

OPTIMIZE

MERCURY QUICKTEST PROFESSIONAL™

FOR BUSINESS PROCESS TESTING

VERSION 9.0

ユ ー ザ ー ズ ・ ガ イ ド

MERCURY™

BUSINESS TECHNOLOGY OPTIMIZATION

Mercury QuickTest Professional for Business Process Testing

ユーザーズ・ガイド

Version 9.0

Mercury QuickTest Professional for Business Process Testing ユーザーズ・ガイド, Version 9.0

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

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

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

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

© 1992 - 2006 Mercury Interactive Corporation, All rights reserved

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

目次

ようこそ	xi
本書の使用法	xi
製品マニュアル	xiii
追加オンライン・リソース	xv
文書の更新	xvi
本書の表記規則	xvii

第 1 部 : Business Process Testing の紹介

第 1 章 QuickTest Professional for Business Process Testing の使用	3
QuickTest Professional for Business Process Testing の使用法について	4
Business Process Testing について	5
QuickTest Professional for Business Process Testing の用語について ..	11
必要なアクセス許可と権限の設定	13
サンプル・サイトの使用	14
ライセンス情報の変更	15
QuickTest ソフトウェアの更新	16
第 2 章 QuickTest の概要	17
QuickTest の開始	18
Quality Center プロジェクトへの接続	20
QuickTest のウィンドウ	25
キーワード・ビュー	28
アプリケーション領域	36
関数ライブラリ	38
情報表示枠	39
欠落リソース表示枠	39
デバッグ ビューア表示枠	40
QuickTest ウィンドウのレイアウトのカスタマイズ	41
複数のドキュメントを使った作業	50
QuickTest コマンドの使用方法	52
QuickTest Professional プログラム・フォルダの参照	64
製品情報の表示	68

第 2 部 : アプリケーション領域とコンポーネントを使用した作業

第 3 章 アプリケーション領域の使用	75
アプリケーション領域の使用について.....	76
アプリケーション領域の作成.....	79
アプリケーション領域を開く.....	81
一般設定の定義.....	83
関数ライブラリの管理.....	89
共有オブジェクト・リポジトリの管理.....	94
キーワードの管理.....	101
追加設定の定義.....	105
アプリケーション領域の保存.....	116
アプリケーション領域の削除.....	118
第 4 章 ビジネス・コンポーネントの使用	121
ビジネス・コンポーネントの使用について.....	121
新規ビジネス・コンポーネントの作成.....	123
ビジネス・コンポーネントの保存.....	126
ビジネス・コンポーネントを開く.....	130
手動コンポーネントを使った作業.....	134
コンポーネントに関連付けられているアプリケーション領域の変更... ..	138
コンポーネントの印刷.....	140
第 5 章 スクリプト・コンポーネントを使った作業	141
スクリプト・コンポーネントにかかわる作業について.....	142
スクリプト・コンポーネントの作成.....	144
スクリプト・コンポーネントへの変換.....	148
第 6 章 ビジネス・コンポーネント・キーワード・ビューを 使った作業	151
ビジネス・コンポーネント・キーワード・ビューを使った作業 について.....	152
コンポーネントへのステップの追加.....	153
ステップの項目の選択.....	156
ステップの操作の選択.....	164
ステップの引数の値の定義.....	166
ステップの出力値の定義.....	169
パラメータの使用.....	172
コメントの使用.....	180
コンポーネントのステップの管理.....	182
第 7 章 欠落リソースの処理	183
欠落リソースの処理について.....	184
欠落している共有オブジェクト・リポジトリの処理.....	186
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... ..	187

第 3 部 : 設定

第 8 章 グローバル・テスト・オプションの設定	191
グローバル・テスト・オプションの設定について	191
[オプション] ダイアログ・ボックスの使用	192
テストの一般オプションの設定	194
テストのフォルダ・オプションの設定	196
テストの実行オプションの設定	199
テストの Windows アプリケーション・オプションの設定	201
テストの Web オプションの設定	212
第 9 章 ビジネス・コンポーネント設定にかかわる作業	227
ビジネス・コンポーネント設定にかかわる作業について	228
[ビジネス コンポーネントの設定] ダイアログ・ボックスへの アクセス	228
コンポーネント・プロパティにかかわる作業	231
コンポーネントのスナップショットの定義	234
アプリケーション設定の表示	235
コンポーネント・リソースの表示	238
コンポーネントのパラメータの定義	239
Web 設定の表示	243
回復シナリオの設定の表示	245
第 10 章 関数ライブラリ・ウィンドウのカスタマイズ	247
関数ライブラリ・ウィンドウのカスタマイズについて	248
エディタの動作のカスタマイズ	248
エレメントの見映えのカスタマイズ	252
編集コマンドのカスタマイズ	253
第 11 章 Web イベント記録の設定	257
Web イベント記録の設定について	258
標準で使用するイベント記録設定の選択	259
イベント記録設定のカスタマイズ	261
マウスの右ボタン・クリックの記録	271
ユーザ定義イベント設定ファイルの保存と読み込み	275
イベント記録設定のリセット	277

第 4 部 : テスト・オブジェクトおよびオブジェクト・リポジトリを使用した作業

第 12 章 テスト・オブジェクト・モデルについて	281
テスト・オブジェクト・モデルについて.....	281
テスト・オブジェクト・モデルの概念の適用.....	285
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の表示...290	
オブジェクト・スパイを使用したオブジェクト・メソッドと メソッド構文の表示.....	294
第 13 章 テスト・オブジェクトを使用した作業	299
テスト・オブジェクトを使用した作業について.....	300
オブジェクト・リポジトリの種類について.....	301
[オブジェクトリポジトリ] ウィンドウについて.....	306
テスト・オブジェクトのプロパティの表示と変更.....	314
リポジトリ・パラメータ値の割り当て.....	338
オブジェクト・リポジトリへのオブジェクトの追加.....	343
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.....	351
オブジェクト・リポジトリからのオブジェクトの削除.....	354
オブジェクトの場所の特定.....	355
実行セッション中のテスト・オブジェクトの操作.....	362
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.....	364
第 14 章 オブジェクトの認識の設定	367
オブジェクトの認識の設定について.....	368
[オブジェクトの認識] ダイアログ・ボックス.....	369
スマート認識の設定.....	382
ユーザ定義のテスト・オブジェクト・クラスの割り当て.....	391
第 15 章 オブジェクト・リポジトリの管理	395
オブジェクト・リポジトリの管理について.....	396
オブジェクト・リポジトリ・マネージャについて.....	398
オブジェクト・リポジトリを使った作業.....	405
オブジェクト・リポジトリの変更.....	410
リポジトリ・パラメータを使用した作業.....	413
テスト・オブジェクトの詳細の変更.....	419
オブジェクトの検索.....	422
結合操作の実行.....	423
インポートおよびエクスポート操作の実行.....	423

第 16 章 共有オブジェクト・リポジトリの結合	427
共有オブジェクト・リポジトリの結合について	428
オブジェクト・リポジトリ結合ツールについて	429
オブジェクト・リポジトリ結合ツールのコマンドの使用法	435
標準設定の定義	436
2つのオブジェクト・リポジトリの結合	441
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	443
結合の統計情報の表示	449
オブジェクトの矛盾について	450
オブジェクトの矛盾の解決	453
ターゲット・リポジトリ表示枠に対するフィルタの設定	455
オブジェクト・リポジトリ・ビューの同期	456
特定のオブジェクトの検索	457
ターゲット・リポジトリの保存	459
第 5 部 : コンポーネントと実行のデバッグ	
第 17 章 コンポーネントと関数ライブラリのデバッグ	465
コンポーネントと関数ライブラリのデバッグについて	466
デバッグ・セッションの低速化	469
単一ステップ・コマンドの使用法	469
[ステップまで実行] コマンドおよび [ステップから開始] コマンドの使用法	473
実行セッションの一時停止	475
ブレークポイントの使用法	476
デバッグ・ビューアの使用	480
実行エラーの処理	482
アクションまたは関数のデバッグの練習	483
第 18 章 コンポーネントの実行	487
コンポーネントの実行について	487
コンポーネント全体の実行	488
コンポーネントの一部の実行	493
コンポーネントの更新	494

第 19 章 テスト結果の分析	501
テスト結果の分析について.....	502
[テスト結果] ウィンドウについて.....	503
実行セッションの結果の表示.....	507
[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.....	518
テスト結果に含まれるスマート認識情報の分析.....	520
テスト結果の削除.....	524
Quality Center プロジェクトへの、実行セッション時に検出された 不具合の手作業による送信.....	533
テスト結果の表示のカスタマイズ.....	534

第 6 部 : 高度な機能を使用した作業

第 20 章 回復シナリオの定義と使用	539
回復シナリオの定義と使用について.....	540
回復シナリオを使用するタイミングの決定.....	542
回復シナリオの定義.....	543
回復シナリオ・ウィザードについて.....	547
回復シナリオの管理.....	570
アプリケーション領域用の回復シナリオ・リストの設定.....	574
プログラムによる回復メカニズムの制御.....	578
第 21 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	581
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.....	582
関数ライブラリの管理.....	583
関連付けられている関数ライブラリを使用した作業.....	594
関数定義ジェネレータの使用.....	597
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.....	610
ユーザ定義関数の使い方のヒント.....	615
第 22 章 関数ライブラリ・ウィンドウを使用した作業	619
[関数ライブラリ] ウィンドウを使った作業について.....	620
関数ライブラリでのステートメントの生成.....	620
関数ライブラリ内での操作.....	625
VBScript の基本的な構文の理解.....	633
プログラムの記述の使用.....	640
プログラムによるアプリケーションの実行と終了.....	649
コメント、フロー制御、その他の VBScript ステートメントの使用.....	650
テスト・オブジェクトのプロパティ値の取得と設定.....	656
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.....	657
DOS コマンドの実行.....	659
Windows API を使用したテストおよび関数ライブラリの拡張.....	659
実行セッション中に報告するステップの選択.....	663

第 23 章 QuickTest 操作のオートメーション	665
QuickTest 操作のオートメーションについて	666
QuickTest オートメーション・プログラムを使用する条件	667
オートメーション・プログラム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	668
QuickTest オートメーション・プログラムの基本要素の学習	670
オートメーション・スクリプトの生成	671
QuickTest オートメーション・オブジェクト・モデル・リファレンスの 使用	672

第 7 部 : 付録

付録 A QuickTest を使用した作業—よくある質問	675
コンポーネントの記録と実行	676
関数ライブラリを使用した作業	676
動的なコンテンツを使用した作業	677
Web に関する高度な問題	678
コンポーネントの保守	679
QuickTest のパフォーマンスの向上	680
付録 B QuickTest アドインの使用法	681
QuickTest アドインの使用法について	681
QuickTest アドインのロード	683
QuickTest アドインの使い方のヒント	688
Web アドインを使った作業	690
索引	697

ようこそ

QuickTest Professional for Business Process Testing ユーザーズ・ガイドへようこそ。ここでは、QuickTest Professional を使用して Business Process Testing の作業を行う方法について説明します。

注：Mercury Business Process Testing は、QuickTest と Quality Center に完全に統合されており、お持ちのライセンスに Business Process Testing サポートが含まれる場合は有効になります。

本書の使用法

本書では、コンポーネントが使用する各種のリソース・ファイルを定義する方法など、QuickTest を使用してコンポーネントの基礎となるアプリケーション領域を作成し管理する方法について説明します。また、QuickTest Professional で Business Process Testing のキーワード駆動のビジネス・コンポーネントおよびスクリプト・コンポーネントを使用した作業の方法についても説明します。

注：QuickTest を使用してビジネス・プロセス・テストで使用するスクリプト・コンポーネントを作成することもできますが、本書の対象は、主にビジネス・コンポーネントに関連する機能性や機能です。スクリプト・コンポーネントとビジネス・コンポーネントの違いの詳細については、第5章「スクリプト・コンポーネントを使った作業」を参照してください。

本書は次の部で構成されています。

第 1 部 Business Process Testing の紹介

QuickTest の概要と、Business Process Testing で作業を行う際のテスト・プロセスの主要な手順について説明します。

第 2 部 アプリケーション領域と コンポーネントを使用した作業

アプリケーション領域を作成し管理する方法について説明します。アプリケーション領域には、コンポーネントが使用するすべてのリソースと設定が含まれます。また、ビジネス・コンポーネント、スクリプト・コンポーネント、およびビジネス・コンポーネント・キーワード・ビューの作成方法と操作方法について説明し、欠落リソースの対処方法についても説明します。

第 3 部 設定

ビジネス・プロセス・テストのニーズに合うように QuickTest の設定を変更する方法について説明します。

第 4 部 テスト・オブジェクトおよび オブジェクト・リポジトリを使用した作業

QuickTest がアプリケーション内のオブジェクトを識別するしくみと、オブジェクト・リポジトリを使用した作業の方法について説明します。また、管理と結合方法についても説明します。

第 5 部 コンポーネントと実行のデバッグ

コンポーネントとそれらに関連付けられた関数ライブラリを実行する方法、結果を分析する方法、実行セッションを制御してバグを特定し切り分けを行う方法について説明します。

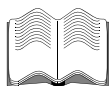
第 6 部 高度な機能を使用した作業

回復シナリオの使用法およびユーザ定義関数と関数ライブラリの作成と作業方法について説明します。また、QuickTest 操作の自動化について説明します。QuickTest においてアプリケーションのオブジェクトが識別される方法、およびオブジェクト・リポジトリの管理と結合の方法など、オブジェクト・リポジトリを使用した作業についても説明します。

第7部 付録

QuickTest の組み込みアドインのロードと使用の方法について説明し、よくある質問に対する情報を提供します。

製品マニュアル



この『Mercury QuickTest Professional for Business Process Testing ユーザーズ・ガイド』の他に、QuickTest Professional には次のマニュアルが付属しています。

『**QuickTest Professional インストール・ガイド**』: QuickTest Professional のインストール方法について説明します。

『**新情報**』（[ヘルプ] > [新情報] メニューから利用できます）では、QuickTest Professional の最新の機能と強化、および最新バージョンでサポートされる環境について説明します。

『**QuickTest Professional 基本機能ユーザーズ・ガイド**』: QuickTest Professional を使ってアプリケーションまたは Web サイトのテストを行うための手順を説明します。

『**QuickTest Professional 上級機能ユーザーズ・ガイド**』: QuickTest Professional を使ってアプリケーションまたは Web サイトのテストを行うための上級機能について説明します。

注: 『**QuickTest Professional 基本機能ユーザーズ・ガイド**』と『**QuickTest Professional 上級機能ユーザーズ・ガイド**』はオンラインでは1つの文書として用意されています。PDF版は、2つの文書に分かれています。

『**QuickTest Professional チュートリアル**』: QuickTest の基本的なスキルと、アプリケーションのテストを設計する方法を習得することができます。

『**Readme**』（[スタート] メニューの QuickTest Professional のプログラム・グループから利用できます）では、QuickTest Professional についての最新のニュースと情報を参照できます。

『印刷用ドキュメント』（[ヘルプ] > [印刷用ドキュメント] メニューから利用できます）では、すべてのマニュアルが Adobe PDF で表示されます。オンライン文書は、Acrobat Reader を使って読んだり印刷したりできます。Acrobat Reader は Adobe の Web サイト（<http://www.adobe.co.jp>）からダウンロードできます。

『QuickTest Professional コンテキスト・センシティブ・ヘルプ』（個々のダイアログ・ボックスとウィンドウから利用できます）は、QuickTest のダイアログ・ボックスおよびウィンドウを説明します。

『QuickTest Professional オブジェクト・モデル・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）では、QuickTest Professional テスト・オブジェクトについての説明、各オブジェクトと関連するメソッドとプロパティの一覧、構文情報とメソッドの例を参照できます。

『QuickTest Professional オートメーション・オブジェクト・モデル・リファレンス』（[スタート] メニューの QuickTest Professional プログラム・グループの [Documentation] からか、[ヘルプ] > [QuickTest オートメーションオブジェクトモデルリファレンス] から利用できます）では、構文と記述情報のほか、オートメーション・オブジェクト、メソッド、およびプロパティの各例を参照できます。また、QuickTest の自動スクリプトを記述する際の詳しい概要も記載されています。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。

『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、Windows Script ランタイム、および Windows Script Host などを含む Microsoft の VBScript 文書が含まれます。

追加オンライン・リソース

QuickTest Professional にはほかに次のオンライン・リソースが用意されています。

本書に含まれる多くの例は、サンプルの **Mercury Tours Web** サイト（[スタート] メニューの QuickTest Professional プログラム・グループか、QuickTest Professional の [記録と実行環境設定] ダイアログ・ボックスから利用できます）および、サンプルの **Mercury Tours Windows** アプリケーション（[スタート] メニューの QuickTest Professional プログラム・グループから利用できます）に基づいています。Mercury Tours Web サイトの URL は、<http://newtours.mercury.com> です。

「ナレッジ・ベース」（[ヘルプ] > [ナレッジ ベース] メニューから利用できます）は、普段お使いの Web ブラウザで Mercury カスタマー・サポートの Knowledge Base を開き、ナレッジ・ベース記事の閲覧や記事の投稿ができます。この Web サイトの URL は、<http://support.mercury.com/cgi-bin/portal/CSO/kbBrowse.jsp> です。

「カスタマー・サポート Web サイト」（[ヘルプ] > [カスタマ サポート Web サイト] メニューから利用できます）は、普段お使いの Web ブラウザで、Mercury カスタマー・サポート Web サイトを開きます。このサイトではサポートの要請をすることができます。この Web サイトの URL は、<http://www.mercury.com/jp/services/support/> です。

フィードバックの送信（[ヘルプ] > [フィードバックを送信] で利用できます）を使用すると、QuickTest Professional に関するオンライン・フィードバックが製品チームに送信されます。

Mercury のホーム・ページ（[ヘルプ] > [Mercury ホーム ページ] から利用できます）は、普段お使いの Web ブラウザで Mercury のホーム・ページを開きます。このサイトでは、Mercury の最新情報や製品に関する情報をご覧になれます。新しいソフトウェアのリリース、セミナー、展示会、カスタマー・サポート、教育サービスなどに関する情報をご覧いただけます。Mercury の Web サイトの URL は、<http://www.mercury.com/jp/> です。

「**Mercury** ベスト・プラクティス」は、第一級の IT 環境を計画、作成、展開および管理するためのガイドラインを含みます。Mercury では 3 タイプのベスト・プラクティスを用意しています。すなわち、プロセスに関するベスト・プラクティス (Process Best Practices)、プロダクトに関するベスト・プラクティス (Product Best Practices)、および人員に関するベスト・プラクティス (People Best Practices) の 3 つです。Mercury 製品を購入されたお客様は、Mercury ベスト・プラクティスをカスタマー・サポート Web サイト (<http://support.mercury.com>) でご覧いただけます。

文書の更新

Mercury では、製品マニュアルに新しい情報が反映されるよう常に更新を行っています。このマニュアルの最新版は Mercury カスタマー・サポート Web サイト (<http://support.mercury.com/>) からダウンロードできます。

更新された文書をダウンロードするには、次の手順を実行します。

- 1 カスタマー・サポート Web サイトで、[**Documentation**] リンクをクリックします。
- 2 [**Please select project**] で [**QuickTest Professional**] を選択します。
QuickTest Professional がリストに表示されていない場合は、顧客プロフィールに追加する必要があります。[**My Account**] をクリックしてプロフィールを更新します。
- 3 [**Retrieve**] をクリックします。文書のページが開き、現在のリリースと以前のリリースに関する使用可能な文書がリストされます。文書が最近更新された場合、文書名の隣に「**Updated**」のマークが表示されます。
- 4 文書のリンクをクリックして、文書をダウンロードします。

本書の表記規則

本書は、次の表記規則に従っています。

- [UI 要素]** このスタイルは、アクションの実行対象となるインタフェース要素の名前、ファイル名またはパス、その他強調を必要とする項目を示します。例：**[保存]** ボタンをクリックします。
- 引数** このスタイルは、メソッド、プロパティ、または関数の引数、および書名を示します。例：『**Mercury ユーザーズ・ガイド**』を参照してください。
- < 置換値 >** ファイル・パスまたは URL アドレスの中で実際の値に置き換える必要がある部分は、山括弧で囲んで示します。例：**<製品のインストール先フォルダ> %bin**
- 例** このスタイルは、使用例や、そのまま入力する必要があるテキストに使用します。例：編集ボックスに「**Hello**」と入力します。
- CTRL+C** このスタイルは、キーボードのキーを示します。例：**ENTER** キーを押します。
- 関数名** このスタイルは、メソッド名や関数名を示します。例：**wait_window** ステートメントには次のパラメータがあります。
- []** 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 3 つの点は、同じ形式で項目をさらに含め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。

ようこそ

第 1 部

Business Process Testing の紹介

第 1 章

QuickTest Professional for Business Process Testing の使用

QuickTest Professional for Business Process Testing へようこそ。Business Process Testing を使用すると、Quality Center で作業を行っている、専門知識のない各分野のエキスペートが、QuickTest Professional で作業を行っているオートメーション・エンジニアと効果的に共同作業を行うことができるようになります。各分野のエキスペートの側でプログラミングの知識がなくても、ビジネス・プロセス・テストの作成、文書化、および実行ができます。



注：QuickTest Professional は、Unicode 規格の要件に従って Unicode に準拠しており、各国の言語で開発されたアプリケーションをテストするための VBScript ステートメントの追加と更新が可能です。Unicode では、文字が 8 ビットまたは 16 ビットのコード値で表されます。Unicode 規格の詳細については、次を参照してください。 <http://www.unicode.org/standard/standard.html>

本書では、オートメーション・エンジニアが Business Process Testing に必要な自動リソースを作成および変更できるようにする QuickTest Professional の機能やオプションについて説明します。また、ビジネス・プロセス・テストの基本要素であるコンポーネントを作成するための機能やオプションについても説明します。

本章では、次の項目について説明します。

- ▶ QuickTest Professional for Business Process Testing の使用法について
- ▶ Business Process Testing について
- ▶ QuickTest Professional for Business Process Testing の用語について

- ▶ 必要なアクセス許可と権限の設定
- ▶ サンプル・サイトの使用
- ▶ ライセンス情報の変更
- ▶ QuickTest ソフトウェアの更新

QuickTest Professional for Business Process Testing の使用法 について

Business Process Testing はロール・ベースのテスト・モデルで、アプリケーションの開発ライフ・サイクルにおいて「オートメーション・エンジニア」と「各分野のエキスパート」が共同作業でアプリケーションのビジネス・プロセスをテストできるようにします。

Automation Engineer は、自動テストのエキスパートです。Automation Engineer は QuickTest を使用して、「コンポーネント」の作成に必要なリソースや設定を定義します。コンポーネントは「ビジネス・プロセス・テスト」の基本要素となるものです。

各分野のエキスパートは、テスト対象のアプリケーションのさまざまな部分や、テストを必要とするビジネス・プロセスを理解しています。ただし、必ずしも自動テストの作成に必要なプログラミング知識を持っているとは限りません。各分野のエキスパートは Quality Center のビジネス・コンポーネント・モジュールとテスト計画モジュールを使用して、キーワード駆動のビジネス・プロセス・テストを作成します。

QuickTest と Quality Center の統合により、Automation Engineer は必要なリソースと設定を効果的に作成し保守できるようになり、同時に各分野のエキスパートはプログラミングの知識がなくても、スクリプトが不要な環境でビジネス・プロセス・テストを作成し実装できるようになります。

注：組織では、それぞれのニーズに応じてオートメーション・エンジニアと各分野のエキスパートの役割を定めます。本書では、読者が上記に示したようなオートメーション・エンジニアの役割を担当しており、組織の他の人員が各分野のエキスパートの役割を担当していることを想定しています。ただし、これらの役割は柔軟なものであり、**Business Process Testing** を使用する人員の能力や時間的なリソースによって異なります。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 について

Business Process Testing では、自動テストと、自動的に生成され分かりやすく記述されたテスト文書を組み合わせることによって、構造化されたアプリケーション・テストを行うことができます。**Business Process Testing** は、詳細なテスト・スクリプトの完成に依存しません。このため、自動テストの準備が整うまでの間、アプリケーションを手動でテストすることが可能です。また、ビジネス・プロセス・テストを他の自動テストよりも短期間に作成し実装できるようになるため、開発プロセスの早い段階で、ダウンタイムが発生する前に、パフォーマンスの潜在的な問題を検出できます。

「コンポーネント」は、特定のタスクを実行するユニットで、保守しやすく再利用が可能です。コンポーネントはビジネス・プロセス・テストを構成する基本的な要素となります。各コンポーネントは、特定の順序で論理的にまとめて実行される、いくつかのアプリケーション・ステップで構成されています。たとえば、Web アプリケーションでは、ログイン・コンポーネントが 4 つのステップで構成されることが考えられます。1 番目のステップでアプリケーションを開き、2 番目のステップでユーザ名を入力します。3 番目のステップでパスワードを入力し、4 番目のステップで Web ページの [送信] ボタンをクリックする、などが考えられます。関数を作成したり、関数ライブラリに格納されている関数を呼び出したりすることによって、コンポーネントを拡張し、ログイン・タスクの重要な詳細部分をテストするためのロジックを追加することができます。

仕様上、コンポーネントはそれぞれアプリケーションの特定の部分をテストします。コンポーネントを組み合わせ、一連のフローとしてビジネス・プロセス・テストに組み込むことによって、特定のビジネス・プロセスの中で実行される主要なタスクを表現します。たとえば、フライト予約アプリケーション用のビジネス・プロセス・テストには、ログイン・コンポーネント、フライト情報検索コンポーネント、フライト予約コンポーネント、購入コンポーネント、およびログアウト・コンポーネントを含めることができます。フライト情報検索、フライト予約、購入の各コンポーネントは、複数の予約事例をテストする場合に、同じビジネス・プロセス・テストの中で複数回再利用されることが考えられます。また、各フライト予約の合間にアプリケーションをリセットするコンポーネントをテストに含めることによって、フライト予約を複数回繰り返して実行することも考えられます。コンポーネントやビジネス・プロセス・テストを作成し実行する作業は、通常は Quality Center で作業を行う各分野のエキスパートによって実行されます。

コンポーネントはモジュール化でき、再利用が可能なため、複数のビジネス・プロセス・テストで使用できます。たとえば、同じログインおよびログアウト・コンポーネントを、アプリケーションのレポートおよびグラフ生成プロセスをテストする分析（レポート）コンポーネントと組み合わせて使用したり、マイルージ・サービス制度の申し込みを行うビジネス・プロセスをテストするマイルージ・サービス制度コンポーネントと組み合わせて使用したりできます。

QuickTest では、「ビジネス・コンポーネント」と「スクリプト・コンポーネント」の2種類のコンポーネントが提供されています。ビジネス・コンポーネント（キーワード駆動コンポーネントとも呼ばれます）は QuickTest と Quality Center の両方に完全に統合されているので、Automation Engineer と各分野のエキスパートの双方がビジネス・コンポーネントを作成、変更、および実行できます。スクリプト・コンポーネントは、プログラミング・ロジックを含んだ、より複雑なコンポーネントです。スクリプト・コンポーネントは複雑であるため、QuickTest においてのみ作成と変更が行えます。各分野のエキスパートは、Quality Center でスクリプト・コンポーネントを表示し、ビジネス・プロセス・テストに組み込むことはできますが、変更を加えることはできません。

注：QuickTest を使用してビジネス・プロセス・テストで使用するスクリプト・コンポーネントを作成することもできますが、本書ヘルプではビジネス・コンポーネントに関連する機能と特徴について説明します。スクリプト・コンポーネントとビジネス・コンポーネントの違いやスクリプト・コンポーネントの使用の詳細については、第 5 章「スクリプト・コンポーネントを使った作業」を参照してください。

自動テスト・リソースが利用できるようになるまでの間、各分野のエキスパートは各コンポーネントの [ステップのデザイン] タブで手動ステップを定義できます (Quality Center のビジネス・コンポーネント・モジュールを使用します)。そして、それらの手動コンポーネントをビジネス・プロセス・テストに追加し、Quality Center マニュアル・ランナーを使用してステップを手動で実行できます。コンポーネントを定義する際、各分野のエキスパートは (Quality Center のビジネス・コンポーネント・モジュール内の) [詳細] タブの [ディスカッション エリア] 領域にコメントを追加できます。ここでは、新しい操作の要求や、コンポーネントに予定されている今後の変更、コンポーネントを使用できるほかのテストなど、各分野のエキスパートがオートメーション・エンジニアに伝える必要のある補足情報や備考を入力できます。

この設計段階で、各分野のエキスパートと連携して作業し、どのようなリソースや設定が各コンポーネントに必要なかを定めることができます。そして、テストの実際のニーズに基づいて、アプリケーションのさまざまな部分に対応する個々の「**アプリケーション領域**」を作成することができます。アプリケーション領域は、ビジネス・プロセス・テストでの作業時にコンポーネントが使用する設定とリソース・ファイルを指定するものです。各分野のエキスパートがコンポーネントを作成するとき、コンポーネントは常に特定のアプリケーション領域に関連付けられ、それによってコンポーネントからそれらの設定やリソース・ファイルへのアクセスが可能になります。オートメーション・エンジニアがアプリケーション領域を作成し、その設定とリソース・ファイルを定義した後、各分野のエキスパートはそれらの自動テスト・リソースをビジネス・コンポーネント・ステップに組み込み、既存の手動コンポーネントを自動コンポーネントに変換して、新しい自動コンポーネントを作成することができます。

アプリケーション領域について

アプリケーション領域はコンポーネントの土台となるものです。アプリケーション領域によって、アプリケーションの特定部分のテストに関係するすべての要素の保守を一か所で行えるようになります。

アプリケーション領域では、アプリケーションの特定部分のテストに関する具体的な設定を定義できます。たとえば、実行セッションの開始時に特定のアドインをロードする、指定したアプリケーションのみを対象としてコンポーネントを実行する、特定の条件のもとで回復シナリオを起動するなどの処理を QuickTest に指示する設定を定義できます。また、対象アプリケーション領域に関連付けられている任意のコンポーネントで使用できるキーワードを指定することもできます。

アプリケーション領域の重要な要素の 1 つに、コンポーネントが使用できるリソース・ファイルがあります。作成したリソース・ファイルは、対象アプリケーションのビジネス・プロセス・テストを作成して実行する各分野のエキスパートが使用している、同じ Quality Center プロジェクトに格納します。通常、リソース・ファイルには関数ライブラリと共有オブジェクト・リポジトリが含まれます。

コンポーネントに呼び出される、関数または**操作**（キーワードとも呼ばれる）を含む関数ライブラリを作成します。これらの関数には、特定のタスクを実行するためのステップをカプセル化するプログラミング・ロジックが含まれており、呼び出し元のコンポーネントの機能はこれらの関数によって強化されます。関数ライブラリは、QuickTest に組み込まれている関数ライブラリ・エディタを使用して作成できます。または、QuickTest 関数定義ジェネレータを使用して基本的な関数定義を挿入し、コードを追加して関数を完成させることもできます。

関数ライブラリ・ファイルをアプリケーション領域に関連付けた後、関連性に従って優先順位を付けることができます。関数ライブラリ・ファイルをアプリケーション領域に関連付けることによって、そのアプリケーション領域をベースとするすべてのコンポーネントから、その関数ライブラリの中で定義されているすべてのパブリック関数にアクセスできるようになります。関数ライブラリを使用した作業の詳細については、第 21 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。

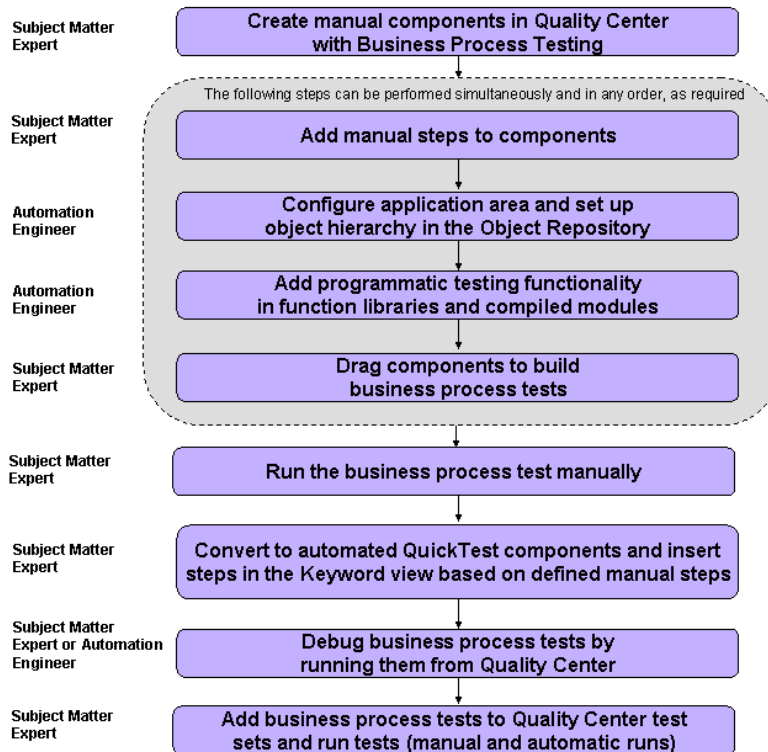
QuickTest がアプリケーション内のオブジェクトを識別するために使用する共有オブジェクト・リポジトリ・ファイルの作成、情報登録、および保守もできます。QuickTest オブジェクト・リポジトリ・マネージャを使用すると、共有オブジェクト・リポジトリ内のテスト・オブジェクト情報を定義および変更できます。共有オブジェクト・リポジトリ・ファイルをアプリケーション領域に関連付けた後、関連性に従って優先順位を付けることができます。共有オブジェクト・リポジトリをアプリケーション領域に関連付けることによって、そのアプリケーション領域をベースとするすべてのコンポーネントから、そのすべてのテスト・オブジェクトや他の要素にアクセスできるようになります。詳細については、第 13 章「テスト・オブジェクトを使用した作業」および第 15 章「オブジェクト・リポジトリの管理」を参照してください。

複数のアプリケーション領域を作成し、それぞれの領域がテスト対象アプリケーションの特定の部分（範囲）を対象とするようにできます。たとえば、フライト予約アプリケーションの場合、アプリケーション領域をログイン・モジュール用に 1 つ、フライト検索モジュール用に 1 つ、フライト予約モジュール用に 1 つ、請求書発行モジュール用に 1 つ、というように作成できます。アプリケーション領域の詳細については、第 3 章「アプリケーション領域の使用」を参照してください。

アプリケーション領域に関連付けられるリソース・ファイルの作成および保守に加えて、QuickTest では、コンポーネントとそれに関連付けられている関数ライブラリのデバッグも可能です。QuickTest でコンポーネントを作成することもできますが、この作業は多くの場合、各分野のエキスパートが Quality Center を使用して行います。詳細については、第 4 章「ビジネス・コンポーネントの使用」を参照してください。

QuickTest Professional を使用した Business Process Testing のワークフロー

次の例は、QuickTest を使用した Business Process Testing の一般的なワークフローです。組織の実際のワークフローは、プロジェクトや、製品開発ライフ・サイクルの段階に応じて異なる場合があります。



QuickTest Professional for Business Process Testing の用語について

本書では、QuickTest Professional for Business Process Testing に固有の次の用語を使用しています。

アプリケーション領域：ビジネス・コンポーネントの作成と実装に使用されるリソースと設定の集まり。これには、関数ライブラリ、共有オブジェクト・リポジトリ、キーワード、テスト設定、または回復シナリオなどのその他のテスト・リソースが含まれます。アプリケーション領域によって、アプリケーションの特定部分のテストに関係するすべての要素の保守を一か所で行えるようになります。アプリケーションの部分ごとに別々のアプリケーション領域を定義し、コンポーネントを該当するアプリケーション領域に関連付けることができます。

ビジネス・コンポーネント（または**コンポーネント**）：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た、保守しやすく再利用可能なユニット。ビジネス・コンポーネントは、外部ソースやほかのコンポーネントからの入力値を必要とする場合や、ほかのコンポーネントに出力値を返す場合があります。

キーワード駆動コンポーネントとも呼ばれます。

手動コンポーネント：Quality Center で作成される非自動のビジネス・コンポーネント。QuickTest では、手動コンポーネントはビジネス・コンポーネントに変換した後のみ、表示および操作できます。

スクリプト・コンポーネント：プログラミング・ロジックを含めることができ、QuickTest でキーワード・ビュー、エキスパート・ビュー、またはQuickTest のその他のツールやオプションを使用して編集できる、自動コンポーネント。

キーワード・ビュー：キーワード方式のモジュール化された表形式を使用して、テストとコンポーネントの作成、表示、およびデバッグを実行できる、スプレッドシート風のビュー。

関数ライブラリ：VBScript の関数、サブルーチン、モジュールなどを含むドキュメント。これらの関数はコンポーネントの中で操作（キーワード）として使用できます。QuickTest 関数ライブラリ・エディタを使用して、関数ライブラリ・ドキュメントの作成とデバッグができます。

ビジネス・プロセス・テスト：アプリケーションの特定のビジネス・プロセスをテストするように設計された、ビジネス・コンポーネントの連続的な流れで構成されるシナリオ。

コンポーネント入力パラメータ：ビジネス・コンポーネントが受け取り、コンポーネントの特定のパラメータ化されたステップの値として使用することのできる変数値。

コンポーネント出力パラメータ：ビジネス・コンポーネントが返すことのできる値。これらの値は、ビジネス・プロセス・テストの結果に表示され、テストの後の段階で使用されるコンポーネントへの入力値としても使用できます。

ローカル入力パラメータ：コンポーネントの内部で定義される変数値。これらの値は、同じコンポーネント内の後続のパラメータ化されたステップが受け取って使用できます。

ローカル出力パラメータ：同じコンポーネント内で使用するために、操作またはコンポーネント・ステップが返すことのできる値。これらの値は、ビジネス・プロセス・テストの結果内に表示され、コンポーネントの後のステップへの入力値としても使用できます。

役割（ロール）：Business Process Testing に関与するさまざまなタイプのユーザ。

オートメーション・エンジニア：QuickTest Professional 自動テストのエキスパート。オートメーション・エンジニアは、ビジネス・コンポーネントの作成と使用に必要なリソースを定義し、管理します。オートメーション・エンジニアはアプリケーション領域を作成します。アプリケーション領域は、各分野のエキスパートが Quality Center においてビジネス・コンポーネントおよびビジネス・プロセス・テストを作成するのに必要なすべてのリソースと設定を指定します。オートメーション・エンジニアは、関数ライブラリを作成および変更でき、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・オブジェクトを共有オブジェクト・リポジトリに登録できます。オートメーション・エンジニアは QuickTest の中でビジネス・コンポーネントの作成とデバッグも行えます。

各分野のエキスパート：アプリケーション・ロジックに関する具体的な知識を持ち、システム全体を高いレベルで理解し、テスト対象アプリケーションの基礎となる個々の要素やタスクの詳細を理解している人。各分野のエキスパートは、Quality Center を使用してコンポーネントやビジネス・プロセス・テストを作成し、実行します。

必要なアクセス許可と権限の設定

QuickTest Professional を実行するには、次のアクセス許可と権限が設定されていることを確認します。

QuickTest Professional の実行に必要なアクセス許可

ファイル・システムに対する次のアクセス許可が必要です。

- ▶ QuickTest のインストール先の配下にあるすべてのファイルとフォルダに対する読み取りおよび書き込みのアクセス許可
- ▶ Temp フォルダに対する読み取りおよび書き込みのアクセス許可
- ▶ Windows フォルダおよび System フォルダに対する読み取りアクセス許可

レジストリ・キーに対する次のアクセス許可が必要です。

- ▶ **HKEY_CURRENT_USER\Software\Mercury Interactive** の配下にあるすべてのキーに対する読み取りおよび書き込みのアクセス許可
- ▶ **HKEY_LOCAL_MACHINE** および **HKEY_CLASSES_ROOT** キーに対する読み取りおよび値クエリのアクセス許可

Quality Center の使用に必要なアクセス許可

次の Quality Center アクセス許可が必要です。

- ▶ Quality Center キャッシュ・フォルダに対する読み取りおよび書き込みのアクセス許可
- ▶ Quality Center インストール・フォルダの QuickTest アドインに対する読み取りおよび書き込みのアクセス許可

Business Process Testing の使用に必要な権限

Quality Center プロジェクト管理者は、プロジェクトにログインできるユーザを定義したり、各ユーザが実行できるタスクの種類を指定したりすることで、プロジェクトに対するアクセスを制御できます。Quality Center プロジェクト管理者は、Quality Center プロジェクトのビジネス・コンポーネント・モジュール内のフォルダ、コンポーネント、ステップおよびパラメータを追加、変更、および削除するための権限を割り当てることができます。

注：アプリケーション領域に変更を加えるには、コンポーネントの変更、およびステップの追加、変更、削除の権限が必要です。4 つすべての権限が必要です。1 つでも割り当てられていない場合は、アプリケーション領域を読み取り専用形式でのみ開けます。

ビジネス・コンポーネントおよびアプリケーション領域の作業を行う前に、必要な Quality Center 権限を持っていることを確認する必要があります。ビジネス・コンポーネント・モジュール内のユーザ・グループ権限の設定の詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。

サンプル・サイトの使用

本書ヘルプに示す例の多くは、サンプルの Web サイト、Mercury Tours に基づいています。この Web サイトの URL は <http://newtours.mercury.com> です。

このサイトを使用するには、ユーザ名とパスワードを登録する必要があります。

また、Mercury Tours サンプル Windows アプリケーションを使用することもできます。これは、[スタート] メニューの [QuickTest Professional] プログラム・フォルダから利用できます。

ライセンス情報の変更

QuickTest を使用するにはライセンスが必要です。QuickTest をインストールする際に、次のライセンス・タイプのいずれかを選択します。

- ▶ 14 日間有効のデモ・ライセンス
- ▶ インストール先のコンピュータに固有の無期限のシート・ライセンス
- ▶ 複数の QuickTest ユーザが使用できるネットワーク・ベースのコンカレント・ライセンス

ライセンスの種類は、いつでも変更できます（コンピュータに管理者権限でログインしている場合）。たとえば、現在デモ・ライセンスを使用している場合、シート・ライセンスをインストールしたり、ネットワーク上にコンカレント・ライセンス・サーバがあれば、コンカレント・ライセンス・サーバに接続することを選択したりできます。

新しいシート・ライセンスの申請が必要な場合は、Mercury カスタマー・サポート Web サイトで申請できます。ライセンス申請用の Web サイトは、<http://www.mercury.com/jp/services/license-req/> です。

外部アドインを購入する場合は、関連するアドイン・ライセンスをインストールする必要があります。詳細については、アドインのマニュアルを参照してください。

ライセンス情報の変更の詳細については、『QuickTest Professional インストール・ガイド』を参照してください。

QuickTest ソフトウェアの更新

標準設定では、QuickTest アプリケーションを起動するたびに、ソフトウェアのオンライン更新がないか自動的に確認されます。更新は、任意の時点で手動で確認することもできます。手動で確認するには、QuickTest 内で **[ヘルプ]** > **[更新の確認]** を選択するか、**[スタート]** メニューの **[QuickTest Professional]** プログラム・グループから **[更新を確認する]** を選択します。

利用可能な更新がある場合は、ダウンロード（および必要に応じてインストール）する更新を選択できます。詳細については、画面に表示される指示に従います。

ヒント：更新の自動チェックを無効にするには、**[オプション]** ダイアログ・ボックスの **[一般]** タブにある **[起動時にソフトウェアの更新を確認する]** チェック・ボックスをクリアします。**[オプション]** ダイアログ・ボックスを開くには、**[ツール]** > **[オプション]** を選択します。

第 2 章

QuickTest の概要

本章では、QuickTest の開始方法を説明し、QuickTest ウィンドウを紹介します。

本章では、次の項目について説明します。

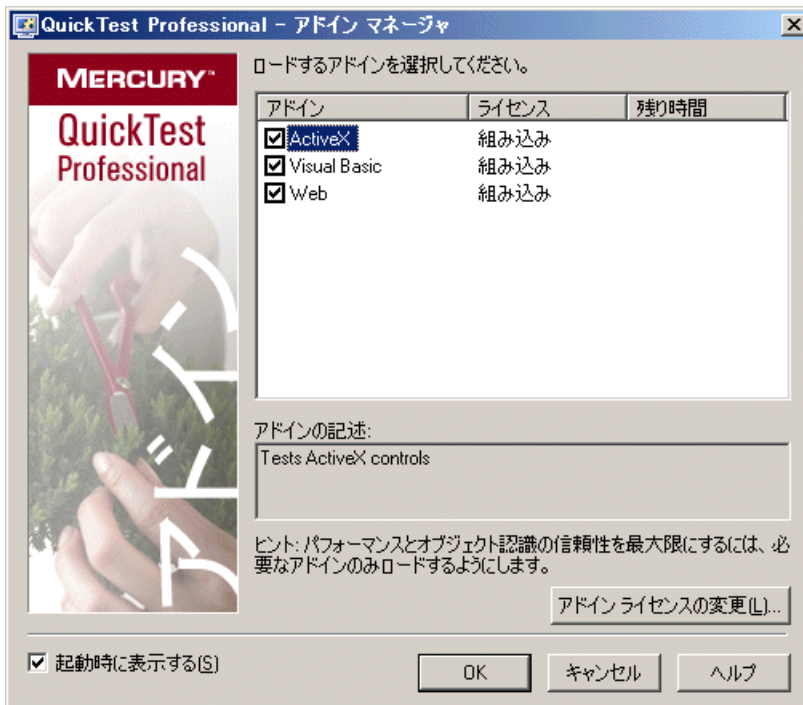
- ▶ QuickTest の開始
- ▶ Quality Center プロジェクトへの接続
- ▶ QuickTest のウィンドウ
- ▶ キーワード・ビュー
- ▶ アプリケーション領域
- ▶ 関数ライブラリ
- ▶ 情報表示枠
- ▶ 欠落リソース表示枠
- ▶ デバッグ ビューア表示枠
- ▶ QuickTest ウィンドウのレイアウトのカスタマイズ
- ▶ 複数のドキュメントを使った作業
- ▶ QuickTest コマンドの使用方法
- ▶ QuickTest Professional プログラム・フォルダの参照
- ▶ 製品情報の表示

QuickTest の開始



QuickTest を開始するには、[スタート] メニューから [QuickTest Professional] プログラム・グループを開き、[QuickTest Professional] を選択するか、デスクトップにある [QuickTest Professional] ショートカットをダブルクリックします。

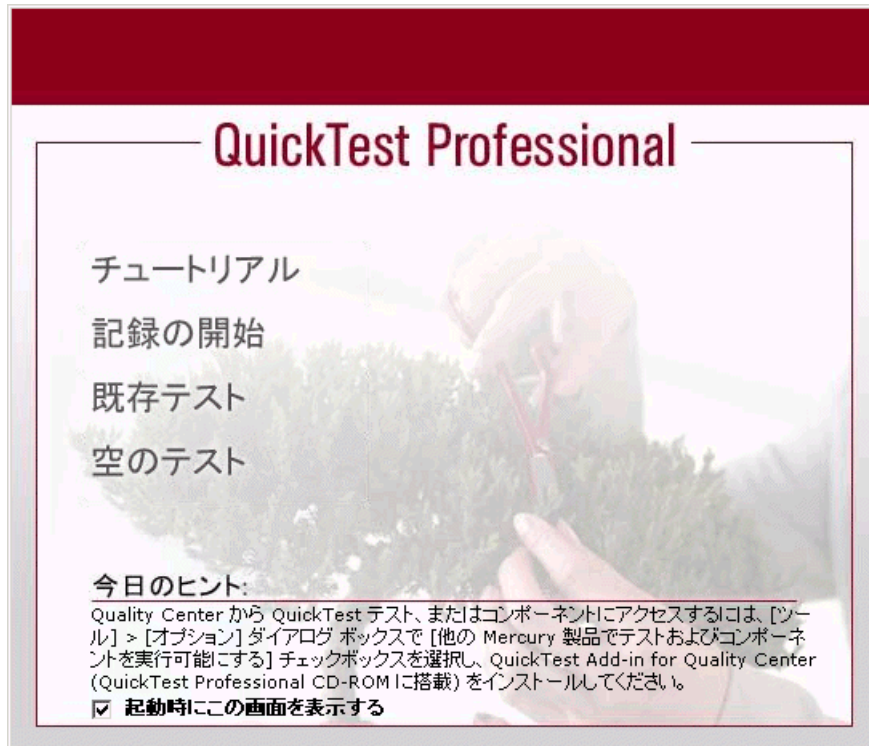
初めて QuickTest を開始すると、[アドインマネージャ] ダイアログ・ボックスが開きます。



ヒント：次回 QuickTest を開始したときにこのダイアログ・ボックスが開かないようにするには、[起動時に表示する] チェック・ボックスをクリアします。

アドインの読み込みの詳細については、683 ページ「QuickTest アドインのロード」を参照してください。

[OK] をクリックします。[QuickTest Professional] ウィンドウが開きます。QuickTest チュートリアルを開くか、新規テストの記録を開始するか、既存のテストを開くか、または空の新規テストを開く操作を選択できます。



ヒント :

Esc キーを押すと、ウィンドウが閉じて空のテストが開きます。

[今日のヒント] をクリックすると、利用できるヒントをすべて参照できます。

次回 QuickTest を開始したときにこのダイアログ・ボックスが開かないようにするには、[起動時にこの画面を表示する] チェック・ボックスをクリアします。

Quality Center プロジェクトへの接続

ビジネス・プロセス・テストの作業を行うためには、QuickTest から、Quality Center プロジェクトが格納されている Quality Center サーバに接続する必要があります。このサーバによって、QuickTest と Quality Center プロジェクトの間の接続が処理されます。

Quality Center プロジェクトには、テスト対象のアプリケーションのコンポーネントと実行セッションに関する情報が格納されます。この情報には、ビジネス・プロセス・テストの作成および実行に必要なすべてのリソース・ファイルと設定が含まれます。初めて Quality Center サーバに QuickTest を接続すると、QuickTest によって、標準設定の Business Process Testing フォルダとファイルがプロジェクトの中に設定されます。これにより、ビジネス・コンポーネントに必要なリソースと設定を準備できるほか、直観的なキーワード駆動型のキーワード・ビューを使ってビジネス・コンポーネントの作成、操作、デバッグを行うことができます。

注：Quality Center 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 から Quality Center プロジェクトに接続するには、次の手順を実行します。



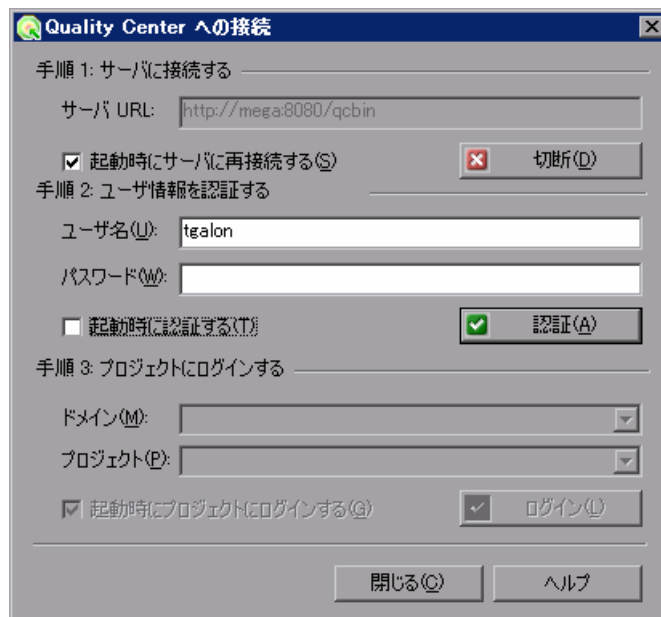
- 1 [ファイル] > [Quality Center への接続] を選択するか、ツールバーの [Quality Center への接続] ボタンをクリックします。[Quality Center への接続 - サーバへの接続] ダイアログ・ボックスが表示されます。



- 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）または広域ネットワーク（WAN）を介してアクセスできる Web サーバを選択します。

- 3 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。[Quality Center への接続] ダイアログ・ボックスが表示されます。



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。

- 5 [ユーザ名] ボックスに、Quality Center ユーザ名を入力します。
- 6 [パスワード] ボックスに、Quality Center のパスワードを入力します。

- 7 [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] 領域の各フィールド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。

- 8 [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。ユーザが接続の権限を持っているドメインのみ表示されます。
- 9 [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。ユーザが接続の権限を持っているプロジェクトのみ表示されます。
- 10 [ログイン] をクリックします。
- 11 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。
- 12 [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。
- 13 [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。

注：初めて Quality Center サーバに接続するときには、QuickTest が標準設定の Business Process Testing フォルダとファイルを Quality Center プロジェクトの中に設定します。

- 14 **[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 への接続を表示するには、ステータス・バーの **[Quality Center]** アイコンをポイントします。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・ボックスを再度開くには、ステータス・バーの **Quality Center** アイコンをダブルクリックします。

Quality Center への QuickTest の接続の解除

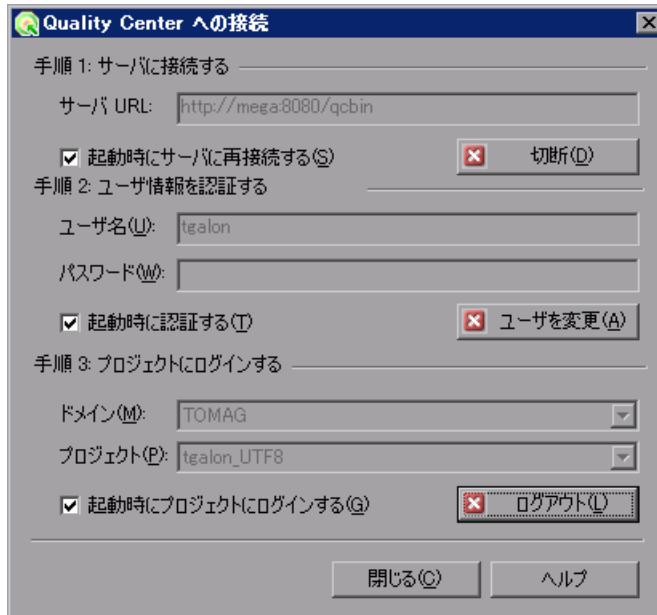
QuickTest から Quality Center プロジェクトまたは Quality Center Web サーバへの接続はいつでも解除できます。ただし、QuickTest のコンポーネント、アプリケーション領域、または共有リソース（共有オブジェクト・リポジトリなど）を Quality Center から開いている間、または QuickTest で Quality Center の共有リソースを使用している間は、QuickTest から Quality Center への接続を解除しないでください。

最初にプロジェクトとの接続を解除せずに、QuickTest から Quality Center Web サーバへの接続を解除すると、QuickTest からプロジェクトへの接続は自動的に解除されます。

Quality Center から QuickTest への接続を解除するには、次の手順を実行します。



- 1 [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。



- 2 選択したプロジェクトへの QuickTest の接続を解除するには、[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 Web サーバから QuickTest への接続を解除するには、[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。

- 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 のウィンドウ

QuickTest ウィンドウのドキュメント領域には、テスト・ドキュメントが表示されます。

1つのコンポーネントまたはアプリケーション領域と、1つ以上の関数ライブラリを対象に同時に作業ができます（アクティブ・ドキュメント1つをドキュメント領域に表示することも、複数のドキュメントを重ねて表示または並べて表示することもできるので便利です）。詳細については、50 ページ「複数のドキュメントを使った作業」を参照してください。

QuickTest ウィンドウのドキュメント表示枠領域には次のタイプのドキュメントが表示されます。

- ▶ **ビジネス・コンポーネント**：キーワードと操作を使用するビジネス・コンポーネントを作成、表示、変更できます。詳細については、第6章「ビジネス・コンポーネント・キーワード・ビューを使った作業」を参照してください。
- ▶ **スクリプト・コンポーネント**：キーワード・ビューまたはエキスパート・ビューで、スクリプト・コンポーネントを作成、表示、変更できます（後述）。スクリプト・コンポーネントの詳細については、第5章「スクリプト・コンポーネントを使った作業」を参照してください。エキスパート・ビューの詳細については、『QuickTest Professional ユーザーズ・ガイド』を参照してください。
- ▶ **アプリケーション領域**：コンポーネントのリソースと設定を定義できます。詳細については、第3章「アプリケーション領域の使用」を参照してください。
- ▶ **関数ライブラリ**：コンポーネントで使用する関数（操作）を作成、表示、変更できます。詳細については、第21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。

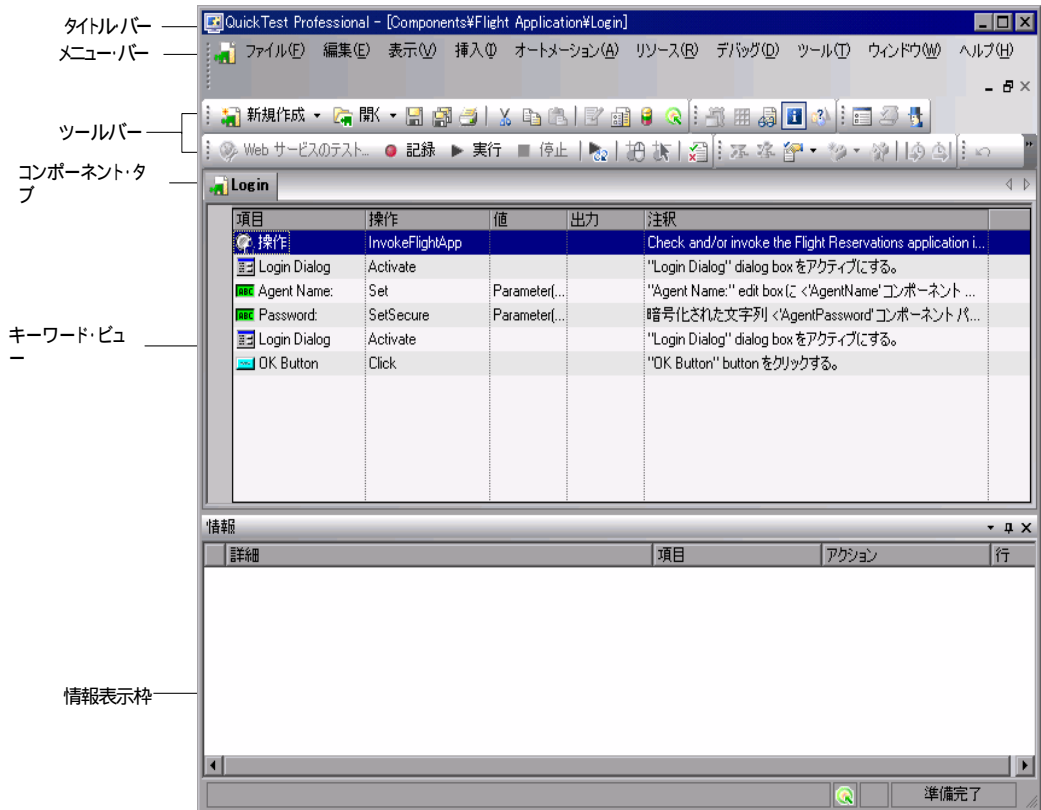
ドキュメント領域に加えて、QuickTest ウィンドウには次の主要な要素があります。

- ▶ **QuickTest タイトル・バー**：アクティブなドキュメントの名前が表示されます。前回保存したとき以降に加えられた変更がある場合は、タイトル・バーにアスタリスク（*）が表示されます。
- ▶ **メニュー・バー**：QuickTest コマンドのメニューが表示されます。
- ▶ **標準ツールバー**：ドキュメント管理を支援するボタンが含まれます。
- ▶ **オートメーション・ツールバー**：テスト・プロセスを支援するボタンが含まれます。

- ▶ **デバッグ・ツールバー**：ドキュメントのデバッグに使用するボタンが含まれません。（標準設定では表示されません）。
- ▶ **編集ツールバー**：関数ライブラリの編集を支援するボタンが含まれます。
- ▶ **挿入ツールバー**：関数ライブラリ内のステートメントを使用する際に役立つボタンが含まれます。
- ▶ **ツール・ツールバー**：テスト・プロセス中に使用するツールのボタンが含まれます。
- ▶ **表示ツールバー**：ドキュメントの参照に使用するボタンが含まれます。
- ▶ **ドキュメント・タブおよびスクロール矢印**：アクティブにする（フォーカスを移動する）ドキュメントのタブを選択することで、ドキュメント領域に開いているドキュメント間を移動できます。すべてのタブを同時に表示するのに十分なスペースがドキュメント領域にない場合は、左右の矢印を使用して、開いているドキュメント間をスクロールできます。
- ▶ **キーワード・ビュー**：各ステップが、モジュール化されたアイコン・ベースの表形式で表示されます。詳細については、第6章「ビジネス・コンポーネント・キーワード・ビューを使った作業」を参照してください。
- ▶ **情報表示枠**：関数ライブラリのスクリプトで見つかった構文エラーのリストが表示されます。
- ▶ **欠落リソース表示枠**：未割り当ての共有オブジェクト・リポジトリ、共有オブジェクト・リポジトリに結び付いているパラメータなど、コンポーネント中で指定されているけれども見つからなかったリソースの一覧が表示されます（標準設定では表示されません）。
- ▶ **デバッグ ビューア表示枠**：ドキュメントのデバッグを支援します。デバッグビューア表示枠には、**[ウォッチ]**、**[変数]**、および **[コマンド]** タブがありません（標準設定では表示されません）。
- ▶ **ステータス・バー**：QuickTest アプリケーションのステータスが表示されます。

QuickTest ウィンドウのレイアウトはカスタマイズ可能で、ほとんどの要素について移動、サイズ変更、表示と非表示の切り替えができます。ユーザのレイアウト設定は記憶されるので、次回以降のセッションはカスタマイズしたレイア

ウトで開かれます。詳細については、41 ページ「QuickTest ウィンドウのレイアウトのカスタマイズ」を参照してください。



QuickTest ウィンドウの外観の変更

標準設定では、QuickTest ウィンドウには Microsoft Office 2003 のテーマが使用されますが、必要に応じて QuickTest のメイン・ウィンドウの外観は変更できます。

QuickTest のメイン・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。

QuickTest 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：QuickTest ウィンドウに Microsoft Windows XP テーマを適用できるのは、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のみです。

ヒント：[テスト結果] 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、506 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。

キーワード・ビュー

[キーワード ビュー] では、コンポーネントのステップを、キーワード駆動のモジュール化された表形式で作成および表示できます。キーワード・ビューは、表に似た表示形式で構成されます。各ステップは表内では個別の行として表示され、各カラムはステップの別々の部分を表します。表示されるカラムは必要に応じて変更できます。

コンポーネントを作成および変更するには、キーワード・ビューの中で項目と操作を選択し、必要に応じて情報を入力します。完了した各ステップは自動的に文書化され、テストのステップの説明がわかりやすい言葉で表されます。

キーワード・ビューのカラム

項目	操作	値	注釈
▼ Welcome: Mercu...			
▼ banner2			
Click	Click		"banner2" image をクリックする。
userName	Set	"mercury"	"userName" edit box に "mercury" を入力する。
password	SetSecure	"43b79fbfd823491578189c43a20f6E"	暗号化された文字列 "43b79fbfd823491578189c43a20f6E" を入力する。
Sign-In	Click	15,7	"Sign-In" image をクリックする。
▶ インストール...	Exist		"インストール オプション (そのほか)" dialog box が存在する
▶ Find a Flight: Mer...	Check	CheckPoint("Find a...")	選択されたプロパティで "Find a Flight: Mercury" Web page (
▶ Select a Flight: ...			
▶ Book a Flight: M...	Check	CheckPoint("Chec...")	選択されたプロパティで "Book a Flight: Mercury" Web page
▶ Flight Confirmation...	Check	CheckPoint("Chec...")	"Flight Confirmation: Mercury" Web page にあるテキストが期
Action2			

ステップは、コンポーネントに手作業で追加することも、アプリケーションを対象に実行したステップを記録することで追加することもできます。セッションの記録中には、各ステップがそれぞれキーワード・ビュー内の1行として記録されます。たとえば、キーワード・ビューには次の行が含まれる場合があります。

Welcome: Mercury ...			
userName	Set	'mercury'	'userName' edit box に 'mercury' を入力する。
password	SetSecure	'3ee35'	暗号化された文字列 '3ee35' を 'password' edit box に入力する。
Sign-In	Click	2,2	'Sign-In' image をクリックする。

これらの行は、Mercury Tours サンプル Web サイト「**Welcome: Mercury Tours**」ページで実行されるステップを示します。

- ▶ **userName** エディット・ボックスに「**Mercury**」と入力されます。
- ▶ 暗号化された文字列 **4129e9544fe92be22d38ebcb7cf5** が、**password** エディット・ボックスに入力されます。
- ▶ **Sign-In** 画像がクリックされます。

[注釈] カラムでは、各ステップがわかりやすい文に変換されます。

ヒント：コンポーネントの内容は、Windows の標準設定のプリンタで印刷したり、印刷する前に画面でプレビューしたりできます。詳細については、140 ページ「コンポーネントの印刷」を参照してください。

ビジネス・コンポーネント・キーワード・ビューには、[項目]、[操作]、[値]、[出力]、[注釈] のカラムを含めることができます。以下では各カラムについて簡潔に説明します。

注：以下で説明するカラムの一部がキーワード・ビューに表示されていない場合は、[キーワード ビュー オプション] ダイアログ・ボックスを使用してそれらを表示できます。詳細については、32 ページ「キーワード・ビューの表示オプションの設定」を参照してください。

【項目】カラム

ステップまたは**【操作】**の実行対象となるテスト・オブジェクト。**【操作】**では、ユーザ定義関数ライブラリから操作を選択できます。項目はドロップダウン・リストから選択します。このリストには、直前のステップの**【操作】**項目とすべての子テスト・オブジェクトが表示されます。使用可能な子テスト・オブジェクトがない場合は、兄弟テスト・オブジェクトが表示されます。たとえば、前のステップで「**userName**」オブジェクトが指定されていた場合は、次のステップのために「**password**」テスト・オブジェクトが表示されます。

オブジェクト・リポジトリ・ツリーから、またはテスト対象アプリケーションから直接、ほかのテスト・オブジェクトを選択できます。詳細については、156 ページ「ステップの項目の選択」を参照してください。共有オブジェクト・リポジトリ内のテスト・オブジェクトの管理については、94 ページ「共有オブジェクト・リポジトリの管理」を参照してください。

コメントや手動ステップをコンポーネントに追加するには、**【項目】**カラムから**【コメント】**を選択します。これにより、キーワード・ビューにコメント行が追加されます。コメントは自由にテキストを入力できるセルに表示され、このセルは行全体に伸びて**【項目】**カラムの一部ではなくなります。

【操作】カラム

項目に対して実行する操作。このカラムには、**Click** や **Select** など、**【項目】**カラムで選択した項目に対して実行できる、よく使用される操作（メソッド、関数、サブプロシージャ）のリストが含まれています。標準設定で、**【項目】**カラムで選択した項目に対して最もよく使用される操作が表示されます。詳細については、164 ページ「ステップの操作の選択」を参照してください。

RegisterUserFunc メソッドを使用して、テスト・オブジェクトに対して追加の操作を定義できます。詳細については、581 ページ「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。

【値】カラム

選択されている操作に対する引数値。**【値】**セルは、**【操作】**カラムで選択したオプションの引数の数に従って分割されています。選択されたオプションに応じて、定数、**ローカル・パラメータ**、または**コンポーネント・パラメータ**を値として指定できます。

ローカル・パラメータ：ローカル・パラメータは該当するビジネス・コンポーネントに固有のパラメータであり、そのコンポーネントでしかアクセスできません。ローカル・パラメータは、単一のステップ内で、またはコンポーネント・ステップ間で使用されます。たとえば、あるステップの出力パラメータを後続のステップの入力パラメータとして使用できます。詳細については、172 ページ「パラメータの使用」を参照してください。

コンポーネント・パラメータ：コンポーネント・パラメータは、Quality Center プロジェクト内のどのコンポーネントからでもアクセスできるパラメータです。詳細については、239 ページ「コンポーネントのパラメータの定義」を参照してください。

【出力】 カラム

ステップの出力値が格納されるパラメータ。たとえば、**cCols** という名前の出力パラメータを選択した場合、現在のステップの出力値は **cCols** パラメータに格納されます。出力パラメータに格納された値は、以降、同じコンポーネント内で入力パラメータとして使用できます。[値] カラムと同様に、出力パラメータを指定するときには2種類のパラメータ（ローカル・パラメータまたはコンポーネント・パラメータ）を使用できます。

【注釈】 カラム

自動的に生成される読み取り専用の説明です。ステップの実行内容がわかりやすい文で表示されます。ステップだけを印刷または表示する場合は、このカラムだけを表示するようにできます。たとえば、手動テストの指示だけを印刷または表示できます。

キーワード・ビューの表示オプションの設定

表示するカラムと、表示順を指定できます。また、キーワード・ビューの行および内容に対してさまざまな色やフォントを適用することもできます。

表示するキーワード・ビューのカラムを指定するには、次の手順を実行します。

- 1 [ツール] > [表示オプション] を選択します。[キーワード ビュー オプション] ダイアログ・ボックスが表示されます。



[**利用可能なカラム**] リストには、現在キーワード・ビューに表示されていないカラムの一覧が表示されます。[**表示するカラム**] リストには、現在キーワード・ビューに表示されているカラムの一覧が表示されます。

- 2 [**利用可能なカラム**] リストと [**表示するカラム**] リストの間でカラム名を移動するには、カラム名をダブルクリックするか、カラム名を選択して矢印ボタン (> および <) をクリックします。

ヒント：一方のリストから他方のリストにすべてのカラム名を移動するには、二重の矢印ボタン (>> および <<) をクリックします。選択した複数のカラム名のみを一方のリストから他方のリストに移動するには、(SHIFT キー /CTRL キーを使用して) 複数のカラム名を選択し、矢印ボタン (> および <) をクリックします。



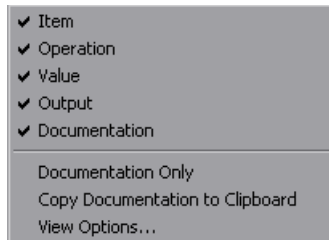
- 3 **〔表示するカラム〕** リストで、1つまたは複数のカラムを選択して、上矢印ボタンと下矢印ボタンを使って、キーワード・ビューでのカラムの表示順を設定します。

注：キーワード・ビューでのカラムの順序は、各ステップに関して完了しなければならないセルの順序には影響しません。たとえば、**〔項目〕**カラムの左側に**〔操作〕**カラムが表示されるように選択した場合も、まずその項目を選択する必要があります。その場合のみ、**〔項目〕**カラムで行った選択に一致するよう**〔操作〕**カラム・リストが更新されます。

- 4 **〔OK〕** をクリックしてダイアログ・ボックスを閉じ、新しいカラム表示を適用します。

キーワード・ビューを使用した作業のヒント

特定のカラムの表示 / 非表示を切り替えるには、キーワード・ビュー内のカラム・ヘッダ行を右クリックして、表示されるメニューから必要なカラム名を選択または選択解除します。



たとえば、使用するステップを手動テスト用の指示書として印刷する場合などに、**[ドキュメントのみ]**を選択することによって**[注釈]**カラムだけを素早く表示することができます。**[注釈]**カラムとコンポーネント内で定義されたコメントが表示されます。

また、カラム・ヘッダをキーワード・ビューの新しい位置にドラッグすることで、カラムの並べ替えができます。カラム見出しを有効な場所にドラッグすると、赤い矢印が表示されます。



キーワード・ビューのフォントと色の設定

キーワード・ビュー内の各種要素に対して異なるテキストや色の表示オプションを指定するには、**[キーワード ビュー オプション]** ダイアログ・ボックスの**[フォントと色]** タブを使用します。



[フォントと色] タブには、次のオプションがあります。

オプション	詳細
[要素]	<p>キーワード・ビューの要素ごとに、フォントと色のオプションを個別に指定できます。現在の定義を確認および変更するには、次の要素のいずれかを選択します。</p> <ul style="list-style-type: none"> • [コメント]：コメント行の行とテキスト。 • [交互の行]：1行おきに使用する代替の背景色。この背景色が適用される行でも、フォントとテキストについては[標準設定]要素で定義したフォントとテキストの色が適用されます。 • [選択されている行]：現在選択されている（強調表示されている）行とテキスト。 • [標準設定]：キーワード・ビューのすべての行とテキスト（次に示す要素を除く）。
[フォント名]	<p>選択した要素のテキストに使用するフォントを変更できます。[交互の行]要素および[選択されている行]要素のフォントは変更できません。</p> <p>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。選択しなかった場合、テストまたはコンポーネント内の要素がキーワード・ビューで正しく表示されません。ただし、テストおよびコンポーネントは、選択したフォントにかかわらず、同じように実行されます。</p>
[サイズ]	<p>選択した要素のテキストに使用するフォント・サイズを変更できます。[交互の行]要素および[選択されている行]要素のフォント・サイズは変更できません。</p>
[スタイル]	<p>選択した要素のテキストに使用するフォント・スタイルを変更できます。[通常]、[太字]、[斜体]、または[下線]フォント・スタイルを選択できます。[交互の行]または[選択されている行]要素のフォント・スタイルは変更できません。</p>
[前景]	<p>選択した要素のテキストに使用する色を変更できます。[交互の行]の前景の色は変更できません。</p>
[背景]	<p>選択した要素の行の色を変更できます。</p>

オプション	詳細
[読み取り専用向けの前景]	読み取り専用の行のテキストに使用する色を変更できます。このオプションは、 [交互の行] が対象の場合には変更できません。
[すべて元に戻す]	[フォントと色] タブのすべてのオプションを標準設定に戻します。

アプリケーション領域

各ビジネス・コンポーネントは、それぞれ何らかのアプリケーション領域に基づいています。アプリケーション領域によって、ビジネス・コンポーネントが使用する設定や特定のリソース・ファイル（関数ライブラリ・ファイル、共有オブジェクト・リポジトリ（アプリケーションによって使用されるテスト・オブジェクトを含みます）、関連するアドイン、回復シナリオ・ファイルなど）へのリンクが提供されます。こうした資産はアプリケーション領域ウィンドウで定義します。

アプリケーション領域ウィンドウには4つの表示枠があり、それぞれ左側のサイドバーにあるボタンからアクセスできます。

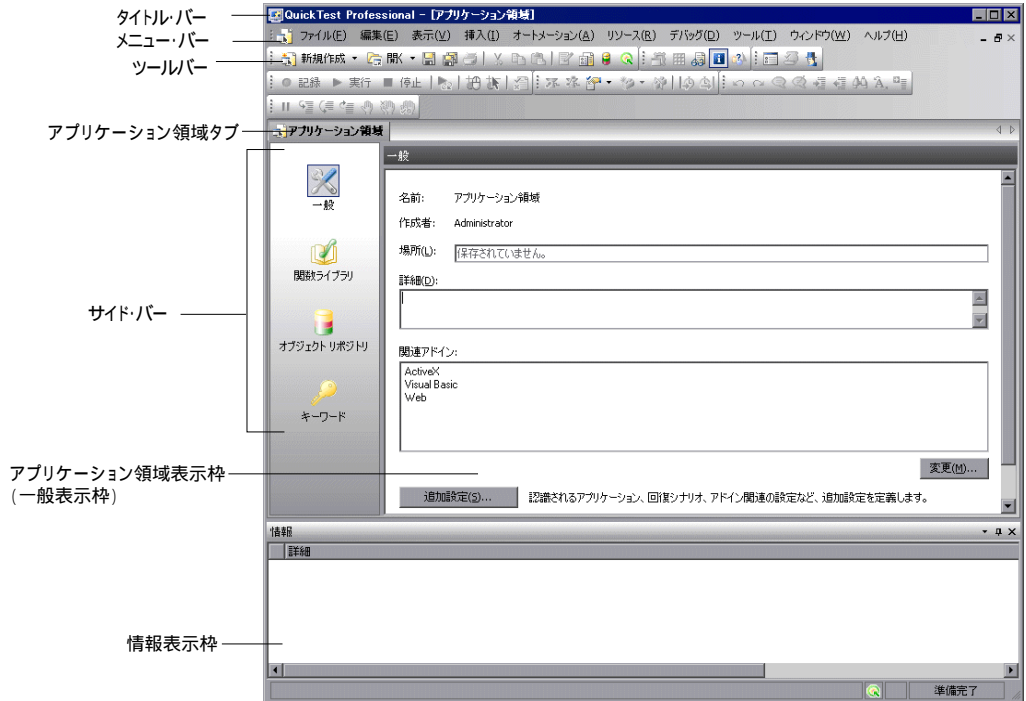
[一般]：アプリケーション領域に関する一般情報を表示します。関連アドインや回復シナリオ、およびその他の設定の指定など、一般設定を変更できます。

[関数ライブラリ]：関数ライブラリをこのアプリケーション領域に関連付けて優先順位を付けることができます。

[オブジェクトリポジトリ]：共有オブジェクト・リポジトリをこのアプリケーション領域に関連付けて優先順位を付けることができます。

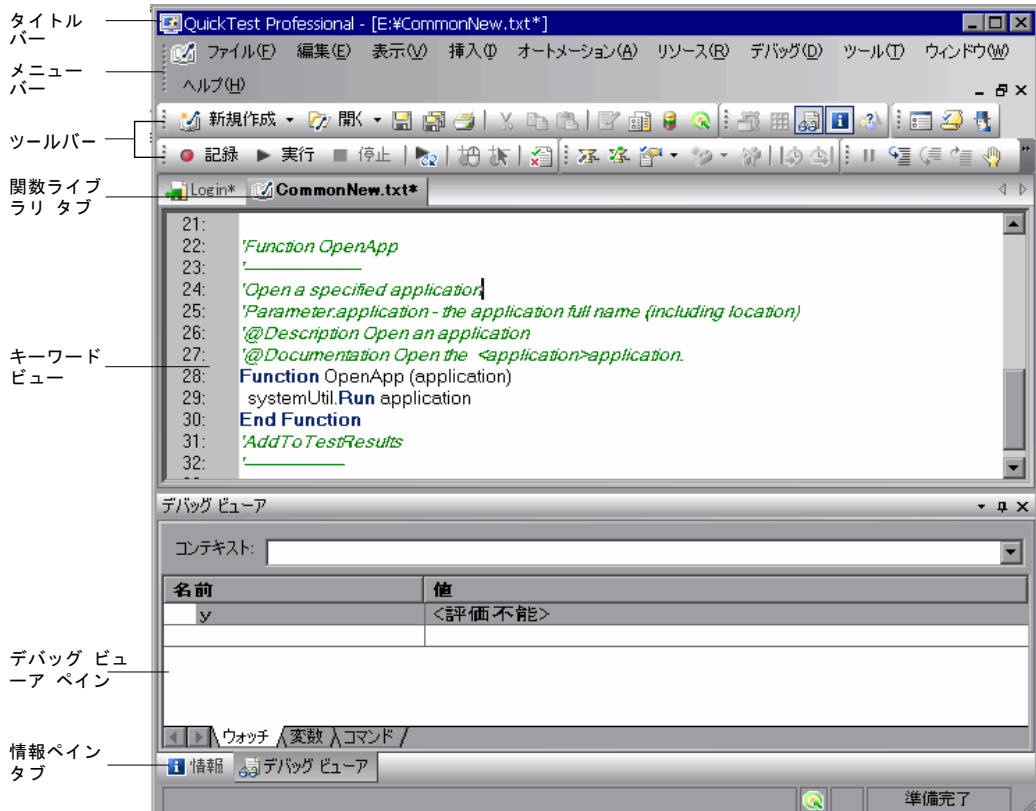
[キーワード]：このアプリケーション領域で使用できるキーワードを設定し、それらの個々のプロパティを表示できます。

詳細については、第3章「アプリケーション領域の使用」を参照してください。



関数ライブラリ

QuickTest の組み込みエディタでは、エキスパート・ビューで使用できるものと
 同じ編集機能を使用して、関数ライブラリを作成およびデバッグできます。各
 関数ライブラリは、VBScript 関数、サブルーチン、クラス、モジュールなどが
 含まれる独立した QuickTest 文書です。個々の関数ライブラリは、すでに開い
 ているコンポーネントに加えて、それぞれ別のウィンドウに開きます。関数ライ
 ブラリは複数と同時に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く
 ことができます。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。詳細については、第 21 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。



情報表示枠



情報表示枠には、関数ライブラリのスクリプトで見つかった構文エラーのリストが表示されます。情報表示枠の表示と非表示を切り替えるには、[表示] > [情報] を選択するか、[情報] ボタンをクリックします。

詳細	項目	アクション	行
Expected "]"	Library1.qfl	N/A	1
Expected "]"	Library1.qfl	N/A	4
Expected end of statement	Library1.qfl	N/A	6
Expected expression	Library1.qfl	N/A	7

構文エラーをダブルクリックすると、関数ライブラリ内でそのエラーに該当する場所が表示され、その場で修正できます。詳細については、638 ページ「VBScript 構文エラーの処理方法」を参照してください。

欠落リソース表示枠



欠落リソース表示枠には、未割り当ての共有オブジェクト・リポジトリ、共有オブジェクト・リポジトリに結び付いているパラメータなど、テストに指定されているにもかかわらず、見つからなかったリソースのリストが表示されます。欠落リソース表示枠の表示と非表示を切り替えるには、[表示] > [欠落リソース] を選択するか、[欠落リソース] ボタンをクリックします。

項目	詳細
欠落しているオブジェクトリポジトリ	C:\Program Files\Mercury Interactive\QuickTest Profess

関数ライブラリを開くたびに、指定されたすべてのリソースがアクセス可能かどうか自動的にチェックされます。アクセスできないリソースが見つかった場合、そのリソースは欠落リソース表示枠に一覧表示されます。欠落リソース表示枠が表示されていない場合でも、存在しないリソースが検出されると自動的に表示されます。

存在しないリソースをダブルクリックすると、そのリソースを再割り当て、または削除できます。表示枠でフィルタを設定して、特定のタイプの欠落リソース（[欠落しているオブジェクトリポジトリ] など）を表示し、それ以外のタイプを非表示にすることもできます。

詳細については、183 ページ「欠落リソースの処理」を参照してください。

デバッグ ビューア表示枠



デバッグ ビューア表示枠には、関数ライブラリのデバッグに役立つ3つのタブ（[ウォッチ]、[変数]、[コマンド]）があります。デバッグ ビューア表示枠を表示するには、[デバッグ ビューア] ボタンをクリックするか、[表示] > [デバッグ ビューア] を順に選択します。

ウォッチ

[ウォッチ] タブでは、[ウォッチ] タブに追加した変数や VBScript 式の現在の値を表示できます。

変数

実行セッション中、[変数] タブには実行セッションで実行された最後のステップまでに認識されている、現在のすべての変数値が表示されます。

コマンド

[コマンド] タブを使って、テストの変数または 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、スクリプトの行を実行できます。テストの実行を続けると、コマンドで設定した新しい値が使用されます。

デバッグ ビューア表示枠の使用方法の詳細については、第17章「コンポーネントと関数ライブラリのデバッグ」を参照してください。

QuickTest ウィンドウのレイアウトのカスタマイズ

QuickTest ウィンドウのレイアウトはカスタマイズできます。また、標準のレイアウトに戻すこともできます。ウィンドウのカスタマイズでは、表示枠の移動およびサイズ変更、表示枠の表示と自動非表示の選択、タブ形式の表示枠の作成、表示するツールバーの選択などが可能です。

注：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、すべてのドキュメント・タイプのレイアウトがカスタマイズまたは復元されます。

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。

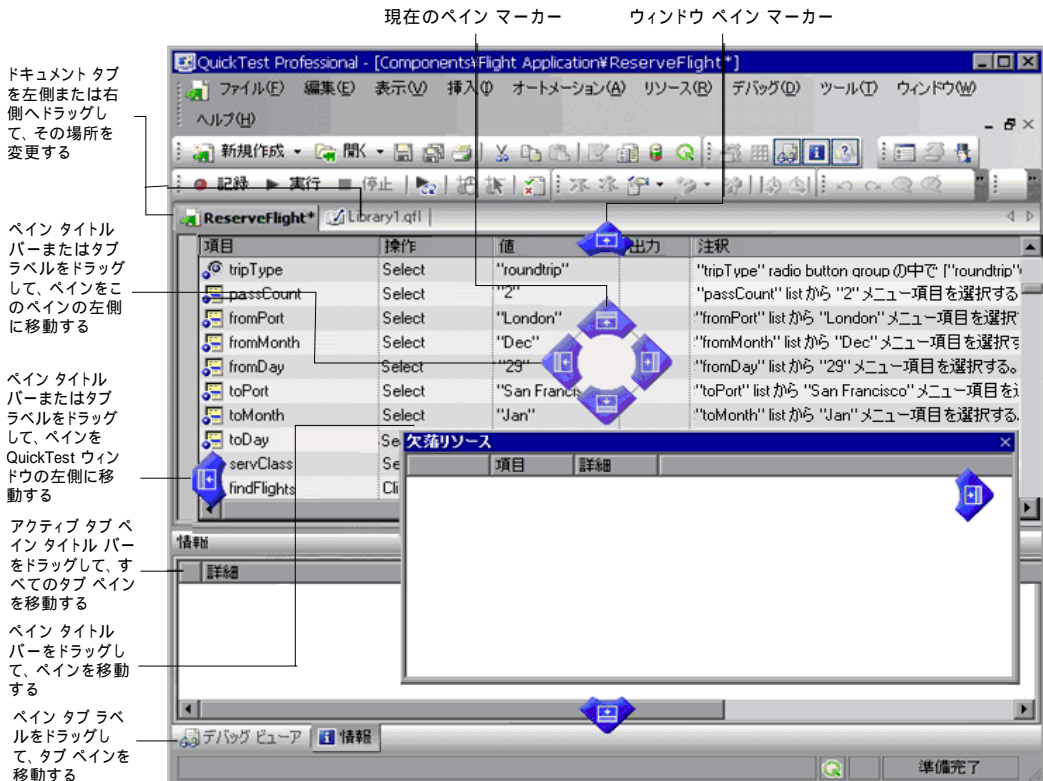
ヒント：ドッキング可能な表示枠がはめ込まれないようにして移動するには、CTRL キーを押したままで目的の場所へドラッグします。

表示枠を移動するには、次の手順を実行します。

- 1 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。

ヒント：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、ウィンドウの中央にある [欠落リソース] 表示枠を、新しいタブ形式の表示枠としてウィンドウの一番下に移動できます。表示枠をドラッグする間は、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。








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。

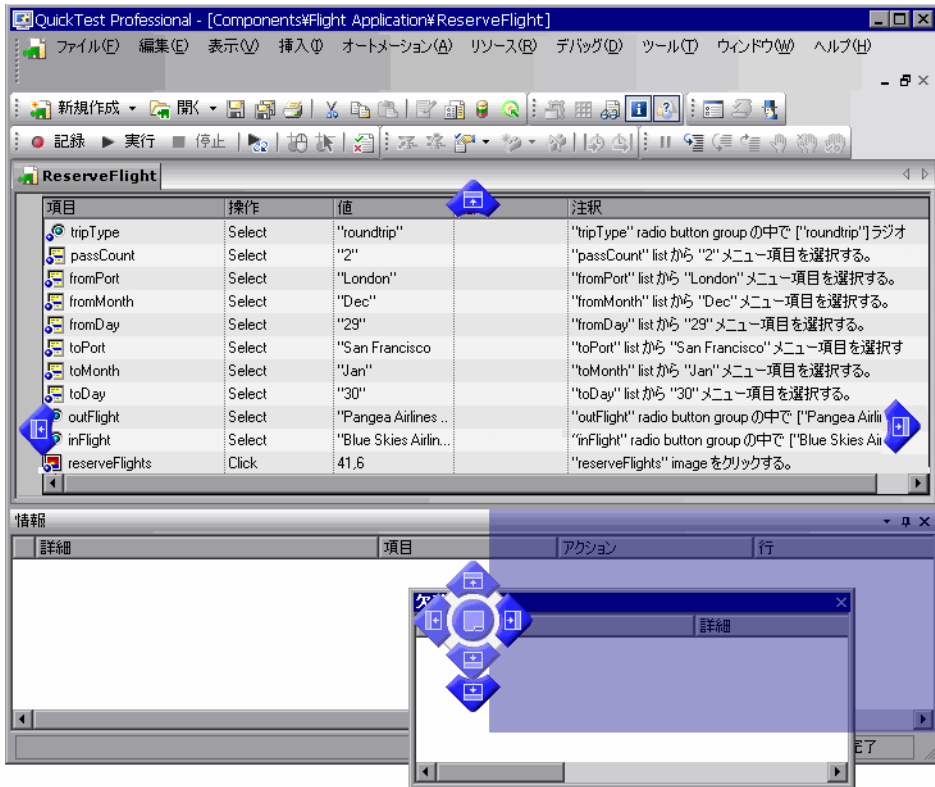
タブ形式の表示枠をすべて移動するには、アクティブなタブ形式の表示枠のタイトル・バーをドラッグします。

次のマーカが表示されます。

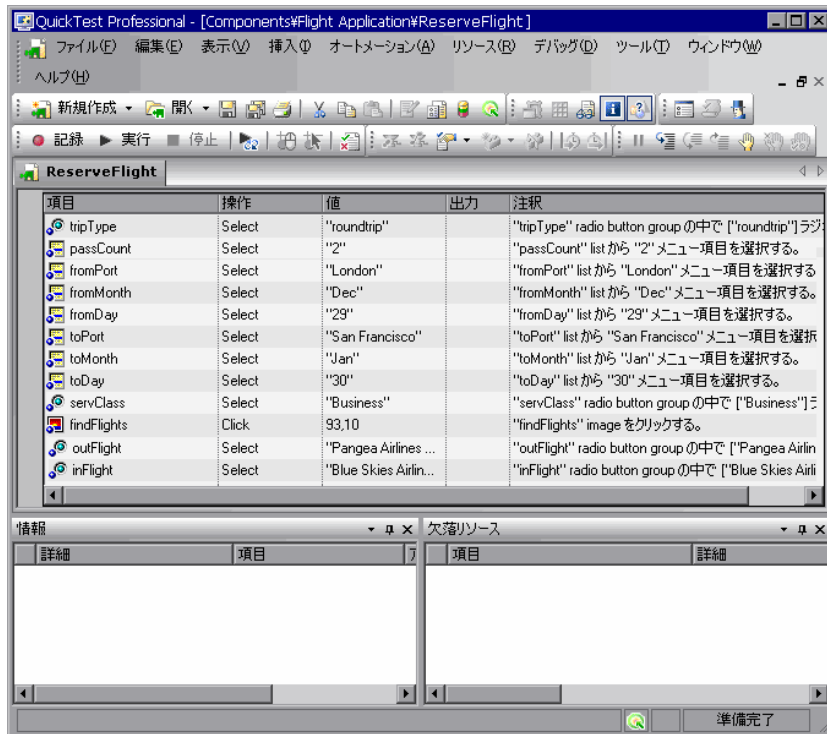
タイプ	マーカ	詳細
現在表示枠 マーカ		<p>次のことが可能です。</p> <ul style="list-style-type: none"> マウス・ボタンを放したときに選択されていた矢印マーカに応じて、移動した表示枠を、アクティブな表示枠内の上下左右いずれかの側、または中央に、新しい表示枠として配置します。 中央のマーカが選択されているときにマウス・ボタンを放して、移動した表示枠を、アクティブなウィンドウ内に新しいタブ形式の表示枠として配置します。 <p>注：中央のマーカは、移動する表示枠を、(ドキュメント表示枠を除く) 既存の表示枠内にドラッグしている場合にのみ表示されます。</p>
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。
		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、[欠落リソース] 表示枠をドラッグし、アクティブな表示枠の右矢印マーク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。



- 3 マウス・ボタンを放します。グレー表示の領域に [欠落リソース] 表示枠がはめ込まれ、新しい表示枠として表示されます。



ヒント：また、フローティング表示枠にして QuickTest ウィンドウ上または画面の任意の場所に配置しておくこともできます。フローティング表示枠の詳細については、46 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 ウィンドウの右側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なります。



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、画面上の任意の位置に（QuickTest ウィンドウの外側にも）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
	<p>[メニュー] ボタンでは次を選択できます。</p> <ul style="list-style-type: none"> • [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。
	<p>[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。</p>

ボタン	詳細
	<p>[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。</p>
	<p>[閉じる] ボタンは、表示枠を閉じ、QuickTest ウィンドウ上に表示されないようにします。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。</p> <p>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。</p>

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 [**フォローティング**] 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。

注： **[オプション]** ダイアログ・ボックスの詳細については、第8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

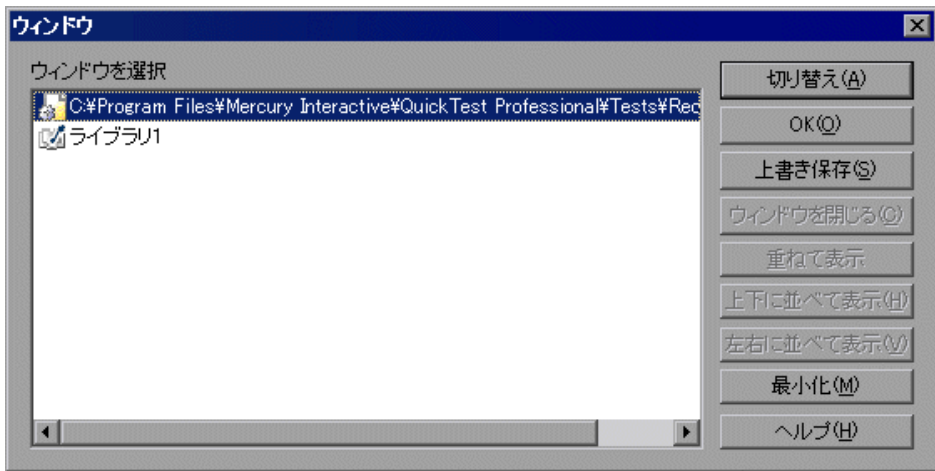
QuickTest では、1つのコンポーネントまたはアプリケーション領域を開いて作業できるのに加え、複数の関数ライブラリを同時に開いて操作できます。現在開いているコンポーネントまたはアプリケーション領域に関連付けられているかどうかに関係なく、任意の関数ライブラリを開くことができます。

[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。
[OK]	[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。

QuickTest コマンドの使用方法

QuickTest のコマンドはメニュー・バーまたはツールバーから選択できます。QuickTest では、コンポーネントおよびアプリケーション領域に対して、それぞれ異なるコマンドとツールバーのセットが表示されます。各セットは、作成または変更されるドキュメントに合わせてカスタマイズされています。QuickTest コマンドには、ショートカット・キーを押したり、ショートカット（右クリック）・メニューでコマンドを選択したりして実行できるものがあります。メニューとツールバーは、アクティブなドキュメントのタイプに応じて有効になります。

メニュー・バーからはほとんどのコマンドを使用できます。よく使用する QuickTest コマンドは、ツールバーのボタンをクリックして実行できます。

メニューでのコマンドの選択

メニュー・バーからはほとんどの QuickTest コマンドを使用できます。

ツールバーでのコマンドのクリック

ツールバーのボタンをクリックして、いくつかの QuickTest コマンドを実行できます。QuickTest には 8 個の組み込みツールバーがあります。すなわち、**標準** ツールバー、**編集** ツールバー、**オートメーション** ツールバー、**表示** ツールバー、**挿入** ツールバー、**ツール** ツールバー、**デバッグ** ツールバー、**アクション** ツールバーです。

注：

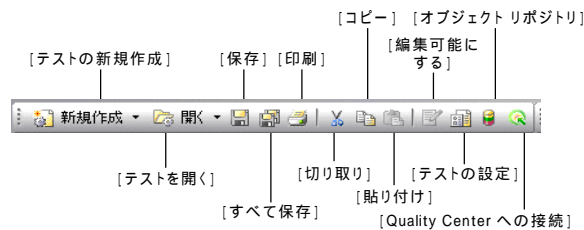
すべてのツールバーがすべてのドキュメント・タイプに関係するわけではありません。ここでは、コンポーネント、アプリケーション領域、および関数ライブラリに関するツールバーについてのみ説明しています。

ツールバーは、表示と非表示を切り替えたり移動したりできますが、カスタマイズはできません。

標準ツールバー

標準ツールバーには、コンポーネント、アプリケーション領域または関数ライブラリを管理するためのボタンがあります。ビジネス・コンポーネントの使用方法的詳細については、第4章「ビジネス・コンポーネントの使用」を参照してください。スクリプト・コンポーネントの使用方法的詳細については、第5章「スクリプト・コンポーネントを使った作業」を参照してください。アプリケーション領域の使用方法的詳細については、第3章「アプリケーション領域の使用」を参照してください。関数ライブラリの使用方法的詳細については、第21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。

標準ツールバーに表示されるのは次のボタンです。



注： [新規作成] ボタンおよび [開く] ボタンのアイコンは、アクティブなドキュメントのタイプ（コンポーネント、アプリケーション領域、関数ライブラリなど）に応じて変わります。

オートメーション・ツールバー

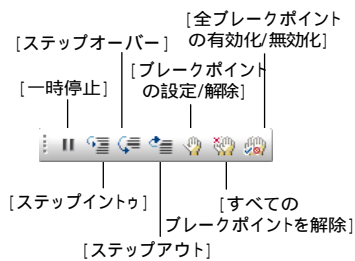
オートメーション・ツールバーには、コンポーネントの記録および実行のためのボタンがあります。オートメーション・ツールバーに表示されるのは次のボタンです。



注：[アナログ記録] と [低レベル記録] はテストにのみ適用されます。

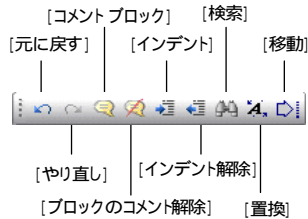
デバッグ・ツールバー

デバッグ・ツールバーには、コンポーネント、および関連付けられている関数ライブラリのステップをデバッグする際に使用するコマンドのボタンがあります。デバッグ・ツールバーに表示されるのは次のボタンです。



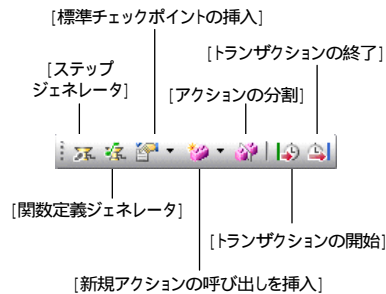
編集ツールバー

編集 ツールバーには、関数ライブラリの編集に使用するコマンドのボタンがあります。**編集** ツールバーは表示されるのは次のボタンです。



挿入ツールバー

挿入 ツールバーには、関数ライブラリの操作に使用するコマンドのボタンがあります。**挿入** ツールバーに表示されるのは次のボタンです。



注： 関数ライブラリに関するボタンは、[**ステップジェネレータ**] ボタンと [**関数定義ジェネレータ**] ボタンのみです。コンポーネントに関するボタンはありません。

ツール・ツールバー

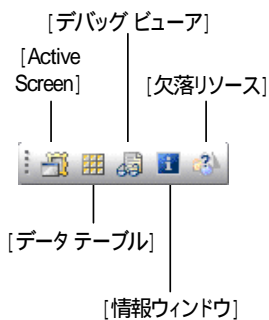
ツール・ツールバーには、テストでの作業に役立つツールにアクセスするために使用するコマンドのボタンがあります。ツール・ツールバーに表示されるのは次のボタンです。



注：[構文チェック] ボタンは関数ライブラリ（および QuickTest テスト）にのみ関係するもので、コンポーネントには関係しません。

表示ツールバー

表示ツールバーには、QuickTest ウィンドウのさまざまな要素を表示するためのボタンがあります。表示ツールバーに表示されるのは次のボタンです。



注：[ActiveScreen] と [データ テーブル] はテストにのみ適用されます。

ショートカット・キーを使用したコマンドの実行

ショートカット・キーを押すことで、いくつかの QuickTest コマンドを実行できます。次に示すショートカット・キーは、対応するメニュー・コマンドに表示されます。

対応するショートカット・キーを押すことで、次の [ファイル] メニュー・コマンドを実行できます。

コマンド	ショートカット・キー	機能
[新規作成] > [テスト]	CTRL+N	新規のテストを作成する。
[新規作成] > [ビジネスコンポーネント]	CTRL+SHIFT+N	新規のビジネス・コンポーネントを作成する。
[新規作成] > [アプリケーション領域]	CTRL+ALT+N	新規のアプリケーション領域を作成する。
[新規作成] > [関数ライブラリ]	SHIFT+ALT+N	新規の関数ライブラリを作成する。
[開く] > [テスト]	CTRL+O	既存のテストを開く。
[開く] > [ビジネス/スクリプトコンポーネント]	CTRL+SHIFT+O	既存のビジネス・コンポーネントまたはスクリプト化コンポーネントを開く。
[開く] > [アプリケーション領域]	CTRL+ALT+O	既存のアプリケーション領域を開く。
[開く] > [関数ライブラリ]	SHIFT+ALT+O	既存の関数ライブラリを開く。
[保存]	CTRL+S	アクティブなドキュメントを保存する。
[Zip ファイルにテストをエクスポート]	CTRL+ALT+S	アクティブなドキュメントの zip ファイルを作成する。
[Zip ファイルからテストをインポート]	CTRL+ALT+M	zip ファイルからドキュメントをインポートする。
[スクリプトコンポーネントに変換]	CTRL+ALT+C	ビジネス・コンポーネントをスクリプト・コンポーネントに変換する。
[印刷]	CTRL+P	アクティブなドキュメントを印刷する。

対応するショートカット・キーを押すことで、次の **〔編集〕** メニュー・コマンドを実行できます。

コマンド	ショートカット・キー	機能
[元に戻す]	CTRL+Z	直前に行ったコマンドを元に戻すか、直前に入力したエントリを削除する。
[やり直し]	CTRL+Y	[元に戻す] コマンドで元に戻したことをやり直す。
[切り取り]	CTRL+X	選択した内容をドキュメントから切り取る。
[コピー]	CTRL+C	選択した内容をドキュメントからコピーする。
[貼り付け]	CTRL+V	選択した内容をドキュメントに貼り付ける。
[削除]	DELETE キー	選択した内容をドキュメントから削除する。
[アクション] > [アクション名の変更]	SHIFT+F2	アクション名を変更する。
[ステップのプロパティ] > [オブジェクトのプロパティ]	CTRL+ENTER, ALT+ENTER	選択したオブジェクトの [オブジェクトのプロパティ] ダイアログ・ボックスを表示する。
[検索]	CTRL+F	特定の文字列を検索する。
[置換]	CTRL+H	特定の文字列を検索して置換する。
[移動]	CTRL+G	コンポーネントの特定の行にカーソルを移動する。
[ブックマーク]	CTRL+B	スクリプト内を簡単に移動できるように、スクリプト内にブックマークを作成する。
[詳細設定] > [コメントブロック]	CTRL+M	現在の行または選択した行をコメント・アウトする。
[詳細設定] > [ブロックのコメント削除]	CTRL+SHIFT+M	現在の行または選択した行のコメント書式設定を削除する。

コマンド	ショートカット・キー	機能
[詳細設定] > [関数定義に移動]	ALT+G	選択した関数の定義に移動する。
[詳細設定] > [単語入力候補]	CTRL+SPACE	VBScript メソッドまたはオブジェクトの最初の部分を入力したとき、入力候補を表示する。
[詳細設定] > [引数詳細]	CTRL+SHIFT+SPACE	メソッドの構文を表示する。
[詳細設定] > [スクリプトに "With" を追加]	CTRL+W	[エキスパート ビュー] に表示されているアクションに対して With ステートメントを生成する。
[詳細設定] > ["With" ステートメントを削除]	CTRL+SHIFT+W	[エキスパート ビュー] に表示されているアクションの With ステートメントを、通常の (単独行の) VBScript ステートメントに変換する。

対応するショートカット・キーを押すことで、次の [挿入] メニュー・コマンドを実行できます。

コマンド	ショートカット・キー	機能
[チェックポイント] > [標準チェックポイント]	F12 キー	オブジェクトまたはテーブルの標準チェックポイントを作成する。
[出力値] > [標準出力値]	CTRL+F12	オブジェクトまたはテーブルの標準の出力値を作成する。
[ステップ ジェネレータ]	F7 キー	ステップ・ジェネレータを開く。
[ステップの新規作成]	F8 キー, INSERT キー	キーワード・ビューに新規ステップを挿入する。
[ブロックの後の新規ステップ]	SHIFT+F8	キーワード・ビューで、条件ブロックまたはループ・ブロックの後に新規ステップを挿入する。

第1部・Business Process Testing の概要

対応するショートカット・キーを押すことで、次の「オートメーション」メニュー・コマンドを実行できます。

コマンド	ショートカット・キー	機能
[記録]	F3 キー	記録セッションを開始する。
[実行]	F5 キー	実行セッションを、最初から、またはセッションが一時停止された場所から開始する。
[停止]	F4 キー	記録または実行セッションを停止する。
[ステップから実行]	CTRL+F5	選択したステップから実行セッションを開始する。
[アナログ記録]	SHIFT+ALT+F3	アナログ記録モードで記録を開始する。
[低レベル記録]	CTRL+SHIFT+F3	低レベル記録モードで記録を開始する。

対応するショートカット・キーを押すことで、次の「リソース」メニュー・コマンドを実行できます。

コマンド	ショートカット・キー	機能
[オブジェクトリポジトリ]	CTRL+R	[オブジェクトリポジトリ] ダイアログ・ボックスを開く。

対応するショートカット・キーを押すことで、次の [デバッグ] メニュー・コマンドを実行できます。

コマンド	ショートカット・キー	機能
[ステップイントゥ]	F11 キー	スクリプトの現在のステップだけを実行する。現在の行がメソッドを呼び出した場合、そのメソッドがビューに表示されるが、実行されない。
[ステップオーバー]	F10 キー	スクリプトの現在のステップだけを実行する。現在の行がメソッドを呼び出した場合、そのメソッド全体が実行されるが、ビューには表示されない。
[ステップアウト]	SHIFT+F11	メソッドの最後まで実行してから実行セッションを一時停止する ([ステップイントゥ] を使用してメソッドを実行した後にのみ使用可能)。
[ステップまで実行]	CTRL+F10	現在のステップまで実行する。
[ウォッチに追加]	CTRL+T	選択した項目を [ウォッチ] タブに追加する。
[ブレークポイントの設定 / 解除]	F9 キー	コンポーネントのブレークポイントを設定または削除する。
[ブレークポイントの有効化 / 無効化]	CTRL+F9	コンポーネントのブレークポイントを有効または無効にする。
[すべてのブレークポイントを解除]	CTRL+SHIFT+F9	コンポーネントのすべてのブレークポイントを削除する。

対応するショートカット・キーを押すことで、次の [ツール] メニュー・コマンドを実行できます。


コマンド	ショートカット・キー	機能
[構文チェック]	CTRL+7	アクティブなドキュメントの構文をチェックする。

データ・テーブルの1つ以上のセルを選択しているときに、対応するショートカット・キーを押すことで、次の**「データ テーブル」**メニュー・コマンドを実行できます。

コマンド	ショートカット・キー	機能
[編集] > [切り取り]	CTRL+X	テーブルの選択範囲を切り取り、クリップボードに格納する。
[編集] > [コピー]	CTRL+C	テーブルの選択範囲をコピーし、クリップボードに格納する。
[編集] > [貼り付け]	CTRL+V	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。
[編集] > [クリア] > [内容]	CTRL+DEL	現在の選択範囲から内容を消去する。
[編集] > [挿入]	CTRL+I	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る。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る。
[編集] > [削除]	CTRL+K	現在の選択範囲を削除する。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る。
[編集] > [右方向へコピー]	CTRL+R	選択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する。
[編集] > [縦方向にコピー]	CTRL+D	選択された範囲内で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する。
[編集] > [検索]	CTRL+F	指定されたテキストを含むセルを検索する。テーブルの行ごとまたは列ごとの検索、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に限定した検索ができる。

コマンド	ショートカット・キー	機能
[編集] > [置換]	CTRL+H	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る。テーブルの行ごとまたは列ごとの検索、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に限定した検索ができる。すべて置換することも可能。
[データ] > [再計算]	F9 キー	データ・テーブルの選択データを再計算する。
データ・テーブル・シート間の切り替え	CTRL+PAGE UP/PAGE DOWN	データ・テーブルにフォーカスがあるときに、データ・テーブル・シート間を切り替えます。

ショートカット・キーを使用して、次の特別なオプションを実行できます。

オプション	ショートカット・キー	機能
キーワード・ビューとエキスパート・ビューの切り替え	CTRL+PAGE UP/PAGE DOWN	[キーワード ビュー] と [エキスパート ビュー] を切り替える。
開いているドキュメント間の切り替え	CTRL+TAB	開いているほかのドキュメントのタイプに表示を切り替える。
ショートカット・メニューを開く	SHIFT+F10, またはアプリケーション・キー () を押す (Microsoft ナチュラル・キーボードのみ)	データ・テーブルで選択されているステップ・データ・セルに対応するショートカット・メニューを開く。
ツリーをすべて展開	* (テンキーを使用)	キーワード・ビューのすべてのツリーを展開する。
サブツリーを展開	+ (テンキーを使用)	キーワード・ビューで選択した項目のツリーと、それ以下のすべてのサブツリーを展開する。

オプション	ショートカット・キー	機能
サブツリーを折りたたむ	- (テンキーを使用)	キーワード・ビューで選択した項目のツリーと、それ以下のすべてのサブツリーを折りたたむ。
[項目] または [操作] リストを開く	キーワード・ビューで [項目] または [操作] が選択されているときに SHIFT+F4 または SPACE キー	[項目] または [操作] カラムが選択されているときに、キーワード・ビューで [項目] または [操作] リストが開く。

QuickTest Professional プログラム・フォルダの参照

QuickTest Professional セットアップ・プロセスが完了すると、QuickTest Professional プログラム・フォルダ ([スタート] > [プログラム] > [QuickTest Professional] または [スタート] > [すべてのプログラム] > [QuickTest Professional]) に次の項目が追加されます。

注：更新インストールを実行した場合、または現在のバージョンをインストールする前に QuickTest Professional の旧バージョンをアンインストールした場合には、QuickTest Professional プログラム・フォルダに余計な（無効な）項目が存在することがあります。さらに、QuickTest Professional の外部アドインをインストールしてある場合には、それらのアドインにだけ関係する項目が、プログラム・フォルダに追加されることがあります。

- ▶ **[Documentation]** : 次に示すよく使用されるマニュアルへのリンクを提供します。
- ▶ **[Printer-Friendly Documentation]** : QuickTest のすべてのマニュアルの印刷用バージョン (Adobe Acrobat Reader (PDF) 形式) へのリンクを含んだページが開きます。

- ▶ **[QuickTest Automation Reference]** : 『QuickTest オートメーション・オブジェクト・モデル・リファレンス』が開きます。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 オートメーション・オブジェクト・モデル・リファレンス』には、構文、機能説明、およびオブジェクト、メソッド、プロパティの使用例が記載されています。また、QuickTest の自動スクリプトを記述する際の詳しい概要も含まれます。
- ▶ **[QuickTest Professional Code Samples Plus]** : QuickTest Professional Code Samples Plus ヘルプが開きます。このヘルプには、ブラウザコントロール登録ユーティリティとライセンス検証ユーティリティについてのドキュメント、およびよくある質問に対する回答、ヒント、関数ライブラリのサンプル、およびコードと SDK のサンプルが含まれています。
- ▶ **[QuickTest Professional Help]** : 『QuickTest ユーザーズ・ガイド』、『QuickTest for Business Process Testing ユーザーズ・ガイド』、アドインがインストールされている場合は対応するガイド、『オブジェクト・モデル・リファレンス』（インストールされているアドインに関連するセクションを含む）、および『Microsoft VBScript リファレンス』が含まれる包括的なヘルプ・ファイルが開きます。
- ▶ **[Tutorial]** : QuickTest Professional チュートリアルが開きます。このチュートリアルでは、QuickTest の基本技術、およびアプリケーションのテストを開始する方法について説明します。
- ▶ **[Sample Applications]** : 次に示す、QuickTest でのテストの練習に使用できるサンプル・アプリケーションへのリンクが収められています。
- ▶ **[Flight]** : サンプルのフライト予約 Windows アプリケーションが開きます。このアプリケーションにアクセスするには、任意のユーザ名とパスワード (**mercury**) を入力します。
- ▶ **[Mercury Tours Web site]** : サンプルのフライト予約 Web アプリケーションが開きます。この Web アプリケーションは、QuickTest のチュートリアルで使用します。詳細については、『QuickTest Professional チュートリアル』を参照してください。

- ▶ **[Tools]** : 次に示す、テスト・プロセスを支援するユーティリティとツールが収められています。
- ▶ **[Action Conversion Tool]** : QuickTest Professional を使用して作成したテスト・アクションを、ビジネス・プロセス・テストで使用するスクリプト・コンポーネントに変換できます。詳細については、[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。
- ▶ **[Password Encoder]** : [パスワードエンコーダ] 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、パスワードを暗号化できます。生成された文字列は、メソッドの引数またはデータ・テーブルのパラメータ値として使用できます (テストのみ)。詳細については、168 ページ「メソッド引数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。
- ▶ **[Remote Agent]** : QuickTest リモート・エージェントを有効にします。QuickTest リモート・エージェントは、Quality Center などのリモート・アプリケーションによってテストが実行されるときの QuickTest の振る舞いを決定します。
- ▶ **[Silent Test Runner]** : (テストにのみ関係) [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。このダイアログ・ボックスでは、LoadRunner および Business Availability Center から実行する場合と同じ方法で QuickTest テストを実行できます。詳細については、『**QuickTest Professional 上級機能ユーザーズ・ガイド**』を参照してください。
- ▶ **[Test Batch Runner]** : (テストにのみ関係) [Test Batch Runner] 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、複数のテストを連続して実行するように QuickTest を設定できます。
- ▶ **[Test Results Deletion Tool]** : [テスト結果削除ツール] 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、定義された特定の条件に基づいて、不要な結果をシステムから削除できます。詳細については、524 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。
- ▶ **[License Validation Utility]** : [QTPlus ライセンス検証ユーティリティ] が開きます。このユーティリティを使用して、ライセンス情報の取得および有効化が可能です。詳細については、『**QuickTest Professional Code Samples Plus ヘルプ**』を参照してください。

- ▶ **[Register New Browser Control]** : ブラウザ・コントロール登録ユーティリティが開きます。このユーティリティでは、テストの記録時や実行時に QuickTest Professional が Web オブジェクトを認識できるようブラウザ・コントロール・アプリケーションを登録することができます。詳細については、『**QuickTest Professional Code Samples Plus ヘルプ**』を参照してください。
- ▶ **[QuickTest Professional のアンインストール]** : QuickTest Professional, および組み込みアドインと外部アドインを含むすべてのコンポーネントをアンインストールします。詳細については、『**QuickTest Professional インストール・ガイド**』を参照してください。
- ▶ **[QuickTest Professional]** : QuickTest Professional アプリケーションを開きます。
- ▶ **[Readme]** : [Mercury QuickTest Professional 9.0 最初にお読みください] が開きます。ここでは、QuickTest Professional に関する最新のニュースおよび情報が提供されます。
- ▶ **[Test Results Viewer]** : [テスト結果] ウィンドウが開きます。このウィンドウでは、コンポーネントまたはビジネス・プロセス・テストを選択して、実行セッション中に実行されたステップに関する情報を表示できます。詳細については、503 ページ「[テスト結果] ウィンドウについて」を参照してください。
- ▶ **[更新を確認する]** : QuickTest Professional の更新がないかオンラインで調べます。ダウンロードしてインストール（任意）する更新を選択できます。詳細については、『**QuickTest Professional ユーザーズ・ガイド**』を参照してください。

製品情報の表示

コンピュータにインストールされている QuickTest アドインとパッチに関する情報、およびオペレーティング・システムに関する情報を表示できます。この情報は、トラブルシューティングや Mercury カスタマー・サポートへの問い合わせ時に役立ちます。

製品情報を表示するには、次の手順を実行します。

- 1 QuickTest で、[ヘルプ] > [QuickTest Professional のバージョン情報] を選択します。[QuickTest Professional 9.0 のバージョン情報] ウィンドウが開きます。



[QuickTest Professional 9.0 のバージョン情報] ウィンドウには次の情報が表示されます。

- ▶ コンピュータにインストールされている QuickTest のバージョン，ビルド番号，製品 ID 番号。
- ▶ コンピュータにインストールされている QuickTest アドインのリスト。アドイン名の横に表示されるチェック・マークは，そのアドインが現在ロードされていることを示します。QuickTest アドインの詳細については，付録 A 「QuickTest アドインの使用法」を参照してください。

ヒント：コンピュータにインストールされている QuickTest Professional ライセンスの詳細を表示する場合，またはライセンスを変更する場合は，[**ライセンス**] ボタンをクリックします。詳細については，『**QuickTest Professional インストール・ガイド**』を参照してください。



- 2 コンピュータにインストールされている QuickTest Professional 製品の詳細情報を表示するには、[製品情報] ボタンをクリックします。[製品情報] ウィンドウが開きます。

製品情報

製品名:	QuickTest Professional
製品バージョン:	9.0
製品 ID:	QTPRPID9.0/03
製品ビルド:	2462
オペレーティング システム:	Microsoft Windows 2000 Service Pack 4 (Build 2195)
Internet Explorer のバージョン:	6.0.2800.1106
Quality Center 接続機能:	9.1.0.7030

アドイン情報:

名前	バージョン
ActiveX	9.0.0.0
Visual Basic	9.0.0.0
Web	9.0.0.0

パッチ情報:

名前	最初にお読みください
QuickTest Professional Japanese User Interface Pack	QuickTest Professional Japanese User Interface Pack Readme

Copyright © Mercury Interactive Corporation 1992-2006

MERCURY™

[製品情報] ウィンドウには次の情報が表示されます。

- ▶ コンピュータにインストールされている QuickTest Professional のバージョン、製品 ID 番号、ビルド番号。
- ▶ **オペレーティング・システム**：コンピュータにインストールされているオペレーティング・システムのバージョン。
- ▶ **Internet Explorer のバージョン**：コンピュータにインストールされている Microsoft Internet Explorer のバージョン。
- ▶ **Quality Center 接続機能**：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。

- ▶ **アドイン情報**：コンピュータにインストールされている QuickTest アドイン，およびそのバージョンとビルド番号。
- ▶ **パッチ情報**：コンピュータにインストールされている QuickTest パッチの名前，およびそのパッチの **Readme** ファイルへのリンク。

第2部

アプリケーション領域と
コンポーネントを使用した作業

第 3 章

アプリケーション領域の使用

アプリケーション領域は、ビジネス・コンポーネントの作成に必要なリソースと設定をすべて提供します。アプリケーション領域の設定やそれらの設定に加えた変更は、アプリケーション領域が関連付けられているすべてのビジネス・コンポーネントに自動的に適用されます。

注：旧バージョンの QuickTest Professional では、アプリケーション領域はビジネス・コンポーネント・テンプレートと呼ばれていました。旧バージョンでは、すべてのビジネス・コンポーネントで同じテンプレートを使用していました。現在の QuickTest では、複数のアプリケーション領域を作成して、それをアプリケーションの各領域に合わせてカスタマイズできます。

本章では、次の項目について説明します。

- ▶ アプリケーション領域の使用について
- ▶ アプリケーション領域の作成
- ▶ アプリケーション領域を開く
- ▶ 一般設定の定義
- ▶ 関数ライブラリの管理
- ▶ 共有オブジェクト・リポジトリの管理
- ▶ キーワードの管理
- ▶ 追加設定の定義
- ▶ アプリケーション領域の保存
- ▶ アプリケーション領域の削除

アプリケーション領域の使用について

アプリケーションの特定の範囲をテストする1組のコンポーネントを作成する場合、通常は数多くの同じテスト・オブジェクト、キーワード、テスト設定、および関数ライブラリや回復シナリオなどのその他のテスト・リソースを使用する必要があります。これらのファイルや設定は、アプリケーション領域に定義します。アプリケーション領域によって、アプリケーションの特定の部分のテストに関連するすべての要素を1か所で保守できるようになります。

アプリケーション領域は、ビジネス・コンポーネントの内容の作成に必要な設定とリソースの集まりです。リソースには共有オブジェクト・リポジトリや関数ライブラリを含めることができます。共有オブジェクト・リポジトリには、コンポーネントのテスト対象アプリケーション内のテスト・オブジェクトが収められます。関数ライブラリには、そのアプリケーションを対象としてコンポーネントによって実行されるユーザ定義の操作が収められます。コンポーネントは、関連付けられているアプリケーション領域に定義されているすべてのリソースと設定に自動的に結び付けられます。

必要に応じて、アプリケーション領域はいくつでも作成できます。たとえば、アプリケーション内の Web ページ、モジュール、ウィンドウ、またはダイアログ・ボックスのそれぞれについて、アプリケーション領域を1つずつ作成することもできます。また、小さなアプリケーションでは、1つのアプリケーション領域で十分な場合もあります。コンポーネントにはそれぞれ1つのアプリケーション領域のみを関連付けることができます。

注：アプリケーション領域を使用するには、コンポーネントの変更、およびステップの追加、変更、削除の権限が必要です。4つのすべての権限が必要です。1つでも割り当てられていなければ、アプリケーション領域は読み取り専用形式でのみ開けます。ビジネス・コンポーネント・モジュール内の権限設定の詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。

アプリケーション領域を作成する前に、そのアプリケーション領域を使ってビジネス・コンポーネントを作成する各分野のエキスパートの要件を検討します。例を次に示します。

- ▶ 各分野のエキスパートはどのようなテスト・オブジェクトを必要としていますか。
- ▶ 誰にでも意味がよくわかるようにするには、テスト・オブジェクトやほかの項目の名前をどのように変更しますか。
- ▶ 必要な操作をすべて提供するためには、どのようなユーザ定義関数を追加できますか。

必要な操作を確実に提供するために、アプリケーション領域を作成する前に、関数ライブラリを **Quality Center** プロジェクトに保存することをお勧めします (アプリケーション領域はいつでも更新できます)。**QuickTest** は、アプリケーション領域に関連付けることのできる一連の定義済みリソース・ファイル (関数ライブラリや回復シナリオ・ファイルなど) も提供します。サンプルの関数ライブラリの一部は、標準設定ですべての新規のアプリケーション領域に関連付けられます。これらのサンプル・ファイルは、**Quality Center** プロジェクトのテストの計画モジュールの **Subject/BPT Resources** に含まれます。

コンポーネントで使用するアプリケーション領域を作成するときは、次の作業を実行する必要があります。

- ▶ アプリケーション領域の詳細な説明を提供する
- ▶ **QuickTest Professional** アドインへの関連付けを指定する
- ▶ 必要な関数ライブラリを関連付ける
- ▶ 必要な共有オブジェクト・リポジトリを関連付ける
- ▶ どのキーワードをコンポーネント・ステップの作成時に表示して各分野のエキスパートによる使用を可能にするかを指定する
- ▶ アプリケーション領域が関連付けられているコンポーネントによる記録と実行の対象にできる **Windows** ベース・アプリケーションを指定する
- ▶ 必要な回復シナリオを関連付け、それらの設定を定義する
- ▶ アプリケーション領域を保存する

アプリケーション領域を保存するときには、わかりやすい名前と明確な説明を付けて保存してください。各分野のエキスパートが新規のビジネス・コンポーネントを作成するときにアプリケーション領域の用途を判断するには、この名前と説明だけが手がかりとなります。たとえば、アプリケーション領域がログイン・ダイアログ・ボックスをテストするコンポーネント用である場合は、「LoginDialog」というように命名します。

アプリケーション領域を作成した後、そのアプリケーション領域を使ってビジネス・コンポーネントを作成できるようになったことを各分野のエキスパートに通知することができます（必要であれば、各分野のエキスパートは、アプリケーション領域が使用できるようになる前にコンポーネントの作成を開始し、アプリケーション領域を作成した後にアプリケーション領域をコンポーネントと関連付けることができます）。

アプリケーション領域内のリソースや設定を変更した場合、それらの変更は、変更を加えたアプリケーション領域が関連付けられているすべてのビジネス・コンポーネントに自動的に反映されます。

リソースをコンポーネント・ステップで使用していて、後でそれらのリソースを変更した場合、コンポーネントが正しく実行しないことがあります。たとえば、コンポーネントが **MyRepository.tsr** 共有オブジェクト・リポジトリからのテスト・オブジェクトを使用しているときに、その共有オブジェクト・リポジトリをアプリケーション領域から削除すると、オブジェクト・リポジトリがアプリケーション領域に含まれなくなるため、コンポーネントは必要なテスト・オブジェクトにアクセスできなくなります。

このため、アプリケーション領域に変更を加えるときは、アプリケーション領域が関連付けられているビジネス・コンポーネントに悪影響がないことを確認することをお勧めします。

ヒント：コンポーネントには異なるアプリケーション領域をいつでも関連付けることができます。詳細については、138 ページ「コンポーネントに関連付けられているアプリケーション領域の変更」を参照してください。

アプリケーション領域の作成

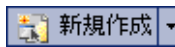
新しいアプリケーション領域を作成するときは、新しいビジネス・コンポーネントの作成に必要なアプリケーション領域の設定とリソースをすべて定義します。

注：アプリケーション領域を作成するには、まず、アプリケーション領域の保存先となる Quality Center プロジェクトに接続する必要があります。これは、各分野のエキスペートがビジネス・コンポーネントを定義しビジネス・プロセス・テストを作成するために使用することになる Quality Center プロジェクトです。詳細については、20 ページ「Quality Center プロジェクトへの接続」を参照してください。

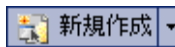
アプリケーション領域を作成するには、次の手順を実行します。

1 次の手順のいずれかを実行します。

▶ **[ファイル]** > **[新規作成]** > **[アプリケーション領域]** を選択します。



▶ **[新規作成]** ボタンの下矢印をクリックし、**[アプリケーション領域]** を選択します。







ヒント：すでにアプリケーション領域が開いている場合は、**[新規作成]** ボタンをクリックすると新規のアプリケーション領域が開きます。

アプリケーション領域ウィンドウにはいくつかの表示枠があり、そのアプリケーション領域に関連付けられているビジネス・コンポーネントに使用させる設定とリソース・ファイルを指定できます。コンポーネントをアプリケーション領域に関連付けると、当該コンポーネントは自動的に設定とリソース・ファイルに結び付けられます。

ここで、アプリケーション領域の設定を指定してリソースを定義できます。各表示枠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。

設定とリソースを定義したら、アプリケーション領域を保存できます。詳細については、116 ページ「アプリケーション領域の保存」を参照してください。

アプリケーション領域には次の各表示枠があり、サイドバーの該当するボタンをクリックしてアクセスできます。

表示枠	内容
<p>一般</p> 	<p>アプリケーション領域の説明を定義し、アプリケーション領域に関連付けるアドインを指定できます。アプリケーション領域が関連付けられているコンポーネントが記録と実行の対象にできる Windows ベース・アプリケーションを指定したり、ブラウザのタイムアウト期間を設定したりできます。あるいは、実行セッション中に予期しないイベントやエラーが発生したときに、アプリケーション領域が関連付けられているコンポーネントがどのように回復するのかを指定する、回復シナリオを定義することもできます。詳細については、83 ページ「一般設定の定義」を参照してください。</p>
<p>関数ライブラリ</p> 	<p>関数ライブラリをアプリケーション領域に関連付け、優先順位を付けることができます。関連付ける関数ライブラリの作成および変更も行えます。詳細については、89 ページ「関数ライブラリの管理」を参照してください。</p>
<p>オブジェクトリポジトリ</p> 	<p>共有オブジェクト・リポジトリをアプリケーション領域に関連付け、優先順位を付けることができます。関連付けるオブジェクト・リポジトリの作成および変更も行えます。詳細については、94 ページ「共有オブジェクト・リポジトリの管理」を参照してください。</p>
<p>キーワード</p> 	<p>コンポーネントの作成時に各分野のエキスパートが使用できる組み込みキーワードおよびユーザ定義キーワード（操作）を決定できます。詳細については、101 ページ「キーワードの管理」を参照してください。</p>

アプリケーション領域を開く

アプリケーション領域を保存した後、表示や変更のために開くことができます。たとえば、回復シナリオを更新したり、ユーザ定義関数が収められている関数ライブラリをアプリケーション領域に追加したりできます。

注：

アプリケーション領域を開くには、まず、アプリケーション領域が保存されている Quality Center プロジェクトに接続する必要があります。詳細については、20 ページ「Quality Center プロジェクトへの接続」を参照してください。

以前のバージョンの QuickTest を実行しているコンピュータで、それ以降のバージョンの QuickTest を使用して作成したアプリケーション領域を開くことはできません。たとえば、QuickTest 8.2 を実行しているコンピュータで、QuickTest 9.0 で作成したアプリケーション領域を開くことはできません。

アプリケーション領域を開くには、次の手順を実行します。

- 1 現在の Quality Center プロジェクトに接続しているアプリケーション領域を表示します。

▶ [ファイル] > [開く] > [アプリケーション領域] を選択します。

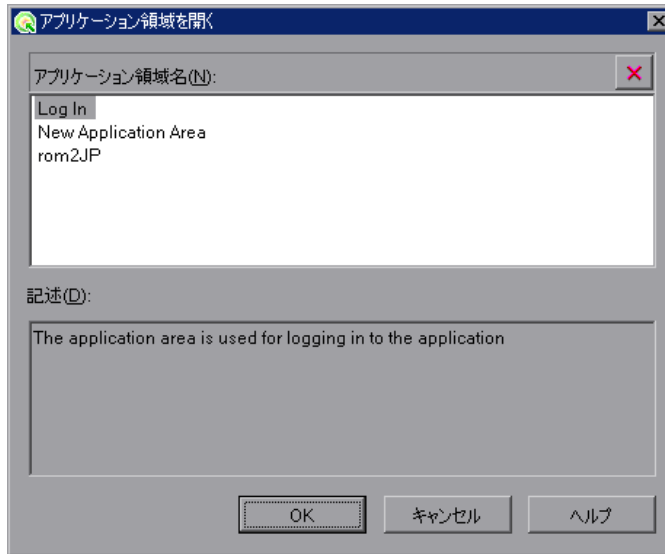


▶ [開く] ボタンの下矢印をクリックし、[アプリケーション領域] を選択します。



ヒント：別のアプリケーション領域がすでに開いている場合は、[開く] ボタンをクリックし、必要なアプリケーション領域を選択できます。

[アプリケーション領域を開く] ダイアログ・ボックスが開き、定義されたアプリケーション領域のリストが表示されます。アプリケーション領域を選択するとその説明が表示されます。



- 2 アプリケーション領域を選択して [OK] をクリックします。選択したアプリケーション領域が開きます。

ここでアプリケーション領域の設定を表示および変更できます。詳細については、次を参照してください。

- ▶ 83 ページ「一般設定の定義」
- ▶ 89 ページ「関数ライブラリの管理」
- ▶ 94 ページ「共有オブジェクト・リポジトリの管理」
- ▶ 101 ページ「キーワードの管理」
- ▶ 105 ページ「追加設定の定義」

注：このダイアログ・ボックスからアプリケーション領域を削除することもできます（他のコンポーネントに関連付けられていない限り）。詳細については、118 ページ「アプリケーション領域の削除」を参照してください。

一般設定の定義

一般表示枠では、説明や関連付けられているアドインなど、アプリケーション領域に関する一般情報の表示と定義ができます。各分野のエキスパートが特定のビジネス・コンポーネントのためのアプリケーション領域を選択するとき判断の手がかりとなるのは名前と説明だけなので、アプリケーション領域の明確な説明を含めることが重要です。



一般表示枠には、次の項目があります。

項目	詳細
【名前】	アプリケーション領域の名前を示します。アプリケーション領域の名前は、保存するときに指定します。詳細については、116ページ「アプリケーション領域の保存」を参照してください。
【作成者】	アプリケーション領域を作成した人物の Windows ユーザ名を示します。

項目	詳細
[場所]	<p>アプリケーション領域の Quality Center パス名とファイル名を示します。アプリケーション領域がまだ保存されていない場合は、「保存されていません」と表示され、[アプリケーション領域] ダイアログ・ボックスのタイトル・バーにはアスタリスクが表示されます。</p>
[詳細]	<p>アプリケーション領域に対して指定されている詳細説明を示します。</p> <p>この必須フィールドにはアプリケーション領域の明確な説明を含めることが重要です。これは、各分野のエキスパートが Quality Center で新しいコンポーネントを作成するときに、アプリケーション領域の [名前] フィールドと [詳細] フィールドに基づいてどのアプリケーション領域を選択するかを決めるからです。詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。</p> <p>説明は必要に応じて更新できます。たとえば、アプリケーション領域を作成してまだ定義を完了していないときに、そのことを [詳細] フィールドに記述しておきます。後で、アプリケーション領域が完成してから [詳細] フィールドの内容を更新できます。</p> <p>注：一般表示枠で説明を入力しなかった場合は、そのアプリケーション領域を保存するときに説明の入力を求められます。詳細については、116 ページ「アプリケーション領域の保存」を参照してください。</p>
[関連アドイン]	<p>アプリケーション領域に関連付けられているアドインが表示されます。関連付けられているアドインとは、ビジネス・コンポーネントがアクセスされたときに QuickTest によって読み込まれるアドインです。</p> <p>注：ビジネス・プロセス・テストを実行すると、テスト内の最初のコンポーネントに関連付けられているアドインが QuickTest によって読み込まれます。したがって、必要な QuickTest アドインはすべて、ビジネス・プロセス・テストの最初のコンポーネントに対応するアプリケーション領域に確実に関連付けておくことが重要です。</p> <p>詳細については、86 ページ「コンポーネントへのアドインの関連付け」を参照してください。</p>

項目	詳細
[追加設定] ボタン	<p>[アプリケーション領域の設定] ダイアログ・ボックスが開きます (105 ページを参照)。このダイアログ・ボックスには複数のタブがあります。</p> <ul style="list-style-type: none"> • [アプリケーション] : アプリケーション領域が関連付けられているコンポーネントが記録と実行の対象にできる Windows ベース・アプリケーションを指定できます。詳細については、106 ページ「アプリケーション領域のアプリケーション設定の定義」を参照してください。 • [Web] : ブラウザの操作のタイムアウト期限を設定できます。詳細については、111 ページ「アプリケーション領域の Web 設定の定義」を参照してください。 • [回復] : アプリケーション領域に関連付けられているコンポーネントが、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回復する方法を定義できます。詳細については、112 ページ「アプリケーション領域の回復シナリオ設定の定義」を参照してください。 <p>Java や SAP などの外部アドインが読み込まれていれば、それらに対応するタブが [アプリケーション領域の設定] ダイアログ・ボックスに含まれることもあります。外部アドインに関連するタブの詳細については、対応する QuickTest アドインのドキュメントを参照してください。</p>
[変更] ボタン	<p>[関連アドインの変更] 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、アドインをコンポーネントに関連付けたり、関連付けを削除したりできます。変更を有効にするために、QuickTest を再起動しなければならないことがあります。詳細については、86 ページ「コンポーネントへのアドインの関連付け」を参照してください。</p>

注 : アプリケーション領域を開いたときに一般表示枠の全体が表示されない場合は、表示枠のサイズを変更できます。たとえば、情報表示枠が関連アドインの下の領域を占めている場合は、情報表示枠のサイズを変更できます。

コンポーネン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。

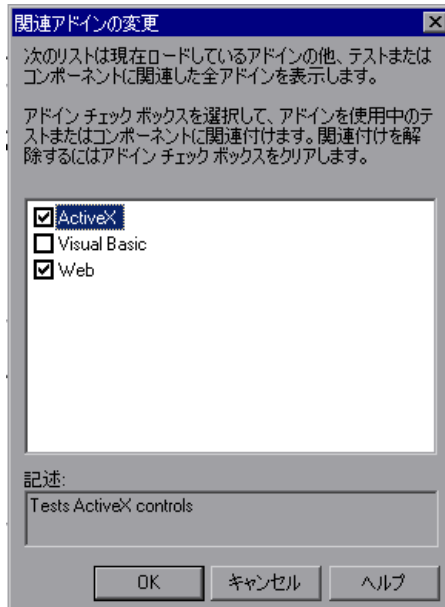
アドインをアプリケーション領域に関連付けると、QuickTest はそのアプリケーション領域に関連付けられているコンポーネントが開くたびに関連アドインが読み込まれているかどうかをチェックします。新しいコンポーネントを作成するとき、コンポーネントのアプリケーション領域に定義されているアドインが当該コンポーネントに関連付けられます。

コンポーネントを開いたときに、関連アドインが現在読み込まれていない場合や、コンポーネントに（アプリケーション領域を通じて）現在関連付けられていないアドインを読み込んだ場合は、QuickTest がそのことを通知します。この処理によって、現在開いているコンポーネントで使用する予定のアドインを必須アドインとして関連アドイン・リストに忘れずに追加できるようになります。その結果、読み込まれていないアドインのために実行セッションが失敗することがなくなります。

各分野のエキスパートが Quality Center でビジネス・プロセス・テストを開くと、そのビジネス・プロセス・テストの最初のコンポーネントに関連付けられている QuickTest Professional アドインが自動的に読み込まれます。ビジネス・プロセス・テスト内のほかのコンポーネントに関連付けられているアドインは読み込まれません。したがって、必要な QuickTest アドインはすべて、ビジネス・プロセス・テストの最初のコンポーネントに対応するアプリケーション領域に関連付けておくことが重要です。

関連アドインの変更

アドインをアプリケーション領域（およびアプリケーション領域に関連付けられているコンポーネント）に関連付けたり、関連付けを解除したりするには、一般表示枠の「**変更**」ボタンをクリックして、「関連アドインの変更」ダイアログ・ボックスを開きます。



このダイアログ・ボックスには、アプリケーション領域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。アプリケーション領域に関連付けられているものの、現在読み込まれていないアドインは、淡色表示されます。

アプリケーション領域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、アプリケーション領域に関連付けたくないアドインのチェック・ボックスをクリアしたりすることができます。

上の例では次のようになっています。

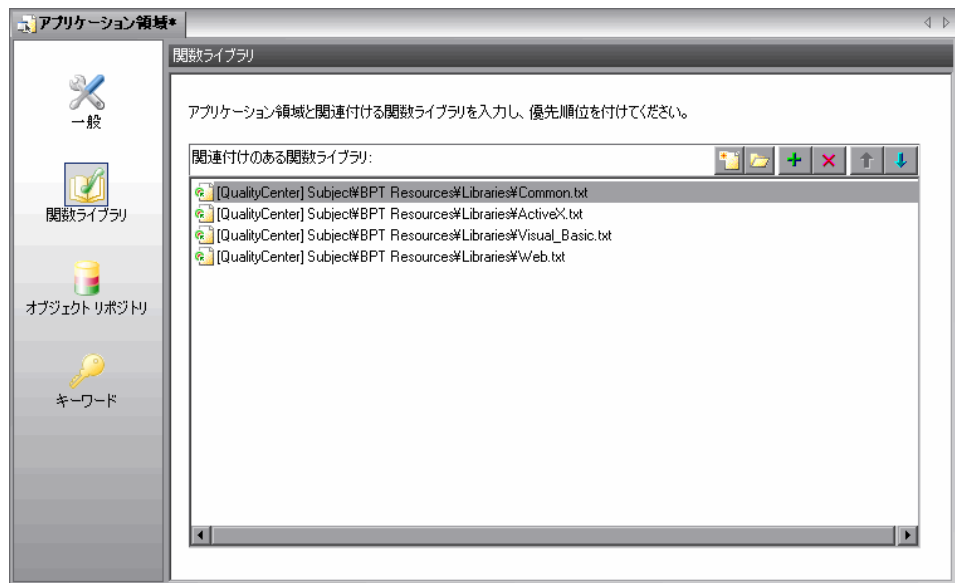
- ▶ Web アドインが読み込まれており、アプリケーション領域に関連付けられている。
- ▶ ActiveX アドインは読み込まれているが、アプリケーション領域に関連付けられていない。
- ▶ Visual Basic アドインはアプリケーション領域に関連付けられているが、読み込まれていない。

注：現在読み込まれていない特定のアドインをアプリケーション領域に関連付けるには、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、**[起動時にアドイン マネージャを表示する]** を選択します。詳細については、194 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。

関数ライブラリの管理

関数ライブラリ表示枠では、QuickTest の関数ライブラリや VBScript の関数ライブラリなどの関数ライブラリ・ファイル、およびテキスト・ファイルを、アプリケーション領域に関連付けることができます。関数ライブラリをアプリケーション領域に関連付けることで、ユーザ定義キーワードとして追加機能を提供できます。これらのキーワードはビジネス・コンポーネントの作成時に使用できます。







関連付けた関数ライブラリはすべて Quality Center プロジェクトに保存する必要があります。




関数ライブラリ表示枠では、現在アプリケーション領域に関連付けられている関数ライブラリのリストが表示され、追加の関数ライブラリを関連付けることができるほか、これらのファイルの変更、削除、優先順位設定が可能です。関数ライブラリを Quality Center プロジェクトに格納する限り、既存の関数ライブラリを追加したり、新規に作成したりできます。

注：QuickTest には、あらかじめ定義されている関数を含むサンプル関数ライブラリが用意されています。標準設定では、これらのファイルはすべての新しいアプリケーション領域に関連付けられます。標準設定の関数ライブラリは、Quality Center プロジェクトの **Subject¥BPT Resources¥Libraries** に格納されています。関数ライブラリでのユーザ定義関数の作成の詳細については、第 21 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。

次のボタンを使用して、コンポーネントに関連付けられている関数ライブラリの追加、変更、削除、および優先順位設定ができます。

ボタン	詳細
	関数ライブラリを新規に作成し、Quality Center プロジェクトに保存して、リストに追加できます。
	表示または編集のために、選択した関数ライブラリを関数ライブラリ・ウィンドウで開きます。他の QuickTest ユーザまたは Quality Center ユーザが現在使用している関数ライブラリはロックされ、読み取り専用モードでのみ開くことができます。詳細については、第 21 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。
	Quality Center プロジェクトのテスト計画ツリーを参照し、アプリケーション領域に関連付ける既存の関数ライブラリを選択できます。詳細については、91 ページ「アプリケーション領域への既存のライブラリの追加」を参照してください。
	選択した関数ライブラリをアプリケーション領域から削除します。
	選択した関数ライブラリをリストの中で上に移動することで、コンポーネントの実行セッションにおける優先順位を上げます。
	選択した関数ライブラリをリストの中で下に移動することで、コンポーネントの実行セッションにおける優先順位を下げます。

注：関連付けられている関数ライブラリを右クリックして、**[開く]** を選択して開くことができます。また、**[削除]** を選択して、アプリケーション領域との関連付けを解除できます。

関連付けられている関数ライブラリが Quality Center プロジェクトから削除されたなどの理由で見つからない場合、QuickTest はリスト内の関数ライブラリの左側に **[見つからない関数ライブラリ]** アイコン  を表示してこのことを知らせます。見つからない関数ライブラリについては、その関数ライブラリを右クリックして、**[探す]** をクリックして必要な関数ライブラリを参照するか、**[削除]** をクリックして見つからない関数ライブラリの関連付けを削除します。

アプリケーション領域への既存のライブラリの追加

既存の関数ライブラリをアプリケーション領域に追加できます。これにより、アプリケーション領域に関連付けられているすべてのビジネス・コンポーネントから、それらの関数ライブラリにキーワードとして定義されている関数にアクセスできるようになります。

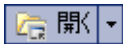
既存の関数ライブラリをアプリケーション領域に関連付けるには、次の手順を実行します。

- 1 QuickTest で、アプリケーション領域をまだ開いていなければ開きます。

▶ **[ファイル]** > **[開く]** > **[アプリケーション領域]** を選択します。



▶ **[開く]** ボタンの下矢印をクリックし、**[アプリケーション領域]** を選択します。



ヒント：別のアプリケーション領域がすでに開いている場合は、**[開く]** ボタンをクリックし、必要なアプリケーション領域を選択できます。

- 2 サイドバーの **[関数ライブラリ]** をクリックします。アプリケーション領域に現在関連付けられている関数ライブラリのリストが関数ライブラリ表示枠に表示されます。
- 3 **[関数ライブラリの追加]** ボタンをクリックします。空の行と参照ボタンがリストに追加されます。



- 4 参照ボタンをクリックします。[関数ライブラリの追加] ダイアログ・ボックスが開きます。

[関数ライブラリの追加] ダイアログ・ボックスに、現在の Quality Center プロジェクトのテスト計画ツリーが表示されます。



- 5 ツリー内の該当する項目を選択し、その項目に添付されている関数ライブラリを表示します。次に、アプリケーション領域に関連付ける関数ライブラリを選択します。名前が [添付名] ボックスに表示されます。
- 6 [OK] をクリックします。[関数ライブラリの追加] ダイアログ・ボックスが閉じ、選択したファイルがアプリケーション領域の関数ライブラリ表示枠に表示されます。関数ライブラリに構文エラーが含まれていると、構文エラーのため、テストが失敗するというメッセージが開きます。

関数ライブラリの新規作成

アプリケーション領域の関数ライブラリ表示枠から直接、関数ライブラリを新規作成し、アプリケーション領域に自動的に関連付けることができます。

Quality Center プロジェクトで関数ライブラリを新規作成するには、次の手順を実行します。

- 1 QuickTest で、アプリケーション領域をまだ開いていなければ開きます。
 - ▶ [ファイル] > [開く] > [アプリケーション領域] を選択します。



- ▶ **「開く」** ボタンの下矢印をクリックし、**「アプリケーション領域」** を選択します。



ヒント：別のアプリケーション領域がすでに開いている場合は、**「開く」** ボタン をクリックし、必要なアプリケーション領域を選択することもできます。

- 2 サイドバーの **「関数ライブラリ」** をクリックします。現在アプリケーション領域に関連付けられている関数ライブラリのリストが関数ライブラリ表示枠に表示されます。



- 3 関数ライブラリ表示枠で、**「関数ライブラリの作成」** ボタンをクリックします。**「関数ライブラリの追加」** ダイアログ・ボックスが開きます。
- 4 テスト計画ツリーで、関数ライブラリを格納するフォルダに移動します。
- 5 **「添付名」** ボックスに、関数ライブラリの名前を入力し、**「OK」** をクリックします。テスト計画ツリー内で選択した場所に、新しい空の関数ライブラリが追加され、関数ライブラリ表示枠に一覧表示されます。

注：標準設定では、関数ライブラリは拡張子を指定しなければ、**.qfl** ファイルとして QuickTest に作成されます。必要ならば、**.txt** ファイルや **.vbs** ファイルも作成できます。



- 6 新しい関数ライブラリに内容を追加する場合や、QuickTest からファイルを直接変更する場合は、関数ライブラリ表示枠でファイルを選択し、**「関数ライブラリを開く」** ボタンをクリックするか、リスト内の関数ライブラリをダブルクリックします。**.** ファイルが関数ライブラリ・ウィンドウに開くので、必要に応じて編集できます。変更を保存するには、ファイルを閉じ、確認を求められたら **「はい」** をクリックします。
- 7 関数ライブラリ名を変更したい場合は、ライブラリを2回クリックするか、選択して F2 キーを押します。

関数ライブラリの編集の詳細については、第21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。

共有オブジェクト・リポジトリの管理

共有オブジェクト・リポジトリには、ビジネス・コンポーネントのステップの作成時に使用できるテスト・オブジェクトがすべて保存されます。共有オブジェクト・リポジトリをアプリケーション領域に関連付けると、そのアプリケーション領域に関連付けられているすべてのコンポーネントからその共有オブジェクト・リポジトリにアクセスできるようになります。



オブジェクト・リポジトリ表示枠では、アプリケーション領域に現在関連付けられている共有オブジェクト・リポジトリのリストが表示され、追加のオブジェクト・リポジトリを関連付けることができるほか、これらのファイルの変更、削除、優先順位設定が可能です。オブジェクト・リポジトリを **Quality Center** プロジェクトに格納する限り、既存のオブジェクト・リポジトリを追加したり、新規に作成したりできます。

アプリケーション内のオブジェクトを学習するか、オブジェクト・リポジトリ・マネージャを使用してテスト・オブジェクトを手作業で追加することで、テスト・オブジェクトをこの共有オブジェクト・リポジトリに追加できます。共有オブジェクト・リポジトリ内のテスト・オブジェクトの管理については、第15章「オブジェクト・リポジトリの管理」を参照してください。

注：QuickTestには標準設定の共有オブジェクト・リポジトリが用意されていますが（**Subject/BPT Resources/Object Repositories** フォルダにあります），それを使用しないよう強くお勧めします。アプリケーション領域または特定のコンポーネントにこの標準設定の共有オブジェクト・リポジトリを関連付けた場合，この共有オブジェクト・リポジトリを使用するコンポーネントはどれも正常に動作しなくなる可能性があります。

すでにテスト・オブジェクトが含まれている既存の共有オブジェクト・リポジトリ・ファイルを使用することも，新規の共有オブジェクト・リポジトリ・ファイルを作成することもできます。これらの共有オブジェクト・リポジトリを参照するアプリケーション領域に関連付けられているビジネス・コンポーネントはすべて，これらの共有オブジェクト・リポジトリ・ファイルにアクセスします。詳細については，97 ページ「共有オブジェクト・リポジトリの新規作成」を参照してください。

テスト・オブジェクトを共有オブジェクト・リポジトリに追加した後，オートメーション・エンジニアと各分野のエキスパートはそれらのテスト・オブジェクトを使用してステップをビジネス・コンポーネントに追加できます。詳細については，156 ページ「ステップの項目の選択」を参照してください。







各分野のエキスパートは，ビジネス・コンポーネントのステップを定義するときに，各種のテスト・オブジェクトを区別できる必要があります。したがって，すべてのテスト・オブジェクト名にわかりやすい名前を付けることが重要です。QuickTest がテスト・オブジェクトを保存するときに自動的に割り当てた名前は，変更できます。たとえば，あるテスト・オブジェクトに「Edit」という名前が自動的に付けられた場合，このエディット・ボックスにユーザがユーザ名を入力する必要がある場合，それを示すために名前を「UserName」に変更することができます。

コンテナ・オブジェクトに対しては，コンテキストを指定することをお勧めします。たとえば，いくつかの確認メッセージ・ボックスがある場合，1つを「Login > Confirm」，もう1つを「ChangePassword > Confirm」，さらにもう1つを「BillingInfo > Confirm」と命名できます。

オブジェクトの名前を変更すると、QuickTest のキーワード・ビューと Quality Center ビジネス・コンポーネント・モジュールの [ステップ] タブに出現する当該オブジェクトの名前がすべて自動的に更新されます (古いテスト・オブジェクト名を使って作成されたステップでも同様です)。同じ共有オブジェクト・リポジトリを使用していて、変更されたオブジェクトが 1 回以上出現する別のコンポーネントを開くと、そのコンポーネント内の名前が更新されます。これにはしばらく時間がかかる場合があります。

テスト・オブジェクトの名前の変更の詳細については、327 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。

次のボタンを使用して、アプリケーション領域 (およびアプリケーション領域に関連付けられているコンポーネント) に関連付けられているオブジェクト・リポジトリの追加, 変更, 削除, および優先順位設定ができます。

ボタン	詳細
	オブジェクト・リポジトリを新規に作成し、Quality Center に保存して、リストに追加できます。
	表示または編集のために、選択したオブジェクト・リポジトリをオブジェクト・リポジトリ・マネージャで開きます。現在ロックされているオブジェクト・リポジトリは読み取り専用形式で開きます。オブジェクト・リポジトリ・マネージャの詳細については、第 15 章「オブジェクト・リポジトリの管理」を参照してください。
	Quality Center プロジェクトのテスト計画ツリーを参照し、アプリケーション領域に関連付ける既存のオブジェクト・リポジトリを選択できます。詳細については、98 ページ「アプリケーション領域への既存の共有オブジェクト・リポジトリの追加」を参照してください。
	選択したオブジェクト・リポジトリをアプリケーション領域から削除します。
	選択したオブジェクト・リポジトリをリストの中で上に移動することで、コンポーネントの実行セッションにおける優先順位を上げます。
	選択したオブジェクト・リポジトリをリストの中で下に移動することで、コンポーネントの実行セッションにおける優先順位を下げます。

注：共有オブジェクト・リポジトリを右クリックし **〔開く〕** を選択して、オブジェクト・リポジトリ・マネージャ内で開くことができます。**〔削除〕** を選択すると、アプリケーション領域との関連付けが解除されます。

共有オブジェクト・リポジトリが見つからない場合は、アプリケーション領域を開いたときにその名前とパスが QuickTest によって欠落リソース表示枠に表示されます。見つからない共有オブジェクト・リポジトリについては、関連付けられているオブジェクト・リポジトリのリストでその共有オブジェクト・リポジトリを右クリックし、**〔探す〕** を選択して必要な共有オブジェクト・リポジトリの場所を探します。または、**〔削除〕** を選択して、共有オブジェクト・リポジトリとの関連付けを解除します。

共有オブジェクト・リポジトリの新規作成

各分野のエキスパートがアプリケーションからテスト・オブジェクトにアクセスできるようにするには、そのテスト・オブジェクトを Quality Center プロジェクト内の共有オブジェクト・リポジトリに保存する必要があります。アプリケーション領域のオブジェクト・リポジトリ表示枠から直接、共有オブジェクト・リポジトリを新規に作成し、アプリケーション領域に自動的に関連付けることができます。

Quality Center プロジェクトに共有オブジェクト・リポジトリを新規に作成するには、次の手順を実行します。

- 1 QuickTest で、アプリケーション領域をまだ開いていなければ開きます。

▶ **〔ファイル〕** > **〔開く〕** > **〔アプリケーション領域〕** を選択します。



▶ **〔開く〕** ボタンの下矢印をクリックし、**〔アプリケーション領域〕** を選択します。



ヒント：別のアプリケーション領域がすでに開いている場合は、**〔開く〕** ボタンをクリックし、必要なアプリケーション領域を選択することもできます。

- 2 サイドバーの [**オブジェクトリポジトリ**] をクリックします。アプリケーション領域に現在関連付けられているオブジェクト・リポジトリのリストがオブジェクト・リポジトリ表示枠に表示されます。



- 3 オブジェクト・リポジトリ表示枠で、 [**オブジェクトリポジトリの作成**] ボタンをクリックします。 [**オブジェクトリポジトリを追加**] ダイアログ・ボックスが開き、現在のプロジェクトのテスト計画ツリーが表示されます。
- 4 テスト計画ツリーの中で、オブジェクト・リポジトリを格納するフォルダに移動します。
- 5 [**添付名**] ボックスに、オブジェクト・リポジトリの名前を入力し、 [**OK**] をクリックします。テスト計画ツリー内の選択した場所に、新しいオブジェクト・リポジトリが追加され、オブジェクト・リポジトリ表示枠に一覧表示されます。



- 6 テスト・オブジェクトを共有オブジェクト・リポジトリに追加する場合や、QuickTest からファイルに直接変更を加える場合は、オブジェクト・リポジトリ表示枠の中でファイルを選択し、 [**オブジェクトリポジトリを開く**] ボタン をクリックするか、リスト内でオブジェクト・リポジトリをダブルクリックします。ファイルがオブジェクト・リポジトリ・マネージャに開くので、必要に応じて編集できます。
- 7 オブジェクト・リポジトリ名を変更する場合は、そのリポジトリを2回クリックするか、選択して F2 キーを押します。

オブジェクト・リポジトリ・マネージャ内のオブジェクト・リポジトリの変更の詳細については、第15章「オブジェクト・リポジトリの管理」を参照してください。

アプリケーション領域への既存の共有オブジェクト・リポジトリの追加

既存の共有オブジェクト・リポジトリ・ファイルをアプリケーション領域に追加できます。これにより、アプリケーション領域が関連付けられているすべてのビジネス・コンポーネントから、それらのファイルに格納されているテスト・オブジェクトにアクセスできるようになります。

アプリケーション領域に対して既存の共有オブジェクト・リポジトリを追加するには、次の手順を実行します。

- 1 QuickTest で、アプリケーション領域をまだ開いていなければ開きます。

▶ **[ファイル]** > **[開く]** > **[アプリケーション領域]** を選択します。



▶ **[開く]** ボタンの下矢印をクリックし、**[アプリケーション領域]** を選択します。



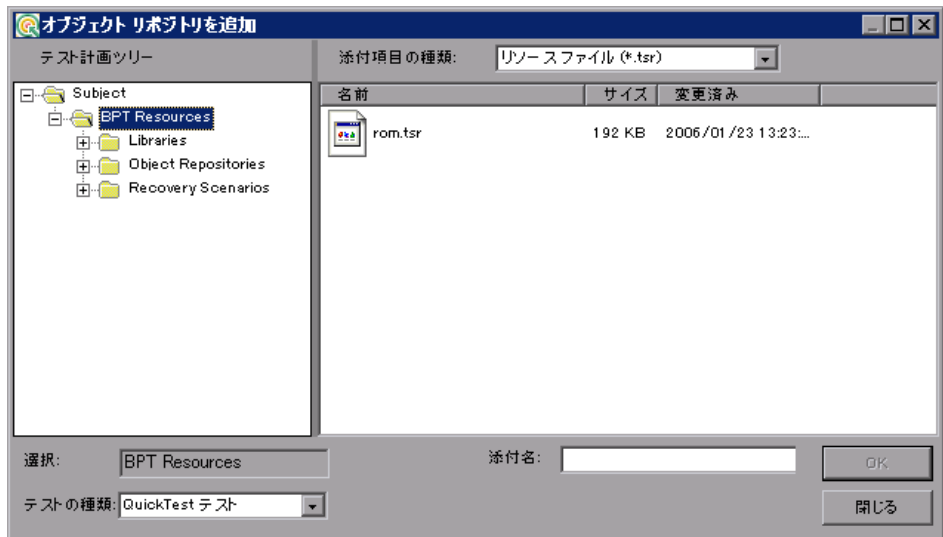
ヒント：別のアプリケーション領域がすでに開いている場合は、**[開く]** ボタンをクリックし、必要なアプリケーション領域を選択することもできます。


- 2 サイドバーの **[オブジェクトリポジトリ]** をクリックします。アプリケーション領域に現在関連付けられているオブジェクト・リポジトリのリストがオブジェクト・リポジトリ表示枠に表示されます。



- 3 **[オブジェクトリポジトリを追加]** ボタンをクリックします。空の行と参照ボタンがリストに追加されます。

- 4 参照ボタンをクリックします。**[オブジェクトリポジトリを追加]** ダイアログ・ボックスが表示されます。ダイアログ・ボックスに、現在の Quality Center プロジェクトのテスト計画ツリーが表示されます。

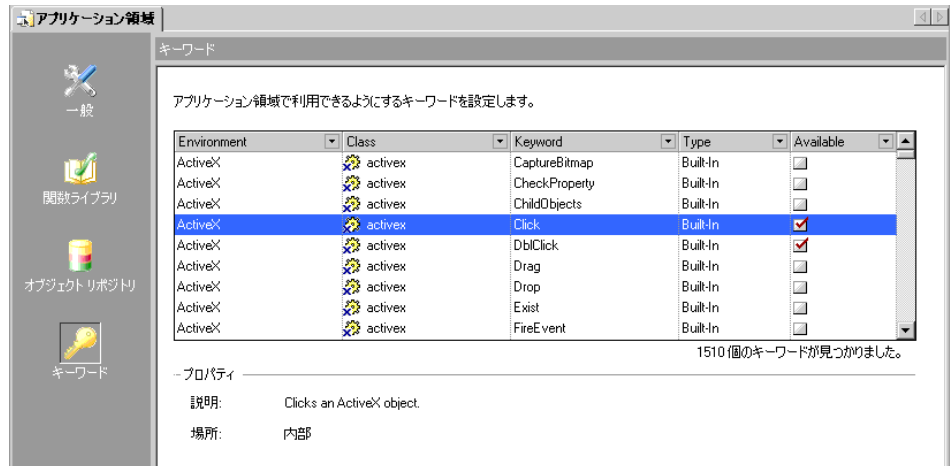


- 5 該当する項目をツリーで選択し、その項目に添付されているオブジェクト・リポジトリを表示します。次に、アプリケーション領域に関連付けるオブジェクト・リポジトリを選択します。名前が **[添付名]** ボックスに表示されます。
- 6 **[OK]** をクリックします。[オブジェクトリポジトリを追加] ダイアログ・ボックスが閉じ、選択したファイルがアプリケーション領域のオブジェクト・リポジトリ表示枠に表示されます。
-  7 テスト・オブジェクトを共有オブジェクト・リポジトリに追加する場合や、QuickTest からファイルを直接変更する場合は、オブジェクト・リポジトリ表示枠でファイルを選択し、[オブジェクトリポジトリを開く] ボタンをクリックするか、リスト内でオブジェクト・リポジトリをダブルクリックします。ファイルがオブジェクト・リポジトリ・マネージャで開き、必要に応じて編集できます。
- 8 オブジェクト・リポジトリ名を変更したい場合は、リポジトリを2回クリックするか、選択して F2 キーを押します。

キーワードの管理

ビジネス・コンポーネントのステップの作成時には、各分野のエキスパートは、テスト対象のアプリケーションに対して実行する必要がある操作を選択します。これらの操作はキーワードとも呼ばれ、組み込みのメソッドやプロパティのほか、アプリケーション領域に関連付けられているユーザ定義関数にも基づいています。

キーワード表示枠には、すべての組み込みのメソッドとプロパティ、およびユーザ定義関数ライブラリ内のすべての関数が、キーワードとして表示されます。キーワード表示枠では、キーワードを管理し、各分野のエキスパートがコンポーネント作成時に使用できるキーワードを選択することができます。標準設定では、一部の組み込みキーワードのみが使用できます。ただし、ユーザ定義キーワードであれば、各分野のエキスパートはすべてを使用できます。



ビジネス・コンポーネント・ステップの操作リストにあるキーワードを各分野のエキスパートが使用できるようにするには、[Available] カラムで該当するチェック・ボックスをクリックします。キーワードを使用可能な操作のリストから削除するには、チェック・ボックスをクリアします。各分野のエキスパートは、チェック・ボックスがクリアされているキーワードは使用できなくなります。

キーワード表示枠には、キーワードに関する情報が次のカラムに表示されます。

カラム	詳細
[Environment]	<p>キーワードが提供されるアドインの名前。たとえば、Web や Visual Basic などです。現在読み込み済みのすべてのアドインについて使用可能なキーワードが表示枠に表示されます。</p> <p>注：テスト・オブジェクトに登録されているユーザ定義関数内のキーワードは、その登録先の環境とオブジェクト・クラスの下に表示されます。テスト・オブジェクトに登録されていないユーザ定義関数内のキーワードと、組み込みのVBScript関数内のキーワードは、「Global」環境の下に表示されます。</p>
[Class]	オブジェクト・クラス。たとえば、 Image や Winbutton などです。
[Keyword]	表示される操作名。たとえば、 Click や VerifyProperty などです。
[Type]	操作が Built-In (組み込み) (QuickTest により提供されているもの) か User-Defined (ユーザ定義) (関数ライブラリにあるもの) かを示します。
[Available]	そのキーワードを、各分野のエキスパートがビジネス・コンポーネント・ステップの中で使用できるかどうかを示します。必要に応じて各チェック・ボックスを選択またはクリアできます。

リスト内のキーワードをクリックすると、そのキーワードに関する情報が表示枠の下部にある **[プロパティ]** 領域に表示されます。これには、キーワードの動作を説明するテキストや、関数ライブラリの名前とパス (ユーザ定義キーワードの場合) が表示されます。組み込みのキーワードの場所は「**内部**」として定義されています。

キーワード表示枠内のデータの表示、並べ替え、およびフィルタリングを実行することで、各分野のエキスパートに使用できるようにする (または各分野のエキスパートに対して非表示にする) キーワードを探すのが簡単になります。

ヒント：

キーワード表示枠でカラムの表示順序を変えるには、カラム見出しを新しい位置までドラッグします。有効な位置にカラムをドラッグすると、赤い矢印が表示されます。






カラムが狭すぎるためにカラム内のデータの一部が隠れている場合は、マウスを使用してカラムのサイズを変更できます。カラム見出しの分割線をドラッグして幅を調整します。

カラムのフィルタリング

キーワード表示枠のデータをフィルタリングして、作業に必要なキーワードだけを表示できます。単一のカラムにあるデータだけをフィルタリングしたり、複数のカラムをフィルタリングして表示される項目の数をさらに減らしたりできます。

たとえば、各分野のエキスパートが現在使用できないようになっている Web アドインのキーワードだけを表示することができます。これには、**[Environment]** カラムをフィルタリングして Web アドインからのキーワードだけを表示した後、**[Available]** カラムをフィルタリングしてチェック・ボックスがクリアされているキーワードだけを表示します (**[Available]** カラム・フィルタ・リストから **[Unchecked]** を選択します)。

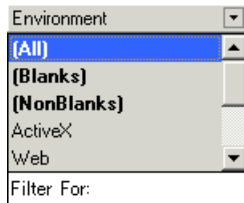
カラムの下には、フィルタ条件と、現在のフィルタに一致するキーワードの数が表示されます。

Web	 browser	GetTOPProperty	Built-In	<input type="checkbox"/>
Web	 browser	Home	Built-In	<input type="checkbox"/>
Web	 browser	Object	Built-In	<input type="checkbox"/>
Web	 browser	Refresh	Built-In	<input type="checkbox"/>
Web	 browser	SetTOPProperty	Built-In	<input type="checkbox"/>
X [Environment] = 'Web' and not IsChecked([Available])				

フィルタ条件の左の **X** をクリックすると、フィルタがクリアされ、すべてのキーワードが表示されます。

カラムのデータをフィルタリングするには、次の手順を実行します。

カラム見出しの矢印 ▼ をクリックします。カラムに含まれている一意の項目のリストが開きます。



次を実行してカラムのデータをフィルタリングできます。

- ▶ リスト内の項目をクリックします。CTRL キーを使用して、フィルタ・リストから複数の項目を選択できます。キーワード表示枠が更新され、選択した項目名を持つキーワードのデータのみが表示されます。

次に、別のカラム見出しの矢印をクリックして、そのリスト内の項目を選択できます。フィルタリングされていたデータがもう一度フィルタリングされ、選択したすべてのフィルタ条件に一致するキーワードだけが表示されます。

- ▶ フィルタ・リストの一番下にある [Filter For:] ボックスには、?、*、# などのワイルドカードを含むフィルタ・パターンを入力できます。ENTER キーを押すと、パターンに従ってデータがフィルタリングされます。任意の1文字を表すには ? を使用します。任意の文字の0個以上の出現を表すには * を使用します。任意の数字を表すには # を使用します。| を使用して、パターン内の候補のいずれか1つにだけ一致する項目を指定することもできます。たとえば、**Verify*|Check*** は、**Verify** または **Check** で始まるすべてのキーワードを表示します。
- ▶ 複数のフィルタを一度に適用できます。たとえば、標準 Windows 環境または ActiveX 環境の（ユーザ定義キーワードではなく）組み込みのキーワードのみを表示したい場合は、3つのフィルタを適用できます。1つは「StandardWindows」、もう1つは「activex」、そして3つめは「Built-in」です。

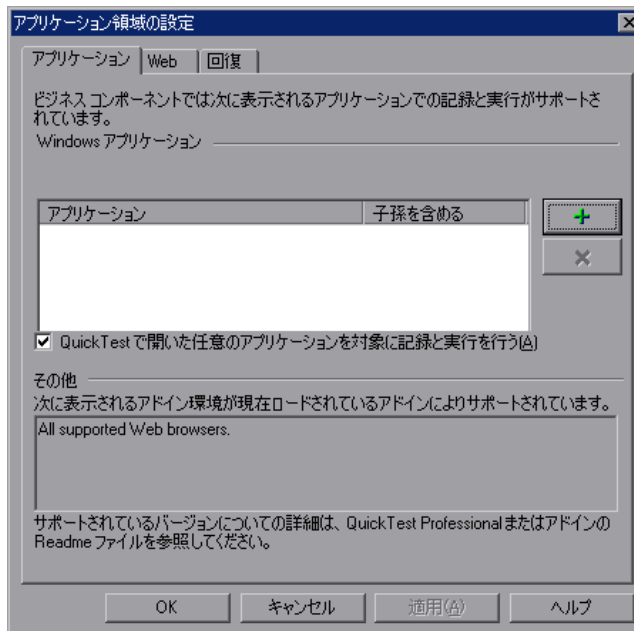
カラム内容の並べ替え

カラム見出しをクリックすると、カラムのデータをアルファベットの昇順または降順に並べ替えることができます。[Available] カラムは、チェック・ボックスが選択されているかクリアされているかに応じて並べ替えられます。

並べ替えの順序はカラム見出し内の矢印で示されます。もう一度カラム見出しをクリックすると、データが逆の順序に並べ替えられます。

追加設定の定義

一般表示枠の [追加設定] ボタンをクリックすると、[アプリケーション領域の設定] ダイアログ・ボックスが開きます。このダイアログ・ボックスはいくつかのタブで構成されています。これらのタブではアプリケーション領域に対する具体的な設定を定義できます。たとえば、アプリケーション領域に関連付けられているコンポーネントが記録と実行の対象にできるアプリケーションや、実行セッション中の予期しないイベントからコンポーネントがどのように回復するのかなどを定義できます。



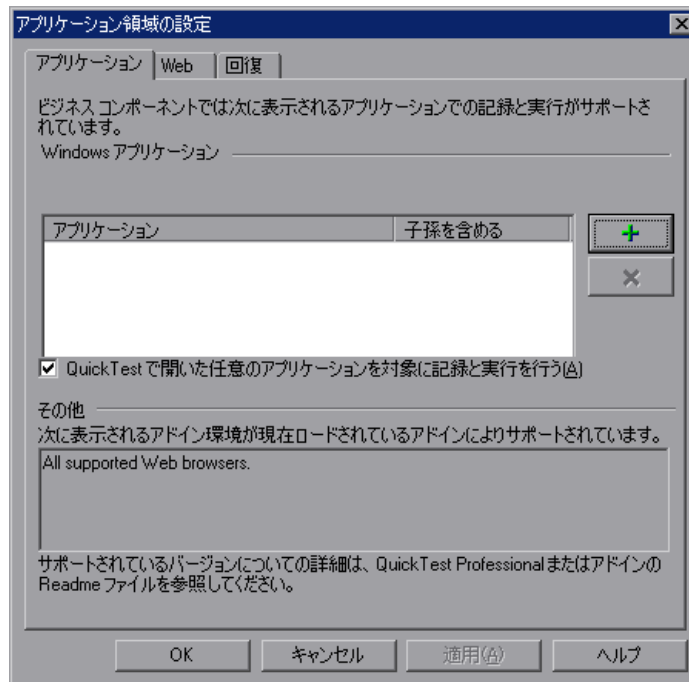
注：SAP や Web サービスなどの外部アドインが読み込まれていれば、それらに対応するタブが [アプリケーション領域の設定] ダイアログ・ボックスに含まれることもあります。外部アドインに関連するタブの詳細については、対応する QuickTest アドインのドキュメントを参照してください。

アプリケーション領域のアプリケーション設定の定義

[アプリケーション] タブでは、このアプリケーション領域に関連付けられているコンポーネントが記録と実行の対象にできる Windows ベース・アプリケーションを指定できます。指定したアプリケーションが対象の場合にのみ、コンポーネント・ステップを記録できます。

ヒント：アプリケーションを対象に記録を行うには、アプリケーションを手作業で開くか、QuickTest に用意されている **Common.txt** 関数ライブラリ内の **OpenApp** キーワード（関数）を使用します。QuickTest コンポーネント用にアプリケーションを自動的に開くための設定はありません。

[その他] 領域には、アプリケーション領域が関連付けられているコンポーネントが（現在読み込まれているアドインに基づいて）現在記録できる環境が表示されます。



次のような場合に、[アプリケーション] タブを使ってアプリケーションにかかわる設定が行えます。



- ▶ 関連付けられているコンポーネントに 1 つ以上のステップを記録済みで、記録を続行する前に設定を変更したい場合。
- ▶ 以前に使用したものと異なるアプリケーションを対象にコンポーネントを実行したい場合。

注：

新規コンポーネントを記録する際に [アプリケーション領域の設定] ダイアログ・ボックスの [アプリケーション] タブにおいてアプリケーションにかかわる設定を済ませていなければ、記録の開始時に [アプリケーション] ダイアログ・ボックスが開きます。[アプリケーション] ダイアログ・ボックスには、[アプリケーション] タブと同じオプションが含まれています。

[アプリケーション] ダイアログ・ボックスと [アプリケーション] タブには、コンピュータにインストールされている任意の QuickTest 外部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、アドインに付属のマニュアルを参照してください。

[アプリケーション] タブには次のオプションがあります。

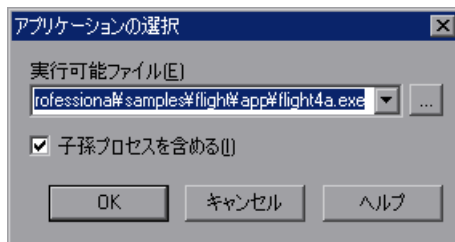
オプション	詳細
Windows アプリケーション	このアプリケーション領域に関連付けられているコンポーネントが記録と実行の対象とするアプリケーションの詳細を一覧表示します。表示される詳細については、109 ページ「アプリケーションの指定」を参照してください。 Windows アプリケーションを記録や実行の対象にしない場合は、アプリケーション・リストを空のままにします（これは、標準設定です）。
	アプリケーション・リストにアプリケーションを追加します。10 個までのアプリケーションを追加できます。詳細については、109 ページ「アプリケーションの指定」を参照してください。
	選択したアプリケーションをアプリケーション・リストから削除します。

オプション	詳細
[QuickTest で開いた任意のアプリケーションを対象に記録と実行を行う]	QuickTest によって起動される任意のアプリケーションを (QuickTest の子プロセスとして) 記録と実行の対象にします。たとえば、セッションの記録または実行中に OpenApp 関数を使用して開かれるアプリケーションや、アプリケーションを開く関数が含まれている別の操作によって開かれたアプリケーションなどが対象になります。
[その他]	現在読み込まれているアドインに対応するアドイン環境を一覧表示します。

アプリケーションの指定



[アプリケーション] タブで [追加] ボタンをクリックすると、[アプリケーションの選択] ダイアログ・ボックスが開きます。



[アプリケーション] タブに表示されるアプリケーション・リストには 10 個までのアプリケーションを追加できるほか、リスト内の既存のアプリケーションを編集できます。また、アプリケーションの子孫プロセスを記録と実行の対象にするかどうかを選択できます。

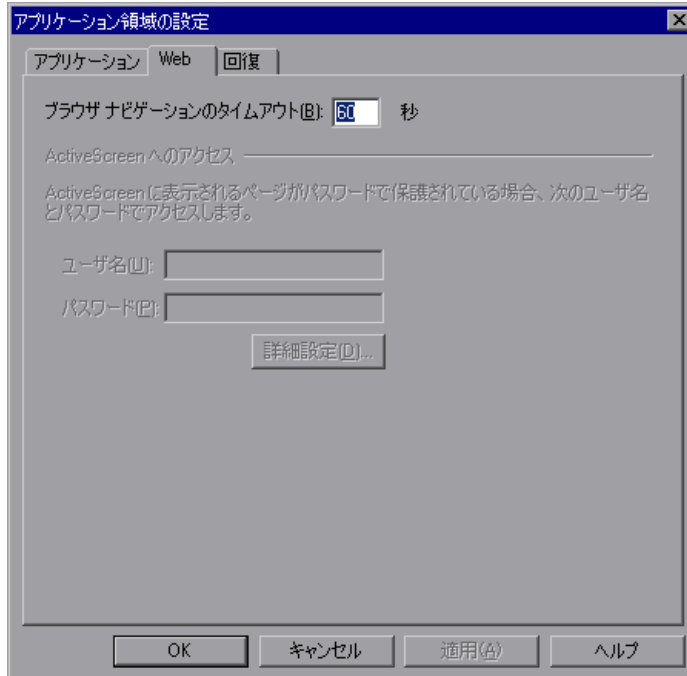
[アプリケーションの選択] ダイアログ・ボックスに入力した詳細は、[アプリケーション] タブの [Windows アプリケーション] 領域にアプリケーションごとに 1 行で表示されます。

[アプリケーションの選択] ダイアログ・ボックスでは、アプリケーションに次の詳細を指定できます。

オプション	詳細
[実行可能ファイル]	指定した実行可能ファイルを記録と実行の対象にするよう QuickTest を設定します。
[子孫プロセスを含める]	このチェック・ボックスを選択すると、指定したアプリケーションによって記録および実行セッション中に作成されるプロセスも記録と実行の対象にするよう、QuickTest を設定します。たとえば、起動専用として使用されるプロセスから、アプリケーションの機能を実際に提供する別のプロセスが作成される場合があります。したがって、このアプリケーションを対象にテストを記録し実行するときは、このような下位プロセスを含める必要があります。そうしなければ、機能が記録されなかったり、実行セッションが失敗したりします。 標準設定では、このオプションは選択されています。

アプリケーション領域の Web 設定の定義

[Web] タブには、Web ページの読み込みが完了するまでの待機時間を定義するオプションがあります。



注：[Web] タブは、Web アドインがインストールされ、読み込まれている場合にのみ利用できます。

[Web] タブには、次のオプションがあります。

オプション	詳細
[ブラウザナビゲーションのタイムアウト]	QuickTest がテストまたはコンポーネントのステップを実行する前に、Web ページの読み込みが完了するのを待機する最長の時間（秒単位）を設定します。
[ユーザ名]	このオプションはコンポーネントの場合は該当しません。
[パスワード]	このオプションはコンポーネントの場合は該当しません。
[詳細設定]	このオプションはコンポーネントの場合は該当しません。

ヒント：このタブのオプションに加えて、Web オブジェクトのタイプごとに記録するイベントを設定することも可能です。たとえば、サブメニューを開く mouseover イベントのようなイベントを記録したい場合には、当該イベントを検出できるように Web イベント設定を変更する必要があるかもしれません。詳細については、第 11 章「Web イベント記録の設定」を参照してください。

アプリケーション領域の回復シナリオ設定の定義

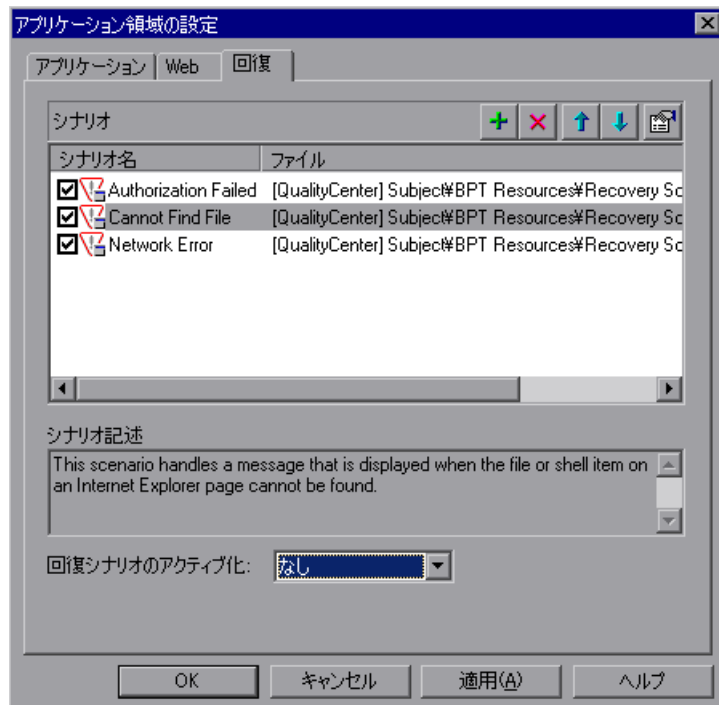
回復シナリオの設定では、ビジネス・コンポーネントが実行セッション中の予期しないイベントやエラーからどのように回復するかを指定できます。

[回復] タブには、現在のアプリケーション領域に関連付けられているすべての回復のシナリオが表示されます。また、アプリケーション領域への追加の回復シナリオの関連付け、アプリケーション領域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の適用順序の変更、および、各シナリオの表示専用サマリの表示が可能です。

アプリケーション領域の特定のシナリオまたは回復メカニズム全体を有効にしたり無効にしたりできます。回復シナリオを Quality Center プロジェクトに格納する限り、既存の回復シナリオを追加したり、新規に作成したりできます。

注：QuickTestにはWeb関連のテストに対応したサンプルの回復ファイルが用意されています。このファイルはQuality Centerプロジェクトの**Subject#BPT Resources#Recovery Scenarios#DefaultWeb.qrs**に格納されています。

アプリケーション領域の回復シナリオは、テストの回復シナリオとまったく同じ方法で定義します。回復シナリオの詳細については、第20章「回復シナリオの定義と使用」を参照してください。



[回復] タブには、次のオプションがあります。






オプション	詳細
[シナリオ]	アプリケーション領域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、114 ページ「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]	<p>関連付けられているシナリオをいつ実行するかを次のように確認するよう QuickTest を設定します。</p> <ul style="list-style-type: none"> ● [各ステップごと]：回復メカニズムはステップの終了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した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。 <p>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。</p>

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、現在のアプリケーション領域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。回復シナリオ・ファイルのパスを変更した場合には、このアプリケーション領域に関連付けられているコンポーネントを実行する前に、回復シナリオが新しいパスの場所に存在することを確認します。

シナリオのタイプは、次のアイコンによって示されます。

アイコン	詳細
	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。
	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。
	対象回復シナリオが、コンポーネントのステップが正常に実行されなかっ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、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、アプリケーション領域で利用できなくなったことを示します。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、メッセージが結果に表示されます。

次のボタンを使用して、コンポーネン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、優先順位設定ができます。

ボタン	詳細
	[回復シナリオの追加] ダイアログ・ボックスが開き、コンポーネントに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、第20章「回復シナリオの定義と使用」を参照してください。
	選択した回復シナリオをビジネス・コンポーネントから削除します。
	選択したシナリオをリストの中で上に移動することで、コンポーネントの実行セッションにおける優先順位を上げます。
	選択したシナリオをリストの中で下に移動することで、コンポーネントの実行セッションにおける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、第20章「回復シナリオの定義と使用」を参照してください。

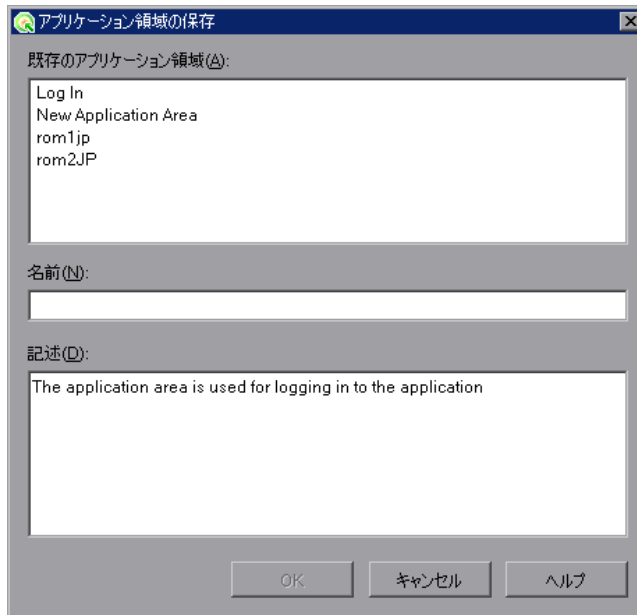
アプリケーション領域の保存

アプリケーション領域の保存は、設定およびリソースを定義する前または定義した後で行うことができます。

アプリケーション領域を保存するときには、一意の名前と、用途を明確に示す説明を付けてください。たとえば、ログイン・モジュールをテストするコンポーネントで使用するアプリケーション領域であれば、「Log In」という名前を付けて、「Intended for use with business components that test the Log In module.」のように用途を示す説明を追加します。

アプリケーション領域を保存するには、次の手順を実行します。

- 1 QuickTest で、Business Process Testing のサポートが組み込まれた 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、20 ページ「Quality Center プロジェクトへの接続」を参照してください。
- 2 アプリケーション領域を作成し、必要に応じて設定を変更します。詳細については、79 ページ「アプリケーション領域の作成」を参照してください。
- 3 **[保存]** をクリックするか、**[ファイル]** > **[保存]** を選択します。[アプリケーション領域の保存] ダイアログ・ボックスが開きます。



[アプリケーション領域の保存] 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
[既存のアプリケーション領域]	Quality Center プロジェクトにあるすべての定義済みのアプリケーション領域を一覧表示します。この一覧で既存のアプリケーション領域の名前を確認すれば、保存するアプリケーション領域に一意の名前を付けられます。
[名前]	アプリケーション領域の名前を示します。各分野のエキスパートがそのアプリケーション領域が自分のコンポーネントに適しているかどうかをすばやく判断できるように、わかりやすい名前を入力します。 注： 入力できる名前は255文字までです。スペースで始めたり、スペースで終わったりすることはできません。また、次の文字を含めることはできません： ¥ / : " ? < > * ! { } ' % ;
[記述]	アプリケーション領域を作成するときに [アプリケーション領域] ダイアログ・ボックスの一般表示枠で入力した説明 ([詳細]) が表示されます。詳細については、79 ページ「アプリケーション領域の作成」を参照してください。アプリケーション領域を作成するときに説明を入力しなかった場合は、ここで入力する必要があります。説明を入力せずにアプリケーション領域を保存することはできません。一般表示枠ですでに説明を定義している場合は、既存の説明を変更することもできます。説明を提供することで、各分野のエキスパートは、各種のアプリケーション領域を簡単に区別して各自のコンポーネントに最適なアプリケーション領域を選択できるようになります。

4 [OK] をクリックし、アプリケーション領域を保存します。

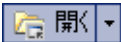
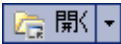
ヒント：既存のアプリケーション領域に似た新しいアプリケーション領域を作成する場合は、**[名前を付けて保存]** オプションを使用できます。その後、必要に応じてアプリケーション領域に変更を加えることができます。

アプリケーション領域の削除

不要になったアプリケーション領域は削除できます。アプリケーション領域を削除する前に、どのビジネス・コンポーネントにも使用されていないことを確かめる必要があります。ビジネス・コンポーネントに使用されているアプリケーション領域は削除できません。

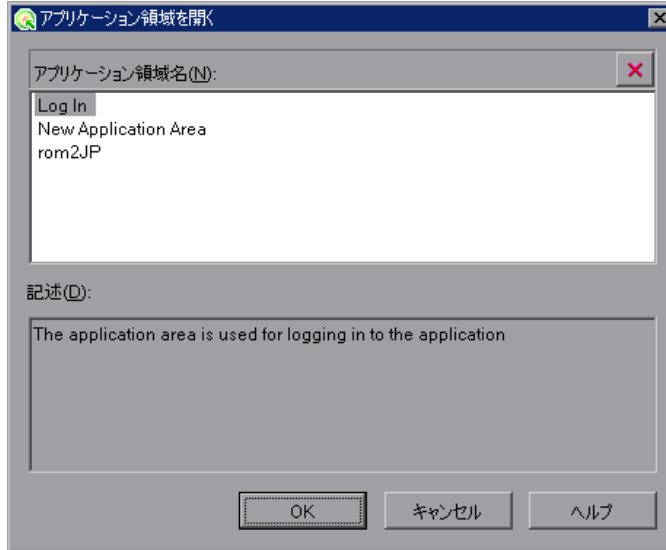
アプリケーション領域を削除するには、次の手順を実行します。

- 1 QuickTest で、削除するアプリケーション領域が保存されている Quality Center プロジェクトに接続します。詳細については、20 ページ「Quality Center プロジェクトへの接続」を参照してください。
- 2 QuickTest で、アプリケーション領域をまだ開いていなければ開きます。
 - ▶ [ファイル] > [開く] > [アプリケーション領域] を選択します。
 - ▶ [開く] ボタンの下矢印をクリックし、[アプリケーション領域] を選択します。



ヒント：別のアプリケーション領域がすでに開いている場合は、[開く] ボタンをクリックし、必要なアプリケーション領域を選択することもできます。

[アプリケーション領域を開く] ダイアログ・ボックスが開きます。



- 3 削除するアプリケーション領域を選択して [アプリケーション領域の削除] ボタンをクリックします。警告メッセージが表示されます。

注：現在開いているアプリケーション領域、現在別のオートメーション・エンジニアが使用しているアプリケーション領域、またはコンポーネントに関連付けられているアプリケーション領域は、削除できません。

- 4 [はい] ボタンをクリックして、確定します。選択したアプリケーション領域が削除されます。
- 5 [OK] をクリックし、[アプリケーション領域を開く] ダイアログ・ボックスを閉じます。

第4章

ビジネス・コンポーネントの使用

ビジネス・コンポーネント・キーワード・ビューでは、QuickTestでのビジネス・コンポーネントの作成、表示、変更、デバッグを行うことができます。

本章では、次の項目について説明します。

- ▶ ビジネス・コンポーネントの使用について
- ▶ 新規ビジネス・コンポーネントの作成
- ▶ ビジネス・コンポーネントの保存
- ▶ ビジネス・コンポーネントを開く
- ▶ 手動コンポーネントを使った作業
- ▶ コンポーネントに関連付けられているアプリケーション領域の変更
- ▶ コンポーネントの印刷

ビジネス・コンポーネントの使用について

通常、ビジネス・コンポーネントは各分野のエキスパートによって Quality Center で作成され変更されます。詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。ただし、ビジネス・コンポーネント・キーワード・ビューでは、必要に応じて QuickTest でビジネス・コンポーネントの作成、表示、変更、デバッグを行うことができます。

キーワード・ビューでは、モジュール化されたキーワード駆動型の表形式に、ビジネス・コンポーネントがステップごとに分けて表示されます。各ステップは、簡単に変更できる独立した部分から構成されています。ステップは、項目と操作を選択して必要に応じて追加情報を入力することによって、作成したり変更したりできます。

ビジネス・コンポーネントの各ステップには、完成したときに自動的に説明が付け加えられます。これにより、ユーザはステップの説明をわかりやすい文章で読むことができます。また、ビジネス・コンポーネントに関連付けられているアプリケーション領域に関数ライブラリを追加した場合は、ユーザ定義操作（関数）を選択してステップを定義するときに、関数ライブラリに追加した文書がステップに対して表示されます。詳細については、606 ページ「関数への説明の追加」を参照してください。

ビジネス・コンポーネントを作成するか開く前に、まず QuickTest を Quality Center プロジェクトに接続する必要があります。これは、Quality Center プロジェクトにはビジネス・コンポーネント、およびアプリケーション領域のリソースと設定が保存されているためです。Quality Center プロジェクトに接続することによって、QuickTest でビジネス・コンポーネントを作成したり開いたりできるようになります。また、ビジネス・コンポーネントから、アプリケーション領域で定義されたそのコンポーネントが基づいているあらゆるリソースにアクセスできるようになります。

注：ビジネス・コンポーネントおよびアプリケーション領域を使用する前に、必要な Quality Center アクセス許可と権限を持っていることを確認する必要があります。ビジネス・コンポーネントモジュール内のユーザ・グループ設定の詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。

選択したアプリケーション領域に必要なリソースと設定がまだすべて含まれていない場合でも、**ManualStep** 関数または [コメント] オプションを使ってステップを追加できます。このオプションを使えば、Quality Center、あるいは Microsoft Excel や Microsoft Word などの他のアプリケーションで入力するように、手動のステップを自由に入力できます。また、コメントを使って、ステップに関する情報を追加したり、ビジネス・コンポーネントのセクション間を区切ったりすることもできます。各手動ステップおよびコメントは、キーワード・ビュー内に独立した行として表示されます。詳細については、137 ページ「コンポーネントの手動ステップの追加および変更」および 180 ページ「コメントの使用」を参照してください。

注：

コンポーネントを削除する場合は、QuickTestまたはQuality Centerのどちらで作成されたものでも、Quality Centerにおいてのみ削除できます。詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。必要であれば、ビジネス・コンポーネントをスクリプト・コンポーネントに変換できます。詳細については、第5章「スクリプト・コンポーネントを使った作業」を参照してください。

新規ビジネス・コンポーネントの作成

QuickTestをQuality Centerプロジェクトに接続すると、そのプロジェクトに新規のビジネス・コンポーネントを作成できます。

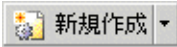
各ビジネス・コンポーネントは特定のアプリケーション領域に基づいており、アプリケーション領域はそのコンポーネントの保存先となるQuality Centerプロジェクトに保存されています。各アプリケーション領域は、ビジネス・コンポーネントの設定とリソースを指定します。これには、共有オブジェクト・リポジトリ、関数ライブラリ・ファイル、回復シナリオの各場所、およびその他の情報が含まれます。使用可能なアプリケーション領域が複数存在する場合もあります。作成するビジネス・コンポーネントに最も適したアプリケーション領域を選択します。詳細については、第3章「アプリケーション領域の使用」を参照してください。

通常、ビジネス・コンポーネントは各分野のエキスパートによってQuality Centerで作成されます。詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。ただし、必要に応じてQuickTestでビジネス・コンポーネントを作成することもできます。この項では、QuickTestで新規コンポーネントを作成する方法を説明します。

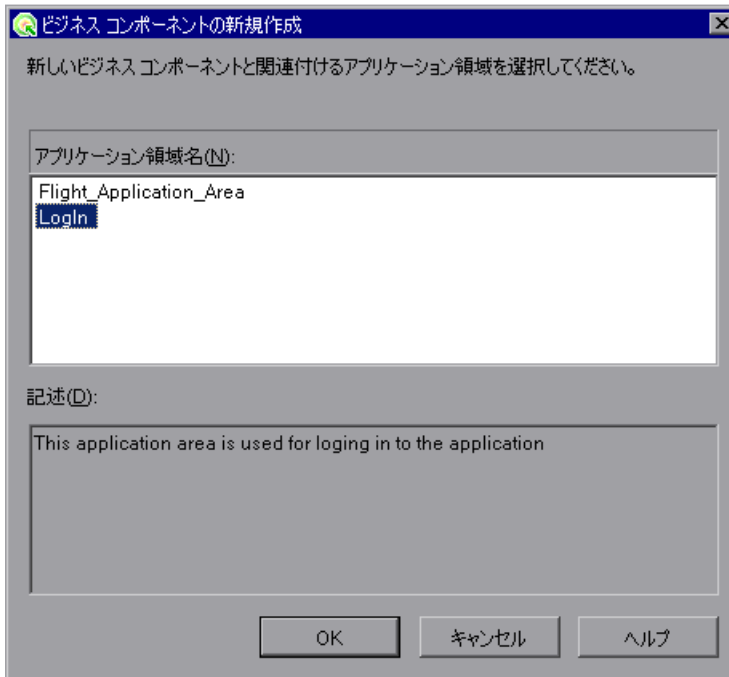
注：QuickTestでビジネス・コンポーネントを新規に作成するためには、ビジネス・プロセス・テストの作成に必要な権限が必要です。詳細については、『Quality Center 管理者ガイド』を参照してください。

新規のビジネス・コンポーネントを作成するには、次の手順を実行します。

- 1 ビジネス・コンポーネントの保存先となる Quality Center プロジェクトに接続します。詳細については、20 ページ「Quality Center プロジェクトへの接続」を参照してください。
- 2 次の手順のいずれかを実行します。
 - ▶ [ファイル] > [新規作成] > [ビジネス コンポーネント] を選択します。
 - ▶ [新規作成] ボタンの下矢印をクリックし、[ビジネス コンポーネント] を選択します。



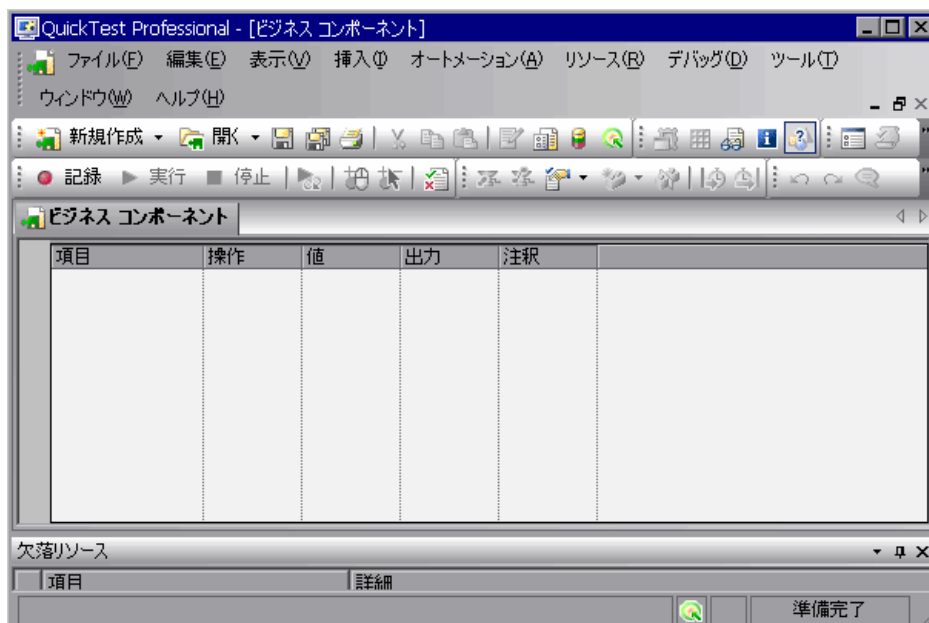
[ビジネス コンポーネントの新規作成] ダイアログ・ボックスが開き、使用可能なアプリケーション領域がすべて表示されます。アプリケーション領域をクリックすると、その説明が表示されます（各分野のエキスペートは新規のビジネス・コンポーネントを作成するときに、これらの説明によってどのアプリケーション領域を選択するべきかを判断できます）。



注：まだアプリケーション領域を定義していない場合は、Business Process Testing に用意されている標準設定を使用して、無題の新しいビジネス・コンポーネントが開きます。アプリケーション領域の定義後、そのアプリケーション領域にビジネス・コンポーネントを関連付けることができます。詳細については、第3章「アプリケーション領域の使用」を参照してください。

- 3 [アプリケーション領域] ボックスから適切なアプリケーション領域を選択します。たとえば、あるログイン・モジュール用のビジネス・コンポーネントを作成する場合は、そのモジュール用に定義されたアプリケーション領域を選択します。[OK] をクリックします。

無題の新しいビジネス・コンポーネントがキーワード・ビュー内に開きます。新規のビジネス・コンポーネントにはまだ内容はありませんが、基盤とするアプリケーション領域に定義されている必要な設定とリソースはすべて含まれています。これらの設定は、[ファイル] > [設定] を選択して、読み取り専用形式で表示できます。設定を変更する必要がある場合は、関連するアプリケーション領域から変更します。



4 この時点で、次の操作を実行できます。

- ▶ ビジネス・コンポーネントへのステップとコメントの追加。詳細については、153 ページ「コンポーネントへのステップの追加」および 180 ページ「コメントの使用」を参照してください。
- ▶ コンポーネントの保存（後からステップを追加できます）。詳細については、126 ページ「ビジネス・コンポーネントの保存」を参照してください。

ビジネス・コンポーネントの保存

作成あるいは変更したビジネス・コンポーネントは、Quality Center プロジェクトに保存できます。ビジネス・コンポーネントを保存する際には、分かりやすい名前を付けて、Quality Center プロジェクト（ビジネス・コンポーネント・モジュール）のコンポーネント・ツリー内の適切なフォルダに保存してください。

また、既存のコンポーネントのコピーを、同じ Quality Center プロジェクト内の任意のフォルダに保存することもできます。すべてのユーザが各種のコンポーネントを区別できるように、コンポーネントのコピーの名前は、たとえ別のフォルダに保存する場合でも変更するようにします。

ヒント：コンポーネントに変更を加えた場合は、コンポーネントを保存するまで、タイトル・バーにアスタリスクが表示されます。

ビジネス・コンポーネントとスクリプト・コンポーネントは同じように保存できます。スクリプト・コンポーネントの詳細については、141 ページ「スクリプト・コンポーネントを使った作業」を参照してください。

ビジネス・コンポーネントをスクリプト・コンポーネントに変換することもできます。詳細については、148 ページ「スクリプト・コンポーネントへのビジネス・コンポーネントの変換」を参照してください。

注：スクリプト・コンポーネントの場合のみ、データ・テーブルのデータ・シート名がスクリプト・コンポーネント名と同じになります。スクリプト・コンポーネントを新しい名前では保存すると（**[ファイル]** > **[名前を付けて保存]**）、データ・シート名は自動的に変更されます。名前によってデータ・シートを参照するステップがある場合、実行セッションでそのステップは失敗します。これはこのステップが以前のデータ・シート名を参照するためです。スクリプト・コンポーネントを新しい名前では保存する場合、エキスパート・ビューの中で以前のデータ・シート名への参照がないか調べ、あればそれを新しいデータ・シート名で置換する必要があります。

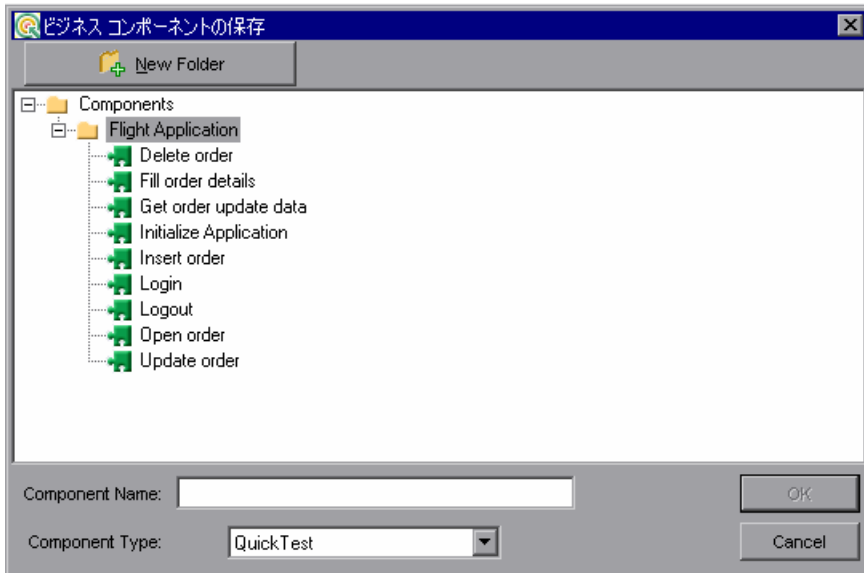
コンポーネントを Quality Center プロジェクトに保存するには、次の手順を実行します。

- 1 次のいずれかの方法でコンポーネントを保存します。




- ▶ まだ一度も保存していない新規のコンポーネントの場合は、**[ファイル]** > **[保存]** を選択するか、**[保存]** ボタンをクリックします。
- ▶ 既存のコンポーネントのコピーを保存するには、**[ファイル]** > **[名前を付けて保存]** を選択します。

[ビジネス コンポーネントの保存] ダイアログ・ボックスが開き、コンポーネント・ツリーが表示されます。



コンポーネント・ツリーでは、各コンポーネントのステータスがアイコンで示されます。詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。

- 2 コンポーネントを保存する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。下位レベルの表示を折りたたむには、開いているフォルダをダブルクリックします。

 フォルダの新規作成

コンポーネントは、Quality Center プロジェクトの既存のフォルダに保存することも、[フォルダの新規作成] ボタンをクリックして新しいフォルダを作成し、そのフォルダに保存することもできます。既存のコンポーネントのコピーを同じ名前で作成するには、別のフォルダに保存する必要があります。

- 3 [コンポーネント名] ボックスに、コンポーネントの名前を入力します。誰もがコンポーネントを識別しやすいよう、分かりやすい名前を付けます。入力できるコンポーネント名は 255 文字までです。スペースで始めたり、スペースで終わったりすることはできません。また、次の文字を含めることはできません。
¥ / : " ? < > | * ! { } ' % ;

- 4 [コンポーネントの種類] では、標準設定の **QuickTest コンポーネント** を受け入れます。
- 5 [OK] をクリックし、コンポーネントを保存してダイアログ・ボックスを閉じます。QuickTest がコンポーネントを保存している間、実行中の操作がステータス・バーに表示されます。

コンポーネントが **Quality Center** プロジェクトに保存されます。これで、ビジネス・コンポーネントを **QuickTest** で表示、変更できるようになります。

注： 各分野のエキスパートは、**Quality Center** ビジネス・コンポーネント・モジュールからもコンポーネントにアクセスできます。詳細については、『**Business Process Testing ユーザーズ・ガイド**』を参照してください。



ヒント： コンポーネントを以前に保存したことがあれば、[ファイル] > [保存] を選択するか、[保存] ボタンをクリックして保存できます。

ビジネス・コンポーネントを開く

QuickTest が Quality Center プロジェクトに接続されている場合、プロジェクトに保存されているコンポーネントを開き、表示、変更、デバッグ、実行を行うことが可能です。コンポーネントは、コンポーネント・ツリー内の場所に応じて見つけます。

注：

現在 Quality Center、あるいは他の QuickTest セッションで開いているコンポーネントはロックされ、読み取り専用形式でのみ開くことができます。これらのコンポーネントを対象に作業をするには、このコンポーネントを開いている Quality Center または別の QuickTest セッションで閉じなくてはなりません。

コンポーネントを開いたとき、関連付けられているアプリケーション領域が見つからない場合、別のアプリケーション領域の関連付けを求められます。以前のバージョンの QuickTest を実行しているコンピュータで、それ以降のバージョンの QuickTest を使用して作成したコンポーネントを開くことはできません。たとえば、QuickTest 8.2 を実行しているコンピュータで、QuickTest 9.0 で作成したコンポーネントを開くことはできません。

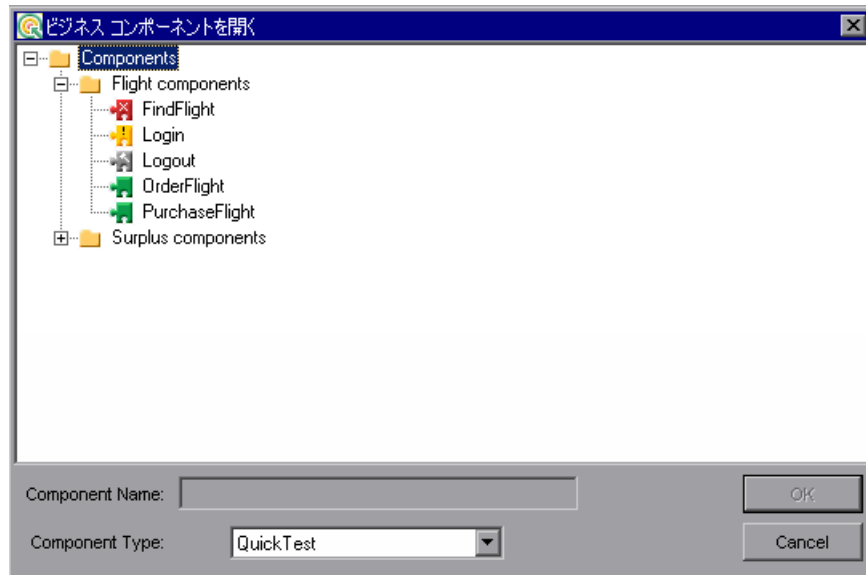
ビジネス・コンポーネントとスクリプト・コンポーネントは同じ方法で開くことができます。詳細については、142 ページ「スクリプト・コンポーネントにかかわる作業について」を参照してください。

既存のコンポーネントを開くには、次の手順を実行します。

- 1 QuickTest で、コンポーネントが保存されている 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、20 ページ「Quality Center プロジェクトへの接続」を参照してください。
- 2 **[ファイル] > [開く] > [ビジネス/スクリプト コンポーネント]** を選択します。または、**[開く]** 矢印をクリックして **[ビジネス/スクリプト コンポーネント]** を選択します。**[ビジネス コンポーネントを開く]** ダイアログ・ボックスが開き、Quality Center プロジェクトに保存されているコンポーネントが表示されます。



ダイアログ・ボックスに表示されるコンポーネントの種類は、手順3に従って変更できます。



コンポーネント・ツリーでは、各コンポーネントのステータスがアイコンで示されます。詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。

ヒント：最近使用したコンポーネントを開くこともできます。その場合は、**[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します。Quality Center プロジェクトに接続されていないときにコンポーネントを選択した場合、あるいは現在の接続先とは異なる Quality Center プロジェクトに保存されているコンポーネントを選択した場合には、そのプロジェクトに接続するかどうかを問うメッセージが表示されます。詳細については、133 ページ「最近使用したコンポーネントのリストからコンポーネントを開く」を参照してください。

- 3 必要ならば、[ビジネス コンポーネントを開く] ダイアログ・ボックスに表示されるコンポーネントのリストをフィルタリングします。これには、開きたいコンポーネントの種類を [コンポーネントの種類] リストから選択します。次のコンポーネントの種類のいずれかを選択することができます。
 - ▶ [QuickTest コンポーネント] : QuickTest Professional または Business Process Testing キーワード・ビューを使用して自動化されたコンポーネントを表示します。
 - ▶ [Manual Component] : Quality Center で作成され、まだ自動コンポーネントに変換されていないコンポーネント (手動コンポーネント) を表示します。詳細については、134 ページ「手動コンポーネントを使った作業」を参照してください。
 - ▶ [All Components] : すべての QuickTest 自動コンポーネントおよび手動コンポーネントを表示します。
- 4 コンポーネント・ツリー内の該当するフォルダをクリックします。ツリーを展開してビジネス・コンポーネントを表示するには、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。
- 5 コンポーネントを選択します。コンポーネントの名前が読み取り専用の [コンポーネント名] ボックスに表示されます。
- 6 [OK] をクリックしてコンポーネントを開きます。

QuickTest がコンポーネン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。

ビジネス・コンポーネントが開くと、QuickTest のタイトル・バーに、Components という文字列と、フルパスおよびコンポーネント名が表示されます。たとえば、flight_login コンポーネントを開くと、タイトル・バーに次のように表示されます。

[Components¥Flight¥flight_login]

最近使用したコンポーネントのリストからコンポーネントを開く

[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コンポーネントを開くことができます。Quality Center プロジェクトのコンポーネン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、コンポーネン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コンポーネントを開いたユーザの名前が表示されます。



プロジェクトにログインし、[OK] をクリックします。

また、前回異なる Quality Center ユーザ名を使用して編集したコンポーネン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したままでもできます。

手動コンポーネントを使った作業

Quality Center で作成した手動コンポーネントをビジネス・コンポーネントに変換することができます。変換後は、他のビジネス・コンポーネントと同じ方法で表示、変更、デバッグ、および実行ができます。

手動コンポーネントをビジネス・コンポーネントに変換した後も、その手動ステップは依然として Quality Center で表示できます。また、Quality Center マニュアル・ランナーを使用して手動コンポーネントとして実行することもできます。さらに、必要に応じて手動ステップを変更したり、手動ステップを追加したりすることも可能です。

注：Quality Center の中から手動コンポーネントをビジネス・コンポーネントに変換することもできます。詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。

手動コンポーネントを開いて変換する

QuickTest では、Quality Center プロジェクトに格納されている手動コンポーネントを開き、ビジネス（キーワード駆動の）コンポーネントに変換できます。QuickTest で手動コンポーネントを開くとき、当該コンポーネントをビジネス・コンポーネントに変換するかどうか尋ねられます。

注：現在 Quality Center、あるいは他の QuickTest セッションで開いているコンポーネントはロックされ、読み取り専用形式でのみ開くことができます。これらのコンポーネントを対象に作業をするには、このコンポーネントを開いている Quality Center または別の QuickTest セッションで閉じなくてはなりません。

手動コンポーネントを開いて変換するには、次の手順を実行します。

1 QuickTest で、コンポーネントが保存されている 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、20 ページ「Quality Center プロジェクトへの接続」を参照してください。

2 次の手順のいずれかを実行します。

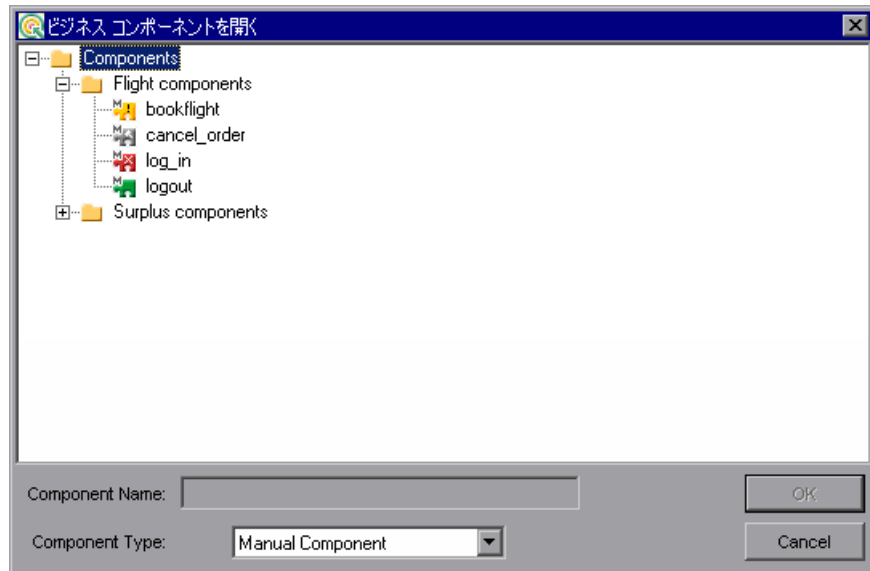
▶ [ファイル] > [開く] > [ビジネス/スクリプト コンポーネント] を選択します。



▶ [開く] 矢印をクリックし、[ビジネス/スクリプト コンポーネント] を選択します。

[ビジネス コンポーネントを開く] ダイアログ・ボックスが開き、Quality Center プロジェクトに保存されているコンポーネントが表示されます。

ダイアログ・ボックスに表示されるコンポーネントの種類は、手順3に従って変更できます。



コンポーネント・ツリーでは、各コンポーネントのステータスがアイコンで示されます。詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。




- 必要であれば、[コンポーネントの種類] リストから、開きたいコンポーネントの種類を選択して、[ビジネス コンポーネントを開く] ダイアログ・ボックスに表示されているコンポーネントの一覧にフィルタを適用します。標準設定では、QuickTest コンポーネントのみ表示されます（QuickTest コンポーネントは、QuickTest Professional または Business Process Testing のキーワード・ビューを使用して自動化されたコンポーネントです）。

次のいずれかのコンポーネント・タイプを選択します。

- ▶ **[Manual Component]** : Quality Center で作成され、まだ自動コンポーネントに変換されていないコンポーネント（手動コンポーネント）を表示します。手動コンポーネントを開くことを選択した場合、当該コンポーネントは QuickTest コンポーネントに変換され、その手動ステップはキーワード・ビュー・ステップに変換されます。この変換プロセスは元に戻すことができません（Quality Center で必要に応じて手動ステップを表示および実行することは可能です）。
 - ▶ **[All Components]** : すべての QuickTest 自動コンポーネントおよび手動コンポーネントを表示します。
- コンポーネント・ツリー内の該当するフォルダをクリックします。ツリーを展開してビジネス・コンポーネントを表示するには、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。
 - 手動コンポーネントを選択します。手動コンポーネントは、アイコンの左隅に「M」の付いたコンポーネント・アイコンによって表されます。コンポーネントの名前が読み取り専用の **[コンポーネント名]** ボックスに表示されます。
 - [OK]** をクリックしてコンポーネントを開きます。QuickTest から、手動コンポーネントをビジネス・コンポーネントに変換するかどうか尋ねられます。
 - 続けて変換を行うには、**[はい]** をクリックします。このプロセスは元に戻すことができないので注意してください。
 - [ビジネス コンポーネントの新規作成]** 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、ビジネス・コンポーネント用のアプリケーション領域を選択します。アプリケーション領域を選択して **[OK]** をクリックします。アプリケーション領域の詳細については、第3章「アプリケーション領域の使用」を参照してください。

QuickTest がコンポーネントをダウンロードして開き、変換するのに応じて、実行される操作がステータス・バーに表示されます。

手動コンポーネントの各手動ステップがキーワード・ビューの **ManualStep** 操作に変換されます。




項目	操作	値
 操作	ManualStep	"Step 1"."Enter the user name and press tab key"."cursor moves to
 操作	ManualStep	"Step 2"."Enter password."."Password should be masked with aste
 操作	ManualStep	"Step 3"."Click the Enter button."."Login dialog box closes and app

各 **ManualStep** 操作について、各手動ステップの名前、説明、および期待結果が引数値として追加されます。入力および出力パラメータが定義されている場合は、それらがローカル・パラメータに変換されます。

これで、他のコンポーネントと同様にコンポーネントを使用できるようになります。Quality Center のマニュアル・ランナーを使用してビジネス・コンポーネントを手動コンポーネントとして実行できるように、手動ステップをさらに追加したり、既存の手動ステップを変更したりできます。詳細については、「コンポーネントの手動ステップの追加および変更」を参照してください。

コンポーネントの手動ステップの追加および変更

手動コンポーネントをビジネス・コンポーネントに変換すると、手動コンポーネントの各手動ステップがキーワード・ビューの **ManualStep** 操作に変換されます。

項目	操作	値
 操作	ManualStep	"Step 1"."Enter the user name and press tab key"."cursor moves to
 操作	ManualStep	"Step 2"."Enter password."."Password should be masked with aste
 操作	ManualStep	"Step 3"."Click the Enter button."."Login dialog box closes and app

ステップ名、ステップの説明、および期待結果を変更するには、キーワード・ビューの該当する **ManualStep** 行の対応する引数値を変更します。

QuickTest では、変換後のコンポーネントに新しいステップを追加できます（通常のビジネス・コンポーネント・ステップ、および **ManualStep** 操作も追加できます）。Quality Center の [ステップのデザイン] タブで **ManualStep** 操作をコンポーネントに追加することも、Quality Center の [自動化] タブでキーワード方式のステップを追加することもできます。ステップを必要に応じて削除することもできます。

QuickTest においてコンポーネントの **ManualStep** 操作および通常のキーワード方式のステップに加えた変更はすべて、変更を保存した後に、Quality Center でコンポーネントの [ステップのデザイン] タブおよび [自動化] タブに反映されます（逆もまた同様です）。このため、Quality Center および QuickTest のどちらからコンポーネントを更新しても、必要に応じて引き続き Quality Center マニュアル・ランナーを使用してコンポーネントを手動で実行できます。

キーワード・ビューにおけるステップの追加の詳細については、151 ページ「ビジネス・コンポーネント・キーワード・ビューを使った作業」を参照してください。ManualStep 操作の詳細については、『QuickTest Professional オブジェクト・モデル・リファレンス』の「ユーティリティ」の項を参照してください。

Quality Center におけるステップの追加、および Quality Center のマニュアル・ランナーを使用した手動コンポーネントの実行の詳細については、『Mercury Business Process Testing ユーザーズ・ガイド』を参照してください。

コンポーネントに関連付けられているアプリケーション領域の変更

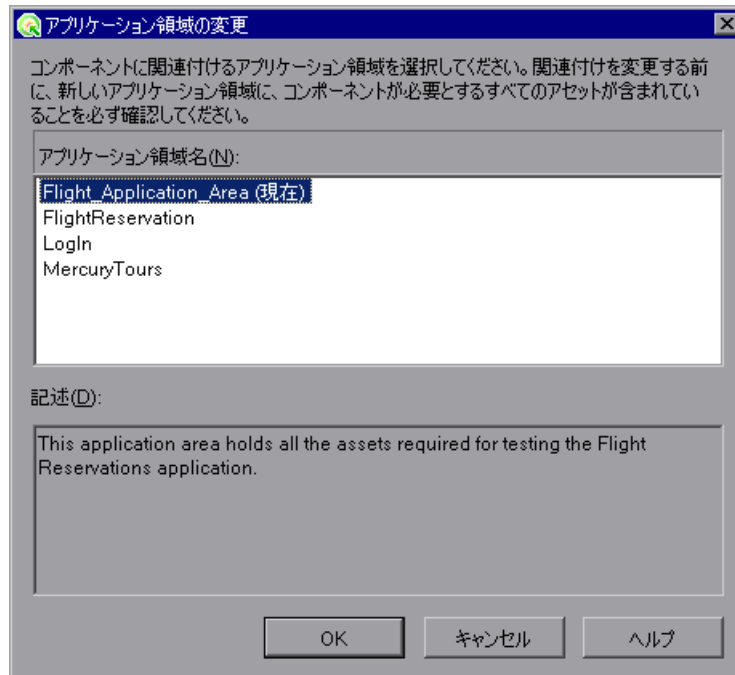
QuickTest でビジネス・コンポーネントを作成するときは、コンポーネントに関連付ける対象となるアプリケーション領域を選択する必要があります。使用可能なアプリケーション領域が複数存在する場合があります。コンポーネントに最も適したアプリケーション領域を選択してください。

アプリケーション、またはアプリケーション領域に関連付けられているリソース・ファイルや設定に対して、変更が加えられた場合、アプリケーション領域が不適切になることがあります。特定のコンポーネントに関連付けられているアプリケーション領域を変更しなければならないことがあります。たとえば、オブジェクト・リポジトリが変更されたり、アプリケーション領域から削除されたりする可能性があります。あるいは、アプリケーションの開発過程で、現在関連付けられているオブジェクト・リポジトリに含まれていない追加の異なるオブジェクトが含まれることもあります。このような場合、コンポーネントやビジネス・プロセス・テストが正しく実行されなかったり、実行に失敗したりする可能性があります。必要なリソース・ファイルや設定が別のアプリケーション領域に含まれている場合は、コンポーネントに関連付けられているアプリケーション領域を変更する必要があります。

注：QuickTest では、コンポーネントを開くたびに、コンポーネントに対して指定されているリソースが使用可能かどうかの確認が行われます。共有オブジェクト・リポジトリが見つからないなど、コンポーネントまたはアプリケーション領域が持つリソースが見つからない場合は、そのことが QuickTest によって欠落リソース表示枠に示されます。詳細については、第7章「欠落リソースの処理」を参照してください。

アプリケーション領域を変更するには、次の手順を実行します。

- 1 130 ページ「ビジネス・コンポーネントを開く」で説明した手順に従って、コンポーネントを開きます。
- 2 [ファイル] > [アプリケーション領域の変更] を選択します。[アプリケーション領域の変更] ダイアログ・ボックスが開きます。



- 3 コンポーネントに関連付けるアプリケーション領域を選択します。アプリケーション領域の説明が〔記述〕領域に表示されます。
- 4 [OK] をクリックすると、コンポーネントに関連付けられるアプリケーション領域が変更されます。

コンポーネントの印刷

コンポーネントを表形式で印刷することができます。

コンポーネントを印刷するには、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。標準の [印刷] ダイアログ・ボックスが開きます。
- 2 [OK] をクリックすると、[キーワード ビュー] の内容が標準設定の Windows プリンタに出力されます。

ヒント：画面上で [キーワード ビュー] の印刷イメージを表示するには、[ファイル] > [印刷プレビュー] を選択します。

第 5 章

スクリプト・コンポーネントを使った作業

スクリプト・コンポーネントは、特定のタスクを実行する、保守および再利用が可能なスクリプトです。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に共通する機能を持っています。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 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。また、既存のビジネス・コンポーネントまたは既存のアクションを、スクリプト・コンポーネントに変換できます。本章では、スクリプト・コンポーネントの作成方法についてのみ説明します。

本章では、次の項目について説明します。

- ▶ スクリプト・コンポーネントにかかわる作業について
- ▶ スクリプト・コンポーネントの作成
- ▶ スクリプト・コンポーネントへの変換

スクリプト・コンポーネントにかかわる作業について

スクリプト・コンポーネントを対象に作業をする際には、キーワード・ビュー、エキスパート・ビュー、および QuickTest ツールやオプションの機能を最大限に使用できます。たとえば、スクリプト・コンポーネントにメソッドや関数を追加するプロセスを案内するステップ・ジェネレータを使用できます。エキスパート・ビューを使って、標準 VBScript ステートメントや QuickTest オブジェクトやメソッドを使用する他のプログラミング・ステートメントを手入力することによって、スクリプト・コンポーネントを拡張できます。また、スクリプト・コンポーネントのステップにユーザ定義関数を統合し、選択した項目をパラメータ化し、スクリプト・コンポーネントにチェックポイントと出力値を追加することもできます。

スクリプト・コンポーネントは、各分野のエキスパートの要求に合わせて作成できます（たとえば、ループや条件文などより複雑な機能を含むコンポーネントが必要な場合など）。Quality Center で作業する各分野のエキスパートは、こうしたスクリプト・コンポーネントをビジネス・プロセス・テストに含め、アプリケーションが希望どおりに動作することを確認できます。

スクリプト・コンポーネントを作成した後で、各分野のエキスパートは、Quality Center プロジェクトのビジネス・コンポーネント・モジュールでコンポーネントによって生成された自動文書（読み取り専用）を表示できます。各分野のエキスパートはスクリプト・コンポーネントを実行したり、ビジネス・プロセス・テストに追加したりできますが、変更が必要になった場合に QuickTest でスクリプト・コンポーネントの保守を行うのは引き続き QuickTest エンジニアです。スクリプト・コンポーネントは Quality Center では変更できません。

スクリプト・コンポーネントは、ビジネス・コンポーネントと同じように保存したり開いたりすることができます。詳細については、126 ページ「ビジネス・コンポーネントの保存」および 130 ページ「ビジネス・コンポーネントを開く」を参照してください。

スクリプト・コンポーネントとその他のテスト・ドキュメントとの類似点

スクリプト・コンポーネントには、QuickTest のアクションやテストと同じ機能がたくさんあります。たとえば、次のことが行えます。

- ▶ エキスパート・ビューでのプログラム・ステートメントの使用（第22章「関数ライブラリ・ウィンドウを使用した作業」を参照）
- ▶ チェックポイントおよび出力値の作成
- ▶ 階層形式のキーワード・ビューの表示を参照
- ▶ 仮想オブジェクトの作成および操作を参照
- ▶ データ・テーブルを使用した複数の反復の実行を参照
- ▶ ActiveScreen を使用して、記録セッション中に特定のステップを実行したときに表示されていたアプリケーションのスナップショットを表示できます。また、アプリケーションが使用できない場合や、選択したオブジェクトに対応するステップがテストまたはコンポーネントにない場合でも、オブジェクト値をパラメータ化したり、ページ内の任意のオブジェクトを対象としたチェックポイント、メソッド、出力値を挿入したりできます。
- ▶ ランダム方式のパラメータおよび環境パラメータの使用を参照
- ▶ 記録または実行セッションの開始時に自動的起動するアプリケーションの設定を参照

スクリプト・コンポーネントが利用できるすべての機能の一般情報については、『QuickTest Professional 基本機能ユーザーズ・ガイド』および『QuickTest Professional 上級機能ユーザーズ・ガイド』を参照してください。

スクリプト・コンポーネントは、次の点でビジネス・コンポーネントにも似ています。

- ▶ 特定のアプリケーション領域に関連付けられています（すべてのリソースを、ファイル・システムではなく Quality Center プロジェクトに格納する必要があります）。
- ▶ ビジネス・プロセス・テストに組み込み可能なスタンドアロンのモジュラー化されたユニットです。
- ▶ ビジネス・プロセス・テストにおいて階層的ではなく直線的です。
- ▶ ネストされません。つまり別のコンポーネントを呼び出すことはできません。

スクリプト・コンポーネントの作成

QuickTest を Quality Center プロジェクトに接続すると、そのプロジェクトに新規のスクリプト・コンポーネントを作成できます。

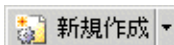
各スクリプト・コンポーネントは、特定のアプリケーション領域に基づいています。アプリケーション領域には、コンポーネントで使用されるリソースと設定が含まれています（例：共有オブジェクト・リポジトリや関数ライブラリの場所など）。作成するスクリプト・コンポーネントに最も適したアプリケーション領域を選択します。コンポーネントの保存先となる Quality Center プロジェクト内の任意のアプリケーション領域を選択できます。詳細については、75 ページ「アプリケーション領域の使用」を参照してください。

ヒント：ビジネス・コンポーネントをスクリプト・コンポーネントに変換することもできます。詳細については、148 ページ「スクリプト・コンポーネントへのビジネス・コンポーネントの変換」を参照してください。

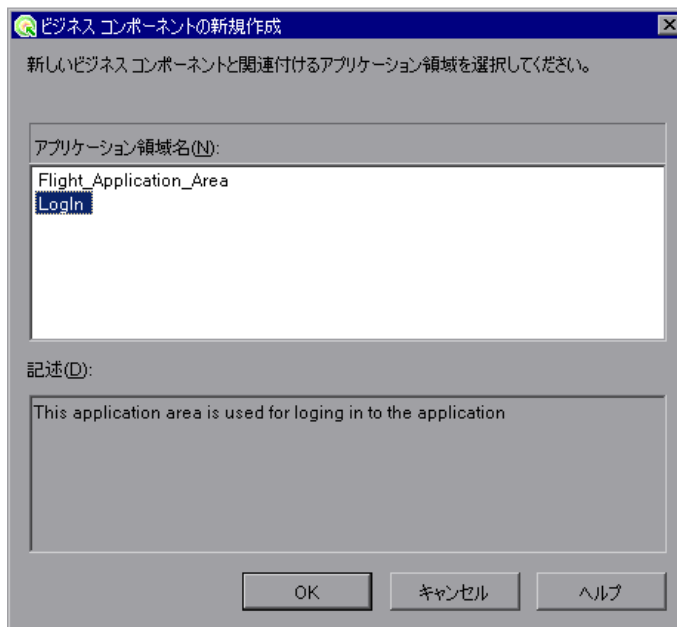
注：まだアプリケーション領域を作成していない場合は、スクリプト・コンポーネントは Business Process Testing に用意されている標準設定のアプリケーション領域設定に基づきます。

スクリプト・コンポーネントを作成するには、次の手順を実行します。

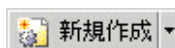
- 1 スクリプト・コンポーネントの保存先となる Quality Center プロジェクトに接続します。詳細については、20 ページ「Quality Center プロジェクトへの接続」を参照してください。
- 2 次の手順のいずれかを実行します。
 - ▶ [ファイル] > [新規作成] > [スクリプト コンポーネント] を選択します。
 - ▶ [新規作成] ボタンの下矢印をクリックして、[スクリプト コンポーネント] を選択します。



[ビジネス コンポーネントの新規作成] ダイアログ・ボックスが開き，使用可能なアプリケーション領域がすべて表示されます。



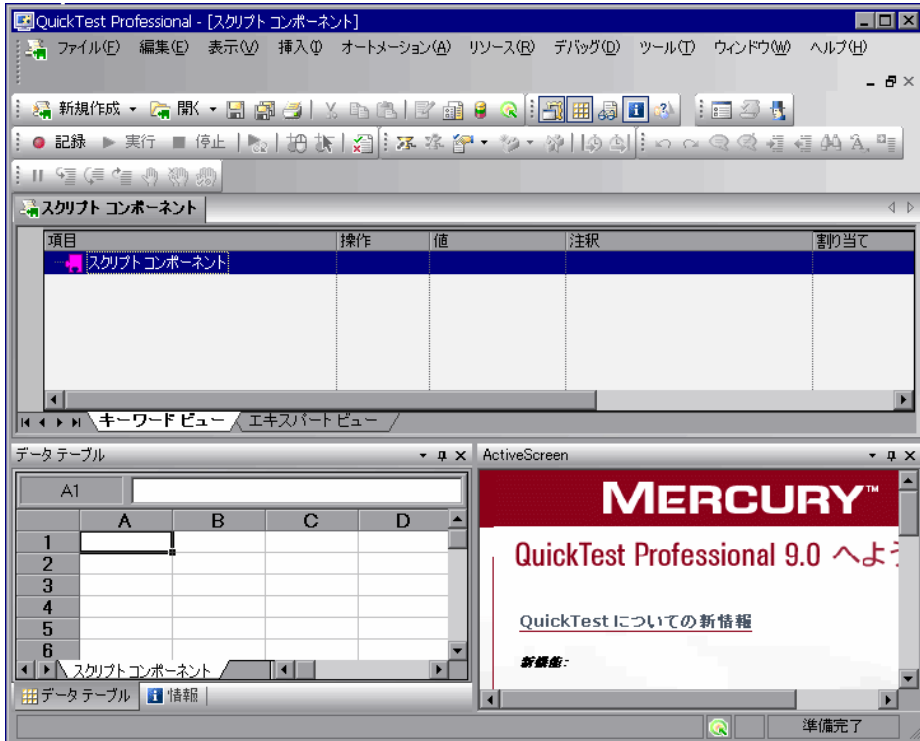
注：まだアプリケーション領域を作成していない場合は，Business Process Testing に用意されている標準の設定を使用して，無題の新しいスクリプト・コンポーネントが開きます。後からアプリケーション領域を変更することはできません。



ヒント：すでにスクリプト・コンポーネントが開いている場合は，ツールバーの [新規作成] ボタンをクリックして新規のスクリプト・コンポーネントを開くこともできます。

- 3 [アプリケーション領域] ボックスから適切なアプリケーション領域を選択します。たとえば、フライト予約モジュール用のスクリプト・コンポーネントを作成する場合は、そのために定義されたアプリケーション領域を選択します。[OK] をクリックします。

無題の新しいスクリプト・コンポーネントが開きます。新規のスクリプト・コンポーネントにはまだ内容はありますが、基盤とするアプリケーション領域に定義されている必要な設定とリソースはすべて含まれています。



4 この時点で、次の操作を実行できます。

- ▶ QuickTest で提供されている機能とオプションを使用して、スクリプト・コンポーネントに内容を追加します。たとえば、エキスパート・ビューで、VBScript ステートメントを手入力したり、QuickTest オブジェクトやメソッドを使ってステートメントを追加したりできます。ステップ・ジェネレータを使って、プログラミング・ロジックを含んだステップを追加できます。また、スクリプト・コンポーネントには、チェックポイントや出力値も追加できます。スクリプト・コンポーネントの作成時に使用できる機能の詳細については、『QuickTest Professional 基本機能ユーザーズ・ガイド』および『QuickTest Professional 上級機能ユーザーズ・ガイド』を参照してください。
- ▶ スクリプト・コンポーネントを保存します（後から内容を追加できます）。スクリプト・コンポーネントは、ビジネス・コンポーネントと同じように保存できます。詳細については、126 ページ「ビジネス・コンポーネントの保存」を参照してください。

注：Quality Center のビジネス・コンポーネント・モジュールの [ステップのデザイン] タブでは、各分野のエキスパートが表示したり、作業の対象にしたりできるのは、スクリプト・コンポーネント向けに定義されている手動ステップ（もしあれば）のみです。[自動化] タブの [起動] ボタンをクリックしてスクリプト・コンポーネントを QuickTest 内で開かない限り（QuickTest が Quality Center のクライアント上にインストールされている場合）、各分野のエキスパートは、自動化されたステップを表示したり変更したりできません。詳細については、『Business Process Testing ユーザーズ・ガイド』を参照してください。

スクリプト・コンポーネントへの変換

必要に応じて、ビジネス・コンポーネントおよびアクションを、スクリプト・コンポーネントに変換できます。ビジネス・プロセス・テストを使用する場合、一般に、既存のテスト・アクションまたはビジネス・コンポーネントをスクリプト・コンポーネントに変換するよりも、Quality Center で新しいビジネス・コンポーネントを作成することをお勧めします。これにより、Quality Center で作業する各分野のエキスパートが、時とともにコンポーネントを保守できるからです。さらに、スクリプト・コンポーネントは QuickTest においてのみ変更できるため（Quality Center では変更できません）、各分野のエキスパートは自動化されたステップを Quality Center で表示できませんが、手動ステップは（もしあれば）表示および編集が可能です。つまり、スクリプト・コンポーネントを、元のアクションまたはビジネス・コンポーネントに変換することはできません。

スクリプト・コンポーネントへのビジネス・コンポーネントの変換

1つのビジネス・コンポーネントをスクリプト・コンポーネントに変換できます。

ビジネス・コンポーネントをスクリプト・コンポーネントに変換するには、次の手順を実行します。

- 1 スクリプト・コンポーネントに変換するビジネス・コンポーネントを開きます。ビジネス・コンポーネントを開く方法の詳細については、130 ページ「ビジネス・コンポーネントを開く」を参照してください。

注：ビジネス・コンポーネントを読み取り専用モードで開いている場合、またはロックされている場合は、当該コンポーネントはスクリプト・コンポーネントに変換できません。

- 2 [ファイル] > [スクリプト コンポーネントに変換] を選択します。
- 3 確認を求められたら [OK] をクリックし、変換処理を進めます。

注：この操作で既存のビジネス・コンポーネントはスクリプト・コンポーネントに置き換えられます。取り消しはできません。

変換が完了すると、新しいスクリプト・コンポーネントが自動的に開きます。

スクリプト・コンポーネントへのアクションの変換

アクション変換ツールを使用して、QuickTest テスト・アクションをビジネス・プロセス・テストで使用するスクリプト・コンポーネントに変換できます。

アクション変換ツールを開くには、[スタート] メニューから [QuickTest Professional] プログラム・グループを開き、[Tools] > [Action Conversion Tool] を選択します。

アクション変換ツールの使用方法については、アクション変換ツールを開いて [ヘルプ] ボタンをクリックしてください。

第 6 章

ビジネス・コンポーネント・キーワード・ビューを使った作業

ビジネス・コンポーネント・キーワード・ビューでは、グラフィカルで使いやすい形式で、ビジネス・コンポーネントの作成、表示、変更、デバッグを簡単に行うことができます。

本章では、次の項目について説明します。

- ▶ ビジネス・コンポーネント・キーワード・ビューを使った作業 について
- ▶ コンポーネントへのステップの追加
- ▶ ステップの項目の選択
- ▶ ステップの操作の選択
- ▶ ステップの引数の値の定義
- ▶ ステップの出力値の定義
- ▶ パラメータの使用
- ▶ コメントの使用
- ▶ コンポーネントのステップの管理

ビジネス・コンポーネント・キーワード・ビューを使った作業について

一般に、各分野のエキスパートは Quality Center ビジネス・コンポーネント・モジュールの [自動化] タブを使用して、コンポーネント・ステップに内容を追加したり、コンポーネント・ステップを変更したりします。ただし、本章で説明するように、この作業は QuickTest でも実行できます。

QuickTest では、ビジネス・コンポーネント・キーワード・ビューを使用して、ビジネス・コンポーネント・ステップの作成、表示、変更、デバッグを行います。

ビジネス・コンポーネント・キーワード・ビューは、ビジネス・コンポーネントを簡単に作成できるようにするために特別に設計されたコンポーネント専用のオプションを提供している点が、QuickTest テスト・キーワード・ビューと異なります。ビジネス・コンポーネント・キーワード・ビューはこの違いにより、各分野のエキスパートにとってビジネス・コンポーネントを Quality Center で作成するための簡単で直観的なツールとなっています (QuickTest に表示されるビジネス・コンポーネント・キーワード・ビューは、各分野のエキスパートが Quality Center で使用する [自動化] タブと同じです)。

ビジネス・コンポーネント・キーワード・ビューには、コメントを含めることができます。これにより、ビジネス・コンポーネントに手動ステップや情報セパレータを入力できます。キーワード・ビュー内の項目 (テスト・オブジェクトと操作) はすべて、たとえ先に実行されるステップまたは操作の下位のオブジェクトであっても、同じ階層レベルに表示されます。これにより、各分野のエキスパートがビジネス・コンポーネントのステップを管理しやすくなっています。

ビジネス・コンポーネント・キーワード・ビューで作業を行うには、Business Process Testing のサポートが組み込まれた Quality Center プロジェクトに QuickTest を接続する必要があります。

項目	操作	値	注釈
Mercury Inter...	Navigate	"http://newtours.mercuryinteractive.com"	Navigate to "http://newtours.mercuryinteractive.com" in the browser.
userName	Set	"Mercury"	Enter "Mercury" in the "userName" edit box.
password	SetSecure	"41442073fc37b0bf9831d26e01b7aab...	Enter the encrypted string "41442073fc37b0bf9831d26e01b7aab8ce...
Sign-In	Click	9,8	Click the "Sign-In" image.
fromPort	Select	"New York"	Select the "New York" item in the "fromPort" list.
fromMonth	Select	"Dec"	Select the "Dec" item in the "fromMonth" list.
fromDay	Select	"29"	Select the "29" item in the "fromDay" list.
toPort	Select	"San Francisco"	Select the "San Francisco" item in the "toPort" list.
toMonth	Select	"Dec"	Select the "Dec" item in the "toMonth" list.
toDay	Select	"30"	Select the "30" item in the "toDay" list.
servClass	Select	"Business"	Select radio button "Business" in the "servClass" radio button group.
findFlights	Click	93,10	Click the "findFlights" image.
findFlights	Select	"Business" class	Select radio button "Business" class


コンポーネントへのステップの追加

新規のビジネス・コンポーネントを作成した後、キーワード・ビューは次の図のように空です。

項目	操作	値	出力	注釈

ビジネス・コンポーネントにステップを追加すると、各ステップがキーワード・ビュー内の単一の行として定義されます。ステップは、現在選択されているステップの下、既存のコンポーネントの最後、または新しいコンポーネントの先頭に追加できます。また、コメントも入力できます。

ステップ：ステップは、実行される操作を表します。ステップを作成した後、その内容を指定します。たとえば、ステップの実行対象であるテスト・オブジェクトを選択したり、ステップで実行される操作を指定したり、関連する入力値や出力値を指定することができます。Quality Center でビジネス・プロセス・テストが実行されると、関連するビジネス・コンポーネントで定義されたステップが実行されます。この項では、ビジネス・コンポーネントにステップを追加する方法を説明します。

コメント：コメントは、行全体にわたって自由に入力できるテキストです。 アイコンはコメントを示します。コメントを使って、手動ステップを定義したり、ビジネス・コンポーネント内で隣り合うステップに関する情報を入力したりできます。コメントは、ビジネス・プロセス・テストが実行される時には処理されません。詳細については、180 ページ「コメントの使用」を参照してください。

ステップを追加するには、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。

- ▶ キーワード・ビュー内の任意の場所（既存のステップが存在する場合はその下）をクリックしてコンポーネントの最後にステップを追加します。まだステップを定義していなければ、ここで追加するステップがコンポーネントの最初のステップになります。
- ▶ [挿入] > [ステップの新規作成] を選択し、（存在する場合は）既存のステップの後ろに新しいステップを追加します。コンポーネントにステップがなければ、ここで追加するステップがコンポーネントの最初のステップになります。
- ▶ たとえば、コンポーネントを2つのステップのセットに分けるためにコメントを追加するには、既存のステップを選択して [挿入] > [ステップの新規作成] を選択し、既存のステップの間に新しいステップを追加します（最後のステップを選択すると、QuickTest はコンポーネントの最後にステップを追加します）。
- ▶ 既存のステップを右クリックして、ショートカット・メニューから [ステップの新規作成] を選択します（これにより、たとえば、コンポーネントを2つのステップのセットに分けるためにコメントを追加する場合、2つの既存のステップの間にステップを追加することもできます）。

キーワード・ビューの選択したステップの下に新しいステップが追加されます。



注：通常，[項目の選択] リストは，対応するすべてのテスト・オブジェクトのほか，[操作] 項目や [コメント] 項目を表示するように展開されます。

- 2 ステップの一部として変更が必要なセルの中をクリックし，後述するように内容を指定して，ステップを定義します。ステップ行の各セルは，ステップの異なる部分を表します。各ステップについて，次を定義できます。

- ▶ **項目**：ステップを実行する対象となるテスト・オブジェクトまたはユーザ定義関数（**操作**）。ステップに内容を追加する前に，[項目] カラムから項目を選択する必要があります。詳細については，156 ページ「ステップの項目の選択」を参照してください。

項目を指定する代わりに，「**コメント**」を追加することもできます。その場合は，手動ステップや，それ以外の任意のテキスト情報を，ステップ間に追加できます。詳細については，180 ページ「コメントの使用」を参照してください。

- ▶ **操作**：項目に対して実行される操作。詳細については，164 ページ「ステップの操作の選択」を参照してください。
- ▶ **値**（該当する場合）：選択した操作の引数値。詳細については，166 ページ「ステップの引数の値の定義」を参照してください。
- ▶ **出力**（該当する場合）：ステップの出力値が格納されるパラメータ。詳細については，169 ページ「ステップの出力値の定義」を参照してください。

注：[注釈] セルは読み取り専用です。このセルには，「Click the "Sign-in" image.」や「Select "San Francisco" in the "toPort" list」のように，わかりやすい文でチェックの実行内容の説明が表示されます。ほとんどの場合，このセルに表示される説明は QuickTest によって自動的に生成されます。関数ライブラリを作成して，関連付けられているアプリケーション領域に追加した（関連付けた）場合，関数ライブラリの中で注釈のテキストを定義してある場合のみ，QuickTest に関数ライブラリの注釈が表示されます。詳細については，606 ページ「関数への説明の追加」および 89 ページ「関数ライブラリの管理」を参照してください。

ヒント：[編集] メニューまたはショートカット・メニューの標準的な編集コマンド（[切り取り]，[コピー]，[貼り付け]，[削除]）を使用して，ステップの定義や変更を簡単に行うことができます。また，ドラッグ・アンド・ドロップ操作で，ステップをコンポーネント内の別の場所にも移動することもできます。詳細については，182 ページ「コンポーネントのステップの管理」を参照してください。

- 3 変更が終わったら，コンポーネントを Quality Center プロジェクトに保存します。詳細については，126 ページ「ビジネス・コンポーネントの保存」を参照してください。

ステップの項目の選択

項目は，共有オブジェクト・リポジトリ内のテスト・オブジェクト，またはユーザ定義関数「操作」です（[操作] 項目は，コンポーネントのアプリケーション領域に関連付けられている関数ライブラリにユーザ定義関数が追加されている場合にのみ使用できます。詳細については，89 ページ「関数ライブラリの管理」および 581 ページ「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）。

本項では，次の項目について説明します。

- ▶ 157 ページ「[項目] リストからのテスト・オブジェクトの選択」
- ▶ 158 ページ「共有オブジェクト・リポジトリからのテスト・オブジェクトの選択」
- ▶ 161 ページ「アプリケーションからのテスト・オブジェクトの選択」
- ▶ 164 ページ「[操作] 項目の選択」

項目を選択した後，その項目に対する操作を指定します。詳細については，164 ページ「ステップの操作の選択」を参照してください。

注：[項目]セルで項目または[操作]を選択するほかに，[コメント]を選択することもできます。[コメント]を選択すると，選択したステップが行全体にまたがる自由テキスト・セルに変換されます。ステップがコメントに変換された後，そのコメントをステップに戻すことはできません。手動ステップの入力，または隣接するステップに関する情報の提供を行う際に，[コメント]オプションを使用します。詳細については，180 ページ「コメントの使用」を参照してください。

[項目] リストからのテスト・オブジェクトの選択

[項目] リストに表示されるテスト・オブジェクトは，前のステップのテスト・オブジェクトの兄弟テスト・オブジェクトまたは子テスト・オブジェクトです（共有オブジェクト・リポジトリで定義されています）。次の例は，**userName** テスト・オブジェクトの後に続くステップで使用できるオブジェクトを示しています。



表示された [項目] リストからテスト・オブジェクトを選択するには，次の手順を実行します。

- 1 [項目]セルをクリックし，矢印ボタンをクリックして，[項目] リストを表示します。新しいステップを作成した直後であれば，新しいステップを作成するとすぐに，リストが自動的に表示されます。
- 2 [項目] リストで，ステップの実行対象にするテスト・オブジェクトを選択します。選択した項目が [項目] セルに表示されます。ここで，ステップの操作を指定する必要があります。詳細については，164 ページ「ステップの操作の選択」を参照してください。

共有オブジェクト・リポジトリからのテスト・オブジェクトの選択

共有オブジェクト・リポジトリには、コンポーネントの基本となっているアプリケーション領域で定義されたテスト・オブジェクトがすべて含まれています（前述の「**項目**」リストに表示されるものも含まれます）。

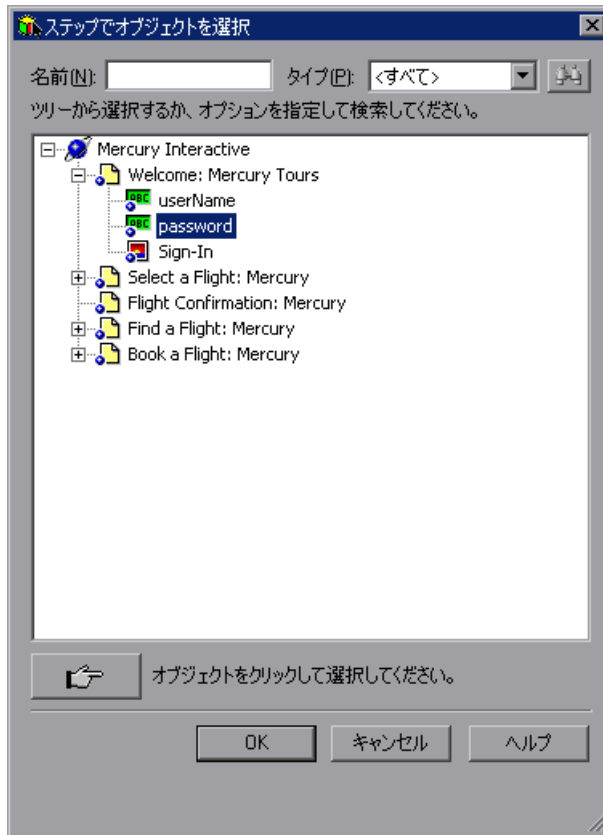
新しいステップのオブジェクト・リポジトリ・ツリーでは任意のオブジェクトを選択できます。オブジェクト・リポジトリが非常に大きい場合、オブジェクトを検索できます。たとえば、**password** オブジェクトを追加するとします。このオブジェクトはエディット・ボックスであることがわかっています。オブジェクトのタイプが「**Edit**」であるすべてのオブジェクトを、「**password**」という名前で検索することも、「**p**」という文字で検索することもできます。

オブジェクト・リポジトリの詳細については、第13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。**Object** ステートメントの詳細については、657 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。

共有オブジェクト・リポジトリからテスト・オブジェクトを選択するには、次の手順を実行します。

- 1 **[項目]** セルをクリックし、矢印ボタンをクリックして、**[項目]** リストを表示します。新しいステップを作成した直後であれば、新しいステップを作成するとすぐに、リストが自動的に表示されます。

- 2 [項目] リストで, [別オブジェクトの選択] を選択します。[ステップでオブジェクトを選択] ダイアログ・ボックスが開きます。



- 3 オブジェクト・リポジトリ・ツリーからオブジェクトを選択します。オブジェクト・リポジトリが非常に大きい場合, 次の説明に従ってオブジェクトを検索できます。検索をしない場合は, 手順 8 に進みます。
- 4 [名前] ボックスに, オブジェクトの名前, または名前の一部を入力します。たとえば, 名前に p という文字が含まれているオブジェクトをすべて検索する場合は, 「p」と入力します。

注：[名前] ボックスが空白のままの場合は、選択したオブジェクトのタイプのすべてのオブジェクトが条件と一致しているとみなされます。

- 5 [タイプ] ボックスの中で、検索するオブジェクトのタイプを選択します。すべてのタイプのオブジェクトを対象に検索を行うには、[すべて] を選択します。
-

注：このリストに示される各オブジェクト・タイプは、オブジェクトを一般的な特性に基づいてグループ分けした一般的な分類です。たとえば、[List] タイプには、リスト・オブジェクトやリスト・ビュー・オブジェクトに加え、コンボ・ボックスが含まれます。[Table] タイプには、テーブルとグリッドの両方が含まれ、[Miscellaneous] タイプには WebElement と WinObject などさまざまなオブジェクトが含まれます。



- 6 [次を検索] ボタンをクリックします。現在選択されているノードから検索が開始され、条件に一致したオブジェクトの数が表示されます。リスト内で条件に一致した最初のオブジェクトが強調表示されます。
- 7 必要に応じて [次を検索] ボタンをクリックすることで、検索条件に一致したすべてのオブジェクトを順に確認していくことができます。検索はツリーの末尾まで続き、それから、ツリーの先頭に戻って続行されます。
-

ヒント：検索条件に一致する次のオブジェクトを見つけるには、F3 キーを押します。前の一致を見つけるには、SHIFT+F3 キーを押します。

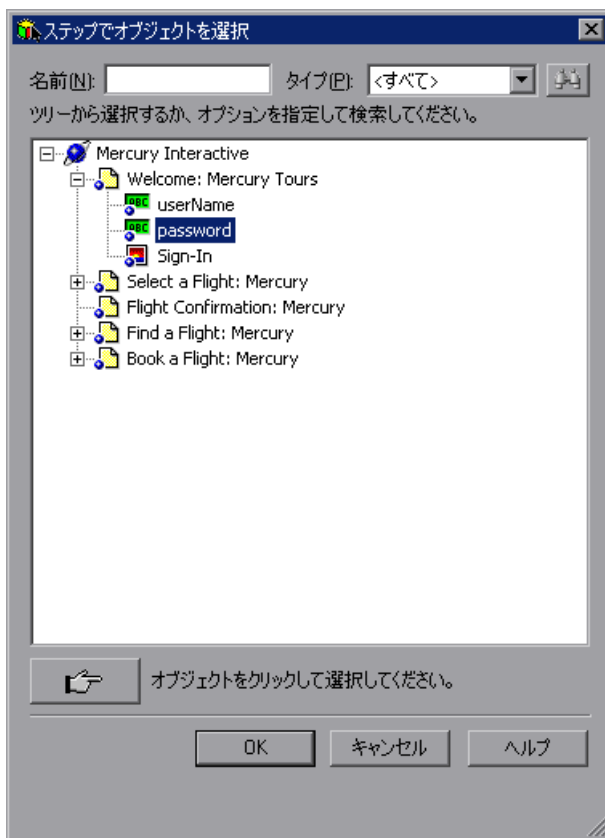
- 8 [OK] をクリックします。オブジェクトがキーワード・ビューの [項目] カラムに表示され、[項目] リストにも追加されます。これで、選択したオブジェクトに対する操作を指定できます。詳細については、164 ページ「[操作] 項目の選択」を参照してください。

アプリケーションからのテスト・オブジェクトの選択

ステップに必要なテスト・オブジェクトが共有オブジェクト・リポジトリに含まれていない場合は、アプリケーションから直接選択して共有オブジェクト・リポジトリに追加すれば、当該のステップやほかのステップでそのテスト・オブジェクトを使用できるようになります。

アプリケーションからテスト・オブジェクトを追加するには、次の手順を実行します。

- 1 [項目] セルをクリックし、矢印ボタンをクリックして、[項目] リストを表示します。新しいステップを作成した直後であれば、新しいステップを作成するとすぐに、リストが自動的に表示されます。
- 2 [項目] リストで、[別オブジェクトの選択] を選択します。[ステップでオブジェクトを選択] ダイアログ・ボックスが開きます。

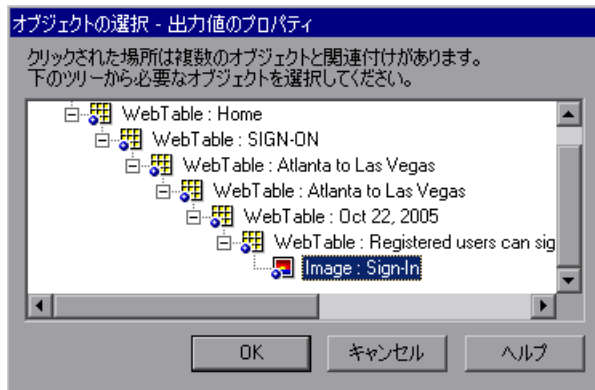




- 3 指差しマークのボタンをクリックします。QuickTest が最小化されます。
- 4 指差しマークを使用して、アプリケーション内の必要なオブジェクトをクリックします。

ヒント：左側の CTRL キーを押しながら、ウィンドウのフォーカスを変更したり、右クリックやマウスオーバーによってショートカット・メニューを表示するなどの操作を実行できます。クリック対象オブジェクトが部分的に別のウィンドウに隠れている場合には、部分的に隠れているウィンドウに指差しボタンを数秒間置いたままにすればウィンドウが前面に表示されるため、必要なオブジェクトをポイントしてクリックすることができます。さらに、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、左の CTRL キーを押しながら Windows タスク・バーのアプリケーションを右クリックし、ショートカット・メニューから **[元のサイズに戻す]** を選択することによって再表示できます。

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] ダイアログ・ボックスが開きます。



- 5 新しいステップのオブジェクトを選択し、[OK] をクリックします。オブジェクトが、[ステップでオブジェクトを選択] ダイアログ・ボックスの共有オブジェクト・リポジトリ・ツリーに表示されます。

- 6 [OK] をクリックします。オブジェクトがキーワード・ビューの [項目] カラムに表示されます。これで、選択したオブジェクトに対する操作を指定できます。詳細については、164 ページ「ステップの操作の選択」を参照してください。

注：Quality Center を使用している各分野のエキスパートは、コンポーネントのアプリケーション領域にある共有オブジェクト・リポジトリに格納されているオブジェクトのみ選択できます。

ヒント：共有オブジェクト・リポジトリ内に存在しないアプリケーションのオブジェクトを選択した場合、新しいステップを挿入すると、テスト・オブジェクトがローカル・オブジェクト・リポジトリに追加されます。新規テスト・オブジェクトをローカル・オブジェクト・リポジトリに追加した後、その名前が用途を明確に表していない場合は名前を変更することをお勧めします。たとえば、Edit（ユーザ名の入力に使用）という名前を **UserName** に変更できます。これにより、各分野のエキスパートは、共有オブジェクト・リポジトリに含まれるテスト・オブジェクトを使用するステップを追加する場合に適切なテスト・オブジェクトを選択できます。

必要なオブジェクトをローカル・オブジェクト・リポジトリに追加したら、[オブジェクト・リポジトリ 結合ツール] を使用して共有オブジェクト・リポジトリを更新して、他のコンポーネントが新しいオブジェクトを使用できるようにします。詳細については、443 ページ「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」を参照してください。

コンテナ・テスト・オブジェクトを追加している場合、コンテキストを指定する方法もお勧めします。たとえば、ログイン・ページからの確認メッセージを追加する場合、「ログイン>確認」という名前を付けます。詳細については、327 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。

【操作】項目の選択

ビジネス・コンポーネントが、少なくとも1つの関数ライブラリを参照しているアプリケーション領域に基づいている場合、**【操作】**項目を選択して、ステップで使用する関数を選択できます。

ユーザ定義関数を使用すると、さまざまな追加操作を実行できます。たとえば、ビジネス・コンポーネントの開始時点でアプリケーションを開いたり、特定のプロパティの値をチェックしたりできます。

注：コンポーネントが基盤にしているアプリケーション領域が関数ライブラリに関連付けられていない場合には、**【項目】**リストに**【操作】**オプションは表示されません。

【操作】項目を選択するには、次の手順を実行します。

- 1 **【項目】**セルをクリックし、矢印ボタンをクリックして、**【項目】**リストを表示します。新しいステップを入力した直後である場合は、新しいステップを作成するとすぐに、リストが自動的に表示されます。
- 2 **【項目】**リストで、**【操作】**を選択します。**【操作】**項目が**【項目】**セルに表示されます。ここで、ステップの操作を指定する必要があります。詳細については、164 ページ「ステップの操作の選択」を参照してください。

ステップの操作の選択

【操作】セルでは、**【項目】**カラムに表示された項目に対して実行する操作を指定します。使用可能な操作は、**【項目】**カラムで選択した項目に応じて異なります。項目を選択すると、その項目に（アプリケーション領域を通じて）関連する操作（キーワード）がすべて表示されます。

たとえば、WebButton オブジェクトのようなブラウザ・テスト・オブジェクトを選択した場合、表示される一覧には、コンポーネントのアプリケーション領域のキーワード表示枠内の使用可能なキーワードの一覧から、WebButton オブジェクト用に選択されたすべてのメソッドが含まれます。[項目] カラムで [操作] を選択した場合、一覧にはビジネス・コンポーネントに（アプリケーション領域を通じて）関連付けられている関連ライブラリ・ファイルの中で定義されたユーザ定義関数が表示されます。

関数ライブラリは、[アプリケーション領域] の関数ライブラリ表示枠で指定します。詳細については、89 ページ「関数ライブラリの管理」を参照してください。

ステップの操作を選択するには、次の手順を実行します。

[操作] セルをクリックし、矢印ボタンをクリックして、項目に対して実行する操作を選択します。操作として、標準の操作またはユーザ定義関数を指定できます。ユーザ定義関数の詳細については、581 ページ「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。

注： リスト内の操作にカーソルを合わせると、その操作の実行内容を説明するツールチップが表示されます。ユーザ定義関数の場合、ツールチップは関連付けられている関数ライブラリに入力された注釈から取得されます。詳細については、606 ページ「関数への説明の追加」を参照してください。




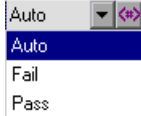
[項目] カラムでテスト・オブジェクトを選択すると、[操作] カラムには、そのテスト・オブジェクト項目の選択されたすべての操作が（アプリケーション領域のキーワード表示枠での定義に従って）自動的に表示されます。選択したテスト・オブジェクトの [操作] リストには、製品にあらかじめ用意されている操作と、そのテスト・オブジェクト・タイプに登録されているユーザ定義関数が含まれます。

[項目] カラムで [操作] を選択すると、関数ライブラリ中に定義されている関数がアルファベット順に表示されます（コンポーネントの関数ライブラリの管理は、[アプリケーション領域] の関数ライブラリ表示枠で行います。詳細については、89 ページ「関数ライブラリの管理」を参照してください）。

ステップの引数の値の定義

[値]セルには、操作の引数の値が一覧表示されます。このセルには、定数値またはパラメータを挿入できます。パラメータを挿入する場合は、ローカル・パラメータまたはコンポーネント・パラメータを挿入できます。詳細については、172 ページ「パラメータの使用」および 239 ページ「コンポーネントのパラメータの定義」を参照してください。パスワードの値を暗号化することもできます。詳細については、168 ページ「メソッド引数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。


[値]セルは、選択した操作の引数の数に従って分割されます。次のように、パーティションに入力可能な引数のタイプに従って、各パーティションにはさまざまなオプションが含まれています。

引数のパーティション	引数のタイプ	説明
	String	引用符で囲まれた任意の英数字文字列を入力できます。引用符を入力しなかった場合、QuickTestにより引用符が自動的に追加されます。セル内の引用符で囲まれた文字列を修正して引用符を削除した場合、引用符が自動的に復元されることはなく、値は変数名として扱われます。
	Integer	任意の数値を入力できます。また上下の矢印を使用して数値を選択できます。
	Boolean	リストから [True] または [False] を選択できます。
	Predefined Constant	リストから定義済み変数を選択できます。

SHIFT+ENTER キーを押して引数値に改行を追加すれば、複数行の引数値を追加することもできます。複数行の引数値を入力すると、その引数値が自動的に文字列に変換され、引数の最初の行とそれに続く省略記号 (...) のみ表示されます。キーワード・ビューの [注釈] カラムでも、複数行の引数値はこの形式で表示されます。

"Readability:good..." 暗号化された文字列 "Readability:good.." を "positive" edit box に入力する。

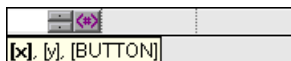
ヒント：ステップの中で使用される引数値の全体を表示するには、セルを選択します。引数値は、ステップ内に表示されるとおりに実行セッション中に使用されます。たとえば、引数値の一部として引用符を入力した場合、その引用符は実行セッション中に使用される引数値にも含まれます。QuickTest では、複数行の値は自動的に文字列と解釈されます。そのため、引用符を追加する必要はありません。

[値] セルで  ボタンをクリックすれば、ローカル・パラメータまたはコンポーネント・パラメータを使用して、引数の値をパラメータ化できます。ローカル値のパラメータ化の詳細については、172 ページ「パラメータの使用」を参照してください。コンポーネント値のパラメータ化の詳細については、239 ページ「コンポーネントのパラメータの定義」を参照してください。

値を定義または変更するには、次の手順を実行します。

[値] セルの各区画をクリックして、選択した操作の引数値を入力します。

[値] セルをクリックすると、ツールチップに各引数の情報が表示されます。ツールチップには、現在強調表示されているパーティションの引数が太字で表示され、オプションの引数があれば大括弧で囲んで表示されます。

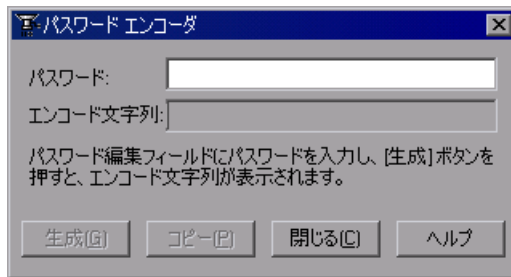


メソッド引数への暗号化されたパスワードの挿入

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として使用できます。たとえば、ユーザがパスワードを入力しなければならないフォームが Web サイトにあるとします。異なるパスワードにサイトがどのように応答するかをテストしたいが、同時にパスワードの安全性も確保したいとします。**パスワード・エンコーダ**を使用すればパスワードを暗号化できます。

パスワードを暗号化するには、次の手順を実行します。

- 1 Windows メニューの [スタート] メニューから [QuickTest Professional] プログラム・グループを開き、[Tools] > [Password Encoder] を選択します。[パスワードエンコーダ] ダイアログ・ボックスが開きます。



- 2 [パスワード] ボックスにパスワードを入力します。
- 3 [生成] をクリックします。パスワード・エンコーダによってパスワードが暗号化され、暗号化された値が [エンコード文字列] フィールドに表示されます。
- 4 [コピー] ボタンを使用して、暗号化された値をコピーし、[値] セルに貼り付けます。
- 5 暗号化したいパスワードごとに、この手順を繰り返します。
- 6 [閉じる] をクリックして、パスワード・エンコーダを閉じます。

ステップの出力値の定義

出力値の出力タイプおよび設定は、[出力]セルで定義します。この定義によって、出力値の保存場所と、コンポーネントの実行セッションにおいて出力値がどのように使用されるかが決まります。出力値のステップに到達すると、QuickTestによって、出力対象として選択された値が取得され、後の実行セッションで使用できるように指定された場所に保存されます。


新しい出力値ステップを作成すると、出力対象として選択されたそれぞれの値に標準設定の定義が割り当てられます。ビジネス・コンポーネントのステップの値を出力すると、次のようになります。

- ▶ コンポーネントで出力コンポーネント・パラメータが1つ以上定義されている場合、標準設定の出力タイプは「**コンポーネントパラメータ**」で、標準設定の出力名は[ビジネス コンポーネントの設定]ダイアログ・ボックスの[パラメータ]タブに表示されている最初の出力パラメータです。
- ▶ コンポーネントで出力コンポーネント・パラメータが定義されていない場合、標準設定の出力タイプは「**ローカルパラメータ**」で、標準設定の出力名は「**p_Local**」です。

出力パラメータは必要に応じて変更します。ローカル・パラメータを選択した場合は、[出力オプション]ダイアログ・ボックスで出力パラメータを直接変更できます。コンポーネント・パラメータを選択した場合、出力パラメータの詳細は読み取り専用となります。パラメータの詳細は、[ビジネス コンポーネントの設定]ダイアログ・ボックスの[パラメータ]タブで変更できます。詳細については、239 ページ「コンポーネントのパラメータの定義」を参照してください。

出力値を指定した後で出力値を保存しないことに決めた場合は、指定をキャンセルできます。詳細については、171 ページ「パラメータへの出力のキャンセル」を参照してください。

パラメータへの出力を設定するには、次の手順を実行します。



- 1 パラメータへの出力を作成または編集するには、[出力] セルをクリックします。[出力パラメータの指定] ボタン  をクリックするか、CTRL + F11 キーを押します。[出力オプション] ダイアログ・ボックスが開きます。



- 2 [出力タイプ] ボックスで、「コンポーネント パラメータ」または「Local Parameter」を選択します。[詳細] 領域には、選択したコンポーネント・タイプで使用可能なオプションが表示されます。

注：「コンポーネント・パラメータ」タイプを選択できるのは、そのコンポーネントについて出力コンポーネント・パラメータが定義されている場合だけです。コンポーネント・パラメータを選択すると、情報が読み取り専用で表示されます。詳細については、239 ページ「コンポーネントのパラメータの定義」を参照してください。


- 3 **[名前]** ボックスから、必要なパラメータを選択します。ローカル・パラメータが1つも定義されていない場合は、標準設定のパラメータ名として **p_Local** が表示されます。
- ▶ 必要に応じて、新しいローカル・パラメータを作成できます。詳細については、172 ページ「パラメータの使用」を参照してください。
 - ▶ ローカル・パラメータを選択した場合は、その詳細を指定します。詳細については、172 ページ「パラメータの使用」を参照してください。
 - ▶ コンポーネント・パラメータを選択した場合、その詳細は読み取り専用で表示されます。
- 4 **[OK]** をクリックします。**[出力]** セルに、出力値の保存先となるパラメータが表示されます。

ヒント：出力パラメータを指定した後でその **[出力]** セルをクリックすると、パラメータのタイプを示す次のようなアイコンがセル内に表示されます。 アイコンはコンポーネント・パラメータを示します。 アイコンは、ローカル・パラメータを示します。

パラメータへの出力のキャンセル

コンポーネント・ステップの出力値を保存しない場合は、出力をキャンセルできます。

パラメータへの出力をキャンセルするには、次の手順を実行します。

[出力] セルをクリックします。**[削除]** ボタン  をクリックするか、DELETE キーを押して、パラメータへの出力を取り消します。

パラメータの使用

ビジネス・コンポーネントに値を渡す入力パラメータと、コンポーネントから外部のソースに、またはステップから別のステップに値を渡す出力パラメータを定義できます。定義したパラメータを使用すれば、ステップの入力値と出力値をパラメータ化できます。

定義できるパラメータには、「ローカル・パラメータ」と「コンポーネント・パラメータ」の2種類があります。

ローカル・パラメータ：同じコンポーネント内での使用するために、コンポーネント内で定義される変数値。

ローカルの入力パラメータ値は、同じコンポーネント内の以降のパラメータ化されたステップで受け取って使用できます。ローカルの出力パラメータは、同じコンポーネント内での使用のために、操作またはコンポーネント・ステップによって返されます。ローカルのパラメータ出力値は、ビジネス・プロセスのテスト結果で確認できます。

ローカル・パラメータはビジネス・コンポーネント・キーワード・ビューで定義します。入力パラメータの定義には [設定値オプション] ダイアログ・ボックス、出力パラメータの定義には [出力オプション] ダイアログ・ボックスを使用します。ローカル・パラメータは削除できませんが、そのパラメータへの入力または出力をキャンセルすることはできます。

コンポーネント・パラメータ：ビジネス・プロセス・テスト内の同じコンポーネントまたは以降のコンポーネントで使用するために、コンポーネント内で定義される変数値。

コンポーネント入力パラメータ値は、コンポーネント内の特定のパラメータ化されたステップに対する値として受け取って使用できます。コンポーネント出力パラメータ値は、以降のテストで使用するコンポーネントの入力パラメータとして返されます。これらの値は、ビジネス・プロセス・テスト結果でも確認することができます。

コンポーネント・パラメータは、[ビジネス コンポーネントの設定] ダイアログ・ボックスの [パラメータ] タブ、または Quality Center ビジネス・コンポーネント・モジュールで定義できます。

本項では、ローカル・パラメータを設定する方法と、ローカル・パラメータおよびコンポーネント・パラメータを使って入力値と出力値をパラメータ化する方法を説明します。コンポーネント・パラメータの設定については、239 ページ「コンポーネントのパラメータの定義」を参照してください。

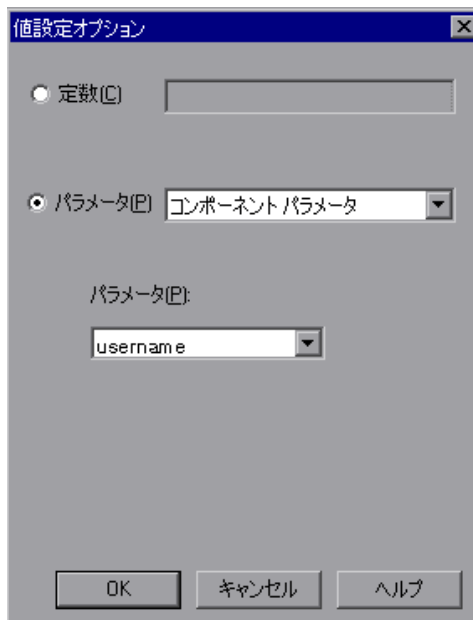
パラメータを定義したら、そのパラメータを使用して値をパラメータ化できます。また、定数値を [値] セルに直接入力することで、パラメータに定数値も適用できます。

入力値のパラメータ化

[値] セルでは、ローカル・パラメータまたはコンポーネント・パラメータを使って、ステップの入力値をパラメータ化できます。

ローカル・パラメータを使って入力値をパラメータ化するには、次の手順を実行します。

- 1 [値] セルで、パラメータ化ボタン  をクリックするか、CTRL+F11 キーを押します。[値設定オプション] ダイアログ・ボックスが開きます。



注：コンポーネントで入力コンポーネント・パラメータが1つ以上定義されている場合、標準設定の入力タイプは「**コンポーネントパラメータ**」で、標準設定の入力名は [ビジネス コンポーネントの設定] ダイアログ・ボックスの [パラメータ] タブに表示されている最初の入力パラメータです。

- 2 [パラメータ] ボックスで、[ローカルパラメータ] を選択します。ローカル・パラメータ・タイプの詳細が表示されます。

The screenshot shows a configuration window for a parameter. At the top, there is a radio button labeled 'パラメータ(P)' and a dropdown menu showing 'Local Parameter'. Below this is a section titled '詳細' (Details) containing three fields: '名前(N):' with a dropdown menu showing 'p_Text', '値(V):' with an empty text box, and '記述(D):' with an empty text box.

- 3 ローカル・パラメータのプロパティの詳細を指定します。
- ▶ [名前]：パラメータに対する分かりやすい名前を入力するか、リストから名前を選択します。
 - ▶ [値]：パラメータに対する入力値を入力します。値を指定しなかった場合は、次の表に示す標準設定の値が自動的に割り当てられます。

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
数値	0
パスワード	空文字列

- ▶ [記述]：パラメータに関する簡単な説明を入力します。

- 4 [OK] をクリックします。ステップの [値] セルにローカル・パラメータが表示されます。コンポーネントの実行では、ステップのパラメータに指定された値が使用されます。

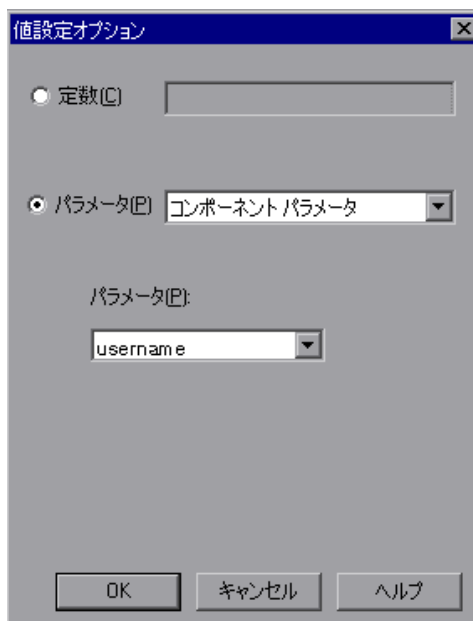
ヒント：

値のパラメータ化をキャンセルするには、[値設定オプション] ダイアログ・ボックスで [定数] オプションを選択して、定数値を入力します。

ローカル・パラメータの定義後に [値] セルをクリックすると、ローカル・パラメータが定義されたセルの各区画に  アイコンが表示されます。

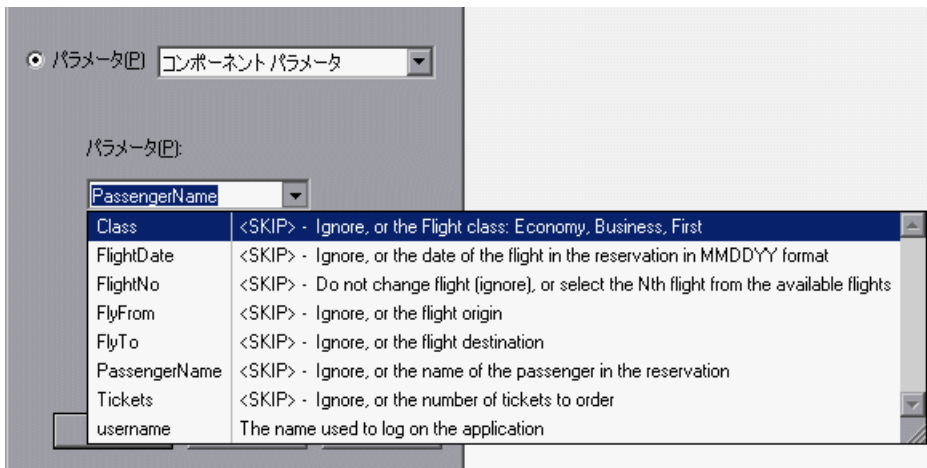
コンポーネント・パラメータを使って入力値をパラメータ化するには、次の手順を実行します。

- 1 [値] セルで、パラメータ化ボタン  をクリックするか、CTRL + F11 キーを押します。[値設定オプション] ダイアログ・ボックスが開きます。



注：コンポーネントで入力コンポーネント・パラメータが1つ以上定義されている場合、標準設定の入力タイプは「**コンポーネントパラメータ**」で、標準設定の入力名は [ビジネス コンポーネントの設定] ダイアログ・ボックスの [パラメータ] タブに表示されている最初の入力パラメータです。コンポーネント・パラメータが定義されていない場合、コンポーネント・パラメータを使って入力値をパラメータ化するためには、コンポーネント・パラメータを定義する必要があります。詳細については、239 ページ「コンポーネントのパラメータの定義」を参照してください。


- 2 [パラメータ] ボックスで、パラメータ化される値に使用するコンポーネント・パラメータを選択します。使用可能なコンポーネント・パラメータの名前と詳細な説明が読み取り専用で表示されます。必要に応じて表示サイズを変更できます。また、パラメータのリストが長ければ、リストをスクロールできます。



- 3 [OK] をクリックします。ステップの [値] セルにコンポーネント・パラメータが表示されます。コンポーネントの実行では、ステップのパラメータに指定された値が使用されます。

ヒント：


値のパラメータ化をキャンセルするには、[値設定オプション] ダイアログ・ボックスで **[定数]** オプションを選択して、定数値を入力します。

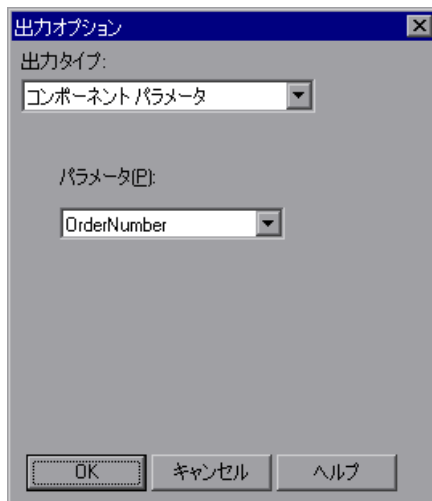
コンポーネント・パラメータの定義後に **[値]** セルをクリックすると、コンポーネント・パラメータが定義されたセルの各区画に  アイコンが表示されます。

出力値のパラメータ化

ステップの出力値は、ステップの **[出力]** セルでローカル・パラメータまたはコンポーネント・パラメータを使用してパラメータ化できます。出力パラメータ値は、コンポーネント内の後続のステップやビジネス・プロセス・テスト内の後続のコンポーネントで入力値として使用できます。

ローカル・パラメータを使って出力値をパラメータ化するには、次の手順を実行します。

- 1 **[出力]** セルで、[出力パラメータの指定] ボタン  をクリックするか、CTRL + F11 キーを押します。[出力オプション] ダイアログ・ボックスが開きます。




注：コンポーネントで出力コンポーネント・パラメータが1つ以上定義されている場合、標準設定の出力タイプは「**コンポーネントパラメータ**」で、標準設定の出力名は [ビジネス コンポーネントの設定] ダイアログ・ボックスの [パラメータ] タブに表示されている最初の出力パラメータです。

- 2 [出力タイプ] ボックスで、[Local Parameter] を選択します。ローカル・パラメータ・タイプの詳細が表示されます。




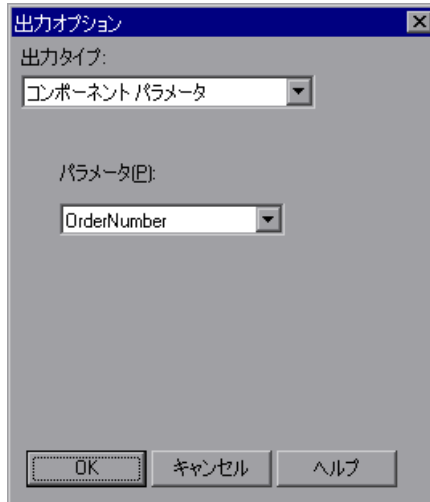
The image shows a dialog box titled '詳細' (Details). It contains two fields: '名前(N):' (Name) with a dropdown menu showing 'p_Local', and '記述(D):' (Description) with an empty text input field.

- 3 ローカル・パラメータのプロパティの詳細を指定します。
- ▶ [名前]：パラメータに対する分かりやすい名前を入力するか、リストから名前を選択します。
 - ▶ [記述]：パラメータに関する簡単な説明を入力します。
- 4 [OK] をクリックします。ステップの [出力] セルにローカル・パラメータが表示されます。コンポーネントは実行時に、ステップに対して指定された出力パラメータに値を出力します。

ヒント：ローカル・パラメータの定義後に [出力] セルをクリックすると、ローカル・パラメータが定義されたセルの各区画に  アイコンが表示されます。

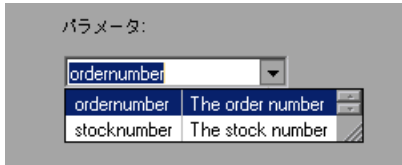
コンポーネント・パラメータを使って出力値をパラメータ化するには、次の手順を実行します。

- 1 [出力] セルで、[出力パラメータの指定] ボタン  をクリックするか、CTRL + F11 キーを押します。[出力オプション] ダイアログ・ボックスが開きます。




注：コンポーネントで出力コンポーネント・パラメータが1つ以上定義されている場合、標準設定の出力タイプは「**コンポーネントパラメータ**」で、標準設定の出力名は [ビジネス コンポーネントの設定] ダイアログ・ボックスの [パラメータ] タブに表示されている最初の出力パラメータです。コンポーネント・パラメータが定義されていない場合、コンポーネント・パラメータを使って出力値をパラメータ化するためには、コンポーネント・パラメータを定義する必要があります。詳細については、239 ページ「コンポーネントのパラメータの定義」を参照してください。


- 2 [パラメータ] ボックスで、出力値を格納するコンポーネント・パラメータを選択します。使用可能なコンポーネント・パラメータの名前と詳細な説明が読み取り専用で表示されます。必要に応じて表示サイズを変更できます。また、パラメータのリストが長ければ、リストをスクロールできます。



- 3 [OK] をクリックします。ステップの [出力] セルにコンポーネント・パラメータが表示されます。コンポーネントは実行時に、ステップに対して指定された出力パラメータに値を出力します。

ヒント：ローカル・パラメータの定義後に [値] セルをクリックすると、ローカル・パラメータが定義されたセルの各区画に  アイコンが表示されます。

コメントの使用

コメントは、ビジネス・コンポーネントに自由に入力できるテキスト・エントリです。キーワード・ビューでは、コメントに  アイコンが付いています。コメントにはいくつかの用途があります。たとえば、アプリケーションがテストできる状態になる前に、ビジネス・コンポーネントに含めるステップを計画しなければならない場合があります。その後、アプリケーションの準備が整ったときに、その計画を使用して、テストする必要のある項目がすべてコンポーネント・ステップに含まれているかどうかをチェックできます。

ビジネス・コンポーネントをわかりやすくして更新しやすくするために、コメントを付け加えることもあります。たとえば、コンポーネントの各セクションの前に、そのセクションに何が含まれているかを説明するコメントを追加します。


追加したコメントは、1つ以上のカラムが表示されてさえいれば、常にコンポーネント内に表示されます。表示するカラムを選択する方法については、32ページ「キーワード・ビューの表示オプションの設定」を参照してください。また、グリッドを横方向にスクロールした場合も、コメントは常に表示されません。QuickTest がビジネス・コンポーネントを実行するときに、コメントは処理されません。

注： 挿入したコメントをステップに変更することはできません。

コンポーネントにコメントを追加するには、次の手順を実行します。

- 1 **[挿入]** > **[コメント]** を選択するか、**[項目]** セルをクリックして表示されたリストから **[コメント]** を選択するか、コンポーネント・ステップを右クリックして **[コメントの挿入]** を選択します。選択したステップの下にコメント行が追加されます。
- 2 コメント行にテキストを入力します。テキストを入力しなかった場合、そのコメントは、カーソル・フォーカスが別の場所に移動したときに自動的に削除されます。

既存のコメントを変更するには、次の手順を実行します。

コメントをダブルクリックします。テキスト・ボックスが自由テキスト・フィールドになります。代わりに  アイコンをクリックするという方法もあります。このアイコンは、コメントの状態を編集可能と読み取り専用の間で切り替えます。

コメントを削除するには、次の手順を実行します。

- 1 コメントを選択し、**[編集]** > **[削除]** を選択するか、キーボードの DELETE キーを押します。または、コメントを右クリックしてショートカット・メニューから **[削除]** を選択します。
- 2 **[コメントの削除]** をクリックして確定します。コメントがビジネス・コンポーネントから完全に削除されます。

コンポーネントのステップの管理

コンポーネントのステップは、コンポーネント内の別のステップやコメントの前または後に移動できます。また、不要になったステップは削除できます。

コンポーネントのステップの移動

ステップをコンポーネント内の別の場所に移動できます。

ステップをコンポーネント内で移動するには、次の手順を実行します。

- ▶ **[項目]** カラムで、ステップを上または下にドラッグし、適切な位置でドロップします。
- ▶ ステップをクリップボードにコピーするか切り取って、目的の場所に貼り付けます。ステップをコピーするには **[編集]** > **[コピー]** か CTRL + C キーを使用し、ステップを切り取るには **[編集]** > **[切り取り]** か CTRL + X キーを使用します。ステップを貼り付けるには、**[編集]** > **[貼り付け]** か CTRL + V キーを使用します。

コンポーネントのステップの削除

コンポーネントのステップは、必要に応じて削除できます。ステップを削除する前に、その削除がコンポーネントの正常な動作を妨げる原因にならないことを確認してください。

注：いずれかのセルが編集モードの場合、ステップは削除できません。

ステップを削除するには、次の手順を実行します。

- 1 削除するステップを選択し、**[編集]** > **[削除]** を選択するか、DELETE キーを押します。または、ステップを右クリックしてショートカット・メニューから **[削除]** を選択します。警告メッセージが表示されます。
- 2 **[ステップの削除]** をクリックして確定します。ステップがコンポーネントから削除されます。

第7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・リポジトリなど）がコンポーネントのアプリケーション領域内にある場合や、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。必要に応じて、欠落リソースを割り当てるか、またはコンポーネントのアプリケーション領域からそれを削除できます。

注：本章で説明す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、テストは失敗します。

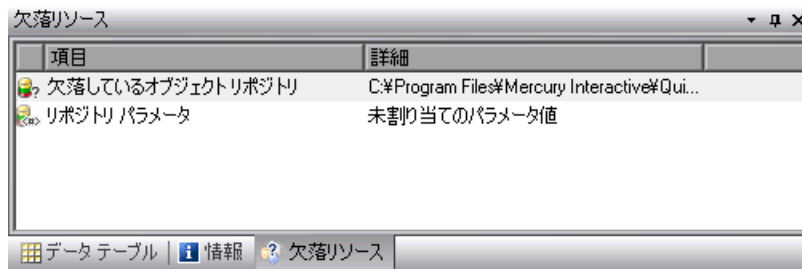
本章では、次の内容について説明します。

- ▶ 欠落リソースの処理について
- ▶ 欠落している共有オブジェクト・リポジトリの処理
- ▶ 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

欠落リソースの処理について

QuickTest では、コンポーネントまたはアプリケーション領域を開くたびに、そのコンポーネントまたはアプリケーション領域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。具体的には、関連付けられているすべての共有オブジェクト・リポジトリがあるかどうかおよび定義されているすべてのリポジトリ・パラメータに値があるかどうかを確認されます。

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が表示され、このリストを使ってコンポーネントまたはアプリケーション領域内の該当するリソースの再割り当てまたは削除を行うことができます。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。

[欠落しているオブジェクトリポジトリ]：コンポーネントまたはアプリケーション領域が欠落している共有オブジェクト・リポジトリに関連付けられている場合は、欠落しているオブジェクト・リポジトリを探すためのパスが示されます。詳細については、186 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」を参照してください。

[リポジトリパラメータ]：コンポーネントまたはアプリケーション領域内の少なくとも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、187 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。

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]** : コンポーネントまたはアプリケーション領域内のすべての欠落リソースが表示されます。
- ▶ **[オブジェクト 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パラメータ化されたプロパティ値が少なくとも1つあることを示す一般的な行が表示されます。
- ▶ **[欠落しているオブジェクト 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1つの行が表示され、その共有オブジェクト・リポジトリを探すためのパスが指定されます。

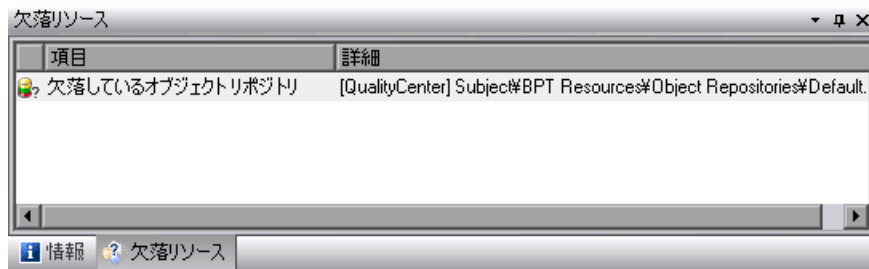
選択したリソース・タイプに従って欠落リソース表示枠がフィルタ処理され、表示枠の下部に適用したフィルタが示されます。

 [= 'Missing Object Repository']

ヒント : 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・リポジトリをアプリケーション領域に関連付けると、指定したリポジトリがアクセス可能かどうかを確認されます。また、コンポーネントまたはアプリケーション領域を開くたびに、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、コンポーネントまたはアプリケーション領域を開いたときに、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。



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・リポジトリを関連付けられたアプリケーション領域に割り当てる必要があります。

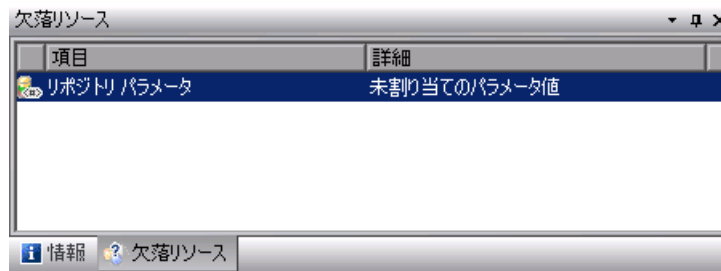
コンポーネントの場合は、欠落しているオブジェクト・リポジトリが表示された行をダブルクリックすると、そのオブジェクト・リポジトリを関連付けられたアプリケーション領域に割り当てる必要があることを示すメッセージが表示されます。この場合、ユーザまたはオートメーション・エンジニアがアプリケーション領域を開き、[オブジェクトリポジトリ]表示枠で共有オブジェクト・リポジトリの関連付けを修正する必要があります。

アプリケーション領域の場合は、欠落しているオブジェクト・リポジトリが表示された行をダブルクリックすると、アプリケーション領域の [オブジェクトリポジトリ]表示枠が開き、必要に応じてオブジェクト・リポジトリの関連付けを修正したり、削除したりできます。詳細については、94 ページ「共有オブジェクト・リポジトリの管理」を参照してください。

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

コンポーネン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、またはコンポーネン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、413 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コンポーネントを開くと、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。この値をローカル・パラメータまたはコンポーネント・パラメータに割り当てたり、定数値を定義したりできます。

[リポジトリ パラメータ] が表示された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、未割り当てのオブジェクト・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、指定されたコンポーネント内の（値が割り当て済みまたは未割り当ての）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、187 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。

第 3 部

設定

第 8 章

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 が特定のコンポーネントを記録したり、実行したりする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
- ▶ [オプション] ダイアログ・ボックスの使用
- ▶ テストの一般オプションの設定
- ▶ テストのフォルダ・オプションの設定
- ▶ テストの実行オプションの設定
- ▶ テストの Windows アプリケーション・オプションの設定
- ▶ テストの Web オプションの設定

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、コンポーネン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 [ようこそ] 画面を表示しないようにしたり、QuickTest がコンポーネン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、すべてのコンポーネントおよび以降のテスト・セッションで有効になります。

また、QuickTest で現在開いているコンポーネン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 9 章「ビジネス・コンポーネント設定にかかわる作業」を参照してください。

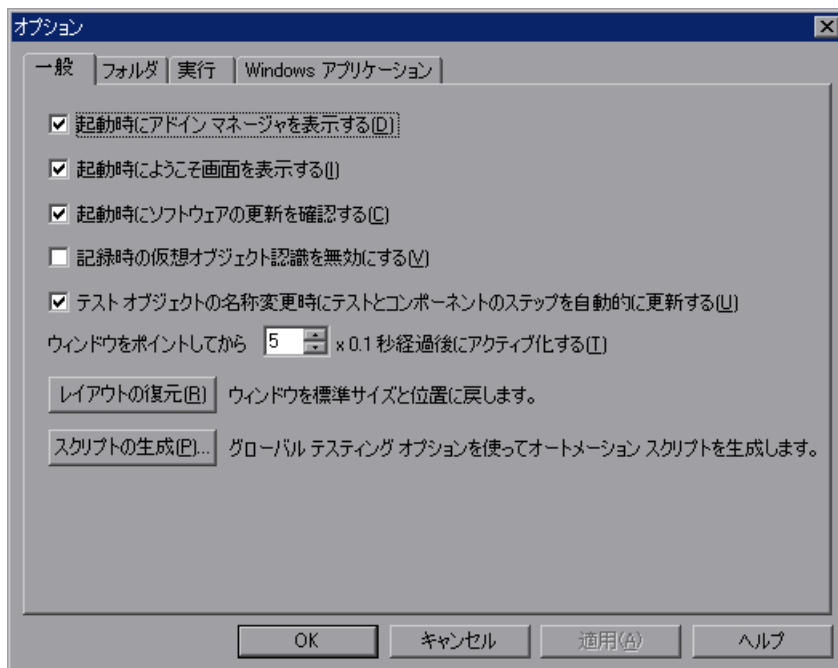
[オプション] ダイアログ・ボックスの使用

[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、以降のすべての記録と実行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。



- 1 [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、内容ごとにいくつかのタブに分かれています。



注：先に示した [Web] タブは、Web アドインがインストールされ、読み込まれている場合にのみ表示されます。

- 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。各タブで利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。

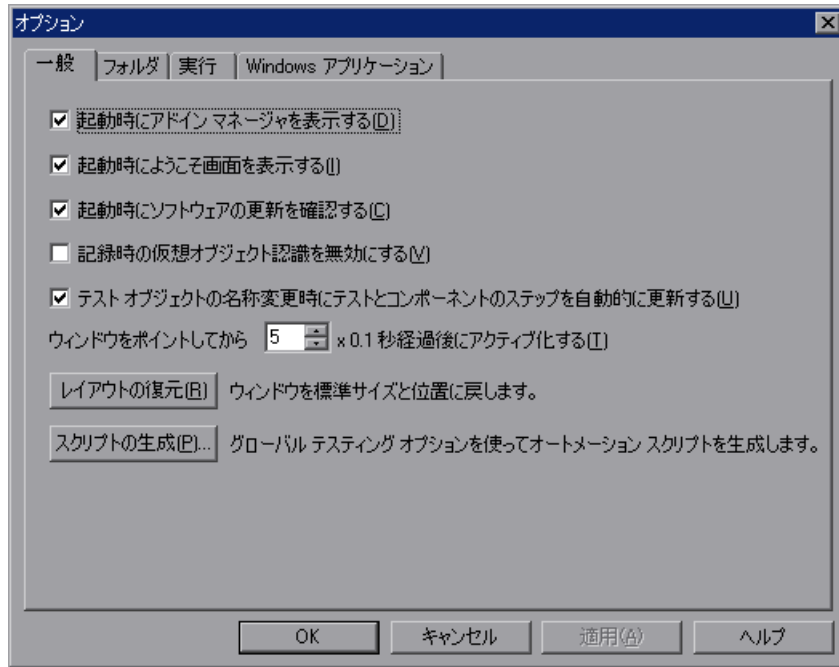
[オプション] ダイアログ・ボックスには、次のタブがあります。

タブ	内容
[一般]	コンポーネントの一般設定のオプション。詳細については、194 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、コンポーネント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。コンポーネントの場合、すべてのファイルを Quality Center のサブジェクト・パスに格納する必要があります。詳細については、196 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。
[実行]	コンポーネントを実行するためのオプション。詳細については、199 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]	次に示す Windows アプリケーションを対象とした QuickTest によるコンポーネン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。 <ul style="list-style-type: none"> ● 標準 Windows アプリケーション ● .NET Windows Forms ● Visual Basic ● ActiveX 詳細については、201 ページ「テストの Windows アプリケーション・オプションの設定」を参照してください。
[Web] (Web アドインがインストールされ、読み込まれている場合にのみ表示されます)	Web 環境での記録と実行セッションの動作を設定するためのオプション。詳細については、212 ページ「テストの Web オプションの設定」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、その時点でインストールされていて読み込まれている外部アドインに対応するタブが含まれる場合もあります。

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。



[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・マネージャでの作業の詳細については、683 ページ「QuickTest アドインのロード」を参照してください。
[起動時にようこそ画面を表示する]	QuickTest を起動したときに、[ようこそ] 画面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、『QuickTest Professional ユーザーズ・ガイド』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。このオプションはテストにのみ適用されます。
[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、327 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。
[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、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。

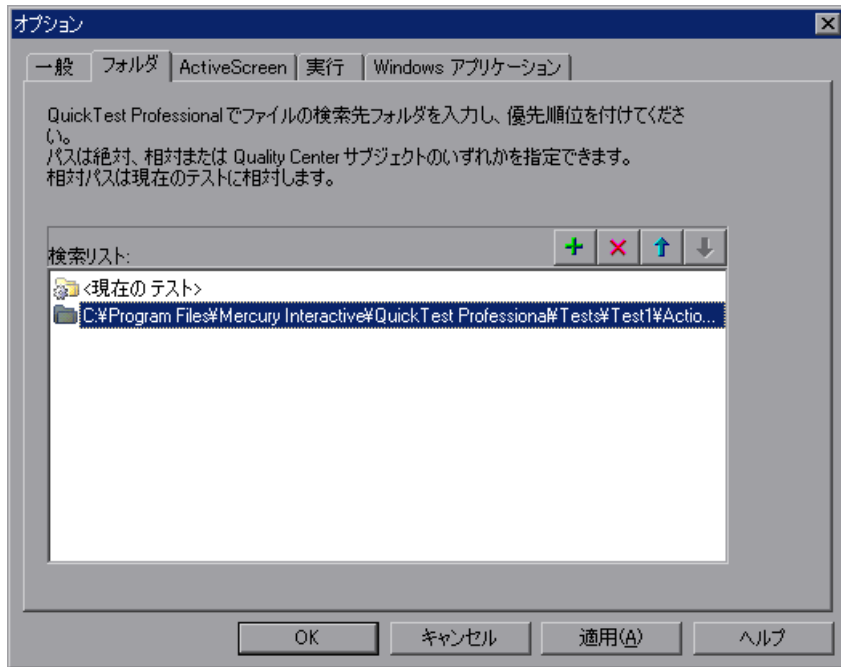
オプション	詳細
[レイアウトの復元]	表示枠およびツールバーが標準のサイズで標準の位置に表示されるように、QuickTest ウィンドウの配置を復元します。
[スクリプトの生成]	現在のグローバル・テスト・オプションを含んだ自動化スクリプトを生成します。詳細については、665 ページ「QuickTest 操作のオートメーション」を参照するか、『QuickTest オートメーション・オブジェクト・モデル・リファレンス』([ヘルプ] > [QuickTest オートメーションオブジェクトモデルリファレンス]) を参照してください。

テストのフォルダ・オプションの設定





[フォルダ] タブには、コンポーネントファイルを QuickTest が検索するフォルダ（検索パス）を入力できます。すべてのファイルを Quality Center のサブジェクト・パスに格納する必要があります。

注：標準設定では現在のコンポーネントが検索リストに表示されます。これは削除できません。

検索リストに表示されるフォルダの順序によって、指定されたコンポーネントまたはファイルを QuickTest が検索する順序が決まります。

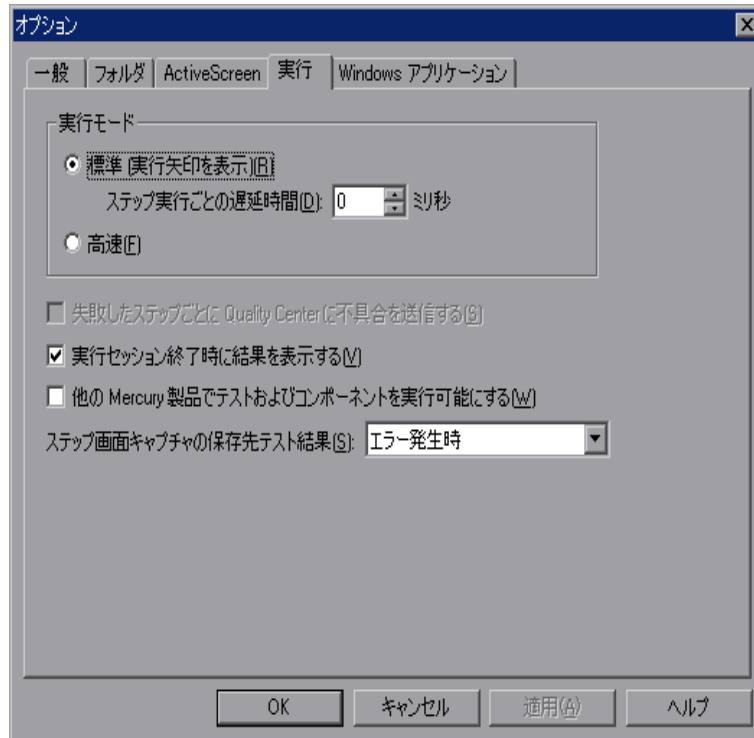


[フォルダ] タブには、次のオプションがあります。

オプション	詳細
[検索リスト]	<p>QuickTest がコンポーネント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、他のダイアログ・ボックスの中でコンポーネント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 が特定のファイルを検索する順序が決まります。</p>
	<p>新しいフォルダを検索リストに追加します。</p> <p>ヒント：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] が追加されて参照ボタンが表示され、Quality Center のパスを検索できるようになります。</p> <p>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\Tests のようにします。</p> <p>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。</p>
	<p>選択したフォルダを検索リストから削除します。</p>
	<p>選択したフォルダを上方向に移動します。</p>
	<p>選択したフォルダを下方向に移動します。</p>

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 がコンポーネントを実行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。

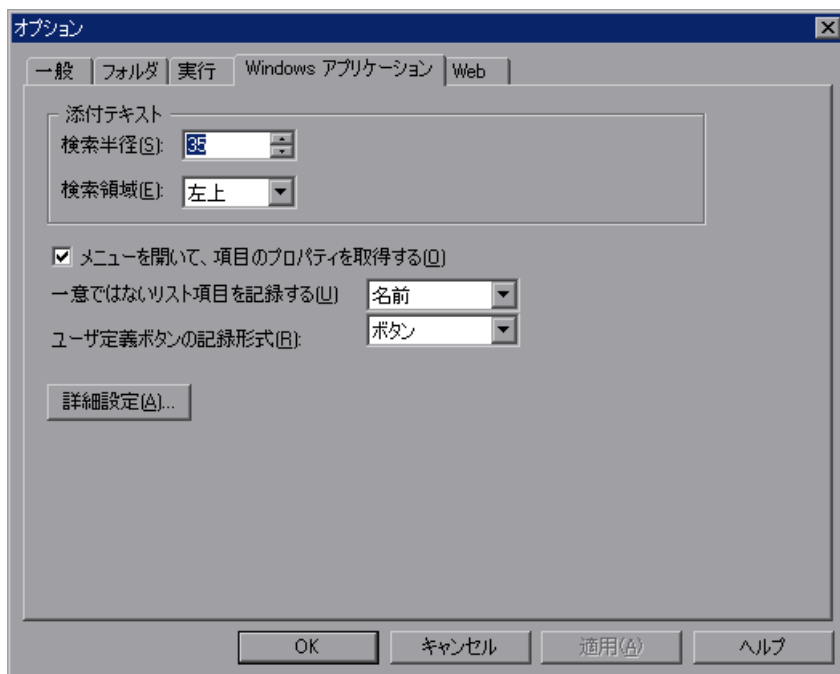


[実行] タブには、次のオプションがあります。

オプション	詳細
[実行モード]	<p>QuickTest に対してコンポーネントの実行方法を指定します。</p> <p>[標準 (実行矢印を表示)]: キーワード・ビューの左に実行矢印を表示しながらコンポーネントを実行し、実行した各ステップがわかるようにします。</p> <ul style="list-style-type: none"> • [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 ミリ秒まで)。 <p>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。</p> <p>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、『QuickTest Professional インストール・ガイド』を参照してください。</p> <p>[高速]: キーワード・ビューの左に実行矢印を表示せずにコンポーネントを実行します。このオプションでは、必要とされるシステム・リソースが少なく済みます。</p> <p>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。</p>
[失敗したステップごとに Quality Center に不具合を送信する]	<p>テストにのみ該当します。</p>
[実行セッション終了時に結果を表示する]	<p>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。</p>
[他の Mercury 製品でテストおよびコンポーネントを実行可能にする]	<p>Quality Center など他の Mercury 製品で QuickTest のコンポーネントを実行できるようにします。</p> <p>注: このオプションは WinRunner で QuickTest のコンポーネントを実行する場合には必要ありません。</p>
[ステップ画面キャプチャのテスト結果保存先]	<ul style="list-style-type: none"> • テストにのみ該当します。

テストの Windows アプリケーション・オプションの設定

[Windows アプリケーション] タブのオプションでは、標準の Windows アプリケーション、ActiveX、.NET Windows Forms アプリケーション、および Visual Basic アプリケーションを対象とした、QuickTest によるコンポーネントの記録および実行の方法を設定できます。



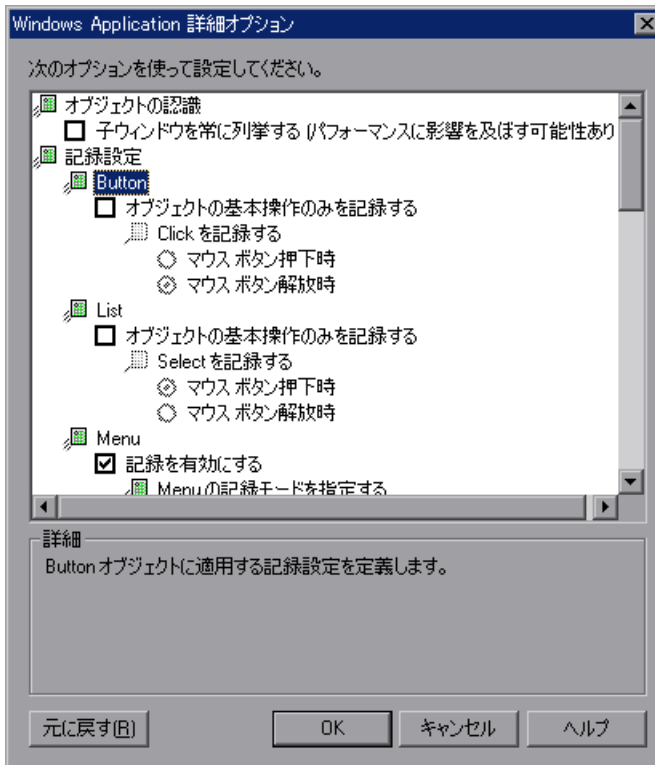
[Windows アプリケーション] タブには、次のオプションがあります。

オプション	詳細
[添付テキスト]	<p>オブジェクトの添付テキストを取得するのに QuickTest によって使われる検索基準を指定できます。オブジェクトの添付テキストは、指定したポイントから指定の範囲内で最も近い位置にある静的テキストです。取得した添付テキストは、対象オブジェクトに対応するテキスト・テスト・オブジェクト・プロパティまたは添付テキスト・テスト・オブジェクト・プロパティに保存されます。</p> <p>注：オブジェクトに最も近いように思える静的テキストが実際には最も近いわけではないこともあります。添付テキストが対象の静的テキストオブジェクトであることを確認するには試行錯誤が必要かもしれません。</p> <p>[検索半径]：QuickTest が添付テキストを検索する最大距離のピクセル数。</p> <p>[検索領域]：QuickTest がオブジェクトの添付テキストの検索を開始するオブジェクト上のポイント。</p> <p>リストから次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> ● [左上]：左上の角 ● [上]：両上角の中間点 ● [右上]：右上角 ● [右]：両右角の中間点 ● [右下]：右下角 ● [下]：両下角の中間点 ● [左下]：左下角 ● [左]：両左角の中間点
[メニューを開いて、項目のプロパティを取得する]	<p>実行セッション中にメニュー項目のプロパティを取得する前にメニュー・オブジェクトを開くよう QuickTest を設定します。</p> <p>注：このオプションを選択すると実行が遅くなる場合がありますが、メニューを開くときにメニュー項目のプロパティが変わる場合には便利です。標準で選択されているこのオプションでは、すべてのメニュー・オブジェクトの標準設定の動作が設定されます。関数ライブラリ内の ExpandMenu プロパティを使用して、この動作を個別のメニュー・オブジェクトに設定できます。詳細については、『QuickTest Professional オブジェクト・モデル・リファレンス』を参照してください。</p>

オプション	詳細
[一意ではないリスト項目を記録する]	<p>リストまたはツリー内に同一名の項目が複数ある場合に QuickTest が記録する対象を決定します。</p> <ul style="list-style-type: none"> • [名前] : 項目名が記録されます。 コンポーネントが実行されると、QuickTest は、記録中に選択された項目に関係なく、名前の最初の出現を探し、選択します。同じ名前の項目のプロパティがすべて同じ場合はこのオプションを選択します。 • [インデックス] : 項目のインデックス番号が記録されます。 同じ名前の項目のプロパティが必ずしもすべて同じでない場合はこのオプションを選択します。
[ユーザ定義ボタンの記録形式]	<p>アプリケーション内のユーザ定義のボタンを特定、記録する方法を QuickTest に設定します。 リストから次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none"> • [ボタン] • [チェックボックス] • [オプションボタン] • [オブジェクト] <p>注 : [オブジェクト] を選択すると、ユーザ定義ボタンが WinObjects として記録されます。</p>
[詳細設定]	<p>[Windows Application 詳細オプション] ダイアログ・ボックスが表示されます。このダイアログ・ボックスでは、Windows アプリケーションに対する記録および実行のオプションをカスタマイズできます。詳細については、204 ページ「Windows Application 詳細オプション」を参照してください。</p>

Windows Application 詳細オプション

[Windows Application 詳細オプション] ダイアログ・ボックスでは、ActiveX または Visual Basic などの Windows ベース・アプリケーションを対象とした、QuickTest によるコンポーネントの記録および実行の方法を変更できます。いつでも **[元に戻す]** ボタンをクリックして、すべてのオプションを標準の設定に戻すことができます。



オブジェクトの認識オプション

コンポーネントの実行時に QuickTest がオブジェクトを識別するために使用する方法を指定できます。

[Windows Application 詳細オプション] ダイアログ・ボックスでは、次の **[オブジェクトの認識]** オプションを設定できます。

オプション	詳細
[子ウィンドウを常に列挙する (パフォーマンスに影響を及ぼす可能性あり)]	コンポーネントの実行時にすべての子ウィンドウを列挙するように QuickTest を設定します。このオプションはパフォーマンスに大きく影響を与えることがあるため、標準ではクリアされており、ほかの手段でオブジェクトを識別できない場合に限り使用します。詳細については、210 ページ「詳細情報」を参照してください。

記録設定オプション

コンポーネントの記録時に QuickTest が特定のオブジェクトを処理する方法を指定できます。

[Windows Application 詳細オプション] ダイアログ・ボックスでは、次の **記録設定** オプションを設定できます。

カテゴリ	オプション
[Button]	<p>ボタン・オブジェクトの記録設定を定義します。</p> <ul style="list-style-type: none"> • [オブジェクトの基本操作のみを記録する] : ボタンの単純な記録を可能にします。このモードでは、標準的でない方法でのユーザ操作の認識が改善されることがあります。このオプションは標準ではクリアされており、標準の記録方式で必要なことができない場合にのみ使用します。詳細については、210 ページ「詳細情報」を参照してください。 • [Click を記録する] : マウス・ボタンが押されたとき ([マウス ボタン押下時]) または解放されたとき ([マウス ボタン解放時]) のどちらの場合に Click 操作を記録するべきかを指定します。このオプションは、[オブジェクトの基本操作のみを記録する] が選択されている場合にのみ有効になります。標準設定 = [マウス ボタン解放時]

カテゴリ	オプション
[List]	<p>Windows ベースのリスト・オブジェクト（WinList, WinListView, VbList など）の記録設定を定義します。</p> <ul style="list-style-type: none"> • [オブジェクトの基本操作のみを記録する]：リストの単純な記録を可能にします。このモードでは、標準的でない方法でのユーザ操作の認識が改善されることがあります。このオプションは標準ではクリアされており、標準の記録方式で必要なことができない場合にのみ使用します。詳細については、210 ページ「詳細情報」を参照してください。 • [Select を記録する]：マウス・ボタンが押されたとき（[マウス ボタン押下時]）または解放されたとき（[マウス ボタン解放時]）のどちらの場合に Select 操作を記録するべきかを指定します。このオプションは、[オブジェクトの基本操作のみを記録する] が選択されている場合にのみ有効になります。標準設定 = [マウス ボタン解放時]
[Menu]	<p>メニュー・オブジェクトの記録設定を定義します。</p> <ul style="list-style-type: none"> • [記録を有効にする]：メニュー・コントロールに対する操作を記録するかどうかを指定します。たとえば、別のウィンドウを開くためのメニュー選択操作を QuickTest に無視させたい場合が考えられます。このオプションは標準で選択されています。 • [Menu の記録モードを指定する]：メニュー・コントロールに対する操作を記録する前のメニュー初期化イベントを検証するか無視するかを指定します。このオプションは、[記録を有効にする] が選択されている場合にのみ有効になります。標準設定 = [メニュー初期化イベントを確認する] <p>詳細については、210 ページ「詳細情報」を参照してください。</p>

カテゴリ	オプション
[Object]	<p>WinObject テスト・オブジェクトとして認識されるオブジェクトの記録設定を定義します。</p> <ul style="list-style-type: none"> [オブジェクトの基本操作のみを記録する] : WinObject テスト・オブジェクトの単純な記録を可能にします。このモードでは、標準的でない方法でのユーザ操作の認識が改善されることがあります。このオプションは標準ではクリアされており、標準の記録方式で必要なことができない場合にのみ使用します。詳細については、210 ページ「詳細情報」を参照してください。 [Click を記録する] : マウス・ボタンが押されたとき（[マウス ボタン押下時]）または解放されたとき（[マウス ボタン解放時]）のどちらの場合に Click 操作を記録するべきかを指定します。このオプションは、[オブジェクトの基本操作のみを記録する] が選択されている場合にのみ有効になります。標準設定 = [マウス ボタン押下時]
[Tab]	<p>タブ・オブジェクトの記録設定を定義します。</p> <ul style="list-style-type: none"> [オブジェクトの基本操作のみを記録する] : タブの単純な記録を可能にします。このモードでは、標準的でない方法でのユーザ操作の認識が改善されることがあります。このオプションは標準ではクリアされており、標準の記録方式で必要なことができない場合にのみ使用します。詳細については、210 ページ「詳細情報」を参照してください。 [Select を記録する] : マウス・ボタンが押されたとき（[マウス ボタン押下時]）または解放されたとき（[マウス ボタン解放時]）のどちらの場合に Select 操作を記録するべきかを指定します。このオプションは、[オブジェクトの基本操作のみを記録する] が選択されている場合にのみ有効になります。標準設定 = [マウス ボタン解放時]

カテゴリ	オプション
<p>[Toolbar]</p>	<p>ツールバー・オブジェクトの記録設定を定義します。</p> <ul style="list-style-type: none"> • [オブジェクトの基本操作のみを記録する]：ツールバーの単純な記録を可能にします。このモードでは、標準的でない方法でのユーザ操作の認識が改善されることがあります。このオプションは標準ではクリアされており、標準の記録方式で必要なことができない場合にのみ使用します。詳細については、210 ページ「詳細情報」を参照してください。 • [Press を記録する]：マウス・ボタンが押されたとき（[マウス ボタン押下時]）または解放されたとき（[マウス ボタン解放時]）のどちらの場合に Press 操作を記録するべきかを指定します。このオプションは、[オブジェクトの基本操作のみを記録する] が選択されている場合にのみ有効になります。標準設定 = [マウス ボタン解放時]
<p>[Tree View]</p>	<p>ツリー・ビュー・オブジェクトの記録設定を定義します。</p> <ul style="list-style-type: none"> • [オブジェクトの基本操作のみを記録する]：ツリー・ビューの単純な記録を可能にします。このモードでは、標準的でない方法でのユーザ操作の認識が改善されることがあります。このオプションは標準ではクリアされており、標準の記録方式で必要なことができない場合にのみ使用します。詳細については、210 ページ「詳細情報」を参照してください。 • [Select を記録する]：マウス・ボタンが押されたとき（[マウス ボタン押下時]）または解放されたとき（[マウス ボタン解放時]）のどちらの場合に Select 操作を記録するべきかを指定します。このオプションは、[オブジェクトの基本操作のみを記録する] が選択されている場合にのみ有効になります。標準設定 = [マウス ボタン解放時] • [ツリー項目を記録する]：ツリー項目を [名前] または [仮想インデックス] のどちらで記録するかを指定します。標準設定 = [名前]

カテゴリ	オプション
[Window]	<p>ウィンドウ・オブジェクトの記録設定を定義します。</p> <ul style="list-style-type: none"> • [オブジェクトの基本操作のみを記録する]：ウィンドウの単純な記録を可能にします。このモードでは、標準的でない方法でのユーザ操作の認識が改善されることがあります。このオプションは標準ではクリアされており、標準の記録方式で必要なことができない場合にのみ使用します。詳細については、210 ページ「詳細情報」を参照してください。 • [Click を記録する]：マウス・ボタンが押されたとき（[マウス ボタン押下時]）または解放されたとき（[マウス ボタン解放時]）のどちらの場合に Click 操作を記録するべきかを指定します。このオプションは、[オブジェクトの基本操作のみを記録する] が選択されている場合にのみ有効になります。標準設定 = [マウス ボタン解放時]
[Keyboard]	<p>キーボードに対する操作の記録設定を定義します。</p> <ul style="list-style-type: none"> • [キーボード状態を検出する]：キーボードの状態を検出するために使用する API を指定します。標準設定 = [標準] <p>詳細については、210 ページ「詳細情報」を参照してください。</p>
[Utility オブジェクト]	<p>ユーティリティ・オブジェクトの記録設定を定義します。</p> <ul style="list-style-type: none"> • [SystemUtil.Run コマンドを記録する]：記録セッション中にアプリケーションを開いたときに SystemUtil.Run コマンドを記録するかどうかを指定します。このオプションは標準で選択されています。SystemUtil.Run メソッドの詳細については、『QuickTest Professional オブジェクト・モデル・リファレンス』を参照してください。

実行環境設定オプション

コンポーネントの実行時に QuickTest が特定のオブジェクトを処理する方法を指定できます。

[Windows Application 詳細オプション] ダイアログ・ボックスでは、次の **[実行環境設定]** オプションを設定できます。

オプション	詳細
[Edit Box]	エディット・オブジェクトの実行設定を定義します。 <ul style="list-style-type: none"> • [テキストの挿入前に、エディット ボックスをクリックする]：コンポーネントの実行時にエディット・ボックスにテキストを入力する際に、QuickTest が Click 操作を行うかどうかを指定します。このオプションは、標準設定ではクリアされています。 • [Set 操作にキーボード イベントを使用する]：選択すると、実行セッション中にエディット・ボックスを対象に Set 操作を実行するときにキーボード・イベントをシミュレートします。クリアすると、エディット・ボックスを対象とした Set 操作に API または Window メッセージを使用します。このオプションは、標準設定ではクリアされています。

詳細情報

次の情報は、Win32 API および Windows メッセージ・モデルに精通している方を対象としています。前節「Windows Application 詳細オプション」で紹介したいくつかの情報に基づいて説明を進めます。

[子ウィンドウを常に列挙する]

QuickTest がアプリケーションのオブジェクトを正しく記録しない場合、このオプションを選択して強制的にシステムのすべてのウィンドウを列挙するように QuickTest を設定できます。これは、QuickTest が WS_CHILD スタイルでないウィンドウを探す場合でも、最上位ウィンドウだけでなく、システムのすべてのウィンドウを列挙することを意味します。

アプリケーションに WS_CHILD スタイルではないけれども親（オーナーでない）ウィンドウを持っているウィンドウがある場合には、このオプションを選択します。

[オブジェクトの基本操作のみを記録する]

一般に、QuickTest はアプリケーションによって送信された Windows メッセージに基づいて Windows オブジェクトに対する操作を記録します。QuickTest はシステムからアプリケーションに送信された Windows メッセージのシーケンスを認識し、高度なアルゴリズムを使用してどの操作を記録するべきかを判断します。

まれに（非標準のメッセージ・シーケンスが使われた場合）、この高度なアルゴリズムによって不要な操作が記録されることがあります。対象イベントが発生したときにオブジェクトの基本の操作のみを記録したい場合には、このオプションを選択します。このオプションを選択した場合、操作を記録するタイミングも選択できます。[マウス ボタン押下時] を選択した場合、WM_LBUTTONDOWN メッセージが検出されると操作が記録されます。[マウス ボタン解放時] を選択した場合、WM_LBUTTONUP メッセージが検出されると操作が記録されます。

[キーボード状態を検出する]

QuickTest がキーボードの組み合わせ（CTRL+Y または ALT+CTRL+HOME など）を正しく記録しない場合、このオプションの標準設定を変更してみます。各オプションを簡単に説明します。

- ▶ [標準] : GetKeyboardState API を使用してキーボードの状態を検出します。詳細については、<http://msdn.microsoft.com/library/en-us/winui/winui/windowsuserinterface/userinput/keyboardinput/keyboardinputreference/keyboardinputfunctions/getkeyboardstate.asp> を参照してください。
- ▶ [代替同期] : GetKeyState API を使用してキーボードの状態を検出します。詳細については、<http://msdn.microsoft.com/library/en-us/winui/winui/windowsuserinterface/userinput/keyboardinput/keyboardinputreference/keyboardinputfunctions/getkeystate.asp> を参照してください。
- ▶ [代替非同期] : GetAsyncKeyState API を使用してキーボードの状態を検出します。詳細については、<http://msdn.microsoft.com/library/en-us/winui/winui/windowsuserinterface/userinput/keyboardinput/keyboardinputreference/keyboardinputfunctions/getasynckeystate.asp> を参照してください。

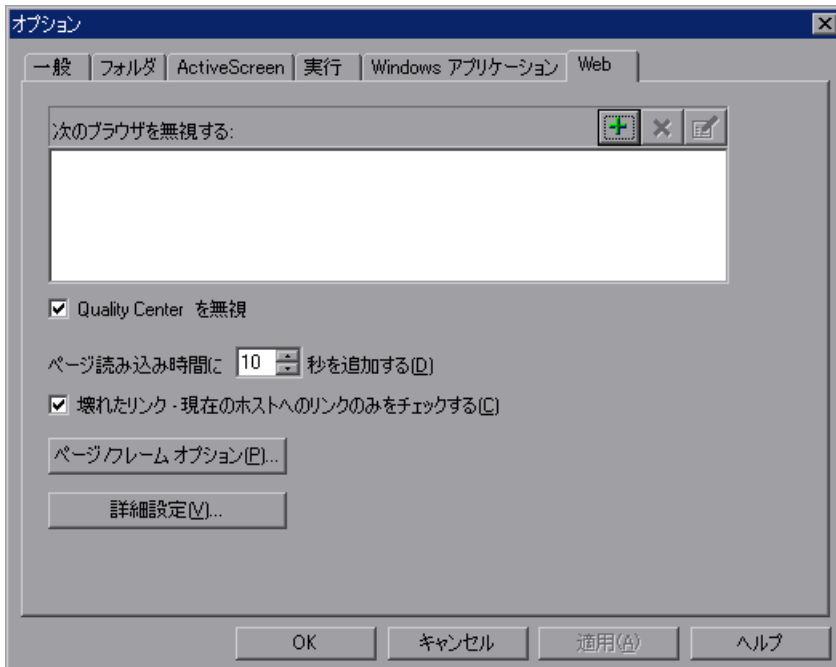
メニューの記録モード

大半のアプリケーションでは、ユーザがメニューを開くと Windows によって WM_CONTEXTMENU メッセージ、WM_ENTERMENULOOP メッセージ、WM_INITMENU メッセージ、WM_INITMENUPOPUP メッセージなどの初期化メッセージが送信されます。その後、ユーザがメニュー項目を選択すると、Windows によって WM_MENUSELECT メッセージが送信されます。

[メニュー初期化イベントを確認する] オプションを選択すると、メニュー初期化メッセージが検出された後にのみメニュー操作が記録されます。QuickTest がメニュー操作を正しく記録しない場合やアプリケーションが WM_MENUSELECT メッセージを送信する前に初期化メッセージを送信しない場合には、[メニュー初期化イベントを無視する] オプションを使用してみます。このオプションを選択した場合、メニュー操作はすべて記録されます。

テストの Web オプションの設定

[Web] タブでは、Web サイトを対象にコンポーネントを記録および実行するときの QuickTest の動作を設定します。



[Web] タブには、次のオプションがあります。

オプション	詳細
[次のブラウザを無視する]	QuickTest がコンポーネントの記録時または実行時に開いている可能性のある特定のブラウザを無視するように QuickTest に指定します。詳細については、214 ページ「無視するブラウザのリストの管理」を参照してください。
[Quality Center を無視]	コンポーネントの記録時または実行時に開いている、Quality Center のすべてのインスタンスを無視するように QuickTest に指定します。標準設定では、このオプションは選択されています。
[ページ読み込み時間に X 秒を追加する]	各ページ・チェックポイントに指定されているページの読み込み時間プロパティに指定秒数を加算するように QuickTest を設定します。 注：このオプションは、実行時のページの読み込み時間が記録セッションのときの時間よりも長くなることによって、ページのチェックポイントが失敗するのを回避するための保護手段です。
[壊れたリンクー現在のホストへのリンクのみをチェックする]	現在のホストをターゲットとするリンクが壊れていないかだけをチェックするように QuickTest を設定します。
[ページ/フレーム オプション]	[ページとフレーム オプション] ダイアログ・ボックスが表示されます。このダイアログ・ボックスでは、Page および Frame テスト・オブジェクトの記録方法をカスタマイズできます。詳細については、217 ページ「ページとフレーム・オプション」を参照してください。
[詳細設定]	[詳細 Web オプション] ダイアログ・ボックスが表示されます。このダイアログ・ボックスでは、Web サイトに対する記録および実行のオプションをカスタマイズできます。詳細については、221 ページ「詳細 Web オプション」を参照してください。

無視するブラウザのリストの管理

コンポーネントの記録時または実行時に開いている特定のブラウザを無視するように、QuickTest に指定することができます。これにより、記録や実行セッションに影響を与えずに、テスト環境に関係のないブラウザを開いておくことができます。たとえば、記録または実行セッション中に、会社の株価やニュースの見出しを確認する必要が生じる場合が考えられます。特定のブラウザを無視するように QuickTest に指定していれば、これらがセッションに影響を与えることはありません。

注：QuickTest では、記録または実行セッションの開始時に、定義された条件に一致するブラウザが無視されます。ただし、定義された条件に、記録または実行セッションの開始時に一致せず、セッション中に一致するようになっても、ブラウザは無視されません。

これらの設定に加えられた変更は、新しいコンポーネントおよび既存のコンポーネントの新しいステップにのみ適用され、その他の既存のステップには適用されません。

また、QuickTest が無視するブラウザを識別したり、無視するブラウザのリストからブラウザを削除したりするために使用するプロパティを変更できます。

ヒント：標準では、QuickTest の [オプション] ダイアログ・ボックスの [Web] タブにある **[Quality Center を無視]** チェック・ボックスが選択されている場合、記録または実行セッション中に開かれた Quality Center のすべてのインスタンスが無視されます。無視するブラウザのリストで Quality Center を指定する必要はありません。

リストへのブラウザの追加

記録または実行セッション中に QuickTest に無視させるブラウザを指定できます。

リストにブラウザを追加するには、次の手順を実行します。



- 1 ブラウザをリストに追加するには、[**ブラウザの追加**] ボタンをクリックします。[ブラウザ詳細] ダイアログ・ボックスが開きます。

ブラウザ詳細

名前(N):
Browser1

プロパティ

次のいずれかまたは両方のプロパティを使用してブラウザを識別します。
ヒント: 両方のプロパティに対して正規表現を使用できます。

タイトル(T):

URL(U):

OK キャンセル ヘルプ

- 2 [名前] フィールドにブラウザの名前を入力します。標準では、ブラウザの名前は「**Browser <リスト内でのブラウザの番号>**」です。指定する名前は、リスト内でのブラウザの識別にのみ使用され、QuickTest には使用されません。
- 3 無視するブラウザを識別するために次のプロパティのいずれかまたは両方を選択して、次の詳細を入力します。
 - ▶ [タイトル] : ブラウザのタイトル・バーに表示される Web ページの名前です。たとえば Yahoo! Finance* です。
 - ▶ [URL] : Web ページの URL です。たとえば <http://www.finance.yahoo.com> です。この Web ページのすべての子孫も、無視するブラウザのリストに自動的に追加されます。

ヒント：これらのプロパティの値の指定には正規表現を使用できます。たとえば、「*.finance.yahoo.com」を使用して、www., http://, https:// で始まる finance.yahoo.com のすべてのドメインと Web サイトを指定できます。QuickTest は自動的にドメインまたはサイト全体を無視するので、サイトの子ページを含めるために正規表現を使用する必要はありません。


注：[タイトル] および [URL] プロパティは AND 関係にあります。つまり、QuickTest でブラウザが無視されるためには、両方のプロパティ値（定義されている場合）にブラウザが一致していなければならないことを意味します。

- 4 [OK] をクリックします。無視されるブラウザのリストにブラウザが追加されます。
- 5 リストに追加するブラウザごとに、手順 1～4 を繰り返します。

リストのブラウザの変更

QuickTest が記録または実行セッション中に無視するべきブラウザの定義を変更できます。

リスト内のブラウザを変更するには、次の手順を実行します。

- 1 変更対象のブラウザを選択して強調表示します。
- 2  [ブラウザの詳細を編集] ボタンをクリックします。[ブラウザ詳細] ダイアログ・ボックスが開きます。
- 3 [ブラウザ詳細] ダイアログ・ボックスで必要な変更を加えて、[OK] をクリックします。

リストからのブラウザの削除

リストからブラウザを削除することで、それまで記録または実行セッション中に QuickTest に無視させていたブラウザを無視しないようにできます。

ヒント：特定のテストの実行にリスト内のあるブラウザが必要な場合、ブラウザのリストでブラウザ名の横のチェック・マークをクリアして、ブラウザをリストから一時的に削除できます。

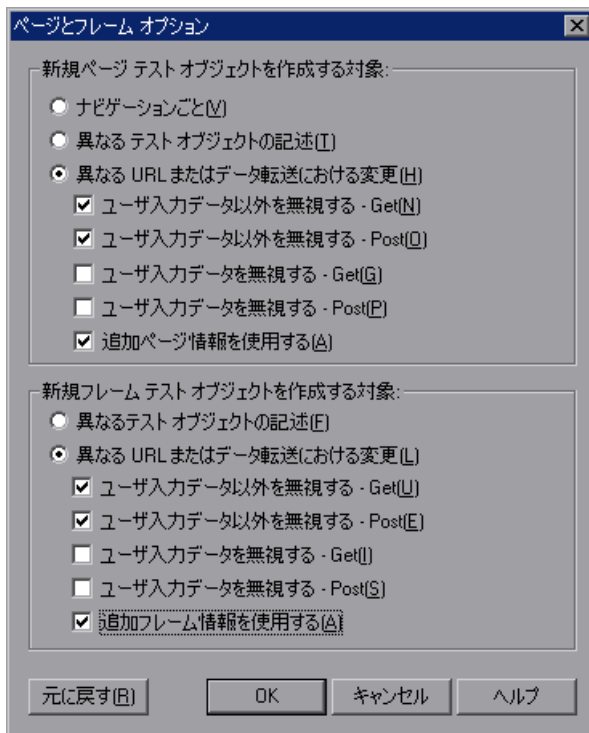
リストからブラウザを削除するには、次の手順を実行します。

- 1 リストから削除する対象のブラウザを選択して強調表示します。
- 2 **[ブラウザの削除]** ボタンをクリックします。



ページとフレーム・オプション

[ページとフレーム オプション] では、Page および Frame オブジェクトの記録方法をカスタマイズできます。



注：いつでも **[元に戻す]** ボタンをクリックして、すべてのオプションを標準のコア設定に戻すことができます。いくつかの外部アドインでは、ページ記録およびフレーム記録を最適化するために、標準設定を変更します。外部アドインを使用している場合は、標準のアドイン設定を維持し、**[元に戻す]** ボタンを使用しないことをお勧めします。

Page のオプション

[新規ページ テスト オブジェクトを作成する対象] のオプション群は、新規 Page オブジェクトをオブジェクト・リポジトリに作成するタイミングを QuickTest に設定します。

注：これらのオプションは、Page テスト・オブジェクトの作成方法にのみ影響します。Frame テスト・オブジェクトは Frame オプションの選択内容に応じて作成されます。詳細については、220 ページ「Frame のオプション」を参照してください。

次の Page オプションを指定できます。

- ▶ **[ナビゲーションごと]**：Web ページでナビゲーションが実行されるたびに新規 Page オブジェクトが作成されます。
- ▶ **[異なるテスト オブジェクトの記述]**：ページのオブジェクト記述が異なる場合に、Page テスト・オブジェクトのために定義されているプロパティに応じて新規 Page オブジェクトが作成されます。

注：Page オブジェクトの標準のオブジェクト記述には、テスト・オブジェクトのクラスのみが含まれます。このオプションを選択する場合には、個々の Page オブジェクトを一意に識別できるオブジェクト識別プロパティを定義することを強くお勧めします。また、時間が経っても変化することのないプロパティを定義して、将来の実行が失敗しないようにします。

- ▶ **[異なる URL またはデータ転送における変更]** : ページの URL が変わったときにのみ、あるいは URL は変わらず、サーバに転送されるデータが変わったときに、次に示すデータ・タイプと転送方法に基づいて、新規 Page オブジェクトを作成します。

注 : ナビゲーションごとに新規 Page オブジェクトを作成するよう QuickTest を設定するには、このオプションをクリアします (QuickTest version 5.6 以前は自動的にこの方法で動作しました)。

- ▶ **[ユーザ入力データ以外を無視する - Get]** : Get メソッドがサーバへのデータ転送に使われている場合にはユーザ以外からのデータ入力を無視するよう、QuickTest を設定します。

たとえば、ユーザがある Web ページにデータを入力し、そのデータが Get メソッドによって隠しフィールドとして挿入されるとします。ユーザはデータをサーバに送るために **[送信]** をクリックします。隠しフィールドのデータに応じて、返される新しい Web ページは異なります。しかし、QuickTest によって新規の Page テスト・オブジェクトは作成されません。

- ▶ **[ユーザ入力データ以外を無視する - Post]** : Post メソッドがサーバへのデータ転送に使われている場合にはユーザ以外からのデータ入力を無視するよう、QuickTest を設定します。

たとえば、ユーザがある Web ページにデータを入力し、そのデータが Post メソッドによって隠しフィールドとして挿入されるとします。ユーザはデータをサーバに送るために **[送信]** をクリックします。隠しフィールドのデータに応じて、返される新しい Web ページは異なります。しかし、QuickTest によって新規の Page テスト・オブジェクトは作成されません。

- ▶ **[ユーザ入力データを無視する - Get]** : Get メソッドがサーバへのデータ転送に使われている場合にはユーザ入力のデータを無視するよう、QuickTest を設定します。

たとえば、ある Web ページのフォームにデータを入力し、**[送信]** をクリックして、サーバにデータを伝送するために Get メソッドを使用したとします。ユーザが入力したデータに応じて、返される新しい Web ページは異なります。しかし、QuickTest によって新規の Page テスト・オブジェクトは作成されません。

- ▶ **[ユーザ入力データを無視する - Post]** : **Post** メソッドがサーバへのデータ転送に使われている場合にはユーザ入力のデータを無視するよう、QuickTest を設定します。

たとえば、ある Web ページのフォームにデータを入力し、**[送信]** をクリックして、サーバにデータを伝送するために **Post** メソッドを使用したとします。ユーザが入力したデータに応じて、返される新しい Web ページは異なります。しかし、QuickTest によって新規の Page テスト・オブジェクトは作成されません。

- ▶ **[追加ページ情報を使用する]** : テスト・オブジェクトの追加プロパティを使って、既存の Page テスト・オブジェクトを識別するよう、QuickTest を設定します。

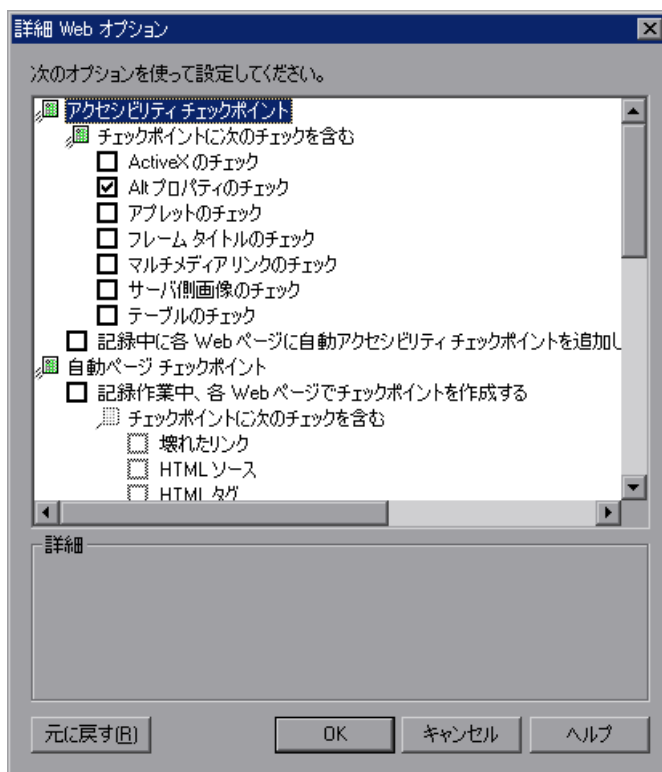
ヒント : このオプションを選択して、**[戻る]** と **[進む]** ナビゲーション・ボタンが使われたときに既存のページを認識するよう、QuickTest を設定します。

Frame のオプション

[新規フレーム テスト オブジェクトを作成する対象] のオプション群は、新規 Frame オブジェクトをオブジェクト・リポジトリに作成するタイミングを QuickTest に設定します。Frame のオプションは Page のオプションとほぼ同じです (ただし、**[ナビゲーションごと]** オプションがありません)。詳細については、218 ページ「Page のオプション」を参照してください。

詳細 Web オプション

[詳細 Web オプション] ダイアログ・ボックスでは、Web サイトを対象とした、QuickTest によるコンポーネントの記録および実行方法を変更できます。いつでも **[元に戻す]** ボタンをクリックして、すべてのオプションを標準の設定に戻すことができます。



注：[アクセシビリティ チェックポイント] および [自動ページ チェックポイント] オプションは、ビジネス・コンポーネントには適用されません。

- ▶ **[チェックポイントに次のチェックを含む]：**すべてのアクセシビリティ・チェックポイントで、選択したアクセシビリティ要素を検査するよう QuickTest を設定します。次から選択します。

- ▶ **[ActiveX のチェック]**：ページまたはフレームに Active X オブジェクトが含まれているかどうかを検査します。含まれている場合、QuickTest から警告が送られ、該当するオブジェクトのリストが [テスト結果] に表示されます。
- ▶ **[Alt プロパティのチェック]**：画像など、関連するすべてのオブジェクトに < alt > 属性が存在するかどうかを検査します。必要な属性が存在しないオブジェクトがあると、テストが失敗し、QuickTest によって属性が不足しているオブジェクトのリストが [テスト結果] に表示されます（標準で選択されています）。
- ▶ **[アプレットのチェック]**：ページまたはフレームに Java オブジェクトが含まれているかどうかをチェックします。含まれている場合、QuickTest から警告が送られ、該当するオブジェクトのリストが [テスト結果] に表示されます。
- ▶ **[フレーム タイトルのチェック]**：ページおよびページ内のすべてのフレームにタイトルがあるかどうかを検査します。必要なタイトルのないフレームまたはページがあると、テストが失敗し、QuickTest によってタイトルが不足しているフレームのリストが [テスト結果] に表示されます。
- ▶ **[マルチメディア リンクのチェック]**：ページまたはフレームにマルチメディア・オブジェクトへのリンクが含まれているかどうかを検査します。含まれている場合、QuickTest から警告が送られ、該当するリンクのリストが [テスト結果] に表示されます。
- ▶ **[サーバ側画像のチェック]**：ページまたはフレームにサーバ側画像が含まれているかどうかを検査します。含まれている場合、QuickTest から警告が送られ、該当する画像のリストが [テスト結果] に表示されます。
- ▶ **[テーブルのチェック]**：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を検査します。含まれている場合、QuickTest から警告が送られ、テーブル形式と各セルで使用されているタグが [テスト結果] に表示されます。
- ▶ **[記録中に各 Web ページに自動アクセシビリティ チェックポイントを追加します。このオプションはテストにのみ適用されます。]**：記録中、上記のオプションで選択したチェック項目を使って各 Web ページにアクセシビリティ・チェックポイントを自動的に追加するように QuickTest を設定します。

- ▶ **【記録作業中、各 Web ページでチェックポイントを作成する】**：記録処理中にナビゲートする Web ページごとに、自動的に Page チェックポイントを追加するように QuickTest を設定します。

注：動的内容が含まれる Web ページをテストする場合、自動 Page チェックポイントを使用すると、これらのチェックポイントでは、記録セッションと実行セッションの間でページの内容が変わらないと想定されているため、テストが失敗することがあります。

すべての自動 Page チェックポイントに、次のオプションから選択するチェックが含まれます。

- ▶ **【壊れたリンク】**：実行セッション中のページに含まれる、壊れたリンクの数を表示します。

注：[壊れたリンク - 現在のホストへのリンクのみをチェックする] オプションが選択されている場合（212 ページ「テストの Web オプションの設定」を参照）、この数には現在のホストをターゲットとする破損リンクのみ含まれます。

- ▶ **【HTML ソース】**：期待ソース・コードが実行セッション中のソース・コードに一致するかどうかを検査します。
- ▶ **【HTML タグ】**：ソース・コードの期待 HTML タグが実行セッションのタグに一致するかどうかを検査します。
- ▶ **【画像 ソース】**：画像の期待ソース・パスが実行セッションのソースに一致するかどうかを検査します。
- ▶ **【リンクの URL】**：リンクの期待 URL アドレスが実行セッション中のソース・コードの URL アドレスに一致するかどうかを検査します。
- ▶ **【ロード時間】**：実行セッション中のページの期待読み込み時間が、記録セッションで読み込みにかかった時間と [ページ読み込み時間に X 秒を追加する] オプションで指定された時間（212 ページ「テストの Web オプションの設定」を参照）の合計以下であるかどうかを検査します。

- ▶ **[画像の数]**：期待画像数が実行セッションで表示される数と一致するかどうかを検査します。
- ▶ **[リンクの数]**：期待リンク数が実行セッションで表示される数と一致するかどうかを検査します。
- ▶ **[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]**：自動的に追加された Page チェックポイントをテストの実行中に無視するように QuickTest を設定します。

記録の設定

Web オブジェクトを記録するための設定を行うことができます。[詳細 Web オプション] ダイアログ・ボックスには、次の記録の設定があります。

- ▶ **[Microsoft Windows Explorer の Web サポートを有効にする]**：選択すると、Microsoft Windows Explorer 内の関係するオブジェクトを Web オブジェクトとして処理するよう QuickTest が設定されます。クリアすると、Microsoft Windows Explorer に表示された Web ページのイベントに記録しないよう QuickTest が設定されます。

注：この設定の変更後、変更を反映するには開いている Microsoft Windows Explorer をすべて閉じ（Windows のタスク マネージャですべての **explorer.exe** プロセスが終了していることを確認します。または、コンピュータを再起動します）、QuickTest を再スタートします。

- ▶ **[座標を記録する]**：操作ごとに、実際の座標をオブジェクトからの相対座標として記録します。
- ▶ **[MouseDown および MouseUp を Click として記録する]**：MouseUp および MouseDown イベントに対して **Click** メソッドを記録します。
- ▶ **[すべてのナビゲーション操作で Navigate を記録する]**：フレームの URL が変わるたびに、**Navigate** ステートメントを記録します。

- ▶ **[標準 Windows マウス イベントの使用]** : 次のイベントについて、ブラウザ・イベントではなく Windows の標準マウス・イベントを使用するよう、QuickTest を設定します。
 - ▶ **OnClick**
 - ▶ **OnMouseDown**
 - ▶ **OnMouseUp**

注 : このオプションは、ブラウザ・イベントを使ってもイベントを正しく記録できない場合にのみ使用します。

QuickTest で要求に合うイベントが記録されない場合、Web オブジェクトの種類ごとに記録するイベントを設定することも可能です。たとえば、サブメニューを開く `mouseover` イベントのようなイベントを記録したい場合には、当該イベントを検出できるように Web イベント設定を変更する必要があるかもしれません。詳細については、第 11 章「Web イベント記録の設定」を参照してください。

実行の設定

実行セッション中に Web オブジェクトで作業するための設定を行うことができます。[詳細 Web オプション] ダイアログ・ボックスには、次の実行の設定があります。

- ▶ **[ブラウザのクリーンアップを行う]** : 現在のコンポーネントが完了したときに、表示されているすべてのブラウザを閉じます。

このオプションが選択されていると、ブラウザが QuickTest より先または後に開かれたかどうかにかかわらず、現在のコンポーネントが終了すると、開いているブラウザがすべて閉じられます。
- ▶ **[Click のみ実行する]** : MouseDown イベント、MouseUp イベント、および Click イベントを使用するか、Click イベントのみを使用して、Click イベントを実行します。
- ▶ **[再生の種類]** : 選択したオプションに従って、マウス操作の実行方法を設定します。
 - ▶ **[イベント]** — ブラウザ・イベントを使用してマウス操作を実行します。
 - ▶ **[マウス]** — マウスを使用してマウス操作を実行します。

- ▶ **【ソースインデックスを使って実行する】**：パフォーマンスを向上するため、ソース・インデックス・プロパティを使用します。
- ▶ **【記録セッション時にブラウザをサイズ変更されたら実行時にもサイズ変更する】**：このオプションが選択されているときに記録セッション中にブラウザのサイズを変更すると、以降の実行セッション開始時に QuickTest によってブラウザのサイズがこのサイズに変更されます。

注：このオプションを使用するには、記録を開始する前に「記録と実行環境設定」ダイアログ・ボックスの「**記録または実行セッションを開始する時、次のブラウザを開く**」オプションを選択します。

このオプションがクリアされていると、実行セッション開始時にブラウザのサイズが変更されません。

第 9 章

ビジネス・コンポーネント設定にかかわる作業

ビジネス・コンポーネントの作成またはデバッグに先立ち、コンポーネントに関連付けられているアプリケーション領域にすでに定義されているコンポーネントの設定を [ビジネス コンポーネントの設定] ダイアログ・ボックスに表示できます。また、[ビジネス コンポーネントの設定] ダイアログ・ボックスでは、コンポーネントに対していくつかの追加設定を定義できます。

本章では、次の項目について説明します。

- ▶ ビジネス・コンポーネント設定にかかわる作業について
- ▶ [ビジネス コンポーネントの設定] ダイアログ・ボックスへのアクセス
- ▶ コンポーネント・プロパティにかかわる作業
- ▶ コンポーネントのスナップショットの定義
- ▶ アプリケーション設定の表示
- ▶ コンポーネント・リソースの表示
- ▶ コンポーネントのパラメータの定義
- ▶ Web 設定の表示
- ▶ 回復シナリオの設定の表示

注：アプリケーション領域でのコンポーネント設定の定義の詳細については、第 3 章「アプリケーション領域の使用」を参照してください。

ビジネス・コンポーネント設定にかかわる作業について

新しいアプリケーション領域を作成するときは、新しいビジネス・コンポーネントの作成に必要な設定とリソースを定義します。設定には、関連アドイン、コンポーネントが記録と実行の対象にできる Windows ベースのアプリケーション、およびコンポーネントと一緒に使用する関数ライブラリや共有オブジェクト・リポジトリの場所が含まれます。

QTP エンジニア（または各分野のエキスパート）が新しいコンポーネントを作成すると、当該コンポーネントに関連付けられているアプリケーション領域に定義されている設定が自動的に関連付けられます。[ビジネス コンポーネントの設定] ダイアログ・ボックスには、これらの設定が読み取り専用形式で表示されます。

[ビジネス コンポーネントの設定] ダイアログ・ボックスでは、入力パラメータや出力パラメータ、コンポーネントのステータスなどのいくつかの追加設定を定義できます。

注：すべての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8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。

【ビジネス コンポーネントの設定】ダイアログ・ボックスへのアクセス

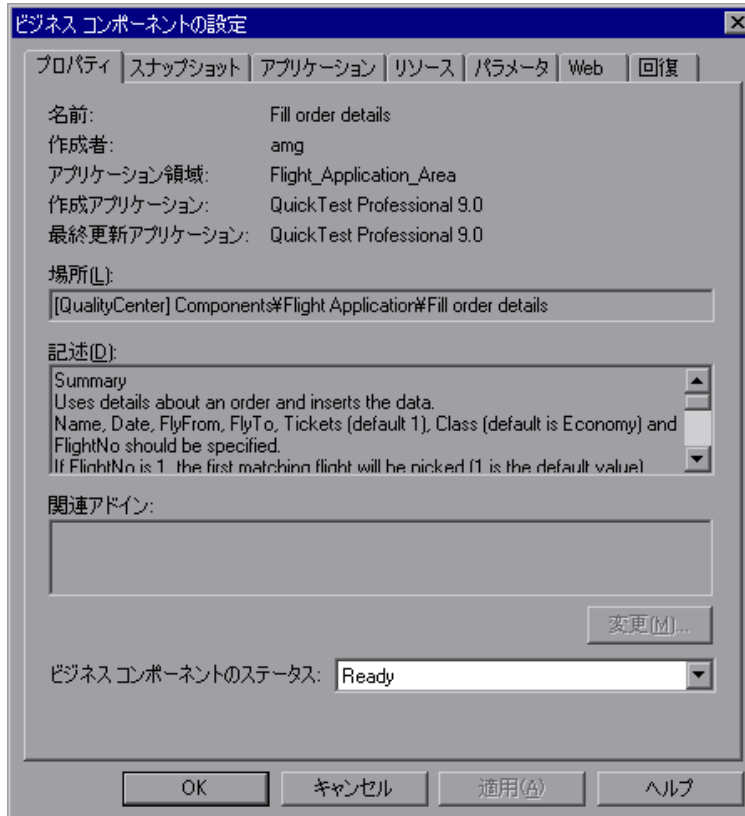
[ビジネス コンポーネントの設定] ダイアログ・ボックスでは、設定の表示やコンポーネントに対する個別のオプションの定義ができます。

[ビジネス コンポーネントの設定] ダイアログ・ボックスを開くには、次の手順を実行します。

- 1 設定を表示または定義する対象となるコンポーネントを開きます。



- 2 [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。[ビジネスコンポーネントの設定] 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。



- 3 必要に応じて, オプションの表示または設定に使用するタブを選択します。各タブで使用できる設定および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。
- 4 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。

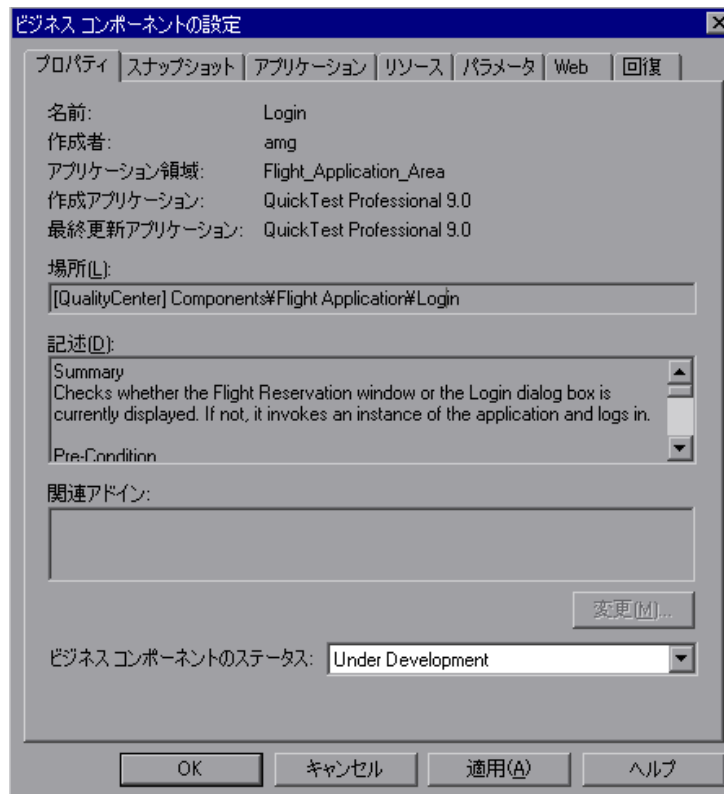
[ビジネス コンポーネントの設定] ダイアログ・ボックスには、次のタブがあります。

タブ	タブの内容
[プロパティ]	ビジネス・コンポーネントの説明や関連アドインなど、ビジネス・コンポーネントのプロパティ。コンポーネントのステータスも設定できます。詳細については、231ページ「コンポーネント・プロパティにかかわる作業」を参照してください。
[スナップショット]	Quality Center に表示するためにコンポーネントに保存するスナップショット画像をキャプチャまたはロードするためのオプション。詳細については、234ページ「コンポーネントのスナップショットの定義」を参照してください。
[アプリケーション]	コンポーネントが記録と実行の対象にできる Windows ベース・アプリケーション。詳細については、235ページ「アプリケーション設定の表示」を参照してください。
[リソース]	関数ライブラリや共有オブジェクト・リポジトリの場所を含め、コンポーネントに関連付けられているリソース。詳細については、238ページ「コンポーネント・リソースの表示」を参照してください。
[パラメータ]	コンポーネン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、239ページ「コンポーネントのパラメータの定義」を参照してください。
[Web]	コンポーネントのブラウザ・ナビゲーション・タイムアウトを指定するためのオプション。[Web] タブは、Web アドインがインストールされ、読み込まれている場合にのみ表示されます。詳細については、243ページ「Web 設定の表示」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コンポーネントが回復する方法。詳細については、245ページ「回復シナリオの設定の表示」を参照してください。

これらのタブのほかに、[ビジネス コンポーネントの設定] ダイアログ・ボックスには、スクリプト・コンポーネント用のタブが含まれる場合があります。これらのタブの詳細については、『QuickTest Professional 基本機能ユーザーズ・ガイド』を参照してください。また、ロードされている外部アドイン（SAP や Web サービス）に対応するタブが含まれる場合もあります。外部アドインに関連したタブの詳細については、対応する QuickTest アドインのドキュメントを参照してください。

コンポーネント・プロパティにかかわる作業

[ビジネス コンポーネントの設定] ダイアログ・ボックスの [プロパティ] タブを使用して、コンポーネントの説明やコンポーネントに関連付けられているアドインなど、コンポーネントに関する一般情報を表示できます。また、コンポーネントのステータスの設定や変更もできます。



[プロパティ] タブには、次の項目があります。

設定	詳細
[名前]	コンポーネントの名前を示します。コンポーネントの名前は、保存するときに指定します。
[作成者]	コンポーネントを作成した人物の Windows ユーザ名を示します。
[アプリケーション領域]	コンポーネントに関連付けられているアプリケーション領域の名前を示します。詳細については、123 ページ「新規ビジネス・コンポーネントの作成」を参照してください。 注： Quality Center でコンポーネントが作成され、アプリケーション領域が選択されなかった場合には、「未選択」と表示されます。ビジネス・コンポーネント・ステップを実装するには、アプリケーション領域を先に選択する必要があります。
[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。
[最終更新アプリケーション]	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]	コンポーネントの Quality Center パスおよびファイル名を示します。 注： コンポーネントがまだ保存されていない場合は、「保存されていません」と表示されます。
[記述]	コンポーネントに対して指定されている詳細説明が表示されます。このフィールドの入力または変更は、Quality Center のみ行うことができます。
[関連アドイン]	コンポーネントに関連付けられているアプリケーション領域を通じてコンポーネントに関連付けられているアドインが表示されます。関連アドインは、ビジネス・コンポーネントへのアクセスがあったときに当該コンポーネントによって読み込まれます。
[ビジネス コンポーネントのステータス]	コンポーネントのステータスを示します。リストから別のオプションを選択することでコンポーネントのステータスを変更できます。ステータス・オプションの詳細については、233 ページ「コンポーネントのステータスについて」を参照してください。

コンポーネントの基礎になっているアプリケーション領域の一般情報の定義方法の詳細については、83 ページ「一般設定の定義」を参照してください。

コンポーネントのステータスについて

ビジネス・コンポーネントには、QuickTest または Quality Center でステータスを割り当てることができます。ビジネス・コンポーネントのステータスは手作業で指定できますが、場合によっては Quality Center によって自動的に割り当てられます。たとえば、ビジネス・プロセス・テスト内で実行する準備ができたビジネス・コンポーネントに **Ready** ステータスを手作業で割り当てることもできれば、ビジネス・プロセス・テスト内での正常な実行を妨げるエラーを含んだビジネス・コンポーネントに **Error** ステータスが自動的に割り当てられることもあります。

ビジネス・コンポーネントのステータスは、当該コンポーネントを含んでいるビジネス・プロセス・テストのステータスに影響するので、ビジネス・コンポーネント全体のステータスを知ることは重要です。一般に、ビジネス・プロセス・テストのステータスは、最も重大なステータスを持つコンポーネントによって決まります。たとえば、**Error** ステータスを持つビジネス・コンポーネントを含むビジネス・プロセス・テストのステータスは **Error** になります。

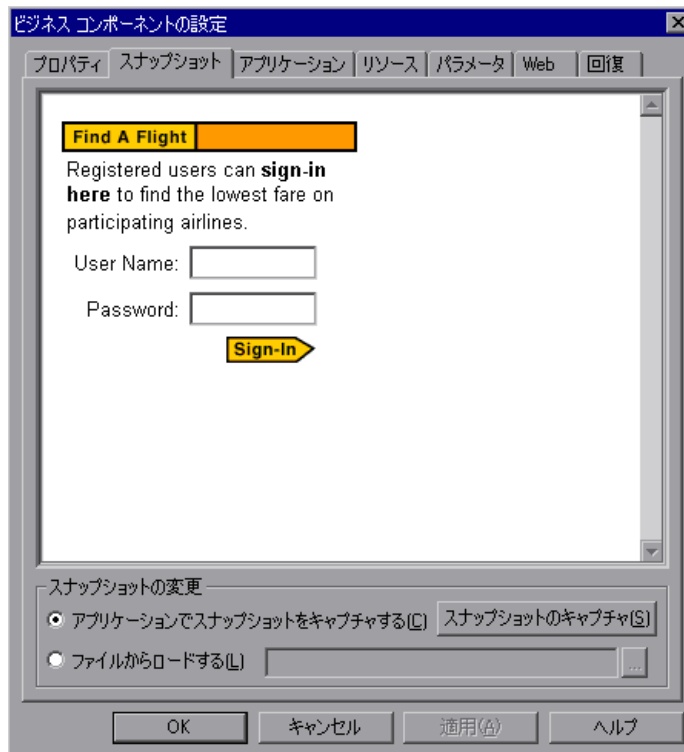
コンポーネントのステータスは次のいずれかです。

- ▶ **Error** : アプリケーションに変更が加えられたことなどが原因で、コンポーネントに修正の必要があるエラーが含まれています。このステータスのコンポーネントがビジネス・プロセス・テストに含まれていると、ビジネス・プロセス・テスト全体のステータスも **Error** となります。
- ▶ **Maintenance** : 対象コンポーネントは、現在開発およびテストが行われていて実行の準備が整っていないか、以前に実装され現在はアプリケーションに加えられた変更に合わせての変更が行われています。
- ▶ **Ready** : 対象コンポーネントは実装が完全に済んでおり、実行の準備が整っています。コンポーネントは指定された要件を満たしており、対象システムについて定義されている基準に基づいたテストが済んでいます。

- ▶ **Under Development** : 対象コンポーネントは現在開発中です。このステータスは自動的に次の項目に割り当てられます。
- ▶ **Business Process Testing** サポート付きの **Quality Center** のビジネス・コンポーネント・モジュールを使用して作成された新規コンポーネント。
- ▶ **Business Process Testing** サポート付きの **Quality Center** のコンポーネント・ツリーにドラッグして入れられたコンポーネント要求。

コンポーネントのスナップショットの定義

[ビジネス コンポーネントの設定] ダイアログ・ボックスの [スナップショット] タブは、画像をキャプチャまたはロードしてコンポーネントに保存することを可能にします。画像はコンポーネントの主要な目的を視覚的に示します。各分野のエキスパートは、**Quality Center**、コンポーネント、およびコンポーネントが含まれている任意のビジネス・プロセス・テストの画像を表示できます。



注：スナップショット画像は、Business Process Testing サポート機能がインストールされている場合は Quality Center の [スナップショット] タブでもキャプチャしてコンポーネントに保存できます。Quality Center でコンポーネントのスナップショットをキャプチャする方法については、『Business Process Testing ユーザーズ・ガイド』を参照してください。

[スナップショット] タブには、次のオプションがあります。

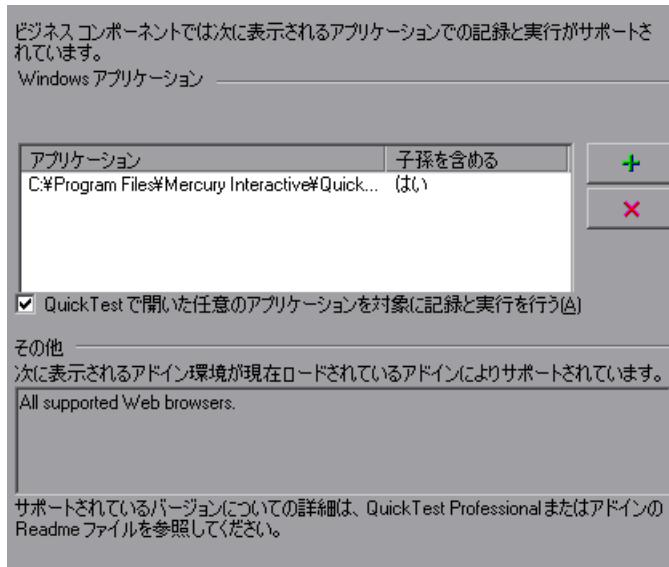
オプション	詳細
[アプリケーションでスナップショットをキャプチャする]	[スナップショットのキャプチャ] ボタンをクリックしてキャプチャする画像を定義することを可能にします。十字ポインタをドラッグしてキャプチャ対象の領域を選択できます。マウス・ボタンを放すと、キャプチャされた領域がスナップショット表示枠に表示されます。
[ファイルからロードする]	必要な画像が格納されている .png ファイルまたは .bmp ファイルを指定します。パスおよびファイル名を入力するか、参照ボタンを使用してファイルを指定します。

[適用] または [OK] をクリックすると、画像がコンポーネントに保存され、Quality Center においてこのコンポーネントを含んでいるビジネス・プロセス・テスト内に表示されます。

アプリケーション設定の表示

[ビジネス コンポーネントの設定] ダイアログ・ボックスの [アプリケーション] タブで、コンポーネント（または現在のアプリケーション領域に基づいたコンポーネント）が記録と実行の対象にできる Windows ベース・アプリケーションを表示できます。指定したアプリケーションが対象の場合にのみステップを記録できます。

(現在読み込まれているアドインに基づいて) コンポーネントが現在記録の対象にできる環境も表示できます。



関連付けられているアプリケーション領域設定で、コンポーネントが記録と実行の対象にできる Windows ベースのアプリケーションを指定します。詳細については、106 ページ「アプリケーション領域のアプリケーション設定の定義」を参照してください。

注：

[アプリケーション領域の設定] ダイアログ・ボックスの [アプリケーション] タブでアプリケーションの設定をせずに新しいコンポーネントの記録を開始すると、記録開始時に [アプリケーション] ダイアログ・ボックスが開きます。[アプリケーション] ダイアログ・ボックスには、[アプリケーション] タブと同じオプションがあります。詳細は、106 ページ「アプリケーション領域のアプリケーション設定の定義」を参照してください。

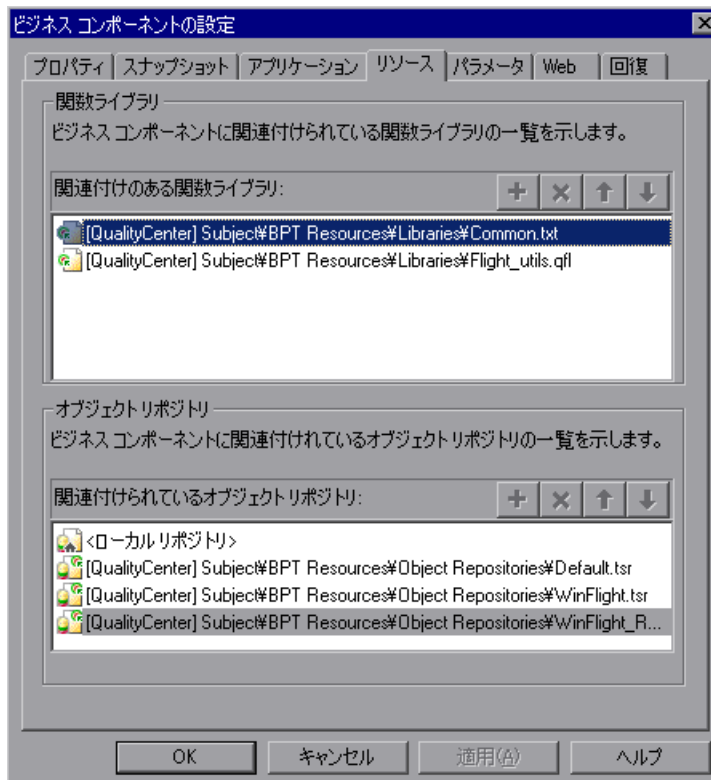
[アプリケーション] ダイアログ・ボックスおよび [アプリケーション] タブには、コンピュータにインストールされている任意の QuickTest 外部アドインに適用されるオプションが含まれている場合もあります。これらのオプションの詳細については、アドインに付属のマニュアルを参照してください。

[アプリケーション] タブには、次の項目があります。

設定	詳細
[アプリケーション]	コンポーネントの記録と実行を行う対象となるアプリケーションの詳細が列挙されます。 Windows アプリケーションを対象に記録と実行を行わない場合には、アプリケーション・リストを空のままにしておきます（これが標準の設定です）。
[QuickTest で開いた任意のアプリケーションを対象に記録と実行を行う]	QuickTest によって起動される任意のアプリケーションを（QuickTest の子プロセスとして）記録と実行の対象にします。たとえば、セッションの記録または実行中に OpenApp 関数を使用して開かれるアプリケーションや、アプリケーションを開く関数が含まれている別の操作によって開かれたアプリケーションなどが対象になります。
[その他]	現在読み込まれているアドインに対応するアドイン環境を一覧表示します。

コンポーネント・リソースの表示

[ビジネス コンポーネントの設定] ダイアログ・ボックスの [リソース] タブを使用して、コンポーネントに関連付けられているアプリケーション領域を通じてコンポーネントに関連付けられている関数ライブラリおよびオブジェクト・リポジトリを表示できます。指定したリソース・ファイルはすべて Quality Center プロジェクトに保存しなければなりません。



[リソース] タブには、次の項目があります。

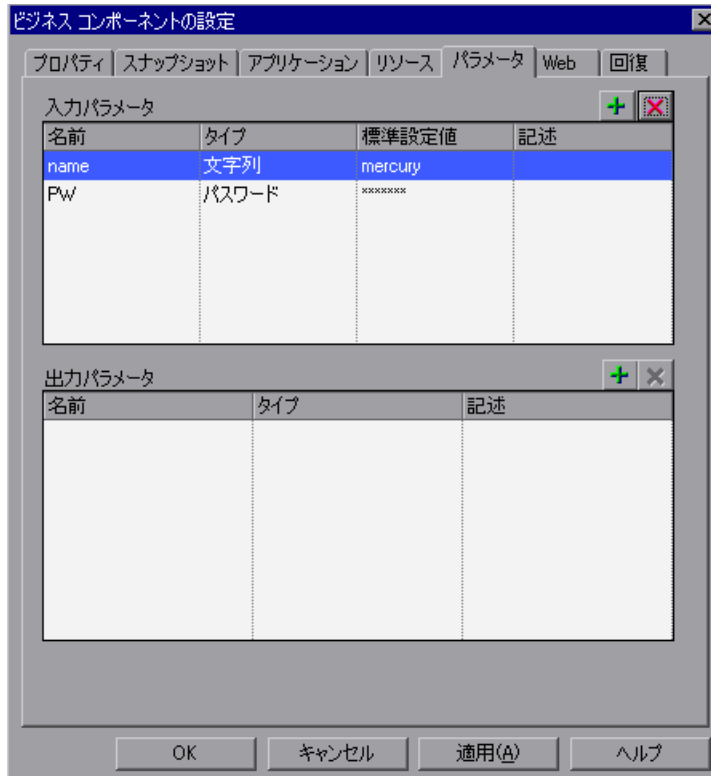
設定	詳細
[関連付けのある関数ライブラリ]	コンポーネントに関連付けられているアプリケーション領域を通じてコンポーネントに現在関連付けられている関数ライブラリのリストが表示されます。関数ライブラリの関連付けの詳細については、89 ページ「関数ライブラリの管理」および 594 ページ「関連付けられている関数ライブラリを使用した作業」を参照してください。
[オブジェクトリポジトリ]	コンポーネントに関連付けられているアプリケーション領域を通じてコンポーネントに現在関連付けられている共有オブジェクト・リポジトリのリストが表示されます。コンポーネントは、Quality Center に格納されている共有オブジェクト・リポジトリ・ファイルを使用します。オブジェクト・リポジトリのアプリケーション領域への関連付けの詳細については、94 ページ「共有オブジェクト・リポジトリの管理」を参照してください。

コンポーネントのパラメータの定義

[ビジネス コンポーネントの設定] ダイアログ・ボックスの [パラメータ] タブでは、コンポーネントに値を渡す入力コンポーネント・パラメータと、コンポーネントから外部のソースに値を渡す出力コンポーネント・パラメータを定義できます。また、[パラメータ] タブを使用して既存のコンポーネント・パラメータの変更や削除を行うこともできます。

コンポーネント・パラメータは、コンポーネント・ステップ内の入力値と出力値をパラメータ化するために使用できるパラメータです。コンポーネント・ステップ内でパラメータ値を使用する方法については、172 ページ「パラメータの使用」を参照してください。

ステップ内でのコンポーネント・パラメータの使用の詳細については、243ページ「ステップにおけるコンポーネント・パラメータの使用」を参照してください。各分野のエキスパートは、Quality Centerでコンポーネント・パラメータを定義することもできます。詳細については、『**Business Process Testing ユーザーズ・ガイド**』を参照してください。





[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]：コンポーネントが、それを実行または呼び出したソースから受け取れるパラメータを指定します。
- ▶ [出力パラメータ]：コンポーネントが、それを実行または呼び出したソースに渡せる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て、詳細を編集します（ただし、名前は変更できません）。

注：入力および出力のパラメータのリストは、Quality Center のビジネス・コンポーネント・モジュールでも変更できます。詳細については、『**Business Process Testing ユーザーズ・ガイド**』を参照してください。

次のボタンを使用して、ビジネス・コンポーネントの入力パラメータおよび出力パラメータの追加および削除ができます。

オプション	詳細
	<p>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を入力し、パラメータのタイプを選択してください。選択できるタイプは、文字列、ブール値、日付、数値、またはパスワードです。コンポーネン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。</p> <p>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」カラムで入力または変更できます。詳細については、次の「入力コンポーネント・パラメータの標準設定値の定義」を参照してください。</p>
	<p>選択したパラメータをビジネス・コンポーネントから削除します。</p>

入力コンポーネント・パラメータの標準設定値の定義

ビジネス・コンポーネン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、コンポーネントを呼び出したアプリケーション（QuickTest または Quality Center）から送信された値です。

ビジネス・コンポーネントの呼び出し元	パラメータ値の指定場所
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、492 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。
Quality Center	[コンポーネントの反復] ダイアログ・ボックス（テスト計画モジュール）。詳細については、『 Business Process Testing ユーザーズ・ガイド 』を参照してください。

コンポーネントの実行時に、1 つ以上の入力パラメータに対して QuickTest または Quality Center から値が渡されなかった場合、QuickTest はパラメータの標準設定値を使用します。

[ビジネス コンポーネントの設定] 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することも、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ることもできます。

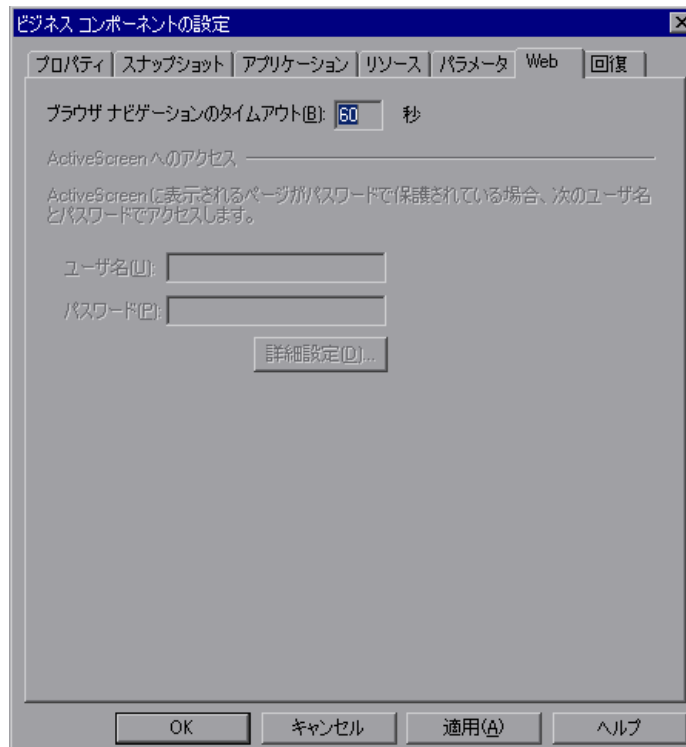
値のタイプ	QuickTest の標準設定値
文字列	空文字列
ブール値	True
日付	現在の日付
数値	0
パスワード	空文字列

ステップにおけるコンポーネント・パラメータの使用

コンポーネント・パラメータを定義したら、[値設定オプション] ダイアログ・ボックスで入力コンポーネント・パラメータを選択するか、[出力オプション] ダイアログ・ボックスで出力コンポーネント・パラメータを選択して、コンポーネントのステップの値をパラメータ化できます。ステップ内でローカル・パラメータを使用することもできます。ステップ内でコンポーネント・パラメータとローカル・パラメータを使用する方法については、172 ページ「パラメータの使用」を参照してください。

Web 設定の表示

[ビジネス コンポーネント設定] ダイアログ・ボックスの [Web] タブには、アプリケーションでコンポーネントを実行する場合に使用される設定があります。



コンポーネントの [Web] タブの設定は、当該コンポーネントのアプリケーション領域設定に定義します。詳細については、111 ページ「アプリケーション領域の Web 設定の定義」を参照してください。

注： [Web] タブは、Web アドインがインストールされ、読み込まれている場合にのみ利用できます。

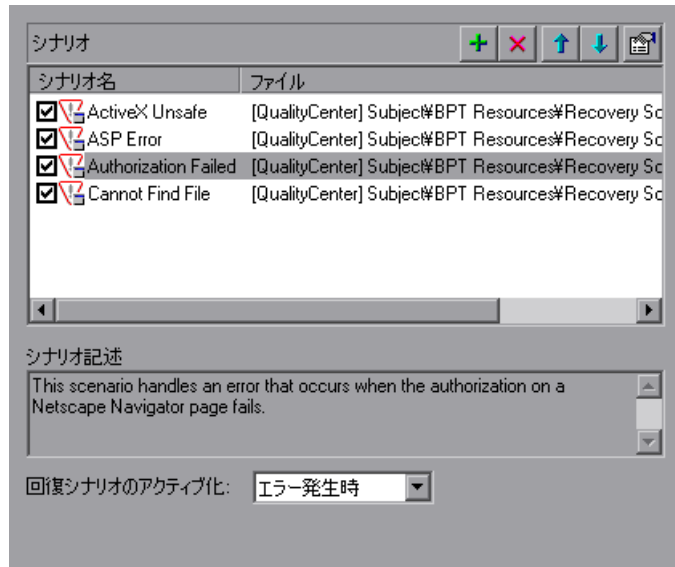
[Web] タブには、次の項目があります。

設定	詳細
[ブラウザナビゲーションのタイムアウト]	QuickTest がコンポーネントのステップを実行する前に、Web ページの読み込みが完了するのを待機する最長の時間（秒単位）を表示します。
[ユーザ名]	このオプションはコンポーネントの場合は該当しません。
[パスワード]	このオプションはコンポーネントの場合は該当しません。
[詳細設定]	このオプションはコンポーネントの場合は該当しません。

回復シナリオの設定の表示

回復シナリオの設定では、ビジネス・コンポーネントが実行セッション中の予期しないイベントやエラーからどのように回復するかを指定できます。

[ビジネス コンポーネントの設定] ダイアログ・ボックスの [回復] タブには、現在のコンポーネントに関連付けられているアプリケーション領域に関連付けられているすべての回復のシナリオが表示されます。



コンポーネントの回復シナリオの設定は、当該コンポーネントのアプリケーション領域に定義します。詳細については、112 ページ「アプリケーション領域の回復シナリオ設定の定義」を参照してください。

[回復] タブには、次の項目があります。

設定	詳細
[シナリオ]	コンポーネントに関連付けられているアプリケーション領域を通じてコンポーネン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。シナリオ・タイプはアイコンによって示されます。詳細については、114 ページ「関連回復シナリオの指定」を参照してください。

設定	詳細
[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]	<p>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 に指示する設定を表示します。</p> <ul style="list-style-type: none"> • [各ステップごと]：回復メカニズムはステップの終了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した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。

第 10 章

関数ライブラリ・ウィンドウのカスタマイズ

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、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。

- ▶ 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

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、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、関数ライブラリ・ウィンドウでの、関数ライブラリの表示方法を変更できます。また、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。

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。

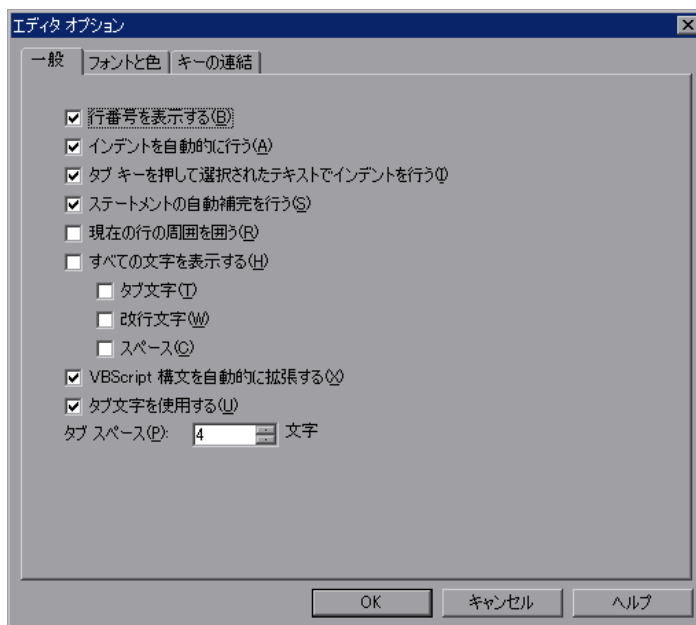
[印刷] ダイアログ・ボックスのオプションを使用して、関数ライブラリの印刷方法を変更することもできます。詳細については、を参照してください。関数ライブラリを使った作業の詳細については、第21章「ユーザ定義関数および関数ライブラリを使用した作業」を参照してください。

エディタの動作のカスタマイズ

関数ライブラリ・ウィンドウにおける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。関数ライブラリを使った作業の詳細については、第21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。

- 1 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。



- 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]	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。
[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。

オプション	詳細
<p>[ステートメントの自動補完を行う]</p>	<p>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。</p> <ul style="list-style-type: none"> ● 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：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。
<p>[現在の行の周囲を囲う]</p>	<p>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。</p>
<p>[すべての文字を表示する]</p>	<p>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。</p>

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 ThenEnd 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。

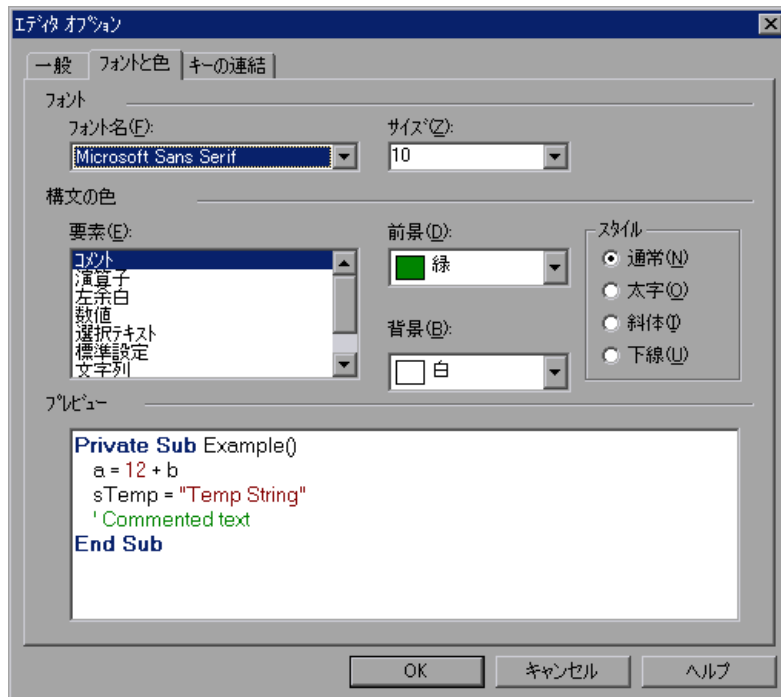
- 4 [OK] をクリックし、変更を保存してダイアログ・ボックスを閉じます。

エレメントの見映えのカスタマイズ

QuickTest 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、数字などの、さまざまなエレメントが含まれています。QuickTest スクリプトのそれぞれのエレメントは、異なる色で表示できます。エキスパート・ビューに表示されるすべてのエレメントのフォントや文字の大きさを指定することもできます。スクリプトの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、スクリプト内のすべてのコメントを、青の文字と黄色の背景で表示することもできます。

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。

- 1 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。



- 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、関数ライブラリ内のエレメントが、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、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。

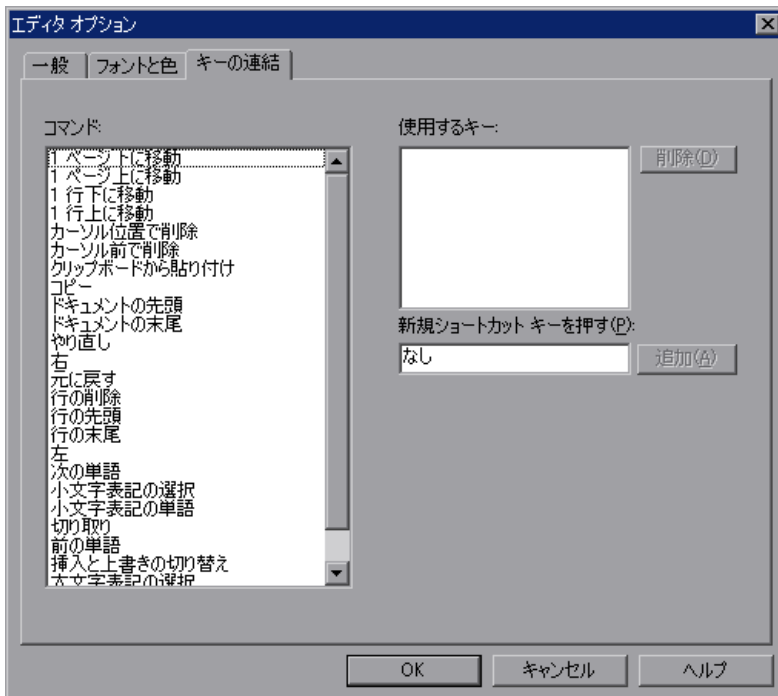
編集コマンドのカスタマイズ

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 メニュー・ショートカット・キーの一覧は、57 ページ「ショートカット・キーを使用したコマンドの実行」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。

- 1 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。



- 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 11 章

Web イベント記録の設定

QuickTest で要求に合うイベントが記録されない場合、Web オブジェクトの種類ごとに記録するイベントを設定できます。

本章では、次の項目について説明します。

- ▶ Web イベント記録の設定について
- ▶ 標準で使用するイベント記録設定の選択
- ▶ イベント記録設定のカスタマイズ
- ▶ マウスの右ボタン・クリックの記録
- ▶ ユーザ定義イベント設定ファイルの保存と読み込み
- ▶ イベント記録設定のリセット

Web イベント記録の設定について

QuickTest では、Web ベースのアプリケーションで実行したイベントを記録することで、コンポーネントを作成できます。イベントとは、状態の変更などの操作に応じて行われる通知や、ユーザがドキュメントを表示しているときにマウスをクリックしたり、キーを押したりした結果行われる通知のことです。記録する必要のあるイベントの数が、QuickTest の標準設定に応じて自動的に記録される数よりも多い場合や少ない場合があります。そのような場合、[Web イベント記録の設定] ダイアログ・ボックスで、3つの定義済みの設定から、いずれかを選択することで標準で使用するイベント記録設定を変更できます。また、特定の条件に合わせて、イベント記録の設定を個別にカスタマイズすることもできます。

たとえば QuickTest では、通常はリンク・オブジェクト上の `mouseover` イベントは記録されません。しかし、マウスを対象の上に移動したときに生じる動作（マウスオーバ動作）がリンクに関連付けられている場合は、`mouseover` イベントを記録することが重要になるかもしれません。この場合、リンク・オブジェクトが操作に関連付けられているときに、リンク・オブジェクト上の `mouseover` イベントが必ず記録されるように、設定をカスタマイズできます。

注：

イベント設定はグローバルな設定のため、設定を変更した後で記録されるすべてのコンポーネントに影響します。

イベント設定の変更は、すでに記録されたコンポーネントには影響しません。必要なイベントが QuickTest で記録されなかった場合や、不要なイベントが記録された場合は、イベント記録設定を変更し、コンポーネントの中でその変更の影響を受ける部分を再度記録します。

[ユーザ定義 Web イベント記録の設定] の設定に対する変更は、すでに開いているブラウザには直接反映されません。[Web イベント記録の設定] ダイアログ・ボックスで必要な変更を行い、その変更を既存のコンポーネントに適用するには、開いているブラウザを更新し、新しい記録セッションを開始します。

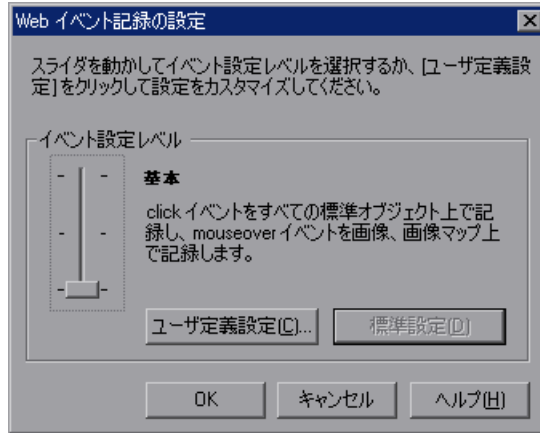
標準で使用するイベント記録設定の選択

[Web イベント記録の設定] ダイアログ・ボックスでは、3 つの定義済みのイベント設定レベルを選択できます。特に指定がなければ、QuickTest では**基本**記録設定レベルが使用されます。必要なイベントの一部が QuickTest で記録されない場合は、イベント設定のレベルを高くする必要があります。

レベル	詳細
[基本]	<p>標準設定。</p> <ul style="list-style-type: none"> • 画像、ボタン、ラジオ・ボタンなどの標準的な Web オブジェクトに対するクリック・イベントを必ず記録します。 • フォーム内での送信イベントを必ず記録します。 • ハンドラまたは動作が関連付けられている他のオブジェクトでのクリック・イベントを記録します。ハンドラおよび動作の詳細については、267 ページ「応答条件」を参照してください。 • イメージおよびイメージ・マップに対する mouseover イベントに続くイベントが当該オブジェクトを対象とするものである場合に限り、当該 mouseover イベントを記録します。
[中]	<p>基本レベルで記録されるオブジェクト以外に、HTML タグ・オブジェクトの <DIV>, , <TD> に対するクリック・イベントも記録します。</p>
[高]	<p>基本レベルで記録されるオブジェクト以外に、ハンドラまたは動作が関連付けられているオブジェクトに対する mouseover イベント、mousedown イベント、および double-click イベントを記録します。ハンドラおよび動作の詳細については、267 ページ「応答条件」を参照してください。</p>

標準で使用するイベント記録の設定を設定するには、次の手順を実行します。

- 1 [ツール] > [Web イベント記録の設定] を選択します。[Web イベント記録の設定] ダイアログ・ボックスが表示されます。



- 2 スライダを使って、標準で使用するイベント記録設定を選択します。

ヒント：[ユーザ定義設定] ボタンをクリックして [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスを開けます。ここでイベント記録の設定をカスタマイズできます。詳細については、イベント記録設定のカスタマイズを参照してください。

[標準設定] ボタンをクリックして、スライダを [基本] レベルの位置に戻すことができます。

- 3 [OK] をクリックします。

イベント記録設定のカスタマイズ

標準のイベント設定レベルで必要な記録が行われない場合は、[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスを使って、イベント記録設定をカスタマイズできます。

[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスでは、いくつかの方法でイベント記録をカスタマイズできます。次のことができます。

- ▶ QuickTest で特別な応答設定または記録設定を適用する対象となるオブジェクトの追加または削除。
- ▶ QuickTest が応答するべきイベントの追加または削除。
- ▶ イベントの応答と記録の設定変更。

イベント記録の設定をカスタマイズするには、次の手順を実行します。

- 1 [ツール] > [Web イベント記録の設定] を選択します。[Web イベント記録の設定] ダイアログ・ボックスが表示されます。
- 2 [ユーザ定義設定] ボタンをクリックします。[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスが開きます。

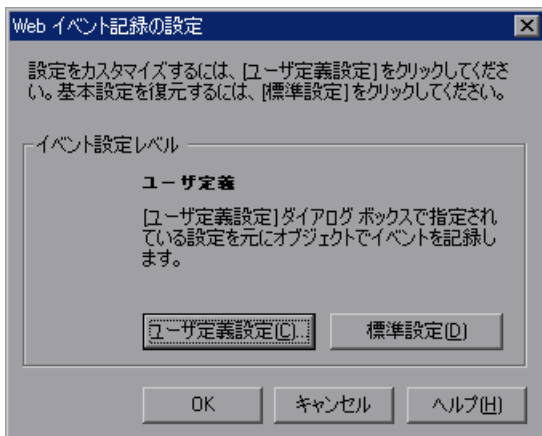


3 次のオプションを使って、イベント記録設定をカスタマイズします。

オプション	詳細
オブジェクト表示枠	<p>Web テスト・オブジェクト・クラスと HTML タグ・オブジェクトのリストが表示されます。</p> <ul style="list-style-type: none"> ● オブジェクトを追加するには、[オブジェクト] > [追加] を選択します。 ● HTML タグ・オブジェクトのみ削除できます。リストから HTML オブジェクトを削除するには、[オブジェクト] > [削除] を選択します。 <p>詳細については、264 ページ「ユーザ定義のオブジェクト・リストに対するオブジェクトの追加と削除」を参照してください。</p>
イベント表示枠	<p>オブジェクトに関連付けられているイベントのリストが表示されます。</p> <ul style="list-style-type: none"> ● イベント表示枠にイベントを追加するには、[イベント] > [追加] を選択します。 ● イベントを削除するには、[イベント] > [削除] を選択します。 <p>詳細については、266 ページ「オブジェクトの応答イベントの追加と削除」を参照してください。</p>
[イベント名]	イベントの名前。

オプション	詳細
【応答】	<p>QuickTest がイベントに応答する際の基準。</p> <ul style="list-style-type: none"> ● Always : 常にイベントに応答します。 ● If Handler : ハンドラが結び付けられているイベントに応答します。ハンドラは、Web ページに含まれているコードであり、通常はスクリプト言語で書かれている関数またはルーチンです。対応するイベントが発生したときに制御が渡されます。 ● If Behavior : DHTML 動作が結び付けられているイベントに応答します。DHTML 動作は、ページ上の特定の機能または動作をカプセル化します。ページ上の標準的な HTML 要素に適用されている場合、その要素の標準設定の動作が拡張されます。 ● If Handler or Behavior : ハンドラまたは動作が結び付けられているイベントに応答します。 ● Never : イベントに一切応答しません。 <p>詳細については、267 ページ「イベントの応答設定と記録設定の変更」を参照してください。</p>
【記録】	<p>選択されたオブジェクトのイベントの記録を有効/無効にする、あるいは同じオブジェクトに対してその後イベントが発生した場合のみイベントの記録を有効/無効にします。</p>
【リセット】	<p>あらかじめ設定されていたレベルに設定を戻します。</p>

- 4 [OK] をクリックします。[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスが閉じます。[Web イベント記録の設定] ダイアログ・ボックスのスライダが消え、設定の内容として「**ユーザ定義**」と表示されます。



ユーザ定義のオブジェクト・リストに対するオブジェクトの追加と削除

[ユーザ定義 Web イベント記録の定義] ダイアログ・ボックスには、オブジェクト階層内のオブジェクトのリストが表示されます。階層の一番上には、[任意の Web オブジェクト] があります。[任意の Web オブジェクト] の設定は、テスト対象となる Web ページ上のすべてのオブジェクトにのうち、特にイベント記録設定が設定されていないすべてのオブジェクトに適用されます。その下には [Web オブジェクト] と [HTML タグ オブジェクト] カテゴリがあり、どちらにもオブジェクトのリストが含まれています。

[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックス内のオブジェクトを使って作業するときは、次の原則に従います。

- ▶ オブジェクトが [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックス内のリストに表示されている場合、そのオブジェクトの設定は [任意の Web オブジェクト] の設定に優先します。
- ▶ [Web オブジェクト] カテゴリのオブジェクト・リストに対する追加と削除はできませんが、任意のオブジェクトの設定を変更できます。
- ▶ Web ページの任意の HTML タグ・オブジェクトを、[HTML タグ オブジェクト] カテゴリに追加できます。

イベント設定オブジェクト・リストにオブジェクトを追加するには、次の手順を実行します。

- 1 [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスで、[オブジェクト] > [追加] を選択します。[HTML タグ オブジェクト] リストに、「新しいオブジェクト」が表示されます。



- 2 名前を変更するには、[新しいオブジェクト] をクリックします。HTML タグをそのまま名前として入力します。

標準設定では、新しいオブジェクトは、ハンドラが関連付けられている **onclick** イベントの応答と記録を行うように設定されています。

イベントの追加および削除の詳細については、オブジェクトの応答イベントの追加と削除を参照してください。応答設定および記録設定の詳細については、267 ページ「イベントの応答設定と記録設定の変更」を参照してください。

[HTML タグ オブジェクト] リストからオブジェクトを削除するには、次の手順を実行します。

- 1 [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスの [HTML タグ オブジェクト] カテゴリで、削除するオブジェクトを選択します。
- 2 [オブジェクト] > [削除] を選択します。選択されていたオブジェクトがリストから削除されます。

注：[Web オブジェクト] カテゴリからオブジェクトを削除することはできません。

オブジェクトの応答イベントの追加と削除

QuickTest がオブジェクトへ応答するトリガとなるイベントのリストに対して、イベントの追加や削除を行うことができます。

オブジェクトに応答イベントを追加するには、次の手順を実行します。

- 1 [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスの中で、イベントを追加する対象となるオブジェクト、または [任意の Web オブジェクト] を選択します。
- 2 [イベント] > [追加] を選択します。使用できるイベントのリストが開きます。



- 3 追加するイベントを選択します。イベントは [イベント名] 列にアルファベット順で表示されます。標準設定では、QuickTest はハンドラが結び付けられているイベントに回答し、そのイベントを（それが何らかのレベルで回答されている限り）必ず記録します。

応答設定および記録設定の詳細については、イベントの応答設定と記録設定の変更を参照してください。

オブジェクトから応答イベントを削除するには、次の手順を実行します。

- 1 [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスの中で、イベントを削除する対象となるオブジェクト，または [任意の Web オブジェクト] を選択します。
- 2 削除するイベントを [イベント名] カラムから選択します。
- 3 [イベント] > [削除] を選択します。[イベント名] カラムからイベントが削除されます。

イベントの応答設定と記録設定の変更

オブジェクトごとにリスト表示される各イベントについて、応答条件を選択し、記録するかどうかを設定できます。

注： 応答と記録の設定は相互に独立しています。つまり、オブジェクトに対するイベントに応答しても、それを記録しないことや、オブジェクトに対するイベントに応答せずに、そのイベントを記録することができます。詳細については、269 ページ「イベントの応答と記録を行うためのヒント」を参照してください。

応答条件

イベントごとに、イベント・ハンドラがイベントに関連付けられている場合、DHTML 動作がイベントに関連付けられている場合、イベント・ハンドラまたは DHTML 動作がイベントに関連付けられている場合に、オブジェクトでイベントが発生するたびに応答するように、あるいはイベントに一切応答しないように QuickTest に指示できます。

イベント・ハンドラは、Web ページに含まれているコードであり、通常はスクリプト言語で書かれている関数またはルーチンです。対応するイベントが発生したときに制御が渡されます。

DHTML 動作は、ページ上の特定の機能または動作をカプセル化します。ページ上の標準的な HTML 要素に適用されている場合、その要素の標準設定の動作が拡張されます。

イベントに対する応答条件を指定するには、次の手順を実行します。

- 1 [ユーザー定義 Web イベント記録設定] ダイアログ・ボックスの中で、応答条件を変更する対象となるオブジェクト、または [任意の Web オブジェクト] を選択します。
- 2 変更するイベントの行で、[応答] カラムから必要な応答条件を選択します。

イベント名	応答	記録
onclick	If Handler	Enabled
onmousedown	If Handler	Enabled
onmouseover	If Handler	Enabled
4	Always	
5	If Handler	
6	If Behavior	
7	If Handler or Behavior	
8	Never	
9		
10		
11		
12		

[Always], [If Handler], [If Behavior], [If Handler or Behavior], または [Never] のいずれかを選択できます。

記録ステータス

イベントごとに、対象イベントを記録するようにも、記録しないようにも、あるいは次のイベントが選択されたイベントに依存している場合だけ記録するようにも設定できます。

- ▶ **[Enabled]** : QuickTest が対象オブジェクトあるいはイベントの「バブリング先」である別のオブジェクトを応答している場合、イベントが生じるたびにそれを記録します。

バブリングとは、子オブジェクトで発生したイベントが、イベントを処理するイベント・ハンドラに遭遇するまで、HTML コード内の階層をさかのぼる処理です。

- ▶ **[Disabled]** : 指定されたイベントを記録せず、イベント・バブリングがある場合はそれを無視します。

- ▶ **[Enabled on next event]** : **[Enabled]** との唯一の違いは、以降のイベントが同じオブジェクトで発生した場合にのみイベントを記録することです。たとえば、マウスオーバ動作によって画像リンクが変わるとします。この画像の上をマウスが通過するたびに、`mouseover` イベントを記録する必要はないかもしれませんが、リンクが `mouseover` イベント後に表示される画像によってのみ有効になるので、`mouseover` イベントを、同じオブジェクトに対するクリック・イベントの前に記録することが重要となります。このオプションは、Image および WebArea オブジェクトに対してのみ適用されます。

イベントの記録ステータスを設定するには、次の手順を実行します。

- 1 [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスで、記録ステータスを変更する対象となるオブジェクト、または **[任意の Web オブジェクト]** を選択します。
- 2 変更対象のイベントの行で、記録ステータスを **[記録]** カラムから選択します。

イベント名	応答	記録
onclick	Always	Enabled
onmouseover	IfHandler	Enabled
3		Disabled
4		Enabled
5		
6		
7		
8		
9		
10		
11		
12		

イベントの応答と記録を行うためのヒント

理想的な応答と記録の設定を見つけるのが困難な場合があります。これらの設定を行うときには、次のガイドラインに留意します。

- ▶ オブジェクト表示枠の異なるオブジェクトの設定が矛盾する場合、QuickTest は、特定の **[HTML タグ オブジェクト]** の設定を最優先し、次に **[Web オブジェクト]** の設定を優先します。QuickTest が **[任意の Web オブジェクト]** の設定を Web オブジェクトに適用する対象は、**[HTML タグ オブジェクト]** と **「標準オブジェクト」** の範囲のどちらにも定義されていない Web オブジェクトに限ります。

- ▶ オブジェクトのイベントを記録するには、QuickTest がイベントに応答し、イベントが生じたときにそれを記録するように指定します。子プロジェクトのイベントは、そのイベントに対するハンドラまたは動作が親オブジェクトに含まれている場合にも応答できます。また、親オブジェクトのイベントは、そのイベントに対するハンドラまたは動作が子オブジェクトに含まれていても、応答できます。

ただし、ソース・オブジェクト（どの親オブジェクトがハンドラまたは動作を含んでいるかによらずイベントが実際に発生したオブジェクト）に対するイベントの記録は有効にしておかなければなりません。

たとえば、`onmouseover` イベント・ハンドラのあるテーブル・セルに2つのイメージが含まれているとします。マウスがどちらかのイメージの上を移動すると、バブリングによってイベントがそのセルに送られます。このバブリングにはマウスがどちらのイメージの上で移動したのかを示す情報が含まれていません。この `mouseover` イベントは、次のようにして記録できます。

- ▶ `<TD>` タグの `mouseover` イベントに対する応答を `[If Handler]` に設定する一方で（イベントが生じたときに QuickTest にイベントが「応答する」ように）、イベントは記録しないようにした上で、`` タグの `mouseover` イベントに対する応答は `[Never]` に設定し、`` タグに対する記録は `[Enabled]`（`<TD>` レベルで応答した後の画像に対する `mouseover` イベントを記録します）に設定します。
- ▶ `` タグの `mouseover` イベントに対する応答を `[Always]`（イメージ・タグが動作またはハンドラを含んでいなくても `mouseover` イベントを応答する）に設定し、`` タグに対する記録を `[Enabled]`（イメージに対する `mouseover` イベントを記録する）に設定します。
- ▶ 多数のオブジェクト上の多数のイベントを応答するように設定すると、QuickTest のパフォーマンスが低下することがあるので、応答の設定は、必要なオブジェクトに限定することをお勧めします。
- ▶ まれに、イベントが発生するオブジェクト（ソース・オブジェクト）に応答していると、イベントが妨害されることがあります。

QuickTest を使ってアプリケーションの記録を始めるまでは、そのアプリケーションが正常に動作していたのであれば、**応答**の設定によって妨害が生じている可能性があります。

この問題がマウス・イベントのときに発生した場合、[詳細 Web オプション] ダイアログ・ボックスで、適切な [標準 Windows マウスイベントの使用] オプションを選択してみます。詳細については、221 ページ「詳細 Web オプション」を参照してください。

この問題がキーボード・イベントまたは内部イベントのときに発生した場合、あるいは [標準 Windows マウスイベントの使用] オプションを指定しても問題が解決しない場合、そのイベントに対するソース・オブジェクトでの応答の設定を [Never] に設定して (ただし、ソース・オブジェクトに対する記録の設定は Enabled のまま)、親オブジェクトに対する応答の設定を [Always] にします。

マウスの右ボタン・クリックの記録

QuickTest では、マウスの左ボタン、中央ボタン、右ボタンを使用したクリックを記録できます。標準設定では、左クリックのみ記録されますが、右ボタンや中央ボタンのクリックも記録するように設定を変更できます。

OnClick イベントが発行されると **Click** ステートメントが記録されます。

QuickTest は、マウスの各ボタンに対して設定されたイベントを応答することによって、マウス・ボタンを区別します。標準設定では、OnMouseUp イベントに応答しますが、[Web イベント記録の設定] ダイアログ・ボックスを使用して、OnMouseDown イベントに応答するように設定することもできます。

注：

複数のマウス・ボタンの同時クリックの記録はサポートされていません。

QuickTest では、ブラウザのショートカット・メニューを開く右クリック、およびショートカット・メニューからの項目の選択は記録されません。スクリプトを手作業で変更してこれらのオプションを有効にする方法の詳細については、次のナレッジ・ベース項目を参照してください。

- ▶ **問題 ID 31270** : How to replay right-clicking on an object to open a pop-up menu (オブジェクト上で右クリックしてポップアップ・メニューを開く動作の再生方法)

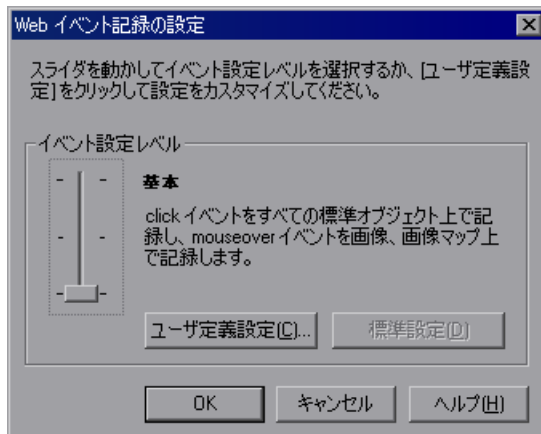
- ▶ **問題 ID 27184** : How to select an item from a right-click menu (右クリック・メニューからの項目の選択方法)
-

マウスの右クリックを記録するための QuickTest の設定

設定ファイルを手作業で変更してから読み込むことによって、マウスの右クリックを記録するよう QuickTest に指示します。

マウスの右クリックを記録するよう QuickTest に指示するには、次の手順を実行します。

- 1 [ツール] > [Web イベント記録の設定] を選択します。[Web イベント記録の設定] ダイアログ・ボックスが表示されます。



- 2 [ユーザ定義設定] ボタンをクリックします。[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスが開きます。



- 3 [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスで、[ファイル] > [設定に名前を付けて保存] を選択します。[名前を付けて保存] ダイアログ・ボックスが表示されます。
- 4 Web イベント記録設定ファイルを保存するフォルダに移動し、設定ファイル名を入力します。設定ファイルの拡張子は **.xml** です。
- 5 [保存] をクリックしてファイルを保存し、ダイアログ・ボックスを閉じます。
- 6 編集のために、保存した設定ファイルをテキスト・エディタで開きます。設定ファイルでは定義済みの構造が使用されています。この XML ファイルの構造の詳細については、276 ページ「Web イベント記録の設定の XML 構造について」を参照してください。

Web オブジェクトに関連するファイルの冒頭部分を以下に示します。

```
- <XML>
- <Object Name="Any Web Object">
  <Event Name="onclick" Listen="2" Record="2" />
  <Event Name="oncontextmenu" Listen="2" Record="2" />
  <Event Name="onkeydown" Listen="1" Record="2" />
  <Event Name="onmouseover" Listen="2" Record="1" />
- <Event Name="onmouseup" Listen="2" Record="1">
  <Property Name="button" Value="2" Listen="2" Record="2" />
```

Property Name 引数が、マウス・ボタンの記録を制御します。マウス・ボタンの値は次のように定義されています。

- ▶ 1 : 左
- ▶ 2 : 右
- ▶ 4 : 中央

7 ファイルを次のように編集します。

- ▶ **onmouseup** イベントに対するマウスの左クリックを記録するには、次の行を追加します。

```
<Property Name="button" Value="1" Listen="2" Record="2"/>
```

- ▶ **onmousedown** イベントに対するマウスの右クリックおよび左クリックを記録するには、次の行を追加します。

```
<Event Name="onmousedown" Listen="2" Record="1">
```

```
  <Property Name="button" Value="2" Listen="2" Record="2"/>
```

```
  <Property Name="button" Value="1" Listen="2" Record="2"/>
```

```
</Event>
```

注 : **onmouseup** または **onmousedown** のどちらか1つのイベントのみを使用してマウス・クリックを処理してください。両方のイベントを使用した場合は、1つではなく2つのクリックが記録されます。標準設定では、QuickTestは**onmouseup** イベントに応答します。

- 8 ファイルを保存します。
- 9 [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスで、[**ファイル**] > [**設定の読み込み**] を選択します。[ファイルを開く] ダイアログ・ボックスが開きます。
- 10 編集した設定ファイルを保存したフォルダに移動し、ファイルを選択して、[**開く**] をクリックします。[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスが再び開きます。
- 11 [**OK**] をクリックします。新しい設定が読み込まれ、すべての設定が XML 設定ファイルで定義したものに対応ようになります。以降、記録する Web オブジェクトはこれらの新しい設定に従って記録されます。

ユーザ定義イベント設定ファイルの保存と読み込み

[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスで行った変更は保存して、必要なときに読み込むことができます。

XML ファイルを読み込む前に変更することもできます。この XML ファイルの構造の詳細については、276 ページ「Web イベント記録の設定の XML 構造について」を参照してください。

ユーザ定義設定を保存するには、次の手順を実行します。

- 1 イベント記録設定を、必要に合わせてカスタマイズします。設定をカスタマイズする方法の詳細については、261 ページ「イベント記録設定のカスタマイズ」を参照してください。
- 2 [ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスで、[**ファイル**] > [**設定に名前を付けて保存**] を選択します。[名前を付けて保存] ダイアログ・ボックスが表示されます。
- 3 イベント設定ファイルを保存するフォルダに移動し、設定ファイル名を入力します。設定ファイルの拡張子は **.xml** です。
- 4 [**保存**] をクリックしてファイルを保存し、ダイアログ・ボックスを閉じます。

ユーザ定義設定を読み込むには、次の手順を実行します。

- 1 [ツール] > [Web イベント記録の設定] を選択し、[ユーザ定義設定] をクリックします。[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスが表示されます。
- 2 [ファイル] > [設定の読み込み] を選択します。[ファイルを開く] ダイアログ・ボックスが開きます。
- 3 読み込むイベント設定ファイル (.xml) を見つけて、[開く] をクリックします。ダイアログ・ボックスが閉じ、選択した設定が読み込まれます。

Web イベント記録の設定の XML 構造について

Web イベント記録の設定の XML ファイルは、特定の構造形式になっています。ファイルに変更を加える場合や、独自のファイルを作成する場合、設定が有効であるためにはこの形式に従う必要があります。

XML ファイルの例を以下に示します。

```
<XML>
  <Object Name="Any Web Object">
    <Event Name="onclick" Listen="2" Record="2"/>
    <Event Name="onmouseup" Listen="2" Record="1">
      <Property Name="button" Value="2" Listen="2" Record="2"/>
    </Event>
  </Object>
  ...
  ...
  ...
  <Object Name="WebList">
    <Event Name="onblur" Listen="1" Record="2"/>
    <Event Name="onchange" Listen="1" Record="2"/>
    <Event Name="onfocus" Listen="1" Record="2"/>
  </Object>
</XML>
```

次の値を使用して応答条件と記録ステータスのオプションを XML 形式で定義します。

設定	指定可能な値
[応答]	1 : Always 2 : If Handler 4 : If Behavior 6 : If Handler or Behavior 0 : Never
[記録]	1 : Disabled 2 : Enabled 6 : Enabled on Next Event

イベント記録設定のリセット

ユーザ定義の設定を行った後で標準の設定に戻すには、[Web イベント記録の定義] ダイアログ・ボックスで、イベント記録設定を基本レベルにリセットします。

注： 標準設定をリセットすると、ユーザ定義の設定は完全になくなります。変更内容を失わないようにするには、イベント設定ファイルに設定を保存しておく必要があります。詳細については、275 ページ「ユーザ定義イベント設定ファイルの保存と読み込み」を参照してください。

[Web イベント記録の設定] ダイアログ・ボックスを使用して設定を基本レベルにリセットするには、次の手順を実行します。

- 1** [ツール] > [Web イベント記録の設定] を選択します。[Web イベント記録の設定] ダイアログ・ボックスが表示されます。
- 2** [標準設定] をクリックします。標準設定スライダが再び表示され、すべてのイベント設定が**基本**イベント記録設定レベルに戻されます。
- 3** 別の定義済みレベルを選択するには、259 ページ「標準で使用するイベント記録設定の選択」を参照してください。

[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスを使用して、設定を特定の（基本）ユーザ定義設定に戻すこともできます。これにより、その状態からカスタマイズを開始できます。

[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスを使用して、設定を特定のユーザ定義レベルにリセットするには、次の手順を実行します。

- 1 [ツール] > [Web イベント記録の設定] を選択します。[Web イベント記録の設定] ダイアログ・ボックスが表示されます。
- 2 [ユーザ定義設定] ボタンをクリックします。[ユーザ定義 Web イベント記録の設定] ダイアログ・ボックスが表示されます。
- 3 [戻した後の値] ボックスで、使用する定義済みイベント記録レベルを選択します。
- 4 [リセット] をクリックします。すべてのイベント設定が、選択したレベルの標準設定に戻されます。

第4部

テスト・オブジェクトおよび
オブジェクト・リポジトリを使用した作業

第 12 章

テスト・オブジェクト・モデルについて

本章では、QuickTest によるアプリケーション内のオブジェクトの学習方法および識別方法について説明します。また、「テスト・オブジェクト」と「実行環境オブジェクト」の概念について説明し、オブジェクトで使用可能なメソッドとその構文を表示する方法について取り上げます。これにより、コンポーネント用の操作の作成時に、[エキスパートビュー] でスクリプトにステートメントを簡単に追加したり、関数でテスト・オブジェクトおよびメソッドを簡単に使用したりできるようになります。

本章では、次の項目について説明します。

- ▶ テスト・オブジェクト・モデルについて
- ▶ 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の表示
- ▶ オブジェクト・スパイを使用したオブジェクト・メソッドとメソッド構文の表示

テスト・オブジェクト・モデルについて

QuickTest は、テスト・オブジェクトとその期待プロパティおよび期待値を学習、識別することによって、動的に変化するアプリケーションをテストします。記録中、QuickTest は、人間が写真を見てその詳細を記憶するのとほぼ同じ方法で、アプリケーションの各オブジェクトを分析します。

次の各項では、テスト・オブジェクト・モデルに関する概念を紹介し、収集した情報を基に QuickTest によってアプリケーションがどのようにテストされるかを説明します。

QuickTest による記録時のオブジェクトの学習方法について

QuickTest は人と同じように学習します。たとえば、ある実験で、Johnny という被験者にピクニックの様子を写した写真を2～3秒間見て、その写真に写っているある何かを覚えてもらうと予告したとします。そして、1週間後に同じ写真か別の似たような写真を見て、その何かを識別してもらうとします。

Johnny は写真を見る前に、テストに備えて、テスト担当者に指示される記憶対象のどのような特徴を覚えるかを考えます。もちろん、Johnny はそれが人、物、動物、植物のどれであるかは、特に意識することもなく覚えるでしょう。記憶する対象が人ならば、性別、皮膚の色、そして年齢を覚えるつもりでいます。そして、動物ならば、その動物の種類や色などといったことを覚えるつもりでいます。

テスト担当者は、Johnny に写真を見せ、地面に敷いたシートの上に座っている3人の子供のうちの1人を指差します。Johnny は女の子が白人で8歳ほどだと記憶します。しかし、写真の他の部分を見て、その記述に当てはまる子供がもう1人いることに気付きます。そこで、覚える予定だった特徴に加え、その女の子の髪が茶色で長いことも覚えます。

これで、写真の中で Johnny が覚えた特徴と一致するのは1人だけになりますから、1週間後に見せられる写真が少し違っていても、かなり確実にその女の子を見分けられるでしょう。

それでもまだ少し写真を見る時間が残っているので、念のために、他の子供達とのもっと微妙な違いを見つけられるかどうか試みます。

もし、写真の中の似ている2人の子供が見分けのつかないような双子ならば、Johnny はもっと一時的でしかない特徴も覚えようとするかもしれません。たとえば、シートの上で座っている位置です。別の見せられる写真でも子供達の位置関係が同じでありさえすれば、Johnny は指定された女の子を見分けることができるでしょう。

QuickTest は、記録処理中にオブジェクトを学習するとき、非常に良く似た方法を使います。

まず、記録対象のオブジェクトを「見て」、これを「テスト・オブジェクト」として保存し、そのオブジェクトに合うテスト・オブジェクト・クラスを決定します。ちょうど Johnny が、項目が人間、動物、植物、物のどれであるかをすぐに検査したようにです。QuickTest は、テスト・オブジェクトを、たとえば標準 Windows ダイアログ・ボックス (Dialog)、Web ボタン (WebButton)、あるいは Visual Basic スクロール・バー・オブジェクト (VbScrollBar) のようにクラス分けします。

次に、QuickTest には、テスト・オブジェクト・クラスごとに必ず学習する必須プロパティのリストがあります。これは Johnny が写真を見る前に覚えようと考えていた特徴のリストに似ています。オブジェクトを対象に記録を行うとき、QuickTest は必ずこれらの標準プロパティ値を学習し、ページ、ダイアログ・ボックス、または他の親オブジェクト上の残りのオブジェクトを「見て」、この記述がオブジェクトを一意に識別するのに十分であるかどうかを確認します。そうでない場合、QuickTest は、一意の記述ができるまで、記述に補足プロパティを1つずつ追加していきます。これは Johnny がリストに髪の毛の長さや色という特徴を追加したのに似ています。補足プロパティがない場合や、あっても一意の記述を作成するには足りない場合、QuickTest は、オブジェクトのページやソース・コードでの位置のような、特殊な**序数識別子**を追加して、一意の記述を作成します。これは、写真に写っている子供のうちの2人が、見分けのつかない双子であったならば、子供の座っているシート上での位置を覚えることとちょうど同じです。

QuickTest が実行セッション中にオブジェクトを識別する方法について

QuickTest は、人間が行うのと非常によく似た方法で実行セッション中にオブジェクトを識別します。

実験の続きとして、新しいけれども似たような環境で、最初に識別したものと同じ「項目」を識別するように Johnny が指示されたとします。

最初に、元の写真を見せます。Johnny は、記憶するように指示された、8歳くらいの長い茶色の髪の毛の同じ白人の女の子を探し、すぐに見付けます。2番目の写真では、子供達は遊具を使って遊んでいます。Johnny はやはり、同じ基準を使ってその女の子を簡単に識別できます。

同様に、実行セッション中、QuickTestは記録時に学習したテスト・オブジェクトの記述と正確に一致する**実行環境オブジェクト**を探します。記録中に一意の記述を作成するのに使われた必須およびすべての補足プロパティが完全一致することが期待されます。アプリケーションのオブジェクトが大きく変更されていなければ、記録中に学習した記述はQuickTestがそのオブジェクトを識別するのにほとんど常に十分です。これは、ほとんどのオブジェクトに当てはまりますが、アプリケーションには以降の実行セッションで識別が困難なオブジェクトが含まれている可能性もあります。

Johnnyの実験の最終段階を考えてみましょう。この段階では、テスト担当者はJohnnyに、同じ場所で同じ家族を写したもう1枚の写真を見せますが、子供達は成長しており、運動場ではもっと多くの子供達が遊んでいます。Johnnyはまず、別の写真で女の子を見分けるために使ったのと同じ特徴を持った女の子を探します(テスト・オブジェクト)が、写真の中には長い茶色の髪の女の子は1人もいません。幸運にもJohnnyは、前の週に最初に写真を見たときにその女の子について他の情報も覚えていました。彼は、その女の子の髪が短くブロンドになっているにもかかわらず彼女を見つけ出す(実行環境オブジェクト)ことができました。

どのようにやってのけたのでしょうか。まず、見つけなければならない、自分が知っている特徴は何であるかを考えました。Johnnyは、探しているのが白人の女性であることに変わりはないことを分かっており、この基準に一致する人が誰も見付からなければ、彼女は写真に写っていないと判断します。

この新しい写真に写っている4人の白人女性に的を絞ると、今度は問題の女の子を見分けるために使ってきた他の特徴(年齢、髪の色、髪の長さ)について考えます。同じ人物であっても、年月が経過しており、覚えている他の特徴も変わっているかもしれないことは分かっています。

したがって、白人の女の子は誰一人として長い茶色の髪ではありませんでしたが、そうした特徴は無視し、記憶にある目と鼻を持つ女の子を探します。目の似ている少女が2人いますが、元の写真で見たような小さな鼻をしているのは、そのうちの1人だけであることに気がきます。あまり目立つ特徴ではありませんが、これらを使って少女を識別できます。

QuickTest のスマート認識メカニズムでは、これに非常に良く似た消去法を使って、記録時の記述がもはや正確とはいえなくなっているオブジェクトを識別します。QuickTest は、テスト・オブジェクトのプロパティ値が変更されていても、スマート認識でオブジェクトを識別することによって、コンポーネントの再利用可能性を維持します。スマート認識の詳細については、第14章「オブジェクトの認識の設定」を参照してください。

次に、テスト・オブジェクト、実行環境オブジェクト、オブジェクト・プロパティ、必須プロパティ、補足プロパティ、スマート認識など、ここで示された概念について説明します。これらの概念を理解することにより、アプリケーションに対して適切に設計された、機能的なコンポーネントを作成できるようになります。

テスト・オブジェクト・モデルの概念の適用

「テスト・オブジェクト・モデル」とは、QuickTest がアプリケーション内のオブジェクトを表現するために使用する、オブジェクト・タイプまたはクラスの大規模なセットです。それぞれのテスト・オブジェクト・クラスには、そのクラスのオブジェクトを一意に識別するためのプロパティのリストと、QuickTest がそのクラスについて記録できる一連の関連メソッドが含まれています。

テスト・オブジェクトとは、アプリケーションに含まれる実際のオブジェクトを表現するために、QuickTest によってコンポーネント内に作成されるオブジェクトです。QuickTest によってオブジェクトに関する情報が格納されます。この情報は、テストの実行中にそのオブジェクトを識別し検査するために使用されます。

「実行環境オブジェクト」とは、実行セッション中にメソッドの実行対象となるアプリケーションに含まれる実際のオブジェクトです。

テストを記録しながらアプリケーションで操作を実行すると、QuickTest によって次の処理が行われます。

- ▶ 操作の実行対象のオブジェクトを表す QuickTest テスト・オブジェクト・クラスが識別され、適切なテスト・オブジェクトが作成される
- ▶ アプリケーションに含まれるオブジェクトのプロパティの現在の値が読み取られ、プロパティのリストと値がテスト・オブジェクトに格納される
- ▶ オブジェクトに一意的な名を付ける。一般には主要なプロパティのどれかの値に基づく
- ▶ オブジェクトに対して実行した操作が、適切な QuickTest テスト・オブジェクト・メソッドを使用して記録される

たとえば、次の HTML ソース・コードによって表示される、[Search] ボタンをクリックしたとします。

```
<INPUT TYPE="submit" NAME="Search" VALUE="Search">
```

QuickTest では、クリックしたオブジェクトが **WebButton** テスト・オブジェクトとして識別されます。また、**Search** という論理名で **WebButton** オブジェクトが作成され、**Search WebButton** について次のプロパティと値が記録されます。

名前	値
記述プロパティ	
type	submit
name	Search
html tag	INPUT

さらに、**WebButton** に対して **Click** メソッドが実行されたことが記録されます。

コンポーネント内の最初のステップが、[Search] ボタンのクリックだとすると、QuickTest には次のようにステップが表示されます。

項目	操作	注釈
 Search	Click	"Search" button をクリックする。

コンポーネントを実行すると、QuickTestにより、テスト・オブジェクト・クラスと記述（オブジェクトを一意に識別するためのテスト・オブジェクトのプロパティと値のセット）に基づいて、アプリケーションに含まれている各オブジェクトが識別されます。テスト・オブジェクトとそのプロパティおよび値のリストは、オブジェクト・リポジトリに格納されます。前の例では、実行セッション中、QuickTestによりオブジェクト・リポジトリ内で **Search** という名前を持つ WebButton オブジェクトが検索され、その記述が調べられます。次に、見つかった記述に基づいて、QuickTestによって、HTML タグが INPUT、タイプが submit、値が「Search」である WebButton オブジェクトがアプリケーションの中で検索されます。該当するオブジェクトが見つかり、そのオブジェクトに対して **Click** メソッドが実行されます。

テスト・オブジェクト記述について

QuickTest では、記録時に、オブジェクト・クラスごとに一連の標準設定のプロパティが学習されます。また、コンポーネントを実行するときに、この記述によってオブジェクトが識別されます。

たとえば、標準設定の QuickTest では、操作を記録する対象となる各 Web 画像の画像タイプ (plain image や image button など)、HTML タグ、**代替**テキストが認識されます。

The image shows two screenshots from the QuickTest application. The top screenshot is a dialog box titled 'オブジェクトのプロパティ' (Object Properties). It displays the name 'Sign-In' and class 'Image'. Below is a table of '記述プロパティ' (Description Properties):

名前	値
image type	Image Button
html tag	INPUT
alt	Sign-In

Annotations point to '名前: Sign-In' as 'テスト・オブジェクト名' (Test Object Name) and 'クラス: Image' as 'テスト・オブジェクト・クラス' (Test Object Class). The '記述プロパティ' section is labeled '既定のプロパティ' (Default Properties). The bottom screenshot shows a table of test objects:

項目	操作	値	割り当て	コメント	注釈
Action1					
Welcome: Mercury Tours					
Welcome: Mercury Tours					
userName	Set	"mercury"			"userNam..."
password	SetSecure	"444c2cd2177cdabb38c9ec182ba63155e03216dbc"			暗号化さ...
Sign-In	Click	24.3			"Sign-In" L...

An annotation points to the 'Sign-In' row as 'テスト・オブジェクト名' (Test Object Name). The 'userName' row has an 'image icon' annotation.

親オブジェクト内のオブジェクトを一意に識別するのにこれら3つの必須のプロパティ値では十分ではない場合、QuickTestは一意的記述を作成するため、いくつかの補助プロパティまたは予約識別子、あるいはその両方を追加します。

コンポーネントを実行すると、学習された記述と一致するオブジェクトが QuickTest によって検索されます。記述と一致するオブジェクトが見つからなかったり複数の一致するオブジェクトが見つかったりした場合、QuickTest ではオブジェクトを識別するため、スマート認識機能が使用されることがあります。

QuickTest によってアプリケーション内のオブジェクトの記述の記録に使用される必須、補助、および序数識別子のプロパティは、ユーザが設定できます。また、スマート認識機能の有効化と設定もユーザができます。詳細については、第14章「オブジェクトの認識の設定」を参照してください。

テスト・オブジェクトおよび実行環境オブジェクトのプロパティおよびメソッドについて

各テスト・オブジェクトのテスト・オブジェクト・プロパティ・セットは、QuickTest によって作成され、維持されます。また、各実行環境オブジェクトの実行環境オブジェクト・プロパティ・セットは、オブジェクト・クリエータ（たとえば、Microsoft Internet Explorer オブジェクトの場合は Microsoft、Netscape オブジェクトの場合は Netscape、ActiveX オブジェクトの場合は製品開発者など）によって作成され、維持されます。

同様に、テスト・オブジェクト・メソッドは、テストを記録しながらオブジェクトに対してメソッドを実行したときに、QuickTest によって認識され、記録されるメソッド、およびコンポーネントの実行時に QuickTest によって実行されるメソッドです。実行環境オブジェクトのメソッドは、オブジェクト・クリエータによって定義される、アプリケーション内のオブジェクトのメソッドです。実行環境オブジェクトのメソッドは、**Object** プロパティを通じてアクセスおよび実行できます。

Object プロパティを使用した実行環境メソッドの実行の詳細については、656 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」を参照してください。

コンポーネントを記録しながら実行する各テスト・オブジェクト・メソッドは、テストに個別のステップとして記録されます。コンポーネントを実行すると、QuickTest によって、記録されているテスト・オブジェクト・メソッドが実行環境オブジェクトに対して実行されます。

テスト・オブジェクトのプロパティは、コンポーネントを記録するときに、Web サイトまたはアプリケーション内のオブジェクトから値がキャプチャされるプロパティです。QuickTest では、これらのプロパティの値によって、実行セッション中にアプリケーション内の実行環境オブジェクトが識別されます。

アプリケーションに含まれるオブジェクトのプロパティの値は、アプリケーションを起動するたび、または特定の条件に基づいて、動的に変化することがあります。テスト・オブジェクトと実行環境オブジェクトのプロパティの値を一致させるには、コンポーネントの作成時に手作業で、または実行セッション中に **SetTOPProperty** ステートメントを使用して（関数ライブラリに定義されている操作によって）、テスト・オブジェクトのプロパティを変更します。また、正規表現を使って、定義した条件やパターンに基づいてプロパティの値を特定できます。オブジェクトのプロパティの変更に関する詳細については、第13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。

コンポーネントに格納されているテスト・オブジェクトのプロパティ値は、[オブジェクトのプロパティ] または [オブジェクトリポジトリ] ダイアログ・ボックスで確認したり、変更したりできます。また、デスクトップ上にある任意のオブジェクトのテスト・オブジェクト・プロパティの現在の値は、オブジェクト・スパイの [プロパティ] タブで確認できます。[オブジェクトのプロパティ] および [オブジェクトリポジトリ] ダイアログ・ボックスの詳細については、319 ページ「テスト・オブジェクトのプロパティの変更」を参照してください。オブジェクト・スパイを使用したテスト・オブジェクトのプロパティ値の表示に関する詳細については、290 ページ「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の表示」を参照してください。

テスト・オブジェクト・メソッドの構文、およびデスクトップ上にある任意のオブジェクトの実行環境メソッドは、オブジェクト・スパイの [メソッド] タブで確認できます。詳細については、294 ページ「オブジェクト・スパイを使用したオブジェクト・メソッドとメソッド構文の表示」を参照してください。

関数ライブラリで定義した操作を使用して、**GetTOPProperty** および **SetTOPProperty** ステートメントを追加すると、実行セッション中にテスト・オブジェクトのプロパティ値の取得と変更が行えます。テストの実行中に実行環境オブジェクトのプロパティ値を読み込むには、**GetROProperty** ステートメントを追加します。詳細については、656 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」を参照してください。


使用可能なテスト・オブジェクト・メソッドまたはオブジェクトのプロパティによって必要な機能が提供されない場合、**Object** プロパティを使用して、任意の実行環境オブジェクトの内部メソッドおよび内部プロパティにアクセスできます。また、オブジェクトのプロパティ属性を使用しても、ユーザ定義のプロパティに従って、アプリケーション内の **Web** オブジェクトを識別できます。詳細については、657 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。

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、『**QuickTest Professional オブジェクト・モデル・リファレンス**』を参照してください。

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の表示

オブジェクト・スパイを使用すると、起動しているアプリケーションに含まれる任意のオブジェクトのプロパティを表示することができます。オブジェクト・スパイ・ポインタを使用して対象オブジェクトを指定します。オブジェクト・スパイによって、[オブジェクトスパイ] ダイアログ・ボックスの [プロパティ] タブに、選択したオブジェクトの階層ツリーと、そのプロパティおよび値が表示されます。

オブジェクトのプロパティを表示するには、次の手順を実行します。

- 1 ブラウザまたはアプリケーションで、調査対象のオブジェクトが含まれるページを表示します。
- 2  [ツール] > [オブジェクトスパイ] を選択するか、[オブジェクトスパイ] ツールバー・ボタンをクリックして、[オブジェクトスパイ] ダイアログ・ボックスを開き、[プロパティ] タブを表示します。または、[オブジェクトリポジトリ] ダイアログ・ボックスの [オブジェクトスパイ] ボタンをクリックします。[オブジェクトリポジトリ] ダイアログ・ボックスの詳細については、306 ページ「[オブジェクトリポジトリ] ウィンドウについて」を参照してください。

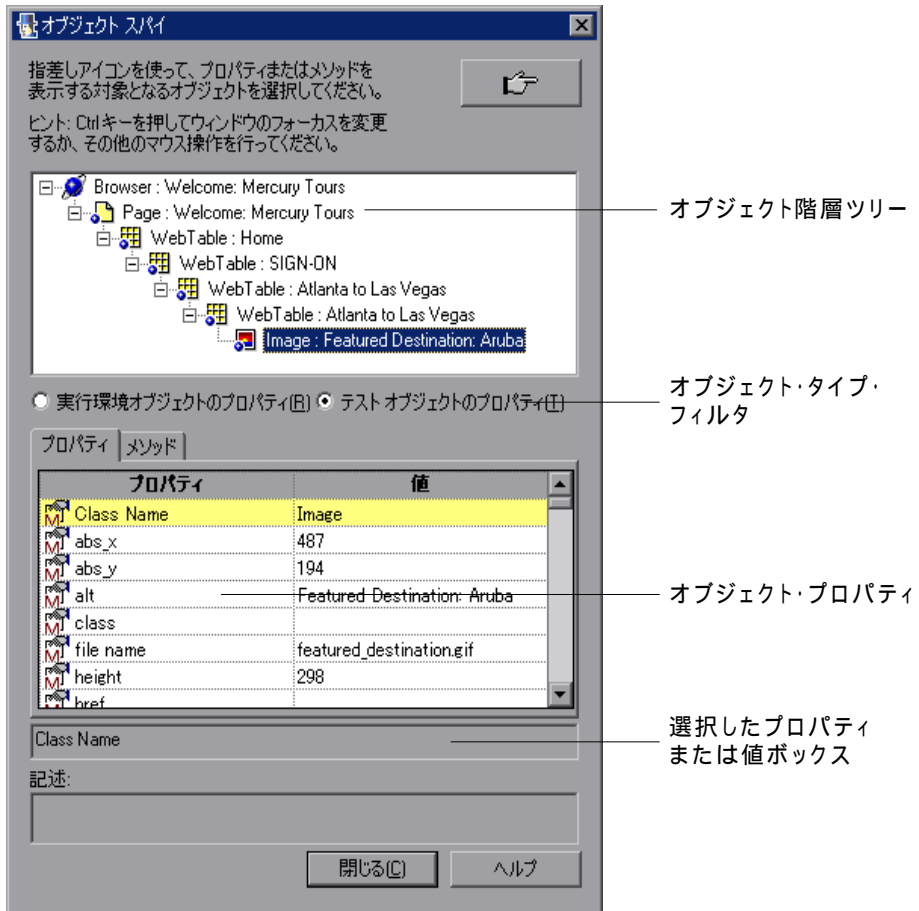


- 3 [オブジェクトスパイ] ダイアログ・ボックスで指差しマークをクリックします。QuickTest およびオブジェクト・スパイが最小化され、起動しているアプリケーションの任意のオブジェクトをポイントしてクリックできるようになります。

注：調査対象のウィンドウの一部が別のウィンドウの裏に隠れている場合、一部が隠れているウィンドウに指差しマークを数秒間置いたままにします。ウィンドウが前景に表示されます。これにより、必要なオブジェクトをポイントし、クリックできるようになります。ウィンドウを前面に出すのに必要な時間の設定は、[オプション] ダイアログ・ボックスの [一般] タブで行えます。詳細については、第8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。また、左側の CTRL キーを押しながら、ウィンドウのフォーカスを変更することもできます。さらには、選択対象オブジェクトが含まれているウィンドウが最小化されている場合は、Ctrl キーを押しながら Windows タスク・バーでアプリケーションを右クリックし、ショートカット・メニューから [元のサイズに戻す] を選択して、ウィンドウを表示できます。

- 4 特定のイベントが生じないと調査対象オブジェクトが表示されない場合（たとえば、ショートカット・メニューを表示するための右クリックやマウスオーバーなど）は、左側の CTRL キーを押し下げます。すると、指差しマークが一時的に標準の矢印に変わり、イベントを実行できるようになります。調査対象のオブジェクトが表示されたら、左側の CTRL キーを放します。矢印が再び指差しマークに変わります。

- 5 プロパティを表示する対象となるオブジェクトをクリックします。オブジェクト・スパイが表示され、ツリー内にオブジェクト階層ツリーと、選択されているオブジェクトのプロパティが表示されます。



ヒント: [オブジェクト スパイ] ダイアログ・ボックスはサイズを変更できます。これは、深い階層、または長いプロパティ名や値がある場合に、スクロールせずすべての情報を表示できるので役立ちます。

- 6 テスト・オブジェクトのプロパティを表示するには、[**テスト オブジェクトのプロパティ**] ラジオ・ボタンをクリックします。実行環境オブジェクトのプロパティを表示するには、[**実行環境オブジェクトのプロパティ**] ラジオ・ボタンをクリックします。

ヒント：

関数内で **Object** プロパティを使用することで、[オブジェクト スパイ] に表示されるランタイムのプロパティの値を取得できます。詳細については、658 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。



また、関数内で **GetTOProperty** メソッドおよび **SetTOProperty** メソッドを使用することで、テスト・オブジェクトに対し、テスト・オブジェクト・プロパティの値を取得したり設定したりできます。また、**GetROProperty** メソッドを使用すれば、実行セッション中にアプリケーション内のオブジェクトから現在のプロパティの値を取得できます。詳細については、656 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」を参照してください。

- 7 表示されているツリーに含まれる別のオブジェクトのプロパティを表示するには、ツリーの中でそのオブジェクトをクリックします。
- 8 オブジェクトのプロパティまたは値をクリップボードにコピーするには、プロパティまたは値をクリックします。選択したプロパティまたは値が下のボックスに表示されます。ボックスに表示されたテキストを選択して強調表示し、CTRL + C キーでテキストをクリップボードにコピーするか、強調表示したテキストを右クリックし、メニューから [**コピー**] を選択します。

オブジェクト・スパイを使用したオブジェクト・メソッドとメソッド構文の表示

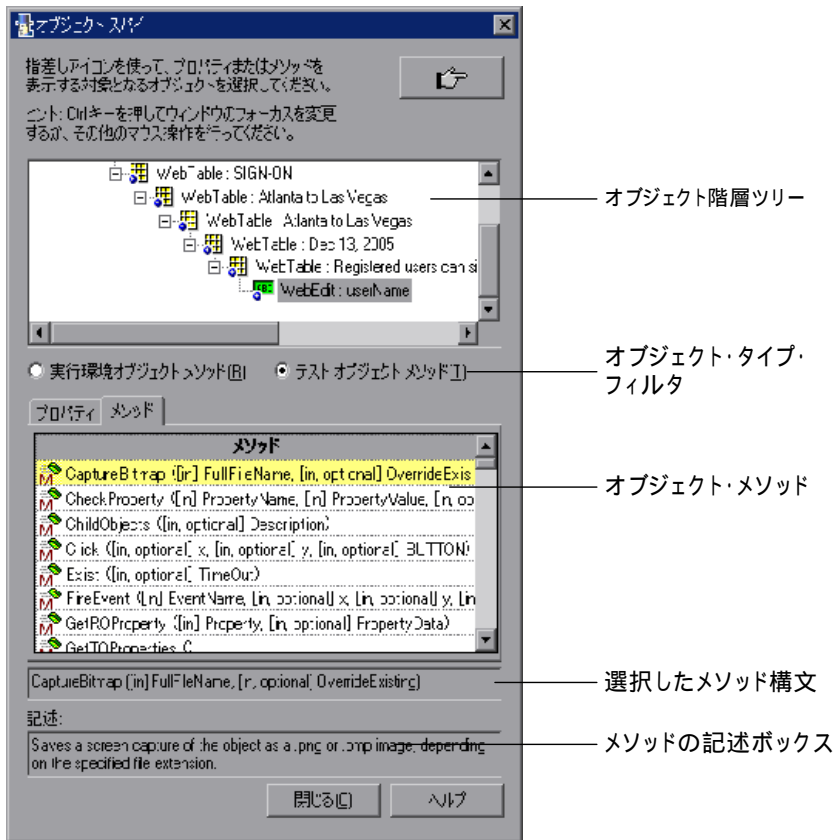
オブジェクト・スパイでは、オブジェクトのプロパティを表示できるほか、オブジェクトに関連付けられている実行環境オブジェクト・メソッドおよびテスト・オブジェクト・メソッドを表示したり、選択したメソッドの構文を表示したりできます。オブジェクト・スパイ・ポインタを使用して対象オブジェクトを指定します。オブジェクト・スパイにより、[オブジェクトスパイ] ダイアログ・ボックスの [メソッド] タブに、オブジェクト階層ツリーと、選択したオブジェクトに関連付けられている実行環境オブジェクト・メソッドまたはテスト・オブジェクト・メソッドが表示されます。

オブジェクト・メソッドを表示するには、次の手順を実行します。

- 1 ブラウザまたはアプリケーションで、調査対象のオブジェクトが含まれるページを表示します。
- 2  [ツール] > [オブジェクトスパイ] を選択するか、[オブジェクトスパイ] ツールバー・ボタンをクリックして、[オブジェクトスパイ] ダイアログ・ボックスを開きます。または、[オブジェクトリポジトリ] ダイアログ・ボックスの [オブジェクトスパイ] ボタンをクリックします。[オブジェクトリポジトリ] ダイアログ・ボックスの詳細については、306 ページ「[オブジェクトリポジトリ] ウィンドウについて」を参照してください。
- 3 [メソッド] タブをクリックします。
- 4  指差しマークをクリックします。QuickTest およびオブジェクト・スパイが最小化され、起動しているアプリケーションの任意のオブジェクトをポイントできるようにします。

注：調査対象オブジェクトの一部が別のウィンドウの裏に隠れている場合、一部が隠れているウィンドウに指差しマークを数秒間置いたままにします。ウィンドウが前景に表示されます。これにより、必要なオブジェクトをポイントし、クリックできるようになります。このオプションの設定は、[オプション] ダイアログ・ボックスで行えます。詳細については、第8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。また、左側の CTRL キーを押しながら、ウィンドウのフォーカスを変更することもできます。さらには、選択対象オブジェクトが含まれているウィンドウが最小化されている場合は、Ctrl キーを押しながら Windows タスク・バーでアプリケーションを右クリックし、ショートカット・メニューから [元のサイズに戻す] を選択して、ウィンドウを表示できます。

- 5 特定のイベントが生じないと調査対象オブジェクトが表示されない場合（たとえば、ショートカット・メニューを表示するための右クリックやマウスオーバーなど）は、左側の CTRL キーを押し下げます。すると、指差しマークが一時的に標準の矢印に変わり、イベントを実行できるようになります。調査対象のオブジェクトが表示されたら、左側の CTRL キーを放します。矢印が再び指差しマークに変わります。
- 6 関連付けられているメソッドを表示する対象となるオブジェクトをクリックします。オブジェクト・スパイが表示され、オブジェクト階層ツリーと、ツリーで選択したオブジェクトに関連付けられている [実行環境オブジェクト] または [テストオブジェクト] メソッドが表示されます。



ヒント：[オブジェクト スパイ] ダイアログ・ボックスはサイズを変更できません。これは、深い階層、または長い構文がある場合に、スクロールせずにすべての情報を表示できるので役立ちます。

- 7 テスト・オブジェクトのメソッドを表示するには、[**テストオブジェクトメソッド**] ラジオ・ボタンをクリックします。実行環境オブジェクトのメソッドを表示するには、[**実行環境オブジェクトメソッド**] ラジオ・ボタンをクリックします。

ヒント：Object プロパティを使用することで、[オブジェクトスパイ] に表示される実行環境オブジェクト・メソッドを実行できます。詳細については、658 ページ「実行環境オブジェクト・メソッドのアクティブ化」を参照してください。

- 8 表示されているツリーに含まれる別のオブジェクトのメソッドを表示するには、ツリーの中でそのオブジェクトをクリックします。
- 9 メソッドの構文をクリップボードにコピーするには、リストの中でそのメソッドをクリックします。構文が[選択したメソッドの構文] ボックスに表示されます。[選択したメソッド構文] ボックスの中でテキストを選択して強調表示し、CTRL+C キーでテキストをクリップボードにコピーするか、強調表示したテキストを右クリックし、メニューから [**コピー**] を選択します。

第 13 章

テスト・オブジェクトを使用した作業

本章では、コンポーネントに含まれるテスト・オブジェクトの管理と保守の方法について説明します。また、テスト・オブジェクトのプロパティを変更する方法、QuickTest によるオブジェクトの識別方法を変更する方法を取り上げます。これは、動的に変化するオブジェクトを対象に作業する場合に便利です。

本章では、次の内容について説明します。

- ▶ テスト・オブジェクトを使用した作業について
- ▶ オブジェクト・リポジトリの種類について
- ▶ [オブジェクト リポジトリ] ウィンドウについて
- ▶ テスト・オブジェクトのプロパティの表示と変更
- ▶ リポジトリ・パラメータ値の割り当て
- ▶ オブジェクト・リポジトリへのオブジェクトの追加
- ▶ 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
- ▶ オブジェクト・リポジトリからのオブジェクトの削除
- ▶ オブジェクトの場所の特定
- ▶ 実行セッション中のテスト・オブジェクトの操作
- ▶ 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

テスト・オブジェクトを使用した作業について

QuickTest でコンポーネントを実行すると、アプリケーション上でのマウス・カーソルの移動、オブジェクトのクリック、およびキーボード入力によって、ユーザの操作がシミュレートされます。ユーザと同様に QuickTest も、アプリケーションを使用できるようになるにはインタフェースを学習する必要があります。QuickTest にとってインタフェースの学習とは、アプリケーションのオブジェクトとそれに対応するプロパティ値を学習し、そうしたオブジェクト記述をオブジェクト・リポジトリに格納することです。

QuickTest はテスト・オブジェクトを学習すると、それらをコンポーネントのローカル・オブジェクト・リポジトリに格納します。格納したオブジェクトはローカル・オブジェクト・リポジトリに維持するか、共有オブジェクト・リポジトリに格納するか選択することができます。オブジェクトをローカル・オブジェクト・リポジトリに格納すると、オブジェクトは特定のコンポーネントにおいてのみ利用できるようになり、その他のコンポーネントでは使用できません。オブジェクトを1つ以上の共有オブジェクト・リポジトリに格納すると、複数のコンポーネント(アプリケーション領域を通じて)においてオブジェクトを使用できるようになります。また必要に応じて、ローカル・オブジェクト・リポジトリと共有オブジェクト・リポジトリを組み合わせて使用することもできます。ローカルおよび共有オブジェクト・リポジトリの詳細については、301 ページ「オブジェクト・リポジトリの種類について」を参照してください。

アプリケーション内のオブジェクトのプロパティ値の中に、QuickTest によってオブジェクトの識別に使用されるプロパティ値と異なるものがあると、コンポーネントは失敗することがあります。そのため、アプリケーション内にあるオブジェクトのプロパティ値に変更があった場合、既存のコンポーネントを継続して使用するには、該当するオブジェクト・リポジトリ内にある該当するテスト・オブジェクトのプロパティ値を修正する必要があります。

本章で説明するように、ローカル・オブジェクト・リポジトリに格納されたオブジェクトは、[オブジェクトリポジトリ] ウィンドウを使用して変更できます。共有オブジェクト・リポジトリ内のオブジェクトは、オブジェクト・リポジトリ・マネージャを使用して変更できます。オブジェクト・リポジトリ・マネージャの詳細については、第15章「オブジェクト・リポジトリの管理」を参照してください。また、本章で説明するように、共有オブジェクト・リポジトリからローカル・オブジェクト・リポジトリにオブジェクトをコピーし、[オブジェクトリポジトリ] ウィンドウを使用してオブジェクトのローカル・コピーを変更することもできます。

オブジェクト・リポジトリの種類について

テスト・オブジェクトは、2種類のオブジェクト・リポジトリに格納できます。1つは**共有オブジェクト・リポジトリ**で、もう1つは**ローカル・オブジェクト・リポジトリ**です。共有オブジェクト・リポジトリでは、複数の（アプリケーション領域を通じて）コンポーネントから読み取り専用モードでアクセス可能なファイルにテスト・オブジェクトが格納されます。ローカル・オブジェクト・リポジトリでは、ある特定のコンポーネントに関連付けられているファイルにオブジェクトが格納されるため、格納されたオブジェクトにアクセスできるのはそのコンポーネントのみです。

コンポーネントを計画し作成するときは、コンポーネント内のオブジェクトを格納する方法について検討する必要があります。各コンポーネントのオブジェクトを対応するローカル・オブジェクト・リポジトリに格納できます。また、コンポーネント内のオブジェクトを1つ以上の共有オブジェクト・リポジトリに格納することもできます。オブジェクトを共有オブジェクト・リポジトリに格納し、それらのリポジトリをコンポーネントのアプリケーション領域に関連付ければ、複数のコンポーネントでそれらのオブジェクトを使用できるようになります。各コンポーネントでは、必要に応じてローカル・オブジェクト・リポジトリのオブジェクトと共有オブジェクト・リポジトリのオブジェクトを組み合わせて使用できます。必要があれば、ローカル・オブジェクトを共有オブジェクト・リポジトリに移動することもできます。そうすることで、オブジェクトを複数の場所ではなく1つの共有された場所で管理できるため、管理作業が軽減され、コンポーネントの再利用性が向上します。詳細については、303ページ「ローカルと共有のどちらのオブジェクト・リポジトリを使用するか」の決定」を参照してください。

QuickTest をまだ使い慣れていない場合は、ローカル・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。そうすれば、対応するコンポーネントからアクセス可能なローカル・オブジェクト・リポジトリにすべてのオブジェクトが自動的に保存されるため、コンポーネントの記録や実行のために共有オブジェクト・リポジトリの作成、選択、変更を行う必要がありません。ローカル・オブジェクト・リポジトリ内のオブジェクトを変更した場合、その変更はほかのコンポーネントに影響しません。

テストに慣れている場合は、共有オブジェクト・リポジトリにオブジェクトを保存するのがおそらく最も効率的です。この方法では、複数のコンポーネントに同じオブジェクトが含まれる場合、それらのコンポーネントに対して同じ共有オブジェクト・リポジトリ・ファイルが使用されるため、複数のコンポーネントに適用されるオブジェクト情報の保管場所が一元化されます。アプリケーション内のオブジェクトに変更があった場合は、この共有オブジェクト・リポジトリを使用するすべてのコンポーネントに対してオブジェクトを一元的に更新できます。

同じコンポーネントに関連付けられているローカル・オブジェクト・リポジトリと共有オブジェクト・リポジトリの両方に、同じ名前と記述を持つオブジェクトが存在する場合は、そのコンポーネントに対するローカルのオブジェクト定義が使用されます。同じコンポーネントに関連付けられている複数の共有オブジェクト・リポジトリに、同じ名前と記述を持つオブジェクトが存在する場合は、そのコンポーネントに共有オブジェクト・リポジトリが関連付けられている順序に従って、そのオブジェクトが最初に見つかった場所のオブジェクト定義が使用されます。共有オブジェクト・リポジトリの関連付けの詳細については、94 ページ「共有オブジェクト・リポジトリの管理」を参照してください。

ローカル・オブジェクトはコンポーネントとともにローカルに保存され、そのコンポーネントからのみアクセスできます。共有オブジェクト・リポジトリを使用する場合は、複数のコンポーネントに対して同じオブジェクト・リポジトリを使用できます。また、1つのコンポーネントで複数のオブジェクト・リポジトリを使用することもできます。

既存のコンポーネントを開いて操作する場合は、そのコンポーネントが関連付けられているアプリケーション領域で指定したオブジェクト・リポジトリが常に使用されます。コンポーネントからアクセスしたときの共有オブジェクト・リポジトリは読み取り専用です。共有オブジェクト・リポジトリを編集するには、オブジェクト・リポジトリ・マネージャを使用します。

ローカルと共有のどちらのオブジェクト・リポジトリを使用するか の決定

オブジェクトの保存先を選択するには、ローカル・オブジェクト・リポジトリと共有オブジェクト・リポジトリの違いを理解しておく必要があります。

一般に、ローカル・オブジェクト・リポジトリはコンポーネントを簡単に記録して実行する場合に使いやすいモードで、特に次の場合に適しています。

- ▶ 1つのアプリケーション、インタフェース、またはオブジェクト・セットについて、対応するコンポーネントの数が1つしかないか、または非常に少ない。
- ▶ テスト・オブジェクトのプロパティを頻繁に変更する可能性がない。

反対に、共有オブジェクト・リポジトリは、通常次の場合に適しています。

- ▶ キーワード駆動方式で（記録を使用せずに）コンポーネントを作成する。
- ▶ 同じアプリケーション、インタフェース、またはオブジェクト・セットの要素を検査するコンポーネントが複数ある。
- ▶ アプリケーション内のオブジェクトのプロパティが時々変更される変更する可能性があり、テスト・オブジェクトのプロパティを定期的に更新あるいは変更する必要がある。

ローカル・オブジェクト・リポジトリについて

ローカル・オブジェクト・リポジトリを使用する場合、コンポーネントごとに個別のオブジェクト・リポジトリが使用されます（また、必要であれば1つ以上の共有オブジェクト・リポジトリを使用することもできます。詳細については、304ページ「共有オブジェクト・リポジトリについて」を参照してください）。ローカル・オブジェクト・リポジトリに対しては、該当するコンポーネント内からすべての編集操作が可能です。

ローカル・オブジェクト・リポジトリを使って作業するときは、次の点に注意してください。

- ▶ **QuickTest** によって、コンポーネントごとに新規（空）のオブジェクト・リポジトリが作成されます。
- ▶ アプリケーション内のオブジェクトに対する操作を記録すると、それらのオブジェクトに関する情報は対応するローカル・オブジェクト・リポジトリ内に格納されます（関連付けられている共有オブジェクト・リポジトリにそれらのオブジェクトがまだ存在しない場合）。1つ以上の共有オブジェクト・リポジトリが既にコンポーネントに関連付けられている場合でも、**QuickTest** は、すべての新しいオブジェクトをローカル・オブジェクト・リポジトリに追加します（これは、関連付けられている共有オブジェクト・リポジトリの1つと同じ名前と記述を持つオブジェクトが存在しないことが前提です）。
- ▶ 子オブジェクトがローカル・オブジェクト・リポジトリに追加される際、その親が共有オブジェクト・リポジトリ内にある場合は、親がローカル・オブジェクト・リポジトリに自動的に移動されます。
- ▶ 新規コンポーネントを作成するたびに、対応する新規ローカル・オブジェクト・リポジトリが作成され、オブジェクトを記録または学習すると、そのローカル・オブジェクト・リポジトリにテスト・オブジェクトが追加され始めます。
- ▶ アプリケーション内の1つのオブジェクトを対象に2つの異なるコンポーネントを記録または学習すると、そのオブジェクトはそれぞれのローカル・オブジェクト・リポジトリに個別のテスト・オブジェクトとして格納されます。
- ▶ コンポーネントを保存すると、ローカル・オブジェクト・リポジトリがそのコンポーネントとともに自動的に保存されます。このローカル・オブジェクト・リポジトリには、（共有オブジェクト・リポジトリと異なり）独立のファイルとしてアクセスできません。

共有オブジェクト・リポジトリについて

共有オブジェクト・リポジトリを使用すると、選択したコンポーネントのアプリケーション領域のために指定した共有オブジェクト・リポジトリが使用されます。1つ以上の共有オブジェクト・リポジトリを使用できます。（アクセス元のコンポーネントが限定されているオブジェクトがある場合は、それらを各コンポーネントのローカル・オブジェクト・リポジトリに保存することもできます。詳細については、303 ページ「ローカル・オブジェクト・リポジトリについて」を参照してください。）

コンポーネントの作成を開始した後は、追加の共有オブジェクト・リポジトリを指定できます。新しい共有オブジェクト・リポジトリを作成し、それらをコンポーネントに関連付けることもできます。コンポーネントを実行する前に、そのコンポーネントで使用するオブジェクト・リポジトリにコンポーネント内のすべてのオブジェクトが含まれていることを確認する必要があります。含まれていなければ、コンポーネントが失敗する場合があります。詳細については、343 ページ「オブジェクト・リポジトリへのオブジェクトの追加」を参照してください。

共有オブジェクト・リポジトリに変更を加えるには、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、第15章「オブジェクト・リポジトリの管理」を参照してください。

共有オブジェクト・リポジトリを使って作業するときは、次の点に注意してください。

- ▶ 共有オブジェクト・リポジトリまたはローカル・オブジェクト・リポジトリにすでに存在しているオブジェクトに対する操作を記録すると、QuickTest によって既存の情報が使用され、当該オブジェクトはオブジェクト・リポジトリに追加されません。
- ▶ 子オブジェクトがローカル・オブジェクト・リポジトリに追加される際、その親が共有オブジェクト・リポジトリ内にある場合は、親がローカル・オブジェクト・リポジトリに自動的に移動されます。
- ▶ オブジェクトに対する操作を記録すると、そのオブジェクトは共有オブジェクト・リポジトリに追加されません。その代わりに、オブジェクトを学習するか、オブジェクトを対象とするステップを記録すると、（共有オブジェクト・リポジトリではなく）ローカル・オブジェクト・リポジトリに新規オブジェクトが追加されます（同じオブジェクトが、関連付けられている共有オブジェクト・リポジトリにすでに存在しない限り）。

ローカル・オブジェクトは、共有オブジェクト・リポジトリにエクスポートできます。詳細については、364 ページ「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」を参照してください。

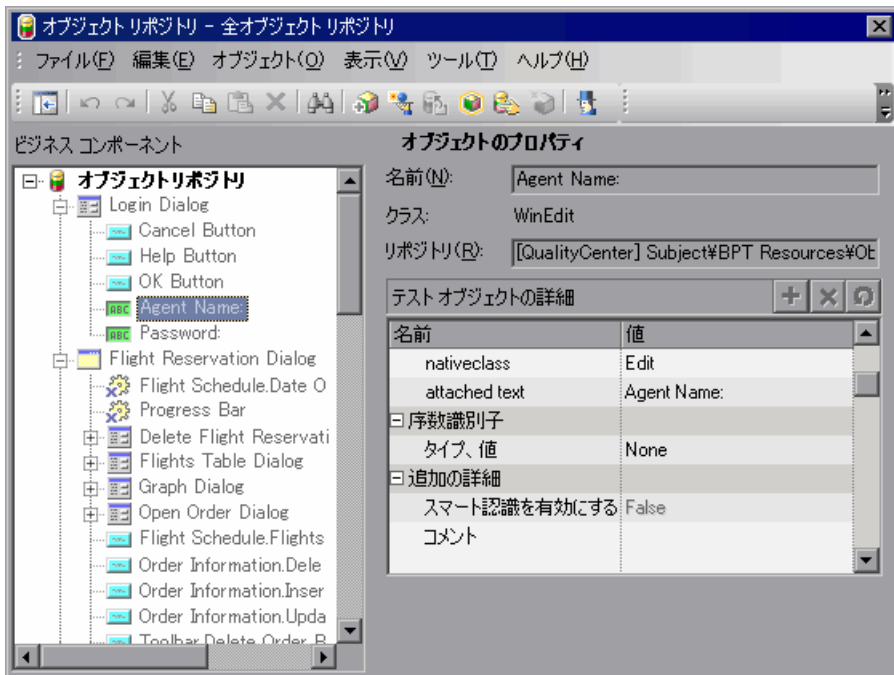
また、ローカル・オブジェクトを、同じコンポーネントに関連付けられている共有オブジェクト・リポジトリに直接結合することもできます。これにより、オブジェクトを複数の場所ではなく1つの共有された場所で管理できるため、管理作業が軽減されます。詳細については、443 ページ「ローカル・オブジェクト・リポジトリからの共有オブジェクト・リポジトリの更新」を参照してください。

[オブジェクト リポジトリ] ウィンドウについて



特定のコンポーネントの [オブジェクト リポジトリ] ウィンドウを開くには、[リソース] > [オブジェクト リポジトリ] を選択するか、[オブジェクト リポジトリ] ボタンをクリックします。

[オブジェクト リポジトリ] ウィンドウには、現在のコンポーネントに含まれるすべてのオブジェクト（すべてのローカル・オブジェクトおよび選択したコンポーネントに関連付けられている共有オブジェクト・リポジトリ内のすべてのオブジェクト）のツリーが表示されます。



[オブジェクト リポジトリ] ウィンドウには、ツリー内で選択したテスト・オブジェクトごとに、テスト・オブジェクトに関する情報（テスト・オブジェクトのタイプ、テスト・オブジェクトが格納されるリポジトリ、およびテスト・オブジェクトの詳細）が表示されます。ローカル・オブジェクトは編集可能（黒色）ですが、共有オブジェクトは読み取り専用形式（灰色）です。

[オブジェクト・リポジトリ] ウィンドウが開いている間、QuickTest を引き続き使用することができます。また、テスト・オブジェクトおよびオブジェクト・リポジトリを引き続き変更することができます。また、必要に応じて、[オブジェクト・リポジトリ] ウィンドウのサイズを変更することもできます。[オブジェクトリポジトリ] ウィンドウには、関連付けられているオブジェクト・リポジトリに加えたすべての変更がリアルタイムで反映されます。たとえば、ローカル・オブジェクト・リポジトリにオブジェクトを追加した場合、または追加のオブジェクト・リポジトリを現在のコンポーネントに関連付けた場合、[オブジェクトリポジトリ] ウィンドウには更新された内容が即座に表示されます。

注：オブジェクト・リポジトリ・ツリーのみを表示するか、オブジェクト・リポジトリ・ツリーとともにテスト・オブジェクトの詳細領域も表示するかを選択できます。詳細については、313 ページ「テスト・オブジェクトの詳細領域の表示と非表示」を参照してください。

[オブジェクトリポジトリ] ウィンドウでは、リポジトリ（ローカルおよび共有オブジェクト・リポジトリ）内の任意のテスト・オブジェクトのテスト・オブジェクト記述を表示したり、ローカル・テスト・オブジェクトとそのプロパティを変更したり、ローカル・オブジェクト・リポジトリにオブジェクトを追加したりできます。



注：ローカル・オブジェクトに対して加えたすべての変更は、変更後直ちに、そのローカル・オブジェクトを使用するすべてのステップに自動的に反映されます。[編集] > [元に戻す] および [編集] > [やり直し] オプション、または [元に戻す] および [やり直し] ボタンを使用して、変更の取り消しや繰り返しができます。現在のコンポーネントを保存した後で、保存操作の前に実行した操作を元に戻したりやり直したりすることはできません。

テスト・オブジェクトのプロパティの表示と変更の詳細については、319 ページ「テスト・オブジェクトのプロパティの変更」を参照してください。

注：テスト・オブジェクトが含まれるステップをコンポーネントから削除しても、オブジェクトはオブジェクト・リポジトリから削除されません。ローカル・オブジェクト・リポジトリ内のオブジェクトは、[オブジェクトリポジトリ] ウィンドウを使用して削除できます。共有オブジェクト・リポジトリ内のオブジェクトは、オブジェクト・リポジトリ・マネージャを使用して削除できます。詳細については、395 ページ「オブジェクト・リポジトリの管理」を参照してください。

[オブジェクトリポジトリ] ウィンドウには、次の情報が表示されます。

情報	詳細
[ビジネス・コンポーネント]	現在のテスト文書がビジネス・コンポーネントであることが表示されます。
[オブジェクトリポジトリ] ツリー	<p>現在のコンポーネントに含まれるすべてのオブジェクト（すべてのローカル・オブジェクトおよび選択したコンポーネントに関連付けられている共有オブジェクト・リポジトリ内のすべてのオブジェクト）が表示されます。</p> <p>注：同じ名前、オブジェクト・クラス、および親階層を持つテスト・オブジェクトが、関連付けられている複数のオブジェクト・リポジトリ内に存在する場合は、定義されている優先順位に従って最初に見つかったオブジェクトだけがオブジェクト・リポジトリ・ツリーに表示されます。オブジェクト・リポジトリの優先順位の詳細については、94 ページ「共有オブジェクト・リポジトリの管理」を参照してください。</p> <p>オブジェクト・リポジトリ・ツリーでは、フィルタを設定して、表示するオブジェクトを絞り込むことができます。詳細については、313 ページ「[オブジェクトリポジトリ] ウィンドウのフィルタ処理」を参照してください。</p>

情報	詳細
【名前】	QuickTest によってテスト・オブジェクトに割り当てられる名前です。ローカル・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は変更できます。詳細については、327 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。
【クラス】	オブジェクトのクラスです。
【リポジトリ】	オブジェクトが格納されているオブジェクト・リポジトリの場所（ファイル名とパス）です。オブジェクトがローカル・オブジェクト・リポジトリに格納されている場合は、 【Local】 と表示されます。
【テストオブジェクトの詳細】	実行セッション中にオブジェクトの識別に使用されるプロパティとプロパティ値が表示されます。また、ローカル・オブジェクト・リポジトリ内のテスト・オブジェクトについては、テスト・オブジェクトの詳細を変更できます。詳細については、309 ページ「[テストオブジェクトの詳細] 領域について」を参照してください。テスト・オブジェクトの詳細領域は、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、313 ページ「テスト・オブジェクトの詳細領域の表示と非表示」を参照してください。

【テストオブジェクトの詳細】領域について

[オブジェクトリポジトリ] ウィンドウの **【テストオブジェクトの詳細】** 領域では、実行セッション中にオブジェクトの識別に使用されるプロパティとプロパティ値を表示および変更できます。

ヒント：テスト・オブジェクトの詳細領域は、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、313 ページ「テスト・オブジェクトの詳細領域の表示と非表示」を参照してください。

[オブジェクトリポジトリ] ウィンドウでは、共有オブジェクト・リポジトリ内のオブジェクトがオブジェクト・プロパティ表示枠 ([**テストオブジェクトの詳細**] 領域を含む) に読み取り専用形式で表示されます。共有オブジェクト・リポジトリ内のオブジェクトを変更するには、その共有オブジェクト・リポジトリをオブジェクト・リポジトリ・マネージャで開きます。詳細については、第15章「オブジェクト・リポジトリの管理」を参照してください。また、共有オブジェクト・リポジトリ内のオブジェクトをローカル・オブジェクト・リポジトリにコピーし、そのローカル・コピーを変更することもできます。詳細については、315 ページ「ローカル・オブジェクト・リポジトリへのオブジェクトのコピー」を参照してください。

ヒント：

オブジェクトのプロパティおよびプロパティ値を表示するには、[オブジェクトのプロパティ] ダイアログ・ボックスを使用します。詳細については、317 ページ「オブジェクトのプロパティおよびプロパティ値の表示」を参照してください。



オブジェクト・スパイを使用して、テスト対象アプリケーション内にあるオブジェクトについて、実行時プロパティまたはテスト・オブジェクト・プロパティとその値をいつでも表示できます。オブジェクト・スパイを開くには、[ツール] > [オブジェクト スパイ] を選択するか、[オブジェクト スパイ] ツールバー・ボタンをクリックします。詳細については、290 ページ「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の表示」を参照してください。

ローカル・オブジェクト・リポジトリに保存したオブジェクトについては、テスト・オブジェクトの詳細を変更できます。また、共有オブジェクト・リポジトリからローカル・オブジェクト・リポジトリにオブジェクトをコピーし、そのローカル・コピーを変更することもできます。

名前	値
記述プロパティ	
image type	Image Button
html tag	INPUT
alt	Sign-In
序数識別子	
タイプ、値	なし
追加の詳細	
スマート認識を有効にする	True
コメント	



注： ローカル・オブジェクトに対して加えたすべての変更は、変更後直ちに、そのローカル・オブジェクトを使用するすべてのステップに自動的に反映されます。**[編集]** > **[元に戻す]** および **[編集]** > **[やり直し]** オプション、または **[元に戻す]** および **[やり直し]** ボタンを使用して、変更の取り消しや繰り返しができます。現在のコンポーネントを保存した後で、保存操作の前に実行した操作を元に戻したりやり直したりすることはできません。

「**テストオブジェクトの詳細**」領域には、次の項目が表示されます。

項目	詳細
<p>【記述プロパティ】</p>	<p>実行セッション中にオブジェクトの識別に使用されるプロパティおよびプロパティ値。 テスト・オブジェクト記述に含まれるプロパティは追加および削除できます。詳細については、329 ページ「テスト・オブジェクト記述へのプロパティの追加」を参照してください。 プロパティ値は、定数として指定することも、パラメータ化することもできます。詳細については、321 ページ「プロパティ値の指定または変更」を参照してください。</p>
<p>【序数識別子】</p>	<p>記述がまったく同じであるほかのオブジェクト（すべてのプロパティが同じ値のオブジェクト）に対する該当オブジェクトの相対的な順序または場所を示す数値。詳細については、336 ページ「序数識別子の指定」を参照してください。</p>
<p>【追加の詳細】</p>	<p>次のオプションが含まれています。</p> <ul style="list-style-type: none"> • 【スマート認識を有効にする】：テスト・オブジェクト記述では識別できないテスト・オブジェクトを認識するために、実行セッション中にスマート認識を使用するかどうかを、[True] または [False] で指定できます。 <p>注：このオプションは、【オブジェクトの識別】 ダイアログ・ボックスでテスト・オブジェクトのクラスにスマート認識プロパティが定義されている場合のみ使用できます。スマート認識の詳細については、382 ページ「スマート認識の設定」を参照してください。</p> <ul style="list-style-type: none"> • 【コメント】：テスト・オブジェクトに関するテキスト情報を追加できます。

テスト・オブジェクトの詳細領域の表示と非表示

[オブジェクトリポジトリ] ウィンドウでは、簡易表示モードと完全表示モードのいずれかを選択できます。簡易表示モードではオブジェクト・リポジトリ・ツリーだけが表示されますが、完全表示モードではオブジェクト・リポジトリ・ツリーとともにテスト・オブジェクトの詳細領域も表示されます。

[オブジェクトリポジトリ] ウィンドウの表示モードを変更するには、次の手順を実行します。

表示するモードに応じて、次のいずれかを実行します。



- ▶ [表示] > [簡略表示] を選択するか、[簡略表示] ボタンをクリックします。
- ▶ [表示] > [全体表示] を選択するか、[全体表示] ボタンをクリックします。

[オブジェクトリポジトリ] ウィンドウが、選択した表示モードに切り替わります。

[オブジェクトリポジトリ] ウィンドウのフィルタ処理

[フィルタ] ツールバーを使って、[オブジェクトリポジトリ] ウィンドウにフィルタを設定して表示するオブジェクトを絞り込むことができます。



次のいずれかの条件を指定すると、条件を満たすオブジェクトが表示されます。

- ▶ 現在のコンポーネントに含まれるすべてのオブジェクト
- ▶ 現在のコンポーネントに含まれるローカル・オブジェクトのみ
- ▶ 現在のコンポーネントに関連付けられている特定の共有オブジェクト・リポジトリ内のオブジェクトのみ

[オブジェクトリポジトリ] ウィンドウにフィルタを設定するには、次の手順を実行します。

[フィルタ] ツールバー・リストで、次のいずれかのオプションを選択します。

- ▶ [全オブジェクト]
- ▶ [ローカルオブジェクト]
- ▶ 現在のコンポーネントに関連付けられている特定の共有オブジェクト・リポジトリの名前

選択した場所のオブジェクトだけを表示するようにオブジェクト・リポジトリ・ツリーにフィルタが設定されます。[オブジェクトリポジトリ] ウィンドウのタイトル・バーには、現在のフィルタが示されます。

テスト・オブジェクトのプロパティの表示と変更

Web サイトまたはアプリケーションに加えられた変更に応じて、コンポーネントのステップのプロパティ値に変更が必要になることがあります。アプリケーションに含まれるオブジェクトに変更が加えられた場合、そのオブジェクトがコンポーネントの一部であれば、引き続き QuickTest での識別が可能なように値を変更する必要があります。たとえば、会社の Web サイトに含まれる「**Contact Us**」というハイパーテキスト・リンクのテキスト文字列が「**Contact My Company**」に変更された場合、このリンク・プロパティが引き続き QuickTest で正しく認識されるようにするために、オブジェクト・リポジトリに含まれるこのオブジェクトの詳細を更新する必要があります。

テスト・オブジェクトのプロパティは、さまざまな方法で表示および変更できます。ローカル・オブジェクト・リポジトリに格納されているオブジェクトの場合、そのプロパティは [オブジェクトリポジトリ] ウィンドウから直接変更できます。共有オブジェクト・リポジトリに格納されているオブジェクトの場合、オブジェクト・リポジトリ・マネージャでそのオブジェクトを開いてプロパティを変更できます。また、そのオブジェクトをローカル・オブジェクト・リポジトリにコピーしてからプロパティを変更することもできます。

テスト・オブジェクトのプロパティを表示および変更する各種の方法の詳細については、次を参照してください。

- ▶ 315 ページ「ローカル・オブジェクト・リポジトリへのオブジェクトのコピー」
- ▶ 317 ページ「オブジェクトのプロパティおよびプロパティ値の表示」
- ▶ 319 ページ「テスト・オブジェクトのプロパティの変更」
- ▶ 321 ページ「プロパティ値の指定または変更」
- ▶ 324 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」
- ▶ 326 ページ「テスト・オブジェクトの標準設定のプロパティの復元」
- ▶ 327 ページ「テスト・オブジェクトの名前の変更」
- ▶ 329 ページ「テスト・オブジェクト記述へのプロパティの追加」
- ▶ 333 ページ「新規テスト・オブジェクト・プロパティの定義」
- ▶ 335 ページ「テスト・オブジェクト記述からのプロパティの削除」
- ▶ 336 ページ「序数識別子の指定」

ローカル・オブジェクト・リポジトリへのオブジェクトのコピー

共有オブジェクト・リポジトリに格納されているオブジェクトを変更するには、オブジェクト・リポジトリ・マネージャを使用して変更します。また、[オブジェクトリポジトリ] ウィンドウを使用してローカルに変更することも可能です。

オブジェクト・リポジトリ・マネージャを使用して変更すると、変更内容はその共有オブジェクト・リポジトリを使用するすべてのコンポーネントに反映されます。オブジェクトのローカル・コピーを作成し、それを [オブジェクトリポジトリ] ウィンドウで変更すると、変更内容は変更を行ったコンポーネントにのみ反映されます。後で共有オブジェクト・リポジトリ内の同じオブジェクトを変更しても、変更内容はコンポーネント内のオブジェクトのローカル・コピーに反映されません。

オブジェクトをローカル・オブジェクト・リポジトリにコピーする場合は、次のことを考慮してください。

- ▶ オブジェクトをローカル・オブジェクト・リポジトリにコピーすると、その親オブジェクトもローカル・オブジェクト・リポジトリにコピーされます。
- ▶ オブジェクトまたはその親オブジェクトの中で未割り当てのリポジトリ・パラメータが使用されている場合、当該オブジェクトはローカル・オブジェクト・リポジトリにコピーできません。オブジェクトをローカル・オブジェクト・リポジトリにコピーする前に、すべてのリポジトリ・パラメータが割り当て済みであることを確認する必要があります。
- ▶ オブジェクトまたはその親オブジェクトが1つ以上のリポジトリ・パラメータを使ってパラメータ化されている場合、リポジトリ・パラメータの値はオブジェクトをローカル・オブジェクト・リポジトリにコピーしたときに変換されます。たとえば、リポジトリ・パラメータがローカル・パラメータに割り当てられている場合、プロパティはローカルパラメータを使用してパラメータ化されます。値が定数値である場合、プロパティには同じ定数値が設定されます。
- ▶ 複数のオブジェクトをローカル・オブジェクト・リポジトリにコピーするとき、未割り当てのリポジトリ・パラメータを含むオブジェクトがある場合や、割り当て済みのリポジトリ・パラメータの値を変換したくない場合は、コピー処理中に当該オブジェクトをスキップできます。その後、元の選択範囲の次のオブジェクトから引き続きコピーを続行できます。

オブジェクトをローカル・オブジェクト・リポジトリにコピーするには、次の手順を実行します。

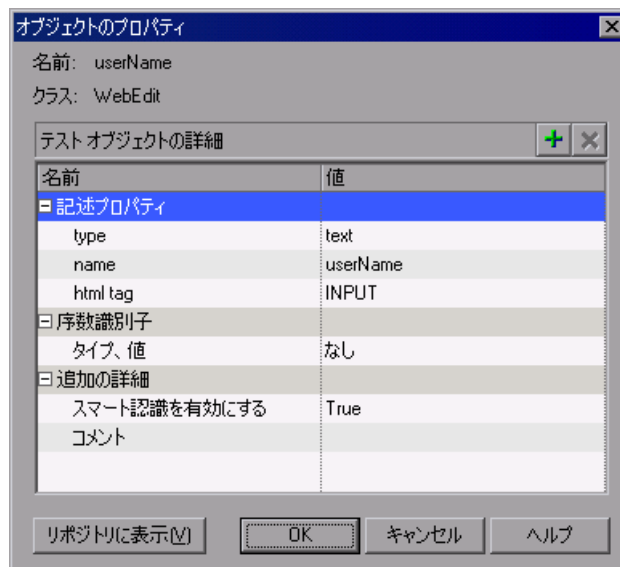
- 1 [オブジェクトリポジトリ] ウィンドウで、ローカル・オブジェクト・リポジトリにコピーするオブジェクトを共有オブジェクト・リポジトリから選択します。共有オブジェクト・リポジトリ内のオブジェクトは灰色で表示されています。同じ親オブジェクトを持つオブジェクトであれば、複数のオブジェクトをコピー対象として選択できます。
- 2 [オブジェクト] > [ローカルにコピー] を選択するか、オブジェクト（複数も可）を右クリックして [ローカルにコピー] を選択します。選択したオブジェクト（およびその親オブジェクト）がローカル・オブジェクト・リポジトリにコピーされ、編集可能になります。

オブジェクトのプロパティおよびプロパティ値の表示

コンポーネントのステップに含まれるオブジェクトについて、テスト・オブジェクトのプロパティおよびプロパティ値を表示できます。

オブジェクトのプロパティおよびプロパティ値を表示するには、次の手順を実行します。

コンポーネント内で、プロパティを表示する対象となるオブジェクトのステップをクリックし、[編集] > [ステップのプロパティ] > [オブジェクトのプロパティ] を選択します。[オブジェクトのプロパティ] ダイアログ・ボックスが開きます。



注：[オブジェクトプロパティ] ダイアログ・ボックスは、選択したオブジェクトがローカル・オブジェクト・リポジトリに現在格納されているのか、共有オブジェクト・リポジトリに現在格納されているのか、あるいは現在のコンポーネントに関連付けられているどのオブジェクト・リポジトリにも格納されていないかどうかに応じて、若干異なります。本項では、ダイアログ・ボックスに表示されるオプションのうち、ローカル・オブジェクト・リポジトリ内のオプション、または関連付けられているどのオブジェクト・リポジトリ内にもないオプションについて説明します。共有オブジェクト・リポジトリ内に格納されているオブジェクトの場合、このダイアログ・ボックスは（上記のように）ローカル・オブジェクトに対するダイアログ・ボックスと同じように表示されますが、読み取り専用形式で表示されます。

[オブジェクトのプロパティ] ダイアログ・ボックスでは、選択したオブジェクトの名前とクラスが表示され、次の操作を実行できます。

- ▶ オブジェクトのプロパティとプロパティ値（記述プロパティ、序数識別子、およびその他の設定値）の表示。
- ▶ オブジェクトの識別に使用されるプロパティとプロパティ値の変更（ローカル・オブジェクト・リポジトリに格納されているオブジェクトの場合）。[オブジェクトのプロパティ] ダイアログ・ボックスでプロパティと値を変更する方法は、[オブジェクトリポジトリ] ウィンドウでテスト・オブジェクトの詳細を変更する方法と同じです。詳細については、319 ページ「テスト・オブジェクトのプロパティの変更」を参照してください。
- ▶ **[リポジトリに表示]** ボタンをクリックして [オブジェクトリポジトリ] ウィンドウを開き、選択したオブジェクトをオブジェクト階層内に表示します（オブジェクト・リポジトリに格納されているオブジェクトの場合）。
- ▶ **[記述プロパティを追加]** ボタンをクリックして、選択したオブジェクトをローカル・オブジェクト・リポジトリに追加します（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の場合）。

テスト・オブジェクトのプロパティの変更

オブジェクトを変更するには、そのオブジェクトの1つ以上のプロパティ値を変更するか、そのオブジェクトの識別に使用されるプロパティ・セットを変更します。そのためには、ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は [オブジェクトリポジトリ] ウィンドウを使用し、共有オブジェクト・リポジトリ内のオブジェクトの場合はオブジェクト・リポジトリ・マネージャを使用します。

オブジェクト・リポジトリに含まれる1つ以上のテスト・オブジェクトの記述を、アプリケーション内で実際に更新されたオブジェクト・プロパティに基づいて自動的に更新することもできます。詳細については、324 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。

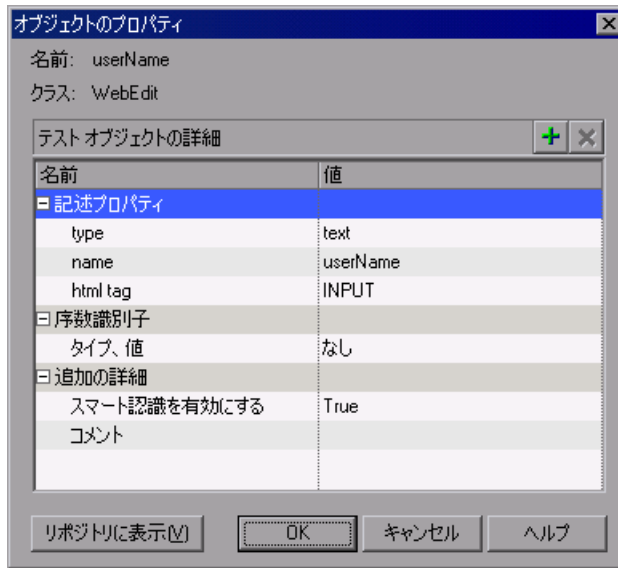


ヒント：オブジェクト・スパイを使用して、テスト対象アプリケーション内にあるオブジェクトについて、実行時プロパティまたはテスト・オブジェクト・プロパティとその値をいつでも表示できます。オブジェクト・スパイを開くには、[ツール] > [オブジェクトスパイ] を選択するか、[オブジェクトスパイ] ツールバー・ボタンをクリックします。詳細については、290 ページ「オブジェクト・スパイを使用したオブジェクトのプロパティの表示」を参照してください。

オブジェクトのプロパティを変更するには、次の手順を実行します。

- 1 変更されたオブジェクトが含まれているステップを右クリックして [オブジェクトのプロパティ] を選択するか、メニュー・バーから [編集] > [ステップのプロパティ] > [オブジェクトのプロパティ] を選択します。

[オブジェクトのプロパティ] ダイアログ・ボックスが開き、QuickTest によってオブジェクトの識別に使用されるプロパティが表示されます。



ヒント：

コンポーネントに含まれるすべてのオブジェクトを表示するには、[リポジトリに表示] ボタンをクリックします。[オブジェクトリポジトリ] ウィンドウが開き、リポジトリに格納されているすべてのオブジェクトがリポジトリ・ツリーに表示されます。



選択したコンポーネントのオブジェクト・リポジトリを開くには、[リソース] > [オブジェクトリポジトリ] を選択する方法や、[オブジェクトリポジトリ] ツールバー・ボタンをクリックする方法もあります。

- 必要に応じて、プロパティと値を変更します。[オブジェクトのプロパティ] ダイアログ・ボックスでプロパティと値を変更する方法は、[オブジェクトリポジトリ] ウィンドウでテスト・オブジェクトの詳細を変更する方法と同じです。詳細については、309 ページ「[テスト オブジェクトの詳細] 領域について」および 314 ページ「テスト・オブジェクトのプロパティの表示と変更」を参照してください。
- [OK] をクリックし、ダイアログ・ボックスを閉じます。

プロパティ値の指定または変更

テスト・オブジェクト記述に含まれるプロパティの値を指定または変更できません。定数値（単純な値か、正規表現を含んだ定数値）を指定することも、値をパラメータ化することもできます。そのためには、ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は [オブジェクトリポジトリ] ウィンドウまたは [オブジェクトのプロパティ] ダイアログ・ボックスを使用し、共有オブジェクト・リポジトリ内のオブジェクトの場合はオブジェクト・リポジトリ・マネージャを使用します。

また、特定のオブジェクト・プロパティ値を検索して置換することもできます。詳細については、355 ページ「オブジェクト・リポジトリ内のオブジェクトの検索」を参照してください。

注：場合によっては、オブジェクトのプロパティ値の一部が変更されても、スマート認識機能を使用することで QuickTest がオブジェクトを識別できることがあります。しかし、特定のオブジェクトのプロパティ値に変更があったことを知っているのならば、QuickTest が基本のオブジェクト記述からオブジェクトを識別できるように、オブジェクト定義を修正する必要があります。スマート認識メカニズムの詳細については、第 14 章「オブジェクトの認識の設定」を参照してください。







ヒント：オブジェクト・スパイを使用して、テスト対象アプリケーション内にあるオブジェクトについて、実行時プロパティまたはテスト・オブジェクト・プロパティとその値をいつでも表示できます。オブジェクト・スパイを開くには、[ツール] > [オブジェクト スパイ] を選択するか、[オブジェクト スパイ] ツールバー・ボタンをクリックします。詳細については、290 ページ「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の表示」を参照してください。

プロパティ値を指定するには、次の手順を実行します。

- 1 プロパティ値を指定する対象となるテスト・オブジェクトを選択します。
- 2 **[テストオブジェクトの詳細]** 領域で、必要なプロパティの値セルをクリックします。

ヒント：ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は、必要なテスト・オブジェクトを選択し、**[編集]** > **[ステップのプロパティ]** > **[オブジェクトのプロパティ]** を選択して、**[オブジェクトのプロパティ]** ダイアログ・ボックスで次のようにプロパティ値を変更します。

- 3 次のいずれかの方法でプロパティ値を指定します。
 - ▶ 定数値を指定する場合は、値セルに定数値を入力します。
 - ▶ 正規表現を使った定数値を指定する場合や値をパラメータ化する場合は、値セル内のパラメータ化ボタンをクリックします。正規表現を使用して定数値を指定した場合、アイコンが値の隣に表示されます。値のパラメータ化の詳細については、172 ページ「パラメータの使用」を参照してください。
- 4 定数値を指定した場合は、その定数値が **[テストオブジェクトの詳細]** 領域の **[値]** カラムに表示されます。値をパラメータ化した場合は、パラメータ名が次のいずれかのアイコンとともに **[値]** カラムに表示されます。

パラメータ・アイコン	詳細
	プロパティの値が現在はコンポーネント・パラメータであることを示します。
	プロパティの値が現在はローカル・パラメータであることを示します。
	プロパティの値が現在は（共有オブジェクト・リポジトリ内で）リポジトリ・パラメータであることを示します。

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


オブジェクト・リポジトリ内のオブジェクトを更新するには、アプリケーション内の対応するオブジェクトを選択し、そのオブジェクトのプロパティおよびプロパティ値をアプリケーションから学習し直します。このようにしてテスト・オブジェクトの記述を更新すると、記述のプロパティと値、序数識別子、スマート認識情報など、現在定義されているすべてのプロパティと値が上書きされます。すでに入力されているオブジェクト固有のコメントは削除されません。

これは、オブジェクトをオブジェクト・リポジトリに追加した後でオブジェクトのプロパティに変更があった場合に便利です。QuickTest では、変更があった場合、オブジェクトの記述を更新しない限りオブジェクトが認識されないためです。

また、アプリケーションの開発が完了する前に（[オブジェクト] > [新規テスト オブジェクトの定義] で）オブジェクトを定義した結果、そのオブジェクトのプロパティや値の一部がテスト・オブジェクトの記述に見つからなかったり、オブジェクトを識別するのに十分でなくなったりした場合は、このオプションを使ってそのようなオブジェクトを更新することもできます。[新規テスト オブジェクトの定義] オプションの詳細については、349 ページ「新規テスト・オブジェクトの定義」を参照してください。

更新は、ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は [オブジェクト リポジトリ] ウィンドウを使用して行い、共有オブジェクト・リポジトリ内のオブジェクトの場合はオブジェクト・リポジトリ・マネージャを使用して行います。

アプリケーション内のオブジェクトからテスト・オブジェクトのプロパティを更新するには、次の手順を実行します。

- 1 オブジェクト・リポジトリ・ツリーの中で、記述を更新する対象となるテスト・オブジェクトを選択します。
- 2  [オブジェクト] > [アプリケーションから更新] を選択するか、[アプリケーションから更新] ボタンをクリックします。QuickTest が最小化され、カーソルが指差し型に変わり、起動しているアプリケーションの任意のオブジェクトをポイントしてクリックできるようになります。
- 3 プロパティを更新する対象となるオブジェクト・リポジトリ内のオブジェクトをアプリケーションの中で探して、それをクリックします。オブジェクト・リポジトリ・ツリーの中で選択したテスト・オブジェクトと同じオブジェクト・クラスのオブジェクトを選択する必要があります。

注：

選択対象オブジェクトを含んでいるウィンドウが部分的に別のウィンドウに隠れている場合は、部分的に隠れているウィンドウに指差しマークを数秒間置いたままにします。ウィンドウが一番手前に表示されます。これで、必要なオブジェクトをポイントし、クリックできるようになります。ウィンドウを一番手前に表示するのに必要な時間の設定は、[オプション] ダイアログ・ボックスの [一般] タブで行えます。詳細については、第8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。また、左側の CTRL キーを押しながら、ウィンドウのフォーカスを変更することもできます。また、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、左側の CTRL キーを押したまま 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。

特定のイベント（ショートカット・メニューを表示するための右クリックやマウスオーバーなど）が生じないと選択対象オブジェクトが表示されない場合は、左側の CTRL キーを押し下げます。すると、指差しマークが一時的に標準の矢印に変わり、イベントを実行できるようになります。調査対象のオブジェクトが表示されたら、左側の CTRL キーを放します。矢印が再び指差しマークに変わります。


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] ダイアログ・ボックスが開きます。オブジェクト・ツリーから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のプロパティとプロパティ値が、オブジェクトを識別するのに必要なプロパティとプロパティ値に従ってオブジェクト・リポジトリ内で更新されます。これらのプロパティとプロパティ値は、アプリケーション内のオブジェクトをクリックしたときに QuickTest によって学習されます。[テストオブジェクトの詳細] 領域に含まれるすべてのプロパティおよびプロパティ値が、序数識別子やスマート認識の設定とともに更新されます。すでに入力されているオブジェクト固有のコメントは削除されません。

テスト・オブジェクトの標準設定のプロパティの復元

選択したテスト・オブジェクトの標準設定のプロパティを復元できます。標準設定のプロパティを復元すると、[オブジェクトの認識] ダイアログ・ボックスで選択したオブジェクト・クラスに対して定義された必須のプロパティ・セットが復元されます。対象テスト・オブジェクトの記述プロパティ・セットに対して行われた変更は、すべて上書きされます。ただし、必須プロパティのいずれかに対してプロパティ値が定義されている場合、それらは変更されません。また、標準設定の必須のプロパティ・セットを復元しても、テスト・オブジェクトの序数識別子やスマート認識の設定値は変更されません。

必須のプロパティ・セットを復元するには、次の手順を実行します。

- 1 オブジェクト・リポジトリ・ツリーの中で、記述を復元する対象となるテスト・オブジェクトを選択します。
- 2  [テストオブジェクトの詳細] 領域で、[必須プロパティ セットを復元する] ボタンをクリックします。
- 3 [はい] をクリックして操作を確定します。テスト・オブジェクトの記述プロパティとして、選択したオブジェクト・クラスの必須のプロパティ・セットが復元されます。

テスト・オブジェクトの名前の変更

アプリケーション内のオブジェクトに変更が加えられた場合や、何らかの理由で現在のテスト・オブジェクト名に不都合がある場合は、格納されているオブジェクトに割り当てられている名前を変更できます。また、ユーザがコンポーネント・ステップの中でオブジェクトを使用するときにオブジェクトを識別しやすいように、オブジェクトにわかりやすい名前を付けることもできます。

たとえば、グラフィック・アプリケーションのツールバーのすべてのツールが WinObjects としてオブジェクト・リポジトリに保存され、それぞれオブジェクトに ToolChild1, ToolChild2, ToolChild3 という具合に名前が付けられているとします。この場合、すべてのボタンの名前を実際のラベル (Color_Picker, Erasor, Airbrush など) に変更すると、ボタンが識別しやすくなります。

ローカル・オブジェクト・リポジトリ内のオブジェクトの名前は、[オブジェクトリポジトリ] ウィンドウを使用して変更します。共有オブジェクト・リポジトリ内のオブジェクトの名前は、オブジェクト・リポジトリ・マネージャを使用して変更します。

共有オブジェクト・リポジトリを使用している場合は、当該共有オブジェクト・リポジトリを使用するすべてのコンポーネントの中で、当該オブジェクトが出現するすべての箇所に変更が適用されます。

ローカル・オブジェクト・リポジトリを使用している場合は、選択したコンポーネントの中で、当該オブジェクトが出現するすべての箇所に変更が適用されます。当該ローカル・オブジェクトに対する操作が、同じビジネス・プロセス・テスト内のほかのコンポーネントにも含まれている場合は、該当する各コンポーネントの中で、当該オブジェクトの名前を変更する必要があります。

ローカル・オブジェクト・リポジトリ内のオブジェクトの名前を変更すると、そのオブジェクトのすべての出現箇所の名前が自動的に更新されます。共有リポジトリ内のオブジェクトの名前を変更すると、当該オブジェクト・リポジトリの変更をまだ保存していなくても、同じコンピュータ上で開いている、当該オブジェクト・リポジトリを使用しているすべてのコンポーネントの中の名前が、変更の直後に自動的に更新されます。変更を保存せずにオブジェクト・リポジトリを閉じると、変更時に開いていたすべてのコンポーネントにおいて、反映されていた変更内容がロールバックされます。保存した変更は、当該オブジェクト・リポジトリを使用するコンポーネントにも、それらを開いた直後に自動的に反映されます。現在別のコンピュータで開いているコンポーネントまたはリポジトリ・ファイルに、保存した変更内容を反映するには、変更内容を読み込むために、コンピュータ上でオブジェクト・リポジトリを開くか編集用にロックする必要があります。

ヒント：すべての出現箇所にテスト・オブジェクト名が自動的に反映されないようにするには、[オプション] ダイアログ・ボックス ([ツール] > [オプション]) の [一般] タブにある **[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]** チェック・ボックスをクリアします。このオプションをクリアした場合は、該当するテスト・オブジェクトが使用されているすべてのステップの中で当該テスト・オブジェクトの名前を手動で変更する必要があります。変更しないと、コンポーネントの実行は失敗します。

注：共有オブジェクト・リポジトリ内のオブジェクト名を変更して変更を保存した後で、同じ共有オブジェクト・リポジトリを使用する別のコンポーネントを開くと、そのコンポーネント内の該当するすべてのステップにオブジェクト名の変更が反映されます。この処理にはしばらくかかる場合があります。後から開いたコンポーネントに対する変更の内容の保存を実行すると、名前の変更が反映されたステップが保存されますが、このコンポーネントを保存せずに閉じた場合は、同じコンポーネントを次回開くときに、ステップに含まれるオブジェクト名を更新するのに再び時間がかかります。

テスト・オブジェクトの名前を変更するには、次の手順を実行します。

オブジェクト・リポジトリ・ツリーの中で、名前を変更する対象となるテスト・オブジェクトを選択し、次のいずれかを実行します。

- ▶ **[編集]** > **[名前の変更]** を選択し、ツリー内の選択したノードにあるオブジェクトに与える新しい名前を入力します。次に、ENTER キーを押すか、ほかの場所をクリックしてフォーカスをオブジェクトから別の場所に移動します。
- ▶ F2 キーを押し、オブジェクトの新しい名前を入力します。
- ▶ オブジェクト・プロパティ表示枠の **[名前]** ボックスに、オブジェクトの新しい名前を入力します。次に、ほかの任意の場所をクリックしてフォーカスをオブジェクトから別の場所に移動します。

注：オブジェクトに割り当てる名前は、該当するオブジェクト・リポジトリ内において一意である必要があります。オブジェクト名では、大文字と小文字は区別されません。

テスト・オブジェクト記述へのプロパティの追加

QuickTest でオブジェクトの識別に使用されるプロパティのリストにプロパティを追加できます。QuickTest では、オブジェクト・クラスごとに、オブジェクトの記述に使用する標準のプロパティ・セットがあります。[プロパティの追加] ダイアログ・ボックスを使用して、オブジェクト記述に含まれるプロパティを変更できます。そのためには、ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は [オブジェクトリポジトリ] ウィンドウまたは [オブジェクトのプロパティ] ダイアログ・ボックスを使用し、共有オブジェクト・リポジトリ内のオブジェクトの場合はオブジェクト・リポジトリ・マネージャを使用します。

注：また、[プロパティの追加] ダイアログ・ボックスに表示されないテスト・オブジェクト・プロパティであっても、有効なプロパティならばテスト・オブジェクト記述に追加できます。詳細については、333 ページ「新規テスト・オブジェクト・プロパティの定義」を参照してください。

プロパティのリストへの追加は、動的に変化するオブジェクトを対象にコンポーネントを作成し、実行する場合に便利です。頻繁に更新される場合、または動的コンテンツ（データベースなど）によってプロパティ値が設定される場合に、オブジェクトが動的に変化することがあります。

また、記録中に自動的に取得されなかったプロパティを使ってオブジェクトを参照する場合は、オブジェクトを識別するプロパティを変更することもできます。たとえば、ニュースレターのアーカイブが含まれている Web サイトをテストするとします。アーカイブ・ページには、最新のニュースレターへのハイパーテキスト・リンクに加えて、過去のすべてのニュースレターへのハイパーテキスト・リンクがあります。このページの最初のハイパーテキスト・リンクのテキストは、最新のニュースレターの変更に伴って変化しますが、常に **current.html** というページにリンクされています。コンポーネントに、アーカイブ・ページの最初のハイパーテキスト・リンクを必ずクリックするステップを作成するとします。ニュースは常に変化しているので、このハイパーテキスト・リンクのテキストもそれに応じて変化していきます。そのため、QuickTest が常にこのリンクを見つけられるように、このハイパーテキスト・リンクを識別する方法を変更する必要があります。

Link オブジェクト（ハイパーテキスト・リンク）の標準設定のプロパティは、「text」と「html tag」です。text プロパティは、リンク内のテキストです。HTML tag プロパティは常にリンクを表す「A」です。


リンクのテキストではなくリンク先ページで識別できるように、記録したオブジェクトのハイパーテキスト・リンクの標準設定のプロパティを変更します。「href」プロパティを使用すれば、リンク先ページに基づいてリンクを検査でき、「text」プロパティでリンクのテキストに基づいて検査を行う必要はありません。



ヒント：オブジェクト・スパイを使用して、テスト対象アプリケーション内にあるオブジェクトについて、実行時プロパティまたはテスト・オブジェクト・プロパティとその値をいつでも表示できます。オブジェクト・スパイを開くには、[ツール] > [オブジェクトスパイ] を選択するか、[オブジェクトスパイ] ツールバー・ボタンをクリックします。詳細については、290 ページ「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の表示」を参照してください。

注：また，[オブジェクトの認識] ダイアログ・ボックスを使用すれば，QuickTest が特定のオブジェクト・クラスのオブジェクトを記録する際に学習する一連のプロパティを変更できます。通常，そのような変更内容は，変更後に学習または記録したオブジェクトにのみ反映されます。詳細については，367 ページ「オブジェクトの認識の設定」を参照してください。また，[オブジェクトの認識] ダイアログ・ボックスで行った変更は，[更新モード] オプションを使用することによって，既存のコンポーネント内のすべてのオブジェクトの記述に適用できます。詳細については，494 ページ「コンポーネントの更新」を参照してください。

テスト・オブジェクト記述にプロパティを追加するには，次の手順を実行します。

- 1 オブジェクト・リポジトリ・ツリーの中で，記述を変更する対象となるテスト・オブジェクトを選択します。
- 2  [テスト オブジェクトの詳細] 領域の中で，[記述プロパティを追加] ボタンをクリックします。



ヒント：ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は，必要なテスト・オブジェクトを選択し，[編集] > [ステップのプロパティ] > [オブジェクトのプロパティ] を選択し，[記述プロパティを追加] ボタンをクリックして，[プロパティの追加] ダイアログ・ボックスで次の手順を実行します。

[プロパティの追加] ダイアログ・ボックスが開き、オブジェクトの識別に使用できるプロパティ（テスト・オブジェクト記述にまだ含まれていないプロパティ）が一覧表示されます。各プロパティの値は、[値] カラムに表示されます。



注：すべてのプロパティの値が表示されるのは、そのオブジェクトを含むアプリケーションが現在開いている場合だけです。アプリケーションが閉じている場合は、オブジェクトの学習時にオブジェクト記述の一部だったプロパティの値のみが表示されます。

長いプロパティ値を表示するには、[プロパティの追加] ダイアログ・ボックスのサイズを変更します。



[新規プロパティを定義] ボタンをクリックすると、このプロパティ・リストに有効なテスト・オブジェクト・プロパティを追加できます。詳細については、333 ページ「新規テスト・オブジェクト・プロパティの定義」を参照してください。

- 3 テスト・オブジェクト記述に追加する1つ以上のプロパティを選択して、[OK] をクリックします。また、プロパティをダブルクリックすることで、テスト・オブジェクト記述に当該プロパティを追加することもできます。プロパティの最初の文字を入力すれば、パターンに一致するリスト内の最初のプロパティを強調表示させることができます。

ヒント：オブジェクト記述に新しいプロパティを追加した後は、その値を変更できます。オブジェクト・プロパティ値の変更の詳細については、321 ページ「プロパティ値の指定または変更」を参照してください。


新規テスト・オブジェクト・プロパティの定義

[プロパティの追加] ダイアログ・ボックスに表示されないテスト・オブジェクト・プロパティであっても、有効なプロパティならばテスト・オブジェクト記述に追加できます。そのためには、ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は [オブジェクトリポジトリ] ウィンドウまたは [プロパティの追加] ダイアログ・ボックスを使用し、共有オブジェクト・リポジトリ内のオブジェクトの場合はオブジェクト・リポジトリ・マネージャを使用します。たとえば、QuickTest によるオブジェクトの識別に特定のプロパティを使用する必要があるけれども、そのプロパティが [プロパティの追加] ダイアログ・ボックスに表示されないとします。そのような場合は [プロパティの追加] ダイアログ・ボックスを開き、目的のプロパティをリストに追加します。



ヒント：オブジェクト・スパイの [プロパティ] タブを使用して、選択したオブジェクトで利用できる有効なテスト・オブジェクト・プロパティのリストを確認できます。オブジェクト・スパイを開くには、[ツール] > [オブジェクトスパイ] を選択するか、[オブジェクトスパイ] ツールバー・ボタンをクリックします。詳細については、290 ページ「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の表示」を参照してください。

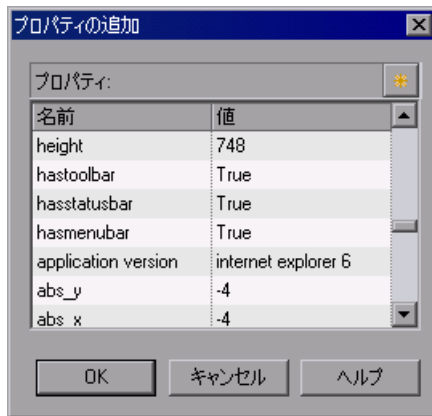
新規テスト・オブジェクト・プロパティを定義するには、次の手順を実行します。

- 1 オブジェクト・リポジトリ・ツリーの中で、新しいプロパティを定義する対象となるテスト・オブジェクトを選択します。
- 2  [テストオブジェクトの詳細] 領域で、[記述プロパティを追加] ボタンをクリックします。

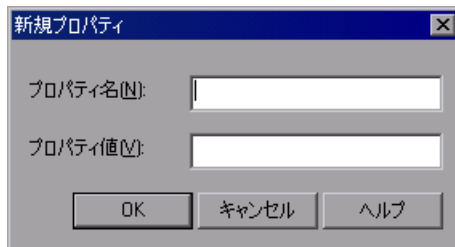


ヒント：ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は、必要なテスト・オブジェクトを選択し、[編集] > [ステップのプロパティ] > [オブジェクトのプロパティ] を選択し、[記述プロパティを追加] ボタンをクリックして、[プロパティの追加] ダイアログ・ボックスで次の手順を実行します。

[プロパティの追加] ダイアログ・ボックスが表示されます。



- 3 [新規プロパティを定義] ボタンをクリックします。[新規プロパティ] ダイアログ・ボックスが開きます。



- 4 有効なテスト・オブジェクト・プロパティを指定します。
 - ▶ [プロパティ名]：プロパティ名を入力します。
 - ▶ [プロパティ値]：プロパティの値を入力します。

注：必ず有効なテスト・オブジェクト・プロパティを入力してください。無効なプロパティを入力し、オブジェクト記述にこのプロパティを含めると、実行セッションが失敗します。

- 5 **[OK]** をクリックすることで、プロパティをリストに追加して **[新規プロパティ]** ダイアログ・ボックスを閉じます。**[プロパティの追加]** ダイアログ・ボックスの中で、新規プロパティが強調表示されます。
- 6 新規プロパティが強調表示されている状態で **[OK]** をクリックすると、オブジェクト記述に新規プロパティが追加され、**[プロパティの追加]** ダイアログ・ボックスが閉じます。

テスト・オブジェクト記述からのプロパティの削除

記述に含める必要がなくなったプロパティは、テスト・オブジェクトの記述から削除できます。そのためには、ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は **[オブジェクトリポジトリ]** ウィンドウまたは **[オブジェクトのプロパティ]** ダイアログ・ボックスを使用し、共有オブジェクト・リポジトリ内のオブジェクトの場合はオブジェクト・リポジトリ・マネージャを使用します。

テスト・オブジェクト記述からプロパティを削除するには、次の手順を実行します。

- 1 オブジェクト・リポジトリ・ツリーの中で、記述を変更する対象となるテスト・オブジェクトを選択します。
- 2 **[テストオブジェクトの詳細]** 領域の中で、テスト・オブジェクト記述から削除するプロパティを1つ以上選択します。

ヒント：ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は、必要なテスト・オブジェクトを選択し、**[編集]** > **[ステップのプロパティ]** > **[オブジェクトのプロパティ]** を選択して、**[オブジェクトのプロパティ]** ダイアログ・ボックスで次の手順を実行します。



- 3 [選択されている記述プロパティを削除] ボタンをクリックします。選択したプロパティがテスト・オブジェクト記述から削除されます。

序数識別子の指定

序数識別子は、記述がまったく同じであるほかのオブジェクト（すべてのプロパティが同じ値のオブジェクト）に対する該当テスト・オブジェクトの相対的な順序または場所を示すために割り当てる数値です。この順序値は、定義されているプロパティがオブジェクトを一意に識別するのに不十分な場合に、一意の記述を作成するための予備的なメカニズムとして使用できます。序数識別子を指定するには、ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は [オブジェクトリポジトリ] ウィンドウまたは [オブジェクトのプロパティ] ダイアログ・ボックスを使用し、共有オブジェクト・リポジトリ内のオブジェクトの場合はオブジェクト・リポジトリ・マネージャを使用します。

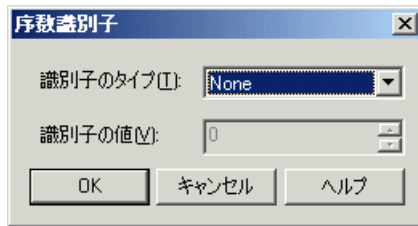
序数識別子の詳細については、374 ページ「序数識別子の選択」を参照してください。

序数識別子を指定するには、次の手順を実行します。

- 1 序数識別子を指定する対象となるテスト・オブジェクトを選択します。
- 2 [テストオブジェクトの詳細] 領域の中で、[序数識別子] 行の [タイプ、値] セルの右側にあるセルをクリックします。

ヒント：ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は、必要なテスト・オブジェクトを選択し、[編集] > [ステップのプロパティ] > [オブジェクトのプロパティ] を選択し、[序数識別子] 行の [タイプ、値] セルの右側にあるセルをクリックして、[オブジェクトのプロパティ] ダイアログ・ボックスで次の手順を実行することもできます。

- 3 [参照] ボタンをクリックします。[序数識別子] ダイアログ・ボックスが開きます。



- 4 [識別子のタイプ] ボックスで、次のいずれかのオプションを選択します。
- ▶ [Location] : 親ウィンドウ、フレーム、またはダイアログ・ボックス内においてオブジェクトが出現する順序を、その他の記述が同じであるほかのオブジェクトに対する相対位置で表します。
 - ▶ [Index] : アプリケーション・コード内においてオブジェクトが出現する順序を、その他の記述が同じであるほかのオブジェクトに対する相対位置で表します。
 - ▶ [CreationTime] : (ブラウザ・オブジェクトのみ) ブラウザが開かれた順序を、その他の記述が同じであるほかのブラウザに対する相対位置で表します。この識別子タイプは、テスト・オブジェクトの学習時に複数のブラウザ・オブジェクトが開いていた場合にのみ使用できます。
 - ▶ [None] : 序数識別子を指定しません。これは、序数識別子が記録または学習されなかった場合の標準設定値です。
- 5 [識別子の値] ボックスに序数識別子の数値を入力します。
- 6 [OK] をクリックします。選択したオブジェクトの[テストオブジェクトの詳細] 領域の該当する行に、序数識別子が表示されます。

リポジトリ・パラメータ値の割り当て

コンポーネントに関連付けられている共有オブジェクト・リポジトリで使用されるリポジトリ・パラメータを割り当てることができます。リポジトリ・パラメータを値やパラメータに割り当てることにより、実行セッション中にテスト・オブジェクトの識別に使用されるプロパティ値が指定されます。プロパティ値は定数値として指定することも、パラメータ化することもできます。パラメータ化には、ローカル・パラメータまたはコンポーネント・パラメータを使用できます。

リポジトリ・パラメータを持つオブジェクト・リポジトリが関連付けられている各コンポーネント内の各リポジトリ・パラメータを、必要に応じて割り当てることができます。たとえば、あるコンポーネントではユーザ名オブジェクトのテキスト・プロパティ値を環境変数パラメータから取得し、別のコンポーネントでは同じオブジェクト・プロパティ値が定数、ローカル・パラメータを使用するといった使い分けが可能です。

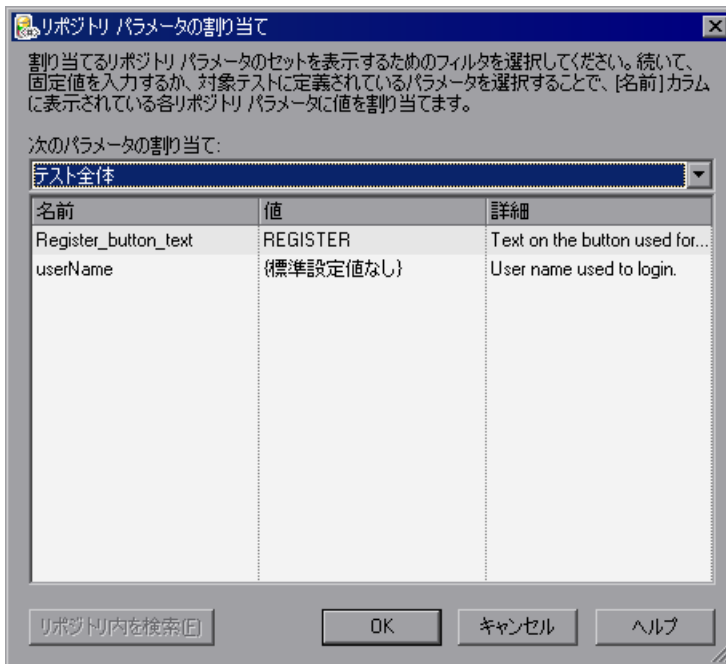
リポジトリ・パラメータを割り当てるまでは、1つのコンポーネントに関連付けられている複数の共有オブジェクト・リポジトリに、同じ名前の複数のリポジトリ・パラメータがある場合、(共有オブジェクト・リポジトリのリストで定義された)優先順位が最も高い共有オブジェクト・リポジトリのリポジトリ・パラメータが使用されます。リポジトリ・パラメータを割り当てると、QuickTest は定義した割り当てを使用します。また、パラメータの割り当て後に、優先順位または標準設定の値を変更しても影響はありません。

標準設定値がないリポジトリ・パラメータを使ってパラメータ化されたオブジェクト・プロパティ値を持つオブジェクト・リポジトリを使用するコンポーネントを開くと、割り当てを必要とするリポジトリ・パラメータが存在することを示すメッセージが欠落リソース表示枠に表示されます。その場合は、そのコンポーネント内のリポジトリ・パラメータを必要に応じて割り当てることができます。また、標準設定値があるリポジトリ・パラメータの割り当てを行ったり、すでに割り当てのあるリポジトリ・パラメータの割り当てを変更したりもできます。

リポジトリ・パラメータを割り当てない場合は、そのパラメータに対して定義された標準設定値が(存在する場合は)コンポーネントの実行中に使用されません。パラメータが未割り当ての(つまり、パラメータの標準設定値が指定されていない)場合、コンポーネントの実行は、未割り当てのパラメータ値を含むテスト・オブジェクトを識別できずに失敗する可能性があります。



リポジトリ・パラメータ値を割り当てるには、次の手順を実行します。

- 1 [リソース] > [共有オブジェクトリポジトリパラメータの割り当て] を選択します。[リポジトリパラメータの割り当て] ダイアログ・ボックスが開きます。






ヒント：コンポーネント内に未割り当てのリポジトリ・パラメータ（標準設定値がないリポジトリ・パラメータ）がある場合は、欠落リソース表示枠の [**リポジトリパラメータ**] 行をダブルクリックしてこのダイアログ・ボックスを開くこともできます。詳細については、第7章「欠落リソースの処理」を参照してください。

[リポジトリ パラメータの割り当て] ダイアログ・ボックスには、次のオプションがあります。

オプション名	詳細
<p>[次のパラメータの割り当て] フィルタ</p>	<p>表示されるパラメータのリストに対してフィルタを設定します。次の表示方法を選択できます。</p> <ul style="list-style-type: none"> ● [未割り当ての全パラメータ] : テスト内の未割り当ての値を持つすべてのパラメータが表示されます。 ● [[<コンポーネント名>] (たとえば, LogIn) : 指定したコンポーネント内のすべてのパラメータが (値の割り当ての有無に関係なく) 表示されます。
<p>[名前] カラム</p>	<p>リポジトリ・パラメータの名前。</p>
<p>[値] カラム</p>	<p>パラメータの現在の値 (存在する場合)。このカラムには、ユーザが定義した新しい値か、パラメータの作成時に定義した標準設定値のいずれかが表示されます。標準設定値が定義されていない場合、パラメータは現在未割り当てであり、「{標準設定値なし}」というテキストが表示されます。</p> <p>次のいずれかを実行できます。</p> <ul style="list-style-type: none"> ● 新しい定数値を入力する ● 該当するパラメータの [値] セルをクリックし、パラメータ化ボタン  をクリックして、値をパラメータ化する ● 該当するパラメータの [値] セルをクリックし、[標準設定値にリセット] ボタン  をクリックして、パラメータを標準設定値にリセットする
<p>[詳細] カラム</p>	<p>パラメータの説明文 (存在する場合)。</p>
<p>[リポジトリ内を検索] ボタン</p>	<p>[オブジェクトリポジトリ] ウィンドウが開き、選択したリポジトリ・パラメータを使用している最初のテスト・オブジェクトがオブジェクト・リポジトリ・ツリー内で強調表示されます。このボタンを再度クリックすると、選択したパラメータの次の出現箇所を検索できます。</p>

注：リポジトリ・パラメータの名前、標準設定値、および記述は、[リポジトリパラメータの管理] ダイアログ・ボックスで定義します。また、名前と記述については、このダイアログ・ボックスでのみ変更できます。詳細については、414 ページ「リポジトリ・パラメータの管理」を参照してください。

- 2 [次のパラメータの割り当て] の矢印をクリックし、値を定義する対象となるパラメータ・グループのリストを選択します。次の表示方法を選択できます。
 - ▶ [未割り当ての全パラメータ]：テスト内の未割り当ての値を持つすべてのパラメータが表示されます。
 - ▶ [[<コンポーネント名>] (たとえば, Login)：指定したコンポーネント内のすべてのパラメータが (値の割り当ての有無に関係なく) 表示されます。
- 3 割り当て対象パラメータの [値] セルをクリックします。値の割り当て方法として、次のいずれかを選択できます。
 -  ▶ 新しい定数値を入力する場合、または既存の定数値を変更する場合は、[値] セルに直接入力します。また、パラメータ化ボタンをクリックして、[値設定オプション] ダイアログ・ボックスに定数値を入力することもできます。このダイアログ・ボックスの詳細については、『QuickTest Professional 基本機能ユーザーズ・ガイド』を参照してください。
 -  ▶ 値をパラメータ化する場合は、パラメータ化ボタンをクリックします。[値設定オプション] ダイアログ・ボックスが開きます。値のパラメータ化には、ローカル・パラメータまたはコンポーネント・パラメータを使用できます。このダイアログ・ボックスの詳細については、『QuickTest Professional 基本機能ユーザーズ・ガイド』を参照してください。
 -  ▶ 標準設定値を復元する場合は、[標準設定値をクリア] ボタンをクリックします。[リポジトリパラメータの追加] ダイアログ・ボックスで定義した標準設定値が (存在する場合は) セルに表示されます。[リポジトリパラメータの追加] ダイアログ・ボックスの詳細については、416 ページ「リポジトリ・パラメータの追加」を参照してください。
- 4 割り当てるパラメータ値が他にもある場合は、手順3を繰り返します。次に、[OK] をクリックして [リポジトリパラメータの追加] ダイアログ・ボックスを閉じます。

オブジェクト・リポジトリへのオブジェクトの追加

コンポーネントを記録すると、操作の対象となった各オブジェクトが QuickTest によってローカル・オブジェクト・リポジトリに追加されます（関連付けられている共有オブジェクト・リポジトリにまだ存在しないオブジェクトの場合）。コンポーネントの編集集中にも、ローカル・オブジェクト・リポジトリにオブジェクトを追加できます。選択したオブジェクトだけを追加するか、特定のタイプのすべてのオブジェクト（すべてのボタン・オブジェクトなど）を追加するか、特定のタイプおよびクラスのすべてのオブジェクト（すべての WebButton オブジェクトなど）を追加するかを選択できます。

たとえば、ユーザが、オブジェクト・リポジトリにはないオブジェクトを対象にステップを実行する必要がある場合があります。また、オブジェクト・リポジトリを構築した後でテスト対象アプリケーションに追加されたオブジェクトがある場合があります。オブジェクト・リポジトリ・マネージャを使用して、このオブジェクトを共有オブジェクト・リポジトリに直接追加し、この共有オブジェクト・リポジトリを使用するすべてのアクションでの利用を可能にできます。または、コンポーネントのローカル・オブジェクト・リポジトリに追加することもできます。

本項で説明している方法でオブジェクトをオブジェクト・リポジトリに追加した場合、そのオブジェクトはローカル・オブジェクト・リポジトリに追加され、現在のコンポーネントでのみ使用できます。オブジェクトを複数のコンポーネントで使用できるように、共有オブジェクト・リポジトリに追加するには、オブジェクト・リポジトリ・マネージャを使用します。

注：オブジェクトをローカル・オブジェクト・リポジトリに追加できるのは、そのコンポーネントに関連付けられている共有オブジェクト・リポジトリに当該オブジェクトがまだ存在していない場合のみです。関連付けられている共有オブジェクト・リポジトリにオブジェクトがすでに存在する場合は、**[ローカルにコピー]** オプションを使ってそのオブジェクトをローカル・オブジェクト・リポジトリに追加できます。詳細については、315 ページ「ローカル・オブジェクト・リポジトリへのオブジェクトのコピー」を参照してください。

必要に応じて、ローカル・オブジェクト・リポジトリのテスト・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。オブジェクトを共有オブジェクト・リポジトリに結合する方法の詳細については、第16章「共有オブジェクト・リポジトリの結合」を参照してください。

注： [オブジェクトリポジトリ] ウィンドウの [**ローカルへオブジェクトを追加**] ボタンまたは、オブジェクト・リポジトリ・マネージャの [**オブジェクトの追加**] ボタンを使って、WinMenu オブジェクトをオブジェクト・リポジトリに直接追加することはできません。WinMenu オブジェクトをオブジェクト・リポジトリに追加する場合は、親オブジェクトを追加するために [**オブジェクトの追加**] ボタンまたは [**ローカルへオブジェクトを追加**] ボタンを使用し、親オブジェクトと一緒に子孫オブジェクトを追加するように選択するか、WinMenu オブジェクトに対するステップを記録してからそのステップを削除します。

ヒント： [**ローカルへオブジェクトを追加**] ボタンを使用して、(キーワード・ビューの新規ステップの) [ステップでオブジェクトを選択] ダイアログ・ボックスでアプリケーションのオブジェクトを選択することで、オブジェクトをローカル・リポジトリに追加することもできます。詳細については、156 ページ「ステップの項目の選択」を参照してください。

アプリケーションまたは Web サイトにまだ存在しないテスト・オブジェクトについても、オブジェクト・リポジトリ内に定義できます。詳細については、349 ページ「新規テスト・オブジェクトの定義」を参照してください。

[ローカルへオブジェクトを追加] オプションまたは [オブジェクトの追加] オプションを使用してオブジェクト・リポジトリにオブジェクトを追加するには、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。



▶ [オブジェクトリポジトリ] ウィンドウで、[オブジェクト] > [ローカルへオブジェクトを追加] を選択するか、[ローカルへオブジェクトを追加] ツールバー・ボタンをクリックします。



▶ オブジェクト・リポジトリ・マネージャで、[オブジェクト] > [オブジェクトの追加] を選択するか、[オブジェクトの追加] ツールバー・ボタンをクリックします。

QuickTest と [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・マネージャが最小化され、カーソルが指差しマークに変わります。

注：追加するオブジェクトを含んでいるウィンドウが部分的に別のウィンドウに隠れている場合は、部分的に隠れているウィンドウに指差しマークを数秒間置いたままにします。ウィンドウが一番手前に表示されます。これにより必要なオブジェクトのポイントとクリックができるようになります。ウィンドウを一番手前に表示するのに必要な時間の設定は、[オプション] ダイアログ・ボックスの [一般] タブで行えます。詳細については、第8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。また、左側の CTRL キーを押したままにすれば、ウィンドウのフォーカスを変更する間、指差しマークを一時的に無効にすることもできます。また、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、左側の CTRL キーを押したまま 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。

2 オブジェクト・リポジトリに追加するオブジェクトをクリックします。

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] ダイアログ・ボックスが開きます。リポジトリに追加するオブジェクトを選択し、[OK] をクリックします。

[オブジェクトの選択] ダイアログ・ボックスで選択したオブジェクトが、Web環境のブラウザやページ、または標準 Windows アプリケーションのダイアログ・ボックスなどの親オブジェクトである場合は、[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・ボックスが開きます。



次のオプションのいずれかを選択できます。

- ▶ **[選択したオブジェクトのみ]**：前もって選択したオブジェクトのプロパティと値をオブジェクト・リポジトリに追加します。子オブジェクトは追加しません。
- ▶ **[選択した次のタイプのオブジェクトとその子孫]**：前もって選択したオブジェクトのプロパティと値をオブジェクト・リポジトリに追加します。さらに、選択したオブジェクトのタイプとクラスに関するチェック・ボックスの選択に従って、子孫オブジェクトのプロパティと値も追加します。

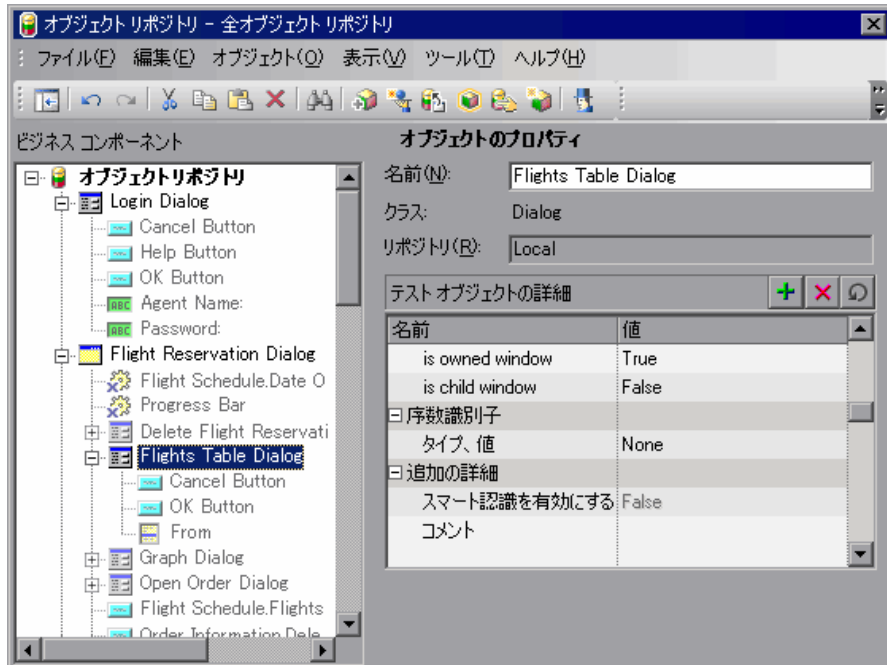
注：このリストに示される各オブジェクト・タイプは、オブジェクトを一般的な特性に基づいてグループ分けした一般的な分類です。たとえば、**[List]**タイプには、リスト・オブジェクトやリスト・ビュー・オブジェクトに加え、コンボ・ボックスが含まれます。**[Table]**タイプには、テーブルとグリッドの両方が含まれています。リストには、選択したオブジェクトだけでなく、インストールされているアドインでサポートされるすべてのオブジェクトが表示されます。一部の外部アドインでは、特定のオブジェクトのすべての子孫を追加しようとしたときに、それらのオブジェクト・タイプをリストで選択していても、特定の子オブジェクトが自動的にフィルタによって除外され、オブジェクト・リポジトリに追加されない場合があります。フィルタによって自動的に除外されるオブジェクトを追加する必要がある場合は、目的のオブジェクトを「オブジェクトの選択」ダイアログ・ボックスで選択することにより追加できます。外部アドインが特定のオブジェクトを自動的にフィルタで除外するかどうかを確認するには、そのアドインのマニュアルを参照してください。

ヒント：「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・ボックスのすべてのチェック・ボックスを選択またはクリアするには、**[すべて選択]** または **[すべてクリア]** をクリックします。

選択後に **[OK]** をクリックすることにより、「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・ボックスを閉じ、指定したオブジェクトをオブジェクト・リポジトリに追加します。

- 3 「オブジェクトリポジトリ」ウィンドウが再表示され、オブジェクト・リポジトリ内に新しいローカル・オブジェクトとそのプロパティおよび値が表示されます。オブジェクト・リポジトリ・マネージャを使用してオブジェクトを追加した場合は、アクティブな共有オブジェクト・リポジトリにオブジェクトが追加されます。

また、オブジェクト・リポジトリに新規オブジェクトの親オブジェクトが存在しない場合は、QuickTestによって親オブジェクトが追加されます。新規オブジェクトは、オブジェクト・リポジトリ・ツリー内では黒色で表示されます。



新しいテスト・オブジェクトの詳細は、ローカル・オブジェクト・リポジトリ内のほかのオブジェクトと同じように、[オブジェクトリポジトリ] ウィンドウで編集できます。詳細については、314 ページ「テスト・オブジェクトのプロパティの表示と変更」を参照してください。

新規テスト・オブジェクトの定義

アプリケーションや Web サイトにまだ存在しないテスト・オブジェクトについても、オブジェクト・リポジトリ内に定義できます。これによって、テスト対象アプリケーションの準備ができる前にオブジェクト・リポジトリを準備し、アプリケーション用のコンポーネントを構築できます。

たとえば、アプリケーション内の一部のオブジェクトについてはすでに名前、タイプ、および記述プロパティがわかっており、アプリケーション内のほかのオブジェクトについてはタイプのみがわかっているとします。このような場合でも、アプリケーションの準備ができる前に、ログイン・ページの **UserName** および **Password** フィールド用の **WebEdit** オブジェクト（および対応する親の **Page** および **Browser** オブジェクト）を作成できます。これらのオブジェクトのプロパティ値がわかっている場合は、値を追加することもできます。わかっていない場合は、アプリケーションの準備ができてテストが可能になってから値を追加します。

本項の説明に従って新規オブジェクトをオブジェクト・リポジトリ内に定義した場合、当該オブジェクトはローカル・オブジェクト・リポジトリに追加され、現在のコンポーネントでのみ使用できます。オブジェクトを複数のコンポーネントで使用できるように、共有オブジェクト・リポジトリに追加するには、オブジェクト・リポジトリ・マネージャを使用する必要があります。詳細については、第 15 章「オブジェクト・リポジトリの管理」を参照してください。

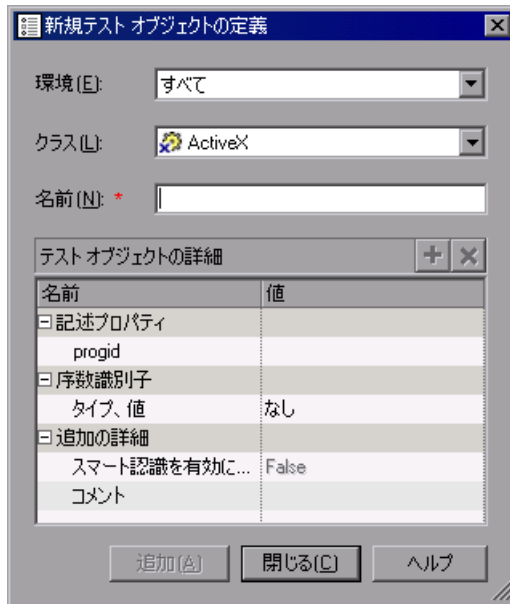
新規テスト・オブジェクトを定義した後で、アプリケーション内のオブジェクトのプロパティが定義したテスト・オブジェクトの記述と一致しない場合や、アプリケーション内のオブジェクトが更新された場合は、いつでもオブジェクト記述を更新できます。詳細については、324 ページ「アプリケーション内のオブジェクトからのテスト・オブジェクト・プロパティの更新」を参照してください。

新規テスト・オブジェクトを定義するには、次の手順を実行します。

- 1 正しいオブジェクト階層に従って、新しくオブジェクトを定義する位置のすぐ上にあるオブジェクトを選択します。



- 2 [新規テスト オブジェクトの定義] ボタンをクリックするか、[オブジェクト] > [新規テスト オブジェクトの定義] を選択します。[新規テストオブジェクトの定義] ダイアログ・ボックスが表示されます。



- 3 [環境] ボックスで、適切な環境を選択します。選択した環境に関連付けられているテスト・オブジェクト・クラスが [クラス] ボックスに表示されます。

注：[環境] ボックスに含まれている環境は、読み込まれているアドイン環境に対応するものです。アドインの読み込みの詳細については、681 ページ「QuickTest アドインの使用法」を参照してください。

- 4 [クラス] ボックスの中で、定義するテスト・オブジェクトのクラスを選択します。

- 5 [名前] ボックスに、新規テスト・オブジェクトの名前を入力します。名前を入力すると、[テストオブジェクトの詳細] 領域が有効になります。
- 6 [テストオブジェクトの詳細] 領域で、テスト・オブジェクトのプロパティと値を定義します。[テストオブジェクトの詳細] 領域には、[オブジェクトの認識] ダイアログ・ボックスの中でオブジェクト・クラスに対して定義された必須のプロパティが自動的に含まれます。必要に応じてプロパティを追加または削除したり、プロパティに対して値を定義したりできます。詳細については、314 ページ「テスト・オブジェクトのプロパティの表示と変更」を参照してください。
- 7 [追加] をクリックします。新規テスト・オブジェクトが、ローカル・オブジェクト・リポジトリ内の選択した場所に追加されます。
- 8 手順3～7を繰り返して追加のテスト・オブジェクトを定義するか、[閉じる] をクリックして [新規テストオブジェクトの定義] ダイアログ・ボックスを閉じます。

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

オブジェクトをコピー、貼り付け、および移動するには、ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は [オブジェクトリポジトリ] ウィンドウを使用し、共有オブジェクト・リポジトリ内のオブジェクトおよび共有オブジェクト・リポジトリ内のオブジェクトの場合はオブジェクト・リポジトリ・マネージャを使用します。ただし、オブジェクト・リポジトリのルート・ノードは変更できません。共有オブジェクト・リポジトリからローカル・オブジェクト・リポジトリにオブジェクトをコピーし、それをローカルで変更することもできます。詳細については、315 ページ「ローカル・オブジェクト・リポジトリへのオブジェクトのコピー」を参照してください。

オブジェクトをドラッグしてツリー内で上下に移動し、必要な場所にドロップすることで、オブジェクトをオブジェクト・リポジトリ内の別の場所に移動できます。CTRL キーを押しながらオブジェクトをドラッグし、ツリー内の必要な場所にドロップすることで、オブジェクトをオブジェクト・リポジトリ内の別の場所にコピーできます。

オブジェクトをドラッグすると、標準設定では子オブジェクトもすべて一緒に移動またはコピーされます。子オブジェクトなしで該当するオブジェクトを移動またはコピーするには、マウスの右ボタンを使ってオブジェクトをドラッグします。必要な場所にオブジェクトをドロップする際に、子オブジェクトと一緒にドロップするかどうかを選択できます。



該当するツールバー・ボタンまたは **[編集]** メニューを使用して、オブジェクト・リポジトリの中でオブジェクトの切り取り、コピー、および貼り付けができます。また、オブジェクト・リポジトリ・マネージャでは、異なる共有オブジェクト・リポジトリ間でオブジェクトの切り取り、コピー、および貼り付けを行うこともできます。オブジェクトの切り取り、コピー、および貼り付け操作を行うと、選択したオブジェクトの子オブジェクトも操作の対象となります。

[オブジェクトリポジトリ] ウィンドウで、コンポーネントに関連付けられている共有オブジェクト・リポジトリからオブジェクトをコピー、貼り付け、または移動すると、コンポーネントのローカル・オブジェクト・リポジトリにオブジェクトがコピー、貼り付け、または移動されます。

未割り当てのリポジトリ・パラメータを含んだオブジェクトを共有オブジェクト・リポジトリからローカル・オブジェクト・リポジトリにコピー、貼り付け、または移動することはできません。オブジェクトを共有オブジェクト・リポジトリからローカル・オブジェクト・リポジトリにコピー、貼り付け、または移動するとき、当該オブジェクトまたはその親オブジェクトの1つが1つ以上のリポジトリ・パラメータを使用してパラメータ化されている場合、当該リポジトリ・パラメータの値は、オブジェクトをコピー、貼り付け、または移動するときに変換されます。たとえば、リポジトリ・パラメータがローカル・パラメータに割り当てられている場合、プロパティはローカルパラメータを使用してパラメータ化されます。値が定数値である場合、プロパティには同じ定数値が設定されます。

オブジェクト・リポジトリ・マネージャの中で、異なる共有オブジェクト・リポジトリ間でオブジェクトをコピーするには、対象となるそれぞれの共有オブジェクト・リポジトリを開き、一方のウィンドウからオブジェクトをドラッグし、もう一方のウィンドウの必要な場所にドロップします。異なる共有オブジェクト・リポジトリ間でオブジェクトを移動するには、それぞれの共有オブジェクト・リポジトリを開き、CTRL キーを押しながら一方のウィンドウからオブジェクトをドラッグし、もう一方のウィンドウの必要な場所にドロップします。オブジェクトを移動すると、そのオブジェクトは一方のオブジェクト・リポジトリから削除され、もう一方のオブジェクト・リポジトリに追加されます。



注：[編集] > [元に戻す] および [編集] > [やり直し] オプション、または [元に戻す] および [やり直し] ボタンを使用して、変更の取り消しや繰り返しができます。オブジェクト・リポジトリを保存した後で、保存操作の前に実行した操作を元に戻したりやり直したりすることはできません。

オブジェクトをコピー、貼り付け、または移動する場合は、次のことを考慮してください。

- ▶ オブジェクト階層を変更する場合は、変更後の階層構造が記録済みの階層構造として有効であることを確認してください。
- ▶ オブジェクトを別の階層レベルに貼り付けたり移動したりするとき、共通の親オブジェクトに至るまでのすべてのオブジェクトをコピーするかどうかを（この操作を実行したときに表示されるメッセージ内で）選択できます。
- ▶ オブジェクトを直接の親に移動する場合、オブジェクトのコピー（1つずつ大きくなる数字が名前の後ろに付加されます）が作成され、元のオブジェクトの兄弟として貼り付けられます。
- ▶ オブジェクトの切り取りまたはコピーを行い、それを親オブジェクトに貼り付けると、そのオブジェクトのコピーが作成され（1つずつ大きくなる数字が名前の後ろに付加されます）、元のオブジェクトと同じレベルに挿入されます。
- ▶ オブジェクトをその子孫に移動することはできません。
- ▶ オブジェクトをオブジェクト階層の最下位にあるオブジェクト（子オブジェクトを持つことができないオブジェクト）の子になるようにコピーも移動もできません。

注： WinMenu オブジェクトは、親オブジェクトまたは最下位オブジェクトにすることができます。

オブジェクト・リポジトリからのオブジェクトの削除

コンポーネントからステップを削除しても、対応するオブジェクトはオブジェクト・リポジトリから削除されずに残ります。


ローカル・オブジェクト・リポジトリを対象に作業をしているときに、削除したステップに含まれるオブジェクトがそのコンポーネント内のほかのステップに一切含まれていなければ、そのオブジェクトをオブジェクト・リポジトリから削除できます。

共有オブジェクト・リポジトリを対象に作業をしている場合は、オブジェクト・リポジトリからオブジェクトを削除する前に、同じ共有オブジェクト・リポジトリを使用しているどのコンポーネントにもそのオブジェクトが含まれていないことを確認する必要があります。

オブジェクトを削除するには、ローカル・オブジェクト・リポジトリ内のオブジェクトの場合は [オブジェクトリポジトリ] ウィンドウを使用し、共有オブジェクト・リポジトリ内のオブジェクトの場合はオブジェクト・リポジトリ・マネージャを使用します。

注：オブジェクト・リポジトリから削除したオブジェクトへの参照がコンポーネントに含まれていると、コンポーネントの実行は失敗します。

オブジェクト・リポジトリからオブジェクトを削除するには、次の手順を実行します。

- 1 リポジトリ・ツリーの中で、削除するオブジェクトを選択します。
- 2  **[削除]** ボタンをクリックするか、**[編集]** > **[削除]** を選択します。
- 3 **[はい]** をクリックし、オブジェクトの削除を確定します。オブジェクトがオブジェクト・リポジトリから削除されます。



ヒント：[削除] ボタンを使用すれば、テスト・オブジェクトだけでなく、オブジェクト・リポジトリ内の選択した任意の値や項目を削除できます。たとえば、このボタンを使用して、オブジェクト名やプロパティ値の一部を削除できます。

オブジェクトの場所の特定

オブジェクト・リポジトリ内の特定のオブジェクトを検索する方法は複数あります。オブジェクトのタイプに基づいて、オブジェクトを検索できます。たとえば、特定のエディット・ボックスを検索したり、アプリケーション内のオブジェクトをポイントして、リポジトリ内の対応するオブジェクトを自動的に強調表示させたりできます。オブジェクト・リポジトリ内でオブジェクトを選択すると、アプリケーション上でそのオブジェクトが強調表示され、どのオブジェクトに該当するかを確認できます。ローカル・オブジェクト（およびオブジェクト・リポジトリ・マネージャを使用する場合、編集可能な共有オブジェクト・リポジトリ内の共有オブジェクト）の場合、特定のプロパティ値を他のプロパティ値で置換することもできます。たとえば、プロパティ値 `userName` を `user name` で置換できます。


オブジェクト・リポジトリ内のオブジェクトの検索

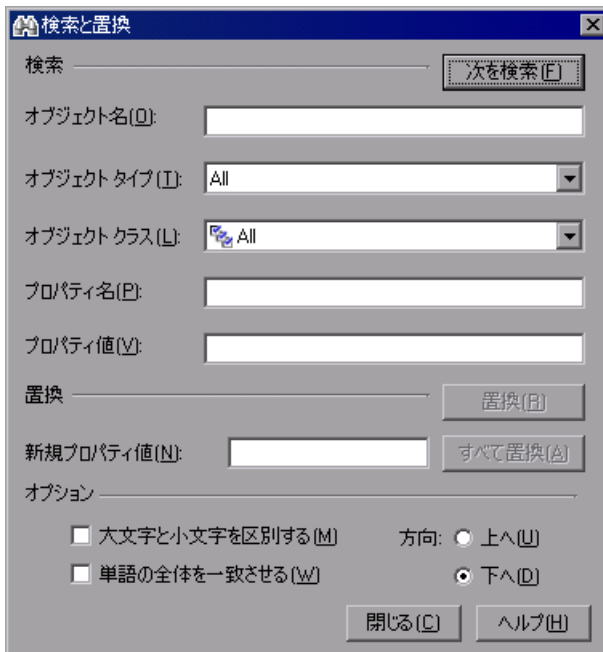
[検索/置換] ダイアログ・ボックスを使用して、オブジェクト・リポジトリ内のオブジェクト、プロパティ、またはプロパティ値を検索できます。また、指定したプロパティ値を検索して置換することもできます。

ローカル・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置換するには、[オブジェクトリポジトリ] ウィンドウを使用します。共有オブジェクト・リポジトリ内のオブジェクトのプロパティ値を置換するには、オブジェクト・リポジトリ・マネージャを使用します。

注： [検索 / 置換] ダイアログ・ボックスを使ってプロパティやオブジェクトの名前を置換することはできません。また、読み取り専用のコンポーネントのプロパティ値を置換することはできません。

オブジェクト・リポジトリ内のオブジェクト、プロパティ、またはプロパティ値を検索するには、次の手順を実行します。

- 1 対象となるオブジェクト・リポジトリが ([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・マネージャで) 開いていることを確認します。
- 2  **[検索 / 置換]** ボタンをクリックするか、**[編集] > [検索 / 置換]** を選択します。**[検索 / 置換]** ダイアログ・ボックスが開きます。



検索と置換

検索 _____ **次を検索(F)**

オブジェクト名(O): _____

オブジェクトタイプ(T): All

オブジェクトクラス(L): All

プロパティ名(P): _____

プロパティ値(V): _____

置換 _____ **置換(R)**

新規プロパティ値(N): _____ **すべて置換(A)**

オプション _____

大文字と小文字を区別する(M) 方向: 上へ(U)

単語の全体を一致させる(W) 下へ(D)

閉じる(C) **ヘルプ(H)**

- 3 オブジェクト、プロパティ、またはプロパティ値の検索に使用する条件を1つ以上指定します。
 - ▶ **[オブジェクト名]**：検索するオブジェクトの名前または名前の一部を入力します。
 - ▶ **[オブジェクトタイプ]**：検索するオブジェクトのタイプ（**Button** など）を選択します。

注：このリストに示される各オブジェクト・タイプは、オブジェクトを一般的な特性に基づいてグループ分けした一般的な分類です。たとえば、**[List]** タイプには、リスト・オブジェクトやリスト・ビュー・オブジェクトに加え、コンボ・ボックスが含まれます。**[Table]** タイプには、テーブルとグリッドの両方が含まれています。
 - ▶ **[オブジェクトクラス]**：検索するオブジェクトのクラス（**WebButton** など）を選択します。選択可能なクラスは、**[オブジェクトタイプ]** ボックスでの選択に応じて異なります。
 - ▶ **[プロパティ名]**：検索するプロパティの名前または名前的一部分を入力します。
 - ▶ **[プロパティ値]**：検索するプロパティ値またはプロパティ値の一部を指定します。
- 4 プロパティ値を指定し、その値を別の値に置換したい場合は、**[新規プロパティ値]** ボックスに新しいプロパティ値を入力します。
- 5 次のようにして、検索パラメータを指定します。
 - ▶ 大文字と小文字を区別して検索するには、**[大文字と小文字を区別する]** を選択します。
 - ▶ 入力した1つの単語に完全に一致する文字列だけを検索するには、**[単語の全体を一致させる]** を選択します。
 - ▶ 検索する方向を指定します。**[上へ]** または **[下へ]** のいずれかを選択できます。

- 6 次のいずれかの方法で検索操作または置換操作を実行します。検索は、オブジェクト・リポジトリ全体を対象に、現在選択されているオブジェクトを起点に、指定した方向に向かって実行されます。次に一致する文字列を検索するには、再度 **[次を検索]** をクリックします。
- ▶ 指定したオブジェクト、プロパティ、またはプロパティ値を検索するには、**[次を検索]** をクリックします。検索文字列に一致する最初の文字列が表示されます。
 - ▶ 検索対象のプロパティ値が出現する箇所を1つずつ検索して置換するには、**[次を検索]** をクリックします。一致する文字列が見つかったら、**[置換]** をクリックします。該当するプロパティ値が置換され、次に一致するプロパティ値が出現する箇所が（存在する場合は）強調表示されます。
 - ▶ 指定したプロパティ値が出現するすべての箇所を新しいプロパティ値で置換するには、**[すべて置換]** をクリックします。編集可能になっていない共有オブジェクト・リポジトリ内の出現箇所は、変更されません。


アプリケーション内のオブジェクトの強調表示

オブジェクト・リポジトリ内のオブジェクトを選択すると、テスト対象アプリケーション内または Web サイト内の対応するオブジェクトが強調表示されます。強調表示するオブジェクトを選択すると、対応するアプリケーションにおける当該オブジェクトの位置を示すために、アプリケーション内のオブジェクトを囲む青色の枠が一時的に表示され、少しの間点滅します。オブジェクトが見えるようアプリケーションが正しいコンテキストで開いている必要があります。

たとえば、Web ページの **[User Name]** エディット・ボックスの位置を特定するには、Web ブラウザで該当するページを開き、オブジェクト・リポジトリ内の「userName」テスト・オブジェクトを選択します。**[アプリケーション内で強調表示]** オプションを選択すると、ブラウザ上の Web ページ内にある **[User Name]** エディット・ボックスが枠に囲まれ、数回点滅します。

注：枠の表示も点滅も一時的な動作です。

アプリケーション内のオブジェクトを強調表示するには、次の手順を実行します。

- 1 アプリケーションまたは Web サイトの適切なウィンドウまたはページが開いていることを確認します。
- 2 オブジェクト・リポジトリの中で、強調表示するオブジェクトをクリックします。
- 3  **[アプリケーション内で強調表示]** ボタンをクリックするか、**[表示]** > **[アプリケーション内で強調表示]** を選択します。選択したオブジェクトが、アプリケーション内または Web サイト内で青い枠線を使用して強調表示されます。

Find A Flight


注：アプリケーションまたは Web サイトの適切なコンテキストが開いていない場合、オブジェクトは強調表示されません。

オブジェクト・リポジトリ内のオブジェクトの場所の特定

テスト対象のアプリケーション内または Web サイト内のオブジェクトを選択して、オブジェクト・リポジトリ内の対応するテスト・オブジェクトを強調表示させることが可能です。

たとえば、Web ページ内にある「Find a Flight」画像の場所を特定するには、指差しマークを使って Web ページ内で当該画像を選択します。選択ダイアログ・ボックスで「Find a Flight」画像オブジェクトを選択して **[OK]** をクリックすると、オブジェクト・リポジトリ・ツリー内の親階層が展開され、「Find a Flight」画像テスト・オブジェクトが強調表示されます。

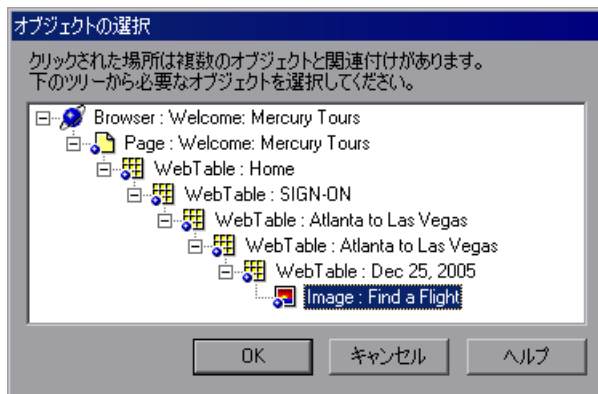
オブジェクト・リポジトリ内のオブジェクトの場所を特定するには、次の手順を実行します。

- 1 アプリケーションまたは Web サイトの適切なウィンドウまたはページが開いていることを確認します。
- 2  **[リポジトリの中で検索]** ボタンをクリックするか、**[表示]** > **[リポジトリの中で検索]** を選択します。QuickTest が最小化し、カーソルが指差し型に変わります。

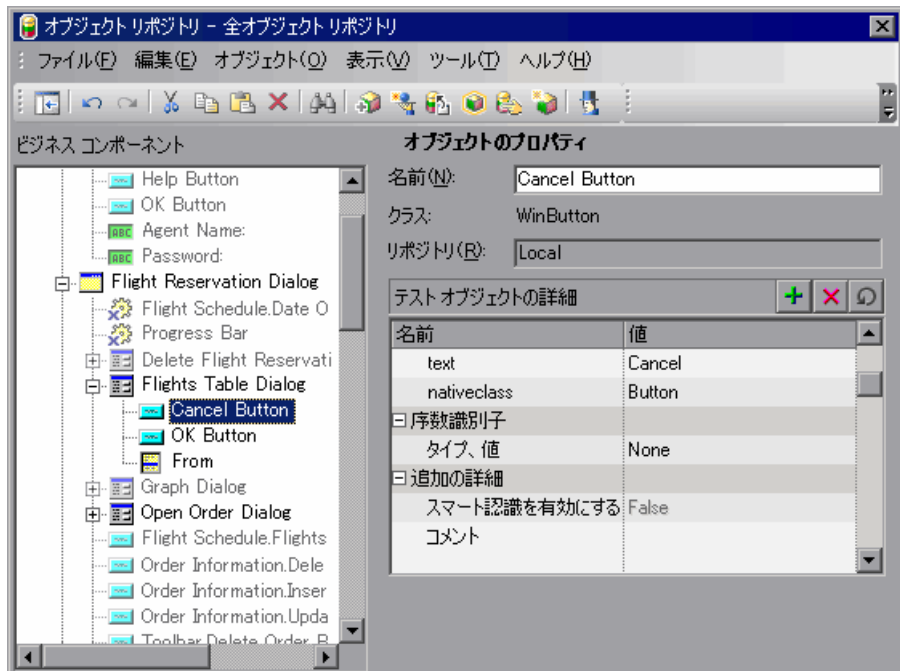
- 3 指差しマークを使用して、アプリケーション内または Web サイト内の目的のオブジェクトをクリックします。

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マウスオーバ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。クリック対象オブジェクトを含んでいるウィンドウが部分的に別のウィンドウに隠れている場合には、部分的に隠れているウィンドウに指差しボタンを数秒間置いたままにすればウィンドウが手前に表示されるため、必要なオブジェクトをポイントしてクリックすることができます。また、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、左側の CTRL キーを押したまま 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] ダイアログ・ボックスが開きます。



- 4 オブジェクト・リポジトリ内の場所を特定する対象とな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が、オブジェクト・リポジトリ内で強調表示されます。



ヒント：該当するオブジェクト・リポジトリが開いていない場合や、オブジェクトが見つからない場合、オブジェクトは強調表示されません。オブジェクト・リポジトリ・マネージャで複数の共有オブジェクト・リポジトリを開いており、選択したオブジェクトがアクティブなオブジェクト・リポジトリ内で見つからない場合は、現在開いているすべてのオブジェクト・リポジトリの中でオブジェクトを検索するかどうかを選択できます。

実行セッション中のテスト・オブジェクトの操作

QuickTest では、実行セッション中に初めてオブジェクトに遭遇したときに、その実行セッションのためにテスト・オブジェクトの一時的なバージョンが作成されます。この一時的なバージョンのオブジェクトは、オブジェクト記述を使って作成されます。以降、コンポーネントでは、オブジェクト・リポジトリ内のテスト・オブジェクトではなく、この一時的なバージョンのテスト・オブジェクトが参照されます。

注：記録および実行セッション中は、[オブジェクトリポジトリ] ウィンドウが読み取り専用になります。

テスト・オブジェクトのプロパティに変更を加える方法は複数あります。必要に応じて最適な方法を選択してください。

- ▶ アプリケーション内のオブジェクトの新しい静的プロパティに合わせてテスト・オブジェクトのプロパティ値を手作業で変更できます。詳細については、321 ページ「プロパティ値の指定または変更」を参照してください。
- ▶ **SetTOProperty** メソッドを使用する関数（操作）を作成すれば、オブジェクト・リポジトリのプロパティ値を変更することなく、実行セッション中にテスト・オブジェクトのプロパティを変更できます。詳細については、363 ページ「実行セッション中のテスト・オブジェクト・プロパティの変更」を参照してください。
- ▶ QuickTest がオブジェクトの識別に使用する一連のプロパティを変更し、一部のプロパティが変化しても、オブジェクトを識別することができます。詳細については、319 ページ「テスト・オブジェクトのプロパティの変更」を参照してください。
- ▶ 正規表現を使用して、定義した条件またはパターンに基づいてオブジェクトを特定できます。

実行セッション中のテスト・オブジェクトの作成

プログラムの記述を使用して、アプリケーションのオブジェクトを表す一時的なテスト・オブジェクトを作成できます。これらのオブジェクトに対しては、オブジェクト・リポジトリを参照せずに操作を実行できます。たとえば、Web サイトのフォームにエディット・ボックスが追加されたとします。プログラムの記述を使用して、新しいエディット・ボックスに値を入力するステートメントをユーザ定義関数に追加することで、記録もオブジェクト・リポジトリへの追加もしていないオブジェクトを QuickTest に認識させることができます。プログラムの記述の詳細については、640 ページ「プログラムの記述の使用」を参照してください。

実行セッション中のテスト・オブジェクト・プロパティの変更

ユーザ定義関数の中で **SetTOProperty** ステートメントを追加することで、オブジェクト・リポジトリ内の永続的な値に影響を与えることなく、実行セッション中に一時的なバージョンのオブジェクトのプロパティを変更できます。

SetTOProperty メソッドには、次の構文を使用します。

Object(description).SetTOProperty Property, Value

詳細については、『**QuickTest Professional オブジェクト・モデル・リファレンス**』を参照してください。

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

コンポーネントのローカル・オブジェクト・リポジトリに含まれているすべてのオブジェクトを、ファイル・システム内の新しい共有オブジェクト・リポジトリまたは（QuickTest が Quality Center に接続されている場合は）Quality Center プロジェクトにエクスポートできます。エクスポートすると、ほかのコンポーネントからもそれらのローカル・オブジェクトにアクセスできるようになります。ローカル・オブジェクトを新しい共有オブジェクト・リポジトリにエクスポートするには、[オブジェクト リポジトリ] ウィンドウを使用します。

注：ローカル・オブジェクト・リポジトリにパラメータ化されたプロパティ値を持つオブジェクトが含まれている場合、それらのオブジェクトを共有オブジェクト・リポジトリにエクスポートすると、パラメータ化された値はリポジトリ・パラメータになり、以前の値が割り当てられます。たとえば、ローカル・オブジェクトがコンポーネント・パラメータを使用してパラメータ化されていた場合、当該ローカル・オブジェクトを共有オブジェクト・リポジトリにエクスポートすると、パラメータはリポジトリ・パラメータに変更され、同じ値を持つコンポーネント・パラメータに割り当てられます。リポジトリ・パラメータの詳細については、第15章「オブジェクト・リポジトリの管理」を参照してください。

ヒント：ローカル・オブジェクトをエクスポートした後は、オブジェクト・リポジトリ結合ツールを使用して、エクスポートされたオブジェクトを含む共有オブジェクト・リポジトリを別の共有オブジェクト・リポジトリと結合できます。詳細については、第16章「共有オブジェクト・リポジトリの結合」を参照してください。

ローカル・オブジェクトを新しい共有オブジェクト・リポジトリにエクスポートするには、次の手順を実行します。

- 1 エクスポートするローカル・オブジェクトを含んでいるコンポーネントを開きます。
- 2 [オブジェクト リポジトリ] ウィンドウが開いていることを確認します。
- 3 [ファイル] > [ローカル オブジェクトをエクスポート] を選択します。[オブジェクト リポジトリのエクスポート] ダイアログ・ボックスが表示されます。

注：Quality Center に接続している場合と、標準ファイル・システムを使用する場合とでは、表示されるダイアログ・ボックスが異なります。それら 2 種類のダイアログ・ボックスは、[オブジェクト リポジトリのエクスポート] ダイアログ・ボックスの [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。

- 4 ファイルを保存する場所を選択し、ファイル名または添付名を指定して、[保存] または [OK] (ファイル・システムにエクスポートするのか Quality Center プロジェクトにエクスポートするのかに応じて異なる) をクリックします。

オブジェクト・リポジトリが、指定した共有オブジェクト・リポジトリ (.tsr 拡張子の付いたファイル) にエクスポートされます。この新しい共有オブジェクト・リポジトリは、ほかの共有オブジェクト・リポジトリと同じように使用できます。

第 14 章

オブジェクトの認識の設定

オブジェクトに対する操作を記録するとき、またはオブジェクト・リポジトリにオブジェクトを追加するとき、QuickTest は当該オブジェクトのプロパティと値のセットを学習します。このプロパティと値のセットは、当該オブジェクトをオブジェクト階層の中で一意に識別する記述です。多くの場合、この記述で、QuickTest が実行セッション中にオブジェクトを十分に識別できます。

特定のオブジェクト・クラスを表す記述が、アプリケーションのオブジェクトを最も論理的に記述するものでないことが判明した場合や、オブジェクト記述の中のプロパティの値が頻繁に変わることが予想される場合には、QuickTest によるオブジェクトの学習方法、識別方法を設定できます。また、ユーザ定義オブジェクトを標準のテスト・オブジェクト・クラスに割り当て、QuickTest によるユーザ定義オブジェクト・クラスのオブジェクトを学習する方法も設定できます。

本章では、次の内容について説明します。

- ▶ 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
- ▶ 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

オブジェクトの認識の設定について

QuickTest には、各テスト・オブジェクトに対して学習するプロパティのセットがあらかじめ用意されています。記録または追加対象オブジェクトを一意に識別するのにこれらの必須プロパティ値では十分でない場合、QuickTest は何らかの補足プロパティまたは序数識別子を追加して、一意の記述を作成します。

「必須プロパティ」とは、QuickTest が特定のテスト・オブジェクト・クラスについて必ず学習するプロパティです。

「補足プロパティ」とは、QuickTest がアプリケーションの特定のオブジェクトについて学習する必須プロパティでは一意の記述を作成するのに不十分である場合にだけ QuickTest が学習するプロパティです。1つのオブジェクト・クラスに複数の補足プロパティが定義されている場合、QuickTest は補足プロパティを1つずつ順番に学習し、オブジェクトの一意の記述ができたところで学習を止めます。QuickTest が学習した補足プロパティはテスト・オブジェクトの記述に追加されます。

注：すべての定義済みの必須および補足プロパティを組み合わせても一意のテスト・オブジェクト記述の作成に十分でない場合、QuickTest は選択された序数識別子の値も学習します。詳細については、374 ページ「序数識別子の選択」を参照してください。

コンポーネントの実行の際、QuickTest は、学習した（序数識別子のない）記述に一致するオブジェクトを検索します。記述に一致するオブジェクトが見つからない場合や、記述に一致するオブジェクトが複数ある場合、QuickTest はスマート認識メカニズムを使って（ただし、有効になっている場合）、オブジェクトを識別します。多くの場合、スマート認識定義が存在すれば、学習した記述が1つ以上のプロパティ値が変更されたことで識別に失敗する場合に、QuickTest によるオブジェクトの識別に役立ちます。テスト・オブジェクト記述は、スマート認識メカニズムでもオブジェクトの候補を1つに絞り込めない場合に限り、序数識別子と組み合わせて使われます。

[オブジェクトの認識] ダイアログ・ボックス ([ツール] > [オブジェクトの認識]) では、QuickTest がアプリケーションの中のオブジェクトの記述を学習するのに使う、必須プロパティ、補足プロパティ、および序数識別子プロパティを設定できるほか、スマート認識の有効化と設定が可能です。

また、新しいユーザ定義クラスを設定し、それらを既存のテスト・オブジェクト・クラスにマップして、コンポーネント実行時にユーザ定義クラスからオブジェクトを QuickTest が認識するようにもできます。

[オブジェクトの認識] ダイアログ・ボックス

[オブジェクトの認識] ダイアログ・ボックスのメイン画面を使って、必須および補足プロパティの設定、序数識別子の選択、および各テスト・オブジェクトに対するスマート認識メカニズムを有効にするかどうかを指定します。

[オブジェクトの識別] ダイアログ・ボックスから、ユーザ定義オブジェクト・クラスの定義とそれらの標準ウィンドウ・オブジェクト・クラスへの割り当ても行えます。また、[**テストオブジェクトクラス**] リストに表示される任意のオブジェクトにスマート識別メカニズムを設定することもできます。

注：

[オブジェクトの認識] ダイアログ・ボックスで行った変更は、すでにオブジェクト・リポジトリに追加されたオブジェクトには影響しません。

WinMenu, VbLabel, VbObject, VbToolbar オブジェクトなど、特定のテスト・オブジェクトの学習したスマート認識プロパティは、設定できません。したがって、これらのオブジェクトには選択した環境の [**テストオブジェクトクラス**] リストが含まれません。

詳細については、次を参照してください。

- ▶ 370 ページ「必須および補足記録プロパティの設定」
- ▶ 374 ページ「序数識別子の選択」
- ▶ 380 ページ「スマート認識の有効化と無効化」
- ▶ 381 ページ「テスト・オブジェクトの標準設定のオブジェクト認識設定の復元」
- ▶ 381 ページ「オブジェクト認識設定用の自動スクリプトの生成」
- ▶ 382 ページ「スマート認識の設定」
- ▶ 391 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」

必須および補足記録プロパティの設定

QuickTest が特定のオブジェクト・クラスのために使う記述が、アプリケーションのオブジェクトを記述するために十分には論理的でないことが判明した場合や、オブジェクト記述の中で現在使われているプロパティの値が変わることが予想される場合には、そのクラスのオブジェクトを学習するときに QuickTest が学習する必須および補足プロパティを変更できます。

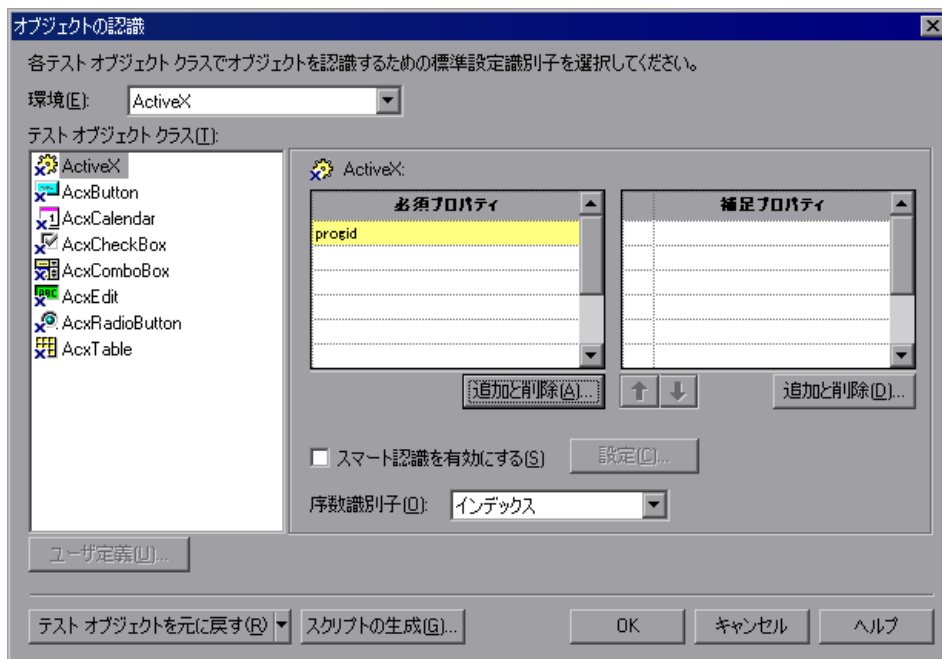
実行セッション中、QuickTest はテスト・オブジェクト記述中のすべてのプロパティに一致するオブジェクトを探します。このとき、必須プロパティとして学習したものと補足プロパティとして学習したものを区別しません。

たとえば、Web Image オブジェクトの標準の必須プロパティは、**alt**、**html tag**、および **image type** プロパティです。標準の補足プロパティは定義されていません。Web サイトに複数の広告を循環して表示するいくつかの広告枠があるものとします。これらの広告枠のそれぞれのイメージをクリックするコンポーネントを記録したいものとします。しかし、それぞれの広告イメージの **alt** 値は異なるため、コンポーネントを作成すると1つの **alt** 値が記録され、そしてコンポーネントを実行するとほとんどの場合、別の **alt** 値がキャプチャされるため、テスト実行が失敗することになります。この場合、Web Image 必須プロパティ・リストから **alt** プロパティを削除できます。その代わりに、サイトの特定の広告枠に表示される各広告イメージは、イメージの **name** プロパティの値が同じなので、必須プロパティにその **name** プロパティを追加して、QuickTest が一意にオブジェクトを識別できるようにします。

また、ページの複数の場所に表示される Web イメージに（たとえば、ロゴがページ一番上と下に表示されるなど）、Web デザイナーによって Image タグに特別な **ID** プロパティが追加されたとします。ページに一度だけ表示されるイメージであれば、一意の記述を作成するには必須プロパティで十分ですが、同じページにイメージが複数回表示される場合には、QuickTest に **ID** プロパティも学習させたいところです。このためには、**ID** プロパティを補足プロパティとして追加して、一意のテスト・オブジェクト記述の作成に必要な場合にだけ QuickTest に **ID** プロパティを学習させるようにします。

テスト・オブジェクト・クラスのための必須および補足プロパティを作成するには、次の手順を実行します。

- 1 [ツール] > [オブジェクトの認識] を選択します。[オブジェクトの認識] ダイアログ・ボックスが表示されます。

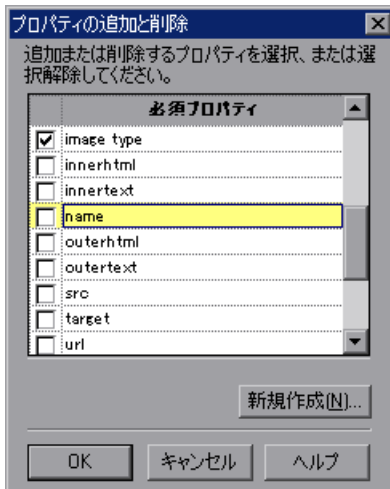


- 2 [環境] リストで適切な環境を選択します。選択した環境に関連付けられているテスト・オブジェクト・クラスが [テストオブジェクトクラス] リストにアルファベット順に表示されます (標準の Windows では、ユーザ定義オブジェクトがリストの一番下に表示されます)。

注：[環境] リストに含まれている環境は、読み込まれたアドイン環境に対応するものです。アドインの読み込みの詳細については、681 ページ「QuickTest アドインの使用法」を参照してください。

- 3 [テストオブジェクトクラス] リストで、設定するテスト・オブジェクト・クラスを選択します。

- 4 [必須 プロパティ] リストで, [追加と削除] をクリックします。必須プロパティのための [プロパティの追加と削除] ダイアログ・ボックスが表示されます。



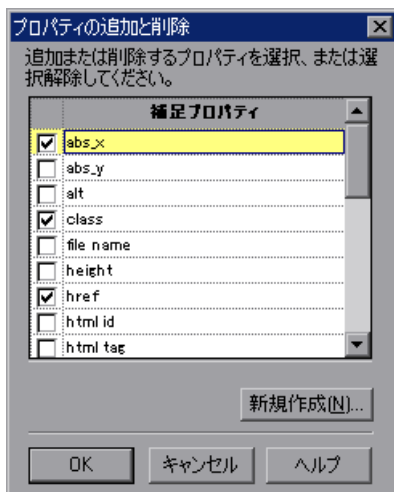
- 5 必須プロパティ・リストに含めるプロパティを選択します。また、リストから削除するプロパティを消去します。

注：同じプロパティを必須と補足の両方のプロパティ・リストに含めることはできません。

[**新規作成**] をクリックして、表示されるダイアログ・ボックスに有効なプロパティ名を指定することにより、新規プロパティを指定できます。

ヒント： attribute/ <プロパティ名> の形式を使用して、Web オブジェクトに使用可能なプロパティの集合にプロパティ名を追加することもできます。これを行うためには、[**新規作成**] をクリックします。[新規プロパティ] ダイアログ・ボックスが開きます。有効なプロパティを、 attribute/ <プロパティ名> の形式で入力して、[**OK**] をクリックします。新プロパティが [**プロパティの追加と削除**] リストに追加されます。たとえば、MyColor というプロパティを追加するには、 attribute/MyColor と入力します。

- 6 [OK] をクリックし、[プロパティの追加と削除] ダイアログ・ボックスを閉じます。更新された必須プロパティの集合が [必須プロパティ] リストに表示されます。
- 7 [補足 プロパティ] リストで、[追加と削除] をクリックします。必須プロパティのための [プロパティの追加と削除] ダイアログ・ボックスが表示されます。



- 8 補足プロパティ・リストに含めるプロパティを選択します。また、リストから削除するプロパティを消去します。

注：同じプロパティを必須と補足の両方のプロパティ・リストに含めることはできません。

[新規作成] をクリックして、表示されるダイアログ・ボックスに有効なプロパティ名を指定することにより、新規プロパティを指定できます。

ヒント：attribute/ <プロパティ名>の形式を使用して、Web オブジェクトに使用可能なプロパティの集合にプロパティ名を追加することもできます。これを行うためには、**[新規作成]** をクリックします。**[新規プロパティ]** ダイアログ・ボックスが開きます。有効なプロパティを、attribute/ <プロパティ名>の形式で入力して、**[OK]** をクリックします。新規プロパティが**[補足プロパティ]** リストに追加されます。たとえば、MyColor というプロパティを追加するには、attribute/MyColor と入力します。

- 9 **[OK]** をクリックし、**[プロパティの追加と削除]** ダイアログ・ボックスを閉じます。プロパティは**[補足プロパティ]** リストに表示されます。



- 10 上向き矢印と下向き矢印を使って、補足プロパティの順序を指定します。オブジェクトを学習するときに一意のオブジェクト記述を作成するために補足プロパティが必要な場合、QuickTest は、一意の記述を作成するのに十分な情報が得られるまで、**[補足プロパティ]** リストでの順序に従って、記述に補足プロパティを1つずつ追加していきます。

序数識別子の選択

[オブジェクトの認識] ダイアログ・ボックスで指定した必須プロパティと補足プロパティを学習するのに加え、QuickTest は予備的に各テスト・オブジェクトの序数識別子も学習できます。「序数識別子」は、同じ記述を持つオブジェクト（必須および補足プロパティ・リストで指定されているすべてのプロパティの値が同じオブジェクト）を区別するために、他のオブジェクトとの相対的な順番を表す数値を割り当てます。この順位の値によって QuickTest は、必須および補足プロパティでは一意の記述を作成するのに不十分な場合でも、一意の記述を作成できます。

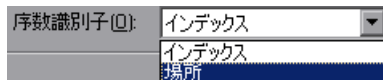
割り当てられた序数プロパティの値は、相対的な値であり、QuickTest がオブジェクトを学習するときに表示されていた他のオブジェクトとの相対関係に基づいています。アプリケーションのページやスクリーンのレイアウトや構成が変われば、オブジェクト自体には一切変化がなくても、この値が変わることがあります。そのため、QuickTest はすべての利用可能な必須および補足プロパティを使っても一意の記述を作成できない場合にだけ、この予備的な序数識別子の値を学習します。

さらに、QuickTest は序数識別子を学習しても、実行セッション中に序数識別子を使うのは、学習した記述およびスマート認識メカニズムを使ってアプリケーションのオブジェクトを十分に識別できない場合だけです。QuickTest が他のテスト・オブジェクト・プロパティを使用して実行セッション中にオブジェクトを識別できれば、序数識別子は無視されます。

QuickTest がオブジェクトの識別に使用できる序数識別子のタイプは次のとおりです。

- ▶ **[インデックス]**：オブジェクトがアプリケーション・コードの中に出現する順序を、それ以外は同じ記述を持つ他のオブジェクトとの相対関係で表します。詳細については、376 ページ「インデックス・プロパティを使用したオブジェクトの識別」を参照してください。
- ▶ **[場所]**：親ウィンドウ、フレーム、またはダイアログ・ボックス内においてオブジェクトが出現する順序を、その他の記述が同じであるほかのオブジェクトとの相対位置で表します。詳細については、377 ページ「場所プロパティを使用したオブジェクトの識別」を参照してください。
- ▶ **[CreationTime]**：(Browser オブジェクトのみ) 同じ記述を持つブラウザが開いた相対的な順番を表します。詳細については、378 ページ「CreationTime プロパティを使用したオブジェクトの識別」を参照してください。

標準設定では、テスト・オブジェクト・クラスごとに序数識別子があります。標準設定の序数識別子を変更するには、**[序数識別子]** ボックスから、目的のタイプを選択します。



ヒント：QuickTest は、記録に必須および補足プロパティを使って一意のテスト・オブジェクト記述を作成できた場合には、序数識別子は学習しません。[オブジェクトのプロパティ] または [オブジェクトリポジトリ] ダイアログ・ボックスの序数識別子ダイアログを使用して、後からオブジェクトのテスト・オブジェクト・プロパティに序数識別子を追加できます。詳細については、第13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。

インデックス・プロパティを使用したオブジェクトの識別

オブジェクトの学習中、QuickTest はオブジェクトを一意に識別できるように、オブジェクトの **インデックス・プロパティ** に値を割り当てることができます。この値は、ソース・コード内のオブジェクトの順番に基づいています。最初の番号は0です。

インデックス・プロパティ値は、各オブジェクトに固有の値です。したがって、ある WebEdit テスト・オブジェクトを記述するのに **Index:=3** を使用すると、QuickTest はページ内の4番目の WebEdit を探します。一方、WebElement オブジェクトを記述するのに **Index:=3** を使用すると、WebElement オブジェクトはすべての Web オブジェクトに適用されるため、QuickTest はタイプに関係なくページ内の4番目の Web オブジェクトを探します。

たとえば、次のオブジェクトを含んだ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

次のステートメントは、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")
```

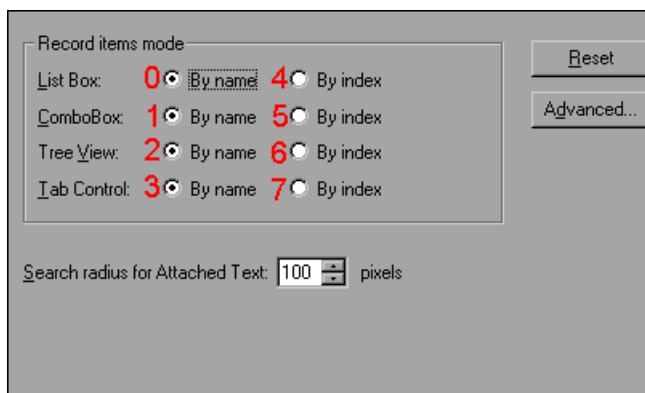
一方、次のステートメントは、リストの 2 番目の項目を表します。ページ内で **UserName** という名前を持つ最初の任意のタイプ (**WebElement**) のオブジェクトだからです。

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

場所プロパティを使用したオブジェクトの識別

オブジェクトの学習中、QuickTest はオブジェクトを一意に識別できるように、オブジェクトの **場所** プロパティに値を割り当てることができます。この値は、ウィンドウ、フレーム、ダイアログ・ボックス内に現れる同一のプロパティを持つ他のオブジェクトとの相対的な順番に基づいて決まります。最初のオブジェクトの場合、値は 0 です。値はカラム内で上から下に、そして左から右への順序で割り当てられます。

次の例では、ダイアログ・ボックス内のラジオ・ボタンは、位置のプロパティに従って番号が付けられています。



場所 プロパティ値はオブジェクト固有の値です。したがって、ある WinButton テスト・オブジェクトを記述するのに **Location:=3** を使用すると、QuickTest は 4 番目の WinButton をページ内の上から下、左から右に探します。一方、WinObject オブジェクトを記述するのに **Location:=3** を使用すると、WinObject オブジェクトはすべての標準オブジェクトに適用されるため、QuickTest はタイプに関係なくページ内の 4 番目の標準オブジェクトをページ内の上から下、左から右に探します。

たとえば、次のオブジェクトを含んだダイアログ・ボックスがあるとします。

- ▶ OK という名前のボタン・オブジェクト
- ▶ Add/Remove という名前のボタン・オブジェクト
- ▶ Add/Remove という名前のチェック・ボックス・オブジェクト
- ▶ Help という名前のボタン・オブジェクト
- ▶ Check spelling という名前のチェック・ボックス・オブジェクト

次のステートメントは、リストの3番目の項目を表します。ページ内で **Add/Remove** という名前を持つ最初のチェック・ボックス・オブジェクトだからです。

```
WinCheckBox("Name:=Add/Remove", "Location:=0")
```

一方、次のステートメントは、リストの2番目の項目を表します。ページ内で **Add/Remove** という名前を持つ最初の任意のタイプ (**WinObject**) のオブジェクトだからです。

```
WinObject("Name:=Add/Remove", "Location:=0")
```

CreationTime プロパティを使用したオブジェクトの識別

Browser オブジェクトの学習中、テスト・オブジェクトの記述によってオブジェクトを一意に特定できない場合、**QuickTest** は **CreationTime** テスト・オブジェクト・プロパティに値を割り当てます。この値は、同じ記述を持つブラウザが開いた相対的な順番を示します。最初に開いたブラウザは、**CreationTime** = 0 の割り当てとなります。

実行セッション中、テスト・オブジェクトの記述だけに基づいて **Browser** オブジェクトを一意に識別できないとき、**QuickTest** はブラウザが開いた順番を確認し、**CreationTime** プロパティを使用して正しい **Browser** オブジェクトを識別します。

たとえば、それぞれ 9:01 pm, 9:03 pm, 9:05 pm と、開いた時間だけが違う同一の3つのブラウザを対象にコンポーネントを記録する場合、**QuickTest** は 9:01 pm のブラウザに **CreationTime** = 0 を、9:03 pm のブラウザに **CreationTime** = 1 を、9:05 pm のブラウザに **CreationTime** = 2 を割り当てます。

10:30 pm にコンポーネントを実行したときに、ブラウザが 10:31 pm, 10:33 pm, 10:34 pm に開いたとします。QuickTest は 10:31 pm のブラウザが CreationTime = 0 のブラウザ・テスト・オブジェクト, 10:33 pm のブラウザが CreationTime = 1 のブラウザ・テスト・オブジェクト, 10:34 pm のブラウザが CreationTime = 2 のブラウザ・テスト・オブジェクトであると識別します。

開いているブラウザがいくつかある場合、CreationTime の最も低いものが最初に開いたものであり、最も高いものが最後に開いたものになります。たとえば、3 つ以上のブラウザが開いている場合、CreationTime = 2 のブラウザは 3 番目に開いたブラウザです。セッションの記録中に 7 つのブラウザが開いている場合、CreationTime = 6 のブラウザが最後に開いたブラウザです。

特定の CreationTime 値のブラウザを対象にステップが記録され、実行セッション中にこの CreationTime 値を持つブラウザが開いていない場合、ステップは CreationTime 値の最も高いブラウザで実行されます。たとえば、ステップが CreationTime = 6 のブラウザを対象に記録され、CreationTime = 0 と CreationTime = 1 という 2 つのブラウザだけが実行セッション中に開いているとすると、ステップは最後に開いたブラウザ（この例では CreationTime = 1）で実行されます。

注：セッション中の特定の時間に使用できる CreationTime 値は連番になっていない可能性があります。たとえば、記録または実行セッション中に 6 つのブラウザを開いたとして、セッション中にそのうち 2 番目と 4 番目に開いたブラウザ（CreationTime 値 1 と 3）を閉じたとすると、セッションの最後で開いているブラウザは CreationTime 値が 0, 2, 4, 5 のブラウザになります。

スマート認識の有効化と無効化

特定のテスト・オブジェクト・クラスの **[スマート認識を有効にする]** チェック・ボックスを選択すると、QuickTest によってオブジェクトの基本フィルタ・プロパティとオプション・フィルタ・プロパティの一方または両方で指定されているすべてのプロパティ値が学習されます。

標準では、一部のテスト・オブジェクトはすでにスマート認識が設定されており、他は設定されていません。標準で設定されているものは、**[スマート認識を有効にする]** チェック・ボックスも標準で選択されています。

スマート認識設定が定義されているテスト・オブジェクト・クラスのみスマート認識メカニズムを有効にします。しかし、特定のテスト・オブジェクト・クラスを対象としたスマート認識設定を定義した場合でも、スマート認識プロパティ値を学習したくないこともあります。スマート認識プロパティを学習しない場合は、**[スマート認識を有効にする]** チェック・ボックスをクリアします。

注：特定のオブジェクトのスマート認識プロパティを学習するように設定してある場合でも、[オブジェクトのプロパティ] または [オブジェクトリポジトリ] ダイアログ・ボックスで、特定のオブジェクトに対するスマート認識機能の使用を無効にできます。また、詳細については、第13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。

ただし、スマート認識プロパティを学習しなければ、オブジェクトに対して後でスマート認識メカニズムを有効にすることはできません。

スマート認識メカニズムの詳細については、382 ページ「スマート認識の設定」を参照してください。

テスト・オブジェクトの標準設定のオブジェクト認識設定の復元

ロードされているすべての環境、現在の環境のみ、または選択したテスト・オブジェクトの、オブジェクトの認識設定およびスマート認識プロパティ設定を、標準の設定に戻すことができます。

リセットできるのは組み込みオブジェクトのプロパティのみです。標準の Windows 環境をリセットすると、ユーザ定義のオブジェクトは削除されます。

注： [オブジェクトの認識] ダイアログ・ボックスの [環境] ボックスには、現在ロードされている環境のみが一覧表示されます。

標準では [テスト オブジェクトを元に戻す] ボタンが表示されますが、下矢印をクリックして、次のいずれかのオプションを選択することができます。

- ▶ [テスト オブジェクトを元に戻す]：選択したテスト・オブジェクトの設定をリセットして、システム標準の値に戻します。
- ▶ [環境を元に戻す]：現在の環境のすべてのテスト・オブジェクトの設定をリセットして、システム標準の値に戻します。
- ▶ [すべて元に戻す]：現在ロードされている環境のすべての設定をリセットして、システム標準の値に戻します。

オブジェクト認識設定用の自動スクリプトの生成

[スクリプトの生成] ボタンをクリックして、現在のオブジェクトの認識設定を含む自動スクリプトを生成できます。詳細については 665 ページ「QuickTest 操作のオートメーション」を参照するか、『QuickTest オートメーション・オブジェクト・モデル・リファレンス』（[ヘルプ] > [QuickTest オートメーション オブジェクト モデル リファレンス]）を参照してください。

スマート認識の設定

スマート認識プロパティを設定すれば、学習したオブジェクト記述に含まれるプロパティの一部が変更されている場合に、QuickTest がアプリケーションのオブジェクトを識別するのに役立ちます。

QuickTest は、学習した記述を使ってオブジェクトを識別するとき、記述中のすべてのプロパティ値と一致するオブジェクトを検索します。ほとんどの場合、この記述はオブジェクトを識別する最も簡単な方法です。そしてオブジェクトの主要なプロパティが変更されない限り、この方法は有効です。

QuickTest が学習したオブジェクト記述に一致するオブジェクトを見つけられない場合や、複数のオブジェクトが記述に適合する場合、QuickTest は学習した記述を無視し、スマート認識メカニズムを使ってオブジェクトの識別を試みます。

スマート認識メカニズムはもう少し複雑ですが、柔軟性が高いので、スマート認識定義を適格に設定すれば、学習した記述では識別できないときに、QuickTest がオブジェクト（存在していれば）を識別するのに役立ちます。

スマート認識メカニズムは次の2タイプのプロパティを使います。

- ▶ **[基本フィルタのプロパティ]** は、特定のテスト・オブジェクト・クラスの最も基本的なプロパティです。その値は、元のオブジェクトの根本的な部分を変えなければ変わりません。たとえば、Web のリンクのタグが <A> から何か別の値に変わった場合には、もはやそれを同じオブジェクトとは呼べません。
- ▶ **[オプションフィルタのプロパティ]** : 特定のクラスのオブジェクトを識別するのに役立つ別のプロパティです。これらのプロパティは、通常は変わらないとみなされますが、適用できなくなった場合には無視できます。

スマート認識の処理過程について

QuickTest が実行セッション中にスマート認識メカニズムに切り替わると（学習した記述ではオブジェクトを識別できなかったため）、スマート認識は次のプロセスでオブジェクトを識別します。

- 1 QuickTest は学習したテスト・オブジェクト記述を「忘れ」、[基本フィルタのプロパティ] リストに含まれているすべてのプロパティに適合するオブジェクト（親オブジェクト内のオブジェクト）を含んだ新しいオブジェクト候補リストを作成します。

- 2 QuickTest は、[オプションフィルタのプロパティ] リストの最初のプロパティに適合しないオブジェクトをオブジェクト候補リストからすべて除外します。残りのオブジェクトが新しいオブジェクト候補リストになります。
- 3 QuickTest が新しいオブジェクト候補リストを評価します。
 - ▶ 新しいオブジェクト候補リストに、まだ複数のオブジェクトがある場合、QuickTest はこの新しい（より小さい）オブジェクト候補リストを使って、リスト中の次のオプション・フィルタ・プロパティを使って手順 2 を繰り返します。
 - ▶ 新しくできたオブジェクト候補リストが空の場合、QuickTest はこのオプション・フィルタ・プロパティを無視し、前のオブジェクト候補リストに戻って手順 2 をリストの次のオプション・フィルタ・プロパティを使って繰り返します。
 - ▶ オブジェクト候補リストにオブジェクトが 1 つだけ含まれている場合、QuickTest はそれが識別されたオブジェクトであると判断し、そのオブジェクトを含んでいるステートメントを実行します。
- 4 QuickTest は手順 2 と 3 で説明した処理を、1 つのオブジェクトを識別するか、オプション・フィルタ・プロパティを使い果たすまで実行し続けます。

スマート認識の除外処理完了後も、QuickTest がまだオブジェクトを識別できない場合は、QuickTest は学習した記述に加え、序数識別子を使ってオブジェクトを識別します。

学習したスクリプトと序数識別子の組み合わせでもオブジェクトを識別するのに不十分な場合は、QuickTest は実行セッションを中止し、実行エラー・メッセージを表示します。

テスト結果に含まれるスマート認識情報の参照

学習した記述では QuickTest が指定されたオブジェクトを一度で識別できず、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識別を試みます。

QuickTest が、学習した記述では一致するオブジェクトを見つけられず、スマート認識でオブジェクトを見つけるのに成功した場合、テスト結果は警告ステータスを受け取り、スマート認識メカニズムが使用されたことを示します。

スマート認識メカニズムでオブジェクトを識別できない場合、QuickTestは学習した記述に加え、序数識別子を使ってオブジェクトを識別します。それでもオブジェクトを識別できない場合は、コンポーネントは失敗し、結果に通常の失敗ステップが表示されます。

詳細については、520 ページ「テスト結果に含まれるスマート認識情報の分析」を参照してください。

スマート認識の処理過程の例

以下では、あるオブジェクトの認識の課程を見ていきます。

コンポーネント中に次のステートメントがあるものとします。

```
Browser("Mercury Tours").Page("Mercury Tours").Image("Login").Click 22.17
```

コンポーネントを作成したとき、QuickTestはLogin画像について次のオブジェクト記述を学習しました。

名前	値
記述プロパティ	
image type	Image Button
html tag	INPUT
alt	Sign-In

しかし、コンポーネントを作成した後で、ページに2つ目のログイン・ボタン（WebサイトのVIPセクションにログインするためのもの）が追加されたため、Webデザイナーは元のログイン・ボタンのaltタグをbasic loginに変えました。

Web Image オブジェクトの標準の記述（alt, html tag, image type）はサイト内のほとんどの画像に使えますが、もはやログインの画像には使えません。その画像のaltプロパティが学習した記述とは一致しないからです。したがって、コンポーネントを実行すると、QuickTestは学習した記述に基づいてログイン・ボタンを識別することができません。しかし、QuickTestはスマート認識定義を使って、ログイン・ボタンをうまく識別できました。

下の例では、QuickTest がスマート認識を使って Login オブジェクトを見つける課程を示します。

- 1 Web 画像オブジェクトに対するスマート認識定義によれば、QuickTest は Login 画像に対するクリックを記録したとき、次のプロパティの値を学習しています。



学習した値は次のとおりです。

基本フィルタのプロパティ

プロパティ	値
html tag	INPUT

オプション・フィルタのプロパティ

プロパティ	値
alt	Login
image type	Image Button
name	login
file name	login.gif
class	<null>
visible	1

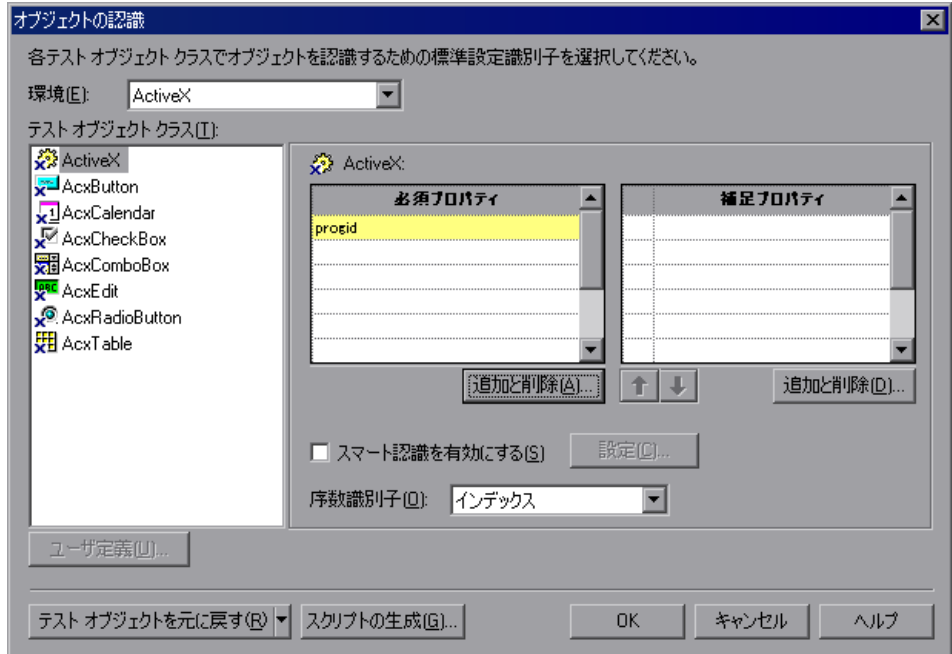
- 2 QuickTest は Mercury Tours ページの基本フィルタ・プロパティ定義 (**html tag = INPUT** および **image type = Image Button**) に適合する 5 つのオブジェクトを識別することによってスマート認識処理を開始します。QuickTest はこれらをオブジェクト候補と考え、**[オプション フィルタのプロパティ]** リストを使った確認を開始します。
- 3 QuickTest は各オブジェクト候補の **alt** プロパティを確認しますが、どれにも **alt** の値が **Login** ではありません。したがって、QuickTest はこのプロパティを無視し、次に移ります。
- 4 QuickTest は各オブジェクト候補の **name** プロパティを確認し、2 つのオブジェクト (基本および VIP ログイン・ボタン) に **name:login** があることを見つけます。QuickTest は他の 3 つのオブジェクトをリストから除外し、これら 2 つのログイン・ボタンが新しいオブジェクト候補になります。
- 5 QuickTest は残った 2 つのオブジェクト候補の **file name** プロパティを確認します。そのうちの 1 つだけにファイル名 **login.gif** があるので、QuickTest はログイン・ボタンを見つけたと正しく結論を出し、それをクリックします。

スマート認識定義のステップごとの設定

[オブジェクトの認識] ダイアログ・ボックスからアクセスできる [スマート認識プロパティ] ダイアログ・ボックスを使って、テスト・オブジェクト・クラスのスマート認識定義を設定できます。

スマート認識のプロパティを設定するには、次の手順を実行します。

- 1 [ツール] > [オブジェクトの認識] を選択します。[オブジェクトの認識] ダイアログ・ボックスが表示されます。

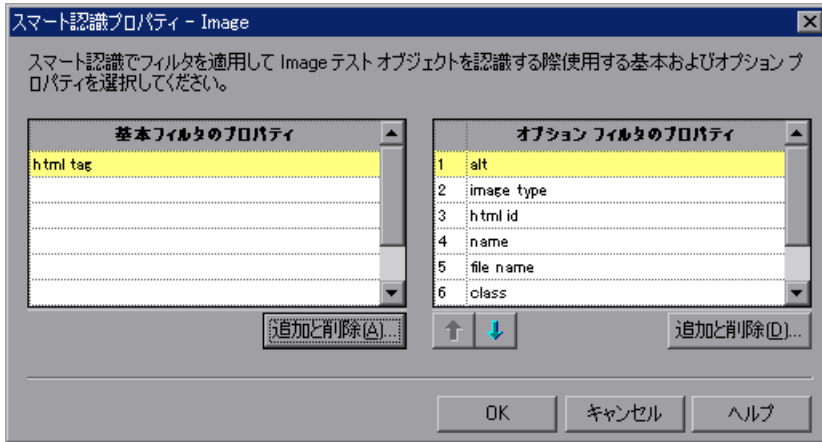


- 2 [環境] リストで適切な環境を選択します。選択した環境に関連付けられているテスト・オブジェクト・クラスが [テストオブジェクトクラス] リストに表示されます。

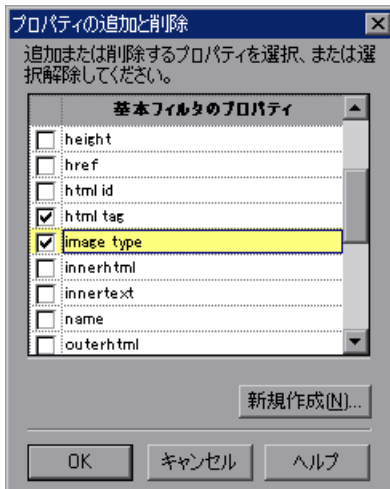
注: [環境] リストに含まれている環境は、読み込まれたアドイン環境に対応するものです。アドインの読み込みの詳細については、681 ページ「QuickTest アドインの使用法」を参照してください。

- 3 設定するテスト・オブジェクト・クラスを選択します。

- 4 [スマート認識を有効にする] チェック・ボックスの横にある [設定] ボタンをクリックします。[設定] ボタンは、[スマート認識を有効にする] オプションが選択されている場合のみ有効になります。[スマート認識プロパティ - Image] ダイアログ・ボックスが表示されます。



- 5 [基本フィルタのプロパティ] リストで、[追加と削除] をクリックします。基本フィルタ・プロパティのための [プロパティの追加と削除] ダイアログ・ボックスが表示されます。



- 6 **[基本フィルタのプロパティ]** リストに含めるプロパティを選択します。また、リストから削除するプロパティを消去します。

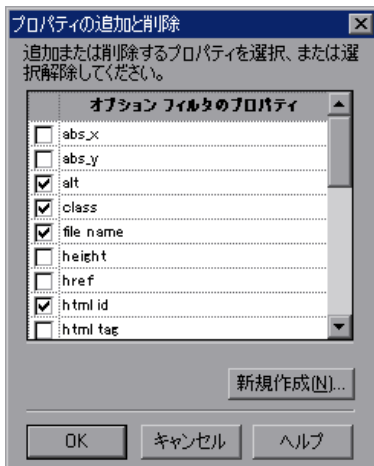
注： 同じプロパティを基本とオプションの両方のプロパティ・リストに含めることはできません。

[新規作成] をクリックして、表示されるダイアログ・ボックスに有効なプロパティ名を指定することにより、新規プロパティを指定できます。

ヒント： `attribute/ <プロパティ名>` の形式を使用して、Web オブジェクトに使用可能なプロパティの集合にプロパティ名を追加することもできます。これを行うためには、**[新規作成]** をクリックします。**[新規プロパティ]** ダイアログ・ボックスが開きます。有効なプロパティを、`attribute/ <プロパティ名>` の形式で入力して、**[OK]** をクリックします。新規プロパティが **[基本フィルタのプロパティ]** リストに追加されます。たとえば、MyColor というプロパティを追加するには、`attribute/MyColor` と入力します。

- 7 **[OK]** をクリックし、**[プロパティの追加と削除]** ダイアログ・ボックスを閉じます。更新された基本フィルタ・プロパティの集合が **[基本フィルタのプロパティ]** リストに表示されます。

- 8 [オプションフィルタのプロパティ] リストで、[追加と削除] をクリックします。オプション・フィルタ・プロパティのための [プロパティの追加と削除] ダイアログ・ボックスが表示されます。



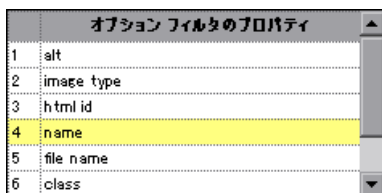
- 9 [オプションフィルタのプロパティ] リストに含めるプロパティを選択します。また、リストから削除するプロパティを消去します。

注：同じプロパティを基本とオプションの両方のプロパティ・リストに含めることはできません。

[新規作成] をクリックして、表示されるダイアログ・ボックスに有効なプロパティ名を指定することにより、新規プロパティを指定できます。

ヒント：attribute/ <プロパティ名>の形式を使用して、Web オブジェクトに使用可能なプロパティの集合にプロパティ名を追加することもできます。これを行うためには、[新規作成] をクリックします。[新規プロパティ] ダイアログ・ボックスが開きます。有効なプロパティを、attribute/ <プロパティ名>の形式で入力して、[OK] をクリックします。新規プロパティが [オプションフィルタのプロパティ] リストに追加されます。たとえば、MyColor というプロパティを追加するには、attribute/MyColor と入力します。

- 10 [OK] をクリックし、[プロパティの追加と削除] ダイアログ・ボックスを閉じます。プロパティは[オプションフィルタのプロパティ] リストに表示されます。



- 11 上向き矢印と下向き矢印を使って、オプション・プロパティの順序を指定します。QuickTest はスマート認識メカニズムを使うとき、オプション・プロパティに対する残りのオブジェクト候補を[オプションフィルタのプロパティ] で設定した順序に従って、オブジェクト候補が1つになるまで1つずつチェックします。

ユーザ定義のテスト・オブジェクト・クラスの割り当て

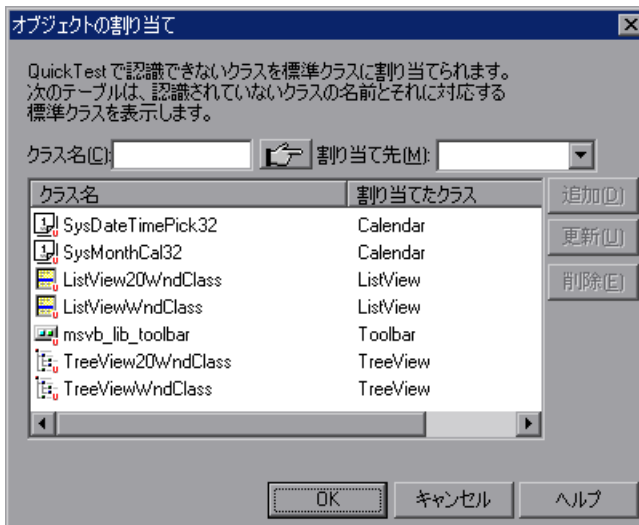
[オブジェクトの割り当て] ダイアログ・ボックスを使って未定義クラスまたはユーザ定義クラスを標準の Windows クラスに割り当てることができます。たとえば、アプリケーションに識別できないボタンがある場合、そのボタンは汎用の WinObject として学習されます。QuickTest に対して、そのオブジェクトが標準の Windows button クラスに属しているものとして識別するように指示できます。そのようにしておくと、記録中にそのボタンをクリックすると、QuickTest はその操作を標準の Windows ボタンをクリックしたのと同じように記録します。未定義オブジェクトまたはユーザ定義オブジェクトを標準オブジェクトに割り当てると、そのオブジェクトは標準の Windows テスト・オブジェクト・クラスのリストに、ユーザ定義オブジェクトとして追加されます。オブジェクトの認識設定を他のあらゆるオブジェクト・クラスと同様に、ユーザ定義オブジェクト・クラスに設定できます。

認識されないオブジェクトの割り当てを行うときは、同等の動作をする標準の Windows クラスにのみ割り当てるようにします。たとえば、ボタンと同等の動作をするオブジェクトを edit クラスに割り当ててはなりません。

注：ユーザ定義クラスを定義できるのは、[環境] ボックスで [Standard Windows] が選択されている場合だけです。

未定義クラスまたはユーザ定義クラスを標準の Windows クラスに割り当てるには、次の手順を実行します。

- 1 [ツール] > [オブジェクトの認識] を選択します。[オブジェクトの認識] ダイアログ・ボックスが表示されます。
- 2 [環境] ボックスで [Standard Windows] を選択します。[ユーザ定義] ボタンが有効になります。
- 3 [ユーザ定義] ボタンをクリックします。[オブジェクトの割り当て] ダイアログ・ボックスが表示されます。



- 4 指差しボタンをクリックしてから、ユーザ定義クラスに追加するクラスのオブジェクトをクリックします。ユーザ定義オブジェクトの名前が [クラス名] ボックスに表示されます。

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マウスオーバーなどの操作を実行したりするには、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、左の CTRL キーを押したまま、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。

- 5 **[割り当て先]** ボックスで、ユーザ定義オブジェクト・クラスを割り当てる対象となる標準オブジェクト・クラスを選択して **[追加]** をクリックします。クラス名と割り当てがオブジェクト割り当てリストに追加されます。
- 6 標準クラスにさらにオブジェクトを追加するにはオブジェクトごとに手順 4～5 を繰り返します。
- 7 **[OK]** をクリックします。[オブジェクトの割り当て] ダイアログ・ボックスが閉じ、オブジェクトが標準の Windows テスト・オブジェクト・クラスのリストにユーザ定義テスト・オブジェクトとして追加されます。追加したオブジェクトのアイコンの右下角には、ユーザ定義クラスであることを表す、赤い **U** の文字が入ります。
- 8 ユーザ定義オブジェクト・クラスのオブジェクトの認識設定は、他のあらゆるオブジェクト・クラスと同様に設定できます。詳細については、370 ページ「必須および補足記録プロパティの設定」および 382 ページ「スマート認識の設定」を参照してください。

既存の割り当てを変更するには、次の手順を実行します。

- 1 [オブジェクトの割り当て] ダイアログ・ボックスのオブジェクト割り当てリストで、変更するクラスを選択します。そのクラス名と現在の割り当てが、[クラス名] および [割り当て先] ボックスに表示されます。
- 2 選択したユーザ定義オブジェクト・クラスを割り当てる対象となる標準オブジェクト・クラスを選択して **[更新]** をクリックします。オブジェクト割り当てリストのクラス名と割り当てが更新されます。
- 3 **[OK]** をクリックし、[オブジェクトの割り当て] ダイアログ・ボックスを閉じます。

既存の割り当てを削除するには、次の手順を実行します。

- 1 [オブジェクトの割り当て] ダイアログ・ボックスのオブジェクト割り当てリストで、削除するクラスを選択します。
- 2 **[削除]** をクリックします。そのクラス名と割り当てが、[オブジェクトの割り当て] ダイアログ・ボックスのオブジェクト割り当てリストから削除されます。
- 3 **[OK]** をクリックします。[オブジェクトの割り当て] ダイアログ・ボックスが閉じ、[オブジェクトの認識] ダイアログ・ボックスの標準の Windows テスト・オブジェクト・クラスからクラス名が削除されます。

第 15 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・マネージャでは、オブジェクトの追加および定義、オブジェクトおよびその記述の変更、リポジトリの汎用性を高めるためのパラメータ化、リポジトリの保守および組織化、リポジトリの結合、XML 形式でのリポジトリのインポートおよびエクスポートなど、組織で使用されているすべての共有オブジェクト・リポジトリを一元管理することができます。

本章では、次の内容について説明します。

- ▶ 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・マネージャについて
- ▶ オブジェクト・リポジトリを使った作業
- ▶ オブジェクト・リポジトリの変更
- ▶ リポジトリ・パラメータを使用した作業
- ▶ テスト・オブジェクトの詳細の変更
- ▶ オブジェクトの検索
- ▶ 結合操作の実行
- ▶ インポートおよびエクスポート操作の実行

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・マネージャでは、共有オブジェクト・リポジトリの作成および保守ができます。ファイル・システムおよび Quality Center プロジェクトに保存されているオブジェクト・リポジトリを使用できます。

各オブジェクト・リポジトリには、QuickTest によるアプリケーション内のオブジェクトの識別を可能にする情報が含まれています。QuickTest により、テスト・オブジェクトに関するすべての情報を共有オブジェクト・リポジトリに格納することで、コンポーネントの再利用性を維持できます。アプリケーションのオブジェクトに変更があった場合、オブジェクト・リポジトリ・マネージャは、複数のコンポーネントのテスト・オブジェクト情報を1か所で集中して更新できる場所となります。

注：共有オブジェクト・リポジトリの代わりに、または共有オブジェクト・リポジトリに加えて、一部または全部のオブジェクトをコンポーネントごとにローカル・オブジェクト・リポジトリに保存するという選択肢もあります。ローカル・オブジェクト・リポジトリの詳細については、第13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。

同じコンポーネントに関連付けられているローカル・オブジェクト・リポジトリと共有オブジェクト・リポジトリの両方に、同じ名前と記述を持つオブジェクトが存在する場合は、そのコンポーネントに対するローカルのオブジェクト定義が使用されます。同じコンポーネントに関連付けられている複数の共有オブジェクト・リポジトリ内に同じ名前と記述を持つオブジェクトがある場合、共有オブジェクト・リポジトリがコンポーネントに関連付けられている順序に従って、最初に出現したオブジェクトのオブジェクト定義が使用されます。共有オブジェクト・リポジトリの関連付けの詳細については、94 ページ「共有オブジェクト・リポジトリの管理」を参照してください。

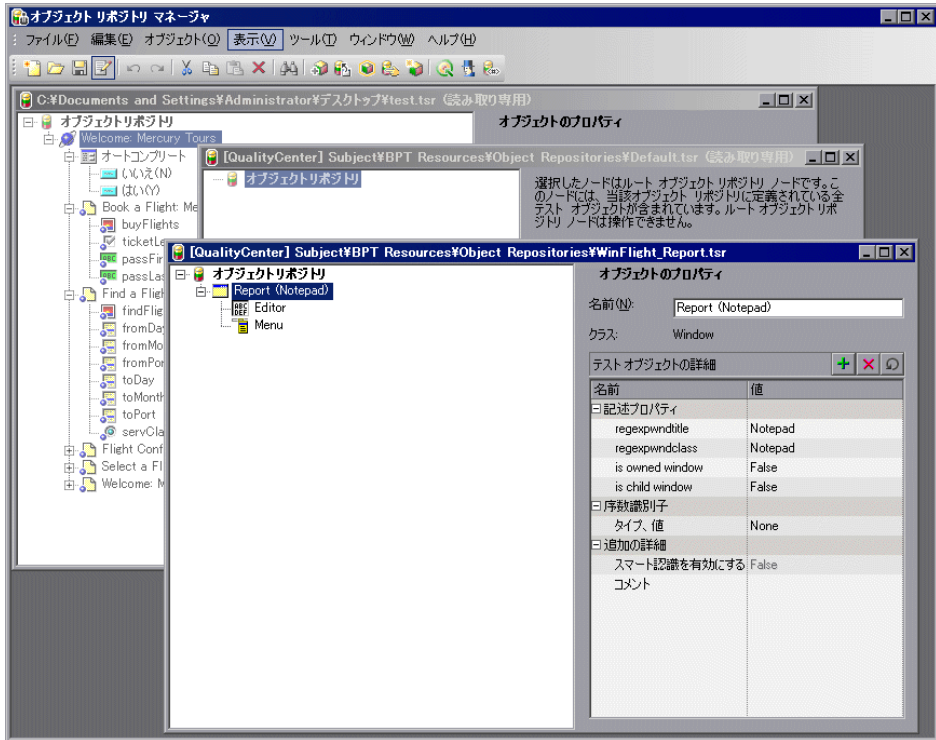
複数のコンポーネントで同じ共有オブジェクト・リポジトリを使用できます。1つのコンポーネントで複数のオブジェクト・リポジトリを使用することもできます。また、コンポーネントを使用してオブジェクトをローカル・オブジェクト・リポジトリに直接保存することもできます。これにより、オブジェクトはそのコンポーネントからのみアクセスできます。

アプリケーション内のオブジェクトのプロパティ値の中に、QuickTest によってオブジェクトの識別に使用されるプロパティ値と異なるものがあると、コンポーネントは失敗することがあります。そのため、アプリケーション内にあるオブジェクトのプロパティ値に変更があった場合、既存のコンポーネントを継続して使用するには、該当するオブジェクト・リポジトリ内にある該当するテスト・オブジェクトのプロパティ値を修正する必要があります。

本章で説明するように、共有オブジェクト・リポジトリのオブジェクトを変更するには、オブジェクト・リポジトリ・マネージャを使用します。ローカル・オブジェクト・リポジトリに格納されているオブジェクトを変更するには、[オブジェクトリポジトリ] ウィンドウを使用します。[オブジェクトリポジトリ] ウィンドウの詳細については、第 13 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。

オブジェクト・リポジトリ・マネージャについて

オブジェクト・リポジトリ・マネージャを開くには、[リソース] > [オブジェクトリポジトリマネージャ] を選択します。オブジェクト・リポジトリ・マネージャでは、複数の共有オブジェクト・リポジトリを開き、必要に応じてそれらを変更できます。オブジェクト・リポジトリは、ファイル・システムまたは Quality Center プロジェクトから開くことができます。



ヒント：オブジェクト・リポジトリ・マネージャを開いている間も、引き続き QuickTest の他のウィンドウで作業が行えます。

共有オブジェクト・リポジトリは、必要な数だけ開けます。共有オブジェクト・リポジトリは、それぞれ別のドキュメント・ウィンドウに開きます。開いたウィンドウは、必要に応じてサイズ変更、最大化、最小化して配置することで、共有オブジェクト・リポジトリ間でオブジェクトのコピー、ドラッグしての移動などができるほか、個別のオブジェクト・リポジトリを対象に操作が行えます。共有オブジェクト・リポジトリ・ウィンドウに表示される情報の詳細については、403 ページ「共有オブジェクト・リポジトリ・ウィンドウについて」を参照してください。











共有オブジェクト・リポジトリは、[共有オブジェクト リポジトリを開く] ダイアログ・ボックスから開きます。このダイアログ・ボックスの**[読み取り専用モードで開く]** チェック・ボックスは標準設定で選択されています。このチェック・ボックスをクリアすると、共有オブジェクト・リポジトリは編集可能なモードで開きます。それ以外の場合は、共有オブジェクト・リポジトリは読み取り専用モードで開きます。変更するには**[編集を有効化]** ボタンをクリックする必要があります。詳細については、412 ページ「オブジェクト・リポジトリの編集」を参照してください。








オブジェクト・リポジトリ・マネージャの中でメニュー項目を選択するかツールバー・ボタンをクリックすると、ウィンドウが現在アクティブな（フォーカスがある）共有オブジェクト・リポジトリを対象に、選択した操作が実行されます。ウィンドウのタイトル・バーに、共有オブジェクト・リポジトリの名前およびファイル・パスが表示されます。オブジェクト・リポジトリ・マネージャのツールバー・ボタンの詳細については、400 ページ「オブジェクト・リポジトリ・マネージャ・ツールバーの使用について」を参照してください。

オブジェクト・リポジトリ・マネージャの中で実行できる共有オブジェクト・リポジトリ操作の多くは、([オブジェクト リポジトリ] ウィンドウを使用して) ローカル・オブジェクト・リポジトリに格納されているオブジェクトに変更を加える場合と似ています。したがって、手順の多くは第13章「テスト・オブジェクトを使用した作業」で説明しています。手順のほとんどは、オブジェクト・リポジトリ・マネージャと [オブジェクト リポジトリ] ウィンドウで同じですが、ウィンドウとオプションに若干の相違がある場合があります。





オブジェクト・リポジトリ・マネージャ・ツールバーの使用について

よく実行する操作は、オブジェクト・リポジトリ・マネージャ・ツールバーから実行できます。オブジェクト・リポジトリ・マネージャ・ツールバーには、次のボタンがあります。

ボタン	詳細
	新規の共有オブジェクト・リポジトリを作成します。詳細については、405 ページ「新しいオブジェクト・リポジトリの作成」を参照してください。
	ファイル・システムまたは Quality Center から共有オブジェクト・リポジトリを開きます。詳細については、405 ページ「オブジェクト・リポジトリを開く」を参照してください。
	アクティブな共有オブジェクト・リポジトリをファイル・システムまたは Quality Center に保存します。詳細については、407 ページ「オブジェクト・リポジトリの保存」を参照してください。
	アクティブな共有オブジェクト・リポジトリを編集可能にすることで、共有オブジェクト・リポジトリを編集します。詳細については、412 ページ「オブジェクト・リポジトリの編集」を参照してください。
	アクティブな共有オブジェクト・リポジトリで行った1つ前の操作を元に戻します。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、351 ページ「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」を参照してください。
	アクティブな共有オブジェクト・リポジトリで元に戻した操作を再度実行します。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、351 ページ「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」を参照してください。
	アクティブな共有オブジェクト・リポジトリで選択されている項目またはオブジェクトを切り取ります。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、351 ページ「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」を参照してください。
	アクティブな共有オブジェクト・リポジトリで選択されている項目またはオブジェクトをクリップボードにコピーします。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、351 ページ「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」を参照してください。

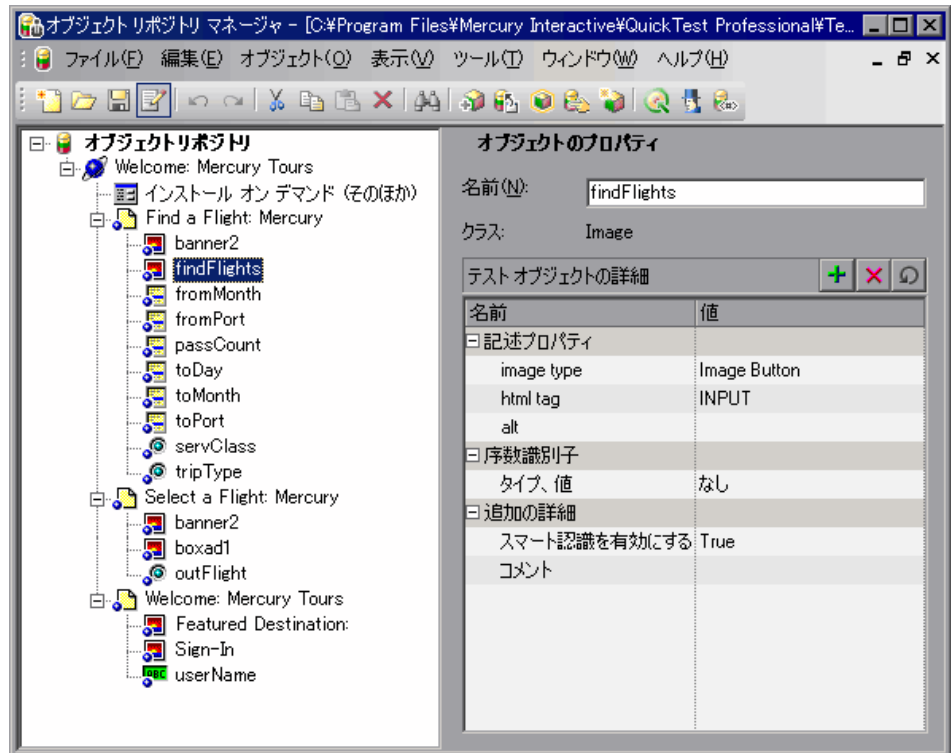
ボタン	詳細
	クリップボードのデータをアクティブな共有オブジェクト・リポジトリに貼り付けます。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、351 ページ「オブジェクト・リポジトリ内のオブジェクトのコピー、貼り付け、および移動」を参照してください。
	アクティブな共有オブジェクト・リポジトリで選択されている項目またはオブジェクトを削除します。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、354 ページ「オブジェクト・リポジトリからのオブジェクトの削除」を参照してください。
	アクティブな共有オブジェクト・リポジトリ内のオブジェクト、プロパティ、またはプロパティ値を検索します。指定したプロパティ値を検索して置換することもできます。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、355 ページ「オブジェクト・リポジトリ内のオブジェクトの検索」を参照してください。
	アクティブな共有オブジェクト・リポジトリにオブジェクトを追加します。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、343 ページ「オブジェクト・リポジトリへのオブジェクトの追加」を参照してください。
	アプリケーション内のオブジェクトの実際のプロパティに従って、アクティブな共有オブジェクト・リポジトリ内のテスト・オブジェクト・プロパティを更新します。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、324 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。
	アクティブな共有オブジェクト・リポジトリ内でオブジェクトを選択すると、それがアプリケーション内で強調表示されます。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、358 ページ「アプリケーション内のオブジェクトの強調表示」を参照してください。
	アプリケーション内でオブジェクトを選択すると、アクティブな共有オブジェクト・リポジトリ内でそれが強調表示されます。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、359 ページ「オブジェクト・リポジトリ内のオブジェクトの場所の特定」を参照してください。

第4部・テスト・オブジェクトおよびオブジェクト・リポジトリを使用した作業

ボタン	詳細
	アプリケーションに存在しないテスト・オブジェクトを定義し、アクティブな共有オブジェクト・リポジトリにそれを追加します。ローカル・オブジェクト・リポジトリでの場合と同じように行います。詳細については、349 ページ「新規テスト・オブジェクトの定義」を参照してください。
	Quality Center に接続し、Quality Center プロジェクトに格納されているオブジェクト・リポジトリ・ファイルを使って作業を行います。Quality Center には、QuickTest のメイン・ウィンドウから、またはオブジェクト・リポジトリ・マネージャから接続します。詳細については、20 ページ「Quality Center プロジェクトへの接続」を参照してください。
	オブジェクト・スパイを開き、アプリケーション内の実行時オブジェクトまたはテスト・オブジェクトのプロパティと値を表示します。詳細については、290 ページ「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の表示」を参照してください。
	アクティブな共有オブジェクト・リポジトリ内のリポジトリ・パラメータを追加、編集、および削除します。詳細については、414 ページ「リポジトリ・パラメータの管理」を参照してください。

共有オブジェクト・リポジトリ・ウィンドウについて

オブジェクト・リポジトリ・マネージャで開いた各共有オブジェクト・リポジトリは、個別のドキュメント・ウィンドウに表示されます。各共有オブジェクト・リポジトリのウィンドウには、オブジェクト・リポジトリ内のすべてのオブジェクトのツリーが、選択したオブジェクトのテスト・オブジェクト情報とともに表示されます。



[オブジェクトリポジトリ] ウィンドウには、ツリーの中で選択した各テスト・オブジェクトの情報が表示されます。共有オブジェクト・リポジトリ内の任意のテスト・オブジェクトのテスト・オブジェクト記述の表示、テスト・オブジェクトやそのプロパティの変更、共有オブジェクト・リポジトリへのオブジェクトの追加を行うことができます。詳細については、410 ページ「オブジェクト・リポジトリの変更」および 419 ページ「テスト・オブジェクトの詳細の変更」を参照してください。

各オブジェクト・リポジトリ・ウィンドウには、次の情報が表示されます。

情報	詳細
【オブジェクトリポジトリ】 ツリー	共有オブジェクト・リポジトリのすべてのテスト・オブジェクトが含まれています。
【名前】	QuickTest によって選択したテスト・オブジェクトに割り当てられている名前です。テスト・オブジェクトの名前は変更できます。詳細については、327 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。
【クラス】	選択したオブジェクトのクラスです。
【テストオブジェクトの詳細】	実行セッション中に、選択したオブジェクトの識別に使用されるプロパティおよびプロパティ値の表示や変更ができません。詳細については、419 ページ「テスト・オブジェクトの詳細の変更」を参照してください。

注：テスト・オブジェクトが含まれるステップをコンポーネントから削除しても、オブジェクトはオブジェクト・リポジトリから削除されません。共有オブジェクト・リポジトリのオブジェクトを削除するには、オブジェクト・リポジトリ・マネージャを使用して、ローカル・オブジェクト・リポジトリからオブジェクトを削除する場合とほぼ同じ方法で削除します。詳細については、を参照してください。

オブジェクト・リポジトリを使った作業

オブジェクト・リポジトリ・マネージャを使用して、オブジェクト・リポジトリを新規作成し、既存のオブジェクト・リポジトリを開いて変更し、終了時にはファイルを保存して閉じることができます。

新しいオブジェクト・リポジトリの作成

新しいオブジェクト・リポジトリを作成してオブジェクトを追加し、保存することができます。その後、QuickTest 内から、1 つ以上のコンポーネントをオブジェクト・リポジトリに関連付けることができます。共有オブジェクト・リポジトリの関連付けの詳細については、94 ページ「共有オブジェクト・リポジトリの管理」を参照してください。

新しいオブジェクト・リポジトリを作成するには、次の手順を実行します。



オブジェクト・リポジトリ・マネージャの中で、[ファイル] > [新規作成] を選択するか、[新規作成] ボタンをクリックします。新しいオブジェクト・リポジトリが開きます。これで、オブジェクトの追加や、オブジェクト・リポジトリの変更および保存ができます。詳細については、410 ページ「オブジェクト・リポジトリの変更」および 407 ページ「オブジェクト・リポジトリの保存」を参照してください。

オブジェクト・リポジトリを開く

既存のオブジェクト・リポジトリを開き、表示または変更できます。オブジェクト・リポジトリは、ファイル・システムまたは Quality Center プロジェクトから開くことができます。



Quality Center プロジェクトには、QuickTest またはオブジェクト・リポジトリ・マネージャから、[ファイル] > [Quality Center の接続] を選択するか、[Quality Center の接続] ボタンをクリックして接続します。Quality Center への接続の詳細については、20 ページ「Quality Center プロジェクトへの接続」を参照してください。

QuickTest の以前のバージョンをお使いのユーザの方へ：

以前のバージョンの QuickTest を使用して作成されたオブジェクト・リポジトリを開くとき、オブジェクト・リポジトリを編集する前に、QuickTest を使用して最新の形式への変換を行う必要があります。

オブジェクト・リポジトリに外部アドインのテスト・オブジェクトが含まれている場合、オブジェクト・リポジトリを現在の形式に変換するためには、該当するアドインがインストールされている必要があります。インストールされていない場合、オブジェクト・リポジトリは読み取り専用形式でのみ開けます。

オブジェクト・リポジトリを変換しない場合は、オブジェクト・リポジトリ・ファイルは読み取り専用形式で表示されます。いったんファイルを変換して保存すると、そのファイルは以前のバージョンの QuickTest で使用できなくなります。

オブジェクト・リポジトリを開くには、次の手順を実行します。



- 1 オブジェクト・リポジトリ・マネージャの中で、[ファイル] > [開く] を選択するか、[開く] ボタンをクリックします。[共有オブジェクトリポジトリを開く] ダイアログ・ボックスが表示されます。

注： Quality Center に接続している場合と、標準ファイル・システムを使用する場合とでは、表示されるダイアログ・ボックスが異なります。[共有オブジェクトリポジトリを開く] 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・ボックスの切り替えができます。

- 2 開く対象となるオブジェクト・リポジトリを選択し、[開く] または [OK] をクリックします (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開いているかによります)。オブジェクト・リポジトリが開きます。

標準設定では、オブジェクト・リポジトリは読み取り専用モードで開きます。編集可能な形式で開くには、[共有オブジェクトリポジトリを開く] ダイアログ・ボックスで **[読み取り専用モードで開く]** チェック・ボックスをクリアします。412 ページ「オブジェクト・リポジトリの編集」で説明している方法で、オブジェクト・リポジトリを編集することもできます。

オブジェクト・リポジトリが編集可能になっている場合は、オブジェクトの追加、オブジェクト・リポジトリの変更、および保存ができます。詳細については、410 ページ「オブジェクト・リポジトリの変更」および 407 ページ「オブジェクト・リポジトリの保存」を参照してください。

ヒント：オブジェクト・リポジトリは、[ファイル] メニューの **[最近使用したファイル]** のリストからも開けます。

オブジェクト・リポジトリの保存


オブジェクト・リポジトリを作成または変更したら、保存する必要があります。オブジェクト・リポジトリに変更を加えると、オブジェクト・リポジトリを保存するまで、タイトル・バーにアスタリスク (*) が表示されます。



オブジェクト・リポジトリは、ファイル・システムまたは Quality Center プロジェクト (Quality Center プロジェクトに接続している場合) に保存できます。共有オブジェクト・リポジトリをアプリケーション領域に関連付けてコンポーネントからアクセスできるようにするには、共有オブジェクト・リポジトリを Quality Center プロジェクトに保存する必要があります。Quality Center プロジェクトに接続するには、QuickTest またはオブジェクト・リポジトリ・マネージャから、[ファイル] > [Quality Center の接続] を選択するか、[Quality Center の接続] ボタンをクリックします。Quality Center への接続の詳細については、20 ページ「Quality Center プロジェクトへの接続」を参照してください。

注：オブジェクト・リポジトリに加えた変更はすべて、まだ変更内容を保存していなくても、変更した直後に、オブジェクト・リポジトリを使用している同じコンピュータ上で開いているすべてのコンポーネントにおいて自動的に更新されます。変更を保存せずにオブジェクト・リポジトリを閉じると、変更時に開いていたすべてのコンポーネントにおいて、反映されていた変更内容がロールバックされます。オブジェクト・リポジトリを変更したコンピュータ上でコンポーネントを開くと、コンポーネントは、関連付けられているオブジェクト・リポジトリに保存した変更内容で自動的に更新されます。保存した変更を、別のコンピュータ上で開いているコンポーネントまたはリポジトリに反映するには、そのコンポーネントまたはオブジェクト・リポジトリを開くかコンピュータ上で編集用にロックすることによって、変更を読み込む必要があります。

オブジェクト・リポジトリを保存するには、次の手順を実行します。

- 1 保存するオブジェクト・リポジトリがアクティブ・ウィンドウであることを確認します。
- 2  **[ファイル]** > **[保存]** を選択するか、**[保存]** ボタンをクリックします。ファイルがすでに保存されている場合は、変更が保存されます。ファイルがまだ保存されていない場合は、**[共有オブジェクトリポジトリの保存]** ダイアログ・ボックスが開きます。

注：Quality Center に接続している場合と、標準ファイル・システムを使用する場合とでは、表示されるダイアログ・ボックスが異なります。**[共有オブジェクトリポジトリを開く]** 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・ボックスの切り替えができます。

- 3 オブジェクト・リポジトリを保存するフォルダを選択します。
- 4 **[ファイル名]** または **[添付名]** ボックスに、オブジェクト・リポジトリの名前を入力します（ファイル・システムまたは Quality Center プロジェクトのどちらへ保存するかによります）。ファイルを識別しやすいように、分かりやすい名前を使用します。

注：オブジェクト・リポジトリ名には、「¥」、「/」、「:」、「*」、「"」、「?」、「<」、「>」、「|」の文字は使用できません。

- 5 **[保存]** または **[OK]** をクリックします（ファイル・システムまたは Quality Center プロジェクトのどちらへ保存するかによります）。QuickTest はオブジェクト・リポジトリを拡張子 **.tsr** を付けて指定された場所に保存し、オブジェクト・リポジトリの名前とパスをリポジトリ・ウィンドウのタイトル・バーに表示します。

オブジェクト・リポジトリを閉じる

オブジェクト・リポジトリの変更または使用が終了したら、オブジェクト・リポジトリを閉じます。ファイルを閉じると、他の人が使用したり変更したりできるように、ファイルのロックが自動的に解除されます。開いているすべてのオブジェクト・リポジトリを閉じることもできます。

注：QuickTest を終了すると、オブジェクト・リポジトリ・マネージャも終了します。変更内容をまだ保存していない場合は、オブジェクト・リポジトリ・マネージャを終了する前に保存するよう求められます。

オブジェクト・リポジトリを閉じるには、次の手順を実行します。

- 1 閉じる対象となるオブジェクト・リポジトリがアクティブ・ウィンドウであることを確認します。
- 2 **[ファイル]** > **[閉じる]** を選択するか、オブジェクト・リポジトリ・ウィンドウのタイトル・バーで **[閉じる]** ボタンをクリックします。オブジェクト・リポジトリが閉じ、ロックが自動的に解除されます。変更内容をまだ保存していない場合は、ファイルを閉じる前に保存するよう求められます。

開いているすべてのオブジェクト・リポジトリを閉じるには、次の手順を実行します。

[ファイル] > [すべてのウィンドウを閉じる] または、[ウィンドウ] > [Close All Windows] を選択します。開いているすべてのオブジェクト・リポジトリが閉じ、ロックが自動的に解除されます。変更内容をまだ保存していない場合は、ファイルを閉じる前に保存するよう求められます。

オブジェクト・リポジトリの変更

オブジェクト・リポジトリを初めて使用するための準備を整えるために、または、テスト・プロセス全体を通じてオブジェクト・リポジトリを更新するために、さまざまな方法でオブジェクト・リポジトリを変更することができます。共有オブジェクト・リポジトリのオブジェクトやオブジェクト・プロパティの追加および変更、オブジェクト・リポジトリ間でのオブジェクトのコピーまたは移動、階層内の別の場所へのオブジェクトのドラッグ、オブジェクトの削除、オブジェクト名の変更ができます。オブジェクト・リポジトリに変更を加えると、オブジェクト・リポジトリを保存するまで、タイトル・バーにアスタリスク (*) が表示されます。



ヒント：必要に応じて、[編集] > [元に戻す] や [編集] > [やり直し] オプションまたは [元に戻す] ボタンや [やり直し] ボタンを使用して、変更の取り消しや繰り返しを行うことができます。[元に戻す] および [やり直し] オプションは、アクティブなドキュメントを対象とします。オブジェクト・リポジトリを保存すると、保存前にファイルに対して行った操作の取り消しや、やり直しはできません。

オブジェクト・リポジトリを読み取り専用モードで開いた場合、変更を加えるためには、オブジェクト・リポジトリを編集可能にする必要があります。編集可能にすると、オブジェクト・リポジトリがロックされ、複数のユーザが同時に変更できないようになります。

注：オブジェクト・リポジトリに加えた変更はすべて、まだ変更内容を保存していなくても、変更した直後に、オブジェクト・リポジトリを使用している同じコンピュータ上で開いているすべてのコンポーネントにおいて自動的に更新されます。変更を保存せずにオブジェクト・リポジトリを閉じると、変更時に開いていたすべてのコンポーネントにおいて、反映されていた変更内容がロールバックされます。オブジェクト・リポジトリを変更したコンピュータ上でコンポーネントを開くと、コンポーネントは、関連付けられているオブジェクト・リポジトリに保存した変更内容で自動的に更新されます。保存した変更を、別のコンピュータ上で開いているコンポーネントまたはリポジトリに反映するには、そのコンポーネントまたはオブジェクト・リポジトリを開くかコンピュータ上で編集用にロックすることによって、変更を読み込む必要があります。

ヒント：共有オブジェクト・リポジトリは、別の共有オブジェクト・リポジトリと結合することでも変更できます。2つの共有オブジェクト・リポジトリを結合すると、両方のオブジェクト・リポジトリの内容を合わせた新しい共有オブジェクト・リポジトリが作成されます。ローカル・オブジェクト・リポジトリを共有オブジェクト・リポジトリに結合すると、共有オブジェクト・リポジトリが、ローカル・オブジェクト・リポジトリの内容で更新されます。詳細については、第 16 章「共有オブジェクト・リポジトリの結合」を参照してください。

共有オブジェクト・リポジトリが編集可能であり、アクティブ・ウィンドウであることを確認したら、ローカル・オブジェクト・リポジトリを変更するときと同じ方法で共有オブジェクト・リポジトリを変更します。詳細については、次を参照してください。

- ▶ 次の「オブジェクト・リポジトリの編集」
- ▶ 343 ページ「オブジェクト・リポジトリへのオブジェクトの追加」
- ▶ 351 ページ「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」
- ▶ 354 ページ「オブジェクト・リポジトリからのオブジェクトの削除」

オブジェクト・リポジトリの編集


標準設定では、オブジェクト・リポジトリは読み取り専用モードで開きます。編集可能な形式で開くには、開くときに [共有オブジェクトリポジトリを開く] ダイアログ・ボックスで **[読み取り専用モードで開く]** チェック・ボックスをクリアします。

オブジェクト・リポジトリを読み取り専用モードで開いた場合、変更を加えるためには、オブジェクト・リポジトリを編集可能にする必要があります。オブジェクト・リポジトリを表示するだけなら、あるいは、別のオブジェクト・リポジトリにオブジェクトをコピーするだけなら、編集可能にする必要はありません。

オブジェクト・リポジトリを編集可能にすると、他のユーザが変更できないようにオブジェクト・リポジトリがロックされます。オブジェクト・リポジトリを他のユーザから変更できるようにするには、ロックを解除する必要があります（編集モードを無効にするか、オブジェクト・リポジトリを閉じます）。オブジェクト・リポジトリがすでに別のユーザによってロックされている場合、読み取り専用形式で保存されている場合、またはオブジェクト・リポジトリを開くための権限がユーザにない場合は、編集可能にすることはできません。

QuickTest の以前のバージョンをお使いのユーザの方へ：以前のバージョンの QuickTest を使用して作成されたオブジェクト・リポジトリを編集する場合、オブジェクト・リポジトリを編集する前に、QuickTest を使用して最新の形式に変換する必要があります。変換しない場合は、オブジェクト・リポジトリ・ファイルは読み取り専用形式で表示されます。いったんファイルを変換して保存すると、ファイルは以前のバージョンの QuickTest で使用できなくなります。

オブジェクト・リポジトリを編集可能にするには、次の手順を実行します。

- 1 編集対象オブジェクト・リポジトリがアクティブ・ウィンドウであることを確認します。
- 2  [ファイル] > [編集を有効化] を選択するか、[編集を有効化] ボタンをクリックします。オブジェクト・リポジトリが編集可能になります。

リポジトリ・パラメータを使用した作業

リポジトリ・パラメータを使用すれば、特定のプロパティ値をパラメータ化するように指定しつつ、実際のパラメータ化の定義は、パラメータ化の対象となるテスト・オブジェクト・プロパティ値を含んでいるオブジェクト・リポジトリに関連付けられている各コンポーネントで行うことができます。

リポジトリ・パラメータは、動的に変化するオブジェクトを対象にコンポーネントを作成し、実行する場合に便利です。オブジェクトがアプリケーション内で頻繁に更新される場合、またはデータベースなどの動的コンテンツによってプロパティ値が設定される場合、オブジェクトは動的に変化することがあります。

たとえば、ローカライズされたアプリケーションにおいて、テキストのプロパティ値がユーザ・インタフェースの言語に応じて変化するボタンがある場合があります。リポジトリ・パラメータを使用して名前プロパティ値をパラメータ化した後、当該オブジェクト・リポジトリを使用する各コンポーネントにおいて、プロパティ値をどこから取得するかを指定できます。たとえば、このオブジェクト・リポジトリを使用するあるコンポーネントではプロパティ値をコンポーネント・パラメータから取得し、別のコンポーネントではローカル・パラメータから取得し、さらに別のコンポーネントでは定数値を使用するように指定できます。


特定のオブジェクト・リポジトリに対するリポジトリ・パラメータはすべて、[リポジトリ パラメータの管理] ダイアログ・ボックスを使用して定義します。各リポジトリ・パラメータは、任意で標準設定値を指定し、分かりやすい説明とともに定義します。詳細については、414 ページ「リポジトリ・パラメータの管理」を参照してください。

標準設定値が定義されていないリポジトリ・パラメータを持つオブジェクト・リポジトリを使用するコンポーネントを開くと、欠落リソース表示枠に、割り当てが必要なリポジトリ・パラメータが存在することを示す指示が表示されます。その場合は、そのコンポーネント内のリポジトリ・パラメータを必要に応じて割り当てることができます。また、標準設定値があるリポジトリ・パラメータの割り当てを行ったり、すでに割り当てのあるリポジトリ・パラメータの割り当てを変更したりもできます。リポジトリ・パラメータの割り当ての詳細については、187 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。

リポジトリ・パラメータの管理

[リポジトリ パラメータの管理] ダイアログ・ボックスでは、1つの共有オブジェクト・リポジトリを対象に、リポジトリ・パラメータの追加、編集、および削除ができます。

リポジトリ・パラメータを管理するには、次の手順を実行します。



- 1 パラメータを管理する対象となるオブジェクト・リポジトリがアクティブ・ウィンドウであることを確認します。
- 2  オブジェクト・リポジトリが読み取り専用形式の場合は、[ファイル] > [編集を有効化] を選択するか、[編集を有効化] ボタンをクリックします。オブジェクト・リポジトリが編集可能になります。



- 3 [ツール] > [リポジトリ パラメータの管理] を選択するか, [リポジトリ パラメータの管理] ボタンをクリックします。[リポジトリ パラメータの管理] ダイアログ・ボックスが表示されます。



[リポジトリ パラメータの管理] ダイアログ・ボックスには、次の情報およびオプションがあります。

オプション	詳細
[名前]	リポジトリ・パラメータを管理する対象となるオブジェクト・リポジトリの名前およびパスが表示されます。
	新しいリポジトリ・パラメータを追加します。詳細については、416 ページ「リポジトリ・パラメータの追加」を参照してください。
	現在選択されているリポジトリ・パラメータを削除します。詳細については、418 ページ「リポジトリ・パラメータの削除」を参照してください。
パラメータ・リスト	オブジェクト・リポジトリに現在定義されているリポジトリ・パラメータのリストが表示されます。パラメータ・リストでは、パラメータの標準設定の値および説明を直接変更できます。詳細については、418 ページ「リポジトリ・パラメータの変更」を参照してください。
[リポジトリの中で検索]	オブジェクト・リポジトリ・ツリー内で、選択されているリポジトリ・パラメータを使用している最初のテスト・オブジェクトを検索して強調表示します。このボタンを再度クリックすると、選択したパラメータの次の出現箇所を検索できます。

リポジトリ・パラメータの追加

[リポジトリ パラメータの追加] ダイアログ・ボックスでは、新しいリポジトリ・パラメータを定義できます。パラメータの標準設定値を指定したり、パラメータがコンポーネントのステップで使用されている場合に識別に役立つ分かりやすい説明を指定することもできます。

リポジトリ・パラメータを追加するには、次の手順を実行します。



- 1 [リポジトリ パラメータの管理] ダイアログ・ボックスの中で、[リポジトリ パラメータの追加] ボタンをクリックします。[リポジトリ パラメータの追加] ダイアログ・ボックスが表示されます。

The screenshot shows a dialog box titled '+ リポジトリ パラメータの追加'. It contains three text input fields: '名前:' (Name), '標準設定値:' (Default Value), and '詳細:' (Details). The '標準設定値:' field is pre-filled with the text '{デフォルト値なし}'. Below the fields are three buttons: 'OK', 'キャンセル', and 'ヘルプ'.

- 2 [名前] ボックスに、パラメータに対して分かりやすい名前を指定します。パラメータ名は英字で始める必要があり、英数字およびアンダスコアのみを使用できます。
- 3 [標準設定値] ボックスに、リポジトリ・パラメータに使用する標準設定の値を指定できます。この値は、このオブジェクト・リポジトリを使用するコンポーネントの値またはパラメータ・タイプに、リポジトリ・パラメータを割り当てなかった場合に使用されます。標準設定の値を指定しなかった場合、リポジトリ・パラメータは、この共有オブジェクト・リポジトリを使用するコンポーネントにおいて未割り当てとして表示されます。




ヒント：標準設定の値を指定した場合、それを後で削除するには、[リポジトリ パラメータの管理] ダイアログ・ボックスの中で該当するパラメータの [標準設定値] セルをクリックした後、[標準設定値をクリア] ボタンをクリックします。セル内に {デフォルト値なし} というテキストが表示されます。

- 4 [詳細] ボックスに、リポジトリ・パラメータの説明を入力できます。この説明は、コンポーネントの中でリポジトリ・パラメータを割り当てるときに、パラメータの識別に役立ちます。
- 5 [OK] をクリックして、[リポジトリ パラメータの管理] ダイアログ・ボックスのパラメータのリストにパラメータを追加します。

リポジトリ・パラメータの変更

[リポジトリ パラメータの管理] ダイアログ・ボックスでは、リポジトリ・パラメータの標準設定値または説明を直接変更できます。ただし、リポジトリ・パラメータの名前は変更できません。

リポジトリ・パラメータを変更するには、次の手順を実行します。

- 1 [リポジトリ パラメータの管理] ダイアログ・ボックスの中で対象パラメータを選択します。
- 2  標準設定の値を変更するには、対象パラメータの [標準設定値] セルをクリックします。標準設定の値は、新しい値を入力して変更することも、[標準設定値をクリア] ボタンをクリックして削除することもできます。標準設定の値を削除すると、セル内に {デフォルト値なし} というテキストが表示されます。標準設定の値を指定しない場合、リポジトリ・パラメータは、この共有オブジェクト・リポジトリを使用するコンポーネントにおいて未割り当てとして表示されます。



注：テキストを手作業で削除しても、標準設定の値は削除されません。標準設定の値は空の文字列になります。標準設定の値を削除する場合は、[標準設定値をクリア] ボタンをクリックする必要があります。


- 3 パラメータの説明を変更するには、対象パラメータの [詳細] セルをクリックして、必要な説明を入力します。

リポジトリ・パラメータの削除

リポジトリ・パラメータの定義は、不要になった場合は削除できます。テスト・オブジェクトの定義で使用されているリポジトリ・パラメータを削除すると、パラメータがなくなったにもかかわらず、テスト・オブジェクトのプロパティ値は当該パラメータに割り当てられたままです。したがって、リポジトリ・パラメータを削除する前に、パラメータがどのテスト・オブジェクト記述にも使用されていないことを確認する必要があります。使用されていると、これらのテスト・オブジェクトを使用するステップがあるコンポーネントを実行すると失敗します。

ヒント：[リポジトリ パラメータの管理] ダイアログ・ボックスの [リポジトリの中で検索] ボタンを使用すれば、リポジトリ・パラメータが使用されている場所を確認できます。


リポジトリ・パラメータを削除するには、次の手順を実行します。

- 1 [リポジトリ パラメータの管理] ダイアログ・ボックスの中で、パラメータ名の左側にある選択領域をクリックすることで、削除対象のリポジトリ・パラメータを選択します。
- 2  [リポジトリ パラメータを削除] ボタンをクリックします。選択したリポジトリ・パラメータが削除されます。

テスト・オブジェクトの詳細の変更

オブジェクト・リポジトリ・マネージャで開いている共有オブジェクト・リポジトリの [テストオブジェクトの詳細] 領域では、実行セッション中にオブジェクトの識別に使用するプロパティおよびプロパティ値の表示や変更ができます。

共有オブジェクト・リポジトリが編集可能であり、アクティブ・ウィンドウであることを確認したら、ローカル・オブジェクトを変更するときと同じ方法で、共有オブジェクト・リポジトリ内のオブジェクトのテスト・オブジェクトの詳細を変更します。

 注：必要に応じて、[編集] > [元に戻す] や [編集] > [やり直し] オプションまたは [元に戻す] ボタンや [やり直し] ボタンを使用して、変更の取り消しや繰り返しを行うことができます。[元に戻す] および [やり直し] オプションは、アクティブなドキュメントを対象とします。リポジトリを保存すると、保存前にファイルに対して行った操作の取り消しや、やり直しはできません。

オブジェクト・リポジトリ・マネージャを使用して、共有オブジェクト・リポジトリ内のテスト・オブジェクト記述のプロパティ値を指定します。共有オブジェクト・リポジトリにあるオブジェクトのプロパティ値の指定時に使用可能なオプションと、ローカル・リポジトリにあるオブジェクトのプロパティの指定時に使用可能なオプションは異なります。共有オブジェクト・リポジトリにあるオブジェクトのプロパティ値の指定の詳細については、420 ページ「プロパティ値の指定」を参照してください。

プロパティ値の指定

テスト・オブジェクト記述に含まれるプロパティの値を指定または変更できます。定数値を使用して値を指定することも（単純な値または正規表現を含んだ定数値のいずれか）、リポジトリ・パラメータを使用して値をパラメータ化することもできます。リポジトリ・パラメータの詳細については、413 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。

プロパティ値を指定するには、次の手順を実行します。

- 1 プロパティ値を指定する対象となるテスト・オブジェクトを選択します。
- 2 **[テストオブジェクトの詳細]** 領域の中で、対象プロパティの [値] セルをクリックします。
- 3 次のいずれかの方法でプロパティ値を指定します。
 - ▶ 単純な定数値を指定する場合は、[値] セルに値を入力します。[値] セルに定数値を指定した場合は、残りの手順は必要ありません。次の説明のように、[リポジトリ パラメータ] ダイアログ・ボックスの中で正規表現を使用して定数値を指定することもできます。



- ▶ リポジトリ・パラメータを使用して値をパラメータ化するには、[値] セルの中でパラメータ化ボタンをクリックします。[リポジトリ パラメータ] ダイアログ・ボックスが表示されます。

4 次のいずれかのオプションを選択して、プロパティの値を指定します。

- ▶ **[定数]** ラジオ・ボタンを選択して、定数値を指定します。**[テストオブジェクトの詳細]** 領域の [値] セルに、定数値を直接入力することもできます。定数値の中で正規表現を使用した場合は、**[正規表現]** チェック・ボックスを選択します。
- ▶ **[パラメータ]** ラジオ・ボタンを選択して、定義済みのリポジトリ・パラメータのリストからパラメータを選択します。パラメータに対して標準設定の値が定義されていれば、それも表示されます。

注：リポジトリ・パラメータを定義するには、[リポジトリ パラメータの管理] ダイアログ・ボックスを使用します。詳細については、414 ページ「リポジトリ・パラメータの管理」を参照してください。

- 5 [OK] をクリックして [リポジトリ パラメータ] ダイアログ・ボックスを閉じます。値をパラメータ化した場合は、次に示すように、[**テストオブジェクトの詳細**] 領域の [値] カラムにパラメータ名がアイコンとともに表示されます。それ以外の場合には、指定した定数値が [値] カラムに表示されます。

テストオブジェクトの詳細	
名前	値
記述プロパティ	
name	 <name>

オブジェクトの検索

オブジェクト・リポジトリ内にある特定のオブジェクトを検索するには、複数の方法があります。オブジェクトは種類に基づいて検索できます。たとえば、特定のエディット・ボックスを検索したり、アプリケーション内のオブジェクトをポイントすることで、リポジトリ内の該当するオブジェクトを自動的に強調表示したりできます。特定のプロパティ値をほかのプロパティ値に置換できます。たとえば、プロパティ値 `userName` を `user name` という値に置換できます。また、オブジェクト・リポジトリ内のオブジェクトを選択して、アプリケーション内で強調表示させることで、それがどのオブジェクトであるかを確認できます。

共有オブジェクト・リポジトリがアクティブ・ウィンドウであることを確認したら、ローカル・オブジェクト・リポジトリで行うのと同じ方法で、共有オブジェクト・リポジトリ内のオブジェクトを検索します。プロパティ値を置き換える場合は、オブジェクト・リポジトリが編集可能であることも確認する必要があります。

結合操作の実行

オブジェクト・リポジトリ結合ツールでは、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]** オプションを使用して (**[ツール]** > **[ローカル リポジトリから更新]**)、1 つ以上のコンポーネントを持つローカル・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。たとえば、特定のコンポーネントでオブジェクトをローカルに学習していて、オブジェクト・リポジトリを使用するコンポーネントでオブジェクトを使用できるように、それらを共有オブジェクト・リポジトリに追加することが考えられます。また、オブジェクト・リポジトリ結合ツールを使用して、2 つの共有オブジェクト・リポジトリを 1 つにすることもできます。

オブジェクト・リポジトリ結合ツールを開くには、オブジェクト・リポジトリ・マネージャで **[ツール]** > **[オブジェクト リポジトリ結合ツール]** を選択します。結合操作の実行と、ローカル・オブジェクトを使用したオブジェクト・リポジトリの更新の詳細については、第 15 章「オブジェクト・リポジトリの管理」を参照してください。

注： オブジェクト・リポジトリ結合ツールを開いている間は、オブジェクト・リポジトリ・マネージャを使用した作業は行えません。

インポートおよびエクスポート操作の実行

オブジェクト・リポジトリは、XML ファイルに対してインポートおよびエクスポートができます。XML は構造化された利用しやすい形式あるため、任意の XML エディタを使用してオブジェクト・リポジトリに変更を加えた後、それらを QuickTest にインポートし直すことができます。オブジェクト・リポジトリに必要な形式を知るには、保存したオブジェクト・リポジトリをエクスポートします。

ファイルのインポートおよびエクスポートは、ファイル・システムまたは Quality Center プロジェクト (QuickTest が Quality Center に接続されている場合) のどちらに対しても行うことができます。



Quality Center プロジェクトに接続するには、QuickTest またはオブジェクト・リポジトリ・マネージャから、[ファイル] > [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ます。Quality Center への接続の詳細については、20 ページ「Quality Center プロジェクトへの接続」を参照してください。

XML からのインポート

必要な形式を使用して作成された XML ファイルをオブジェクト・リポジトリとしてインポートできます。XML ファイルは、オブジェクト・リポジトリ・マネージャを使用して XML 形式にエクスポートしたオブジェクト・リポジトリ、または QuickTest Siebel Test Express や独自作成のユーティリティなどのツールを使用して作成した XML ファイルです。XML の構造および形式に準拠する必要があります。

ヒント：必要な XML の構造および形式を確認するには、既存の共有オブジェクト・リポジトリを XML ファイルにエクスポートし、参考にします。詳細については、425 ページ「XML へのエクスポート」を参照してください。

XML ファイルからインポートするには、次の手順を実行します。

- 1 [ファイル] > [XML からインポート] を選択します。[XML からインポート] 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。[XML からインポート] 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・ボックスの切り替えができます。

- 2 インポートする XML ファイルを選択し、[開く] または [OK] をクリックします（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開くかによります）。

- XML ファイルがインポートされると、サマリ・メッセージ・ボックスが開き、指定のファイルから正常にインポートされたオブジェクト数、パラメータ数、およびメタデータの数に関する情報が表示されます。
- [OK] をクリックし、メッセージ・ボックスを閉じます。インポートされた XML ファイルが新しいオブジェクト・リポジトリとして開きます。これで、必要に応じて変更を加えたり、オブジェクト・リポジトリとして保存したりできます。

XML へのエクスポート

オブジェクト・リポジトリの内容を XML ファイルにエクスポートできます。これにより、オブジェクト・リポジトリの内容を任意の XML エディタを使用して簡単に編集したり、アクセス可能な柔軟な形式で保存したりできます。

XML ファイルにエクスポートするには、次の手順を実行します。

- エクスポートするオブジェクト・リポジトリがアクティブ・ウィンドウであることを確認します。
- [ファイル] > [XML へエクスポート] を選択します。[XML へエクスポート] 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。[XML へエクスポート] 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・ボックスの切り替えができます。

- ファイルを保存する場所を選択してファイル名または添付名を指定し、[保存] または [OK] をクリックします（ファイル・システムまたは Quality Center プロジェクトのどちらに保存するかによります）。
- オブジェクト・リポジトリが指定の XML ファイルにエクスポートされると、サマリ・メッセージ・ボックスが開き、指定のファイルへ正常にエクスポートされたオブジェクト、パラメータ、およびメタデータの数に関する情報が表示されます。
- [OK] をクリックし、メッセージ・ボックスを閉じます。これで XML ファイルを開き、任意の XML エディタを使用して表示または変更できます。

第 16 章

共有オブジェクト・リポジトリの結合

QuickTest Professional では、オブジェクト・リポジトリ結合ツールを使用して、2つの共有オブジェクト・リポジトリを結合して単独の共有オブジェクト・リポジトリにすることができます。また、このツールを使用して、1つ以上のアクションのローカル・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・リポジトリに結合することもできます。

本章では、次の内容について説明します。

- ▶ 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
- ▶ 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
- ▶ ターゲット・リポジトリの保存

共有オブジェクト・リポジトリの結合について

QuickTest Professional では、オブジェクト・リポジトリ結合ツールを使用して、2つの共有オブジェクト・リポジトリの既存の資産を結合して単独の共有オブジェクト・リポジトリにすることができます。このツールを使用することで、2つの共有オブジェクト・リポジトリ・ファイル（それぞれ「**一次**」リポジトリおよび「**二次**」リポジトリと呼びます）を結合して3つ目の新しいリポジトリ（「**ターゲット**」リポジトリと呼びます）を作成できます。一次リポジトリと二次リポジトリにあるオブジェクトは自動的に比較され、オブジェクト間の矛盾の解決方法を定義した事前設定可能なルールに従って、ターゲット・リポジトリに追加されます。

結合処理の後、一次リポジトリと二次リポジトリにある元のオブジェクト（これらは変更されずに残っています）のほか、結合後のターゲット・オブジェクト・リポジトリにあるオブジェクトが、オブジェクト・リポジトリ結合ツールに視覚的に表示されます。矛盾のあったオブジェクトは強調表示されます。ターゲット・オブジェクト・リポジトリでオブジェクトを選択すると、その矛盾の詳しい説明が表示されます。オブジェクト・リポジトリ結合ツールには、矛盾ごとに、提示されている解決方法を維持するか、矛盾の解決方法を個別に変更するかを、必要に応じて決めることができる専用のオプションが用意されています。

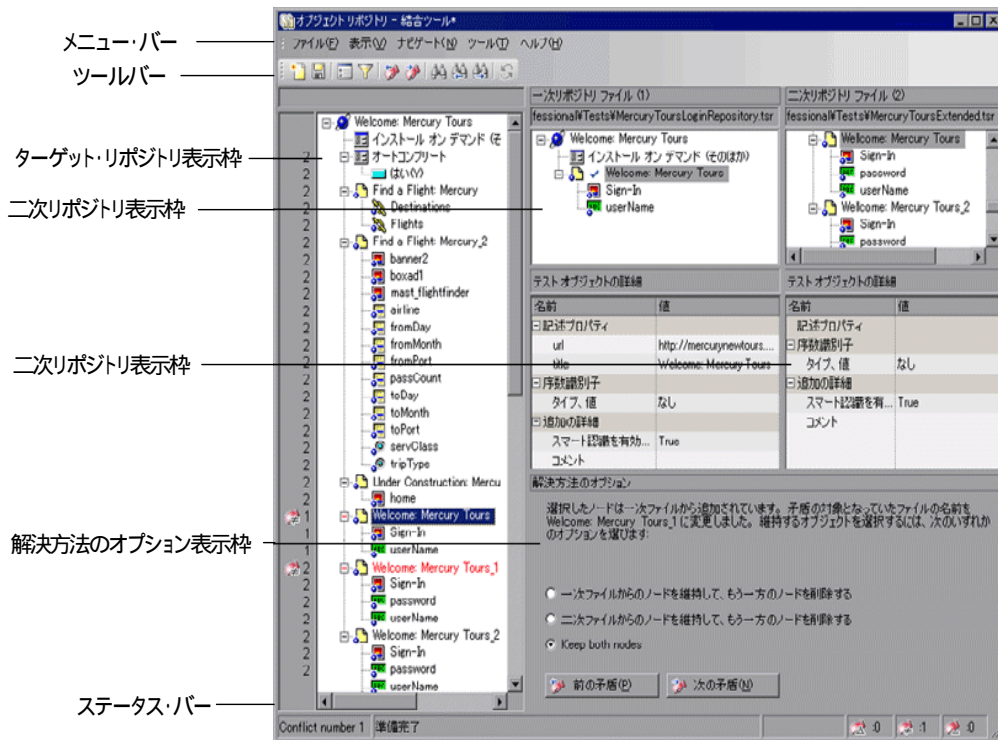
オブジェクト・リポジトリ結合ツールではまた、1つ以上のアクションのローカル・オブジェクト・リポジトリのオブジェクトを、共有オブジェクト・リポジトリに結合することもできます。たとえば、テスト内の特定のアクションの中でオブジェクトをローカルに学習した場合に、それらを共有オブジェクト・リポジトリに追加することで、そのオブジェクト・リポジトリを使用する他のテストのすべてのアクションでそれらのオブジェクトを使用できるようになります。

注：オブジェクト・リポジトリ結合ツールが開いている間は、オブジェクト・リポジトリ・マネージャを操作することはできません。オブジェクト・リポジトリ・マネージャの詳細については、第15章「オブジェクト・リポジトリの管理」を参照してください。

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を開くには、オブジェクト・リポジトリ・マネージャで [ツール] > [オブジェクトリポジトリ結合ツール] を選択します。

[オブジェクトリポジトリ - 結合ツール] ウィンドウの例を次に示します。



[結合ツール] ウィンドウには、次の主要な要素があります。

- ▶ **メニュー・バー**：オブジェクト・リポジトリ結合ツールのコマンドのメニューが表示されます。これらのコマンドについては本章の各所で説明します。メニュー・コマンドのショートカット・キーの詳細については、435 ページ「ショートカット・キーを使用したコマンドの実行」を参照してください。

- ▶ **ツールバー**：よく使用するメニュー・コマンドのボタンがあります。オブジェクト・リポジトリの結合、管理、および保存を行うことができます。ツールバー・ボタンの詳細については、435 ページ「ツールバー・コマンドの使用方法」を参照してください。
- ▶ **ターゲット・リポジトリ表示枠**：一次リポジトリおよび二次リポジトリから結合されたテスト・オブジェクトが表示されます。ターゲット・リポジトリ表示枠で選択したテスト・オブジェクトのプロパティが表示されるターゲットリポジトリオブジェクトのプロパティ表示枠は、表示と非表示を切り替えることもできます。詳細については、431 ページ「ターゲット・リポジトリ表示枠」を参照してください。
- ▶ **一次リポジトリ表示枠**：一次オブジェクト・リポジトリにあるテスト・オブジェクトが表示されます。詳細については、432 ページ「一次リポジトリ表示枠および二次リポジトリ表示枠」を参照してください。
- ▶ **二次リポジトリ表示枠**：二次オブジェクト・リポジトリにあるテスト・オブジェクトが表示されます。詳細については、432 ページ「一次リポジトリ表示枠および二次リポジトリ表示枠」を参照してください。
- ▶ **解決方法のオプション表示枠**：ターゲット・リポジトリ表示枠にあるオブジェクトについて、それらのソース、矛盾、および解決方法の詳細が表示され、矛盾があった場合に適用される解決方法を変更できます。詳細については、433 ページ「解決方法のオプション表示枠」を参照してください。
- ▶ **ステータス・バー**：ターゲット・リポジトリ表示枠の中で選択したオブジェクトのソース、矛盾、および解決方法の詳細のほか、アイコンの凡例が表示されます。詳細については、433 ページ「ステータス・バー」を参照してください。

ビューの変更

オブジェクト・リポジトリ結合ツールに表示されるビューを自分が作業しやすいように変更することができます。

- ▶ オブジェクト・リポジトリ結合ツール・ウィンドウ内で表示枠のサイズを変更するには、表示枠の縁をドラッグします。
- ▶ 結合ツール内でこれらの表示枠の表示と非表示を切り替えるには、**[表示]**メニューから **[一次リポジトリ]**、**[二次リポジトリ]**、**[ターゲットリポジトリオブジェクトのプロパティ]**、または **[解決方法のオプション]** を選択します。

- ▶ 現在のビューを、オブジェクト・リポジトリ結合ツールを開くたびに表示される標準設定のビューとして設定するには、**[表示] > [標準のレイアウトとして設定]** を選択します。変更を加えた後に画面を標準設定に戻すには、**[表示] > [標準のレイアウトを復元]** を選択します。

ターゲット・リポジトリ表示枠


ターゲット・リポジトリ表示枠には、一次および二次リポジトリから結合されたテスト・オブジェクトの階層と、それらのオブジェクトのプロパティと値が表示されます。オブジェクト階層の左側のカラムには、各オブジェクトのソース・ファイルが表示され（一次ファイルの場合は **1** と表示され、二次ファイルの場合は **2** と表示されます）、矛盾がある場合は矛盾のタイプを表すアイコンが表示されます。

ターゲット・オブジェクト・リポジトリを保存すると、そのファイル・パスがオブジェクト階層の上に表示されます。

注：オブジェクトのステータスをひと目で確認できるように、ターゲット・オブジェクト・リポジトリ内のオブジェクト名のテキストの色を、各オブジェクトのソースと、矛盾の原因となったかどうかに基づいて、設定することができます。詳細については、437 ページ「色の設定の指定」を参照してください。

ターゲット・リポジトリ表示枠には次の機能があります。

- ▶ ターゲット・オブジェクト・リポジトリ内のオブジェクトを選択すると、一次ソース・ファイル階層または二次ソース・ファイル階層、あるいはその両方にある対応するオブジェクトが探し出され、チェック・マークによって示されます。
- ▶ ターゲット・リポジトリ内のオブジェクトを選択すると、そのプロパティと値が、ターゲット・リポジトリ表示枠（**[表示] > [ターゲットリポジトリオブジェクトのプロパティ]**）の一番下にある **[オブジェクトプロパティ - ターゲットファイル]** 領域に表示されます。

- ▶ 結合の結果として矛盾が生じた場合は、ターゲット・オブジェクト・リポジトリ内で、矛盾のあるオブジェクトの左側にアイコンが表示されます。アイコンの上にポインタを置くと、矛盾のタイプを説明するツールチップが表示されます。
 - ▶ オブジェクトを右クリックすると、ショートカット・メニューが開きます。ここから選択できるオプションでは、ターゲット・オブジェクト・リポジトリの階層全体を展開または折りたたんだり、該当する場合には矛盾の解決方法とその結果を変更したりできます。
 - ▶ ノードをダブルクリックすると、ノードの階層を展開または折りたたむことができます。また、**[表示]**メニューの**[すべて折りたたみ]**または**[すべて開く]**を選択しても、ターゲット・オブジェクト・リポジトリの階層全体を展開または折りたたむことができます。
- 
- ▶ **[ナビゲート]**メニューの**[次の矛盾]**または**[前の矛盾]**を選択するか、ツールバーあるいは解決方法のオプション表示枠の**[次の矛盾]**または**[前の矛盾]**ボタンをクリックすると、ターゲット・オブジェクト・リポジトリ階層内の次の矛盾または前の矛盾に直接移動できます。
 - ▶ **[検索]** ダイアログ・ボックスを使用して、ターゲット・オブジェクト・リポジトリ内で1つ以上のオブジェクトを検索できます。詳細については、457ページ「特定のオブジェクトの検索」を参照してください。
 - ▶ **[表示]** > **[ターゲット リポジトリ オブジェクトのプロパティ]** を選択すると、ターゲット・リポジトリにあるオブジェクトのプロパティの表示と非表示を切り替えることができます。

一次リポジトリ表示枠および二次リポジトリ表示枠

一次リポジトリ表示枠および二次リポジトリ表示枠には、結合を行う元のソース・リポジトリにあるテスト・オブジェクトと、それらのプロパティおよび値が、階層表示されます。各オブジェクト階層の上にはファイル・パスが表示されます。

この表示枠には次の機能があります。

- ▶ 選択した項目をダブルクリックすると、その項目の階層を展開または折りたたむことができます。
- ▶ テスト・オブジェクトを該当する表示枠の中で選択すると、そのオブジェクトのプロパティと値が**[テストオブジェクトの詳細]**領域に表示されます。

- ▶ **[表示]** メニューの **[一次リポジトリ]** または **[二次リポジトリ]** を選択すると、表示枠の表示と非表示を切り替えることができます。

解決方法のオプション表示枠

解決方法のオプション表示枠には、ターゲット・オブジェクト・リポジトリで選択されているオブジェクトについて、結合中に発生した矛盾に関する情報が表示されます。また、標準設定の解決方法のオプションを使用して適用された矛盾の解決方法について、それを維持するか変更するかを決めるオプションもあります。

解決方法のオプション表示枠には次の機能があります。

- ▶ ターゲット・オブジェクト・リポジトリの中で矛盾のあるオブジェクトを選択すると、矛盾を説明するテキストと、オブジェクト・リポジトリ結合ツールによって採用される解決方法が、表示枠に表示されます。採用されている解決方法の代わりとなる他の解決方法の選択肢が用意されています。
- ▶ ラジオ・ボタンを選択することで、代わりとなる矛盾の解決方法を選択できます。変更を加えるたびに、ターゲット・オブジェクト・リポジトリが自動的に更新され、再表示されます。
- ▶ **[次の矛盾]** ボタンまたは **[前の矛盾]** ボタンをクリックすることで、ターゲット・リポジトリ階層内の次の矛盾または前の矛盾に直接移動できます。
- ▶ **[表示]** メニューの **[解決方法のオプション]** を選択またはクリアすることで、表示枠の表示と非表示を切り替えることができます。

ステータス・バー

ステータス・バーには、ターゲット・リポジトリ表示枠の中で選択されているオブジェクトについて、矛盾（存在する場合）の数と、ターゲット・リポジトリ表示枠で使用されているアイコンの凡例が表示されます。

次のアイコンがステータス・バー（およびターゲット・リポジトリ表示枠）に表示される場合があります。



- ▶ 類似記述の矛盾



- ▶ 同じ名前で記述が異なる矛盾



- ▶ 同じ記述で名前が異なる矛盾

アイコンの上にポインタを置くと、矛盾のタイプを説明するツールチップが表示されます。

矛盾のタイプの詳細については、450 ページ「オブジェクトの矛盾について」を参照してください。

ヒント：

ステータス・バーの矛盾アイコンをクリックすると、[統計情報] ダイアログ・ボックスが表示されます。詳細については、449 ページ「結合の統計情報の表示」を参照してください。



アイコンの左側にあるボックスをクリックすると、[フィルタ] ダイアログ・ボックスが表示されます。この領域には、フィルタが現在使用されている場合にフィルタ・アイコンが表示されます。詳細については、455 ページ「ターゲット・リポジトリ表示枠に対するフィルタの設定」を参照してください。

対応するショートカット・キーを押して、次の [ナビゲート] メニュー・コマンドを実行できます。

コマンド	ショートカット・キー	機能
[次の矛盾]	F4 キー	結合後のオブジェクト・リポジトリの中で次の矛盾オブジェクトを検索します。
[前の矛盾]	SHIFT+F4	結合後のオブジェクト・リポジトリの中で前の矛盾オブジェクトを検索します。
[検索]	CTRL+F	[検索] ダイアログ・ボックスを開きます。
[次を検索]	F3 キー	[検索] ダイアログ・ボックスの検索条件に従って、結合後のオブジェクト・リポジトリの中で次のオブジェクトを検索します。
[前を検索]	SHIFT+F3	[検索] ダイアログ・ボックスの検索条件に従って、結合後のオブジェクト・リポジトリの中で前のオブジェクトを検索します。

標準設定の定義

オブジェクト・リポジトリ結合ツールでは、オブジェクト・リポジトリの結合時に使用される設定があらかじめ定義されています。標準設定は次のとおりです。

- ▶ ターゲット・オブジェクト・リポジトリに表示されるオブジェクト名のテキストの色を指定します。
- ▶ オブジェクト・リポジトリ結合ツールが一次および二次リポジトリ内のオブジェクトの矛盾をどのように処理するのかを設定します。あるいは、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・リポジトリを更新するときに、ローカルおよび共有リポジトリ内のオブジェクトの矛盾をどのように処理するのかを設定します。

これらの設定をいつでも変更して、新しい標準設定を作成できます。設定を変更すると、以降のすべての新しい結合が新しい標準設定に従って実行されます。

ヒント：2つのリポジトリを結合する前に設定を変更するには、**[キャンセル]** をクリックして **[新規結合]** ダイアログ・ボックスを閉じ、以降の各項の説明に従って設定を変更した後、結合を実行する必要があります。

色の設定の指定

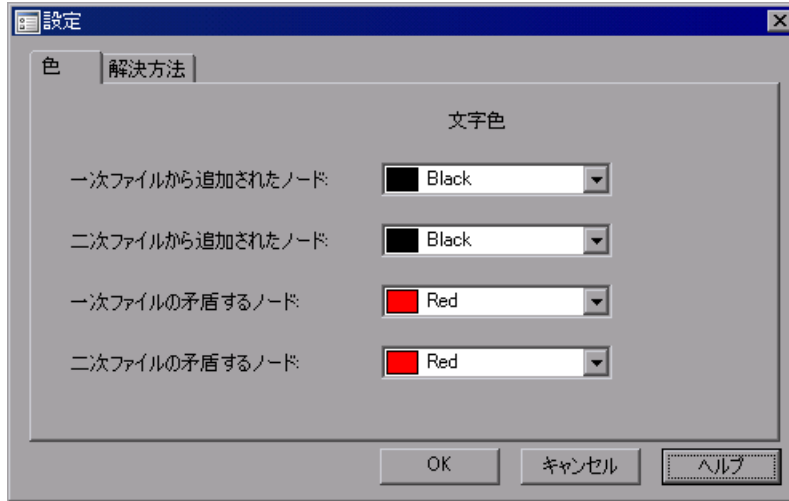
ターゲット・オブジェクト・リポジトリに表示されるオブジェクト名の色を、オブジェクトのソースと、矛盾の原因となったかどうかに基づいて指定できます。これを利用することで、各オブジェクトのステータスを容易に判断できるようになります。


注：[設定] ダイアログ・ボックスの [色] タブにあるオプションは、**[ローカルリポジトリから更新]** 操作を実行するときに、ローカル（一次）オブジェクト・リポジトリおよび共有（二次）オブジェクト・リポジトリから追加されたオブジェクトにも適用されます。

色の設定を指定するには、次の手順を実行します。



- 1 [ツール] > [設定] を選択するか、[設定] ボタンをクリックします。[設定] ダイアログ・ボックスが開きます。



- 2 [色] タブにある各項目について、テキスト・ボックスの横にある下矢印  をクリックし、識別用の色を選択します。
- 3 [OK] をクリックします。ターゲット・オブジェクト・リポジトリ内のオブジェクト名が選択した色で表示されます。

標準の解決方法の設定

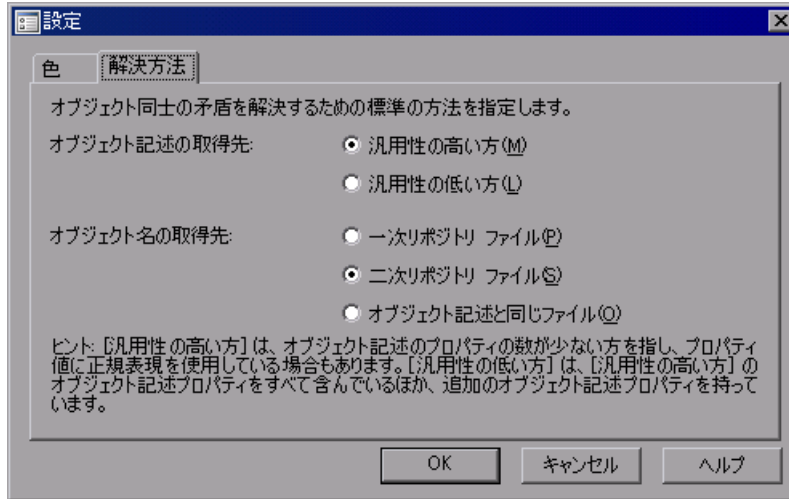
結合処理中のオブジェクトの矛盾をオブジェクト・リポジトリ結合ツールでどのように自動的に処理するかを設定できます。

注：[設定] ダイアログ・ボックスの [解決方法] タブにあるオプションは、**[ローカル リポジトリから更新]** 操作を実行するときに、ローカル・オブジェクト・リポジトリおよび共有オブジェクト・リポジトリから追加されたオブジェクトにも適用されます。

標準の解決方法を設定するには、次の手順を実行します。



- 1 **[ツール]** > **[設定]** を選択するか、**[設定]** ボタンをクリックします。**[設定]** ダイアログ・ボックスが開きます。
- 2 **[解決方法]** タブをクリックします。



- 3 矛盾のあるオブジェクトを処理するときオブジェクト・リポジトリ結合ツールに適用させる標準の解決方法を指定する適切なラジオ・ボタンを選択します。
 - ▶ **[オブジェクト記述の取得先]** : 2つのテスト・オブジェクトの名前が同じで記述が異なる場合の、矛盾の解決方法を指定します。ターゲット・オブジェクト・リポジトリにおいて、汎用性の高い方のオブジェクト記述を採用するか、汎用性の低い方のオブジェクト記述を採用するかを指定できます。
 - **[汎用性の高い方]** : 矛盾相手のオブジェクトよりも識別プロパティが少ないか、プロパティ値の中で正規表現を使用しているオブジェクトを採用するよう、オブジェクト・リポジトリ結合ツールを設定します。これが標準設定です。
 - **[汎用性の低い方]** : 矛盾相手のオブジェクトのすべての識別プロパティに加えて他の識別プロパティも持っているオブジェクトを採用するよう、オブジェクト・リポジトリ結合ツールを設定します。

- ▶ **[オブジェクト名の取得先]** : 2つのテスト・オブジェクトの記述が同一または類似しながらも名前が異なる場合の矛盾を解決する方法を指定します。ターゲット・オブジェクト・リポジトリにおいて採用するオブジェクト名の取得先となるソースを選択できます。
- **[一次リポジトリ ファイル]** : 一次オブジェクト・リポジトリ内のオブジェクトのオブジェクト名を、ターゲット・オブジェクト・リポジトリで採用します。これが標準設定です。
 - **[二次リポジトリ ファイル]** : 二次オブジェクト・リポジトリ内のオブジェクトのオブジェクト名を、ターゲット・オブジェクト・リポジトリで採用します。
 - **[オブジェクト記述と同じファイル]** : オブジェクト記述の取得先と同じオブジェクト・リポジトリ内のオブジェクトのオブジェクト名を、ターゲット・オブジェクト・リポジトリで採用します。
- 4 **[OK]** をクリックします。以降、オブジェクト・リポジトリ結合ツールで実行するリポジトリの結合においてオブジェクト間の矛盾を解決する際に、ここで選択した方法が適用されます。

注 : 結合後のオブジェクト・リポジトリを開いたままの状態では解決方法の設定に変更を加えた場合は、開いているファイルについて新しい設定でもう一度結合するかどうか尋ねられます。新しい設定でもう一度ファイルを結合する場合は、**[はい]** をクリックします。以前の設定で作成した既存の結合を維持する場合は、**[いいえ]** をクリックします。**[いいえ]** をクリックした場合、新しい設定は以降の結合にのみ適用されます。

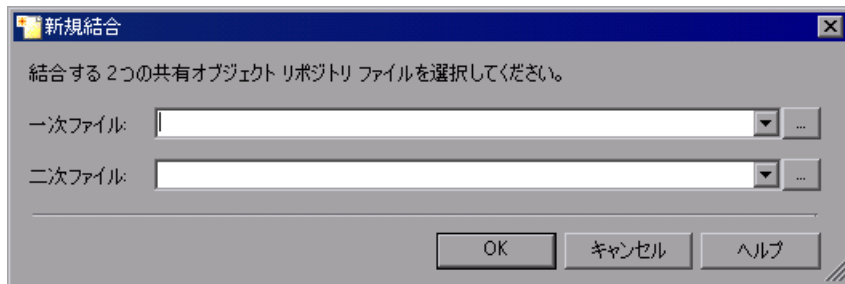
2 つのオブジェクト・リポジトリの結合

オブジェクト・リポジトリ結合ツールを使用すると、2 つのソース・オブジェクト・リポジトリを結合して新しい共有オブジェクト・リポジトリを作成できます。リポジトリにあるオブジェクトは自動的に比較され、オブジェクト間の矛盾の解決方法を定義する設定可能なルールに従って、新しいリポジトリに追加されます。元のソース・ファイルは変更されません。

注：別のユーザが開いているオブジェクト・リポジトリはロックされます。ロックされたファイルを統合しようとした場合、警告メッセージが表示されますが、結合プロセスはソース・ファイルを変更しないため、結合を実行することは可能です。ただし、ロックされたファイルに対して他のユーザによって加えられた変更が、結合後のオブジェクト・リポジトリに含まれない可能性があります。

2 つのオブジェクト・リポジトリを結合するには、次の手順を実行します。

- 1 オブジェクト・リポジトリ・マネージャの中で、[ツール] > [オブジェクトリポジトリ結合ツール] を選択します。[オブジェクトリポジトリ - 結合ツール] ウィンドウの手前に、[新規結合] ダイアログ・ボックスが開きます。




ヒント：



[オブジェクトリポジトリ - 結合ツール] ウィンドウがすでに開いている場合は、[ファイル] > [新規結合] を選択するか、[新規結合] ボタンをクリックすることで、[新規結合] ダイアログ・ボックスを開けます。

リポジトリを結合する前に設定済みの設定を変更するには、[キャンセル] をクリックして [新規結合] ダイアログ・ボックスを閉じ、436 ページ「標準設定の定義」の説明に従って設定を変更した後、結合を実行する必要があります。

- 2 [一次ファイル] および [二次ファイル] ボックスで、結合後の単独のリポジトリとなる **.tsr** オブジェクト・リポジトリの名前を入力するか、または参照して選択します。各ボックスの横にある下矢印  をクリックすると、最近使用したファイルを表示および選択できます。
-

注：

一次リポジトリには、最も作業内容の多いオブジェクト・リポジトリ、つまり、より多くのオブジェクト、オブジェクト・プロパティ、およびオブジェクト値を持つリポジトリを選択することをお勧めします。



拡張子が **.tsr** でないファイル、パスが正しくない **.tsr** ファイル、または存在しないファイルを入力した場合は、対応するテキスト・ボックスの横に警告アイコンが表示されます。アイコンの上にポインタを置くと、エラーを説明するツールチップが表示されます。正しいパスを持つ既存の **.tsr** ファイルを入力または選択してください。

以前のバージョンの QuickTest を使用して作成されたオブジェクト・リポジトリを結合する場合は、はじめにオブジェクト・リポジトリ・マネージャでそのオブジェクト・リポジトリを開いてから保存して新しい形式に更新する必要があります。

- 3 [OK] をクリックします。オブジェクト・リポジトリ結合ツールによって、設定されている解決方法の設定に従い、選択したオブジェクト・リポジトリが新しいターゲット・オブジェクト・リポジトリに自動的に結合されます。結合の結果は、[オブジェクトリポジトリ - 結合ツール] ウィンドウの手前に表示される [統計情報] ダイアログ・ボックスに表示されます。
- 4 449 ページ「結合の統計情報の表示」の説明を参考にして、結合に関する統計情報を確認し、[閉じる] をクリックします。

[オブジェクトリポジトリ - 結合ツール] ウィンドウでは、次を実行できます。

- ▶ ソース・リポジトリのオブジェクト間の矛盾を解決する方法を必要に応じて変更できます。詳細については、453 ページ「オブジェクトの矛盾の解決」を参照してください。
- ▶ ターゲット・オブジェクト・リポジトリ内のオブジェクトにフィルタを適用できます。詳細については、455 ページ「ターゲット・リポジトリ表示枠に対するフィルタの設定」を参照してください。
- ▶ ターゲット・オブジェクト・リポジトリ内で特定のオブジェクトを検索できます。詳細については、456 ページ「オブジェクト・リポジトリ・ビューの同期」を参照してください。
- ▶ ターゲット・オブジェクト・リポジトリをファイル・システムまたは Quality Center プロジェクトに保存できます。詳細については、459 ページ「ターゲット・リポジトリの保存」を参照してください。

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

1 つ以上のコンポーネント（アプリケーション領域）に関連付けられているローカル・オブジェクト・リポジトリを、共有オブジェクト・リポジトリに結合することによって、共有オブジェクト・リポジトリを更新することができます。更新後、ローカル・オブジェクト・リポジトリから結合されたオブジェクトは、当該共有オブジェクト・リポジトリを使用する任意のコンポーネントから利用できるようになります。

結合処理では、選択したコンポーネントに対応するローカル・オブジェクト・リポジトリ内のオブジェクトが、ターゲットの共有オブジェクト・リポジトリに移動します。そして、当該コンポーネントで、更新後の共有オブジェクト・リポジトリのオブジェクトが使用されるようになります。

複数のコンポーネント用のローカル・オブジェクト・リポジトリを追加することを選択した場合は、QuickTestによって複数の結合が実行され、各コンポーネントのローカル・オブジェクト・リポジトリがリスト内のすべてのコンポーネントに対して一度に1つずつターゲット・オブジェクト・リポジトリに結合されます。必要ならば、各結合の結果を表示および変更できます。

注：結合できるのは、アプリケーション領域が更新対象の共有オブジェクト・リポジトリに関連付けられているコンポーネントのローカル・オブジェクト・リポジトリのみです。

ローカル・オブジェクト・リポジトリから共有オブジェクト・リポジトリを更新するには、次の手順を実行します。

- 1 [リソース] > [オブジェクト リポジトリ マネージャ] を選択します。オブジェクト・リポジトリ・マネージャが開きます。

注：オブジェクト・リポジトリ・マネージャの詳細については、第15章「オブジェクト・リポジトリの管理」を参照してください。



- 2 オブジェクト・リポジトリ・マネージャの中で、[ファイル] > [開く] を選択するか、[開く] ボタンをクリックします。[共有オブジェクト リポジトリを開く] ダイアログ・ボックスが表示されます。

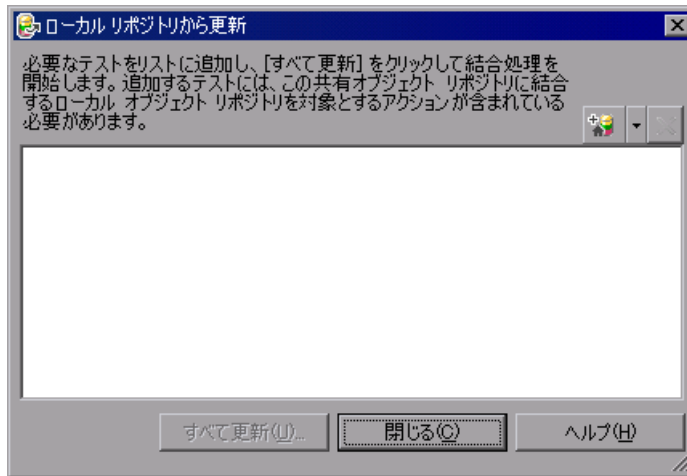
現在 Quality Center プロジェクトに接続している場合は、[共有オブジェクト リポジトリを開く] ダイアログ・ボックスにプロジェクトのコンポーネント・ツリーが表示されます。コンポーネントを選択すると、コンポーネントに添付されている共有オブジェクト・リポジトリが表示されます。


- 更新対象の共有オブジェクト・リポジトリが格納されている **.tsr** ファイルを参照し、**[読み取り専用モードで開く]** チェック・ボックスの選択を解除して、**[開く]** をクリックします。または、Quality Center の添付ファイルの場合は **[OK]** をクリックします。ファイルが開き、オブジェクトとプロパティが編集可能な形式で表示されます。



ヒント：オブジェクト・リポジトリを読み取り専用モードで開いた場合は、**[ファイル]** > **[編集を有効化]** を選択するか、オブジェクト・リポジトリ・マネージャのツールバーの **[編集を有効化]** ボタンをクリックします。オブジェクト・リポジトリ・ファイルが編集可能になります。

- [ツール]** > **[ローカル リポジトリから更新]** を選択します。**[ローカル リポジトリから更新]** ダイアログ・ボックスが開きます。



- Quality Center プロジェクトに接続していることを確認します。**[テストを追加します]** ボタンの横にある下矢印  をクリックして、**[テストを参照]** を選択します。**[Quality Center プロジェクトから QuickTest コンポーネントを開く]** ダイアログ・ボックスが開きます。

コンポーネントを参照します。このコンポーネントのローカル・オブジェクト・リポジトリが共有オブジェクト・リポジトリに結合されます。

注：追加可能なコンポーネントは、そのアプリケーション領域が更新対象の共有オブジェクト・リポジトリに関連付けられていて、そのローカル・オブジェクト・リポジトリにオブジェクトが含まれているコンポーネントのみです。

- 6 必要ならば、手順5を繰り返してコンポーネントをさらに追加します。



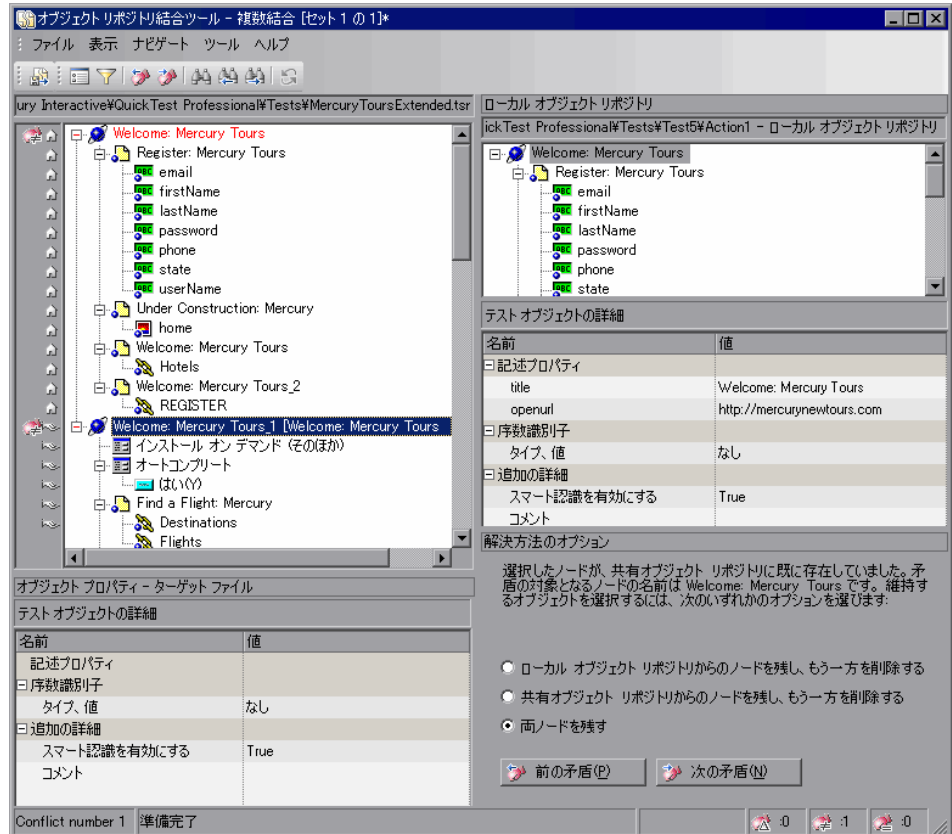
注：すべてのコンポーネントに関連付けられているローカル・オブジェクト・リポジトリが結合に含まれます。特定のコンポーネントを結合から除外するには、一覧の中でそれを選択して **[削除]** をクリックします。

- 7 **[すべて更新]** をクリックします。QuickTestによって、設定済みの設定に従い、最初のコンポーネントのローカル・オブジェクト・リポジトリが共有オブジェクト・リポジトリに自動的に結合されます。結合の結果は、[オブジェクトリポジトリ - 結合ツール] ウィンドウの手前に表示される **[統計情報]** ダイアログ・ボックスに表示されます。


注：各結合の前に、別のユーザがローカル・オブジェクト・リポジトリを使用していないかどうか QuickTestによって確認されます。別のユーザが使用している場合、そのローカル・オブジェクト・リポジトリはロックされているので、選択したコンポーネントに対応したオブジェクトをターゲットの共有オブジェクト・リポジトリに移動することはできません。警告メッセージが表示されます。他のユーザがローカル・オブジェクト・リポジトリの使用を止めれば、結合を実行できます。


- 8 449 ページ「結合の統計情報の表示」の説明を参考にして、結合に関する統計情報を確認し、**[閉じる]** をクリックします。

ローカル・オブジェクト・リポジトリの結合の場合、[オブジェクトリポジトリ - 結合ツール] ウィンドウには、ローカル・オブジェクト・リポジトリが一次オブジェクト・リポジトリとして表示され、共有オブジェクト・リポジトリがターゲット・オブジェクト・リポジトリとして表示されます。



ターゲット・オブジェクト階層内の各オブジェクトの左側には、オブジェクトのソースを示すアイコンが表示されます。

 は、ノードがローカル・オブジェクト・リポジトリから追加されたことを示します。

 は、ノードがすでに共有オブジェクト・リポジトリに存在することを示します。

注： [ローカル リポジトリから更新] ダイアログ・ボックスで複数のコンポーネントを指定した場合は、QuickTest によって複数の結合が実行され、各コンポーネントのローカル・オブジェクト・リポジトリが一度に1つずつターゲット・オブジェクト・リポジトリに結合されます。この手順の後に表示される [統計情報] ダイアログ・ボックスと [オブジェクト リポジトリ結合ツール - 複数結合] ウィンドウには、最初の結合の結果が表示されます（最初のコンポーネントのローカル・オブジェクト・リポジトリが共有オブジェクト・リポジトリに結合されます）。必要ならば、QuickTest で各結合の結果を順に表示および変更できます。複数結合では、各結合セットの番号がタイトル・バーに表示されます。たとえば、「[セット 3 の 2]」などと表示されます。

- 9 共有オブジェクト・リポジトリに結合された各オブジェクトに対しては、自動結合を選択するか、解決方法のオプション表示枠を使用して次のことを行うことができます。
- ▶ 特定のオブジェクトを共有オブジェクト・リポジトリに追加し、ローカル・オブジェクト・リポジトリから削除する。
 - ▶ 特定のオブジェクトをローカル・オブジェクト・リポジトリに維持し、共有オブジェクト・リポジトリには追加しない。

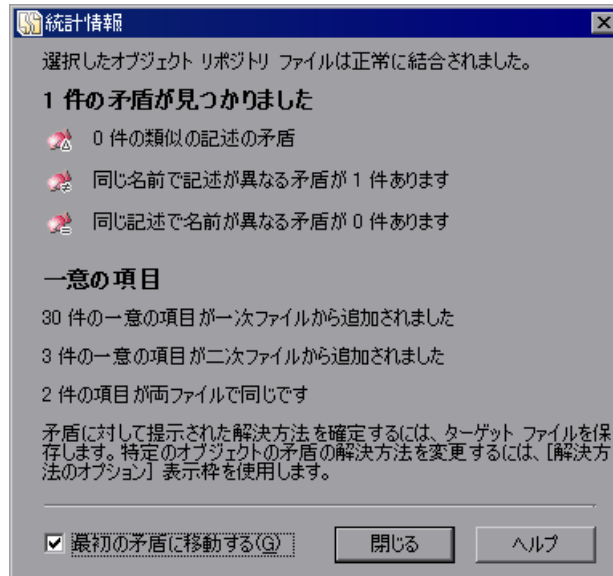
詳細については、453 ページ「オブジェクトの矛盾の解決」を参照してください。



- 10 複数の結合を実行している場合は、[オブジェクト リポジトリ - 結合ツール] のツールバーの [保存して次を結合] ボタンをクリックして、次の結合を実行します（次のコンポーネントのローカル・オブジェクト・リポジトリが共有オブジェクト・リポジトリに結合されます）。
- 11 結合ごとに [はい] をクリックして変更を保存します。[いいえ] をクリックした場合は、現在の結合（最後のコンポーネントから結合されたオブジェクト）は保存されません。
- 12 手順 8 から 11 を繰り返して複数結合を最後まで実行します。
- 13 [ファイル] > [終了] をクリックし、[はい] をクリックすると、更新されたオブジェクト・リポジトリが保存されます。

結合の統計情報の表示

2つのオブジェクト・リポジトリを結合した後、オブジェクト・リポジトリ結合ツールには「統計情報」ダイアログ・ボックスが表示されます。このダイアログ・ボックスには、ファイルが結合された方法と、結合中に解決されたすべての矛盾の数とタイプが表示されます。



注：「ローカル リポジトリから更新」操作を行った後に表示される統計情報は、前述のオプションとは若干異なります。

ヒント：「統計情報」ダイアログ・ボックスの結合に関する統計情報は、「オブジェクト リポジトリ - 結合ツール」ウィンドウで「表示」>「統計情報」を選択するか、ステータス・バーの矛盾アイコンをクリックすることで、いつでも表示できます。

[統計情報] ダイアログ・ボックスには次の情報が表示されます。

- ▶ ターゲット・オブジェクト・リポジトリに追加されたオブジェクト間で発生した、すべての矛盾の数とタイプ。矛盾タイプの詳細については、453 ページ「オブジェクトの矛盾の解決」を参照してください。
- ▶ ターゲット・オブジェクト・リポジトリに追加された、一次ファイルまたは二次ファイル（またはローカル・ファイル）の中の一意の項目または両方のファイルでまったく同じ項目の数。

ヒント：[最初の矛盾に移動する] チェック・ボックスを選択すると、[統計情報] ダイアログ・ボックスを閉じた直後に、ターゲット・オブジェクト・リポジトリ内の最初の矛盾に移動します。

オブジェクトの矛盾について

2つのオブジェクト・リポジトリを結合する際、それらに含まれているテスト・オブジェクトどうしが類似しているために、矛盾が発生することがあります。オブジェクト・リポジトリ統合ツールでは、発生する可能性のある3つの矛盾タイプが識別されます。



- ▶ **[類似の記述の矛盾]**：名前とオブジェクト階層が同じで、記述がわずかに異なる、2つのテスト・オブジェクト。この矛盾タイプでは、一方のオブジェクトが常に他方のプロパティ・セットのサブセットを持っています。これらの矛盾の詳細については、451 ページを参照してください。

標準設定では、このタイプの矛盾に対する矛盾の解決方法の設定は、矛盾相手のオブジェクトよりも識別プロパティの少ない方のオブジェクトがターゲット・オブジェクト・リポジトリで採用されるように設定されます。標準設定の変更方法の詳細については、436 ページ「標準設定の定義」を参照してください。



- ▶ **[同じ名前で記述が異なる矛盾]**：名前とオブジェクト階層が同じであるものの、それらの記述がどこか異なる（たとえば、プロパティが異なる、プロパティが同じでも値が異なるなど）、2つのテスト・オブジェクト。これらの矛盾の詳細については、452 ページを参照してください。

標準設定では、このタイプの矛盾に対する矛盾の解決方法の設定は、両方のファイルのオブジェクトがターゲット・オブジェクト・リポジトリで採用されるように設定されます。二次ファイルから追加されるオブジェクトの名前は変更され、1 つずつ値が大きくなる数字の接尾辞が名前に付けられます。たとえば、Edit_1 などという名前になります。標準設定の変更方法の詳細については、436 ページ「標準設定の定義」を参照してください。



- ▶ **[同じ記述で名前が異なる矛盾]**：記述がまったく同じで、オブジェクト階層が同じものの、オブジェクト名が異なる、2 つのテスト・オブジェクト。これらの矛盾の詳細については、452 ページを参照してください。

標準設定では、このタイプの矛盾に対する矛盾の解決方法の設定は、一次ソース・ファイルのオブジェクト名がターゲット・オブジェクト・リポジトリで採用されるように設定されます。標準設定の変更方法の詳細については、436 ページ「標準設定の定義」を参照してください。

注：記述を持たない Page オブジェクトや Browser オブジェクトなどのオブジェクトは名前のみ比較されます。両方のソース・リポジトリに同じオブジェクトが含まれていて、それらの名前が異なる場合は、2 つの別々のオブジェクトとしてターゲット・オブジェクト・リポジトリに結合されます。

類似記述の矛盾

一次オブジェクト・リポジトリ内のオブジェクトと二次オブジェクト・リポジトリ内のオブジェクトが、同じ名前を持ち、まったく同じではないけれども類似する記述プロパティおよび値を持っている場合です。一方のオブジェクトが常に他方のプロパティ・セットのサブセットを持っています。たとえば、二次オブジェクト・リポジトリにある Button_1 という名前のオブジェクトが、一次オブジェクト・リポジトリにある Button_1 という名前のオブジェクトと同じ記述プロパティおよび値を持っているものの、さらに追加のプロパティと値を持っているとします。

この矛盾タイプは次のようにして解決できます。

- ▶ 一次リポジトリから追加されるオブジェクトのオブジェクト記述を採用する。
- ▶ 二次リポジトリから追加されるオブジェクトのオブジェクト記述を採用する。
- ▶ 両方のオブジェクトをターゲット・オブジェクト・リポジトリに取り込む。この場合、ターゲット・オブジェクト結合ツールでは、二次ファイルから追加されるオブジェクトの名前が自動的に変更され、1つずつ値が大きくなる数字の接尾辞が名前に付けられます。たとえば、Edit_1 などという名前になります。

同じ名前で記述が異なる矛盾

一次オブジェクト・リポジトリ内のオブジェクトと二次オブジェクト・リポジトリ内のオブジェクトが、同じ名前を持つものの、完全に異なる記述プロパティおよび値を持っている場合です。

この矛盾タイプは次のようにして解決できます。

- ▶ 一次リポジトリから追加されるオブジェクトのみを維持する。
- ▶ 二次リポジトリから追加されるオブジェクトのみを維持する。
- ▶ 両方のリポジトリからのオブジェクトを維持する。この場合、ターゲット・オブジェクト結合ツールでは、二次ファイルから追加されるオブジェクトの名前が自動的に変更され、1つずつ値が大きくなる数字の接尾辞が名前に付けられます。たとえば、Edit_1 などという名前になります。

同じ記述で名前が異なる矛盾

一次オブジェクト・リポジトリ内のオブジェクトと二次オブジェクト・リポジトリ内のオブジェクトが、異なる名前を持つものの、同じ記述プロパティおよび値を持っている場合です。

この矛盾タイプは次のようにして解決できます。

- ▶ 一次リポジトリにある該当オブジェクトのオブジェクト名を採用する。
- ▶ 二次リポジトリにある該当オブジェクトのオブジェクト名を採用する。

オブジェクトの矛盾の解決

一次オブジェクト・リポジトリと二次オブジェクト・リポジトリにあるオブジェクトどうしの矛盾は、オブジェクト・リポジトリ結合ツールによって、標準として設定されている解決方法に従って自動的に解決されます。標準の解決方法は、結合を実行する前に設定できます。詳細については、436 ページ「標準設定の定義」を参照してください。

ただし、オブジェクト・リポジトリ結合ツールでは、矛盾の原因となった個々のオブジェクトごとに、結合の実行方法を変更することも可能です。

たとえば、一次リポジトリ内のオブジェクトが、二次リポジトリ内のオブジェクトと同じ名前だったものの、記述が異なるとします。このとき、標準の設定として、汎用性の高い方のオブジェクト記述を持つオブジェクト、つまり、プロパティの数の少ない方のオブジェクトをターゲット・オブジェクト・リポジトリに追加すると定義していたとしましょう。しかし、自動結合の後に矛盾を確認した結果、その特定の矛盾を別の方法（たとえば、両方のオブジェクトを維持するなど）で扱うように判断することが考えられます。

注：標準の矛盾の解決方法に変更を加えること自体が新しい矛盾の原因となり、ターゲット・オブジェクト・リポジトリに影響を与えることがあります。前述の例では、両方のオブジェクトを維持することが名前の矛盾の原因となります。したがって、矛盾の解決方法を変更するたびにターゲット・オブジェクト・リポジトリが更新され、再表示されます。

オブジェクト・リポジトリ結合ツールのターゲット・オブジェクト・リポジトリ表示枠で、オブジェクト名の左側に表示されるアイコンと、テキストの色によって、矛盾の原因となったオブジェクトと、矛盾のタイプを、識別することができます。矛盾のあるオブジェクトを選択すると、矛盾の詳細な説明が、オブジェクト・リポジトリ結合ツールによる矛盾の自動解決方法とともに、解決方法のオプション表示枠に表示されます。

解決方法のオプション表示枠には、代わりの解決方法のオプションが提示されます。標準の解決方法がニーズに合っていればそれを維持することも、代わりのオプションを使用して矛盾を別の方法で解決することも選択できます。

ヒント：また、標準の解決方法の設定を変更して、ファイルを再び結合することもできます。詳細については、436 ページ「標準設定の定義」を参照してください。

オブジェクトの矛盾を解決するには、次の手順を実行します。

- 1 ターゲット・オブジェクト・リポジトリで、矛盾のあるオブジェクトを選択します。矛盾のあるオブジェクトはオブジェクト名の左側のアイコンで示されます。矛盾しているオブジェクトはソース・リポジトリで強調表示されています。
矛盾の説明と、オブジェクト・リポジトリ結合ツールが使用する解決方法が、解決方法のオプション表示枠に表示されます。使用できる代替の解決方法ごとに、対応するラジオ・ボタンが表示されます。各矛盾タイプの詳細については、450 ページ「オブジェクトの矛盾について」を参照してください。
- 2 解決方法のオプション表示枠で、ラジオ・ボタンを選択し、代わりとなる矛盾の解決方法を選択します。選択した解決方法に従ってターゲット・オブジェクト・リポジトリが更新され、再表示されます。
- 3 解決方法のオプション表示枠で、[次の矛盾] または [前の矛盾] ボタンをクリックすると、ターゲット・リポジトリ階層内の次または前の矛盾に直接移動できます。
- 4 矛盾の解決方法をほかにも変更する必要がある場合は、手順1から3を繰り返します。
- 5 ターゲット・オブジェクト・リポジトリを保存します。詳細については、459 ページ「ターゲット・リポジトリの保存」を参照してください。

ターゲット・リポジトリ表示枠に対するフィルタの設定

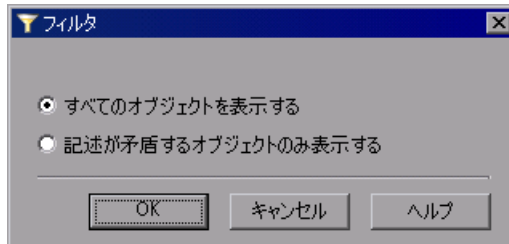
2つのオブジェクト・リポジトリを結合した結果、ターゲット・オブジェクト・リポジトリに含まれるオブジェクトの数が非常に多くなることがあります。ターゲット・リポジトリ表示枠でのナビゲーションや特定のオブジェクトの検索が容易になるように、オブジェクト・リポジトリ結合ツールでは、表示枠内のオブジェクトを絞り込み（フィルタ処理）、結合中に解決された矛盾のあるオブジェクトだけを表示することができます。

注：フィルタは、ターゲット・リポジトリ表示枠にどのオブジェクトを表示するかのみを決めるものです。どのオブジェクトをターゲット・オブジェクト・リポジトリに含めるのかを決めるものではありません。

ターゲット・リポジトリ表示枠のオブジェクトを絞り込むには、次の手順を実行します。



- 1 [ツール] > [フィルタ] を選択するか、[フィルタ] ボタンをクリックします。[フィルタ] ダイアログ・ボックスが表示されます。



ヒント：ステータス・バー内のアイコンの左側にあるボックスをクリックして [フィルタ] ダイアログ・ボックスを表示することもできます。この領域には、フィルタが現在使用されている場合にフィルタ・アイコンが表示されます。

- 2 ターゲット・オブジェクト・リポジトリに表示するオブジェクトに対応したラジオ・ボタンを選択します。
 - ▶ **[すべてのオブジェクトを表示する]**：ターゲット・オブジェクト・リポジトリ内のすべてのオブジェクトが表示されます。
 - ▶ **[記述が矛盾するオブジェクトのみ表示する]**：ターゲット・オブジェクト・リポジトリ内のオブジェクトのうち、記述の矛盾が生じたもののみが表示されます。
- 3 **[OK]** をクリックします。表示枠内のオブジェクトが絞り込まれ、設定したオブジェクト・タイプのみがターゲット・オブジェクト・リポジトリに表示されます。

オブジェクト・リポジトリ・ビューの同期

ターゲット・オブジェクト結合ツールでは、ターゲット、一次、二次の各オブジェクト・リポジトリを独立してナビゲートできます。各種の表示枠のサイズを変更して、リポジトリに格納されているオブジェクトの一部だけを表示することもできます。このため、大きなオブジェクト・リポジトリを扱っている場合に、各種の表示枠にリポジトリ階層の異なる領域が表示され、結合プロセスの影響を受ける特定のオブジェクトを探して追跡することが難しくなることがあります。



リポジトリの同期をとり、両方のビューに同じオブジェクトが表示されるようにするには、当該オブジェクトが現在表示されている一次または二次オブジェクト・リポジトリ内でそのオブジェクトを選択し、**[ビューを同期化]** をクリックします。

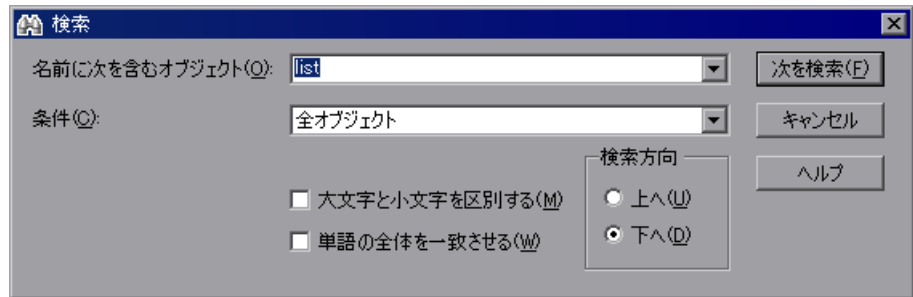
特定のオブジェクトの検索

オブジェクト・リポジトリ結合ツールの検索機能を使用して、名前に指定の文字列が含まれている 1 つ以上のオブジェクトをターゲット・オブジェクト・リポジトリの中で検索できます。見つかったオブジェクトは、対応する一次リポジトリまたは二次リポジトリの一方または両方でも強調表示されます。

オブジェクトを検索するには、次の手順を実行します。



- 1 **[ナビゲート]** > **[検索]** を選択するか、**[検索]** ボタンをクリックします。**[検索]** ダイアログ・ボックスが開きます。



- 2 **[名前に次を含むオブジェクト]** ボックスに、検索するオブジェクトの名前の全体または一部を入力します。
- 3 **[条件]** ボックスで、検索するオブジェクトを選択して検索対象を絞込みます。次の条件を使用できます。
 - ▶ **[全オブジェクト]**
 - ▶ **[一方のソースにのみ存在するオブジェクト]**
 - ▶ **[矛盾するオブジェクト]**
 - ▶ **[矛盾するオブジェクトまたは一方にのみ存在するオブジェクト]**

- 4 次のオプションの一方または両方を選択して検索の絞り込みに役立てることができます。
 - ▶ **[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、**[名前に次を含むオブジェクト]** 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。
 - ▶ **[単語の全体を一致させる]**：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。
- 5 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を指定します。**[上へ]** または **[下へ]** のいずれかを選択できます。

ヒント：**[上へ]** または **[下へ]** を選択すると、現在のカーソルの位置から先頭または末尾に向かって、ターゲット・オブジェクト・リポジトリが検索されます。リポジトリ全体を検索するには、階層内の最初（または最後）のオブジェクトを選択し、**[下へ]**（または **[上へ]**）を選択します。

- 6 **[次を検索]** をクリックすると、ターゲット・オブジェクト・リポジトリ内で指定の条件に一致する次のオブジェクトが強調表示されます。
[検索] ダイアログ・ボックスを閉じて次のコマンドを使用することもできます。
 - ▶ **[次を検索]** ボタンをクリックするか、**[ナビゲート]** > **[次を検索]** を選択すると、指定の条件に一致する次のオブジェクトが強調表示されます。
 - ▶ **[前を検索]** ボタンをクリックするか、**[ナビゲート]** > **[前を検索]** を選択すると、指定の条件に一致する前のオブジェクトが強調表示されます。



ターゲット・リポジトリの保存

オブジェクトの矛盾が意図どおりに解決されたことを確認したら、ターゲット・リポジトリをファイル・システムまたは Quality Center プロジェクト (QuickTest が現在 Quality Center プロジェクトに接続している場合) に保存できます。

保存できるファイルは、結合したオブジェクト・リポジトリの種類に応じて異なります。2つの共有オブジェクト・リポジトリを結合した場合は、作成された新しいターゲット・オブジェクト・リポジトリを保存することができます。1つ以上のローカル・オブジェクト・リポジトリを共有オブジェクト・リポジトリと結合した場合は、ローカル・オブジェクト・リポジトリからのオブジェクトおよびデータを含んでいる既存の共有オブジェクト・リポジトリ・ファイルを保存することができます。

ファイル・システムへのオブジェクト・リポジトリの保存

新しく結合した共有オブジェクト・リポジトリを、いつでもファイル・システムに保存できます。

オブジェクト・リポジトリをファイル・システムに保存するには、次の手順を実行します。



- 1 **[ファイル]** > **[保存]** を選択するか、**[保存]** ボタンをクリックします。ファイルを以前に保存したことがあれば、現在の変更内容が保存されます。ファイルを保存したことがなければ、**[共有オブジェクトリポジトリの保存]** ダイアログ・ボックスが開きます。

注： Quality Center に接続している場合の **[共有オブジェクトリポジトリの保存]** ダイアログ・ボックスは、標準のファイル選択ダイアログ・ボックスとは異なります。このダイアログ・ボックスにある **[ファイル システム]** ボタンをクリックすることで、ファイルをファイル・システムに保存するよう切り替えることができます。

- 2 オブジェクト・リポジトリを保存するフォルダに移動し、そのフォルダを選択します。オブジェクト・リポジトリの名前を「**ファイル名**」ボックスに入力します。

ファイルが識別しやすいように、分かりやすい名前を使用します。オブジェクト・リポジトリ・ファイルの名前では、次の文字は使用できません。

: ¥ / : " ? < > | *

- 3 「**保存**」をクリックします。QuickTest によって、ファイル名に **.tsr** 拡張子が付けられ、指定された場所にオブジェクト・リポジトリが保存されます。そして「オブジェクトリポジトリ - 結合ツール」ウィンドウ内のターゲット・オブジェクト・リポジトリの上に、ファイル名とパスが表示されます。

Quality Center プロジェクトへのオブジェクト・リポジトリの保存

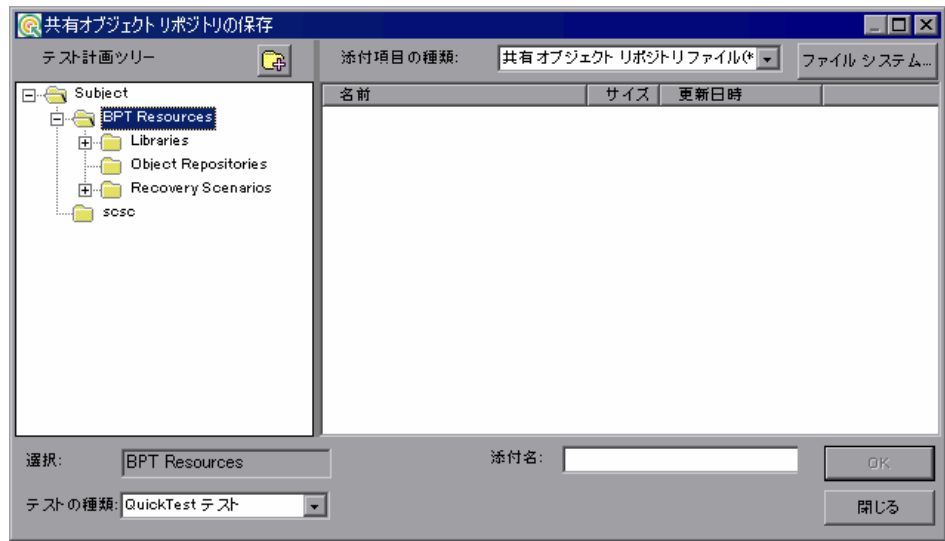
Quality Center に接続している場合は、結合後の共有オブジェクト・リポジトリを、プロジェクトのテスト計画ツリー内の添付ファイルとして保存できます。後で、リポジトリを1つ以上の必要なアプリケーション領域と関連付けることで、コンポーネントからリポジトリ内のオブジェクトにアクセスできるようになります。詳細については、94 ページ「共有オブジェクト・リポジトリの管理」を参照してください。

注：Quality Center 内の既存のオブジェクト・リポジトリを上書きすることはできません。

オブジェクト・リポジトリを 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 [ファイル] > [保存] を選択するか、[保存] ボタンをクリックします。ファイルを以前に Quality Center に保存したことがあれば、現在の変更内容がオブジェクト・リポジトリに保存されます。ファイルを保存したことがなければ、[共有オブジェクトリポジトリの保存] ダイアログ・ボックスが開きます。



- 2 テスト計画ツリーの中で、オブジェクト・リポジトリの保存先となるフォルダを選択します。



[**フォルダの新規作成**] ボタンをクリックして、Quality Center のテスト計画ツリーに新しいテスト・フォルダを作成することもできます。

注： [共有オブジェクトリポジトリの保存] ダイアログ・ボックスにある [**ファイル システム**] ボタンをクリックすると、ファイルをファイル・システムに保存するように切り替えることができます。[**Quality Center**] ボタンをクリックすれば、Quality Center 用の [共有オブジェクトリポジトリの保存] ダイアログ・ボックスに戻ることができます。

- 3 オブジェクト・リポジトリの名前を [添付名] ボックスに入力します。

オブジェクト・リポジトリが識別しやすいように、分かりやすい名前を使用します。オブジェクト・リポジトリ・ファイルの名前では、次の文字は使用できません。

¥ / : " ? < > | *

注：既存のオブジェクト・リポジトリを上書きすることはできません。

- 4 [OK] をクリックします。QuickTest によって、オブジェクト・リポジトリが Quality Center に保存され、[オブジェクトリポジトリ - 結合ツール] ウィンドウ内のターゲット・オブジェクト・リポジトリの上にファイル名とパスが表示されます。Quality Center では、ファイルは対応するフォルダの [添付ファイル] タブに表示されます。

第 5 部

コンポーネントと実行のデバッグ

第 17 章

コンポーネン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、コンポーネント、関数ライブラリ、登録済みユーザ関数内の不具合を特定し対処できます。

注：QuickTest でコンポーネントをデバッグするためには、**[他の Mercury 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（QuickTest から **[ツール]** > **[オプション]** > **[実行]** を選択）して、QuickTest と Quality Center プロジェクトの統合を有効にしておく必要があります。

本章では、次の項目について説明します。

- ▶ コンポーネントと関数ライブラリのデバッグについて
- ▶ デバッグ・セッションの低速化
- ▶ 単一ステップ・コマンドの使用法
- ▶ [ステップまで実行] コマンドおよび [ステップから開始] コマンドの使用法
- ▶ 実行セッションの一時停止
- ▶ ブレークポイントの使用法
- ▶ デバッグ・ビューアの使用
- ▶ 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

コンポーネントと関数ライブラリのデバッグについて

コンポーネン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（アプリケーション領域を通じて）コンポーネントに関連付け、次にそのコンポーネントから関数ライブラリをデバッグする必要があります。

コンポーネン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。

[ステップから開始] コマンドを使用すれば、コンポーネン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、コンポーネント内の特定のポイントで実行を一時停止できます。ブレイクポイントの設定後、コンポーネン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イクポイントを有効にしたり無効にしたりできます。

コンポーネントまたは関数ライブラリの実行がブレイ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、コンポーネン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コンポーネン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、コンポーネン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、『**QuickTest Professional 基本機能ユーザーズ・ガイド**』の「コンポーネントの一部の実行」を参照してください。

注：

- ▶ コンポーネン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、読み取り専用になります。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、591 ページ「読み取り専用関数ライブラリの編集」を参照してください。変更後、コンポーネン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、別のコンポーネントを開いたり、新規のコンポーネン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。
 - ▶ QuickTest では、コンポーネントを開くと、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、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、コンポーネントを閉じて再度開くまでコンポーネントに反映されません（外部リソースとは、外部エディタを使用して作成した関数ライブラリなど、QuickTest を使用して作成したものではないリソースのことです）。
-

デバッグ・セッションの低速化

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。コンポーネントまたは関数ライブラリのデバッグ中、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・オプションの詳細については、199 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。

単一ステップ・コマンドの使用方法

[ステップイントウ] 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、コンポーネン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。

[ステップイントウ]



アクティブなコンポーネン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] > [ステップイントウ] を選択するか、[ステップイントウ] ボタンをクリックするか、あるいは F11 キーを押します。アクティブなコンポーネントまたは関数ライブラリの現在の行が、関数を呼び出している場合には、呼び出し先の関数が [QuickTest] ウィンドウに表示され、呼び出し先の関数の最初の行で関数ライブラリが一時停止します。



[ステップアウト]

[**ステップイントウ**] を使用して、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、あるいは SHIFT + F11 を押します。[**ステップアウト**] によって、ユーザ定義関数が最後まで実行された後、呼び出し元のコンポーネン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]



アクティブなコンポーネン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、あるいは F10 キーを押します。現在のステップで、ユーザ定義関数が呼び出されている場合は、呼び出し先の関数の全体が実行されますが、呼び出し先の関数スクリプトは QuickTest ウィンドウに表示されません。

ステップ・コマンドの使用 - 例

次の手順に従って、サンプルの関数ライブラリを作成し、[**ステップイントウ**] 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実行します。

サンプルの関数ライブラリを作成するには、次の手順を実行します。

- 1 [ファイル] > [新規作成] > [アプリケーション領域] を選択します。新しいアプリケーション領域が開きます（詳細については、79 ページ「アプリケーション領域の作成」を参照してください）。

2 **SampleFL.qfl** という新しい関数ライブラリを作成し、Quality Center プロジェクトに保存します（詳細については、89 ページ「関数ライブラリの管理」を参照してください）。

3 **SampleFL.qfl** を開き、次の行を正確に入力します。

```
public Function myfunc()  
msgbox "one"  
msgbox "two"  
msgbox "three"  
End Function
```

4 関数ライブラリをコンポーネントのアプリケーション領域に関連付けるには、**[ファイル] > [ライブラリ'<関数ライブラリ名>'を'<アプリケーション領域名>'に関連付ける]** を選択するか、右クリックして **[ライブラリ'<関数ライブラリ名>'を'<アプリケーション領域名>'に関連付ける]** を選択します。関数ライブラリがアプリケーション領域に関連付けられます。

[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**を使用してコンポーネントを実行するには、次の手順を実行します。


1 前項で作成したアプリケーション領域に基づいて新規のコンポーネントを作成します。

2 まったく同じステップを3つ挿入します。各ステップについて、次を実行します。

▶ **[項目]** セルの中で、**[操作]** を選択します。

▶ **[操作]** セルの中で、**[myfunc]** を選択します。

3 **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、**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。

 4 F9 キー（**ブレークポイントの設定 / 解除**）を押して、コンポーネントの1行目（最初の **myfunc** 関数の呼び出し）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、477 ページ「ブレークポイントの設定」を参照してください。

5 コンポーネントを実行します。コンポーネントはブレークポイントで一時停止します。

- 6 F11 キー ([**ステップ イントウ**]) 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。
- 7 F11 キー ([**ステップ イントウ**]) 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。
- 8 [**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。
- 9 実行矢印が関数から出てスクリプトの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 イントウ**]) を繰り返し押します。
- 10 F11 キー ([**ステップ イントウ**]) を押して、もう一度関数に入ります。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。
- 11 **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たままです。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。
- 12 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 [ステップから開始] コマンドの使用方法

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・オーバーのほかに、**[ステップまで実行]** および **[ステップから開始]** を使用して、特定のステップに到達するまでコンポーネント（関連付けられている任意の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指定できます。

[ステップまで実行]

コンポーネントの最初から、あるいはコンポーネント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定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、コンポーネント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があります。

このオプションは、コンポーネント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+F10 を押します。
- ▶ 実行を停止するステップで右クリックし、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：コンポーネントの編集中に **[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「**[ステップから開始]**」の項の手順2を参照してください。

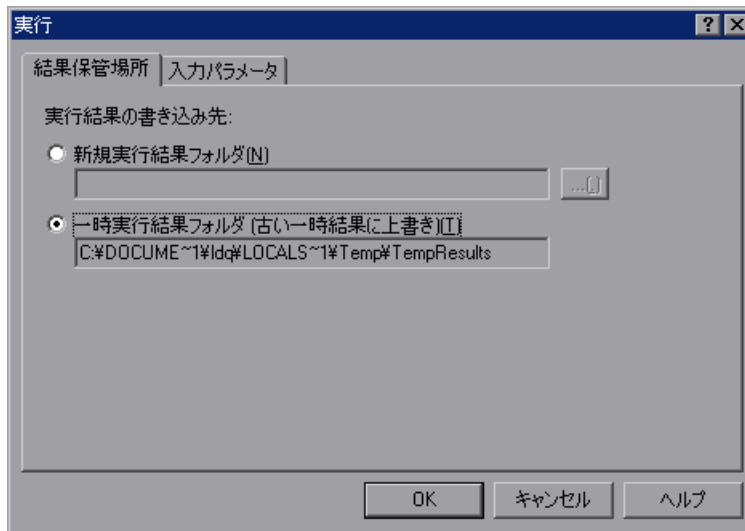
[ステップから開始]

コンポーネント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または Web サイトが開いていることを確認しておく必要があります。コンポーネントの編集中にコンポーネント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] > [ステップから開始] を選択します。または
 - ▶ 実行を開始するステップで右クリックし、ショートカット・メニューから [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。



- 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、490 ページ「[結果保管場所] タブについて」および 492 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。[ステップイントウ] コマンド、[ステップオーバー] コマンド、[ステップまで実行] コマンドなどの任意の 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] ウィンドウが開きます。実行結果の表示の詳細については、第 19 章「テスト結果の分析」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てあると、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 8 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



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。一時停止されたコンポーネン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。



一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] を選択します。あるいは、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法

ブレークポイントを使用することによって、コンポーネン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に指定できます。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コンポーネン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、サイトまたはアプリケーションの状態を点検する。
- ▶ ステップ・コマンドを使ってコンポーネン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、コンポーネン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、コンポーネン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、実行セッションをコンポーネン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。

- ▶ コンポーネン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。
- ▶ 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コンポーネン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、477 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、コンポーネン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、コンポーネン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、コンポーネン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、コンポーネン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横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丸のアイコン  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。



[デバッグ] > **[全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、コンポーネン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、それら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

現在のコンポーネン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。
- ▶ コンポーネン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解除] 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解除]** 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。

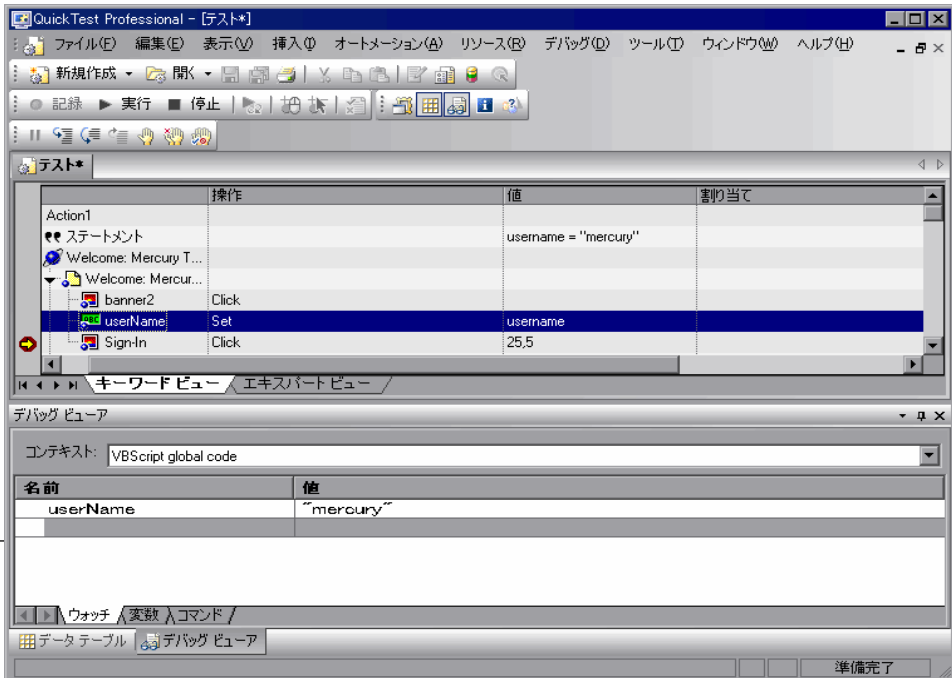
デバッグ・ビューアの使用

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。デバッグ・ビューアは、ビジネス・コンポーネント内の操作（関数）のデバッグに役立ちますが、ほかのタイプのコンポーネント・ステップでを使用することを目的としたものではありません。

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、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。

【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、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。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。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、変数の値を手作業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：関数ライブラリから [ウォッチ] タブに式を追加できます（ビジネス・コンポーネントからは追加できません）。

[変数] タブ

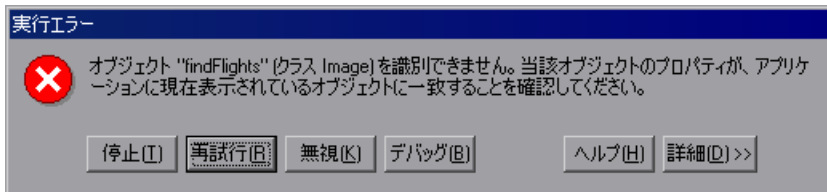
[変数] タブには、関数ライブラリが停止または一時停止するポイントまでの、現在の関数のすべての変数の現在の値が自動的に表示されます。たとえば、関数をステップ実行している場合、各ステップを実行すると、[変数] タブのグリッドにステップの変数の現在の値が追加されます。後続のステップを引き続き実行すると、変数に変更があった場合、[変数] タブに表示されている変数の値が自動的に更新されます。ブレークポイントで一時停止している間は、変数の値を手作業で変更することもできます。

[コマンド] タブ

[コマンド] タブは、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される [実行エラー] メッセージ・ボックスには、遭遇したエラーに対処するためのボタンがいくつか用意されています。



- ▶ [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 QuickTest 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、コンポーネント、およびコンポーネントによって呼び出される関数が含まれる関連する関数ライブラリのデバッグが可能になります。

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、コンポーネン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 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。
- ▶ **[詳細]**：メッセージ・ボックスを拡張し、エラーについてさらに詳細な情報を表示します。

アクションまたは関数のデバッグの練習

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関数を作成するとします。関数にブレークポイントを追加することで、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、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：新しい関数を作成する

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。関数を使った作業の詳細については、第 21 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。

VBScript コードを入力します。

```
Function SetVariables  
  
Dim a  
a="hello"  
b="me"  
MsgBox a  
  
EndFunction
```

手順 2 : 関数ライブラリをアプリケーション領域に関連付ける

- 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] <関数ライブラリ名> 'を' <アプリケーション領域名> 'に関連付ける] を選択するか、右クリックして [ライブラリ] <関数ライブラリ名> 'を' <アプリケーション領域名> 'に関連付ける] を選択します。QuickTest によって、開いているアプリケーション領域に関数ライブラリが関連付けられます。

手順 3 : コンポーネントに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、新しい操作を挿入し、[操作] リストから [SetVariables] を選択します。

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方法の詳細については、477 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 : コンポーネントの実行を開始する

コンポーネントを実行します。コンポーネント関数ライブラリは、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 が宣言される前にコンポーネントが停止したため、[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。
SetVariables (値は **Empty**) と変数 **a** (値は **hello**) の両方が表示されます。
変数 **b** が宣言される前にコンポーネントが停止したため、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、コンポーネントの実行を続けます。コンポーネン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、コンポーネントの実行を続けます。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 18 章

コンポーネントの実行

コンポーネントの作成後は、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。

- ▶ コンポーネントの実行について
- ▶ コンポーネント全体の実行
- ▶ コンポーネントの一部の実行
- ▶ コンポーネントの更新

コンポーネントの実行について

コンポーネントを実行すると、それに含まれるステップが **QuickTest** によって実行されます。コンポーネン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 19 章「テスト結果の分析」を参照してください。

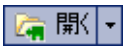
コンポーネント全体を先頭から実行することも、一部分だけを実行することも可能です。コンポーネントを更新して、テストのオブジェクト記述を変更できます。動的な記述を持つオブジェクトについてコンポーネントを実行することもできます。詳細については、第 13 章「テスト・オブジェクトを使用した作業」を参照してください。

コンポーネント全体の実行

QuickTest では、コンポーネントの実行を開始するステップを特に指定しない限り、常に最初のステップから開始されます。ステップを指定して、そこから（またはそこまで）コンポーネントを実行するには、**[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、コンポーネントを最初からまたは最後まで実行せず、コンポーネント内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、493 ページ「コンポーネントの一部の実行」を参照してください。

コンポーネントを開始するには、**[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、コンポーネントに定義したパラメータの値を入力します。

コンポーネントを実行するには、次の手順を実行します。

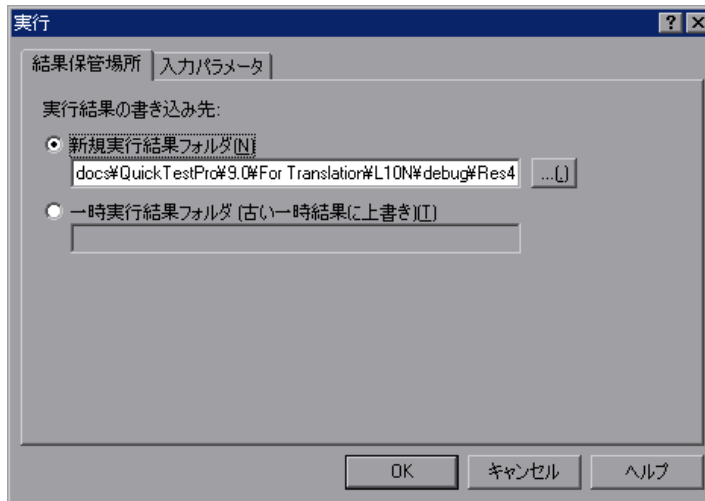


- 1 コンポーネントをまだ開いていない場合は、**[ファイル]** > **[開く]** > **[ビジネス/スクリプト コンポーネント]** を選択するか、**[開く]** ボタンをクリックして開きます。

ヒント：最近使用したコンポーネントを開く場合は、**[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。



- 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、490 ページ「[結果保管場所] タブについて」および 492 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]** ウィンドウが開きます。実行セッション結果の表示の詳細については、第19章「テスト結果の分析」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8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] を選択します。

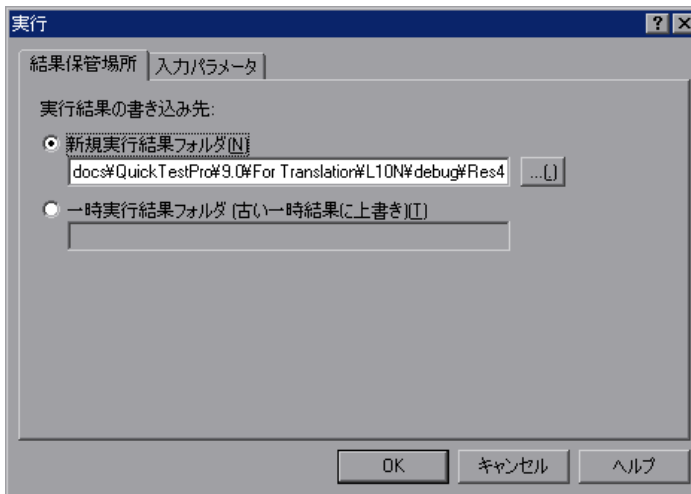


停止

[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。

[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。



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名前が表示されます。標準設定では、コンポーネントの結果はお使いのコンピュータの Quality Center キャッシュ・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。

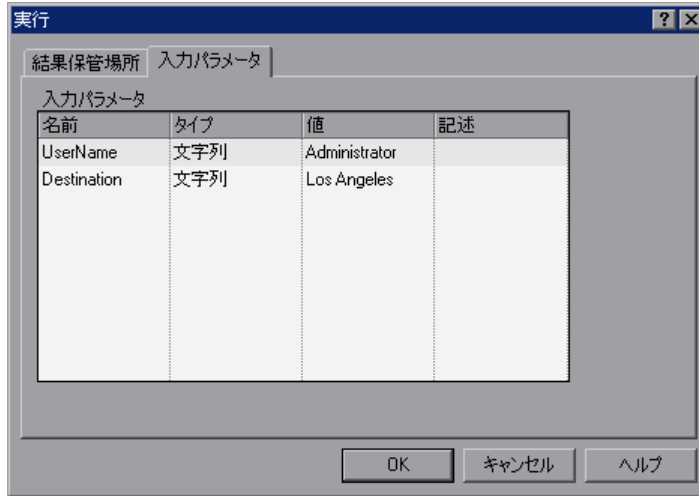
- ▶ **[一時実行結果フォルダ]**：一時フォルダに実行結果を保存します。このオプションによって、このフォルダに保存されている結果はすべて上書きされます。

注：QuickTest では、すべてのコンポーネントの一時的な実行結果が **<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。



【入力パラメータ】タブには、コンポーネン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 **【ビジネス コンポーネントの設定】** ダイアログ・ボックスの標準設定の値が使用されます。

コンポーネント・パラメータの設定については、239 ページ「コンポーネントのパラメータの定義」を参照してください。パラメータの使用については、172 ページ「パラメータの使用」を参照してください。

コンポーネントの一部の実行

選択したステップからコンポーネントの最後までコンポーネン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、コンポーネン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、コンポーネントをコンポーネン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場合は、**[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、473 ページ「**[ステップまで実行]** コマンドおよび **[ステップから開始]** コマンドの使用方法」を参照してください。

選択したステップからコンポーネントを実行するには、次の手順を実行します。

- 1 アプリケーションを開き、実行するステップの場所を表示します。
- 2 コンポーネントの実行を開始するステップを選択します。

選択したステップが、それよりも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]** を選択します。
- 4 490 ページ「**[結果保管場所]** タブについて」および 492 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。
- 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]** ウィンドウが開きます。実行セッション結果の表示の詳細については、第 19 章「**テスト結果の分析**」を参照してください。

テスト結果のサマリには、**[ステップから実行]** オプションを使用してコンポーネン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8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。

コンポーネントの更新

コンポーネントを更新すると、QuickTest によってコンポーネントが実行され、テスト・オブジェクト記述が更新されます。選択した実行オプションおよびデバッグ実行オプションに従って、コンポーネントの全体または一部のデータを更新できます。コンポーネントを保存した後は、更新されたデータが以降のテスト実行に使用されます。

QuickTest によってコンポーネントが更新されると、更新対象のオブジェクトが元は共有オブジェクト・リポジトリ内にあった場合でも、オブジェクトは常にローカル・オブジェクト・リポジトリに保存されます。次にコンポーネントを実行するとき、QuickTest では、共有オブジェクト・リポジトリよりもローカル・オブジェクト・リポジトリのほうが優先度が高いため、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]** を使用してコンポーネントを更新した後は、必要に応じて、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]** オプションを使用し、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・リポジトリに結合します。詳細については、を参照してください。



コンポーネントを実行して期待結果を更新するには、次の手順を実行します。

- 1 コンポーネントをまだ開いていない場合は、[ファイル] > [開く] > [ビジネス/スクリプト コンポーネント] を選択するか、[開く] ボタンの下矢印をクリックして、[ビジネス/スクリプト コンポーネント] を選択します。
- 2 コンポーネントを選択し、[OK] をクリックします。コンポーネントが開き、タイトル・バーにコンポーネントの名前が表示されます。

ヒント：最近使用したコンポーネントを開く場合は、[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。



- 3 [オートメーション] > [更新モード] を選択するか、[更新モード] ボタンをクリックします。[更新モード] のメニュー・オプションとボタンが切り替わり、次の手順で選択する実行の種類に応じてコンポーネントが更新されます。
- 4 次のオプションから、実行の種類を選択します。



- ▶ [オートメーション] > [実行]：コンポーネント全体を実行して更新します。これは [実行] ボタンをクリックするのと同じです。
- ▶ [オートメーション] > [ステップから実行]：コンポーネント内の選択したステップから、そのコンポーネントの最後まで実行して更新します

次のデバッグ実行モードのいずれかを選択し、コンポーネントをデバックして更新することもできます。



- ▶ [デバッグ] > [ステップイントゥ] を選択、または [ステップイントゥ] ボタンをクリック：コンポーネントの現在のステップのみをデバックして更新します。



- ▶ [デバッグ] > [ステップオーバー] を選択、または [ステップオーバー] ボタンをクリック：コンポーネントの現在のステップのみをデバックして更新します。

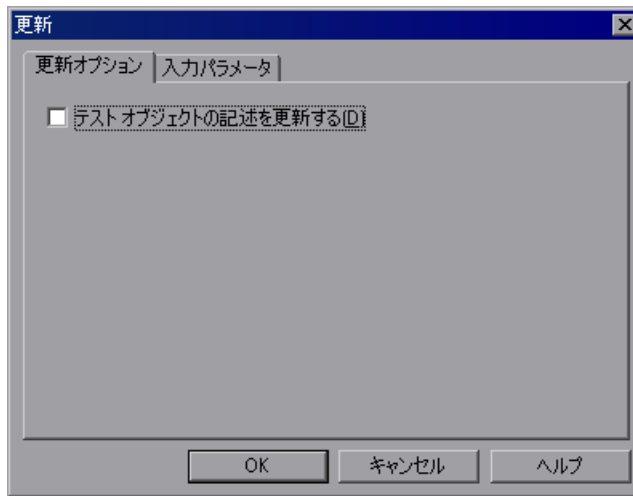


- ▶ [デバッグ] > [ステップアウト] を選択、または [ステップアウト] ボタンをクリック：コンポーネントの現在のステップをデバックして更新します。

- ▶ **[デバッグ]** > **[ステップまで実行]** : コンポーネントまたは関数ライブラリの現在の場所から選択したステップに到達するまで、コンポーネントをデバッグして更新します。
- ▶ **[デバッグ]** > **[ステップから開始]** : 選択したステップからコンポーネントの最後まで、コンポーネントをデバッグして更新します

デバッグの詳細については、第17章「コンポーネントと関数ライブラリのデバッグ」を参照してください。

[更新] ダイアログ・ボックスが表示されます。



- 5 実行の更新プロセスの設定を行います。詳細については、498 ページ「[更新オプション] タブについて」および 492 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 : 実行の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。

- 6 **[OK]** をクリックします。**[更新]** ダイアログ・ボックスが閉じ、QuickTest によってコンポーネントの更新が開始されます。コンポーネントの更新中は、ステータス・バーで「**更新**」というテキストが点滅します。

コンポーネントが実行され、テスト・オブジェクト記述が更新されます。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。結果の表示の詳細については、第19章「テスト結果の分析」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、実行の更新セッションの最後に [テスト結果] ウィンドウが開きません。[オプション s] ダイアログ・ボックスの詳細については、第8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：コンポーネントの更新が終了したら、**[更新モード]** ボタンをクリックして、更新モードでの実行を必ず終了してください。

実行の更新が終了すると、[テスト結果] ウィンドウには更新したテスト・オブジェクト記述が表示されます。たとえば、次のように表示されます。

ステップ名: Notifications-Update

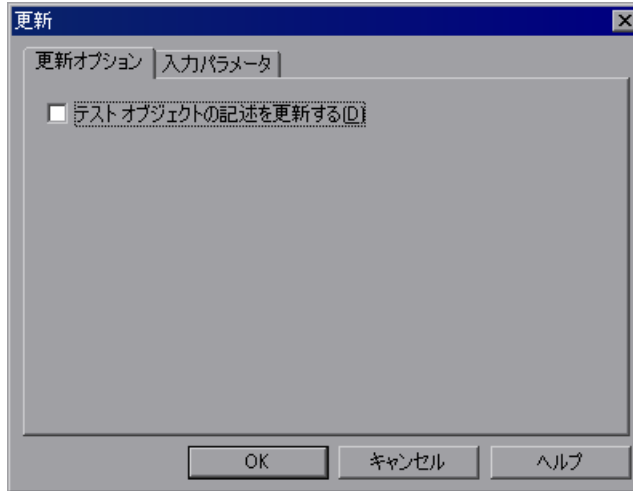
Description

ステップ* 完了

オブジェクト	詳細	結果	時間
Notifications-Update Description	Test object's previous description: Selection = Low Mercury Class = WebList Test object's new description: Mercury Class = WebList Attached Text =	完了	2004/06/10 - 11:05:21

[更新オプション] タブについて

[更新オプション] タブでは、テストのオブジェクト記述を更新できます。更新されたコンポーネントの結果は、以降の実行に使用されます。



[更新オプション] タブには、[テスト オブジェクトの記述を更新する] チェック・ボックスがあります。このオプションを選択すると、各オブジェクト・クラスに対して [オブジェクトの認識] ダイアログ・ボックスで現在定義されているプロパティに従って、ビジネス・コンポーネントのテスト・オブジェクト記述が更新されます。このオプションを使用して、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。このオプションを使用すると、(パラメータ化されている場合や正規表現が使用されている場合でも) すべての値が更新されます。

ヒント: 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] オプションを使用して、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、324 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。

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コンポーネントのコピーを保存し、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）を使用してコンポーネン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。コンポーネン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特定部分の英語版を対象にコンポーネン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・プロパティの値に基づいて認識されますが、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対象部分について、フランス語版でも同じコンポーネン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識別に使用されます。たとえば、リンク・オブジェクトの識別に **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対象部分の英語版に対して、定義した新しいプロパティを使用して実行の更新を行います。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コンポーネントを正しく実行できるようになります。

ヒント：正しく実行できるコンポーネント内で、オブジェクトの識別にスマート認識が使用されている場合は、**[テストオブジェクトの記述を更新する]** オプションを使用してテスト・オブジェクト記述のプロパティ値を更新することもできます。

[**テストオブジェクトの記述を更新する**] を選択してコンポーネントを実行すると、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・オブジェクトの識別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。

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。

第 19 章

テスト結果の分析

コンポーネン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。

注：QuickTest から [テスト結果] ウィンドウを開いた場合は、ビジネス・プロセス・テスト実行結果を表示することはできません。ビジネス・プロセス・テストの実行結果を表示するには、表示する反復結果を選択して Quality Center から開きます。

本章では、次の項目について説明します。

- ▶ テスト結果の分析について
- ▶ [テスト結果] ウィンドウについて
- ▶ 実行セッションの結果の表示
- ▶ [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
- ▶ テスト結果に含まれるスマート認識情報の分析
- ▶ テスト結果の削除
- ▶ Quality Center プロジェクトへの、実行セッション時に検出された不具合の手作業による送信
- ▶ テスト結果の表示のカスタマイズ

テスト結果の分析について

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。単一のテスト反復が表示されます。

コンポーネントの実行が終わると、[テスト結果] ウィンドウに、テスト実行のあらゆる側面が表示されます。

- ▶ 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
- ▶ コンポーネント内で不具合が発生した正確な位置
- ▶ コンポーネントの各段階で、成功または失敗した各ステップの詳細な説明

注：[テスト結果] ウィンドウは、結果をツリー階層で 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、手作業で **results.xml** ファイルを開いてレポート全体を確認できます。

[テスト結果] ウィンドウについて

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、199 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。

注：[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、[結果] ボタンをクリックするか、[スタート] メニューの [QuickTest Professional] プログラム・グループから [Test Results Viewer] を選択します。

コンポーネントの実行結果の例は次のとおりです。

The screenshot shows the 'Test Results' window for a script component named 'ビジネス コンポーネント [Res1] - テスト結果'. The window is divided into a left sidebar and a main content area.

Left Sidebar (Test Results Tree):

- テスト結果 ツールバー
- 実行時データ テーブル
- ログイン
- 代理店名.SetText
- 代理店名.Type
- パスワード.SetSecure
- OK.Click
- フライト予約
- MaskedTextBox.Type
- 出発地.Select
- 到着地.Select
- FLIGHT.Click
- フライト テーブル
- 出発地.Select
- OK.Click
- フライト予約.close

Main Content Area (Script Component Results Summary):

ビジネス コンポーネント: スクリプト コンポーネント
 製品名: QuickTest Professional
 結果名: Res1
 タイムゾーン: 東京 (標準時)
 実行開始: 2006/04/24 - 11:42:15
 実行終了: 2006/04/24 - 11:42:43
 結果: 完了

ステータス	回
成功	0
失敗	0
警告	0

ステータスバー: [F1] キーを押すと、ヘルプが表示されます。

注：この例では、コンポーネントは、関連付けられている関数ライブラリ内での実行エラーによって失敗しました。実行エラーが発生しなかった場合、**[結果]** は **[完了]** を示します。








[テスト結果] ウィンドウには、次の主要な要素があります。

- ▶ **テスト結果タイトル・バー**：コンポーネントの名前が表示されます。
- ▶ **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。
- ▶ **テスト結果ツールバー**：実行セッション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、506 ページ「テスト結果ツールバー」。
- ▶ **テスト結果ツリー**：実行結果がテスト結果ツリーに視覚的に表示されます。詳細については、次に示す「テスト結果ツリー」を参照してください。
- ▶ **テスト結果詳細**：選択したステップの詳細が表示されます。詳細については、505 ページ「テスト結果詳細」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、506 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。

テスト結果ツリー

[テスト結果] ウィンドウの左側の表示枠には、**テスト結果ツリー**が表示されず。テスト結果ツリーには、実行セッション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。このアイコンは、コンポーネント・ステップに次が含まれている場合にのみ表示されます。
- ▶ **VerifyProperty** などの検証操作（関数）
- ▶ **micPass** ステータスの **AddToTestResults**（またはそれと同等のもの）
- ▶  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（ルートアクションまたはテストまで）が失敗となります。
- ▶  は警告を示します。これは、そのステップは成功しなかったが、コンポーネントの失敗には至らなかったことを意味します。
- ▶  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。
- ▶  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、終了する前に停止されたことを示します。

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

テスト結果詳細

標準設定では、[テスト結果] ウィンドウが開くと、コンポーネント・サマリがウィンドウの右側の表示枠に表示されます。ここには、コンポーネントの名前、製品の名前（コンポーネントの場合）、結果名、テストを実行した日時、および反復が成功したか失敗したかが表示されます。コンポーネントの場合、起こりうる結果は **[完了]** または **[失敗]** です。

ツリーの分岐またはステップを選択すると、右ウィンドウに、選択した項目の詳細情報が表示されます。

テスト結果ツールバー

テスト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] ウィンドウの概観は 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、27 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、199 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、現在のコンポーネントの以前の実行結果や、他のコンポーネントの結果を表示することもできます。さらに、実行セッション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。

テストの実行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、「結果」ボタンをクリックするか、「スタート」メニューの「QuickTest Professional」プログラム・グループから「Test Results Viewer」を選択します。

- ▶ 現在のコンポーネントに関する実行セッション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、503 ページ「[テスト結果] ウィンドウについて」を参照してください。
- ▶ 現在のコンポーネントに関する実行セッション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。任意のコンポーネントについて実行セッション結果を選択するか、ファイル・システムに存在する実行セッション結果（**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。実行セッション結果の表示の詳細については、511 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。

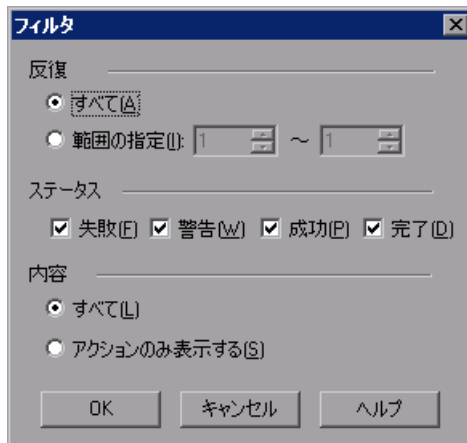
- 2 ツリーに表示される詳細のレベルを選択するには、テスト結果ツリーの分岐を折りたたむか、展開します。
 - ▶ 分岐を閉じるには、その分岐を選択して分岐のアイコンの左にある折りたたみ (-) 記号をクリックするか、キーボードのテンキー部分にあるマイナス・キー (-) を押します。結果ツリーでは、その分岐の詳細が消えて、折りたたみ記号が展開記号 (+) に変わります。
 - ▶ テスト結果ツリーのすべての分岐を折りたたむには、**[表示] > [すべて折りたたみ]** を選択するか、または分岐を右クリックして **[すべて折りたたみ]** を選択します。
 - ▶ 分岐を展開するには、その分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。
 - ▶ テスト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 反復、またはステップの結果を表示できます。結果は、次の3種類のいずれかになります。
 - ▶ 成功しなかったが、コンポーネントの実行停止には至らなかったステップは、[テスト結果] ウィンドウの右下に「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コンポーネントでも、「完了」と表示される場合があります。



- 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。



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。

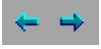
[ステータス] 領域：

- ▶ [失敗]：失敗したステップの実行結果を表示します。
- ▶ [警告]：ステータスが**警告**のステップ（成功はしなかったが、コンポーネントが失敗する原因にはならなかったステップ）に関する実行結果を表示します。
- ▶ [成功]：成功したステップの実行結果を表示します。
- ▶ [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）に関する実行結果を表示します。

注：[反復] と「内容」領域は、コンポーネントの場合は該当しません。



- 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。



- 6 テスト結果ツリー内で以前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。



- 7 他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、511 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 実行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、513 ページ「実行セッション結果の印刷」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、533 ページ「Quality Center プロジェクトへの、実行セッション時に検出された不具合の手作業による送信」を参照してください。

- 9 **[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。

特定の実行を表示するためのテスト結果の表示

現在のコンポーネントに関して保存されている結果を表示できます。また、他のコンポーネン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、表示する実行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。
- ▶ 現在のコンポーネントに関して複数の結果がある場合、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。



現在のコンポーネン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、セットを選択して [開く] をクリックします。

ヒント：指定したコンポーネン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。

他のコンポーネントの実行結果を表示するには、コンポーネント結果ファイルを検索します。

注：QuickTest から [テスト結果] ウィンドウを開いた場合は、ビジネス・プロセス・テスト実行結果を表示することはできません。ビジネス・プロセス・テストの実行結果を表示するには、表示する反復結果を選択して Quality Center から開きます。

ファイル・システムでの結果の検索

標準設定では、コンポーネントの実行結果はコンピュータの Quality Center キャッシュ・フォルダに保存されます。コンポーネントを実行するとき、[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、490 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・システムではコンポーネン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・システムの中でコンポーネン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、コンポーネン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コンポーネン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、表示するコンポーネン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 コンポーネン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。

- 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。

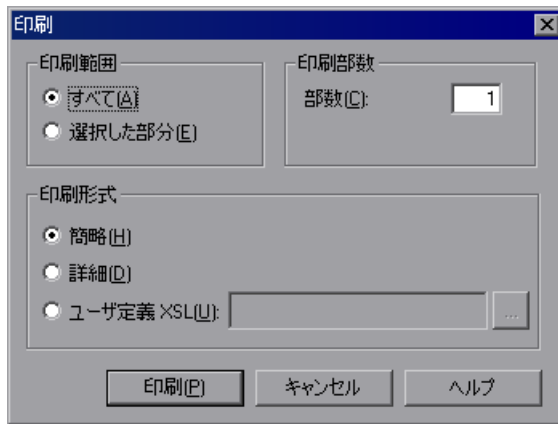
実行セッション結果の印刷

[テスト結果] ウィンドウから、実行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。

実行結果を印刷するには、次の手順を実行します。



- [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。



- [印刷範囲]** オプションを選択します。
 - ▶ **[すべて]** : コンポーネント全体の結果を印刷します。
 - ▶ **[選択した部分]** : テスト結果ツリーで選択した分岐の実行結果情報を印刷します。
- [印刷部数]** で、印刷する実行結果の部数を指定します。
- [印刷形式]** オプションを選択します。
 - ▶ **[簡略]** : テスト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、テスト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、534 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 以降で作成した実行結果にのみ使用できます。

- 5 **[印刷]** をクリックして、選択した実行結果情報を、通常使う Windows プリンタに出力します。

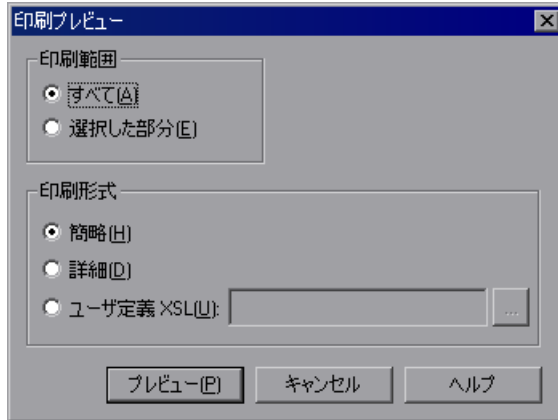
テスト結果のプレビュー

実行結果は、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、情報をユーザ定義の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 以降で作成した実行結果にのみ使用できます。

実行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。



- 2 **[印刷範囲]** オプションを選択します。
 - ▶ **[すべて]** : コンポーネント全体の実行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実行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]** : テスト結果ツリーの各項目のサマリ行 (使用可能な場合) をプレビューします。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、テスト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プレビューします。
 - ▶ **[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、534 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、画面に実行結果のプレビューを表示します。



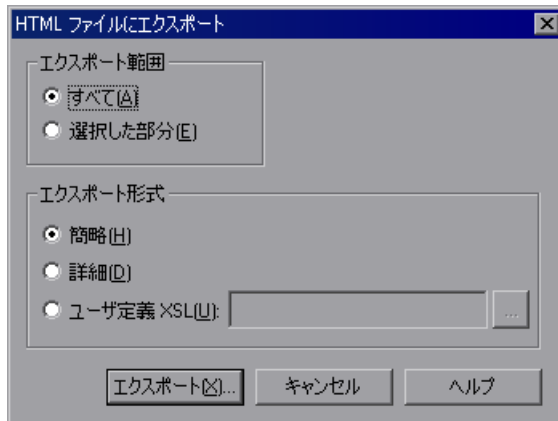
ヒント：プレビューに表示されない情報がある場合は（たとえば、チェックポイント名が長すぎてディスプレイに表示されないなど）、[印刷プレビュー] ウィンドウの [ページの設定] ボタンをクリックして、ページの向きを [縦] から [横] に変更します。

テスト結果のエクスポート

[テスト結果] ウィンドウから HTML ファイルに実行結果をエクスポートできます。この機能により、QuickTest 環境を使用していないときも、実行結果を容易に確認できます。たとえば、実行結果を取めた HTML 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。

実行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。



2 **[エクスポート範囲]** オプションを選択します。

- ▶ **[すべて]** : コンポーネン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実行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]** : テスト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、テスト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、534 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 バージョン 8.0 以降で作成した実行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<コンポーネント名> [<実行結果名>] という名前が付けられ、実行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、コンポーネント内から値が割り当てられる変数です。[テスト結果] ウィンドウにコンポーネン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 コンポーネントの実行結果を [テスト結果] ウィンドウに表示します。詳細については、507 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、コンポーネントの名前を含んだルート・ノードを選択します。

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。

InputParam Results Summary

Business Component: InputParam
Product name: QuickTest Professional
Results name: TempResults
Time Zone: Eastern Standard Time
Run started: 12/20/2005 - 13:58:44
Run ended: 12/20/2005 - 13:59:00

Result: Done

Status	Times
Passed	0
Failed	0
Warnings	0

Input Parameters	Value
UserName	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」を使ってコンポーネント用に定義された入力パラメータ「UserName」を示します。




コンポーネントでのパラメータの定義と使用については、『172 ページ「パラメータの使用」』を参照してください。

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一度で識別できず、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識別を試みます。次の例では 2 つのシナリオを説明します。

スマート認識—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：  "WebEdit"- Description mismatch	オブジェクト（たとえば、 <code>userName</code> WebEdit オブジェクト）が見つからなかったことを示します。
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：  "WebEdit"- Smart Identification	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で、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、記録されているテスト・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なります。
実際に実行されたステップ。 例：  "userName".Set "jackieros"	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 14 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 オブジェクトを識別したコンポーネントの結果を示します。

The screenshot shows a window titled "スクリプト コンポーネント [Res2] - テスト結果". The main content area displays a warning for a step named "userName". The warning message is "Step Name: 'userName' - Description mismatch". Below this, a table provides details of the error.



Object	Details	Result	Time
"userName"-Description mismatch	The object was not found using the test object description. Check the object's properties.	Warning	12/21/2005 - 11:40:46

At the bottom of the window, there is a status bar with the text "[F1] キーを押すと、ヘルプが表示されます。" and a "準備完了" button.

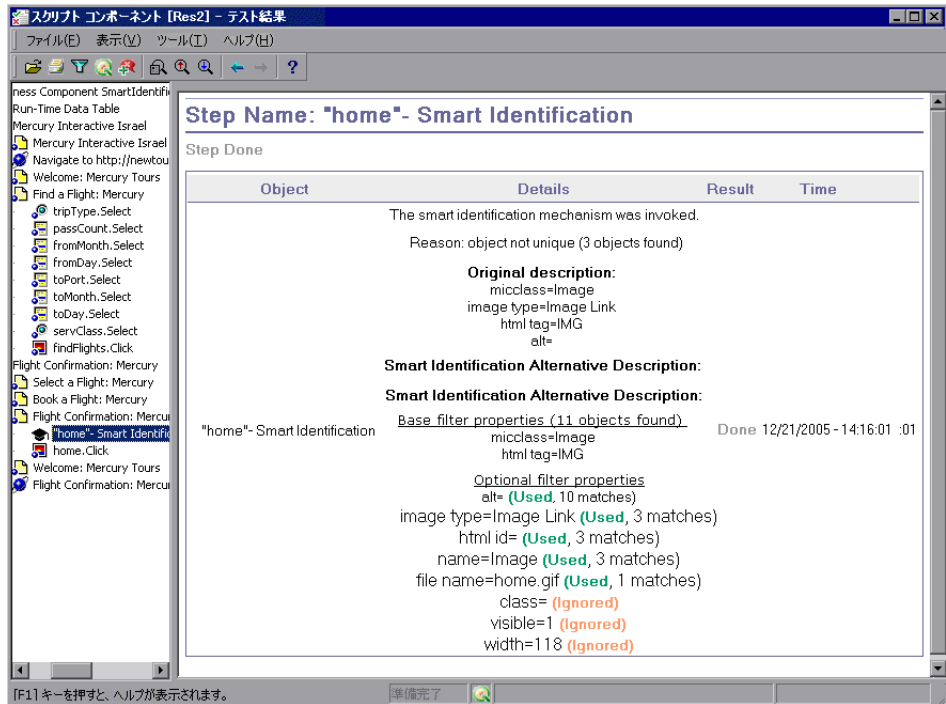
スマート認識—記録されている記述に複数のオブジェクトが一致する場合

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識別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
<p>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：  "home"- Smart Identification</p>	<p>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なります。</p>
<p>実際に実行されたステップ。 例：  "home":Click</p>	<p>実行されたステップの詳細を示す通常の結果。</p>

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識別したコンポーネン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識別できない場合、コンポーネントは失敗し、テスト結果に通常の失敗ステップが表示されます。

テスト結果の削除

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実行結果や古い実行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。

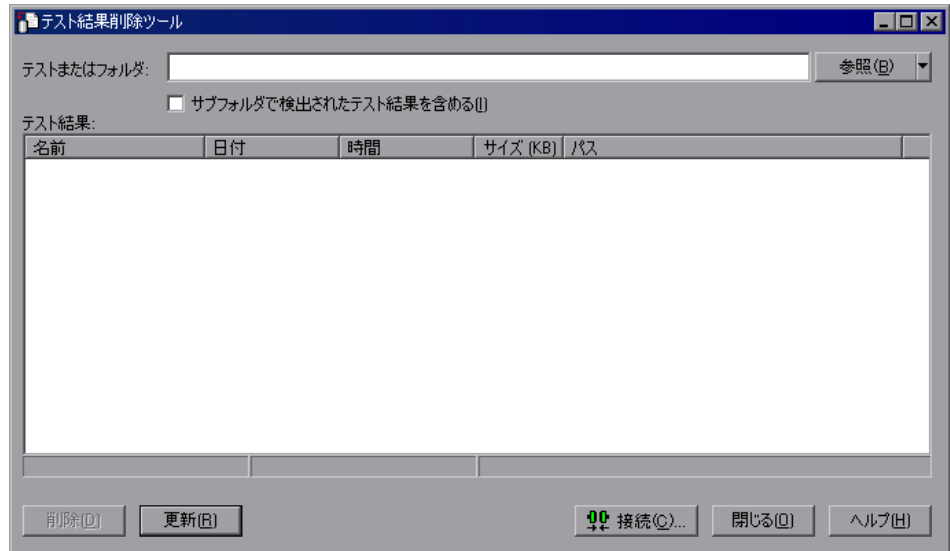
テスト結果削除ツールを使った結果の削除

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、すべての実行セッション結果の一覧を表示できます。その後、不要な実行結果を削除できます。

削除対象の実行結果を簡単に特定できるように、テスト結果削除ツールでは、結果を名前、日付、サイズなどに基づいて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実行結果を削除するには、次の手順を実行します。

- 1 **[スタート]** メニューの **[QuickTest Professional]** プログラム・グループから **[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。



- 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、20 ページ「Quality Center プロジェクトへの接続」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には、**[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が対象の場合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び替えられます。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なります。

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。続いてスペースを 1 つ入力してから、使用するコマンド・ライン・オプションを入力します。詳細については、次の「コマンド・ライン・オプション」を参照してください。

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テスト結果ファイルごとの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格納されているフォルダ内の **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格納されてい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・システム内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にはサポートされません。

次の例では、ファイル・サイズが 10,000 バイト以上のテスト結果がすべて削除されます。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。指定し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、『**QuickTest Professional 基本機能ユーザーズ・ガイド**』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** の各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 "_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

指定したフォルダおよびその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格納されてい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・システム内のフォルダに対してのみ使用できます。Quality Center のテストが対象の場合にはサポートされません。

-Server < Quality Center サーバのパス >

接続先となる Quality Center サーバのフルパスを指定します。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を使用して削除するテスト結果に対応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 のフル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 に接続するために、**-Domain**, **-Server**, **-Project**, **-User**, **-Password** の各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 **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"
```

注：-Test オプションと -Recursive オプションを組み合わせると、指定した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 <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。

Quality Center プロジェクトへの、実行セッション時に検出された不具合の手作業による送信

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。

Quality Center および QuickTest を使った作業の詳細については、『**QuickTest Professional 上級機能ユーザーズ・ガイド**』を参照してください。Quality Center の詳細については、『**Quality Center ユーザーズ・ガイド**』を参照してください。

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。



- 1 [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、20 ページ「Quality Center プロジェクトへの接続」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。



- 2 [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 [新規不具合] ダイアログ・ボックスを開きます。[新規不具合] ダイアログ・ボックスが開きます。
- 3 必要に応じて不具合の情報を変更できます。コンポーネントに関する基本情報は詳細領域に含まれています。

オペレーティング システム： Windows XP
 テスト パス： [QualityCenter] Components#hasegawa#hasegawaReservation

- 4 [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。
- 5 [閉じる] をクリックし、ダイアログ・ボックスを閉じます。

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。この **.xml** ファイルには、表示されているそれぞれのテスト結果ノードに関する情報が保存されています。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。

テスト結果ツリーの各ノードは **results.xml** ファイル内の1つの要素に対応します。このほかに、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。

スクリーンショット: スクリプト コンポーネント [Res2] - テスト結果

左側のツリー構造:

- ビジネス コンポーネント スクリプト
 - 実行時データテーブル
 - ログイン
 - 代理店名.SetText
 - 代理店名.Type
 - パスワード.SetSecure
 - OK.Click
 - フライト予約
 - MaskEdBox.Type
 - 出発地.Select
 - 到着地.Select
 - FLIGHT.Click
 - フライトテーブル
 - 出発地.Select
 - OK.Click
 - フライト予約.Close

右側の「結果サマリ」:

ビジネス コンポーネント: スクリプト コンポーネント
 製品名: QuickTest Professional
 結果名: Res2
 タイムゾーン: 東京 (標準時)
 実行開始: 2006/04/24 - 11:46:35
 実行終了: 2006/04/24 - 11:47:00
 結果: 完了

ステータス	回
成功	0
失敗	0
警告	0

左側のラベル:

- レポート要素
- 変更要素
- アクション要素
- Tname 要素
- Res 要素
- サマリ要素の sTime および eTime 属性
- ステップ要素
- コンポーネントの実行サマリ要素

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) は変更することができます。詳細については、506 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 ファイルに変更を加えて、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、別の要素タグには実行セッションが開始される時間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、時間の情報は全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。これらのファイルは **< 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 [選択した部分] 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、513 ページ「実行セッション結果の印刷」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、516 ページ「テスト結果のエクスポート」を参照してください。HTML ファイルにエクスポート

XML スキーマの構造の詳細、およびテスト結果レポートのカスタマイズに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』（英語版のみ）。**< QuickTest Professional のインストール・フォルダ > %help%XMLReport.chm** にあります）を参照してください。

第 6 部

高度な機能を使用した作業

第 20 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、次の内容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
- ▶ 回復シナリオを使用するタイミングの決定
- ▶ 回復シナリオの定義
- ▶ 回復シナリオ・ウィザードについて
- ▶ 回復シナリオの管理
- ▶ アプリケーション領域用の回復シナリオ・リストの設定
- ▶ プログラムによる回復メカニズムの制御

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、コンポーネントを自動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、コンポーネントが一時停止状態になるからです。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、542 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。回復シナリオは、予期しないイベントと、テストの実行を回復するために必要な操作の定義です。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、コンポーネントを続行するよう 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**：コンポーネントの実行を続行するために実行しなければならない操作です。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コンポーネン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、コンポーネン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、コンポーネン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、まずそのコンポーネントに（アプリケーション領域を通じて）回復シナリオを関連付ける必要があります。コンポーネン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。コンポーネン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、112 ページ「アプリケーション領域の回復シナリオ設定の定義」を参照してください。

回復シナリオを定義した対象のコンポーネントを実行して、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。

コンポーネントに 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、578 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： [ビジネス コンポーネントの設定] 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、コンポーネン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、かつコンポーネン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、コンポーネン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

イベントがコンポーネン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** ステートメントをユーザ定義関数に追加したりなどを追加して、コンポーネン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場合は、メッセージ・ボックスが開くと [**OK**] をクリックする **If** ステートメントをユーザ定義関数に追加することでこのイベントを処理できます。コンポーネン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に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。標準設定では、回復処理はステップがエラーを返した後にのみ有効になります。つまり、実際にエラーの原因となったステップよりもいくつか後のステップで行われる可能性があります。また、ステップごとにトリガ・イベントを確認すれば、パフォーマンスが低下します。

回復シナリオは、予期しないイベントや、コンポーネント内の特定のステップと同期化できないイベントにのみ使用します。たとえば、回復シナリオは [プリンタエラー] メッセージ・ボックスの標準設定のボタンをクリックすることによってプリンタ・エラーを処理できます。ネットワークがプリンタ・エラーを返すのがどのポイントか知ることは不可能なので、このエラーはコンポーネント内から直接処理することはできません。コンポーネントでこのイベントを処理するには、ユーザ定義関数の中で、プリンタにファイルを送るステップの直後に **If** ステートメントを追加します。しかし、ネットワークがプリンタ・エラーを返すまでに時間がかかると、コンポーネントはエラーが表示されるまでにステップがいくつか進んでしまう場合があります。したがって、この種のイベントの場合は、回復シナリオだけが処理できることになります。

回復シナリオの定義

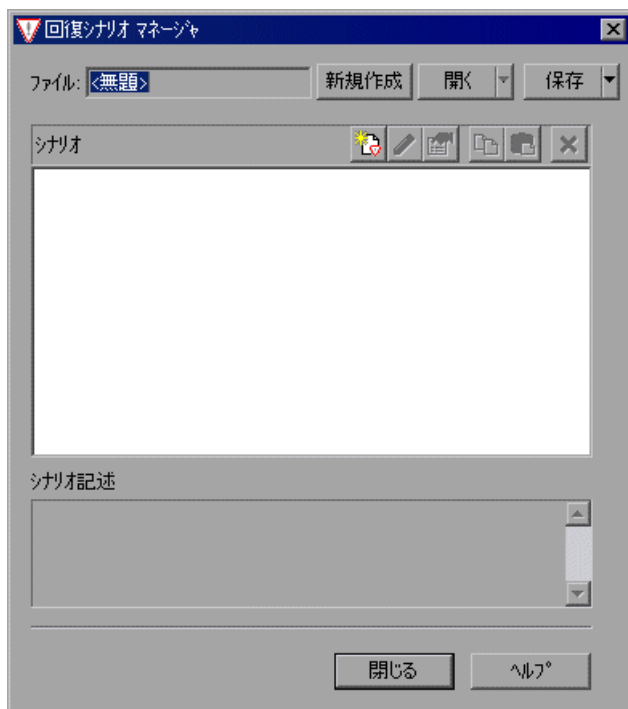
[回復シナリオ マネージャ] ダイアログ・ボックスでは、回復シナリオを作成し、回復ファイルにシナリオを保存できます。回復シナリオ・ウィザードを使用して回復シナリオを作成します。このウィザードは、回復シナリオの各段階の定義プロセスを案内します。回復ファイルに回復シナリオを保存し、特定のコンポーネントと関連付けます。

回復ファイルの作成

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。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。





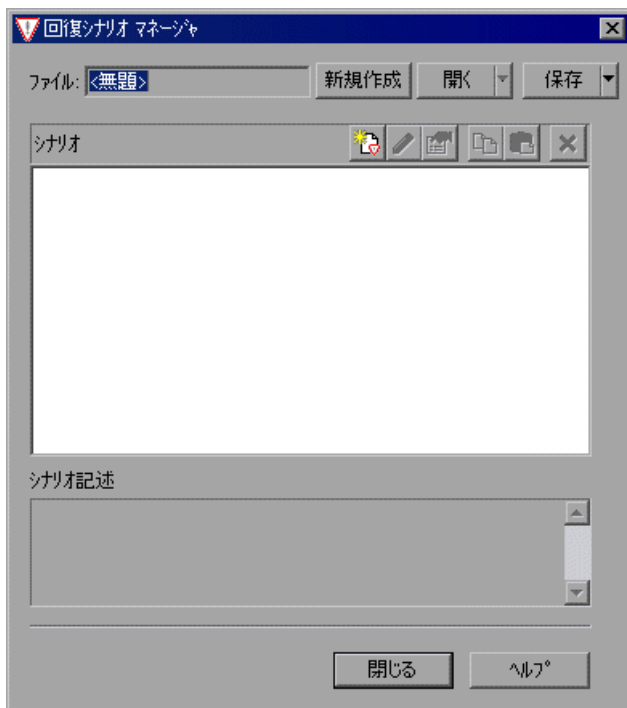
- 標準設定では、[回復シナリオ マネージャ] ダイアログ・ボックスには新規の回復ファイルが表示されます。この新規ファイルを使用するか、[開く] ボタンをクリックし、既存の回復ファイルを選択します。あるいは、[開く] ボタンの横にある矢印をクリックして、最近使用した回復ファイルをリストから選択します。

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。

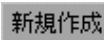

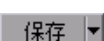






【回復シナリオ マネージャ】 ダイアログ・ボックスについて

【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] 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
	新規の回復ファイルを作成します。詳細については、543 ページ「回復ファイルの作成」を参照してください。
	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。
	現在の回復ファイルを保存します。詳細については、569 ページ「回復ファイルへの回復シナリオの保存」を参照してください。
	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、547 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。
	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、572 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サマリ・プロパティを読み取り専用形式で表示します。詳細については、571 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、573 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、573 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、573 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、570 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 5 つの主要なステップが含まれています。

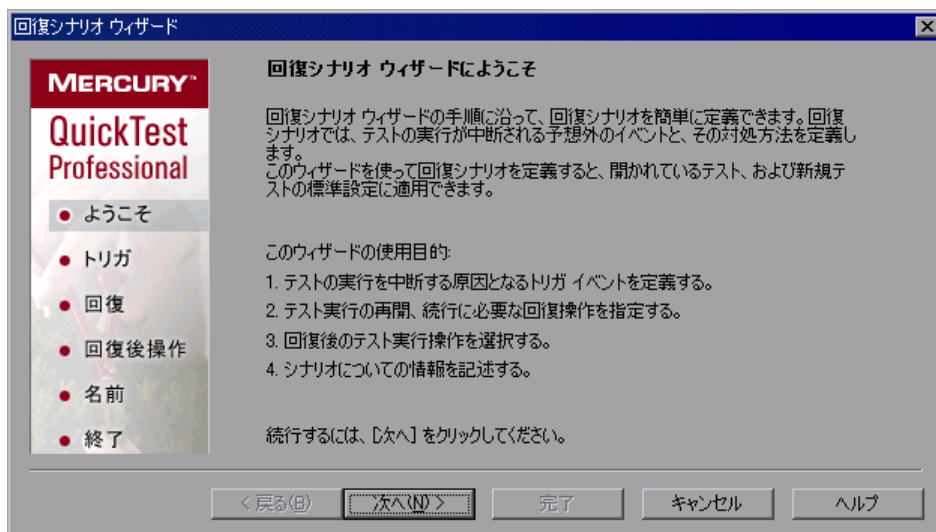
- ▶ 実行セッションを中断するトリガ・イベントの定義
- ▶ 継続に必要な回復操作の指定
- ▶ 回復後のテスト実行の操作の選択
- ▶ 回復シナリオの名前と説明の指定



回復シナリオ・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 ウィザードによるこそ] 画面

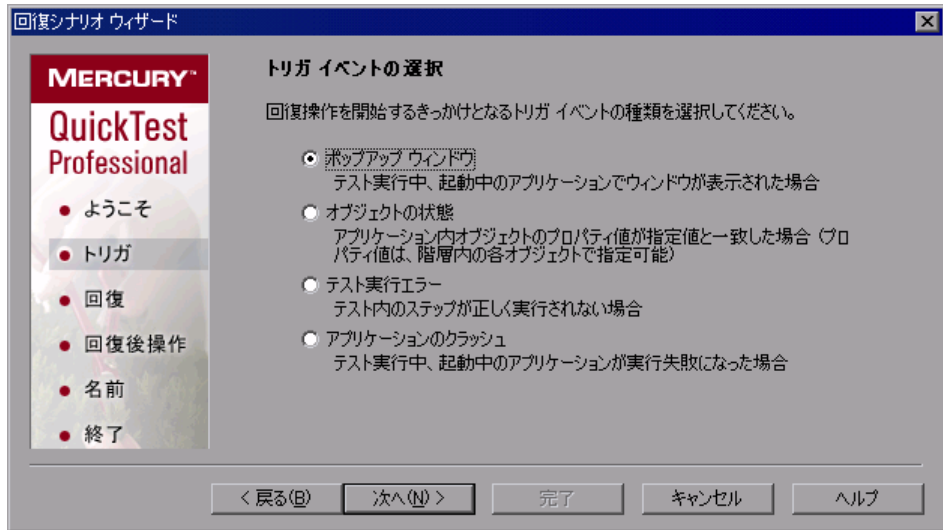
[回復シナリオ ウィザードによるこそ] 画面には、回復シナリオ・ウィザードのさまざまなオプションについての一般的な情報と、回復シナリオの定義に関連する段階の概要が表示されます。



[次へ] をクリックし、[トリガ イベントの選択] 画面に進みます。

【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。



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、次のどのトリガのタイプを選択したかによって異なります。

- ▶ **【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。

- ▶ **【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。

- ▶ **[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、[テスト実行エラーの選択] 画面に進みます。

- ▶ **[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がある、このクラッシュによって実行セッションを失敗させないように、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 [プロセスの選択] 画面に進みます。

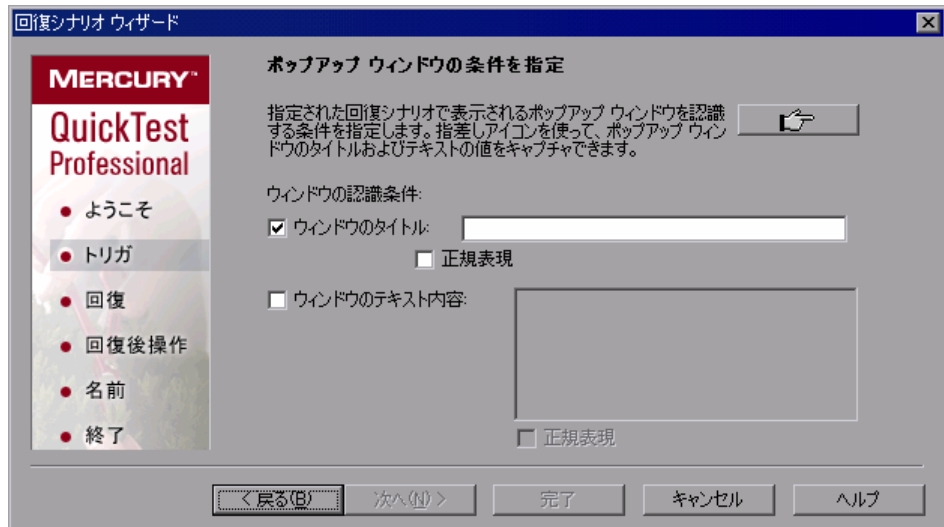
注：

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、コンポーネン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。コンポーネン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、コンポーネン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、[ポップアップ ウィンドウの条件を指定] 画面が開きます。



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、『**QuickTest Professional 基本機能ユーザーズ・ガイド**』を参照してください。
- ▶ 指差しマークをクリックした後ポップアップ・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。

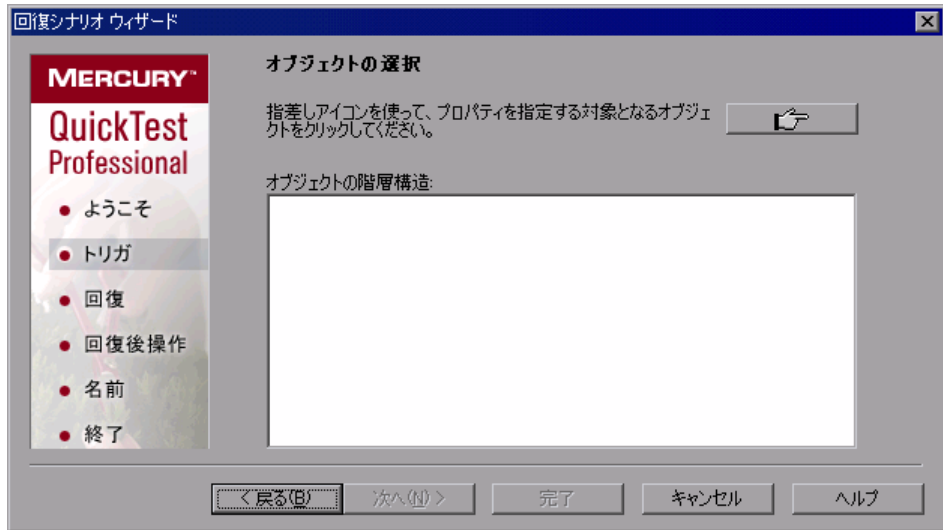
注：前者（[**ウィンドウのタイトル**] または [**ウィンドウのテキスト内容**], あるいはその両方）を実行した場合、QuickTest は指定したタイトルまたはテキスト、あるいはその両方を含んでいる任意のポップアップ・ウィンドウを識別します。後者（指差しマーク）を実行した場合、QuickTest は選択したウィンドウのオブジェクト・プロパティ値に一致するポップアップ・ウィンドウのみを識別します。

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マウスオーバーなどの操作を実行したりするには、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、左の CTRL キーを押したまま、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。

【オブジェクトの選択】画面

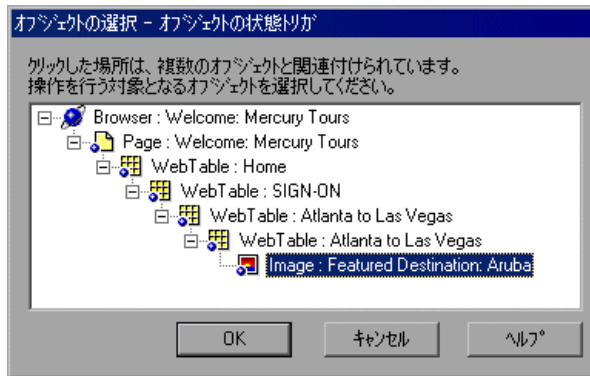
【トリガ イベントの選択】画面にある【オブジェクトの状態】トリガを選択すると、【オブジェクトの選択】画面が開きます。



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マウスオーバーなどの操作を実行したりするには、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、左の CTRL キーを押したまま、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。



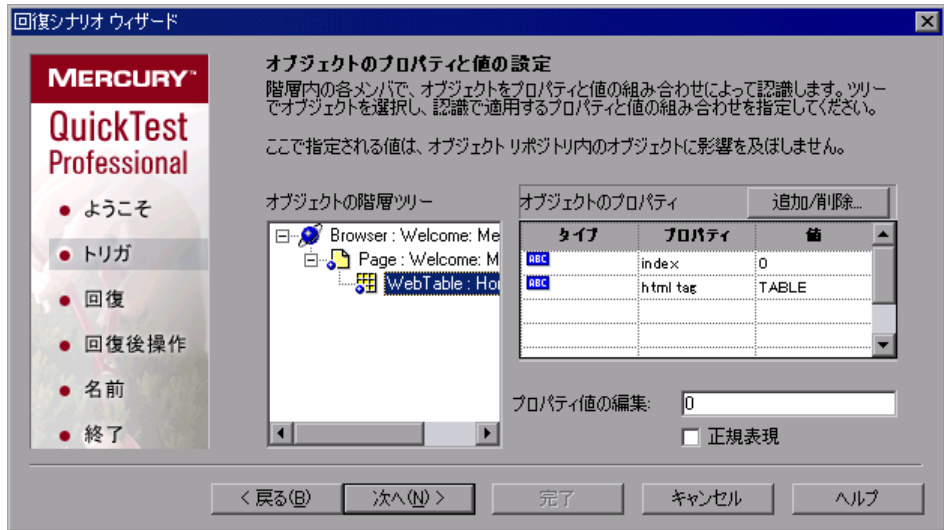
指定するプロパティを含んでいるオブジェクトを選択し、[OK] をクリックします。選択したオブジェクトとその親が [オブジェクトの選択] 画面に表示されます。

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 が通常は記録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。

【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、【オブジェクトのプロパティと値の設定】画面が開きます。



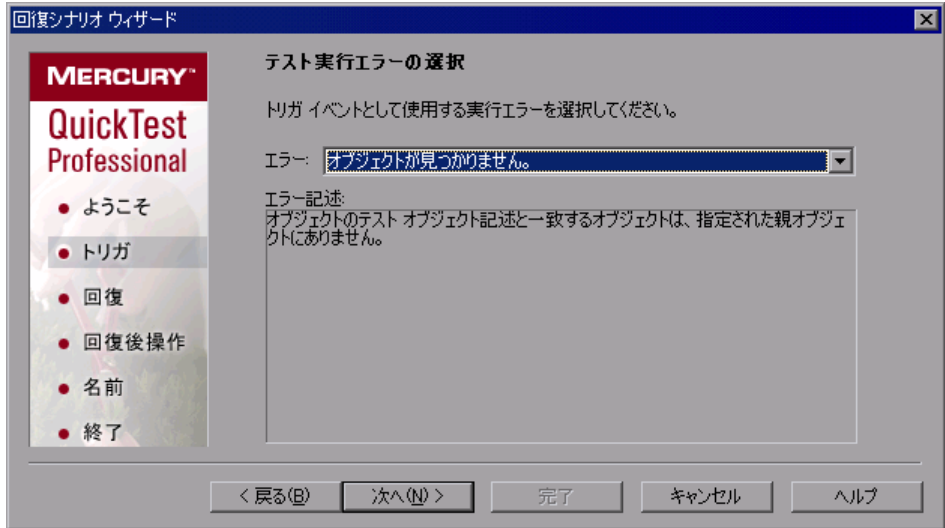
階層の各オブジェクトに対して、【プロパティ値の編集】ボックスで、オブジェクトの識別に使用するプロパティ値を変更できます。【追加 / 削除】ボタンをクリックし、検査対象のプロパティ値のリストからオブジェクト・プロパティを追加または削除することもできます。オブジェクトはクラスではなく、プロパティ値によってのみ識別されます。

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、『QuickTest Professional 基本機能ユーザーズ・ガイド』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。

【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、【テスト実行エラーの選択】画面が開きます。



【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。

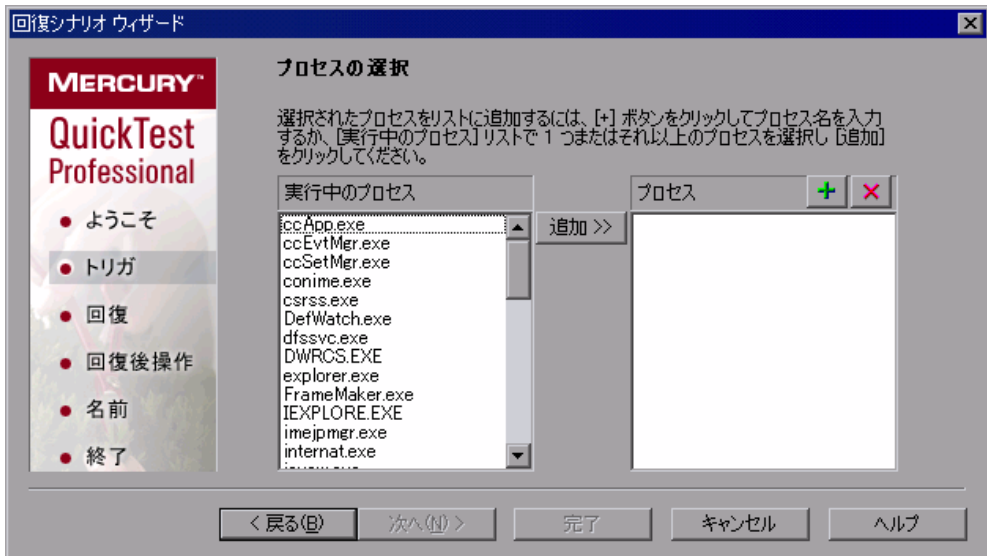
- ▶ **【任意のエラー】**：テスト・オブジェクト・メソッドによって返されるすべてのエラー・コード。
- ▶ **【リストまたはメニュー内の項目が一意ではありません】**：リスト，メニュー，またはツリーにある複数の項目に，メソッド引数で指定されている名前がある場合に発生します。
- ▶ **【リストまたはメニュー内の項目が見つかりません】**：メソッド引数で指定されているリスト，メニュー，またはツリー項目が QuickTest によって識別できない場合に発生します。この原因としては，その項目を現在利用できない，または名前が変更されていることが考えられます。
- ▶ **【物理記述に対して複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、[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] トリガを選択した場合、[プロセスの選択] 画面が開きます。



[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。



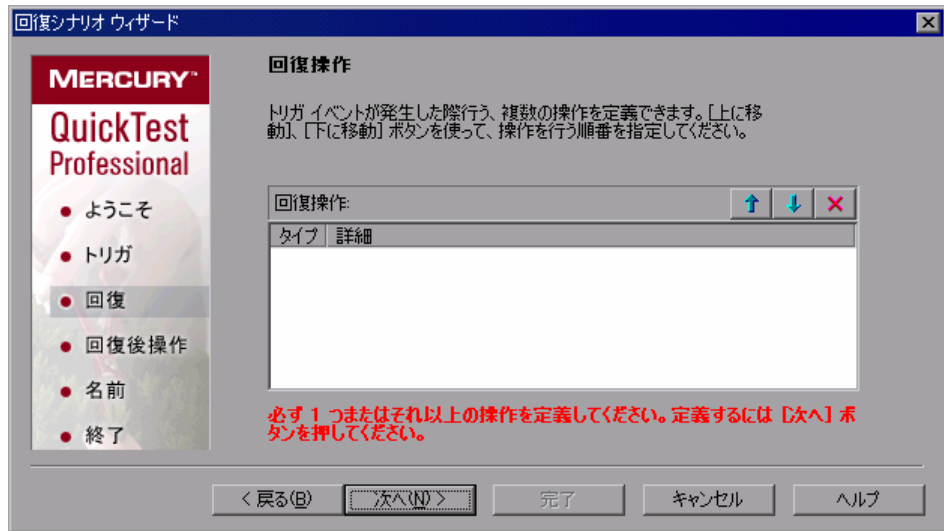
[プロセス] リストからプロセスを削除するには, プロセスを選択して [プロセスの削除] ボタンをクリックします。

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。

【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。



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。

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。

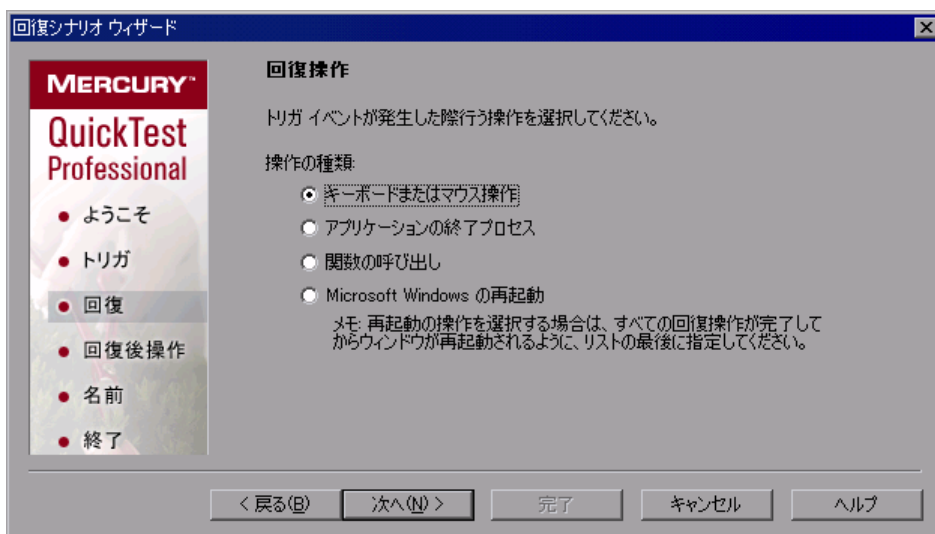
注：【Microsoft Windows の再起動】回復操作を定義する場合、常に最後の回復操作としてこの操作が挿入されるため、リストで位置は変更できません。

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。

- ▶ 他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。
- ▶ [他の回復操作を追加する] チェック・ボックスをクリアし、[次へ] をクリックして、[回復後操作] 画面に進みます。

【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。

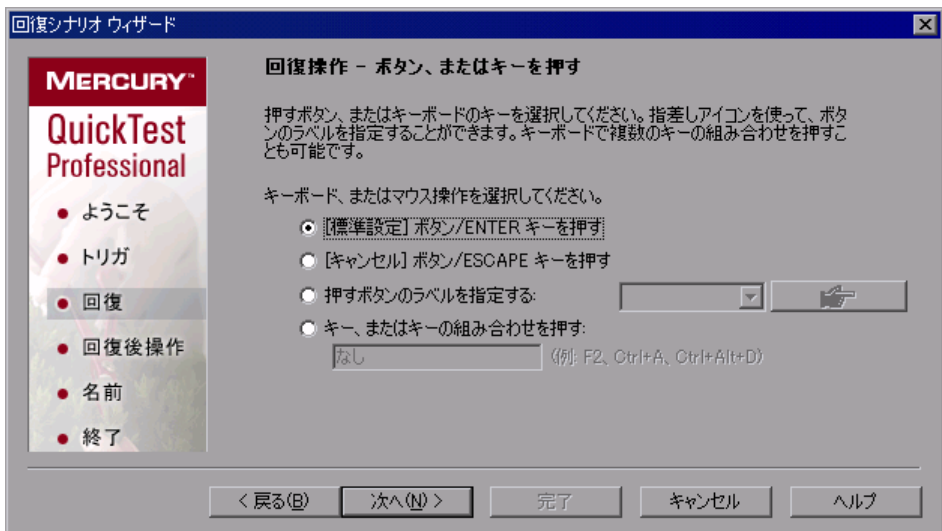
- ▶ [キーボードまたはマウス操作] : QuickTest によって、ウィンドウのボタンをクリックする操作、またはキーボードのキーを押す操作がシミュレートされます。このオプションを選択し、[次へ] をクリックして [回復操作—ボタン、またはキーを押す] 画面に進みます。
- ▶ [アプリケーションの終了プロセス] : QuickTest によって、指定のプロセスが閉じます。このオプションを選択し、[次へ] をクリックして [回復操作—プロセスの終了] 画面に進みます。

- ▶ **[関数の呼び出し]** : QuickTest によって、VBScript 関数が呼び出されます。このオプションを選択し、**[次へ]** をクリックして **[回復操作－関数の呼び出し]** 画面に進みます。
- ▶ **[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。

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コンポーネン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、コンポーネン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、**[回復操作－ボタン、またはキーを押す]** 画面が開きます。



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マウスオーバーなどの操作を実行したりするには、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、左の CTRL キーを押したまま、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。

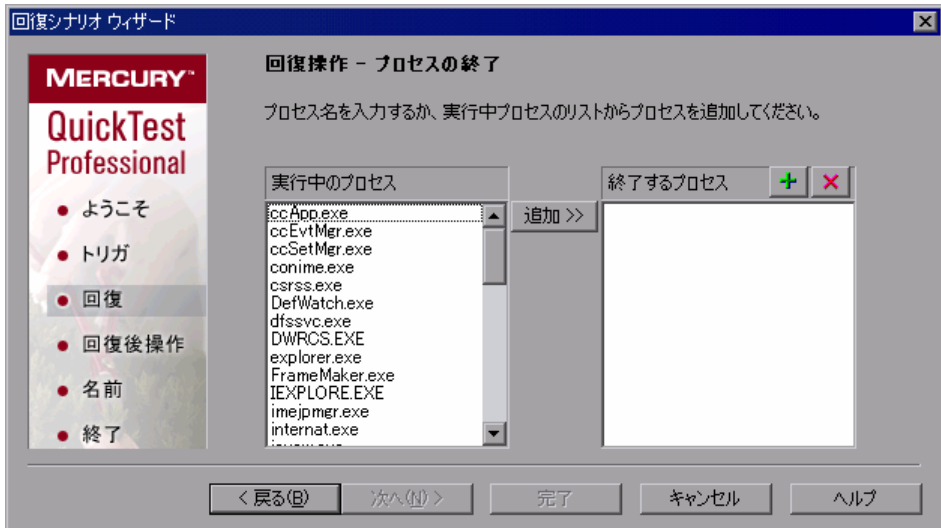
選択されたウィンドウのボタン・ラベルはすべて、リスト・ボックスに表示されます。リストから必要なボタンを選択します。

- ▶ **[[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—プロセスの終了】画面

【回復操作】画面の【アプリケーションの終了プロセス】回復操作を選択すると、【回復操作—プロセスの終了】画面が開きます。



【**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。



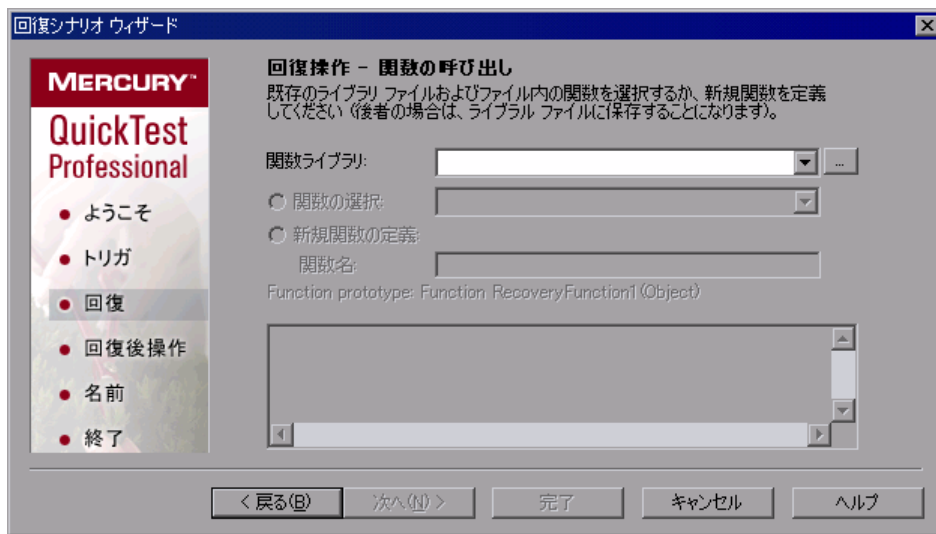
【**終了するプロセス**】リストからプロセスを削除するには、プロセスを選択して【**プロセスの削除**】ボタンをクリックします。

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「プロセスを閉じる」回復操作が表示されます。

[回復操作－関数の呼び出し] 画面

[回復操作] 画面の [関数の呼び出し] 回復操作を選択すると、[回復操作－関数の呼び出し] 画面が開きます。



[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：関数ライブラリは Quality Center プロジェクトに格納されている必要があります。

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。

注：[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。

)

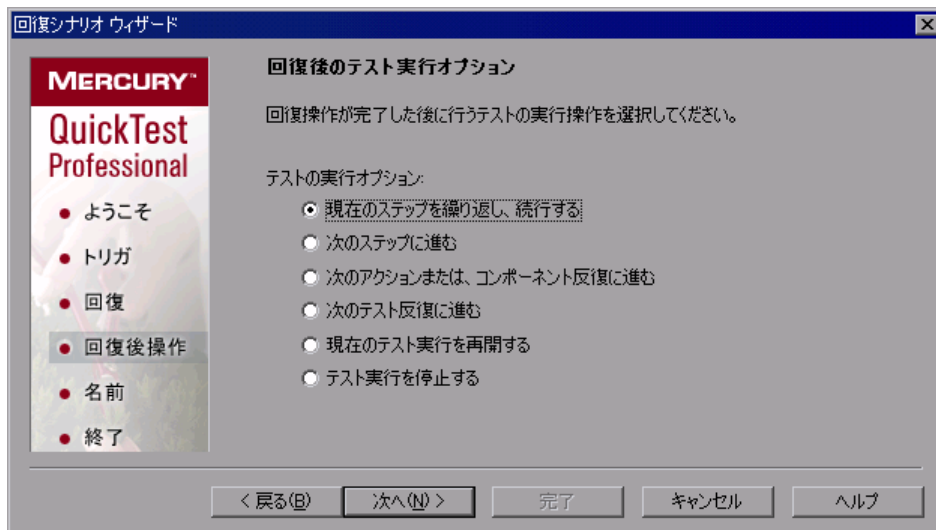
- ▶ **[新規関数の定義]**：一意の名前を指定し、表示される関数のプロトタイプに従って **[関数名]** ボックスで関数を定義することによって、新規の関数を作成します。新しい関数が、選択した関数ライブラリに追加されます。

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、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、**[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。



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、576 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可能性があることに注意してください。

▶ **[現在のテスト実行を再開する]**

ステップの実行を停止し、コンポーネントを始めから再実行します。

▶ **[テストの実行を停止する]**

コンポーネントの実行を停止します。

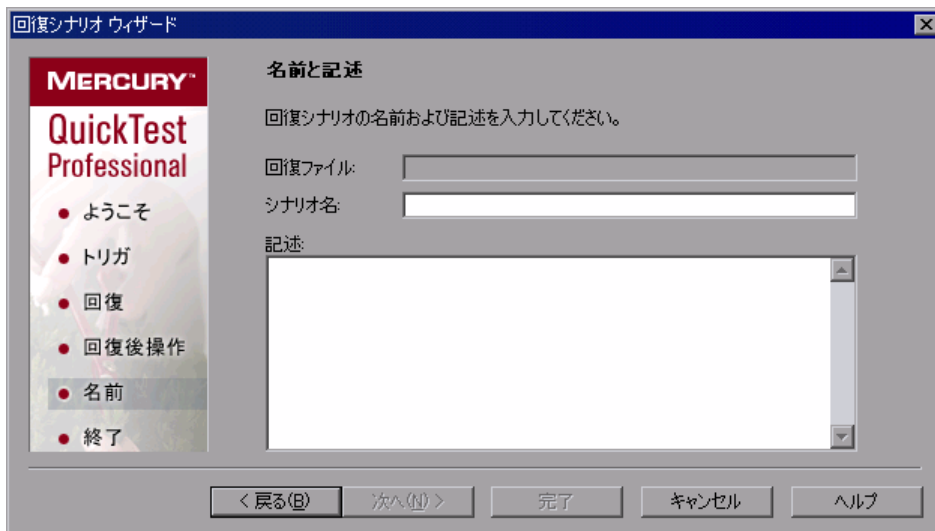
注：回復操作として [**Microsoft Windows の再起動**] を選択した場合、上記の最後2つのテスト実行のオプションのみから選択できます。

テスト実行の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。

【名前と記述】画面

[回復後のテスト実行オプション]画面でテスト実行のオプションを指定し、[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。

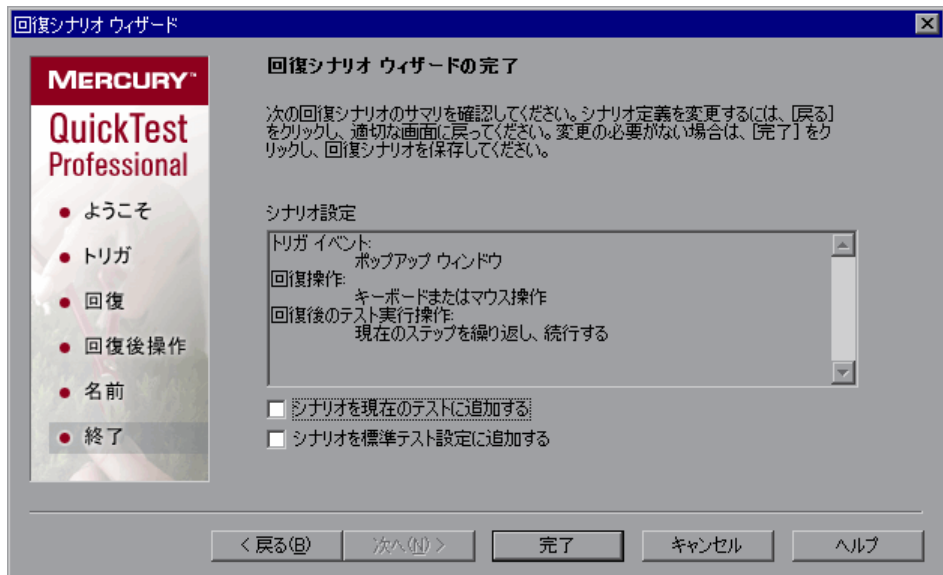


回復シナリオの名前とテキスト形式の説明を入力し、[次へ]をクリックして、[回復シナリオ ウィザードの完了]画面に進みます。

【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、[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。



注：コンポーネントの回復シナリオを、コンポーネントのアプリケーション領域に関連付けます。また、特定のアプリケーション領域に関連付けられているすべての新しいコンポーネントに対して、標準設定の回復シナリオを定義することもできます。詳細については、75 ページ「アプリケーション領域の使用」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。

回復ファイルへ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。

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。



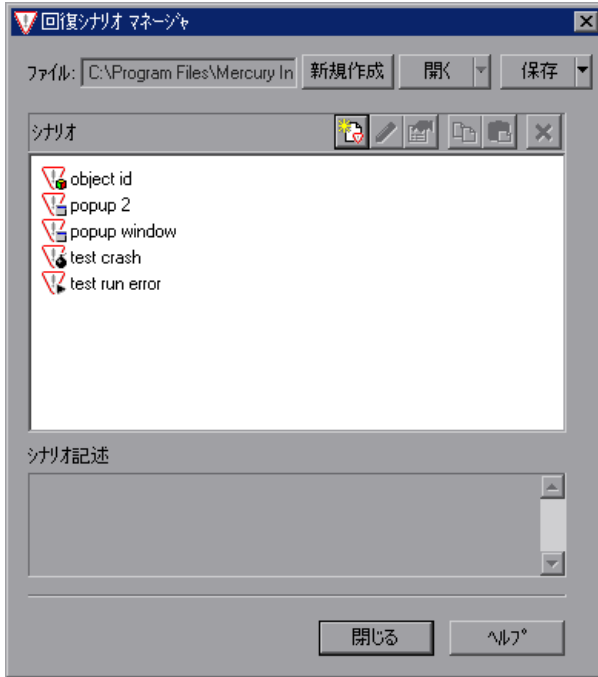
ヒント：**[保存]** ボタンの右側の矢印をクリックし、**[名前を付けて保存]** を選択して、別の名前で回復ファイルを保存することもできます。

- 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ます。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。

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、上記のステップ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。

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
	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。
	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。
	コンポーネン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。
	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。

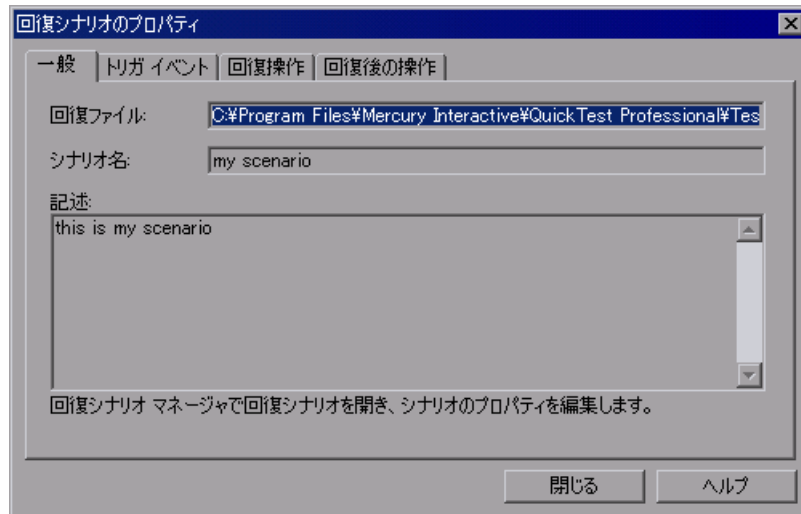
- ▶ 回復シナリオのプロパティの表示
- ▶ 回復シナリオの変更
- ▶ 回復シナリオの削除
- ▶ 回復シナリオ・ファイル間での回復シナリオのコピー

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] 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、543 ページ「回復シナリオの定義」を参照してください。

注：行った変更は、[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コンポーネン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。
- 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。

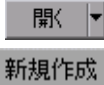
回復シナリオ・ファイル間での回復シナリオのコピー

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となる回復シナリオ・ファイルを選択するか、[新規作成] ボタンをクリックして、シナリオのコピー先となる回復シナリオ・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：

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。

アプリケーション領域用の回復シナリオ・リストの設定

回復シナリオを作成した後は、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、作成した回復シナリオと選択したテストまたはコンポーネントを関連付けます。シナリオに優先順位を付け、実行セッション中のシナリオの適用順序を設定できます。アプリケーション領域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。


注：アプリケーション領域内のコンポーネントの回復シナリオを定義します。詳細については、75 ページ「アプリケーション領域の使用」を参照してください。

回復シナリオのプロパティの表示

アプリケーション領域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、572 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 アプリケーション領域の一般表示枠の中で、[追加設定] ボタンをクリックします。[アプリケーション領域の設定] ダイアログ・ボックスが開きます。
- 2 [回復] タブをクリックします。
- 3 [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。
- 4  [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、571 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、[アプリケーション領域の設定] 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 アプリケーション領域の一般表示枠の中で、[追加設定] ボタンをクリックします。[アプリケーション領域の設定] ダイアログ・ボックスが開きます。
- 2 [回復] タブをクリックします。
- 3 [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。
- 4  [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。

- 5 優先順位を変更するシナリオごとに3～4の手順を繰り返します。

アプリケーション領域からの回復シナリオの削除

[アプリケーション領域の設定] ダイアログ・ボックスを使用して、特定のシナリオとアプリケーション領域の間の関連付けを削除できます。アプリケーション領域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。

回復シナリオをアプリケーション領域から削除するには、次の手順を実行します。

- 1 アプリケーション領域の一般表示枠の中で、[追加設定] ボタンをクリックします。[アプリケーション領域の設定] ダイアログ・ボックスが開きます。
- 2 [回復] タブをクリックします。
- 3 [シナリオ] ボックスで、削除するシナリオを選択します。
- 4  [削除] ボタンをクリックします。選択したシナリオは、アプリケーション領域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

[アプリケーション領域の設定] 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、アプリケーション領域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。

- 1 アプリケーション領域の一般表示枠の中で、[追加設定] ボタンをクリックします。[アプリケーション領域の設定] ダイアログ・ボックスが開きます。
- 2 [回復] タブをクリックします。
- 3 [シナリオ] ボックスで、次のいずれかを実行します。
 - ▶ 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**：ステップを実行するたびに回復メカニズムが呼び出されます。
 - ▶ **エラー発生時**：エラーの戻り値を返すステップの後のみ、回復メカニズムが呼び出されます。

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しょう。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、現在のステップはこの 10 番目のステップであり、トリガ・イベントの原因となったステップではありません。

- ▶ **なし**：回復メカニズムが無効になります。

注：[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。

ヒント: 実行セッション中、アプリケーション領域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、578 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。

すべての新しいコンポーネント用の標準回復シナリオの設定

すべての新規コンポーネントに対する標準の回復シナリオの定義は、コンポーネントのアプリケーション領域で行います。詳細については、75 ページ「アプリケーション領域の使用」を参照してください。

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分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。

Recovery オブジェクトとそのメソッドの詳細については、『**QuickTest Professional オブジェクト・モデル・リファレンス**』を参照してください。

第 21 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、その関数をコンポーネントで操作として使用できます。

注：本章では、「関数」、「メソッド」、および「操作」を同義語として使用しています。これは、ビジネス・コンポーネント・キーワード・ビューで「関数」と「メソッド」が「操作」として表示されるのに対し、QuickTest では「関数」と「メソッド」が使用されるためです。

本章では、次の内容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
- ▶ 関数ライブラリの管理
- ▶ 関連付けられている関数ライブラリを使用した作業
- ▶ 関数定義ジェネレータの使用方法
- ▶ 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
- ▶ ユーザ定義関数の使い方のヒント

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

ユーザ定義関数を作成することにより、コンポーネントに機能を追加できます。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、コンポーネントの設計、理解、保守が容易になります。QuickTest エンジニアまたは各分野のエキスパートが関連するキーワード（または操作）をコンポーネントに挿入することで、そのコンポーネント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、597 ページ「関数定義ジェネレータの使用法」および 610 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せん。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）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、638 ページ「VBScript 構文エラーの処理方法」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アプリケーション領域に関連付ければ、その関数ライブラリのパブリック関数をアプリケーション領域に関連付けられたすべてのコンポーネントで呼び出すことができます。詳細については、594 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（関数ライブラリの場合）、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、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他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、597 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、コンポーネントのアプリケーション領域に関数ライブラリを関連付けた後、その関数をコンポーネント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アプリケーション領域を通じてコンポーネントに関連付けることによって、各分野のエキスパートなど他のユーザでも、プログラミングの知識がなくても、複雑な操作（コンポーネント・ステップへの if/then ステートメントとループの追加など）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、『QuickTest Professional 上級機能ユーザース・ガイド』を参照してください。

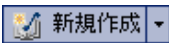
関数ライブラリの作成

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、247 ページ「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。

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。

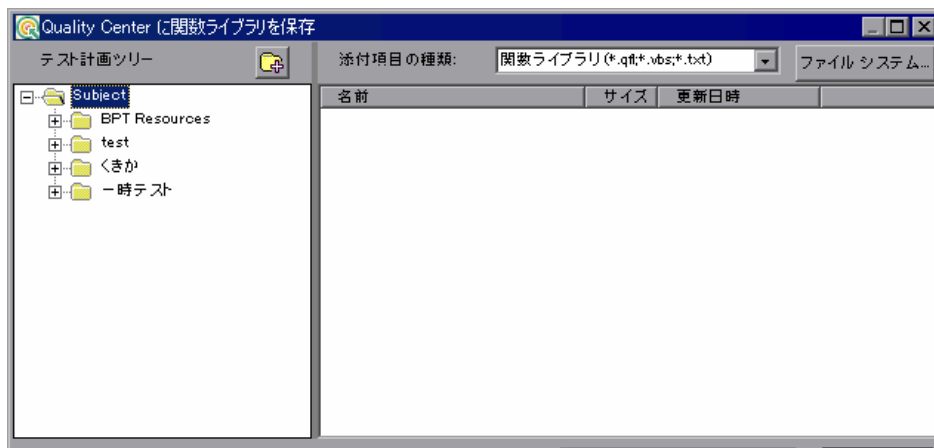
関数ライブラリを保存するには、次の手順を実行します。

- 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）。
- 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 Quality Center に保存] ダイアログ・ボックスが開きます。



- 3 [テスト計画ツリー] ボックスの中で、関数ライブラリの保存先フォルダを選択します。

注：関数ライブラリは（ファイル・システムではなく）Quality Center プロジェクトに保存する必要があります。

- 4 [添付名] ボックスに関数ライブラリ名を入力します。
- 5 [OK] をクリックします。関数ライブラリは、拡張子 **.qfl** を付けて保存され（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）、タイトル・バーに関数ライブラリ名が表示されます。

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。

注：関数ライブラリ内に定義されている関数をコンポーネントまたはアプリケーション領域の中で使えるようにするには、関数ライブラリを Quality Center プロジェクトに保存し、アプリケーション領域に関連付ける必要があります。詳細については、89 ページ「関数ライブラリの管理」を参照してください。

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、589 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。

開いているコンポーネントまたはアプリケーション領域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 から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：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、590 ページ「関数ライブラリの編集」および 591 ページ「関数ライブラリのデバッグ」を参照してください。

開いている QuickTest 文書間でのフォーカスの移動

コンポーネントまたはアプリケーション領域が開いている間、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、コンポーネントまたはアプリケーション領域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、同じ関数ライブラリ、またはコンポーネント（アプリケーション領域を通じて）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 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。

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、コンポーネント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、633 ページ「VBScript の基本的な構文の理解」を参照してください。

読み取り専用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です。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、586 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（コンポーネン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、関数ライブラリをコンポーネント（アプリケーション領域を通じて）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的に停止するように指定したりできます。詳細については、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。

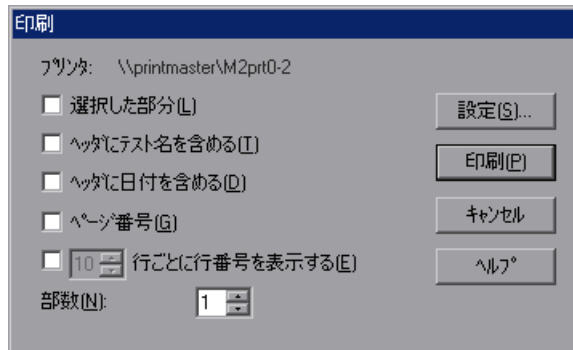
関数ライブラリの印刷

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。



- 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、そのファイルをアプリケーション領域に関連付けることができます。これによって、QuickTest エンジニアまたは各分野のエキスパートは、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、当該アプリケーション領域に関連付けられている任意のコンポーネン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コンポーネント（アプリケーション領域を通じて）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）。

コンポーネン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  アイコンが表示されます。そのコンポーネントまたはビジネス・プロセス・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。

既存のアプリケーション領域に関連付けられている関数ライブラリのリストはアプリケーション領域の関数ライブラリ表示枠（[**アプリケーション領域**] > [**関数ライブラリ**] のサイドバー・ボタン）で編集できます。詳細については、89 ページ「関数ライブラリの管理」を参照してください。

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

Quality Center および関連関数ライブラリを使用するときには、アプリケーション領域の関数ライブラリ表示枠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。

コンポーネントは、そのアプリケーション領域に関連付けられている関数にアクセスします。したがって、Quality Center プロジェクトに格納されていて、アプリケーション領域に関連付けられている関数ライブラリに加えた変更は、関連コンポーネントに影響を与える場合があります。Quality Center プロジェクトに格納されていて、アプリケーション領域に関連付けられている関数ライブラリに変更を加える場合は、当該アプリケーション領域を使用するコンポーネントに及ぼす影響を考慮してください。

アプリケーション領域との関数ライブラリの関連付け

現在開いているアプリケーション領域に、開いている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、現在開いているアプリケーション領域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、596 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。

アプリケーション領域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 関数ライブラリを関連付ける対象となるアプリケーション領域が 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、アプリケーション領域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、583 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 プロジェクトに保存します。詳細については、584 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' < Function Library >' を 'アプリケーション領域' 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' < Function Library >' を 'アプリケーション領域' に関連付ける]** を選択します。QuickTest によって、開いているアプリケーション領域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

アプリケーション領域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。

アプリケーション領域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 QuickTest で、アプリケーション領域を開いてサイドバーの **[関数ライブラリ]** ボタンをクリックします。



- 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。ファイル・パスに **[QualityCenter]** が追加され、参照ボタンが表示されて **Quality Center** プロジェクト内の関数ライブラリを参照できるようになります。

- 3 アプリケーション領域に関連付ける関数ライブラリを選択し、**[OK]** をクリックします。



ヒント：関連付けられている関数ライブラリをリストから削除するには、削除する関数ライブラリを選択して、**[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、89 ページ「関数ライブラリの管理」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アプリケーション領域に関連付けます。最後に、内容（コード）を追加して関数を完成させます。

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（関数ライブラリの場合）、キーワード・ビューで **[操作]** 項目が選択されている場合は **[操作]** リストに、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（関数ライブラリの場合）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、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、**[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。以降、関連付けられているコンポーネント（アプリケーション領域経由）は関数にアクセスできるようになります。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。

- 1 599 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。
- 2 600 ページ「関数の定義」の説明に従って、関数を定義します。
- 3 601 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（関数ライブラリの場合）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。

- 4 605 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、606 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、608 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。
- 7 608 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。
- 8 608 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、関数ライブラリ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。

関数定義ジェネレータ

関数定義

名前(N):

タイプ(T): Function

対象(O): Public

引数:

名前	成功モード
----	-------

テストオブジェクトに登録する(B)

テストオブジェクト(O): 操作(O):

標準設定操作として登録する(O)

追加情報

詳細(D):

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)

OK キャンセル ヘルプ

関数定義ジェネレータを開いた後、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。

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コンポーネン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（関数ライブラリがそのアプリケーション領域に関連付けられている場合のみ）。（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。

- 1 **[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（関数ライブラリの場合）またはキーワード・ビューから簡単に選択できるように、何をやる操作なのかがはっきりと分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ない、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**、**MessageBox**、**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] (アプリケーション領域がこの関数ライブラリに関連付けられている任意の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、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、605 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、キーワード・ビューの中で [項目] 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 (関数ライブラリの場合) 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。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 (関数ライブラリの場合)、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 (関数ライブラリの場合)、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。

QuickTest によって、[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関数が検索されます。指定した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、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つのアプリケーション領域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 エンジニアまたは各分野のエキスパートがキーワード・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、コンポーネントのアプリケーション領域に関連付けられている関数ライブラリに追加された後にはのみ、コンポーネン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左上角にある**[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(T): Function

対象(O): Public

テスト オブジェクトに登録する(B)

テスト オブジェクト(O): link 操作(O): VerifyProperty

標準設定操作として登録する(O)

名前	値
test_object	値

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、601 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。分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  と  を使用します。引数の順序は関数コードの分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。

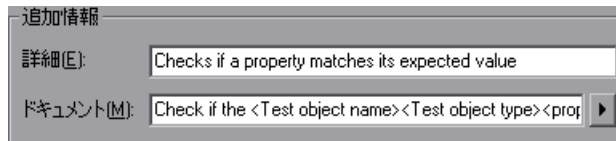
関数への説明の追加

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 エンジニアと各分野のエキスパートは、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、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、明確で分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「Mercury」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>.
```



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 "Mercury Business Technology" link "text" value matches the  
expected value: "Mercury 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または Quality Center に **[注釈]** カラムだけが（ビジネス・コンポーネントの場合にはコメントも）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。

- 1 **[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（関数ライブラリの場合）、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。

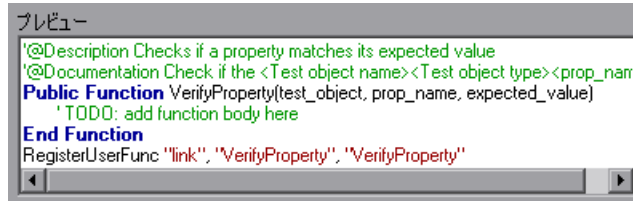
Checks whether a property value matches the actual value.

- 2 **[ドキュメント]** ボックスに、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。



```
プレビュー
'@Description Checks if a property matches its expected value
'@Documentation Check if the <Test object name><Test object type><prop_name>
Public Function VerifyProperty(test_object, prop_name, expected_value)
    ' TODO: add function body here
End Function
RegisterUserFunc "link", "VerifyProperty", "VerifyProperty"
```

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。
- 2 600 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、アプリケーション領域を通じて関数ライブラリに関連付けられているコンポーネントは、関数にアクセスできます。

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。
- 2 関数ライブラリの中で、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、必要に応じてコンポーネントまたはビジネス・プロセス・テストのステータスを変更したりできます。

- 3 関数ライブラリをアプリケーション領域に関連付けて、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、そのアプリケーション領域に関連付けられているコンポーネントがその関数にアクセスできることと、QuickTest エンジニアおよび各分野のエキスパート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、594 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

601 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（関数ライブラリの場合）、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

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 つの引数が必要になります。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ができます。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、コンポーネントで使用されるたびに、**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、コンポーネントの実行中、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。

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。

RegisterUserFunc TOClass, MethodName, FunctionName, SetAsDefault

項目	詳細
TOClass	注： 任意のテスト・オブジェクト・クラス。
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	コンポーネントから呼び出すユーザ定義関数の名前。この関数はコンポーネントのアプリケーション領域に関付けられている任意の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・ビューでテスト・オブジェクトを選択すると、 [操作] カラムに標準設定のメソッドが自動的に表示されます。

ヒント：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コンポーネン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、『QuickTest Professional オブジェクト・モデル・リファレンス』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、そのメソッドはコンポーネントによって使用されている間、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。他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。

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。
- ▶ 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コンポーネントでメソッドをすぐに利用できるようになります。

- ▶ コンポーネン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、ステートメントをコンポーネン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、コンポーネン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コンポーネント・ステップは  アイコンで示されます。その後、そのコンポーネントまたはビジネス・プロセス・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。
- ▶ コンポーネン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、コンポーネン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す。混乱を避けるために、1つのアプリケーション領域またはコンポーネン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。

- ▶ 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 を使用して関数とサブルーチンを作成する方法については、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])。

第 22 章

関数ライブラリ・ウィンドウを使用した作業

QuickTest 関数ライブラリ・ウィンドウを使用して、VBScript を利用した関数ライブラリを作成できます。本章では、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、次の内容について説明します。

- ▶ [関数ライブラリ] ウィンドウを使った作業について
- ▶ 関数ライブラリでのステートメントの生成
- ▶ 関数ライブラリ内での操作
- ▶ VBScript の基本的な構文の理解
- ▶ プログラム的記述の使用
- ▶ プログラムによるアプリケーションの実行と終了
- ▶ コメント、フロー制御、その他の VBScript ステートメントの使用
- ▶ テスト・オブジェクトのプロパティ値の取得と設定
- ▶ 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
- ▶ DOS コマンドの実行
- ▶ Windows API を使用したテストおよび関数ライブラリの拡張
- ▶ 実行セッション中に報告するステップの選択

[関数ライブラリ] ウィンドウを使った作業について

[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、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、『QuickTest Professional 基本機能ユーザーズ・ガイド』を参照してください。

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。関数ライブラリの印刷の詳細については、592 ページ「関数ライブラリの印刷」を参照してください。

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、『QuickTest Professional 基本機能ユーザーズ・ガイド』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メソッドまたはプロパティを選択し、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、次の「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。
- ▶ 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、623 ページ「VBScript 構文の自動補完」を参照してください。

オブジェクトを対象としたステートメントの生成

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メソッドまたはプロパティをドロップダウン・リストからの選択し、対応する構文を表示できます。

[**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 10 章「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。

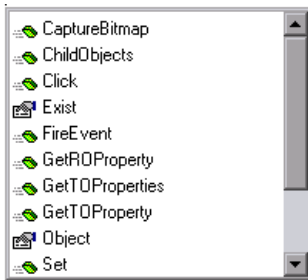
- ▶ ステートメントの中でテスト・オブジェクトの後にピリオドを入力すると、入力したオブジェクトの後に追加できるメソッド、プロパティおよび登録済みの関数のリストが表示されます。
- ▶ メソッドまたはプロパティの名前を入力すると、使用可能なメソッドおよびプロパティのリストが表示されます。CTRL + SPACE キーを押すと、選択肢が 1 つしかなければキーワードが自動的に補完されます。選択肢が複数ある場合は、入力したテキストに一致する (アルファベット順で) 最初のメソッドまたはプロパティが強調表示されます。
- ▶ メソッドまたはプロパティを入力すると、メソッドまたはプロパティの構文が、そ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メソッド、プロパティ、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。**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、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、657 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。

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

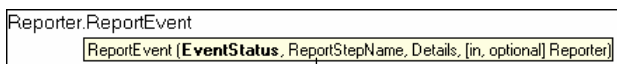
```
Browser("Welcome: Mercury Tours").Page("Book a Flight: Mercury").WebEdit("username").
```

- 3 ("username"). のように、オブジェクト記述の後ろにピリオド (.) を入力すると、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。



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を入力を始めてから 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。

- 4 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。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。



ステートメント補完ツールチップ

上の例では、**ReportEvent** メソッドには、4 つの引数があります。

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。

- 5 メソッドの後ろにメソッド引数を入力します。

QuickTest メソッドの詳細と用例については、『**QuickTest Professional オブジェクト・モデル・リファレンス**』を参照してください。

VBScript の構文の詳細については、633 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

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、248 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。



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、633 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、関数ライブラリ内の特定の行に移動できます。また、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。

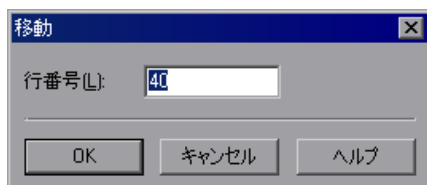
[移動] ダイアログ・ボックスの使用

[移動] ダイアログ・ボックスを使用して、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、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 10 章「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 必要であれば、関数ライブラリをアクティブにします。
- 2  **[移動]** ボタンをクリックするか、[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。



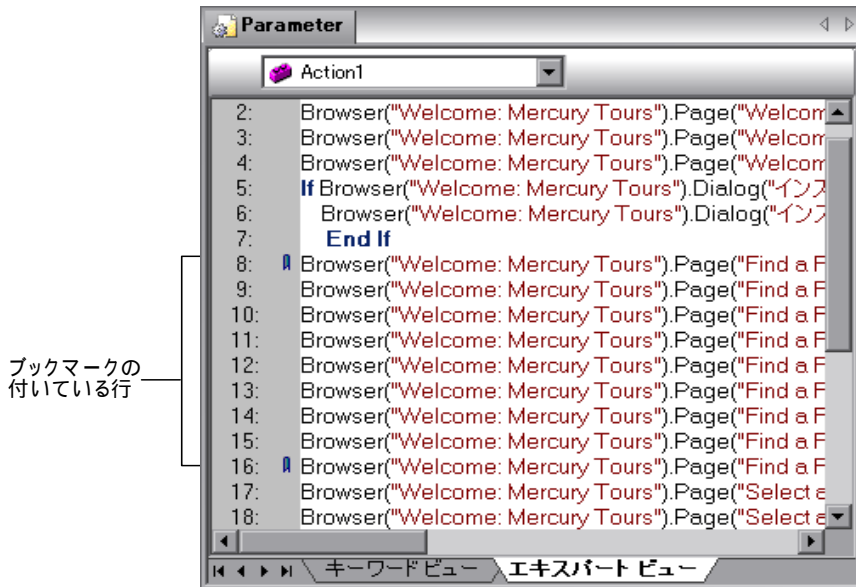
- 3 **[行番号]** ボックスに移動先の行番号を入力し、[OK] をクリックします。指定した行の先頭にカーソルが移動します。

ブックマークの使用

ブックマークを使って、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簡単に移動できます。ブックマークは、文書間を移動するときには維持されず、関数ライブラリとともに保存されません。

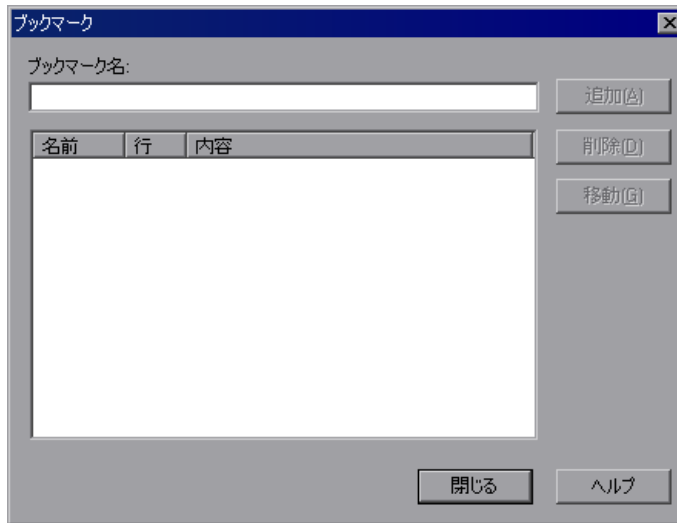
ブックマークを割り当てると、関数ライブラリ内で選択した行の左に、アイコンが付加されます。[ブックマーク] ダイアログ・ボックスの [移動] ボタンを使用して、ブックマークの付いている行に移動できます。


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 必要であれば、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。



- 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、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 必要であれば、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 が関数ライブラリの該当する行に移動します。

ヒント：標準設定では、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10章「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。

テキスト文字列の検索

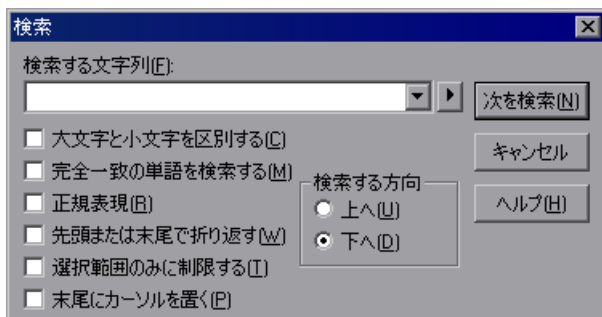
関数ライブラリ内で、検索する文字列を指定できます。リテラル・テキストを検索することも、正規表現を使用した高度な検索を行うこともできます。また、他のオプションを使用して検索結果の絞込みを行うことも可能です。


文字列を検索するには、次の手順を実行します。

- 1 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。
 - ▶ **[編集]** > **[検索]** を選択します。



[検索] ダイアログ・ボックスが開きます。



- 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、632 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。
- 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]：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 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 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

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。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、他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 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。
- ▶ [編集] > [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。

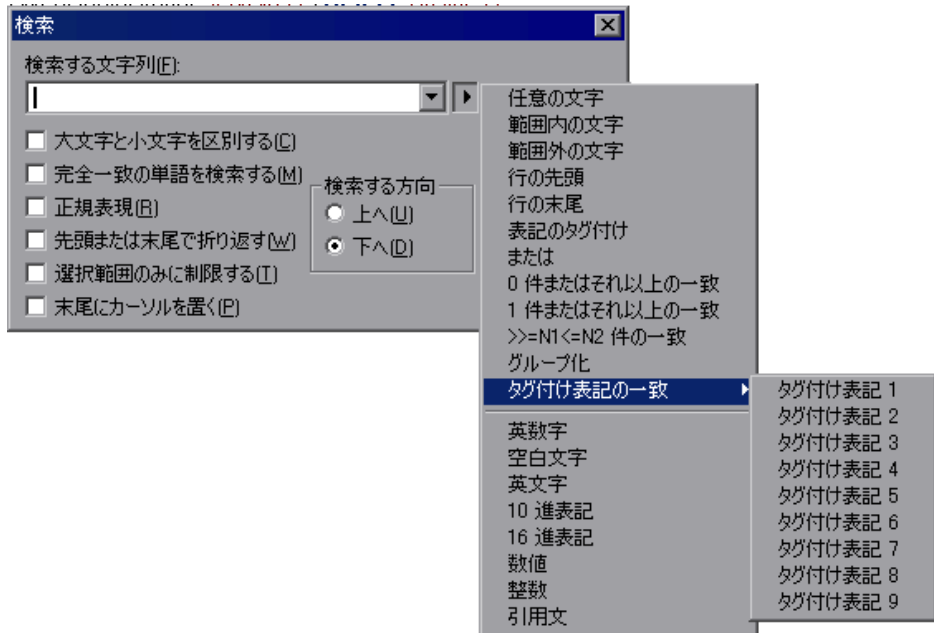
- 2 [検索する文字列] ボックスに、検索する文字列を入力します。
- 3 [置換後の文字列] ボックスに、見つかったテキストを置き換える文字列を入力します。

- 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、632 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。
- 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]：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。
- 6 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。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。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 の他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。

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

強力なスクリプト言語である VBScript を使用して、関数ライブラリを記述します。

この項では、VBScript ステートメントを使用して QuickTest 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 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]**）。



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。**[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、いつでも構文をチェックできます。

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、次の「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、636 ページ「括弧の使用」を参照してください。
- ▶ **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、637 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。スクリプトを分かりやすくして保守しやすいように、可能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、637 ページ「VBScript テキストの書式設定」および 650 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分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 を使用して関数ライブラリを拡張する方法の詳細については、650 ページ「コメント、フロー制御、その他の 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

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、他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。

Dim ステートメントを使用して、文字列、整数、配列など、他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、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 **actual_value** 変数を宣言し、現在の関数ライブラリの他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 actual_value  
    ' Get the actual property value  
    actual_value = obj.GetROProperty(PropertyName)
```

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。

ヒント：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。

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 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、**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

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、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、650 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、ステートメントを選択して、**[インデント]** ボタンをクリックします。あるいは、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。248 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、ステートメントを選択して、**[インデント解除]** ボタンをクリックします。あるいは、**[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 での書式設定の詳細については、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]**)。

VBScript 構文エラーの処理方法



現在の関数ライブラリの構文は、**[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、いつでもチェックできます。エラーが検出されると、情報表示枠に表示されます。

「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。

「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。

詳細	項目	アクション	行
Expected ']'	Library1.qfl	N/A	1
Expected ']'	Library1.qfl	N/A	4
Expected end of statement	Library1.qfl	N/A	6
Expected expression	Library1.qfl	N/A	7

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】	<p>構文エラーの詳細。たとえば、条件ブロックを If ステートメントで開始したけれども End If ステートメントで終了しなかった場合、[詳細] には Expected 'End If' と表示されます。</p> <p>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。</p> <p>Expected 'End Sub', or 'End Function', or 'End Property'</p> <p>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。</p>
【項目】	問題のステートメントが含まれている関数ライブラリの名前。
【アクション】	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マウスのポインタを合せたままにすると、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。

プログラムの記述の使用

オブジェクトに対する操作を記録すると、QuickTest により、適切なテスト・オブジェクトがオブジェクト・リポジトリに追加されます。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で記録されたものです。

```
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的な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、Web サイトまたは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の記述は、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、非常に便利があります。プログラムの記述は、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。

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、プログラムの記述を使うことで **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。

プログラムの記述には2つのタイプがあります。テスト・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列挙することも、プロパティと値のコレクションを **Description** オブジェクトに追加してから、ステートメントにその **Description** オブジェクトの名前を入力することもできます。

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、ステートメントにプログラムの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、**Description** オブジェクト方式のほうが強力で効率的です。

ステートメントへのプログラムの記述の直接入力

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")
```

TestObject : 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。

注：QuickTest はプログラムの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。

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の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の記述を使用する必要があります。プログラムの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の記述を使用しているため、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ため、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の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ため、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せんが、親オブジェクトがプログラムの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・オブジェクトを使った作業の詳細については、第 13 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 つの関数ライブラリで同じプログラムの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。

たとえば、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
```

プログラムの記述のための 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。

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、『**QuickTest Professional オブジェクト・モデル・リファレンス**』の「ユーティリティ」の節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。

たとえば、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
```

ヒント : ActiveX テスト・オブジェクトにプログラムの記述を作成し、対応する実行環境オブジェクトはウィンドウレス（ウィンドウ・ハンドルのないウィンドウ）になった場合は、記述に **windowless** プロパティを追加して、その値を **True** に設定する必要があります。

例を次に示します。

```
Set ButDesc = Description.Create  
ButDesc("ProgId").Value = "Forms.CommandButton.1"  
ButDesc("Caption").Value = "OK"  
ButDesc("Windowless").Value = True  
Window("Form1").AcxButton(ButDesc).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の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の記述を使用する必要があります。プログラムの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の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ため、このステートメントは使用できません。

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` の名前を使用しているため、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)。

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、『**QuickTest Professional オブジェクト・モデル・リファレンス**』を参照してください。

子オブジェクトの取得

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
```

ChildObjects メソッドの詳細については、『QuickTest Professional オブジェクト・モデル・リファレンス』を参照してください。

WebElement オブジェクトに対するプログラムの記述の使用

WebElement オブジェクトを使用すれば、ほかの Mercury テスト・オブジェクト・クラスには適合しない Web オブジェクトを対象にメソッドを実行できます。**WebElement** テスト・オブジェクトが記録されることは決してありませんが、プログラムの記述と **WebElement** オブジェクトを使って、Web サイト内の任意の Web オブジェクトを対象にメソッドを実行できます。

たとえば、次のステートメントを実行するとします。

```
Browser("Mercury Tours").Page("Mercury Tours").
    WebElement("Name:=UserName", "Index:=0").Click
```

または

```
set WebObjDesc = Description.Create()
WebObjDesc("Name").Value = "UserName"
WebObjDesc("Index").Value = "0"
Browser("Mercury Tours").Page("Mercury Tours").WebElement(WebObjDesc).
    Click
```

QuickTest は、Mercury Tours ページにある **UserName** という名前の最初の Web オブジェクトをクリックする操作を実行します。

WebElement オブジェクトの詳細については、『**QuickTest Professional オブジェクト・モデル・リファレンス**』を参照してください。

プログラムの記述での 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。

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。

プログラムによるアプリケーションの実行と終了

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、関数ライブラリ内で **SystemUtil.Run** ステートメントを使います。これはコンポーネント内からアプリケーションを開く操作（関数）を提供したい場合、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。

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
```

詳細については、『**QuickTest Professional** オブジェクト・モデル・リファレンス』を参照してください。

コメント、フロー制御、その他の VBScript ステートメントの使用

QuickTest では、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、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。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。

注：を参照してください。「**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、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、252 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。



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う方法もあります。詳細については、「Microsoft VBScript Language Reference」([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。

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、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 - 単項）
*	乗法
/	除法
^	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 つまたは複数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
```

```
Next
```

```
MsgBox "!" & 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。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。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 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。構文は次のとおりです。

```
While condition
    statement
```

```
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となります。構文は次のとおりです。

```
If condition Then
    statement
Elseif 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、**If**、**Elseif**、および **Else** ステートメントを使用して値が 1、2、またはその他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"
Elseif value = 2 Then
    msgbox "two"
Else
    msgbox "three"
End If
```

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。

テストまたはコンポーネントを実行すると、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。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("Mercury Technologies").Page("Mercury Technologies").  
    Link("Jobs").GetROProperty("href")
```

ヒント： Web サイトや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 12 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**、および **SetTOProperty** メソッドの詳細については、『**QuickTest Professional オブジェクト・モデル・リファレンス**』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。

QuickTest 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。ステートメント補完機能の詳細については、620 ページ「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、プログラムの記述の中で「**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、658 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、『QuickTest Professional オブジェクト・モデル・リファレンス』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、『QuickTest Professional オブジェクト・モデル・リファレンス』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/ <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、プログラムの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。

```
<IMG src="logo.gif" LogoID="122">
<IMG src="logo.gif" LogoID="123">
```

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の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
```

プログラムの記述の詳細については、640 ページ「プログラムの記述の使用」を参照してください。

DOS コマンドの実行

QuickTest 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、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://msdn.microsoft.com/library/en-us/winprog/winprog/windows_api_start_page.asp?frame=true

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://msdn.microsoft.com/library/en-us/winprog/winprog/windows_api_reference.asp?frame=true

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、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、『**QuickTest Professional オブジェクト・モデル・リファレンス**』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の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、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 および GW_HWNDDISABLED。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分かります。

```
/*
 * 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
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
' API 関数を使用する
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定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、『**QuickTest Professional オブジェクト・モデル・リファレンス**』を参照してください。

第 23 章

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コンポーネントを実行したりする操作を QuickTest のインタフェースを使用して手作業で行う方法の代わりに、これらを行うプログラムを作成することができます。

オートメーション・プログラムは、同じ作業を複数回実行したい場合や、複数のコンポーネントを対象に実行したい場合、あるいは、特定の環境またはアプリケーションのニーズに合わせて QuickTest をすばやく設定したい場合に特に有用です。

本章では、次の内容について説明します。

- ▶ QuickTest 操作のオートメーションについて
- ▶ QuickTest オートメーション・プログラムを使用する条件
- ▶ オートメーション・プログラム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・プログラムの基本要素の学習
- ▶ オートメーション・スクリプトの生成
- ▶ QuickTest オートメーション・オブジェクト・モデル・リファレンスの使用

QuickTest 操作のオートメーションについて

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 操作を自動化するプログラムを作成できます。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簡単に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、QuickTest オートメーション・オブジェクト・モデルにおいて、オブジェクト、メソッド、およびプロパティを通じて何らかの方法で表されます。必ずしも 1 対 1 の関係とはなりません。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・コマンドおよび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ることでプログラムを設計できます。

オートメーション・プログラムは、同じ作業を複数回実行したい場合や、複数のコンポーネントを対象に実行したい場合、あるいは、特定の環境またはアプリケーションのニーズに合わせて QuickTest をすばやく設定したい場合に特に有効です。

たとえば、Microsoft Visual Basic を使用して、コンポーネントに必要なアドインをロードし、QuickTest を可視モードで開始し、コンポーネントを開き、[オプション]、[ビジネス コンポーネントの設定] および [記録と実行環境設定] の各ダイアログ・ボックスに対応する設定を行い、コンポーネントを実行し、コンポーネントを保存するオートメーション・プログラムを作成して実行できます。

以後、プログラムに簡単なループを追加し、1つのプログラムで上記の操作を複数のコンポーネン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プログラム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・プログラムを使用して QuickTest を起動するように指示することで、テスト担当者の全員が必ず同じ設定で作業を行うことができます。

QuickTest オートメーション・プログラムを使用する条件

QuickTest を使用して設計するコンポーネントと同様に、有用な QuickTest オートメーション・プログラムを作成するには、計画、設計、テストの段階を踏む必要があります。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、定期的に行う必要があったりする QuickTest 操作は、QuickTest オートメーション・プログラムを使用する有力な候補となります。

次にいくつかの有用な QuickTest オートメーション・プログラムを示します。

- ▶ **初期化プログラム**：QuickTest を自動的に起動し、特定の環境での記録に必要なオプションおよび設定を指定するプログラムを作成できます。
- ▶ **コンポーネントの維持**：コンポーネントのコレクションを反復処理して特定の目的を達成するプログラムを作成できます。例を次に示します。
- ▶ **値の更新**：適切なアドインを使用して各コンポーネントを開き、更新されたアプリケーションを対象に更新モードで実行した後に保存することで、アプリケーションの更新された値に合わせてすべてのコンポーネントの値を更新します。

- ▶ **既存のコンポーネン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、既存のコンポーネン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。既存のコンポーネン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プログラムを作成できます。
- ▶ **他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・プログラム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・プログラム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・プログラム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・プログラム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。

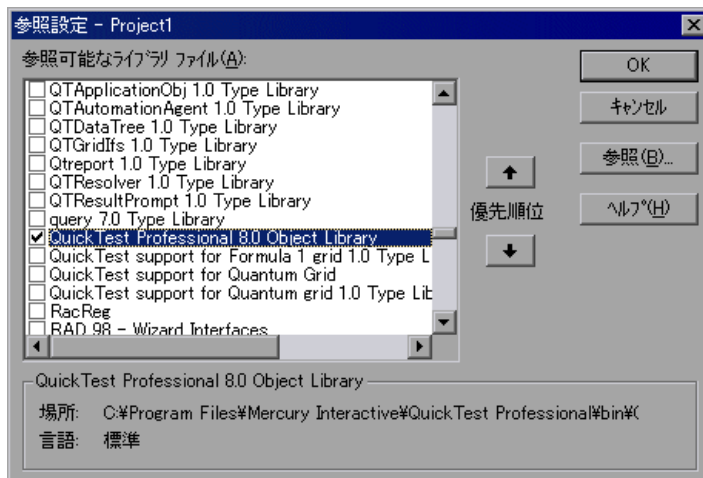
オートメーション・プログラムの作成

QuickTest オートメーション・プログラム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、VBScript、JavaScript、Visual Basic、Visual C++、Visual Studio.NET を使用できます。

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、プログラム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・プログラム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**」（ただし、<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・プログラムの実行

オートメーション・プログラムを実行できるアプリケーションがいくつかあります。Microsoft Windows Script Host を使用すれば、オートメーション・プログラムをコマンド・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・プログラムを実行できます。

```
WScript.exe /E:VBSCRIPT myScript.vbs
```


QuickTest オートメーション・プログラムの基本要素の学習

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、**Test** オブジェクト（コンポーネント・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）、**Addins** コレクション（[アドインマネージャ] 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 の他の要素を返したり、アドインをロードする、QuickTest を起動する、コンポーネントを開いて保存する、QuickTest を終了する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・プログラムは必ず **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（プログラム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 オートメーション・オブジェクト・モデル・リファレンス**』の「リモート・コンピュータ上でオートメーション・プログラムの実行」の節を参照してください。

プログラムの残りの部分の構造は、プログラムの目的に応じて異なります。QuickTest を開始する前に、コンポーネントに対応する関連アドインの取得、アドインのロード、可視モードでの QuickTest 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いければ、**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・プログラムの大半の操作は、**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、『**QuickTest オートメーション・オブジェクト・モデル・リファレンス**』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、672 ページ「**QuickTest オートメーション・オブジェクト・モデル・リファレンスの使用**」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必要があ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

[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
```

【スクリプトの生成】 ボタンの詳細、および、【オプション】 ダイアログ・ボックスと【オブジェクトの認識】 ダイアログ・ボックスのオプションの詳細については、第8章「グローバル・テスト・オプションの設定」と第14章「オブジェクトの認識の設定」を参照してください。

QuickTest オートメーション・オブジェクト・モデル・リファレンスの使用

『QuickTest オートメーション・オブジェクト・モデル・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・オブジェクト・モデル・リファレンス』は、次の場所から表示できます。

- ▶ QuickTest プログラム・フォルダ (【スタート】メニューの【QuickTest Professional】プログラム・グループから【Documentation】 > 【QuickTest Automation Reference】を選択します)
- ▶ QuickTest の【ヘルプ】メニュー (【ヘルプ】 > 【QuickTest オートメーション オブジェクト モデル リファレンス】)

第7部

付録

付録 A

QuickTest を使用した作業—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。

- ▶ コンポーネントの記録と実行
- ▶ 関数ライブラリを使用した作業
- ▶ 動的なコンテンツを使用した作業
- ▶ Web に関する高度な問題
- ▶ コンポーネントの保守
- ▶ QuickTest のパフォーマンスの向上

コンポーネントの記録と実行

- ▶ QuickTest は、どのようにして Web ページのユーザ・プロセスをキャプチャするのででしょうか。

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（記録されるユーザ・アクションを変更する方法の詳細については、第 11 章「Web イベント記録の設定」を参照してください）。記録したコンポーネントは、QuickTest を使用して元の順序のままステップを実行できます。

- ▶ 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。

さまざまな方法があります。

- ▶ QuickTest では標準設定で、いくつかの開発環境がサポートされます。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することによって、Java, Oracle, .NET, SAP Solutions, Siebel, Web サービスなどの環境をさらにサポートできます。
- ▶ 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、391 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。

関数ライブラリを使用した作業

- ▶ 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。

関数が含まれる 1 つまたは複数の VBScript ライブラリ・ファイルを作成できます。そして、任意のコンポーネントでそれらを使用できます。

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 21 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。

動的なコンテンツを使用した作業

- ▶ 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コンポーネントを記録し、実行するにはどうすればよいでしょうか。

Web ページまたは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。これらのオブジェクトを示す動的な記述を作成することで、QuickTest がテストを実行する際にそれらのオブジェクトを認識することができます。詳細については、第 13 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。

- ▶ 子ウィンドウの有無を検査するには、どうすればよいでしょうか。

ある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チェックすることができます。次に例を示します。

```
Browser("Window_name").Exist
```

また、**ChildObjects** メソッドを使用して、テスト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、特定の記述に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。

Exist プロパティと **ChildObjects** メソッドの詳細については、『**QuickTest Professional オブジェクト・モデル・リファレンス**』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いか。

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。

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。

- ▶ **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、クライアントは一般にそれに気付くことはなく、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 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。これ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 オブジェクトを操作できます。

▶ **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。

詳細については、『**QuickTest Professional 基本機能ユーザーズ・ガイド**』を参照してください。

コンポーネントの保守

▶ **アプリケーションに変更が加えられた場合、コンポーネン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コンポーネントの保守方法は、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 つの大きなコンポーネントを作成するのではなく、コンポーネントを小さなグループに分けて作成すべき主な理由の 1 つがこれです。アプリケーションに変更が加えられた場合に、コンポーネントの一部だけを再記録できます。大きな変更でない場合は、コンポーネントを手作業で編集して更新できます。

同じテスト・オブジェクトが含まれるコンポーネント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。

オブジェクトのプロパティが変更されたときに、テスト・オブジェクト・プロパティの情報を更新するには、[**更新モード**] オプションを使用します。詳細については、494 ページ「コンポーネントの更新」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、次のいずれも実行できます。

- ▶ QuickTest が起動する際に、アドイン・マネージャの中で、不必要なアドインが読み込まれないようにします。これにより、記録時間と実行セッションのパフォーマンスの両方を向上させることができます。アドインの読み込みの詳細については、683 ページ「QuickTest アドインのロード」を参照してください。
- ▶ コンポーネントを「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、199 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。

付録 B

QuickTest アドインの使用法

QuickTest Professional には、Web、ActiveX、Visual Basic を含む、複数の組み込みアドインがあります（これらのアドインは、QuickTest Professional セットアップからインストールします）。外部アドインは、個別に購入できます。

それらのアドインを使用することにより、専用のメソッド、プロパティおよびさまざまな専用のオプションを使って、アプリケーションに応じた最適のコンポーネントを作成できます。

本章では、次の項目について説明します。

- ▶ QuickTest アドインの使用法について
- ▶ QuickTest アドインのロード
- ▶ QuickTest アドインの使い方のヒント
- ▶ Web アドインを使った作業

QuickTest アドインの使用法について

QuickTest Professional のインストール時に QuickTest 組み込みアドイン（Web、ActiveX、Visual Basic）をインストールできます。QuickTest 組み込みアドインは、後からインストール・プログラムを再度実行してインストールすることもできます。QuickTest Professional の組み込みインストールとは別に使用できるアドインを「外部」アドインといいます。

ActiveX アドイン：組み込みの ActiveX アドインを使用して、ActiveX コントロールのテストが行えます。これらのコントロールを対象としたコンポーネントの作成と実行、およびプロパティの検査が可能です。ActiveX コントロールを対象としたコンポーネントは、標準的な Windows アプリケーションの場合とほとんど同じように作成および実行できます。

サポートされている ActiveX コントロールとバージョンの詳細については、「**QuickTest Professional 最初にお読みください**」を参照してください。

Visual Basic アドイン：組み込みの Visual Basic アドインを使用して、Visual Basic アプリケーションのテストが行えます。これらのコントロールを対象としたコンポーネントの作成と実行、およびプロパティの検査が可能です。Visual Basic アプリケーションを対象としたコンポーネントは、標準的な Windows アプリケーションの場合とほとんど同じように作成および実行できます。

Visual Basic .NET Windows Forms アプリケーションをテストするには、.NET アドインをインストールして読み込む必要があります。.NET アドインは外部アドインであり、コアの QuickTest の標準インストールに含まれていません。このアドインの購入方法については、QuickTest 販売代理店または Mercury カスタマー・サポートまでお問い合わせください。

Web アドイン：組み込みの Web アドインを使用して、Web ページおよび Web アプリケーションのテストが行えます。ハイパーリンク、画像、イメージマップ、Viewlink オブジェクトなどの Web オブジェクトをテストできます。Web アドインの使用法の詳細については、690 ページ「Web アドインを使った作業」を参照してください。

QuickTest を起動したら、[QuickTest Professional - アドイン マネージャ] ダイアログ・ボックスを使ってインストール済みのアドインからロードするものを選択します。

インストール済みのアドインをロードすると、そのアドインに対応する環境で記録を行う該当オブジェクトが QuickTest によって認識され、その環境に適したメソッド、プロパティ、および専用のオプションが使えるようになります。

各外部アドインには、シートまたはコンカレント・ライセンス・コードが必要です。シート・ライセンスのアドインをコンピュータにインストールするには、[アドイン マネージャ] ダイアログ・ボックスを使用します。コンカレント・ライセンスのアドインのインストールは、Mercury 機能テスト・コンカレント・ライセンス・サーバ・コンピュータで行います。

アドインとライセンスのインストールの詳細については、『**QuickTest Professional インストール・ガイド**』を参照してください。

QuickTest アドインのロード

複数の QuickTest アドインがインストールされている場合は、QuickTest セッションを開始するたびにロードするアドインを指定できます。

注：QuickTest Professional 9.0 では、バージョン 8.x および 9.0 のアドインをすべて使用できます。Java、.NET、PeopleSoft、Terminal Emulator、Siebel バージョン 8.x の各アドインにはアドイン・アップグレード・パッチが必要です。このパッチは QuickTest Professional のアップグレード時に自動的にインストールされます。これらのアドインを QuickTest 9.0 のインストールまたはアップグレード後にインストールした場合、適切なアドイン・アップグレード・パッチをインストールする必要があります。

Java 8.x アドインがインストールされているコンピュータ上に Siebel または Terminal Emulator の 8.x アドインをインストールするには、先に Java アドイン・アップグレード・パッチがインストールされている必要があります。

アドイン・アップグレード・パッチをインストールするには、**< QuickTest Professional のインストール先 > ¥AddinsUpgrade** フォルダから適切なパッチの実行可能ファイルを実行するか、[ヘルプ] > [更新の確認] オプションを使用して適切なアドイン・アップグレード・パッチをダウンロードおよびインストールします。[更新の確認] オプションの詳細については、『QuickTest Professional 基本機能ユーザーズ・ガイド』を参照してください。

PeopleSoft、Terminal Emulator、または Siebel 8.x アドインの適切なアドイン・アップグレード・パッチをインストールしなかった場合、当該アドインには [Incompatible] というラベルが付けられ、ロードできません。Java または NET 8.x アドインの適切なアドイン・アップグレード・パッチをインストールしなかった場合、アドインのロードと使用は可能ですが、一部の機能が利用できない場合があります。詳細については、「QuickTest Professional 最初にお読みください」を参照してください。

QuickTest を起動すると、[アドイン マネージャ] ダイアログ・ボックスが開きます。このダイアログ・ボックスには、QuickTest にインストールされているすべてのアドインの一覧と、各アドインに使用されているライセンスが表示されます。シート・ライセンスのアドインを使用している場合、ライセンスが期限切れとなるまでの残り日時も表示されます。[アドイン マネージャ] ダイアログ・ボックスに表示される情報の詳細については、685 ページ「アドイン・マネージャのオプションについて」を参照してください。



注：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、**[起動時にアドイン マネージャを表示する]** を選択します。

QuickTest の現在のセッションにロードするアドインを選択できます。パフォーマンスとオブジェクト認識の信頼性を最大化するため、必要なアドインのみをロードしてください。

外部アドイン用のアドイン・ライセンスがまだインストールされていない場合、外部アドインは、[アドイン マネージャ] ダイアログ・ボックスの [**ライセンス**] カラムに「**ライセンスされていません**」と表示されます。サブネット内のコンカレント・ライセンス・サーバに特定のアドインのライセンスが登録されていない場合、あるいはすべてのコンカレント・ライセンス・サーバで使用中である（そのため使用不能である）場合、アドインが「**ライセンスされていません**」と表示される可能性があります。このような場合、LSFORCEHOST 変数を使用して、該当するアドイン・ライセンス（使用可能な場合）がインストールされているサブネット外部のコンカレント・サーバへ接続することができます。コンカレント・サーバへの接続の詳細については、『**QuickTest Professional インストール・ガイド**』を参照してください。

アドイン・マネージャのオプションについて

アドイン・マネージャには次のオプションが含まれています。

- ▶ **[アドイン]** カラム：インストールされている QuickTest Professional アドインの名前がリストされます。
- ▶ **[ライセンス]** カラム：アドインによって使用される、次のライセンスがリストされます。
 - ▶ **[組み込み]**：QuickTest Professional と一緒に提供されるアドインです。組み込みアドインには、QuickTest Professional と同じライセンスが使用されます。そのため、QuickTest で「**永久**」ライセンスを使用している場合は組み込みアドインにも「**永久**」ライセンスが適用され、QuickTest で「**期日**」ライセンスを使用している場合は組み込みアドインにも「**期日**」ライセンスが適用されます。
 - ▶ **[期日]**：14 日間デモ・ライセンスや1年間のライセンスなどの、一時的なライセンスです（QuickTest シート・ライセンスを使用する場合にのみ表示されます。コンカレント・ライセンスでは表示されません）。
 - ▶ **[永久]**：使用期間に制限がありません。

- ▶ **[ライセンスされていません]** : シート・ライセンスがインストールされていないか、コンカレント・ライセンス・サーバにアクセスできない（たとえば、すべてのコンカレント・ライセンスが使用中のときや、サブネットのコンカレント・ライセンス・サーバに必要なアドイン・ライセンスがインストールされていない場合）アドインです。アドインをロードするには、最初にライセンスにインストールするか、ライセンスにアクセスする必要があります。詳細については『QuickTest Professional インストール・ガイド』を参照してください。
- ▶ **[Incompatible Version]** : QuickTest によってサポートされなくなったアドインです。サポートされている（新しい）バージョンを入手できる場合は、アドインをそのバージョンにアップグレードするかアップグレード・パッチをインストールすることでロード可能になります。詳細については、『QuickTest Professional インストール・ガイド』を参照してください。
- ▶ **[残り時間]** : 「期日」アドイン・ライセンスが期限切れとなるまでの、残りの日数と時間を示します（QuickTest シート・ライセンスを使用する場合にのみ表示されます。コンカレント・ライセンスでは表示されません）。
- ▶ **[アドインの記述]** : アドインがサポートする環境の説明です。
- ▶ **[起動時に表示する]** : QuickTest を起動するたびに、[アドイン マネージャ] ダイアログ・ボックスを表示するように QuickTest を設定します。

このチェック・ボックスをクリアすると、QuickTest の起動時に [アドイン マネージャ] は表示されません。[アドイン マネージャ] が表示されるように設定するには、[ツール] > [オプション] > [一般] を選択し、**[起動時にアドイン マネージャを表示する]** を選択します。[オプション] ダイアログ・ボックスの使用方法については、第8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 ライセンスの変更]** : [QuickTest Professional ライセンスのインストールようこそ] ウィンドウを開きます。このウィンドウでは、外部アドイン用のアドイン・ライセンスをインストールできます（QuickTest シート・ライセンスを使用する場合にのみ表示されます。コンカレント・ライセンスでは表示されません）。

アドイン・ライセンスのインストール手順は、QuickTest Professional ライセンスをインストールするときの手順と同じです。詳細については、『QuickTest Professional インストール・ガイド』を参照してください。

ロードするアドインの選択

QuickTest にロードするアドインを選択するには、該当するアドインの隣にあるチェック・ボックスを選択します。[OK] をクリックすると、選択したアドインが QuickTest によってロードされます。また QuickTest では選択したアドインが記憶されるため、次回 QuickTest を開くときに [アドイン マネージャ] ダイアログ・ボックスの中で同じアドインが選択されています。[起動時に表示する] チェック・ボックスをクリアすると、QuickTest を開いたときに、選択したアドインが自動的にロードされます。

ロードされているアドインと関連アドインの一致

コンポーネントを開く際には、現在ロードされているアドインと、そのコンポーネントに関連付けられているアドインとが比較されます。一致しない場合は、警告メッセージが表示されます。

コンポーネントに関連付けられているアドインで現在ロードされていないものがある場合は、必要に応じて次の作業を実行します。

- ▶ QuickTest を一度終了して再度起動し、[アドイン マネージャ] ダイアログ・ボックスで必要なアドインを選択する。
- ▶ そのコンポーネントに関連付けられているアドインのリストからアドインを削除する。コンポーネントに関連付けられているアドインのリストを変更するには、[ファイル] > [設定] を選択し、[プロパティ] タブの [変更] オプションを使用します。アドインの関連付けの詳細については、748 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。

アドインがロードされているのにコンポーネントと関連付けられていない場合は、必要に応じて次の作業を実行します。

- ▶ QuickTest を一度終了して再度起動し、[アドイン マネージャ] ダイアログ・ボックスで不要なアドインのチェック・ボックスをクリアする。
- ▶ そのコンポーネントに関連付けられているアドインのリストにアドインを追加する。コンポーネントに関連付けられているアドインのリストを変更するには、[ファイル] > [設定] を選択し、[プロパティ] タブの [変更] オプションを使用します。アドインの関連付けの詳細については、748 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。

QuickTest アドインの使い方のヒント

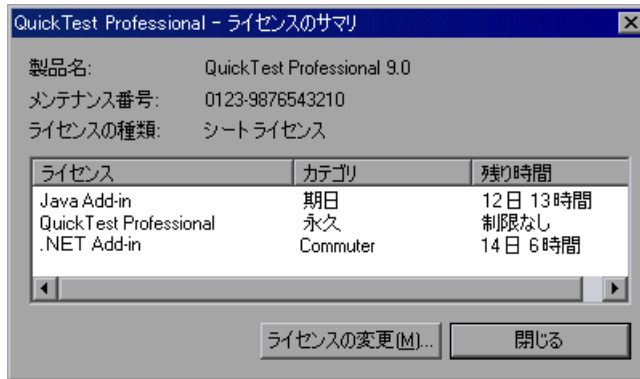
QuickTest アドインは、さまざまな開発環境のアプリケーションを対象としたコンポーネントの作成と実行を支援します。アドインをロードすると、対象の開発環境のアプリケーションに対して、他のアプリケーションと同様にコンポーネントを記録および実行できるようになります。

QuickTest アドインを使用したコンポーネントを設計する際には、QuickTest アドインの機能を十分に活用するために、次の点に注意してください。

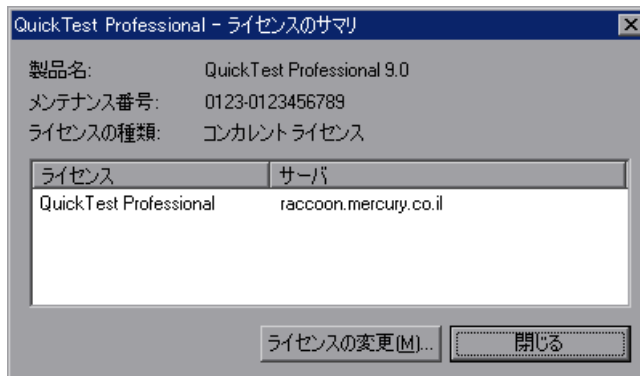
- ▶ QuickTest で、対応する環境のオブジェクトを認識できるようにするには、アドインをインストールしロードする必要があります。アドインをロードするには、QuickTest の起動時に開く [アドイン マネージャ] ダイアログ・ボックスでアドインを選択します。
- ▶ QuickTest の起動時に [アドイン マネージャ] ダイアログ・ボックスが開かない場合は、[オプション] ボタンをクリックするか、[ツール] > [オプション] を選択し、[一般] タブをクリックします。[起動時にアドイン マネージャを表示する] チェック・ボックスを選択し、[OK] をクリックします。QuickTest を再起動します。
- ▶ パフォーマンスとオブジェクト認識の信頼性を最大限に高めるには、必要なアドインのみをロードしてください。たとえば、Web アプリケーションや .NET アプリケーションに関連するテスト・プロセスが必要な場合は、Web アドインと .NET アドインだけをロードします。すべてのアドインを使う必要がない限り、すべてのアドインをロードしないようにしてください。
- ▶ 現在インストールまたはロードされているアドインの一覧を見るには、[ヘルプ] > [QuickTest Professional のバージョン情報] を選択します。ダイアログ・ボックスに、お使いのコンピュータにインストールされているアドインの一覧が表示されます。現在ロードされているアドインは、チェック・マークによって示されます。
- ▶ 現在ロードされているライセンス済みアドインすべてについてライセンスの詳細を表示するには、[QuickTest Professional のバージョン情報] ダイアログ・ボックス ([ヘルプ] > [QuickTest Professional のバージョン情報]) で [ライセンス] をクリックします。



- ▶ シート・ライセンスでは、各ライセンスの種類が表示されます。ライセンスの種類には、「体験版」、「永久」、「Commuter」および「期日」があります。「体験版」、「Commuter」（コンカレント・ライセンスとともに使用されず）および「期日」シート・ライセンスでは、ライセンスが期限切れとなるまでの残り日数および時間数也表示されます。



- ▶ コンカレント・ライセンスでは、各ライセンスに使用されるコンカレント・ライセンス・サーバの URL またはホスト名が表示されます。



シート・ライセンスとコンカレント・ライセンスを切り替えるには、[**ライセンスの変更**]をクリックします。QuickTest Professional およびロードされているすべてのアドインに対して（シートとコンカレントのどちらも）、セッションあたり1つのライセンスのみ使用できます。

ライセンスの種類、インストール、変更については、『**QuickTest Professional インストール・ガイド**』を参照してください。

Web アドインを使った作業

QuickTest の Web アドインを使用して Web ページおよび Web アプリケーションのテストが可能です。ハイパーリンク、画像、イメージマップ、Viewlink オブジェクトなどの Web オブジェクトをテストできます。

Web サイトや Web アプリケーションの記録を開始する前に、Web アドインがインストールされ、ロードされていることを確認します。Web アドインがインストールされていることを確認するには、[ヘルプ] > [QuickTest Professional のバージョン情報] を選択します。ロード済みのアドインは、アドインのリストの中で横にチェックマークが付いています。

[ビジネス コンポーネントの設定] ダイアログ・ボックスの [Web] タブ、および [オプション] ダイアログ・ボックスの [Web] タブの設定も行う必要があります。詳細については、第9章「ビジネス・コンポーネント設定にかかわる作業」および第8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。

QuickTest で要求に合うイベントが記録されない場合、記録すべきイベントを Web オブジェクトの種類ごとに設定することも可能です。たとえば、サブメニューを開く mouseover イベントのようなイベントを記録したい場合には、当該イベントを検出できるように Web イベント設定を変更する必要があるかもしれません。詳細については、第11章「Web イベント記録の設定」を参照してください。

注： Web ページまたは Web アプリケーション内のリストを記録するには、リストをクリックし、当初表示されていなかったエントリまでスクロールして、そのエントリを選択します。すでに表示されているリスト内の項目を選択するには、リスト内の別項目を選択（クリック）し、当初表示されていた項目に戻ってそれを選択（クリック）します。QuickTest は、リスト内の値が変わったときのみステップを記録するのでこの操作が必要です。

Web ブラウザを使用した作業

Web オブジェクトを検査するコンポーネントを記録するには、Web ブラウザを使用します。記録を開始する前に、該当する Web ページをブラウザで開く必要があります。

注：標準設定では、オブジェクト・リポジトリ内の **Browser** テスト・オブジェクトに割り当てられる名前は、当該ブラウザ・オブジェクトに関して記録された最初のページに割り当てられている名前です。以降の記録セッションにおいては、同じ順番 ID を持つブラウザで記録を行うたびに同じテスト・オブジェクトが使用されます。したがって、記録したステップ内で使用したブラウザの名前は、実際のブラウザ名を反映しない場合があります。

QuickTest では、Microsoft Internet Explorer を対象としたコンポーネントの記録、および次の Web ブラウザを対象としたコンポーネントの実行がサポートされています。

- ▶ Microsoft Internet Explorer
- ▶ Netscape ブラウザ
- ▶ Mozilla Firefox
- ▶ Microsoft Internet Explorer の Web ブラウザ・コントロールが埋め込まれているアプリケーション

注：QuickTest のコンポーネントは、基本的にはブラウザに依存しません。つまり、Microsoft Internet Explorer でコンポーネントを記録し、サポートされている別のブラウザでコンポーネントを実行できます。サポートされているブラウザのバージョンの詳細については、**Readme** ファイルを参照してください。

Microsoft Internet Explorer での作業

Web ブラウザとして Microsoft Internet Explorer を使用する場合には、次の点を考慮します。

- ▶ QuickTest Professional の Web サポート機能は、Microsoft Internet Explorer ではブラウザ拡張として動作します。したがって、Microsoft Internet Explorer で Web アドインを使用するには、**[サードパーティ製のブラウザ拡張を有効にする (再起動が必要)]** オプションを有効にする必要があります。このオプションを設定するには、Microsoft Internet Explorer の **[ツール] > [インターネット オプション] > [詳細設定]** を選択し、**[サードパーティ製のブラウザ拡張を有効にする (再起動が必要)]** をオンにします。
- ▶ QuickTest Professional ではタブ式ブラウジングはサポートされていません。したがって、Web アドインを使用する前に、Internet Explorer でタブ式ブラウジングを無効にする必要があります。このオプションを無効にするには、Microsoft Internet Explorer の **[ツール] > [インターネット オプション] > [詳細設定]** を選択し、**[タブ機能を有効]** をクリアします。このオプションをクリアしたら、QuickTest Professional Web アドインを使用する前にブラウザを再起動する必要があります。

Netscape ブラウザおよび Mozilla Firefox を使った作業

Web ブラウザとして Netscape ブラウザまたは Mozilla Firefox を使用する場合には、次の点を考慮します。

- ▶ QuickTest で Mozilla Firefox をサポートするようにするには、ブラウザのインストール・フォルダにファイルを作成する必要があります。そのため、QuickTest で初めて Mozilla Firefox を起動するときには、管理者権限で（またはブラウザのインストール・フォルダに対する書き込み許可を持ったユーザとして）QuickTest コンピュータにログインする必要があります。
- ▶ コンポーネントは、Microsoft Internet Explorer で記録し、Netscape ブラウザおよび Mozilla Firefox で実行できます。Netscape ブラウザおよび Mozilla Firefox ではコンポーネントの記録はできません。Netscape ブラウザまたは Mozilla Firefox で実行するコンポーネントを作成する方法は、2 とおりあります。
- ▶ Microsoft Internet Explorer でコンポーネントを記録する。

- ▶ キーワード駆動方式を用いる。[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), またはオブジェクト・リポジトリ・マネージャ (共有オブジェクト・リポジトリ) を使用してアプリケーションのオブジェクト・リポジトリを作成した後, キーワード・ビューまたはステップ・ジェネレータを使用してステップを追加します。キーワード駆動方式を用いる場合は, **Mozilla Firefox** または **Netscape** を使用してオブジェクトを追加できます。Microsoft Internet Explorer を使用する必要はありません。
- ▶ 通常, Microsoft Internet Explorer で記録されたコンポーネントは, 何の変更も加えることなく Netscape ブラウザおよび Mozilla Firefox で動作します。ただし, 注意すべきいくつかの違いがあります。

- ▶ QuickTest では, Netscape ブラウザおよび Mozilla Firefox のメニューまたはサイドバーをサポートしません。サポートされるツールバー・ボタンは, [ホーム], [更新], [戻る], [次へ], [停止] の各ボタンだけです。それ以外のツールバーおよびツールバー・ボタンは, いずれもサポートされません。

Netscape ブラウザでは, 次のツールバー・ボタンだけがサポートされます。



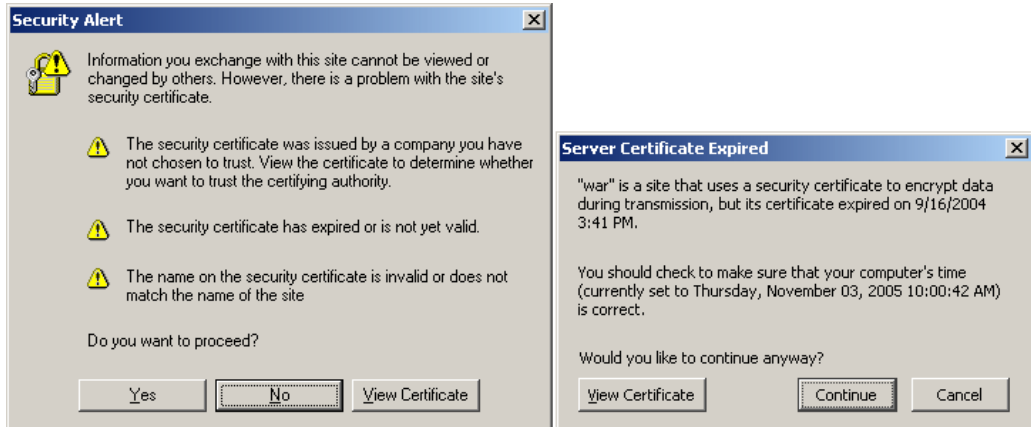
Mozilla Firefox では, 次のツールバー・ボタンだけがサポートされます。



サポートされていないメニューまたはツールバー・オブジェクトに対するステップを Microsoft Internet Explorer で記録した場合は, Netscape ブラウザおよび Mozilla Firefox でコンポーネントを実行する前に, 該当するステップを削除または置き換える必要があります。

- ▶ Netscape ブラウザおよび Mozilla Firefox が使用する標準のダイアログ・ボックスと, Microsoft Internet Explorer が使用する標準の Windows ダイアログ・ボックスは異なります。そのようなダイアログ・ボックスを対象とするステップがコンポーネントに含まれている場合は, Netscape ブラウザおよび Mozilla Firefox で実行する際に使用される適切なステップを作成する必要があります。

たとえば、次の2つは同じ Web サイトのセキュリティ警告ダイアログ・ボックスで、左側は Microsoft Internet Explorer、右側は Mozilla Firefox のものです。どちらも Windows のダイアログ・ボックスのように見えますが、Mozilla Firefox の方は実際にはブラウザ・ウィンドウです。



- ▶ 前述のように標準のダイアログ・ボックスが異なるため、**[押下するボタンのラベル指定する]** 回復操作で使用する Microsoft Internet Explorer 用に作成されたポップアップ回復シナリオは、Netscape ブラウザおよび Mozilla Firefox では動作しません。
- ▶ どの種類のブラウザについても、タブ式ブラウジングは QuickTest ではサポートされません。Netscape ブラウザおよび Mozilla Firefox では、新しいタブではなく新しいブラウザ・ウィンドウを開くようにブラウザを設定する必要があります。この設定方法はお使いのアプリケーションによって異なるため、設定手順については説明しません。ほとんどの場合は、すでに別のタブが開かれていても、オブジェクト・リポジトリにオブジェクトを追加してアクティブなブラウザ・タブ上でコンポーネントを実行できます。
- ▶ Netscape ブラウザでは Firefox および Internet Explorer の両方のエンジンをサポートしていますが、QuickTest では Firefox エンジンのみをサポートします。コンポーネントが失敗するか、QuickTest が Web ページのオブジェクトを認識しない場合、Netscape ブラウザが表示に Internet Explorer エンジンを使用していないことを確認してください。

レンダリング・エンジンを Firefox に変更するには、Netscape ブラウザの左下角にあるアイコンをクリックし、**[Display Like Firefox]** を選択します。



ヒント :

常に Firefox エンジンを使用してページを表示するようブラウザを設定することもできます (Netscape ブラウザで、**[ツール]** > **[オプション]** > **[Site Controls]** を選択し、**[Rendering Engine]** 領域の **[Firefox]** ラジオ・ボタンが選択されていることを確認します)。

常に Firefox エンジンを使用してページを表示するようブラウザを設定するには、Netscape ブラウザのインストール時に **[Automatically use the Internet Explorer Engine]** チェック・ボックスをクリアします。

- ▶ **Object** プロパティは、DOM オブジェクトにアクセスします。それらのオブジェクトは Netscape ブラウザおよび Mozilla Firefox でサポートしていません。**Object** プロパティの詳細については、『**QuickTest Professional 上級機能ユーザーズ・ガイド**』を参照してください。

Web ブラウザ・コントロールが埋め込まれているアプリケーションでの作業

Web ブラウザ・コントロールが埋め込まれているアプリケーションを対象にコンポーネントの記録と実行を行うには、次の手順を実行します。

- ▶ ActiveX アドインがロードされていることを確認します。
- ▶ アプリケーションを QuickTest の後に起動します。
- ▶ テストの記録または実行を開始します。

注 : 埋め込みブラウザ・コントロールは Microsoft Internet Explorer でのみサポートされています。

索引

A

Active Server Page テクノロジー 678

API

Windows の使用 659

asp ファイル 678

attribute プロパティ 658

B

Business Process Testing

QuickTest を使用したワークフロー 10

C

CGI スクリプト 678

Close メソッド 649

COM 678

CreationTime 識別子, 「序数識別子」を参照

CreationTime プロパティについて 378

D

Dim ステートメント, 関数ライブラリにおける 635

Do...Loop ステートメント, 関数ライブラリにおける 653

Domain コマンド・ライン・オプション 528

DOS コマンド, テスト内で実行 659

E

Exist プロパティ 677

eXtensible Markup Language (XML) 679

F

FAQ 675

For...Each ステートメント, 関数ライブラリにおける 653

For...Next ステートメント, 関数ライブラリにおける 652

FromDate コマンド・ライン・オプション 528

G

GetROProperty メソッド 656

H

HTML ファイルにエクスポート・ダイアログ・ボックス 516

I

If...Then...Else ステートメント, 関数ライブラリにおける 655

Index 識別子, 「序数識別子」を参照

Index プロパティについて 376

Index プロパティ, プログラム的記述と 648

IntelliSense 620, 250

L

Location 識別子, 「序数識別子」を参照

Location プロパティについて 377

Log コマンド・ライン・オプション 528

M

Mercury Tours, サンプル 14

Mercury Tours サンプル・アプリケーション xv

Mercury Tours, サンプル・アプリケーション xv

Mercury ベスト・プラクティス xvi

Mercury へのフィードバックの送信 xv

META タグ 678

Microsoft Internet Explorer での作業 692

Microsoft Windows の再起動操作 559

MinSize コマンド・ライン・オプション 529

N

Name コマンド・ライン・オプション 529

Netscape での作業 692

O

Object プロパティ, 実行環境メソッドの 658

Option Explicit ステートメント 616

P

Password コマンド・ライン・オプション 530

Project コマンド・ライン・オプション 530

Q

Quality Center

関連付けられている関数ライブラリ
594

コンポーネントのためのスナップ
ショットの取得 234

手動による不具合の報告 533

接続解除 23

不具合の報告 533

プロジェクトへの接続 20

Quality Center への接続 20

Quality Center への接続ダイアログ・ボックス
20, 23

QuickTest

ウィンドウ 25, 54

オートメーション・オブジェクト・モ
デル 665

開始 18

概要 17

製品情報 68

ソフトウェアの更新 16

QuickTest Professional のバージョン情報ウィン
ドウ 68

QuickTest オートメーション・オブジェクト・
モデル・リファレンス 672

QuickTest アドインのロード 683

QuickTest ウィンドウ

アクション・ツールバー 25

オートメーション・ツールバー 25, 54

概観 27

欠落リソース 39

情報表示枠 25, 39

ステータス・バー 26

挿入ツールバー 55

タブの移動 41

ツール・ツールバー 56

テーマ 27

デバッグ・ツールバー 25

表示ツールバー 56

標準ツールバー 53

標準のレイアウトの復元 49

表示枠の移動 41

表示枠の自動非表示 46

複数のドキュメント 50

編集ツールバー 55

メニュー・バー 25

レイアウトのカスタマイズ 41

QuickTest ソフトウェアの更新 16

QuickTest の開始 18

QuickTest のレイアウト

カスタマイズ 41

R

Recursive コマンド・ライン・オプション 530

RegisterUserFunc ステートメント 601, 610

S

Server コマンド・ライン・オプション 531

SetTOPProperty メソッド 363

Set ステートメント, 関数ライブラリにおける
635

SGML 679

Silent コマンド・ライン・オプション 531

SystemUtil.Run メソッド 649

T

Test コマンド・ライン・オプション 531

U

Unicode 3

UnregisterUserFunc ステートメント 610

UntilDate コマンド・ライン・オプション 532

User コマンド・ライン・オプション 532

V

VBScript

関連付けられている関数ライブラリ
Quality Center 594

構文 633

構文エラー 638

構文を自動的に拡張 251

テキストの書式設定 637
 マニュアル 650
 Viewlink オブジェクト 690

W

WebElement オブジェクト, プログラム的記述 647
 Web イベント記録の設定 257
 カスタマイズ 261
 標準 259
 Web イベント記録の定義ダイアログ・ボックス 260, 272
 Web コンテンツ, 動的な 677
 Web サイト, Mercury xv
 Web サイト, コンポーネントの記録と実行 690
 Web 設定
 オプション・ダイアログ・ボックス 212
 詳細 Web オプション・ダイアログ・ボックス 221
 ブラウザ詳細ダイアログ・ボックス 215
 ブラウザを無視 214
 ページ/フレーム・オプション・ダイアログ・ボックス 217
 Web タブ, ビジネス・コンポーネントの設定ダイアログ・ボックス 243
 Web ブラウザ, サポート 690
 While ステートメント, 関数ライブラリにおける 654
 Windows API 659
 Windows Application 詳細設定オプション・ダイアログ・ボックス 204
 Windows アプリケーション設定 201
 Windows コマンド・ライン・オプション 527
 Windows ダイアログ・ボックス 50

X

XML
 オブジェクト・リポジトリからエクスポート 425
 オブジェクト・リポジトリとしてインポート 424

あ

アクセス許可
 Quality Center の使用に必要な 13
 QuickTest の実行に必要な 13
 値
 オブジェクトのプロパティ値の指定 321
 オブジェクトのプロパティ値の表示 317
 オブジェクト・プロパティの標準設定の復元 324, 326
 出力 169
 出力のキャンセル 171
 出力のパラメータ化 177
 入力 166
 入力のパラメータ化 173
 値カラム 30
 値設定オプション・ダイアログ・ボックス 173
 値セル 166
 アドイン
 関連付けられているアドインとロードされているアドイン 687
 コンポーネントへの関連付け 86
 選択の変更 87
 使い方のヒント 688
 ロード 683
 アドイン・マネージャ・ダイアログ・ボックス 683
 アドイン・ライセンス 682, 686
 アプリケーション
 コンポーネントに関連付けられている 235
 実行 649
 終了 649
 アプリケーション, コンポーネントのための指定 106
 アプリケーション, サンプル xv, 14
 アプリケーションのクラッシュ・トリガ 548
 アプリケーションの指定ダイアログ・ボックス 109
 アプリケーション・プロセスを閉じる操作 559
 アプリケーション領域 75
 Web 設定 111
 アプリケーション領域の設定ダイアログ・ボックス 105
 一般表示枠 83

- オブジェクト・リポジトリの追加 97
- オブジェクト・リポジトリ表示枠 94
- 回復シナリオの削除 576
- 回復シナリオの保存 112
- 概要 76
- 関数ライブラリの指定 238
- 関数ライブラリの追加 91
- 関数ライブラリ表示枠 89
- 共有オブジェクト・リポジトリの
 - Quality Center プロジェクトへの追加 97
- 共有オブジェクト・リポジトリの選択 98
- コンポーネントに対する変更 138
- 削除 118
- 作成 79
- 設定の定義 for 83
- 説明 36
- 追加設定の定義 105
- 定義 11
- 開く 81
- 保存 116
- アプリケーション領域の保存ダイアログ・ボックス 116
- アプリケーション領域を開くボタン 81
- 暗号化, パスワード 168

い

- 一時停止, 実行セッション 475
- 一次リポジトリ 428
- 一次リポジトリ表示枠 432
- 一般オプション 248
- 一般表示枠
 - アプリケーション領域 83
 - 追加設定 83
- 移動ダイアログ・ボックス 625
- イベント記録の設定 257
 - 設定レベル 259
 - リセット 277
 - レベルのカスタマイズ 261

色

- キーワード・ビューでの変更 34
- 色の設定
 - オブジェクト・リポジトリ結合ツール 437

印刷

- 関数ライブラリ 592
- コンポーネント 140
- 実行セッション結果 513
- 印刷ダイアログ・ボックス 513
- 印刷プレビュー・ダイアログ・ボックス 514
- インポート
 - XML ファイルからのオブジェクト・リポジトリ 424

う

- 埋め込み Web ブラウザ・コントロール 695

え

- エクスポート
 - オブジェクト・リポジトリから XML ファイルへ 425
 - オブジェクト・リポジトリ・ファイルへのローカル・オブジェクトのエクスポート 364
- エディタ・オプション・ダイアログ・ボックス 248
- エラー, VBScript の構文 638

お

- オートメーション
 - Application オブジェクト 670
 - オートメーション 665
 - 開発環境 668
 - 言語 668
 - タイプ・ライブラリ 668
 - 定義 666
- オートメーション・エンジニア 4, 12
- オートメーション・ツールバー, QuickTest ウィンドウ 25, 54
- オブジェクト
 - オブジェクト・リポジトリからの削除 354
 - 識別 281
 - 認識 367
 - プロパティ, 実行環境 657
 - メソッド, 実行環境 657
 - メソッドの表示 281
- オブジェクト・スパイ 294
- オブジェクトの状態トリガ 548
- オブジェクトの選択画面 552

- オブジェクトの選択ダイアログ・ボックス 161
 - オブジェクトの認識
 - 自動スクリプトの生成 381
 - 標準設定の復元 380
 - オブジェクトの認識ダイアログ・ボックス 369
 - オブジェクトのプロパティ
 - 表示 317
 - オブジェクトのプロパティ値
 - 指定または変更 321
 - 表示 317
 - 標準設定の復元 324, 326
 - オブジェクトのプロパティと値の設定画面 554
 - オブジェクトの割り当てダイアログ・ボックス 391
 - オブジェクト名
 - 変更 327
 - オブジェクト・リポジトリ
 - XML からインポート 424
 - XML へのエクスポート 425
 - アプリケーション領域への追加 97
 - オブジェクトのコピー、貼り付け、および移動 351
 - オブジェクトの削除 354
 - オブジェクトの追加 343
 - オブジェクトの場所の特定 359
 - 管理 396
 - 共有 304
 - 作成 405
 - 閉じる 409
 - 開く 405
 - 変換 405
 - 変更 412
 - 保存 407
 - 未割り当て 186
 - ローカル 303
 - ローカル・オブジェクトのエクスポート 364
 - オブジェクト・リポジトリ・ウィンドウ 306
 - オブジェクトのフィルタ処理 313
 - 簡易表示と完全表示 313
 - テスト・オブジェクトの詳細 314
 - オブジェクト・リポジトリ結合ツール 427
 - 一次リポジトリ表示枠 432
 - 色の設定 437
 - ウィンドウ 429
 - 解決方法のオプション表示枠 433
 - 矛盾の解決方法の設定 438
 - ターゲット・リポジトリのフィルタ処理 455
 - ターゲット・リポジトリ表示枠 431
 - 二次リポジトリ表示枠 432
 - ビューの変更 430
 - 矛盾 450
 - 矛盾の解決 453
 - 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 346
 - オブジェクト・リポジトリの種類 301
 - オブジェクト・リポジトリ表示枠
 - アプリケーション領域 94
 - オブジェクト・リポジトリ・マネージャ 398
 - オブジェクト・リポジトリ・モード
 - 選択 303
 - オプション・ステップ 500
 - オプション・ダイアログ・ボックス 192
 - Web タブ 212
 - Windows アプリケーション・タブ 201
 - 一般タブ 194
 - 実行タブ 199
 - 自動スクリプトの生成 194
 - スクリプトの生成オプション 671
 - フォルダ・タブ 196
- か**
- 解決方法のオプション表示枠 433
 - ガイドライン
 - ユーザ定義関数 615
 - 回復
 - アプリケーション領域からのシナリオの削除 576
 - 回復シナリオ 569
 - シナリオ 539
 - シナリオのコピー 573
 - シナリオの削除 573
 - シナリオのプロパティの表示 571, 575
 - シナリオの変更 572
 - シナリオの無効化 576
 - シナリオの優先順位の設定 575
 - 操作 540
 - ファイル 543
 - アプリケーション領域
 - シナリオの保存 112
 - 回復後のテスト実行オプション画面 565

- 回復後のテスト実行のオプション 540
- 回復シナリオ・ウィザード 547
 - オブジェクトの選択画面 552
 - オブジェクトのプロパティと値の設定画面 554
- 回復後のテスト実行オプション画面 565
- 回復シナリオ・ウィザードの完了画面 568
- 回復操作画面 558, 559
- 関数の呼び出し画面 563
- テスト実行エラー画面 555
- トリガ・イベントの選択画面 548
- 名前と記述画面 567
- プロセスの終了画面 562
- プロセスの選択画面 556
- ボタン, またはキーを押す画面 560
- ポップアップ・ウィンドウの条件を指定画面 550
- 回復シナリオ・ウィザードの完了画面 568
- 回復シナリオ・マネージャ・ダイアログ・ボックス 543
- 回復操作
 - Microsoft Windows の再起動 559
 - アプリケーション・プロセスを閉じる 559
 - 関数呼び出し 559
 - キーボードまたはマウスの操作 559
- 回復操作画面 558, 559
- 回復操作—関数の呼び出し画面 563
- 回復操作—プロセスの終了画面 562
- 回復操作—ボタン, またはキーを押す画面 560
- 各分野のエキスパート 4, 12, 76
- カスタマー・サポート, Web サイト xv
- カスタマイズ, 関数ライブラリ 247
 - 一般オプション 248
- 画像, コンポーネントのための取得 234
- 画面ショット, 「スナップショット」を参照
- カラム
 - 表示オプション 32
- 簡易表示
 - オブジェクト・リポジトリ・ウィンドウ 313
- 環境, コンポーネントのために表示 235
- 関数
 - ユーザ定義 581
 - 関数コードのコピー 608
 - 関数コードの仕上げ 608
 - 関数定義ジェネレータ 600
 - 関数コードのプレビュー 608
 - 関数に説明を付ける 606
 - 関数の定義 600
 - 関数の登録 601
 - 使用法 597
 - 開く 599
 - 関数に説明を付ける 606
 - 関数呼び出し操作 559
 - 関数ライブラリ 581
 - アプリケーションの実行 649
 - アプリケーションの終了 649
 - アプリケーション領域での作成 92
 - アプリケーション領域への追加 91, 92
 - 一般オプション 248
 - エレメントの強調表示 252
 - 関数ライブラリにおけるプログラミング 619
 - 管理 583
 - 関連付けの変更 596
 - 関連付けられているものを使用した作業 594
 - 現在のものを関連付け 595
 - コンポーネントに関連付けられている 238
 - コンポーネントへの関連付け 238
 - 作成 584
 - 実行の一時停止 475
 - 説明 38
 - 定義 11
 - テキストの検索 628
 - テキストの置換 630
 - デバッグ 465, 591
 - ナビゲーション 588
 - 開く 586, 593
 - 編集 590
 - 保存 584
 - 見映えのカスタマイズ 247
 - 読み取り専用, 編集 591
 - 関数ライブラリのカスタマイズ
 - エレメントの強調表示 252
 - 関数ライブラリ表示枠
 - アプリケーション領域 89

完全表示

オブジェクト・リポジトリ・ウィンドウ 313

管理, コンポーネント

印刷 140

管理, テスト・オブジェクトの 299

関連付け

現在の関数ライブラリ 595

コンポーネントへのアドインの 86

関連付けられているアドインとロードされているアドイン 687

関連付けられている関数ライブラリ 594

変更 596

き

キーの割り当て

関数ライブラリでの 253

キーボード・ショートカット

関数ライブラリでの 253

キーボードまたはマウスの操作 559

キーワード

管理 101

キーワード・ビュー 28, 121

概要 152

カラム表示オプション 32

定義 11

フォントと色の変更 34

キーワード表示枠 (アプリケーション領域内)

カラムのフィルタリング 103

カラム内容の並べ替え 105

技術サポート, 「カスタマー・サポート」を参照

記述的プログラミング, 「プログラムの記述」を参照

記述, テスト・オブジェクト 285

「オブジェクト」を参照

記述プロパティ

テスト・オブジェクトの記述プロパティの追加 329

規則, 表記 xvii

規則, 「表記規則」参照

基本のイベント記録設定レベル 259

矛盾の解決方法の設定

オブジェクト・リポジトリ結合ツール 438

強調表示

アプリケーション内のテスト・オブジェクト 358

共有オブジェクト・リポジトリ 301, 304

Quality Center への追加 97

管理 94

結合 427

選択 98

未割り当て 186

共有オブジェクト・リポジトリ・ウィンドウ 403

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 459, 461

共有オブジェクト・リポジトリを開くダイアログ・ボックス 98

許可

Quality Center の使用に必要な 13

QuickTest の実行に必要な 13

記録

Web サイト 690

時間, 向上 680

ステータス, オプション 268

マウスの右ボタン・クリック 271

記録設定オプション 205

く

クッキー 678

グローバル・オプション

すべてのコンポーネントの設定 191

け

計算, 関数ライブラリにおける 651

結果

コンポーネント実行の結果の表示 501

テストの結果の表示 501

結果削除ユーティリティ

コマンド・ラインからの実行 527

結果スキーマ 534

結合

共有オブジェクト・リポジトリ 427

ローカル・オブジェクト・リポジトリ 443

欠落リソース表示枠 39

未割り当てのリポジトリ・パラメータ 187

言語サポート, Unicode 3

索引

検索

関数ライブラリでテキストを検索 628
テスト・オブジェクトとプロパティ
355

検索ダイアログ・ボックス
オブジェクト・リポジトリ結合ツール
457

検索 / 置換ダイアログ・ボックス 355

こ

高位のイベント記録設定レベル 259

更新, コンポーネントの 494

更新, コンポーネント 494

更新, マニュアル xvi

更新, ローカルから

リポジトリの結合 443

高度な機能ユーザズ・ガイド, QuickTest
Professional xiii

構文エラー, VBScript 638

項目カラム 30

項目セル 156

項目, 選択

アプリケーションからの 161

共有オブジェクト・リポジトリからの
158

項目リストからの 157

項目リスト 157

コマンド・ライン・オプション, 使用したテ
スト結果の削除 527

コメント 154, 180

関数ライブラリにおける 650

コレクション, プロパティの, 「プログラムの
記述」を参照

コンポーネント 121, 233, 121

アプリケーション領域の変更 138

印刷 140

関数ライブラリの指定 238

更新 494

作成 123

実行 487

実行結果の表示 501

実行の一時停止 475

手動 134

スクリプト・コンポーネントへ

ビジネス・コンポーネントの変換 148

ステップからの実行 493

ステップの移動 182

ステップの管理 182

ステップの削除 182

ステップの追加 153

設定の定義 227

定義 11

デバッグ 465

開く 130

保存 126

コンポーネント, 手動 11

コンポーネント, スクリプト, 「スクリプト・
コンポーネント」を参照

コンポーネントの実行

HTML への結果のエクスポート 516

印刷前の結果のプレビュー 514

コンポーネントの設定, 「ビジネス コンポーネ
ントの設定」ダイアログ・ボックスを
参照

コンポーネント・パラメータ 31, 166, 169, 172

出力 12

出力のパラメータ化 177

使用法 243

入力 12

入力のパラメータ化 173

標準設定値の定義 242

コンポーネント・リソース, 欠落 183

さ

サーバ

Quality Center, 接続 20

Quality Center との接続の解除 23

サーバ側の接続 678

リダイレクト 678

サーバのリダイレクト 678

最初にお読みください, QuickTest Professional
xiii

削除

アプリケーション領域 118

オブジェクト・リポジトリのオブジェ
クト 354

ステップ 182

リストからのオブジェクト 265

リポジトリ・パラメータ 418

作成

アプリケーション領域 79

コンポーネント 123

- 実行セッション中のテスト・オブジェクト 363
- サポート
 - ナレッジ・ベース xv
 - Web サイト xv
- サンプル・アプリケーション, Mercury Tours 14
- し
- 実行
 - 実行に必要なアクセス許可 13
 - 実行オプション, オプション・ダイアログ・ボックスの 199
- 実行環境
 - オブジェクト 657
- 実行結果
 - コンポーネントの結果の表示 501, 511
 - 手動による不具合の報告 533
- 実行, コンポーネント 487
 - Web サイト 690
 - 期待結果の更新 494
 - 高度な問題 676
 - 実行ダイアログ・ボックス 489
 - 実行の更新ダイアログ・ボックス 498
 - ステップから 493
- 実行, コンポーネントの
 - 期待された結果の更新 494
- 実行セッション
 - 回復シナリオの無効化 576
 - 結果の印刷 513
 - 結果の削除 524
 - テスト・オブジェクトの操作 362
 - テスト・オブジェクトの変更 363
 - プログラムによるテスト・オブジェクトの作成 363
- 実行設定オプション 210
- 実行ダイアログ・ボックス 489
- 実行, テスト
 - 結果の表示 507
- 実行の更新ダイアログ・ボックス 498
- 自動的に拡張, VBScript 構文 251
- シナリオ
 - 回復 539
 - 回復のコピー 573
 - 回復の削除 573
 - 回復の変更 572
 - 回復の保存 569
 - 回復の無効化 576
 - 回復の優先順位の設定 575
 - 回復プロパティの表示 571, 575
 - テストからの回復の削除 576
- 手動ステップ 134
- 出力オプション・ダイアログ・ボックス 169, 177
- 出力カラム 31
- 出力, キャンセル 171
- 出力セル 169
- 手動コンポーネント 11
- 手動ステップ 154, 180
- 手動テスト 34
- 詳細 Web オプション・ダイアログ・ボックス 221
- 情報表示枠 25, 39
- ショートカット
 - QuickTest の 57
 - エキスパート・ビューでの 253
 - オブジェクト・リポジトリ結合ツール内 435
 - 関数ライブラリでの 253
 - メニュー項目の 57
- ショートカット・キー
 - QuickTest の 57
- 初期化スクリプト 667
- 序数識別子 374
 - テスト・オブジェクトの序数識別子の指定 336
- 新規結合ダイアログ・ボックス 441
- す
- スキーマ, 結果の 534
- スクリプト・コンポーネント 11, 141
 - 作成 144
 - ビジネス・コンポーネントから変換 148
 - ビジネス・コンポーネントからの変換 148
- スクリプト・コンポーネントへの変換 141
- スクリプト, テスト「テスト・スクリプト」を参照
- スクリプトの生成オプション 671
- ステータス 233
- ステータス, コンポーネント 233

索引

ステータス・バー

QuickTest ウィンドウ 26

オブジェクト・リポジトリ結合ツール
433

ステートメントの完了 250

ステートメントの補完 620

ステップ

移動 182

コンポーネントのステップの管理 182

削除 182

手動 154, 180

追加 153

ステップ, オプション 500

ステップから開始 473

ステップ・コマンド 469

ステップでオブジェクトを選択ダイアログ・
ボックス 158

ステップの移動 182

ステップまで実行 473

スナップショット, コンポーネントのための
取得 234

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 380, 381

情報の分析 520

設定 381

スマート認識プロパティ・ダイアログ・ボッ
クス 371, 387

せ

正規表現

関数ライブラリでの使用 632

製品情報 68

製品情報ウィンドウ 68

製品情報ボタン 68

接続の解除, Quality Center との 23

セッション ID 678

設定レベル

カスタマイズ 261

標準 259

そ

操作

項目リストからの選択 156, 157, 164

ステップ用に選択 164

引数 166

操作カラム 30

操作セル 164

挿入ツールバー, QuickTest ウィンドウ 55

ソフトウェアの更新 16

た

ターゲット・リポジトリ 428

ターゲット・リポジトリ表示枠 431

タブ

移動 41

レイアウトのカスタマイズ 41

ち

置換

テスト・オブジェクトのプロパティ値
355

置換, 関数ライブラリでのテキストの置換 630

中位のイベント記録設定レベル 259

注釈カラム 31

チュートリアル xiii

つ

追加と削除ダイアログ・ボックス, オブジェ
クトの認識 387

ツール・ツールバー, QuickTest ウィンドウ 56

ツールバー

オブジェクト・リポジトリ結合ツール
435

ツールバー, QuickTest ウィンドウ

オートメーション 54

挿入 55

ツール 56

テスト 25

デバッグ 25, 54

表示 56

標準 53

編集 55

て

ディスク領域の節約 680

データ・テーブル 40

テキスト

関数ライブラリでの検索 628

関数ライブラリでの置換 630

- テスト・オブジェクト
 - アプリケーションからの選択 161
 - アプリケーション内での強調表示 358
 - 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 351
 - オブジェクト・リポジトリ内の場所の特定 355, 359
 - オブジェクト・リポジトリへの追加 343
 - 管理 299
 - 記述プロパティの削除 335
 - 記述プロパティの追加 329
 - 共有オブジェクト・リポジトリからの選択 158
 - 共有オブジェクト・リポジトリの管理 94
 - 共有オブジェクト・リポジトリの選択 98
 - 共有オブジェクト・リポジトリのプロジェクトへの追加 97
 - 検索 355
 - 項目リストからの選択 157
 - 識別 281
 - 実行セッション中の 362
 - 実行セッション中の作成 363
 - 実行セッション中の変更 363
 - 序数識別子の指定 336
 - 新規の定義 349
 - 新規プロパティの定義 333
 - 名前の変更 327
 - プログラムの記述を使った作成 363
 - プロパティ値の取得と設定 656
 - プロパティの変更 319
 - ローカル・リポジトリへのコピー 315
 - テスト・オブジェクトの識別 281
 - テスト・オブジェクトの詳細
 - 変更 314
 - テスト・オブジェクトの選択
 - アプリケーションからの 161
 - 共有オブジェクト・リポジトリからの 158
 - 項目リストからの 157
 - テスト・オブジェクトのプロパティ値
 - 置換 355
 - テスト結果
 - コマンド・ライン・オプションによる
 - 削除 527
 - 削除、テスト結果削除ツール 524
 - テストの結果の表示 501
 - パラメータ化された値 518
 - 表示のカスタマイズ 534
 - 報告、不具合 533
 - 有効化とフィルタ処理 663
 - テスト結果ウィンドウ 503
 - 概観 506
 - テーマ 506
 - テスト結果ツールバー 506
 - テスト結果ツリー 505
 - テスト結果削除ツール 524
 - テスト結果ツールバー、テスト結果ウィンドウ 506
 - テスト結果ツリー 505
 - テスト結果の表示、カスタマイズ 534
 - テスト実行エラー画面 555
 - テスト実行エラー・トリガ 548
 - テスト実行時間、向上 680
 - テスト・スクリプトのカスタマイズ 247
 - テスト・データベースの維持 667
 - テス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 671
 - テスト・バッチ、実行 500
 - デバッグ
 - 関数ライブラリ 465, 591
 - コンポーネント 465
 - 実行の一時停止 475
 - ステップから開始 473
 - ステップまで実行 473
 - ブレークポイントの設定 477
 - デバッグ・ツールバー、QuickTest ウィンドウ 25, 54
 - デバッグ・ビューア 40, 480
- と
- 統計情報ダイアログ・ボックス 449
 - 動作, DHTML 267
 - 動的な Web コンテンツ 677
 - 動的に生成される URL や Web ページ 677
 - 登録、関数 601
 - ドキュメント
 - オンライン xiv
 - 上級機能ユーザーズ・ガイド xiii

索引

閉じる

オブジェクト・リポジトリ 409

ドッキング表示枠 46

トランザクション終了ボタン 55

トリガ

アプリケーションのクラッシュ 548

イベント 540

オブジェクト状態 548

テスト実行エラー 548

ポップアップ・ウィンドウ 548

トリガ・イベントの選択画面 548

ドキュメントのみのオプション 34

な

名前

変更, テスト・オブジェクトの名前
327

名前と記述画面 567

名前の変更

テスト・オブジェクト 327

ナレッジ・ベース xv

に

二次リポジトリ 428

二次リポジトリ表示枠 432

は

パスワード, 暗号化 168

パスワード・エンコーダ・ダイアログ・ボック
クス 168

パフォーマンス, 向上 680

バブリング 268

パラメータ

オブジェクト・リポジトリの割り当て
338

コンポーネントの指定 239

作業 172

出力のキャンセル 171

未割り当てのオブジェクト・リポジ
トリの処理 187

リポジトリ 413

リポジトリ内で欠落 183

リポジトリの管理 414

リポジトリの削除 418

リポジトリの追加 416

リポジトリの変更 418

パラメータ化

プロパティ値, リポジトリ・パラメ
ータを使用 420

パラメータ化された値, テスト結果の表示 518

パラメータ, コンポーネント, 「コンポーネ
ント・パラメータ」を参照

パラメータ・タイプ

コンポーネント 172

ローカル 172

パラメータ, ローカル, 「ローカル・パラメ
ータ」を参照

パワー・ユーザ, 高度な機能 675

ハンドラ 267

ひ

引数の定義 600

ビジネス・コンポーネント・キーワード・

ビュー, 「キーワード・ビュー」を参照

ビジネス・コンポーネント, 「コンポーネ
ント」を参照

ビジネス・コンポーネント, 「コンポーネ
ント」を参照

ビジネス・コンポーネント, スクリプト・コ
ンポーネントへの 148

ビジネス・コンポーネント設定ダイアログ・
ボックス 227

ビジネス・コンポーネントの新規作成ダイア
ログ・ボックス 123

ビジネス・コンポーネントの設定ダイアロ
グ・ボックス 228

Web タブ 243

アプリケーション・タブ 235

回復タブ 245

スナップショット・タブ 234

パラメータ・タブ 239

プロパティ・タブ 231

リソース・タブ 238

ビジネス・コンポーネントの保存ダイアロ
グ・ボックス 126

ビジネス・コンポーネントを開くダイアロ
グ・ボックス 130

ビジネス・プロセス・テスト 12

必須プロパティ, 設定 369

表記規則 xvii

表現, 関数ライブラリでの使用 632

表示, コンポーネント実行結果 501

- 表示, 実行結果
 - コンポーネントの 511
 - 表示, 実行セッションの結果
 - テスト結果の印刷 513
 - 表示ツールバー 56
 - 表示, テスト結果
 - HTML へのテスト結果のエクスポート 516
 - 結果の絞り込み 509
 - テスト結果ウィンドウ 503
 - プレビュー, テスト結果 514
 - 標準設定のオブジェクト認識設定 380
 - 標準設定のプロパティ, 変更 281, 299
 - 標準ツールバー, QuickTest ウィンドウ 53
 - 標準で使用するイベント記録設定 259
 - 表示枠
 - 移動 41
 - 欠落リソース 39
 - 自動非表示 46
 - 情報 39
 - デバッグ・ビューア 40
 - ドッキング 46
 - フローティング 47
 - レイアウトのカスタマイズ 41
 - 開く
 - アプリケーション領域 81
 - コンポーネント 130
 - 開く, アプリケーション領域ダイアログ・ボックス 118
- ふ**
- フィードバック, Mercury へ送信 xv
 - フィルタ処理
 - オブジェクト, オブジェクト・リポジトリ・ウィンドウ内の 313
 - ターゲット・リポジトリ 455
 - 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 455
 - フィルタ・プロパティ (スマート認識) 381
 - フォント
 - キーワード・ビューでの変更 34
 - 不具合の報告 533
 - テスト結果からの 533
 - 複数のドキュメント
 - 作業 50
 - ブックマーク 626
 - ブラウザ
 - サポートされている 691
 - 無視する 214
 - ブラウザ詳細ダイアログ・ボックス 215
 - ブレイクポイント
 - 使用法 476
 - 設定 477
 - プレビュー, 関数のコード 608
 - プレビュー, テスト結果 514
 - フローティング表示枠 47
 - プログラミング
 - VBScript での 633
 - 関数ライブラリにおける 619
 - プログラムの記述 363, 640
 - Index プロパティの使用 648
 - WebElement オブジェクトの 647
 - 記述オブジェクトのための 644
 - ステートメントでの 641
 - 変数の使用 641
 - プロジェクト (Quality Center)
 - 接続 20
 - 接続解除 23
 - プロセスの選択画面 556
 - プロパティ
 - CreationTime 378
 - Index 376
 - 位置 377
 - 回復シナリオの表示 571, 575
 - 実行環境オブジェクト 657
 - テスト・オブジェクト記述からの削除 335
 - テスト・オブジェクト記述のプロパティの追加 329
 - テスト・オブジェクトの新規プロパティの定義 333
 - テスト・オブジェクトの変更 319
 - 標準設定 299
 - プロパティ標準設定 t 281
 - プロパティ・コレクション, 「プログラムの記述」を参照
 - プロパティ値
 - テスト・オブジェクト記述に指定 420
 - プロパティの追加と削除ダイアログ・ボックス 329
 - 分析, コンポーネント実行結果 501

索引

分析, 実行結果

HTML への実行結果のエキスポート
516

印刷前の結果のプレビュー 514

分析, テスト結果

結果の印刷 513

結果の絞り込み 509

テスト結果ウィンドウ 503

パラメータ化された値 518

へ

ページ/フレーム・オプション・ダイアログ・
ボックス 217

ヘルプ, オンライン, QuickTest Professional か
ら xiv

変換

オブジェクト・リポジトリ 405

変更

オブジェクト・リポジトリ 412

実行セッション中のテスト・オブジェ
クト・プロパティ 363

標準設定のプロパティ 281, 299

ライセンス 15

リポジトリ・パラメータ 418

編集ツールバー, QuickTest ウィンドウ 55

変数

グローバル・スコープ内で一意 616

ほ

報告, 不具合

自動 533

手動 533

ホーム・ページ, Mercury xv

補足プロパティ, 設定 369

保存

アプリケーション領域 116

オブジェクト・リポジトリ 407

回復シナリオ 569

コンポーネント 126

ターゲット・リポジトリ 459

ポップアップ・ウィンドウ・トリガ 548

ポップアップ・ウィンドウの条件を指定画面
550

ま

マニュアル

更新 xvi

チュートリアル xiii

マニュアル, 印刷

インストール・ガイド xiii

基本機能ユーザズ・ガイド xiii

チュートリアル xiii

み

右ボタン, マウス

QuickTest の記録設定 271

クリックの記録 271

む

無視するブラウザのリスト 214

ブラウザの削除 216

ブラウザの追加 215

ブラウザの変更 216

矛盾, 結合オブジェクト・リポジトリでの解
決 453

矛盾の解決

オブジェクト・リポジトリ結合ツール
453

め

メソッド

実行環境オブジェクト 657

新規の追加または振る舞いの変更 610

テスト・オブジェクトの表示 281

ユーザ定義 610

メソッドの登録 610

RegisterUserFunc ステートメントの使用
613

メソッドの登録解除, UnregisterUserFunc ス
テートメントの使用 614

メニュー・バー, QuickTest ウィンドウ 25

も

文字セット・サポート, Unicode 3

や

役割 12

ゆ

- ユーザ定義
 - 関数 581
 - テスト・オブジェクト, 割り当て 391
 - プロパティへのアクセス 658
 - メソッド 610
- ユーザ定義 Web イベント記録の設定ダイアログ・ボックス 261, 272
- ユーザ定義 Web イベント設定ファイル
 - 保存 275
 - 読み込み 276
- ユーザ定義オブジェクト, 割り当て 391
- ユーザ定義関数
 - ガイドライン 615
 - 関数定義ジェネレータ 597
 - 関数定義ジェネレータでのコードのレビュー 608
 - 仕上げ 608
 - 説明を付ける 606
 - 追加の生成 608
 - ツールチップの追加 606
 - 登録 601
- ユーザ定義のイベント記録設定 261
 - 手順 261
 - リストからのオブジェクトの削除 265
 - リストへのオブジェクトの追加 265
 - リッスン・イベントの追加 266
 - リッスン条件の指定 267
- 優先順位
 - 回復シナリオの設定 575

よ

- 用語, QuickTest Professional 11
- 用語集 11
- よくある質問 675
- 予約済みオブジェクト 594

ら

- ライセンス情報 15
- ライブラリ・ファイル, 「関数ライブラリ」を参照

り

- リソース, コンポーネント内で欠落 183

リポジトリ

- 欠落 183
- リポジトリ, 「オブジェクト・リポジトリ」参照
- リポジトリの種類 301
- リポジトリ・パラメータ 413
 - 値のパラメータ化 420
 - 管理 414
 - 削除 418
 - 追加 416
 - 変更 418
 - 割り当て 338
- リポジトリ・パラメータ・ダイアログ・ボックス 420
- リポジトリ・パラメータの管理ダイアログ・ボックス 414
- リポジトリ・パラメータの追加ダイアログ・ボックス 416
- リポジトリ・パラメータの割り当てダイアログ・ボックス 338

れ

- レイアウト
 - QuickTest ウィンドウのカスタマイズ 41
 - タブの移動 41
 - 標準設定の復元 49
 - 表示枠の移動 41
- レポート, フィルタの 663

ろ

- ローカル・オブジェクト
 - オブジェクト・リポジトリ・ファイルへのエクスポート 364
- ローカル・オブジェクト・リポジトリ 301, 303
 - オブジェクトのコピー 315
 - 結合 443
- ローカル・パラメータ 31, 166, 169, 172
 - 出力のパラメータ化 177
 - 出力パラメータの定義 12
 - 入力のパラメータ化 173
 - 入力パラメータの定義 12
- ロードされているアドインと関連付けられているアドイン 687

わ

- ワークフロー, QuickTest の使用 10

索引

割り当て

カスタム・オブジェクト 391

未割り当てのオブジェクト・リポジトリ 186

未割り当てのリポジトリ・パラメータ 187

リポジトリ・パラメータ 338