムのプロセス・ガイダンス・パッケージを配布およ びインストールできます。

- ▶ zip ファイルからのプロセス・ガイダンス・パッケージのインストール
- ▶ レジストリ・キーによるプロセス・ガイダンス・パッケージのインストール

zip ファイルからのプロセス・ガイダンス・パッケージのインストール

- Configuration.xml ファイルとすべての HTML データ・ファイル (および, HTML ファイル参照されるすべてのファイルまたはフォルダ)を格納するフォ ルダを作成します。
- 2 フォルダを zip ファイルに圧縮し、その.zip ファイルを関連するすべての QuickTest ユーザに送信するか、ユーザがアクセスできる場所に格納します。
- **3** QuickTest で, [**ファイル**] > [**プロセス ガイダンス管理**] を選択します。[プ ロセス ガイダンス管理] ダイアログ・ボックスが開きます。
- 4 [追加] ボタンをクリックし, .zip ファイルを参照します。パッケージが追加さ れ,そのプロセスがダイアログ・ボックスに表示されます。

レジストリ・キーによるプロセス・ガイダンス・パッケージのインストール

- **1 Configuration.xml**ファイルとデータ・ファイルを用意します。
- データ・ファイルを、ローカルまたは共有ネットワーク・フォルダ、または Web サーバに置きます。Configuration.xml ファイルで、Activity 要素の Address 属性がこの場所を指し示していることを確認します。
- **3 Configuration.xml** を QuickTest コンピュータのローカル・ドライブにコピーします。
- **4** レジストリ・エディタを開き,次のキーを探します。

HKEY_LOCAL_MACHINE¥SOFTWARE¥Mercury Interactive¥QuickTest Professional¥MicTest¥ProcessGuidance¥ConfFiles

5 このキーに, **Configuration.xml** ファイルへのパスを値として追加します。次回 QuickTest を開いたときに,新しいパッケージが追加されます。

索引

この索引は、『HP QuickTest Professional ユーザーズ・ガイド』の第1巻と第2巻両方の 項目を含みます。

A

ActionIteration, 環境変数 666 Active Server Page テクノロジ 1413 ActiveScreen 388 Web 設定の定義 1179 更新 391 標準のキャプチャ設定 1175 ファイルの保存と削除 346 保存される情報を増やす / 減らす 1418 ActiveScreen キャプチャのユーザ設定ダイアロ グ・ボックス 1175 API, Windows の使用 875 ASCII 1132 ASP ファイル 1413 attribute/ <プロパティ名>表記法 874

B

Business Process Testing 1367 役割 1368 ワークフロー 1371 Business Process Testing の役割 1368 Business Process Monitor, QuickTest との統合 1389 Business Process Testing でのワークフロー 1371

С

CGI スクリプト 1413 Close メソッド 862 COM 1414 ControllerHostName, 環境変数 666 CreationTime 識別子 「序数識別子」*を参照* CreationTime プロパティ,オブジェクトの識 別に使用 210

D

Dictionary オブジェクト 460 Dim ステートメント,エキスパート・ビュー および関数ライブラリにおける 844 Do...Loop ステートメント,エキスパート・ ビューおよび関数ライブラリにおける 867 Domain コマンド・ライン・オプション 984 DOS コマンド,テスト内で実行 875

Е

Excel,「Microsoft Excel」を参照 Excel の数式 値をパラメータ化するための 1145 チェックポイントでの 1146 データ・テーブルでの 1145 ExecuteFile 関数 933 ExecuteFile ステートメント 906 Exist ステートメント 812 Exit プロパティ 1411 eXtensible Markup Language (XML) 1414

F

FAQ 1407 For...Each ステートメント,エキスパート・ビュー および関数ライブラリにおける 867 For...Next ステートメント,エキスパート・ビュー および関数ライブラリにおける 866 FromDate コマンド・ライン・オプション 984

G

GetROProperty メソッド 872 GroupName, 環境変数 666

H

HP Software Web サイト xxv

HTML ファイルにエクスポート・ダイアロ グ・ボックス 979

Ι

If...Then...Else ステートメント,エキスパー ト・ビューおよび関数ライブラリにお ける 869 Index プロパティ プログラム的記述 858 IntelliSense 824, 884

J

JavaScript 1314

L

LoadRunner, QuickTest との統合 1389 LocalHostName, 環境変数 666 Log コマンド・ライン・オプション 984

М

Mercury Application Management, QuickTest と の統合 1389 Mercury Micro Player 974 Mercury Quality Center,「Quality Center」を参照 Mercury Tours,サンプル・アプリケーション 42 Mercury カスタマー・サポート Web サイト xxv META タグ 1413 Microsoft Excel 1132, 1145 Microsoft Query データベース・チェックポイントのた めのデータベースの選択 595, 1144 Microsoft の Visual Basic スクリプト言語 38 MinSize コマンド・ライン・オプション 985 共有オブジェクト・リポジトリ・パラメータ

の割り当てダイアログ・ボックス 158

N

Name コマンド・ライン・オプション 985

0

Object プロパティ,実行環境メソッドの 874
 ODBC,データベース・チェックポイントのためのデータベースの選択 1144
 Option Explicit ステートメント 931
 OSVersion,環境変数 666

OS,環境変数 666 output.txt ログ・ファイル 1403

Р

Parameter 予約済みオブジェクト 1207 Password コマンド・ライン・オプション 986 PathFinder.Locate, ステートメント 1172 Print ユーティリティ・ステートメント 805 ProductDir, 環境変数 666 ProductVer, 環境変数 666 Project コマンド・ライン・オプション 986

Q

QA エンジニア、「自動化エンジニア」参照 OCUtil オブジェクト 1332 Quality Center 1321 QuickTest からの接続 1323 OuickTest での使用 39 QuickTest との統合 1332 環境変数ファイル 665 関連付けられている関数ライブラリ 905 接続アドイン 1332 切断 1329 データ・テーブル 1141 テスト・プロセスの管理39 テストを開く1334 バージョン・コントロール 1348 不具合の報告 自動 990 手動 989 プロジェクトへのテストの保存1333 リモートでの OuickTest テストの実行 1359 Quality Center OTA 1332 Quality Center からの切断 1329 Quality Center への QuickTest の接続 1323 Ouality Center からテストを開くダイアログ・ ボックス 1335, 1338 Ouality Center にテストを保存ダイアログ・ ボックス 1333 Quality Center への接続 - サーバへの接続ダイ アログ・ボックス 1324 Quality Center への接続ダイアログ・ボックス 1325 Quality Center への接続 - プロジェクトへの接 続ダイアログ・ボックス 1327

OuickTest Mercury のアプリケーション管理製品 およびパフォーマンス・テスト製品 との統合1389 アクセス許可, 必要41 ウィンドウ 「OuickTest ウィンドウ」を 参照 オートメーション・オブジェクト・モ デル 1311 開始 45.46 製品情報 94 説明 29 ソフトウェアの更新43 レイアウト1087 レイアウトのカスタマイズ1087 QuickTest Print Log ウィンドウ 805 QuickTest Professional のバージョン情報ウィン ドウ94 OuickTest ウィンドウ アクション・ツールバー48,68 オートメーション・ツールバー 49,66 欠落リソース 62 情報表示枠 48,58 ステータス・バー 50 挿入ツールバー 67 タイトル・バー 50 タブの移動1088 ツール・ツールバー 67 データ・テーブル50 テーマ 52 デバッグ・ツールバー48 表示ツールバー 68 標準ツールバー 66 標準のレイアウトの復元 1097 表示枠の移動 1088 表示枠の自動非表示 1093 ファイル・ツールバー 50 複数のドキュメント 1097 編集ツールバー 67 メニュー・バー48 ルック・アンド・フィール 52 レイアウトのカスタマイズ 1087 QuickTest オートメーション・リファレンス 1318 OuickTest テストを開くダイアログ・ボックス 344, 346 OuickTest の開始 46

QuickTest のテスト・バージョン 1348 QuickTest ヘリモート・アクセス 1359

R

Recursive コマンド・ライン・オプション 986 RegisterUserFunc ステートメント 924 ResultDir, 環境変数 666

S

ScenarioId, 環境変数 667 Server コマンド・ライン・オプション 987 Setting オブジェクト 1222 SetTOProperty メソッド 190 Set ステートメント, エキスパート・ビューお よび関数ライブラリにおける 843 SGML 1414 Silent コマンド・ライン・オプション 987 SQL ステートメント画面, データベース・ チェックポイントの作成 596 SystemTempDir, 環境変数 667 SystemUtil.Run メソッド 862

Т

TestDirector 「Quality Center」を参照 TestDirector,「Quality Center」を参照 TestDir,環境変数 667 TestIteration,環境変数 667 TestName,環境変数 667 Test コマンド・ライン・オプション 987 TSL 関数の呼び出し, OuickTest からの 1384

U

Unicode 30 UnregisterUserFunc ステートメント 914, 924, 927 UntilDate コマンド・ライン・オプション 988 UpdatingActiveScreen, 環境変数 667 UpdatingCheckpoints, 環境変数 667 UserName, 環境変数 667 User コマンド・ライン・オプション 988

V

VALUE 関数, データ・テーブル・ワークシー トの 1145 VBScript 1314
関連付けられている関数ライブラリ Quality Center 905
構文 841
構文エラー 847
構文を自動的に拡張 885
テキストの書式設定 846
マニュアル 863
VBScript 構文を自動的に拡張する 885
Visual Basic 1314
Visual Studio.NET 1314
VuserId、環境変数 667

W

W3C O Web Content Accessibility Guidelines 32 WaitProperty ステートメント 809 Wait ステートメント 812 Web Web オブジェクトへのキーボード・コ マンドの送信 1415 Web Page の外観ダイアログ・ボックス 1179 Web コンテンツ・アクセシビリティ・チェッ クポイント テスト結果の1020 Web コンテンツ,動的な 1411 While ステートメント, エキスパート・ビュー および関数ライブラリにおける 868 Windows API 875 Windows コマンド・ライン・オプション 983 Windows ダイアログ・ボックス 1097 WinRunner QuickTest からのTSL 関数の呼び出し1384 OuickTest からのテストの呼び出し1380 関数の引数, QuickTest からのパラメー タの引き渡し1387 作業 1379 テスト, OuickTest からのパラメータの 引き渡し1382 テスト結果内の WinRunner ステップの 表示 992 WinRunner 関数の呼び出しダイアログ・ボッ クス1384 WinRunner テストの呼び出しダイアログ・ ボックス 1380

With ステートメント "With" ステートメント作成結果ウィン ドウ 801 エキスパート・ビュー 797 既存のアクションのために生成 800 削除 802 自動生成,記録中 799 手作業で入力 870 wscript.exe 1315

X XML

XML をテキストとして編集ダイアロ グ・ボックス 630 オブジェクトとメソッド 640 オブジェクト・リポジトリからエクス ポート 262 オブジェクト・リポジトリとしてイン ポート 261 出力値の結果 属性の詳細 1033 分析 1029 チェックポイント 609 Web ページ / フレーム 613 XPath 640 結果の分析 639,1010 スキーマ妥当性チェック・ダイアログ・ ボックス 635 スキーマの追加ダイアログ・ボックス 638 スキーマの編集ダイアログ・ボックス 638 テスト・オブジェクト用 620 名前空間 611, 640, 730 ファイル用 617 変更 639 チェックポイントの結果 属性の詳細 1014 チェックポイント・サマリ 1013 XML 構造 インポート 631.743 更新 631, 743 実行の更新モードを使った更新 631,743 XML 出力值 688 XML 出力値結果ウィンドウ 1030 XML出力のプロパティ・ダイアログ・ボックス738 XML チェックポイント結果ウィンドウ 1011

 XML チェックポイントのプロパティ・ダイア ログ・ボックス 625
 XML チェックポイント (ファイルから)・ダ イアログ・ボックス 617
 XML 値,出力 730
 XML をテキストとして編集ダイアログ・ボッ クス,XML チェックポイント 630

Z

Zip ファイル テストのインポート 349 テストのエクスポート 348

あ

アクション・パラメータ 499.642.651 オプションの設定652 アクション 435.465 欠落の呼び出し1114 値、「アクションの値,共有」*を参照* 値の共有458 Dictionary オブジェクトの使用 460 環境変数の使用 459 グローバル・データ・テーブルの使用 458 外部 468 概要 436, 466 キーワード・ビューへの追加 404 基本構文を使用した呼び出し461 欠落に対する呼び出しの削除1118 欠落に対する呼び出しの割り当て1115 構文 461 再利用可能な 468 再利用不可能な468 作業についてのガイドライン 479 削除 499 作成 477 ステップから実行943 挿入 既存 436 コピー 438 呼び出し441 ダイアグラム 436, 437, 466 テスト・フロー 475 テスト・フロー表示枠 471 テンプレート 503 名前の変更 496 ネスト 448, 492

パラメータ、「アクション・パラメー タ」を参照 パラメータ化データ,格納場所492 パラメータの構文 461 パラメータの設定486 複数, テストの467 プロパティ 473 プロパティの設定481 分割 494 戻り値の格納の構文 462 アクション間でのデータの引渡し469 アクション・タブ,データ・テーブル469 アクション・ツールバー、キーワード・ ビュー 68,475 アクション・データ・シート 469.1131 アクションの値, 共有 Dictionary オブジェクトの使用 460 環境変数の使用 459 グローバル・データ・テーブルの使用458 アクションの選択ダイアログ・ボックス439.442 アクションのネスト448.492 アクションの分割494 アクションの分割ダイアログ・ボックス 495 アクションの呼び出し 欠落 1111 実行プロパティ 454 パラメータ値 455 反復 454 プロパティ453 アクション・パラメータ 448 ガイドライン 451 出力値の格納 690, 699 アクション・リスト475 アクセス許可 Quality Center に必要 41 OuickTest の実行に必要 41 値 オブジェクトのプロパティの指定143 オブジェクトのプロパティ値の表示138 オブジェクト・プロパティの標準設定 の復元145,147 出力 685 設定 751 入力 416 パラメータ化 641 値カラム、キーワード・ビュー 400

値設定オプション・ダイアログ・ボックス 646.755 値セル 416 値の設定 751 値の設定領域 752 アドイン Ouality Center 内の OuickTest テストとの 関連付け 1339 テストへの関連付け1192 アドイン, OuickTest 31 アナログ記録 380.383 アプリケーションの管理, QuickTest との統合 1389 アプリケーション 実行 862 終了 862 ローカライズされたバージョンのテスト1419 アプリケーション,サンプル42 アプリケーションのクラッシュ・トリガ 1251 アプリケーションの静止画像、キャプチャと 表示 971 アプリケーション・プロセスの終了 862 アプリケーション領域 回復シナリオ、削除 1283 暗号化,パスワード418

V

一次リポジトリ 268 一次リポジトリ表示枠 274 一般オプション 882 移動ダイアログ・ボックス 831 色 オブジェクト・リポジトリ結合ツール での設定 285 オブジェクト・リポジトリ比較ツール での設定 320 キーワード・ビューでの設定 429 印刷 関数ライブラリ 903 テスト349 印刷ダイアログ・ボックス, テスト結果ウィ ンドウ 976 印刷プレビュー・ダイアログ・ボックス 977 インデックス識別子 「序数識別子」を参照 インデックス・プロパティ オブジェクトの識別に使用 208

インポート XML ファイルからオブジェクト・リポ ジトリ 261 zip ファイルからのテストのインポート 349

え

エージェント,リモート1361 エキスパート・ビュー 817,1409 アクションの戻り値の構文462 アクション・パラメータの構文461 アプリケーションの実行862 アプリケーションの終了862 一般カスタマイズ・オプション 882 エレメントの強調表示 885 概要 819 基本アクション構文 461 チェックポイント822 テキストの検索835 テキストの置換837 パラメータについて 823 見映えのカスタマイズ 881 エクスポート Screen Recorder ムービー 974 zipファイルへのテストのエクスポート348 オブジェクト・リポジトリから XML ファイルへ262 共有オブジェクト・リポジトリへの ローカル・オブジェクト 195 エディタ・オプション・ダイアログ・ボックス882 エラー, VBScript の構文 847 エラー時の動作,テストのオプション1196

お

オートメーション Application オブジェクト 1316 オブジェクト・モデル 1311 オブジェクト・リポジトリ 264 開発環境 1314 言語 1314 タイプ・ライブラリ 1314 定義 1312 テスト用スクリプトの生成 1192 オートメーション・ツールバー, QuickTest ウィンドウ 49,66 オブジェクト オブジェクト・リポジトリからの削除 181
識別 101 「テスト・オブジェクト」も参照 ドラッグ・アンド・ドロップ 59,1107 ナビゲートして学習の使用による追加246 認識 199 プロパティ,実行環境873 メソッド,実行環境 873 メソッドの表示 101 オブジェクト・スパイ 111 使い方のヒント115 オブジェクトの状態トリガ 1251 オブジェクトの選択画面 1255 オブジェクトの選択ダイアログ・ボックス 411 オブジェクトの認識 自動スクリプトの生成 213 標準設定の復元 213 オブジェクトの認識ダイアログ・ボックス 201 オブジェクトのプロパティ値 指定または変更 143 表示 138 標準設定の復元145.147 オブジェクトのプロパティと値の設定画面1257 オブジェクトの割り当てダイアログ・ボックス224 オブジェクト・フィルタの定義ダイアログ・ ボックス 172 オブジェクト・モデル オートメーション 1311 定義 1312 オブジェクト・リポジトリ XML からのインポート 261 XML へのエクスポート 262 アクションとの関連付け486 以前のバージョンから変換237 オートメーションの使用による管理264 オブジェクトの検索187 オブジェクトのコピー,貼り付け,お よび移動178 オブジェクトの削除181 オブジェクトの追加163 管理 228 関連付けの管理 191 共有 124 欠落 1111 作成 237 選択 122 閉じる 241

開く 237 変更 244 保存 239 未割り当て1122 ローカル 123 ローカル・オブジェクトのエクスポート195 オブジェクト・リポジトリ・ウィンドウ126 オブジェクトのフィルタ処理135 簡易表示と完全表示134 テスト・オブジェクトの詳細 136 オブジェクト・リポジトリ結合ツール 267 一次リポジトリ表示枠 274 色の設定 285 ウィンドウ270 解決方法のオプション表示枠 274 矛盾の解決方法の設定 282 ターゲット・リポジトリのフィルタ処 理 301 ターゲット・リポジトリ表示枠 272 二次リポジトリ表示枠 274 ビューの変更 272 矛盾 296 矛盾の解決 299 オブジェクト・リポジトリにオブジェクトを 追加ダイアログ・ボックス 166 オブジェクト・リポジトリの種類119 オブジェクト・リポジトリ比較ツール 309 色の設定 320 ウィンドウ312 相違点のタイプ 319 統計情報 324 リポジトリの同期化 327 リポジトリのフィルタ処理 325 リポジトリ表示枠 313 オブジェクト・リポジトリ・マネージャ 230 オブジェクト・リポジトリ・モード テストの設定 1199 オブジェクト・リポジトリ、ローカルからの エクスポートと置換195 オプション・ステップ 946 設定 947 標準設定 948 オプション・ダイアログ・ボックス 1162 ActiveScreen タブ 1172 一般タブ 1164 実行タブ 1181

スクリプトの生成オプション1164.1317 フォルダ・タブ1169 オンライン・ドキュメント xxiii オンライン・リソース xxv か 解決方法のオプション表示枠、オブジェク ト・リポジトリ結合ツール 274 ガイドライン ユーザ定義関数 931 開発環境 1314 外部アクション 定義 468 データの格納場所 492 外部関数、スクリプトからの実行 933 回復後のテスト実行オプション画面 1269 回復後のテスト実行のオプション1242 回復シナリオ 1241 欠落の削除 1125 欠落の特定 1123 コピー 1279 削除 1278 テストからの削除 1283 テストとの関連付け 1280 ファイル1245 プロパティの設定1283 プロパティの表示 1276, 1283 変更 1277 保存 1273 無効化 1284 回復シナリオ・ウィザード1249 オブジェクトの選択画面 1255 オブジェクトのプロパティと値の設定 画面 1257 回復後のテスト実行オプション画面 1269 回復シナリオ・ウィザードの完了画面1272 回復操作画面 1261, 1262 回復操作--関数の呼び出し画面 1267 回復操作ープロセスの終了画面 1266 回復操作-ボタン,またはキーを押す 画面 1264 関数の呼び出し画面 1267 テスト実行エラー画面 1258 トリガ・イベントの選択画面 1251 名前と記述画面 1271 プロセスの終了画面 1266

プロセスの選択画面 1259 ボタン、またはキーを押す画面 1264 ポップアップ・ウィンドウの条件を指 定画面 1253 回復シナリオ・ウィザードの完了画面 1272 回復シナリオ・マネージャ・ダイアログ・ ボックス 1245 回復操作1242 Microsoft Windows の再起動 1262 アプリケーション・プロセスの終了 1262 関数呼び出し 1262 キーボードまたはマウスの操作1262 学習、オブジェクト 246 各分野のエキスパート, Business Process Testing における役割 1368 仮想オブジェクト 1229 削除 1238 定義 1233 仮想オブジェクト・ウィザード 1234 仮想オブジェクト・マネージャ 1238 画像チェックポイント 画像内容の比較 544 プロパティの値の編集 544 画像チェックポイント・プロパティ・ダイア ログ・ボックス 543 画面レコーダタブ,テスト結果ウィンドウ972 画面レコーダのオプション・ダイアログ・ ボックス 1185 カラム、キーワード・ビューでの表示 427 簡易表示,オブジェクト・リポジトリ・ウィ ンドウ134 環境, テスト31 環境変数 661.1207 組み込み 666.1207 出力値の格納 702 タイプ 661 ファイル, Quality Center での 665 環境変数,ユーザ定義1211 エクスポート 1213 外部 662 内部 661 表示 1211 変更 1211 関数 コード,仕上げ922 コード、挿入 922

ユーザ定義 891 関数コードのプレビュー 921 関数定義ジェネレータ 913 概要 909 関数コードのプレビュー921 関数に説明を付ける 919 関数の定義 913 関数の登録 914 開く 911 関数に説明を付ける 919 関数の登録 914 関数の引数, QuickTest から WinRunner へのパ ラメータの引き渡し1387 関数呼び出し ドラッグ・アンド・ドロップ 59,1107 関数ライブラリ 891 一般オプション 882 一般オプションのカスタマイズ 882 エレメントの強調表示 885 概要 38 管理 893 関連付けの変更 908 関連付けられているものを使用した作業905 現在のものを関連付け 907 作業 1304 作成 894 実行の一時停止 1046 スクリプト・エディタで閉じる 1309 スクリプト・エディタでの作成 1306 スクリプト・エディタでのテストとの 関連付け 1306 スクリプト・エディタでの編集 1307 スクリプト・エディタでの保存 1308 スクリプト・エディタで開く 1304 説明 54 テキストの検索835 テキストの置換 837 テストの設定1199 デバッグ 902, 1037 ナビゲーション 899 開く 895,904 プロパティ 1296 編集 900 保存 897 見映えのカスタマイズ 881 読み取り専用,編集 902

完全表示,オブジェクト・リポジトリ・ウィ ンドウ 134 関連付け Quality Center で作成したテストとアド イン 1341 オブジェクト・リポジトリとアクショ ン 486 関数ライブラリ 905, 907, 908 共有オブジェクト・リポジトリ 191 テストへのアドイン 1192

き

キー・カラム 563,605 キーの割り当て エキスパート・ビューでの888 関数ライブラリでの 888 キーボード・コマンド, Web オブジェクトへ の送信 1415 キーボード・ショートカット エキスパート・ビューでの888 関数ライブラリでの 888 キーワード・ビューでの 426 キーワード駆動型のテスト OuickTest の設定 359 アプリケーションの分析 354 オートメーション・インフラストラク チャ354 オブジェクト・リポジトリの設定 356 概要 352 関数ライブラリの作成 358 テスト・ステップの作成 360 テストの作成 360 テストの実行 362 テストのトラブルシューティング 362 方法 354 キーワード・ビュー 52.395.397 カラム,説明398 カラム,表示 427 キーボードのキー 426 ステップ, 削除 425 ステップ,追加 404 ステップ、ブロックの後への追加 421 ステップ,変更 422 表示オプション 427 フォントと色 429

記述的プログラミング、「プログラム的記述」 を参照 記述, テスト・オブジェクト 105 「テスト・オブジェクト」を参照 規則,表記 xxvi 既存のアクション,挿入436 既存の出力値の追加ダイアログ・ボックス 747 既存のチェックポイントの追加ダイアログ・ ボックス 510 行範囲の変更ダイアログ・ボックス 724 共有オブジェクト・リポジトリ 119,124 アクションとの関連付け486 関連付けの管理 191 結合 267 比較 309 未割り当て1122 ローカル・リポジトリからの更新288 共有オブジェクト・リポジトリ・ウィンドウ235 共有オブジェクト・リポジトリの保存ダイア ログ・ボックス 305,306 許可 Quality Center に必要 41 OuickTest の実行に必要 41 記録 アナログ380 時間, 向上 1420 低レベル 380,1408 テスト 375 く クエリ・ファイル、データベース・チェック ポイントの

ハイントの ODBC/Microsoft Query を使った作業 1144 作成 1144 テスト 595 クッキー 1413 組み込み環境変数 666, 1207 グローバル・データ・シート 469, 1130 グローバル・データ・シートとアクション・ データ・シート, 選択 469 グローバル・データ・テーブル・パラメータ 659 グローバル・テスト・オプション 1161

け 計算

エキスパート・ビューでの 865

関数ライブラリでの865 結果削除ユーティリティ、コマンド・ライン からの実行 983 結果セット 592 結果の詳細タブ、テスト結果ウィンドウ956.972 結合 共有オブジェクト・リポジトリ 267 ローカル・オブジェクト・リポジトリ288 欠落アクションの特定ダイアログ・ボックス 1115, 1118 欠落リソース 1111 欠落リソース表示枠 62 説明 1112 フィルタ処理1113 未割り当ての共有オブジェクト・リポ ジトリ 1122 未割り当てのリポジトリ・パラメータ1126 言語 1314 言語サポート, Unicode 30 検索ダイアログ・ボックス エキスパート・ビュー 835 オブジェクト・リポジトリ結合ツール303 オブジェクト・リポジトリ比較ツール328 テスト結果 975 検索/置換ダイアログ・ボックス 183

2

構文 アクション・パラメータ 461 アクション461 アクションの戻り値462 構文エラー、VBScript 847 項目カラム,キーワード・ビュー 399 項目セル406 項目, 選択 アプリケーションからの 411 共有オブジェクト・リポジトリからの408 項目リストからの407 項目リスト 407 コマンド オブジェクト・リポジトリ結合ツール277 オブジェクト・リポジトリ比較ツール316 コマンド・ライン・オプション Domain 984 FromDate 984 Log 984

MinSize 985 Name 985 Password 986 Project 986 Recursive 986 Server 987 Silent 987 **Test 987** UntilDate 988 User 988 を使用したテスト結果の削除983 コメント エキスパート・ビュー864 関数ライブラリ 864 キーワード・ビュー806 コンポーネント 422 コメント・カラム、キーワード・ビュー 400 コレクション、仮想オブジェクト 1229 コレクション、プロパティの、「プログラム的 記述 | を参照 コンポーネント 実行結果,「実行結果」参照951 実行の一時停止 1046 ステップ,移動 423 ステップ,管理423 ステップ, 削除 425 ステップ,追加 404 デバッグ 1037 コンポーネントの実行 期待結果の更新 1076 実行ダイアログ・ボックス 939 Update Run ダイアログ・ボックス 1080 ステップから943 コンポーネント・パラメータ 416 コンポーネント・リソース, 欠落 1111

さ

サーバ
サーバ側の接続 1413
切断, Quality Center からの 1329
リダイレクト 1413
サーバのリダイレクト 1413
最初にお読みください xxiii
再利用可能なアクション 468
再利用不可能なアクション 468
サイレント・テスト・ランナー 1400

テストの実行 1401
開く 1401
サイレント・テスト・ランナー・ダイアロ グ・ボックス 1401
削除
アクション 499
オブジェクト・リポジトリのオブジェ クト 181
テスト結果 981
ブレークポイント 1049
リポジトリ・パラメータ 253
サマリ・カラム,キーワード・ビュー 401
サンプル・アプリケーション, Mercury Tours 42

ι

シート・メニュー・コマンド, データ・テー ブル1137 時間, ユーザ定義書式の設定 1140 実行オプション、オプション・ダイアログ・ ボックスの1181 実行環境 オブジェクト 873 実行結果 951 HTML にエクスポート 979 WinRunner ステップの表示 992 印刷 976 印刷前のプレビュー 977 検索 968, 975 コマンド・ライン・オプションによる 削除 983 削除, テスト結果削除ツール 981 実行時データ・テーブル 1028 出力值 1027 手動による不具合の報告 989 スキーマ 995 チェックポイント 1002 テスト結果ウィンドウ953 特定の実行の表示 961 パラメータ化された値 1025 表示のカスタマイズ 995 フィルタ処理967 不具合の自動報告 990 メッセージの送信 803 有効化とフィルタ処理879 テスト結果,「実行結果」参照 結果,「実行結果」参照951

実行結果の分析、「実行結果」参照951 実行結果の分析「実行結果」 参照 実行時 設定,追加と削除1225 データ・テーブル 1028.1128 実行セッション 一時停止 1046 回復シナリオの無効化 1284 結果の印刷 976 テスト・オブジェクト・プロパティの 変更 190 テスト・オブジェクトを使った作業 190 プログラミングによるテスト・オブ ジェクトの作成190 実行セッションの一時停止 1046 実行セッションのムービー Quality Center での結果の表示 970 エクスポート 974 キャプチャと表示 972 テスト結果からの削除 973 キャプチャのオプション設定1181 Mercury 画面レコーダ、「実行セッションの ムービー | 参照972 実行ダイアログ・ボックス 939 Update Run ダイアログ・ボックス 1080 実行プロパティ、アクションの呼び出しの設 定 454 自動化エンジニア, Business Process Testing に おける役割 1369 シナリオ「回復シナリオ」を参照 終了,アプリケーション・プロセス 1262,1266 出力 XML 值 730 値 685 データベース値 726 テキスト値703,705 プロパティ値 692 出力タイプ 698 アクション・パラメータ 699 環境変数 702 テスト・パラメータ 699 データ・テーブル700 出力値 XML 730, 738 アクション・パラメータまたはテス ト・パラメータへの格納 690

オブジェクトのプロパティ 695 環境変数への格納 691 既存の追加 747 結果の表示 1027 定義 685 データ・テーブルへの格納 690 データベース 726, 728, 730 テーブル 712,716,721 テキスト 703,706 テキストの作成 705 テキスト領域 705 表示 691 標準 692 編集 691 出力値カテゴリ XML 出力值 688 データベース出力値 688 テキスト出力値 687 テキスト領域出力値 687 標準出力値 687 出力パラメータのプロパティ・ダイアログ・ ボックス 695 手動ステップ 422 手動テスト 431 使用可能なキーワード表示枠 59.1107 条件ステートメント 788 キーワード・ビューでの使用 433 仕様、データ・テーブル 1134 ショートカット・キー OuickTest 69 キーワード・ビューでの 426 情報表示枠 48,58 ショートカット OuickTest 69 エキスパート・ビューでの888 オブジェクト・リポジトリ結合ツール277 オブジェクト・リポジトリ比較ツール316 関数ライブラリでの888 メニュー項目の69 初期化スクリプト1313 書式メニュー・コマンド、データ・テーブル1140 序数識別子 207 テスト・オブジェクトの指定157 新規アクションの呼び出しを挿入ダイアロ グ・ボックス 477 新規結合ダイアログ・ボックス 286

す

数式 値をパラメータ化するための1145 チェックポイントでの1146 データ・テーブルでの1145 数式、データ・テーブルでの1145 スキーマ、実行結果の 995 スキーマ妥当性チェック・ダイアログ・ボッ クス, XML チェックポイント 635 スキーマの追加ダイアログ・ボックス, XML チェックポイント 638 スキーマの編集ダイアログ・ボックス、XML チェックポイント 638 スクリプト・エディタ 1287 ウィンドウのカスタマイズ 1290 関数ライブラリ 1304 テスト 1300 テスト表示枠 1292 表示領域 1298 メイン・ウィンドウ 1289 リソース表示枠 1295 スクリプト,テスト,「テスト」*を参照* スクリプトの生成オプション 1317 スタート・ページ55 ステータス・バー OuickTest ウィンドウ 50 オブジェクト・リポジトリ結合ツール275 オブジェクト・リポジトリ比較ツール315 ステートメント、キーワード・ビューでの使用422 ステートメントの自動補完 824.884 ステップ 移動 423 オプション946 キーワード・ビューからの削除425 キーワード・ビューでの表示プロパ ティ432 キーワード・ビューでの変更 422 キーワード・ビューへの追加 404 コンポーネントの管理 423 削除 425 手動 422 挿入 769 追加 404 ブロックの後への追加 421 ステップからデバッグ 1043 ステップ・コマンド 1040

ステップ・ジェネレータ 768,769 ステップ・ジェネレータ・ダイアログ・ボッ クス 773 ステップでオブジェクトを選択ダイアログ・ ボックス 408 ステップの移動 423 ステップまで実行1043 スナップショット ActiveScreen のキャプチャ設定 1175 テスト結果ウィンドウ952 スパイ 「オブジェクト・スパイ」 を参照 スマート認識 オブジェクトの認識ダイアログ・ボッ クスからの有効化 212, 213 情報の分析 999 設定 214 テストの実行中に無効化する 1197 スマート認識プロパティ・ダイアログ・ボッ クス219

せ

正規表現 757 エキスパート・ビューおよび関数ライ ブラリでの使用 840 関数ライブラリでの使用 840 チェックポイントでの使用 758 定義 759 定数 752 バックスラッシュ(\)761 プロパティ値758 製品情報ウィンドウ94 製品情報ボタン94 設計時データ・テーブル 1127 セッション ID 1413 接続文字列. データベース・チェックポイン トの指定 596 絶対パス 340 設定 473 設定タブ、データベース・チェックポイントの プロパティ・ダイアログ・ボックス 602 セルの識別タブ、データベース・チェックポ イントのプロパティ・ダイアログ・ ボックス 604

そ

相違点のタイプ オブジェクト・リポジトリ比較ツール319 操作 項目リストからの選択 406, 407 ステップ用に選択 415 引数 416 操作カラム,キーワード・ビュー 400 操作セル 415 相対パス 340 挿入ツールバー,QuickTest ウィンドウ 67 ソフトウェア・アップデート 43

た

ターゲット・リポジトリ 268 保存 304
ターゲット・リポジトリ表示枠 272
タイトル・バー, QuickTest ウィンドウ 50
タイプ・ライブラリ 1314
タイムアウト 設定 1197
テキスト・チェックポイントの指定 585 標準チェックポイントの指定 541
他の HP 製品でテストおよびコンポーネントを 実行可能にするオプション 1359

ち

チェック・アウト・コマンド1349 チェック・イン・コマンド 1349.1352 チェックポイント XML 609 エキスパート・ビューでの822 オブジェクト 536 画像 543 既存の追加 510 失敗 1057 種類 513 新規追加 508 数式の使用 1146 説明 507 定義 338, 507 データベース 591 テーブル 547, 548, 552 テキスト 569.571 テキスト領域 572 パラメータ化 675

ビットマップ 519 標準, テキストの検査 587 変更 543,545 チェックポイントのプロパティ・ダイアロ グ・ボックス オブジェクトの検査 538 データベースの検査 552 置換ダイアログ・ボックス エキスパート・ビュー 837 関数ライブラリ 837

っ

通貨、ユーザ定義書式の設定 1140 ツール・ツールバー, OuickTest ウィンドウ 67 ツールバー OuickTest ウィンドウ アクション68 オートメーション 66 挿入 67 ツール67 テスト49 デバッグ 48.67 表示 68 標準66 ファイル50 編集 67 オブジェクト・リポジトリ結合ツール276 オブジェクト・リポジトリ比較ツール316 ツリー・ビュー、「キーワード・ビュー」を参照

て

定数値オプション・ダイアログ・ボックス 754 定数値オプション・ボタン 754 定数値, 定義 751 ディスク容量, 保存 1420 低レベル記録 380, 386, 1408 データ駆動テスト 642, 690 データ・シート アクション 1131 グローバル 1130 グローバルとアクション, 選択 469 ローカル 1131 データ・テーブル 50, 64, 1127 Quality Center での使用 1141 アクション・タブ 469 アクション・データ・シート 1131

グローバル・タブ469 結果の表示 1028 個別テストの反復オプション1196 シート・メニュー・コマンド1137 実行時 1128 出力値の格納 690 仕様 1134 書式メニュー・コマンド1140 数式の使用 1145 スクリプティング関数,使用1149 設計時 1127 データ・シート 1130 データのインポート、さまざまな形式 での1132 データ・メニュー・コマンド1139 テーブルの編集1132 場所 1131 ファイル・メニュー・コマンド1136 編集メニュー・コマンド1137 保存1131 メニュー・コマンドの使用 1135 ローカル・データ・シート1131 ワークシートの関数1145 データ・テーブルのワークシートの関数 1145 データ・テーブル・ワークシートの TEXT 関 数 1145 データ・ドライバ 678 データベース Microsoft Ouery/SOL ステートメントに よるクエリの作成 595 ODBC/Microsoft Query でのクエリの作 成 1144 SOL ステートメント画面 596 結果セット 592 接続文字列 596 チェックポイントの作成 592 手作業での SQL ステートメントの定義 593 データベース・クエリー・ウィザード 593 データベース出力値 688,726,728 データベース出力値のプロパティ・ダイアロ グ・ボックス 728 データベース・チェックポイント 591 値のタイプの指定602 一般情報 599 期待データの指定 601

結果の分析 1005 説明 591 セルの識別の設定の指定 604 セルの指定 599 変更 606 データベース・チェックポイントのプロパ ティ・ダイアログ・ボックス 597 期待データ・タブ 601 設定タブ 602 セルの識別タブ 604 データベースでの内容のプロパティの検査 592 データ・メニュー・コマンド、データ・テー ブル1139 テーブル出力値716 行の範囲の変更 724 出力オプションの変更 724 テーブルの内容タブ 718 テーブルのプロパティ・タブ718 テーブル出力値のプロパティ・ダイアログ・ ボックス716 テーブルの内容タブ 716 プロパティ・タブ 722 テーブル・チェックポイント Table Properties タブ 556 値のタイプの指定561 一般オプション 555 期待データの指定 560 結果の分析 1005 作成 548 説明 547 セル認識の設定の指定 562 セルの指定 558 テーブルの内容タブ 556 変更 566 テーブル・チェックポイントのプロパティ・ ダイアログ・ボックス 552 期待データ・タブ 560 テーブルの内容タブ 554 プロパティ・タブ 564 テーブルの内容タブ テーブル出力値のプロパティ・ダイア ログ・ボックス 716 テーブル・チェックポイントのプロパ ティ・ダイアログ・ボックス 554 テーブル・プロパティ 検査対象の指定565

出力するプロパティの選択 723 テキスト,検査 テキスト領域チェックポイントの使用572 テキスト出力値 687 作成 703 指定 706 テキスト出力パラメータのプロパティ・ダイ アログ・ボックス 706 テキスト選択の設定ダイアログ・ボックス 578 テキスト・チェックポイント 569.571 後のテキストの指定584 オプションの設定 577 結果の分析 1008 検査対象テキストの指定581 種類 569 タイムアウトの指定585 テキスト選択の設定 578 標準チェックポイント 587 変更 586 前のテキストの指定582 テキスト・チェックポイントのプロパティ・ ダイアログ・ボックス 575 テキスト値,出力 703,705 テキスト領域チェックポイントのプロパ ティ・ダイアログ・ボックス 575 テキスト領域出力値 687 作成 705 テキスト領域出力値のプロパティ・ダイアロ グ・ボックス706 テスト Ouality Center プロジェクトへの保存 1333 Quality Center での管理 39 Ouality Center でのテンプレート・テス トを使用した作成 1342 Ouality Center プロジェクトで開く 1334 OuickTest での起動 344 圧縮 348 印刷 349 解凍 349 回復シナリオの関連付け 1280 回復シナリオの削除 1283 回復シナリオの無効化 1284 拡張 338 管理 344 旧バージョンのテストを開く 345 記録 373.375

更新 1076 コンポーネントとの比較1376 サイレント・テスト・ランナーを使用 した実行 1401 作業 1300 作成 333, 344, 351 実行 937 実行、オプション・ステップの使用946 「実行結果」 参照 979 実行の一時停止 1046 スクリプト・エディタで閉じる 1303 スクリプト・エディタでの編集 1302 スクリプト・エディタでの保存 1303 スクリプト・エディタで開く 1300 ステップからの実行943 設定 473 ダイアグラム 436, 437, 466 チェックポイント「チェックポイン ト」を参照。547 テスト・ステップについて 337 デバッグ 1037 バージョン・コントロールからの チェック・アウト1349 バージョン・コントロールへのチェッ ク・イン1352 バージョン・コントロールへの追加1349 パラメータ化,例 673 プロパティ 1293.1296 保守 1057 保存346 ローカル468 テスト・オブジェクト アプリケーション内での強調表示 186 オブジェクト・リポジトリ内でのコ ピー. 貼り付け. および移動 178 オブジェクト・リポジトリ内の検索182,187 管理 117 記述プロパティの削除 156 検索 183 識別 101 実行セッションでの190 実行セッションでの作成190 序数識別子の指定157 新規の定義 176 新規プロパティの定義154 選択

アプリケーションからの 411 共有オブジェクト・リポジトリからの408 項目リストからの407 追加 オブジェクト・リポジトリへの163 記述プロパティ 150 ドラッグ・アンド・ドロップ 126,245 名前の変更 147 プログラム的記述を使った作成190 プロパティ値, 置換183 プロパティ値の取得と設定872 プロパティの表示138 変更 実行セッション中のプロパティ 190 実行セッションでの 190 名前 147 プロパティ 136,140 ローカル・リポジトリへのコピー137 テスト・オブジェクトの選択 アプリケーションからの411 共有オブジェクト・リポジトリからの408 項目リストからの 407 テスト・オブジェクトのプロパティ 101 テスト・オプション 個別のテストの設定1187 実行時 1225 取得 1224 すべてのテストの設定1161 設定 1222 テスト実行中 1221 復元 1225 テスト結果ウィンドウ953 Screen Recorder タブおよび Result Details タブ 972 結果の詳細タブ 956 実行結果ツールバー 958 実行結果ツリー 955 テーマ 959 ルック・アンド・フィール959 テスト結果削除ツール 981 テスト結果ツールバー、テスト結果ウィンドウ958 テスト結果ツリー 955 テスト実行エラー画面 1258 テスト実行エラー・トリガ 1251 テスト実行時間,向上1420 テスト実行ログ 1403

テスト・セット1347 テスト・データベースの維持1313 テストの圧縮348 テストの解凍 349 テストの実行937 Ouality Center プロジェクトからの 1346 WinRunner テストの実行 1380 オプション・ステップの使用 946 期待結果の更新 1076 結果の表示 960 高度な問題 1407 サイレント・テスト・ランナーの使用1401 実行ダイアログ・ボックス 939 Update Run ダイアログ・ボックス 1080 ステップから943 テストの設定ダイアログ・ボックス 1189 回復タブ 1215 環境タブ 1207 実行タブ 1195 パラメータ・タブ 1204 プロパティ・タブ 1191 リソース・タブ 1199 テストの同期化808 同期化ポイント 809 特定のプロパティ値の待機809 テストの複数のアクション 467 テストの保守 1057 テスト・バッチ,実行948 テスト・パラメータ 642,651 オプションの設定652 出力値の格納 690.699 ステップでの使用 1207 テスト表示枠 スクリプト・エディタ 1292 テスト・フロー (アクション) 475 テスト・フロー表示枠 60 アクション 60,471 テスト・プロセス 32 テスト結果の分析37 テストの作成 33 テストの実行 36 テスト・リソース、欠落 1111 デバッグ 関数ライブラリ 902,1037 コンポーネント 1037 実行の一時停止 1046

ステップからデバッグ 1043 ステップまで実行1043 テスト 1037 テスト,例1054 ブレークポイント 削除 1049 設定 1047 無効化と有効化 1048 デバッグ・ツールバー, QuickTest ウィンドウ48,67 デバッグ・ビューア 64,1049 テンプレート,アクション503 テンプレート・テスト 1339.1341 データ・テーブル 出力値の格納 700 テーブル・カラム 656 テーブルの行656 パラメータ,オプションの設定657

と

同期化タイムアウト 設定 1197 同期化, テスト オブジェクトの表示を待機 812 タイムアウト値の変更 813 同期化ポイント 作成 809 挿入 809 同期化ポイントの追加ダイアログ・ボックス809 統計情報ダイアログ・ボックス 295 比較ツール 324 動的な Web コンテンツ 1411 動的に生成される URL や Web ページ 1412 ドキュメント,オンライン xxiii ドッキング表示枠 1093 トランザクション 1395 終了 1399 挿入 1398 測定 1395 定義 1395 トランザクション開始ダイアログ・ボックス1398 トランザクション終了ダイアログ・ボックス 1399 トランザクション終了ボタン 67 トリガ アプリケーションのクラッシュ 1251 イベント 1242 オブジェクト状態 1251

テスト実行エラー 1251 ポップアップ・ウィンドウ 1251 トリガ・イベントの選択画面 1251 ドキュメントのみのオプション 431

な

ナビゲートして学習オプション 246 名前 テスト・オブジェクトの変更 147 名前と記述画面 1271 ナレッジ・ベース xxv

に

二次オブジェクト・リポジトリ 268 二次リポジトリ表示枠 274 認証 Quality Center への接続 1323

は

バージョン管理1348 バージョン・コントロール 1348 テストのチェック・アウト 1349 テストのチェック・イン1352 テストの追加 1349 パーセント,ユーザ定義書式の設定 1140 場所識別子 「序数識別子」 を参照 場所プロパティ、オブジェクトの識別に使用 209 パス,絶対と相対340 パスワード, 暗号化 418 パスワード・エンコーダ・ダイアログ・ボッ クス 418 バックスラッシュ(\)761 パフォーマンス,向上1420 パフォーマンス・テスト製品, QuickTest との 統合 1389 パラメータ WinRunner 関数への引渡し1387 WinRunner テストへの引渡し1382 アクション 448, 499, 653 アクションのガイドライン 451 アクションの設定486 アクションの呼び出しの構文461 エキスパート・ビューでの823 親アクション 653 環境変数. ユーザ定義 1209.1211 先行するアクション呼び出しの出力653

テスト 653 テストの指定1204 未割り当てのオブジェクト・リポジト リの処理 1126 リポジトリ248 管理 249 欠落 1111 削除 253 追加 251 変更 252 割り当て158 パラメータ・オプション・ダイアログ・ボッ クス 652 パラメータ・オプション・ボタン 753 パラメータ化 値 641 データ・ドライバの使用 678 テスト,例673 メソッド643 リポジトリ・パラメータを使用するプ ロパティ値255 パラメータ化アイコン 646.648.755 パラメータ化された値、テスト結果の表示1025 パラメータ化の例 673 パラメータ・タイプ アクション・パラメータ 642 環境変数パラメータ 661 出力値の格納 691 データ・テーブル・パラメータ 655 テスト・パラメータ 642 乱数パラメータ 671 パラメータ値 アクションの呼び出し455 定義 751 パラメータの定義,標準設定 753,756 反復 454, 655 個別テストのオプション1196

ひ

比較 共有オブジェクト・リポジトリ 309 引数,定義 913 ビジネス・アナリスト Business Process Testing における役割 1368 ビジネス・コンポーネント、概要 40

ビジネス・コンポーネントの設定ダイアロ グ・ボックス スクリプトの生成オプション 1317 ビジネス・プロセス・テスト1372 概要 40 実行 1376 日付,ユーザ定義書式の設定1140 必須プロパティ,設定202 ビットマップ・チェックポイント の結果の分析 1007 ビットマップ・チェックポイント 519 作成 520 修正 530 ビットマップ・チェックポイントのプロパ ティ・ダイアログ・ボックス 520 表記規則 xxvi 表現、エキスパート・ビューおよび関数ライ ブラリでの使用 840 表示ツールバー 68 標準出力値 687 作成 692 指定 695 標準設定のオブジェクトの認識設定 213 標準設定のオプション・ステップ 948 標準設定のパラメータの定義 753,756 標準設定のプロパティ,変更101,117 標準チェックポイント 結果の分析 1004 タイムアウトの指定 541 標準ツールバー, OuickTest ウィンドウ 66 表示領域 スクリプト・エディタ 1298 表示枠 移動 1088 欠落リソース 62 自動非表示 1093 情報 58 デバッグ・ビューア 64 ドッキング 1093 フローティング 1094 レイアウトのカスタマイズ1088

ふ

ファイル・ツールバー, QuickTest ウィンドウ 50 ファイル・メニュー・コマンド, データ・ テーブル 1136

フィードバックの送信 xxv フィルタ オブジェクトの定義172 フィルタ処理 オブジェクト・リポジトリ・ウィンド ウのオブジェクト135 オブジェクト・リポジトリ比較ツール でのリポジトリ 325 ターゲット・リポジトリ 301 フィルタ・ダイアログ・ボックス オブジェクト・リポジトリ結合ツール302 オブジェクト・リポジトリ比較ツール325 フィルタ・プロパティ (スマート認識) 214 フォント,キーワード・ビューでの設定429 不具合の報告 自動 989 手動 989 不具合, 報告 989 自動化, テスト中 990 テスト結果からの989 復元, OuickTest の標準設定のレイアウト 1164 複雑な値 754 複数のドキュメントを使った作業 1097 ブックマーク 832 ブレークポイント 概要 1046 キーワード・ビューでの使用 433 削除1049 設定 1047 無効化と有効化 1048 フローティング表示枠 1094 プログラミング 1409 VBScript 841 エキスパート・ビューおよび関数ライ ブラリ 817 関数ライブラリ 817 コメント806 実行セッション中のメッセージの表示805 条件ステートメント 788 ステップ・ジェネレータ 768.769 テスト結果へのメッセージの送信803 メッセージの生成803 ループ・ステートメント 794 プログラム的記述 190.850 Index プロパティ 858 With ステートメント 854

オブジェクトに対するチェックの実行859 記述オブジェクト 855 ステートメント 851 変数 851 プロジェクト (Quality Center) 接続 1323 切断 1329 テストの保存1333 テストを開く1334 プロセス・ガイダンス 1155 開始 1154 プロセス・ガイダンス表示枠 63,1152 プロセスの選択画面 1259 プロパティ 473, 1293, 1296 CreationTime 210 アクションの設定481 アクションの呼び出しの設定453 インデックス 208 回復シナリオの表示 1276.1283 キーワード・ビューのステップの表示432 実行環境オブジェクト 873 テスト・オブジェクト記述からの削除156 テスト・オブジェクト記述の追加150 テスト・オブジェクトの新規プロパ ティの定義154 テスト・オブジェクトの変更140 場所 209 標準101 標準設定 117 プロパティ・コレクション、「プログラム的記 述」を参照 プロパティ・タブ テーブル出力値のプロパティ・ダイア ログ・ボックス 722 テーブル・チェックポイントのプロパ ティ・ダイアログ・ボックス 564 プロパティ値 テスト・オブジェクト記述での指定255 同期化ポイント 809 プロパティの追加と削除ダイアログ・ボック ス150 プロパティの追加と削除ダイアログ・ボック ス,オブジェクトの認識 203,219

\sim

米国リハビリテーション法の第 508 条,「Web Content Accessibility Guidelines」32
変更 ライセンス 42
編集ツールバー, QuickTest ウィンドウ 67
編集メニュー・コマンド,データ・テーブル 1137
変数 環境 1207
「環境変数,ユーザ定義」も参照 グローバル・スコープ内で一意 931

ほ

補足プロパティ,設定 202 ポップアップ・ウィンドウ・トリガ 1251 ポップアップ・ウィンドウの条件を指定画面 1253

む

 矛盾の解決方法 結合したオブジェクト・リポジトリ 299
 矛盾の解決,オブジェクト・リポジトリ結合 ツール 299
 矛盾の解決方法 設定,オブジェクト・リポジトリ結合 ツール 282

め

メソッド
実行環境オブジェクト 873
新規の追加または振る舞いの変更 924
テスト・オブジェクトの表示 101
ユーザ定義 924
メソッドの登録 924
メソッドの登録解除, UnregisterUserFunc ス テートメントの使用 929
メッセージ
実行セッション中の表示 805
生成 803
テスト結果への送信 803
メニュー・バー, Quick Test ウィンドウ 48
メンテスナンス実行モード 1060

ъ

文字セット・サポート, Unicode 30

ゆ

ユーザ定義 関数,「ユーザ定義関数」 を参照 テスト・オブジェクト,割り当て224 プロパティへのアクセス874 メソッド 924 ユーザ定義オブジェクト,割り当て224 ユーザ定義関数 891 ガイドライン 931 関数定義ジェネレータ 909 関数定義ジェネレータでのコードのプ レビュー921 仕上げ 922 説明を付ける 919 追加の生成 921 ツールチップの追加 919 登録 914 ユーザ定義の外部環境変数 662 ユーザ定義の数値の書式,設定1140 ユーザ定義の内部環境変数 661 優先順位 回復シナリオの設定 1283

よ

予約済みオブジェクト905

6

ライセンス情報 42 乱数パラメータ 671

ŋ

リソース,管理 61
リソース,コンポーネント内で欠落 1111
リソース,テスト内で欠落 1111
リソースの管理 1101
リソース表示枠 61,1101,1295
リポジトリ「オブジェクト・リポジトリ」を参照
リポジトリ内を検索ボタン 534,542,557,698,711,719,730,742,577,606,630
リポジトリの関連付けダイアログ・ボックス 191
リポジトリの関連付けダイアログ・ボックス 191
リポジトリの同期化 オブジェクト・リポジトリ比較ツール 327
リポジトリ・パラメータ 248 値のパラメータ化 255

管理 249 削除 253 追加 251 変更 252 割り当て158 リポジトリ・パラメータ・ダイアログ・ボッ クス 255 リポジトリ・パラメータの管理ダイアログ・ ボックス 249 リポジトリ・パラメータの追加ダイアログ・ ボックス 251 リモート・エージェント 1361

る

ループ・ステート	メント 794
キーワード・	ビューでの使用 433

n
レイアウト
QuickTest ウィンドウのカスタマイズ 1087
タブの移動 1088
標準設定の復元 1097
表示枠の移動 1088
レポート,「テスト結果ウィンドウ」を参照
レポートの挿入ダイアログ・ボックス 803
レポート,フィルタの 879

ろ

```
ローカライズ 661,1131
ローカライズされたアプリケーション, テス
   ト 1419
ローカル・オブジェクトのエクスポートと置
   換 195
ローカル・オブジェクト・リポジトリ 119,123
  結合 288
  へのオブジェクトのコピー137
ローカル・オブジェクト・リポジトリ, エク
   スポートと置換 195
ローカル・データ・シート 「アクション・
   データ・シート」を参照
ローカル・データ・テーブル・パラメータ 660
ローカル・テスト468
ローカル・パラメータ 416
ローカル・オブジェクト, 共有オブジェク
   ト・リポジトリへのエクスポート 195
```

わ

割り当て 欠落アクション 1114 欠落アクションの呼び出し1115 未割り当てのオブジェクト・リポジト リ 1122 未割り当てのリポジトリ・パラメータ 1126 ユーザ定義オブジェクト 224 リポジトリ・パラメータ 158 割り当てカラム,キーワード・ビュー400