

HP Service Manager

Windows® および UNIX® オペレーティングシステム向け

ソフトウェアバージョン: 9.30

ウィザードガイド

ドキュメントリリース日: 2011年7月 (英語版)

ソフトウェアリリース日: 2011年7月 (英語版)



ご注意

保証

HP 製品、またはサービスの保証は、当該製品、およびサービスに付随する明示的な保証文によってのみ規定されるものとします。ここでの記載で追加保証を意図するものは一切ありません。ここに含まれる技術的、編集上の誤り、または欠如について、HP はいかなる責任も負いません。

ここに記載する情報は、予告なしに変更されることがあります。

権利の制限

機密性のあるコンピュータソフトウェアです。これらを所有、使用、または複製するには、HP からの有効な使用許諾が必要です。商用コンピュータソフトウェア、コンピュータソフトウェアに関する文書類、および商用アイテムの技術データは、FAR 12.211 および 12.212 の規定に従い、ベンダーの標準商用ライセンスに基づいて米国政府に使用許諾が付与されます。

著作権

© Copyright 1994 - 2011 Hewlett-Packard Development Company, L.P.

商標

Adobe™ は、Adobe Systems Incorporated の登録商標です。

Microsoft® および Windows® は、米国における Microsoft Corporation の登録商標です。

UNIX® は、The Open Group の登録商標です。

ドキュメントの更新情報

このガイドの表紙には、次の識別情報が記載されています。

- ソフトウェアのバージョン番号は、ソフトウェアのバージョンを示します。
- ドキュメント リリース日は、ドキュメントが更新されるたびに更新されます。
- ソフトウェア リリース日は、このバージョンのソフトウェアのリリース期日を表します。

最新の更新のチェック、またはご使用のドキュメントが最新版かどうかの確認には、次のサイトをご利用ください。

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

このサイトを利用するには、HP Passport への登録とサインインが必要です。HP Passport ID の取得登録は、次の Web サイトから行なうことができます。

<http://h20229.www2.hp.com/passport-registration.html>(英語 サイト)

または、HP Passport のログイン・ページの[**New users - please register**]リンクをクリックします。

適切な製品サポート・サービスをお申し込みいただいたお客様は、最新版をご入手いただけます。詳細については、HP の営業担当にお問い合わせください。

サポート

HP ソフトウェア サポート オンライン Web サイトを参照してください。

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

HP ソフトウェアが提供する製品、サービス、サポートに関する詳細情報をご覧ください。

HP ソフトウェアサポートオンラインでは、セルフソルブ機能を提供しています。お客様の業務の管理に必要な対話型の技術支援ツールに素早く効率的にアクセスいただけます。HP ソフトウェア サポート Web サイトのサポート範囲は次のとおりです。

- 関心のある技術情報の検索
- サポートケースとエンハンスメント要求の登録とトラッキング
- ソフトウェアパッチのダウンロード
- サポート契約の管理
- HP サポート窓口の検索
- 利用可能なサービスに関する情報の閲覧
- 他のソフトウェアカスタマとの意見交換
- ソフトウェアトレーニングの検索と登録

一部を除き、サポートのご利用には HP Passport ユーザとしてご登録の上、ログインしていただく必要があります。また、多くのサポートのご利用には、サポート契約が必要です。HP Passport ID の取得登録は、次の Web サイトから行なうことができます。

<http://h20229.www2.hp.com/passport-registration.html>(英語サイト)

アクセスレベルに関する詳細は、以下の Web サイトにアクセスしてください。

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

目次

ウィザードガイド	1
目次	5
ウィザードの概要	9
この文書の対象範囲	9
前提条件	9
ウィザードの作成について	9
ウィザードフロー	10
ウィザードとスクリプト	11
ウィザードツールの使用	13
ウィザードの実行順序	14
変数について	14
ウィザードで使用する変数	14
[ウィザード情報]タブのフィールドの定義	15
[ファイル選択]タブのフィールドの定義	17
[使用方法]タブのフィールドの定義	18
[アクション]タブのフィールドの定義	22
[メッセージ]タブのフィールドの定義	25
[変数]タブのフィールドの定義	26
[次のウィザード]タブのフィールドの定義	26
[前のウィザード]タブのフィールドの定義	26
[コメント]タブのフィールドの定義	27
[キャンセル式]タブのフィールドの定義	27
ウィザードの呼び出し	29
メニューからウィザードを呼び出す	29
表示オプションからウィザードを呼び出す	30
フォーマットコントロールからウィザードを呼び出す	31
プロセスレコードからウィザードを呼び出す	31
カスタムウィザードの作成例	33

SLA へのサービスレベル目標の追加.....	34
SLA ウィザードの設計.....	34
SLA ウィザードパネルの計画.....	36
SLA ウィザードの構築.....	38
ステップ 1: ウィザードの開始ノードを定義する.....	38
ステップ 2: ユーザに SLA タイプの選択を求める.....	40
ステップ 3: 顧客情報の入力を求める.....	41
タスク 1: 顧客情報を収集するウィザードの作成.....	42
タスク 2: 顧客情報を収集するフォームの作成.....	44
ステップ 4: SLA のタイトルと説明の入力を求める.....	46
タスク 1: SLA の説明用ウィザードの作成.....	46
タスク 2: SLA の説明用フォームの作成.....	48
ステップ 5: 有効期間の入力を求める.....	49
タスク 1: SLA 有効期間のウィザードの作成.....	50
タスク 2: SLA の有効期間用フォームの作成.....	52
ステップ 6: 契約文書について質問する.....	53
タスク 1: 契約について質問するウィザードの作成.....	53
タスク 2: 契約を追加するためのフォームの作成.....	56
ステップ 7: 契約の追加を求める.....	57
タスク 1: 契約を追加するためのウィザードの作成.....	57
タスク 2: 契約を追加するためのフォームの作成.....	59
ステップ 8: 新しい SLA を保存して SLO の追加を求める.....	60
タスク 1: 新しい SLA を保存するプロセスの作成.....	60
タスク 2: SLO の要件を確認するためのウィザードの作成.....	61
タスク 3: SLO の要件を確認するためのフォームの作成.....	63
ステップ 9: 可用性 SLO ウィザードを呼び出す.....	66
タスク: SLO を呼び出すウィザードの作成.....	66
ステップ 10: 可用性 SLO の追加を求める.....	68
タスク 1: 追加の可用性 SLO のウィザードの作成.....	68
タスク 2: 追加の可用性 SLO のフォームの作成.....	70
ステップ 11: 応答 SLO ウィザードを呼び出す.....	71
タスク: 応答 SLO を呼び出すウィザードの作成.....	71

ステップ 12: 応答 SLO の追加を求める.....	73
タスク 1: 追加の応答 SLO のウィザードの作成.....	73
タスク 2: 追加の応答 SLO のフォームの作成.....	76
ステップ 13: 新しい SLA 番号とタイトルの表示.....	77
タスク 1: 完了した SLA のウィザードの作成.....	77
タスク 2: 完了した SLA のフォームの作成.....	79
ステップ 14: 新規 SLA ウィザードを呼び出すメニューの設定.....	81
カスタムウィザードのトラブルシューティング.....	83
ボタンオプションが正しく表示されていません.....	83
ファイル変数がウィザードフローに渡されていません.....	83
原因.....	84
解決方法.....	84
フォーマットが見つかりません.....	84
原因.....	84
解決方法.....	84
JavaScript の構文エラー.....	84
原因.....	84
解決方法.....	85
クエリは正しい結果を返せませんでした.....	85
単一レコードではなく、複数のレコードがクエリによって返されました.....	85
原因.....	85
解決方法.....	85
RAD 式の構文エラー.....	86
原因.....	86
解決方法.....	87
レコードは正しくソートされていません.....	87
原因.....	87
解決方法.....	87
ウィザードが停止し、「続行できませんでした」というエラーが表示されます.....	87
トレースの実行.....	88

第 1 章

ウィザードの概要

ウィザードは、ユーザの入力を収集し、ユーザが Service Manager 内で多数の異なるタスクをすばやく効率的に実行する手段を提供します。Service Manager アプリケーションには、すぐに使用できるウィザードが多数含まれており、Service Manager カスタマイズツールには、独自のカスタムウィザードを設計し構築できるウィザード作成ツールが含まれています。カスタムウィザードは、さまざまな Service Manager アプリケーションおよびモジュールから呼び出すことができます。

この文書の対象範囲

カスタムウィザードの作成に使用するウィザードツールについて説明し、そのツールに含まれる各フィールドの定義を記述します。新しいサービスレベル契約を作成するサンプルのウィザードの計画と構築に関連するすべての手順を、詳細な例を通して解説します。トラブルシューティングの情報および一般的なエラーについても説明します。

前提条件

ウィザードツールを使用するには、以下の事項に関する十分な理解が必要です。これらの事項については、Service Manager のカスタマイズに関するドキュメントを参照してください。

- フォームの設計と作成
- Service Manager システム言語の構文の使用
- デバッグ目的のトレースの実行
- プロセスの作成および文書エンジンの使用

ウィザードの作成について

ウィザードツールによって実装担当者と管理者は、特定のタスクでユーザを支援するウィザードを Service Manager 内に追加できるようになります。特定のタスクとは、連絡先のデータベースへの追加、サービスデスクチケットの該当する Service Manager モジュールへのエスカレート、および構成管理に対する新しいアイテムタイプの作成などです。

ウィザードレコードには、次の一部またはすべての機能が含まれます。

- レコードの作成
- データベースからのレコードの選択
- 現在のレコードの変更
- 式 (RAD または JavaScript)、プロセス、およびフォーマットコントロールの実行
- ユーザ入力の要求
- ウィザードを重ねて順番に実行

設定済みのウィザードについてよく知ることは、ウィザードを作成する上で役立つでしょう。これらのウィザードは、次のようなさまざまな機能を提供します。

- ユーザの作成 (ユーザの簡易追加ユーティリティ)
- インタクションのエスカレート
- 変更の拒否
- CI タイプの追加
- 契約の追加

ウィザードを表示するには、[カスタマイズ]>[ウィザード]をクリックしてから、[検索]ボタンをクリックします。

繰り返しのユーザタスクを単純化するのに加えて、ユーザが前提となる情報を入力する必要があるプロセスに対してウィザードは非常に有用です。例えば、インタクションの作成サイクルでは、オペレータがデータを入力するよう、ウィザードパネルフローを表示できます。コールユーザが問題に関する質問にどのように答えるかに基づいて、次にどのパネルをオペレータに表示するかをウィザードが決定します。ウィザードの動作中に、オペレータが入力したデータはファイル変数に収集され、ウィザードが終了すると呼び出し元アプリケーションに返されます。

ウィザードはフォームを表示し、そのフォーマットコントロールレコードまたはプロセスを実行することができます。“初期化”に true に評価される条件があるフォーマットコントロールレコードの指示が、フォームが開く前に実行されます。“追加”に true に評価される条件のある指示が、ユーザが[OK]をクリックした後に実行されます。

ウィザードが非常に役に立つもう 1 つの場合は、結論に達するために複数の複雑な決定を行う必要があるときです。例えば、変更管理の承認条件は、通常は変更レコードの 1 つのフィールドのデータ内容に基づきます。そのような条件の例は、`risk.assessment in $L.file="1"` です。ただし、承認条件がいくつかの異なるフィールドの値に基づいている場合があります。例えば、承認条件が「division」、「area」、「department」の 3 つのフィールドに影響されるような場合も考えられます。これらのフィールド値の可能なすべての組み合わせを条件ステートメントにハードコード化するのは、非常に手間がかかり、保守もきわめて困難になります。ウィザードレコードを使えばこのような条件を定義できます。この場合、ウィザードはフォームを表示せず、サブルーチンを呼び出すか、標準の Service Manager の処理ステートメント (フォーマットコントロールの計算ステートメントと同様) を実行します。これらの方法はレコードデータを処理できます。実行時には、ウィザードはデジジョンツリーとして機能し、コーディングによる元の方法と比べると大幅に処理の節約になります。

ウィザードフロー

ウィザードフローは、ウィザードパネルが実行される順序を定義します。ウィザードパネルは開始から終了まで一直線に進むことも、いくつかの異なる処理フローに分岐することもできます。

条件ステートメントが true に評価されたときに特定のウィザードパネルを実行することで、より複雑なウィザードフローが実行できます。指定された条件がどれも true に評価されなかった場合、ウィザードフローは終了したと見なされ、コントロールは呼び出し元のアプリケーションに戻ります。

各ウィザードパネルで表示されるフォームの名前 (存在する場合) と、パネルからパネルへのフローをコントロールする条件を表示する、フロー全体のマップを作成することは非常に有用です。ウィザードの作成処理全般においてこのマップをたどることで、ウィザードが正しく実行されない原因となる単純なエラーを避けることができます。

ウィザードとスクリプト

Service Manager ウィザードには、過去の多くのリリースにおいてシステムカスタマイズユーティリティの一部だった従来のツール、Service Manager スクリプトユーティリティと多くの共通点があります。ウィザードはスクリプトより高性能で、機能性と柔軟性に優れています。

ウィザードがスクリプトに比べて非常に有利な点はセキュリティです。ウィザードはより強固でデータの整合性を保証します。それらはスクリプトには不可能です。

スクリプトとは違って、ウィザードではフォーマットコントロールおよび表示画面を指定できます。特定のフォーム用のフォーマットコントロール、または特定の表示画面に限定されることはありません。

さらに、ウィザード自身がフォーマットコントロールレコードおよびプロセスを呼び出すことができます。ウィザードからの RAD アプリケーションの直接呼び出しは許可されていません。

ウィザードツールの使用

ウィザードの作成準備ができれば、ウィザードツールを使用して、ウィザード全体を構成する各連続パネルを作成します。一般的に、各パネルはユーザに表示されるウィザードの画面または手順を表します。ただし、一部のパネルはユーザに対して非表示で、文書エンジンプロセスを呼び出すといったようにバックグラウンドで動作します。

ウィザードツールには 10 個のタブが表示されます。

- ウィザード情報
- ファイル選択
- 使用方法
- アクション
- メッセージ
- 変数
- 次のウィザード
- コメント
- キャンセル式
- 前のウィザード ([追加])をクリックした後でのみ表示されます)

[illegible]

ウィザードの実行順序

ウィザードを実行すると、ウィザードツールインタフェースに表示されるタブの順に実行されます(ウィザード情報、ファイル選択、使用方法、など)。

「ウィザードのタブの実行順序」という名前の図は、この順序を説明しています。[使用方法]タブでは、開発者はエンドユーザにウィザードパネルを表示するかどうかを決定します。パネルを表示する場合、実行順序はダイアグラムで示すようにエンドユーザがクリックするボタン、[次へ]、[前へ]、または[終了]に依存します。

エンドユーザが[前へ]ボタンをクリックすると、前のウィザードパネルが表示されます。またはウィザードツールの[前のウィザード]タブを使用して開発者によって設定されたその他のパネルが表示されます。

[終了]ボタンは、ワークフローの最後で使用されることを想定しています。ワークフローの途中でユーザがそれをクリックすると、ウィザードは連続したウィザードパネルの残りの部分を、ユーザにさらなる入力を求めることなく進めます。

[コメント]タブは内部の開発コメントに使用され、ウィザードの処理フローには影響を及ぼしません。

[ファイル選択]タブ内の、[初期式]および[JavaScript]は、[**\$L.file の選択基準**]タブの後に評価されます。つまり、[初期式]でクエリ変数を入力して、それを「**\$L.file の選択基準**」で使用することはできません。

変数について

スレッド間の変数の衝突の機会を排除するために(ウィザードが外部のアプリケーションを呼び出す場合は同じスレッド内でさえも)、ウィザードではローカル変数を使用することが最善です。ただし、ウィザードアプリケーションにとってローカルな変数は、表示アプリケーションでまたはウィザードによって呼び出されるプロセスレコード内では使用できません。

ローカル変数をウィザード変数として指定すると、変数のスコープが拡大され、表示アプリケーションや呼び出したプロセスが含まれるようになります。変数は、ウィザードフローが完了した時点でサーバによって自動的にクリーンアップされます。したがって、他のスレッドまたはプロセスとの間で変数の名前空間が衝突する危険はありません。

ウィザードで使用を避ける変数

ウィザードの作成時は、**wizard.run** RAD アプリケーションが使用するローカル変数は★ Mismatch Translation★

使用しないことを推奨します。★ Mismatch Translation★ これらの変数を使用すると、エラーおよび想定しない結果の原因となります。

ウィザードの作成時に使用を避ける変数を以下に示します。

\$L.action	\$L.action.file	\$L.action.file.save
\$L.action.save	\$L.addition	\$L.allow.finish
\$L.allow.skip	\$L.base.format	\$L.bitmap
\$L.category	\$L.disable.next	\$L.disable.next.previous
\$L.disable.previous	\$L.display.screen	\$L.ds.i

\$L.ds.junk	\$L.ds.nme	\$L.ds.var
\$L.env	\$L.eval	\$L.exit
\$L.fc	\$L.field.line.is	\$L.field.name.is
\$L.file	\$L.file.save	\$L.filename
\$L.finish	\$L.first.parent	\$L.format
\$L.format.name	\$L.gen.ms	\$L.i
\$L.link	\$L.list.field.name	\$L.lng
\$L.main.title	\$L.messages	\$L.mode
\$L.ms	\$L.msg	\$L.msg.cond
\$L.msg.lvl	\$L.msg.type	\$L.mult.sub
\$L.next.wizard	\$L.no.recs.msg	\$L.object
\$L.object.type	\$L.old.format.name	\$L.old.subformat
\$L.option	\$L.override.link	\$L.par.lng
\$L.parent	\$L.parent.file	\$L.parent.id
\$L.parent.object	\$L.pass.names	\$L.pass.values
\$L.passed.file	\$L.phase	\$L.prev.wizard
\$L.prompt	\$L.qbe	\$L.qbe.format
\$L.same	\$L.save.name	\$L.selected
\$L.selection	\$L.selection.filename	\$L.selection.save
\$L.selection.sort	\$L.selection.sql	\$L.sql
\$L.start	\$L.sub.format	\$L.title
\$L.var	\$L.window.title	\$L.wizard
\$L.wizard.bitmap	\$L.wizard.error.msg	\$L.wizard.format
\$L.wizard.name	\$L.wizard.names	\$L.wizard.prompt
\$L.wizard.sql	\$L.wizard.title	\$L.wizard.values
\$L.x	\$exit	\$normal
\$error		

[ウィザード情報]タブのフィールドの定義

このタブを使用して、エンドユーザに表示されるパネル内のウィザード名、説明、ウィンドウタイトル、および画像といった基本情報を入力します。

フィールド	説明
ウィザード名 <i>name</i>	<p>このフィールドは、dbdict の wizard テーブル内のウィザードに対して一意のキーを生成します。このウィザードレコードの一意な識別子として、ウィザードの簡潔で説明的な名前を入力します。ベストプラクティスとして、ウィザードのテーブル(またはモジュール)、ウィザードパネルが実行するアクション、およびウィザードが開始ノードであるかどうかなどのその他の詳細を含めます。たとえば、SLA 用ウィザードパネルを作成するための最初のウィザードパネル名は、"SLA Create 1 - Start - Type" のようになります。</p> <p>このフィールドは、必須です。</p>
開始ノード <i>start</i>	<p>このチェックボックスをオン (true に設定) にすると、これが一連のウィザードパネルの中で最初のものであることを示します。開始ノードとして設定されたウィザードがユーザの入力も要求する場合、ウィザードパネルがユーザに表示された時点で、[前へ] ボタンは自動的に無効になります。</p> <p>注: 開始ノードウィザードがユーザに表示されない場合、表示される最初のウィザード上の[前へ] ボタンを必ず無効にします。</p>
概要 <i>brief.description</i>	<p>重要な識別情報を含む、ウィザードの簡潔で意味のある説明を入力します。</p>
ウィンドウタイトル <i>window.title</i>	<p>ウィンドウタイトルは、ウィザードパネルがユーザに表示される場合に、ウィザードスレッドを保持するタブに表示されます。このフィールドは、scmessage データベースのメッセージ番号でも指定できます。例えば、scmsg(18,"wizard") です。</p> <p>注: 作成中のウィザードがローカライズ(別の言語に翻訳)される場合、scmessage データベースを使用することが推奨される方法です。</p>
タイトル <i>title</i>	<p>ウィザードパネルが表示されると、このフィールドに入力したウィザードのタイトルが左上に太字で表示されます。タイトルは、scmessage データベースのメッセージ番号で入力できます。例えば、scmsg(18,"wizard") などです。"Create SLA" のように、パネルの目的をユーザに示すタイトルを入力します。さらに詳細な情報が必要な場合は、"Create SLA - Enter Description" のように入力することができます。</p>
プロンプト <i>prompt</i>	<p>ユーザに表示されるウィザードパネルの場合、エンドユーザに指示するプロンプトを入力します。HP Service Manager は、この情報をパネルの中央 (タイトルとウィザードサブフォーマットの間) に表示し、パネルにどのような種類の情報を入力する必要があるのかをより詳しくユーザに指示します。プロンプトは Large タイプである必要があります。</p> <p>注: テキストでウィザードをハードコードするのではなく、scmsg(18,"wizard") のようなローカライズが可能なメッセージをタイトルやプロンプトに使用できます。これにより、複数言語の環境において、ローカライズされた言語が使用できます。</p>
ビットマップ <i>bitmap</i>	<p>ウィザードパネルの左側に表示する画像です。デフォルトの画像は、Service Manager のインストールディレクトリの次の場所にあります。Client\plugins\com.hp.ov.sm.client.common_<SM バージョンおよびビルド番号>\src\resources\icons\obj16\wizard<1,2,3>.gif</p> <p>別の画像を使用するには、その画像を上記のインストールパスに追加して、こ</p>

フィールド	説明
	のフィールドに画像ファイルの名前を入力します。
グローバルリスト <i>global.list</i>	[フィル]をクリックして構築または更新するグローバルリストを選択し、ウィザードの実行時に利用できるようにします。

[ファイル選択]タブのフィールドの定義

このタブを使用して、ファイル選択の前に実行するシステム言語または JavaScript コードを指定し、\$L.file の内容を決定します。タブには以下に説明する 3 つのサブタブが含まれます。[**\$L.file の選択基準**]サブタブには 4 つのメインオプションがあり、画面では上から下にリストされます。

フィールド	説明
[初期式]サブタブ <i>init.expressions</i>	\$L.file が初期化された後に評価される、システム言語構文を使用した式を入力します。これらの式を使用して変数を初期化します。
[JavaScript]サブタブ <i>javascript.init</i>	\$L.file が初期化された後に評価される、実行される Javascript のコードを入力します。このセクションを使用して変数を初期化するか、関連テーブルから値を選択します。
[\$L.file の選択基準]サブタブ	ウィザードが \$L.file 変数を選択する方法を指定するため、以下の 4 つのメインオプションから 1 つを選択する必要があります。
\$L.file なし (typecheck を使用) <i>file.selection</i>	このオプションは、ウィザードが呼び出された場所からウィザードにファイル変数が渡されないことを想定します。変数が渡されても無視されます。その代わりに、"typecheck" テーブル内のレコードとして初期化することにより \$L.file を作成します。それはシステムによってサポートされている異なるフィールドタイプに対応したフィールドを含む内部のテーブルです。 これは、入力する内容が少なく、データベースへの保存を必要としないシンプルなウィザードで頻繁に使用されます。
渡された \$L.file <i>file.selection</i>	このオプションを選択すると、ファイル変数 (\$L.file) がこのウィザードに、呼び出し元のアプリケーションまたはウィザードからパラメータとして渡されたことを示します。ウィザードが次のウィザードとして呼び出されると、ファイル変数は自動的に渡されます。\$L.file に保存された現在のレコードで引き続き作業を行う場合は、このオプションを選択する必要があります。
レコードの作成 <i>file.selection</i>	このオプションを選択すると、ウィザードが新しいレコードを作成することを示します。レコード [対象タイプ] フィールドに、dbdict 名を入力または選択する必要があります。これは \$L.file を、「対象タイプ」オプションで選択したテーブルのレコードの構造で初期化します。
対象タイプ (レコードの作成用) <i>create.record.filename</i>	[フィル]をクリックすると、作成されるレコードを定義する dbdict を選択します。

フィールド	説明
レコードの選択 <i>file.selection</i>	このオプションを選択すると、1 つまたは数多くのレコードを選択するクエリステートメントを使用します。
対象タイプ (レコードの選択用) <i>select.record.filename</i>	このフィールドは、クエリのレコードタイプを決定します。location のような、レコードの選択元のテーブルの dbdict 名を入力します。[フィル]をクリックして、すべての dbdict 名のリストから選択します。
使用するクエリ <i>select.query</i>	システム言語の構文を使用して、レコードの選択に使用するレコードを検索するクエリを入力します。例： "agreement.id="+\$G.new.sla 注：引用符 (") が必要です。
変数を解決する <i>resolve.variables</i>	このボックスを選択 (true に設定) すると、[レコードの選択] フィールドに入力されたすべての変数が、クエリの実行前に必ず評価されます。
レコードなしメッセージ <i>select.no.records</i>	クエリがレコードを返さなかった場合に表示されるテキストです。このフィールドはまた、メッセージデータベースからメッセージ番号を取得します。 例：scmsg (111, "wizard")

[使用方法] タブのフィールドの定義

このパネルを使用して、ウィザードの使用オプションを指定します。例えば、サブフォーマットの表示/非表示、表示するサブフォーマット、表示されるパネルで利用可能なボタンなどを指定します。

フィールド	説明
ウィザードの使用方法 セクション	ウィザードの実行時に HP Service Manager が実行する必要があるアクションを決定します。画面に表示される 5 つのオプションから 1 つを選択する必要があります。
リストからレコードを 1 つ選択 <i>wizard.type</i>	以下のリストから 選択基準 を選択します。
選択基準	リストからレコードを 1 つ 選択 を選択すると、以下のオプションの 1 つを選択することで、リストからレコードを選択できるようになります。リストは、選択したオプションに基づいて生成されます。
\$L.file を \$L.selection として使用 <i>use.file.as.selection</i>	このオプションは、\$L.file 変数を使用して、選択の実行対象となるリストのレコードを生成します。
レコードのクエリ <i>query.for.records</i>	このオプションでは、レコードのリストを生成するクエリを指定できます。クエリに必要なフィールドを以下に示します。
対象タイプ <i>query.for.records.filename</i>	クエリの実行対象となるテーブルの名前 (dbdict 名) を入力します。フィールド名の dbdict がクエリに対して有効であることを確認します。

フィールド	説明
使用するクエリ <i>query.for.records.query</i>	このオプションは、オプション「レコードのクエリ」および「対象タイプ」と組み合わせて使用します。Service Manager システム言語の構文を使用したクエリを入力します。
ソート基準 <i>query.for.records.sort</i>	このオプションを使用すると、このオプションで指定したフィールドに基づいてソートした順序で、リストを表示できます。選択した dbdict に対して有効でかつ昇順にソートされるフィールド名を入力します。
レコードなしの場合 <i>query.select.no.records</i>	<p>このオプションでは、リストにレコードが選択されない場合のアクションを指定できます。アクションは次のとおりです。</p> <p>戻る: 前のウィザードパネルがある場合はそこへ、またはウィザードを呼び出した場所に (これが最初のパネルの場合) 戻ります。</p> <p>続行: 空のリストを表示して、続行できます。</p> <p>表示をスキップ: ウィザードパネルをスキップして、次のパネルがある場合は移動します。</p>
レコードなしメッセージ <i>query.no.records.msg</i>	このオプションでは、リストにレコードが選択されない場合に表示するメッセージを表示できます。scmessage レコードは、メッセージを構成するのに使用できます。
レコード 1 個の場合 <i>query.select.one.record</i>	このオプションでは、リストにただ 1 つのレコードがある場合のアクションを指定できます。リストに 1 レコードを表示するか、またはユーザに選択を表示することなしに現在のファイル変数として使用できます。
[スキップ] オプションを許可 <i>allow.skip</i>	選択 (true に設定) すると、ユーザはパネルをスキップできます。
ユーザ入力の要求 <i>wizard.type</i>	このフィールドは、ユーザが情報の入力を求められることを示します。このオプションを選択すると、 表示するサブフォーマット エントリが必須になり、 [ウィザード情報] タブに手順を伴ったプロンプトを入力する必要があります。
表示をスキップ <i>wizard.type</i>	「表示をスキップ」は、現在のウィザードがユーザの入力なしに現在のレコードを操作する場合に使用する必要があります。一般的に、ユーザに表示されないウィザードは、複雑なウィザードワークフローの一部です。
すぐにキャンセル <i>wizard.type</i>	<p>このオプションは、ウィザードがウィザードの使用を意図しないユーザタイプによって開始された場合に、即座に処理をキャンセルする場合に使用します。</p> <p>例: 会社は、上級管理職の顧客から情報を収集するウィザードを作成します。たとえば、基準に適合しないユーザがウィザードの開始を選択してウィザードパネルを進んでいくとします。そのユーザが情報を収集する必要がないことが明白になると、「すぐにキャンセル」オプションによってプロセスは即座に終了します。</p>

フィールド	説明
リストからレコードを複数選択 <i>wizard.type</i>	このオプションを選択するとリストが表示され、ユーザはリストから複数のレコードを選択できます。リストは、「選択基準」セクションで選択したオプションに基づいて生成されます。
複数選択フィールド <i>mult.field.name</i>	このオプションは、ユーザが複数のレコードを選択できるリスト内のフィールドを決定します。
表示するサブフォーマット <i>sub.format</i>	ウィザードフォーマット内で、ユーザ入力用に表示するフォーマットの名前を入力します。

フィールド	説明
メインフォーマット (デフォルトは Medium) <i>form.Name</i>	<p>ウィザード作成ツールは、ウィザードのサイズとボタンの配置に影響を与える、複数のフォーマットの選択肢を提供します。HP Service Manager は次のウィザードフォーマットオプションを提供します。フォーマットが選択されない場合、デフォルトは“Medium”です。</p> <p>Small (wizard.small) - このウィザードは、テキストおよびオプションに提供するスペースが最小限だけでなく、必要な画面のスペースも最小限です。このウィザードは、低解像度の環境で実行中のクライアントに理想的です。サブフォーマットが使用できるスペースが狭いため、サブフォーマットを小さくする必要があります。利用可能なボタンは、[前へ]、[次へ]、[終了]、[キャンセル]で、フォームの下部に横に一行に配置されます。</p> <p>Medium (wizard) - このウィザードはほとんどのウィザードタスクに対して十分なスペースを提供します。利用可能なボタンは、[前へ]、[次へ]、[終了]、[キャンセル]で、フォームの下部に横に一行に配置されます。</p> <p>Large (wizard.large) - このウィザードは、テキストおよびオプションに提供するスペースが最大限だけでなく、必要な画面のスペースも最大限です。このウィザードは、高解像度の環境で実行中のクライアントに理想的です。これは、表示するサブフォーマットに対して最大限のスペースを提供しますので、より複雑なサブフォームが使用できます。利用可能なボタンは、[前へ]、[次へ]、[終了]、[キャンセル]で、フォームの下部に横に一行に配置されます。</p> <p>OK Only - Small (wizard.okonly.small) - ウィザードサイズ Small を使用して、タスク内の移動に使用するボタンを制限します。利用可能なボタンは、[OK]と[キャンセル]で、フォームの下部に横に一行に配置されます。</p> <p>OK Only - Medium (wizard.okonly) - ウィザードサイズ Medium を使用して、タスク内の移動に使用するボタンを制限します。利用可能なボタンは、[OK]と[キャンセル]で、フォームの下部に横に一行に配置されます。</p> <p>ボタンを上部に配置 (wizard.buttonsOnTop) - ウィザードサイズ Medium を使用して、ボタンをウィザードの上部に沿って配置します。利用可能なボタンは、[前へ]、[次へ]、[終了]、[キャンセル]で、フォームの上部に横に一行に配置されます。</p> <p>ボタンを右端に配置 (wizard.buttonsOnSide) - ウィザードサイズ Medium を使用して、ボタンをウィザードの右端に沿って配置します。利用可能なボタンは、[前へ]、[次へ]、[終了]で、フォームの右側に縦に配置されます。</p>
表示アプリケーション (デフォルトは wizard.display) <i>display.screen</i>	<p>ウィザード内で利用可能なボタンおよびメニューオプションをカスタマイズする場合は、表示画面を入力します。デフォルトの表示画面は <i>wizard.display</i> です。</p>

フィールド	説明
<p>[終了]オプションを有効にする</p> <p><i>allow.finish</i></p>	<p>このオプションは一連のウィザードの最後に使用します。このチェックボックスを選択すると、ウィザードパネルに[終了]ボタンが表示されます。</p> <p>注: このウィザードの後に表示されるウィザードパネルが、オプションの情報提供のみの場合は、ワークフローで最後のウィザードよりも前に[終了]ボタンを表示することは可能です。[終了]ボタンは、ユーザーから必要な情報がすべて収集されるまでは有効にできません。</p>
<p>[次へ]および[前へ]ボタンをオフにする</p> <p><i>disable.next.previous</i></p>	<p>このチェックボックスをオンにすると、[次へ]および[前へ]ボタンがウィザードで利用不可になり、ユーザーは[次へ]または[前へ]をクリックしてウィザード内で前後に移動することはできません。</p> <p>注: このチェックボックスは、ワークフローの一番最後のウィザードパネルでのみオンにしてください。また、[次へ]および[前へ]ボタンが無効な場合、[終了]ボタンを有効にする必要があります。</p>
<p>[次へ]ボタンだけをオフにする</p> <p><i>disable.next</i></p>	<p>このチェックボックスをオンにすると、[次へ]ボタンがウィザードで利用不可になり、ユーザーは[次へ]をクリックしてウィザード内で先に進むことはできません。</p>
<p>[前へ]ボタンだけをオフにする</p> <p><i>disable.previous</i></p>	<p>このチェックボックスをオンにすると、[前へ]ボタンがウィザードで利用不可になり、ユーザーは[前へ]をクリックしてウィザード内で前に戻ることはできません。</p> <p>注: このオプションは、ウィザードの最初に表示されたパネルが、開始ノードでない場合に使用してください。開始ノードで[前へ]を選択すると、適切な取り消し処理を経ることなしにウィザードが終了します。</p>
<p>条件によって前のウィザードへ戻る機能を使用する</p> <p><i>previous.cond</i></p>	<p>このオプションを選択すると、[前のウィザード]タブがフォームに表示されます。[前のウィザード]タブでは、[前へ]ボタンをクリックした時に考えられるさまざまな終了方法を定義します。</p>
<p>キャンセル時にプロンプトを表示しない</p>	<p>ウィザードをキャンセルで終了するときに、ユーザーにプロンプトを表示するかどうかを決定します。[戻る]では、ユーザーはパネルに戻ります。</p>

[アクション]タブのフィールドの定義

このタブを使用して、実行されるプロセスまたはフォーマットコントロール、および実行対象のファイル変数またはレコードを含む、ウィザードのアクションを設定します。

フィールド	説明
<p>初期プロセス</p> <p><i>initial.process</i></p>	<p>このオプションでは、実行されるアクションより前に実行されるプロセス定義を指定します。プロセス定義の詳細については、文書エンジンのマニュアルを参照してください。</p>

フィールド	説明
アクションの実行対象 セクション	<p>3つのオプションから1つを選択し、実行するアクションによって影響を受けるレコードを指定します。ファイルは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。アクションはウィザードで定義され、ユーザがウィザードで[次へ]をクリックした時に発生することを定義します。</p> <p>注:[ファイル選択]タブで選択した項目によって、ユーザが[次へ]をクリックした時に実行されるアクションを、現在のファイルまたはレコードの選択に対して実行するかを選択できます。レコードの選択 (\$L.selection) は、[ファイル選択]タブで[レコードの選択]を選択した場合に \$L.selection ファイル変数がフィルされた場合で、なおかつ [使用方法] タブの一覧から[リストからレコードを1つ選択]または[リストからレコードを複数選択]のいずれかを選択した場合にのみ選択する必要があります。</p>
現在のファイル (\$L.file) <i>perform.action.on</i>	<p>アクションは \$L.file ファイル変数で表される単一のレコードに対して実行されます。現在のファイルは、渡され、作成され、または単一レコードのみを返すクエリで選択されている可能性があります。</p> <p>注:[実行するアクション]セクションで定義されたアクションは、[ファイル選択]タブおよびまたは[使用方法]タブで選択された現在のファイルに適用されます。</p>
選択したファイル (\$L.selection) <i>perform.action.on</i>	<p>[実行するアクション]セクションで定義されたアクションは、[ファイル選択]タブで選択されたレコードのグループに適用されます。</p>
選択したファイル (\$L.selection) 内の 各レコード <i>perform.action.on</i>	<p>[実行するアクション]ブロックで定義されたアクションは、選択した各レコードに適用されます。</p>
実行するアクション 実行 セクション	<p>このセクションの3つのサブタブを使用して、選択したファイルに実行するアクションを示します。</p>
[式]サブタブ <i>expressions</i>	<p>初期プロセスの後で、JavaScript およびフォーマットコントロール (またはプロセス) の前に、ウィザードの一部として実行する必要がある式を入力します。これらの式は、次のような Service Manager システム言語の構文を使用します。</p> <pre>name in \$L.file="Test"</pre> <p>これらの式は、[現在のファイルを選択したファイルにリセット]アクションが実行される前に評価されます。したがって、ユーザが選択したレコードを操作する場合、ここで実行されるアクションは \$L.file ではなく \$L.selection を使用する必要があります。</p> <p>注:これらの式は、フィールドの情報を保存するといった、選択したレコードを使用できます。</p>

フィールド	説明
[JavaScript] サブタブ <i>javascript.actions</i>	JavaScript 式はウィザードの一部として実行され、初期プロセスおよび式の後で、次のサブタブのフォーマットコントロールまたはプロセスが入力される前に実行されます。これらの JavaScript 言語式は、現在のファイル変数または初期化変数の操作に使用できます。
[フォーマットコントロール / プロセス名] サブタブ	
フォーマットコントロール <i>format.control</i>	使用するフォーマットコントロールレコード。式および JavaScript を一通り実行した後かつプロセスを呼び出す前に、クエリ、計算、検証、またはサブルーチンを実行するフォーマットコントロールレコードを入力できます。
対象タイプ <i>format.control.type</i>	このウィザードで使用するフォーマットコントロールタイプは、アプリケーションの条件を評価し、それによりフォーマットコントロールは、フォーマットコントロールレコードで評価されるアクション ([追加]、[更新]、または[削除]) で実行できます。
検証失敗の場合 <i>bad.validation.action</i>	<p>フォーマットコントロールのデータの検証に失敗した場合、ユーザを現在のウィザードパネルに戻して正しい情報を入力させるか、失敗した検証を無視して次のパネルでウィザードを続行するかを選択します。</p> <p>*[戻る]では、ユーザはパネルに戻ります。失敗した検証を修正できる機会がユーザにある場合にのみ使用します。ウィザードが、情報を入力するサブフォーマットを表示することでユーザに情報を要求しない場合に、[戻る]を選択すると結果として無限ループに陥ります。</p> <p>*[続行]でユーザは続行できます。</p>
プロセス名 <i>process.name</i>	このウィザードパネル上で実行するプロセスの名前を入力します。詳細については、文書エンジンのマニュアルを参照してください。
現在のファイルを選択したファイルにリセット <i>reset.to.selections</i>	<p>選択すると(trueに設定)、現在のファイル変数 (\$L.file) をユーザが選択した値 (\$L.selection) で置き換えます。</p> <p>注: このフィールドは、[使用方法]タブでの選択に依存します。[リストからレコードを1つ選択]または[リストからレコードを複数選択]が選択されると、これらのオプションによって \$L.selection が生成されます。 \$L.selection</p>
パネルを再起動する条件 <i>restart.condition</i>	ある状況下においては、ウィザードパネルを再度実行する必要がある場合があります。たとえば、予想されたフィールドが正しくフィルされていない場合、またはフォーマットコントロールまたはプロセスからエラーコードが返された場合です。このパネルを再実行する必要があるかどうかを判断するための、true または false に評価される式を入力します。
完了時にレコードを表示 <i>display.when.complete</i>	ウィザードの完了時に、ユーザに対してレコードを表示するかどうかを決定します。条件がtrueの場合、HP Service Manager は完了時にユーザにレコードを表示します。条件がfalseなら、レコードは表示されません。

フィールド	説明
モード <i>display.mode</i>	<p>このフィールドは、ユーザに対するレコードの表示方法を決定します。</p> <p>*ブラウズ: (これがデフォルトです)。ブラウズモードには、変更したレコードを保存するボタンは何もありません。</p> <p>*[追加]: [追加] ボタンが使用できるようになります。</p> <p>*[更新]: [保存] ボタンが使用できるようになります</p>
呼び出し元アプリケーションに現在のファイルを返す <i>reset.current.file</i>	<p>このオプションは、さらなる処理のためにウィザードを呼び出したアプリケーションに、現在のレコードを戻します。</p>

[メッセージ] タブのフィールドの定義

このタブを使用して、指定した条件が発生した時にエンドユーザに表示される任意のメッセージを設定します。

フィールド	説明
メッセージ <i>message</i>	<p>特定の条件が発生した時に表示されるメッセージ。このフィールドはまた、scmessage データベースからメッセージ番号を取得します。</p> <p>例: <code>scmsg(106, "fc")</code></p>
条件 <i>message.cond</i>	<p>メッセージを発行する必要があるかどうかを判断するための、true または false で評価するメッセージの条件を入力します。例:</p> <p><code>not nullsub(\$L.finish, false)</code></p>
タイプ <i>message.type</i>	<p>ユーザに表示するメッセージのフォーマットの選択 (ポップアップまたは画面)。デフォルトは画面です。</p>
レベル <i>message.level</i>	<p>メッセージの重要度のレベルを選択します。</p> <p>*情報: 情報のみ (デフォルト)</p> <p>*アクション</p> <p>*エラー: 深刻なエラーが発生した場合にユーザに送られるエラーメッセージ。</p>

[変数]タブのフィールドの定義

このタブを使用して、ウィザードワークフローで使用する変数を文書化します。

フィールド	説明
ウィザード変数 wizard.variables	<p>このタブを使用して、ウィザードワークフローで使用する変数を文書化します。変数は、ウィザードで作成中のアプリケーションまたはフォーマットに渡すことができますが、割り当てる必要があります。たとえば、\$L.return.action などの変数は、ウィザードで使用する場合はここで定義する必要があります。</p> <p>変数のデータ型および値は、場合によって異なる可能性があります。変数は基本データ型または複合データ型を値として取ることができます。Service Manager には、以下の 3 種類の変数があります。1) ローカル、2) グローバル、3) スレッドです。ローカル変数は、\$L で始まり、現在実行している RAD アプリケーション内でのみ持続します。サーバは、RAD アプリケーションを終了する時にローカル変数を自動でクリーンアップします。</p>

[次のウィザード]タブのフィールドの定義

このタブを使用して、呼び出される次のウィザードパネルを指定します (ある場合)。

フィールド	説明
ウィザード名 next.wizard	ワークフロー(一連のウィザードパネル)内で次に実行するウィザードの名前。
条件 next.wizard.cond	true または false に評価される条件を入力します。ワークフローの次のウィザードは、true と評価される最初の条件によって決定されます。続く条件が同じように true と評価される場合でも関係ありません。すべての条件が false と評価された場合、ウィザードワークフローは終了します。

[前のウィザード]タブのフィールドの定義

このタブは、[使用方法]タブで[条件によって前のウィザードへ戻る機能を使用する]チェックボックスをオンにした場合に表示されます。

フィールド	説明
ウィザード名 prev.wizard	ユーザが終了する必要があるウィザードの名前を指定します。
条件 prev.wizard.cond	ユーザをこの前のウィザードに移動する必要がある場合の条件を設定します。true または false に評価される条件が必要です。

[コメント] タブのフィールドの定義

このタブを使用して、ウィザードに関する内部の開発コメントを入力します。これらのコメントはウィザードのワークフロー自体には何の影響もありません。

フィールド	説明
コメント <i>comments</i>	ウィザードに関する開発者のすべてのコメントをここに入力します。ベストプラクティスとして、[コメント]タブは詳細を記述するのに使用します。たとえば、ウィザードパネルの目的、その目的を実現するために選択された設定、使用した変数、ウィザードワークフロー、および終了条件などです。

[キャンセル式] タブのフィールドの定義

エンドユーザが[キャンセル]ボタンを押した場合に実行される、システム言語式または JavaScript コードを入力します。これにより、ウィザードで使用された変数がクリーンアップされ、次のウィザード実行時に正しい動作が行われるようにします。

フィールド	説明
[キャンセル時に実行される式]サブタブ <i>cancel.expressions</i>	ユーザがウィザードのプロセスをキャンセルした場合に、実行する式を入力します。これにより、ウィザードの作成者がウィザードで初期化された値をリセットまたは変数をクリーンアップする機会が与えられます。
[キャンセル時に実行される JavaScript]サブタブ <i>javascript.cancel</i>	ウィザードで初期化された、値をリセットまたは変数をクリーンアップするのに使用する必要がある JavaScript コードを入力します。

第 3 章

ウィザードの呼び出し

ウィザードの実行を担当する RAD アプリケーションは **wizard.run** と呼ばれます。このアプリケーションは次の場所から呼び出すことができます。

- メニュー
- 表示オプション
- フォーマットコントロール
- プロセスレコード

以下のセクションでは、これらの場所からウィザードを呼び出すために必要な手順を説明します。

ウィザード自体が呼び出せるものは次のとおりです。

- フォーマットコントロールレコード
- プロセス
- その他のウィザード

ウィザードからの RAD アプリケーションの直接呼び出しは許可されていません。

警告： ウィザードが無限ループに入ってしまう可能性があります。たとえば、あるプロセスから `wizard.run` を呼び出して、元のプロセスを呼び出すウィザードを実行すると無限ループが発生します。

メニューからウィザードを呼び出す

メニューからウィザードを呼び出すよう設定するには、次の手順を実行します。

1. **[カスタマイズ]>[カスタマイズツール]>[メニュー]** をクリックします。
2. メニューレコードを選択します。
3. オプション番号、グループ、必要に応じてメニューに表示する説明と名前を追加します。
4. アプリケーションに対して「**wizard.run**」と入力します。
5. **[パラメータ名]** フィールドには、アプリケーションに渡すパラメータの名前を指定します。アプリケーションに渡すパラメータの名前は、引用符 `"` と波括弧 `{ }` で囲む必要があります。`wizard.run` アプリケーションで利用可能なパラメータは、`file`、`name`、`text` および `prompt` です。
6. **[パラメータ値]** フィールドには、渡されるパラメータの値を指定し、引用符 `"` と波括弧 `{ }` で囲みます。例えば、`{"createUser Pre"}` と入力します。

例：

この例では、ウィザードは**[システム管理]**メニューから、**[簡易追加ユーティリティ]**オプションで呼び出されるように設定します。ウィザードの設定が完了したら、**[システム管理]>[進行中のメンテナンス]>[ユーザの簡易追加ユーティリティ]**をクリックします。

ユーザがメニューの[ユーザ簡易追加ユーティリティ]をクリックすると、情報の入力およびオペレータを追加するタスクの完了を支援するウィザードが、事前に定義された手順でオープンします。

メニューからウィザードを呼び出す設定の内容を表示するには、次の手順を実行します。

1. [カスタマイズ]>[カスタマイズツール]>[メニュー]をクリックします。
2. SYSTEM ADMINISTRATION メニューレコードを開きます。
3. 該当レコードの[説明]フィールドには[User Quick Add Utility]と表示されます。
4. [アプリケーション]フィールドには、メニューから呼び出される RAD アプリケーションである「**wizard.run**」を指定します。
5. [パラメータ名]フィールドには、メニュー項目に渡される単一のパラメータの名前が指定されています。この例では、{"name"} はウィザードの名前を表します。名前は引用符 "" と波括弧 {} で囲まれます。
6. [パラメータ値]フィールドには、渡されるパラメータの値を指定します。この例で、{"createUser Pre"} です。名前は引用符 "" と波括弧 {} で囲まれます。

表示オプションからウィザードを呼び出す

ウィザードが表示オプションから呼び出されるように設定するには、次の手順を実行します。

1. [カスタマイズ]>[カスタマイズツール]>[表示オプション]をクリックします。
2. ウィザードの呼び出しに使用する表示オプションを選択します。
3. [RAD]タブを選択します。
4. [RAD アプリケーション]フィールドに、RAD アプリケーション「**wizard.run**」を入力します。
5. [名前]フィールドには、RAD アプリケーションに渡すパラメータを指定します。選択できるパラメータは、file、name、text および prompt です。**name** パラメータは必須です。
6. [値]フィールドには、RAD アプリケーションに渡すパラメータの値を指定します。
7. [保存]をクリックします。
8. [OK]をクリックします。

例: 表示オプションから呼び出されるウィザードの設定

この例では、ウィザードが表示アプリケーション **wizard.display.kmsearch** から呼び出されるように設定し、オペレータが"Search Again"オプションを選択すると、ウィザード"KM choose shared content"が呼び出されるようにします。

ウィザードが表示オプションから呼び出されるように設定するには、次の手順を実行します。

1. [カスタマイズ]>[カスタマイズツール]>[表示オプション]をクリックします。
2. **wizard.display.kmsearch** の、[テキストオプション]定義が 7500 のレコードを選択します。
3. [RAD]タブを選択します。
4. [RAD アプリケーション]フィールドに、RAD アプリケーション「**wizard.run**」を入力します。
5. [名前]フィールドには、RAD アプリケーションに渡すパラメータを指定します。選択できるパラメータは、file、name、text および prompt です。**name** パラメータは必須です。

6. この例では、名前が指定されている各パラメータに対して以下の値を指定します。

パラメータの名前	パラメータの値
file	\$L.file
name	KM choose shared content

7. [保存]をクリックします。
8. [OK]をクリックします。

フォーマットコントロールからウィザードを呼び出す

フォーマットコントロールからウィザードを呼び出すには、次の手順を実行します。

1. [カスタマイズ]>[フォーマットコントロール]をクリックします。
2. ウィザードを呼び出すフォーマットコントロールを選択します。
3. [サブルーチン]をクリックします。
[フォーマットコントロールメンテナンス - サブルーチン]フォームがオープンします。
4. [アプリケーション]フィールドに、アプリケーション「**wizard.run**」を入力します。
5. [名前]フィールドに、サブルーチンアプリケーションにデータを渡すパラメータの名前 (name、file、text および prompt) を入力します。
6. [値]フィールドに、サブルーチンアプリケーションに渡すパラメータを入力します。このフィールドのパラメータの種類は、[名前]フィールドのパラメータの種類と一致する必要があります。
7. 各パラメータについて条件を指定します。条件が "true" に設定されている場合にユーザが[保存]をクリックすると、フォーマットコントロールは、レコードを更新する前に、[更新]フィールドが true に設定されているすべてのセクションまたは式を実行します。

プロセスレコードからウィザードを呼び出す

プロセスレコードからウィザードを呼び出すよう設定するには、次の手順を実行します。

1. [カスタマイズ]>[文書エンジン]>[プロセス]をクリックします。
2. プロセスレコードを選択します。
3. [RAD]タブを選択します。
4. [RAD アプリケーション]フィールドに、ウィザードをこのレコードに関連付ける RAD アプリケーション「**wizard.run**」を入力します。
5. [条件]フィールドには、[RAD アプリケーション]フィールドに関連づける条件を入力します。例えば、"true" です。
6. [パラメータ名]フィールドには、RAD アプリケーションに渡すパラメータの名前を入力します。
7. [値]フィールドには、RAD アプリケーションに渡すパラメータの値を入力します。
8. [保存]をクリックします。
9. [OK]をクリックします。

例：

この例では、add.device プロセスレコードからウィザードを呼び出すように設定します。add.device プロセスレコードは RAD 呼び出しを評価するためにオープンされます。

プロセスレコードからウィザードを呼び出すよう設定するには、次の手順を実行します。

1. [カスタマイズ]>[文書エンジン]>[プロセス]をクリックします。
2. **add.device** プロセスレコードを開きます。
3. [RAD]タブを選択します。
4. [RAD アプリケーション]フィールドに、ウィザードをこのレコードに関連付ける RAD アプリケーション「**wizard.run**」を入力します。
5. [条件]フィールドには、使用されるパラメータの名前(ファイルおよび名前)が含まれます。
6. [パラメータ名]フィールドには、RAD アプリケーションに渡すパラメータの名前を入力します。
7. [値]フィールドには、RAD アプリケーションに渡すパラメータの値を指定します。この例では、名前が指定されている各パラメータに対して以下の値を指定します。

パラメータの名前	パラメータの値
file	\$L.file
name	"Add Device"

8. [保存]をクリックします。
9. [OK]をクリックします。

カスタムウィザードの作成例

初期設定では、Service Manager ユーザは[サービスレベル管理]>[サービスレベル契約]>[新規 SLA の作成]をクリックして新しい SLA を作成します。ユーザは次に、サービスレベル契約エントリフォームの、必須のフィールドおよび関連する省略可能なフィールドに値を入力します。

[illegible]

HP Service Manager(9.30)

SLA へのサービスレベル目標の追加

SLA には一般的に、関連付けられたサービスレベル目標 (SLO) があります。Service Manager には、応答 SLO および可用性 SLO を追加するための 2 つの設定済みウィザードがあります。

サービスレベル契約のフォームに入力して SLA を作成した後は、ユーザは SLA レコードに移動して、[応答] タブおよび[可用性] タブで[SLO を追加] ボタンをクリックし、設定済みの Add Response and Add Availability SLO(応答 SLO と可用性 SLO を追加) ウィザードを起動します。応答 SLO を追加するための設定済みウィザードを起動するこのボタンは、次の画面に表示されます。

サービスレベル契約

SLA ID: 168
タイプ: 顧客
顧客: advantage
サービス契約:
サービス時間:
タイトル: Base Monitoring SLA for IT services
説明:
SLA が有効になる日付: 08/01/01 01:00:00
有効期限: 15/01/01 01:00:00

◆ 応答 ◆ 可用性 ◆ SLA ◆ 添付ファイル ◆ 履歴

応答目標

タグから SLO を追加 SLO を追加 SLO を編集 SLO を削除

SLO ID	サービス範囲	SLO 名	説明	初期ステータス	最終ステータス
179	インシデント	All Levels - inci...		Open	Resolved
180	問題	All Levels - pro...		Problem Detection, ...	Problem Investigation and Diagnosis
218	インシデント	KPI Availability ...	SLO for Availabi...	Open	Closed
219	インシデント	KPI Availability ...	SLO for Availabi...	Open	Closed
220	インシデント	KPI Availability ...	SLO for Availabi...	Open	Closed
221	インシデント	KPI Availability ...	SLO for Availabi...	Open	Closed
222	インシデント	KPI Performance...	SLO for Perform...	Open	Closed
223	インシデント	KPI Performance...	SLO for Perform...	Open	Closed
224	インシデント	KPI Performance...	SLO for Perform...	Open	Closed
225	インシデント	KPI Performance...	SLO for Perform...	Open	Closed

ここでは、設定済みの SLO を追加ウィザードを呼び出す新しい SLA の作成ウィザードが必要で、それによりユーザは新しい SLA の作成から、それに関連する SLO の追加にシームレスに移動できます。

SLA ウィザードの設計

新しいウィザードを設計するにあたって、エンドユーザが必要な作業、および実行する必要があるバックグラウンドプロセスの両方を含む、全体のプロセスの流れを考慮することが役に立ちます。

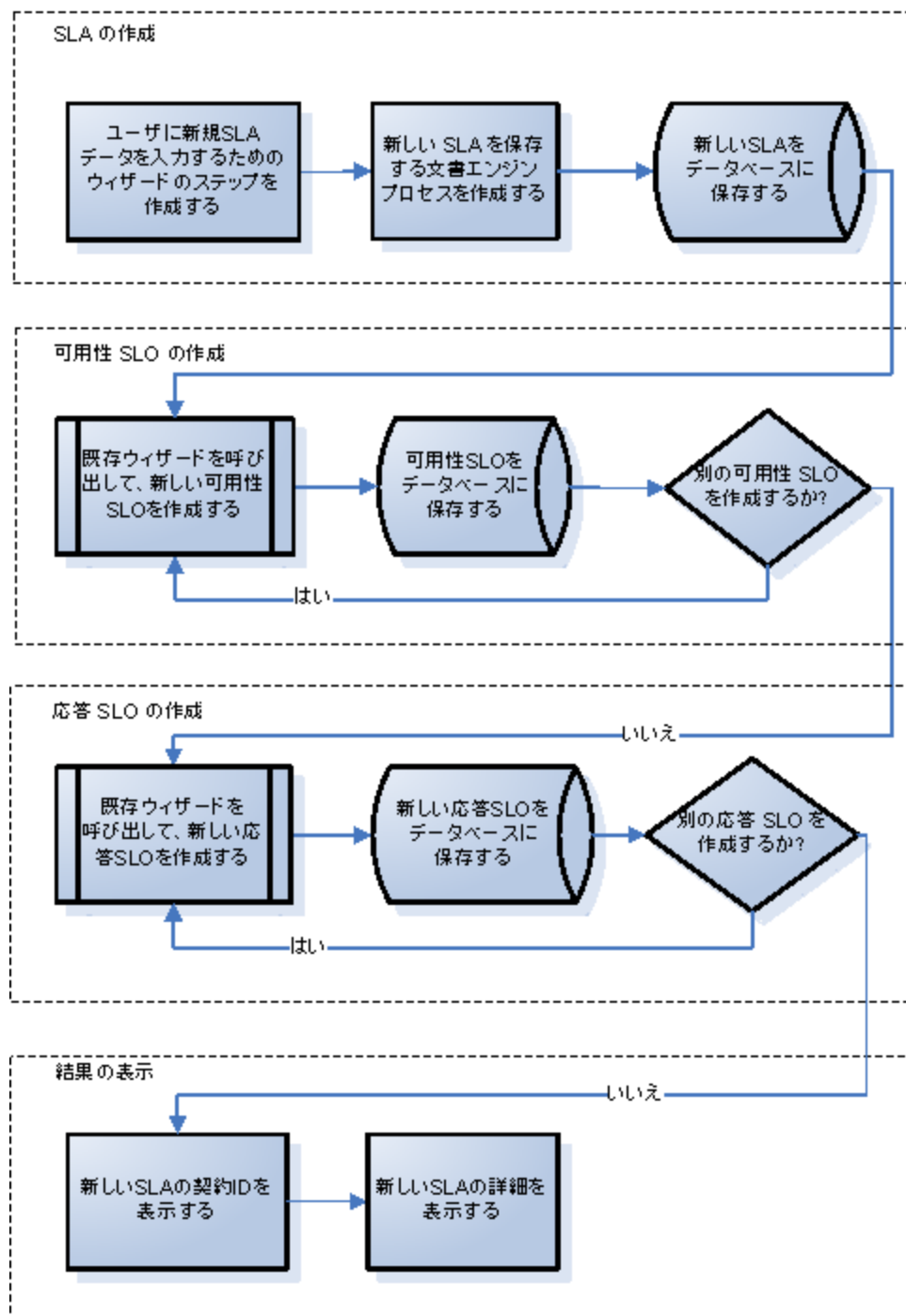
この場合、ユーザに表示して新しい SLA レコードに必要な基本情報の入力を求める、一連の連続したウィザードの手順またはパネルを作成する必要があります。必要な情報が集まったら、文書エンジンのプロセスによって新しい SLA レコードをデータベースに保存する必要があります。

ウィザードは次に、ユーザが可用性 SLO または応答 SLO を追加するかどうかを尋ねる必要があります。追加をする場合、設定済みの SLO ウィザードを呼び出して最初に可用性 SLO を、次に応答 SLO を追加します。

最後に、ウィザードはレコード番号と新しく作成した SLA のタイトルを表示し、そしてレコードの詳細を表示するボタンを提供する必要があります。

新しい SLA の作成ウィザードの全体のデザインを次のフローチャートに示します。

SLAウィザード設計



SLA ウィザードパネルの計画

全体的な観点からのウィザードの設計に加え、各ウィザードパネルの機能を計画することが役に立ちます。ウィザードパネルの一般的な目的は、SLA レコードの必須フィールドの入力をユーザに求めることです。

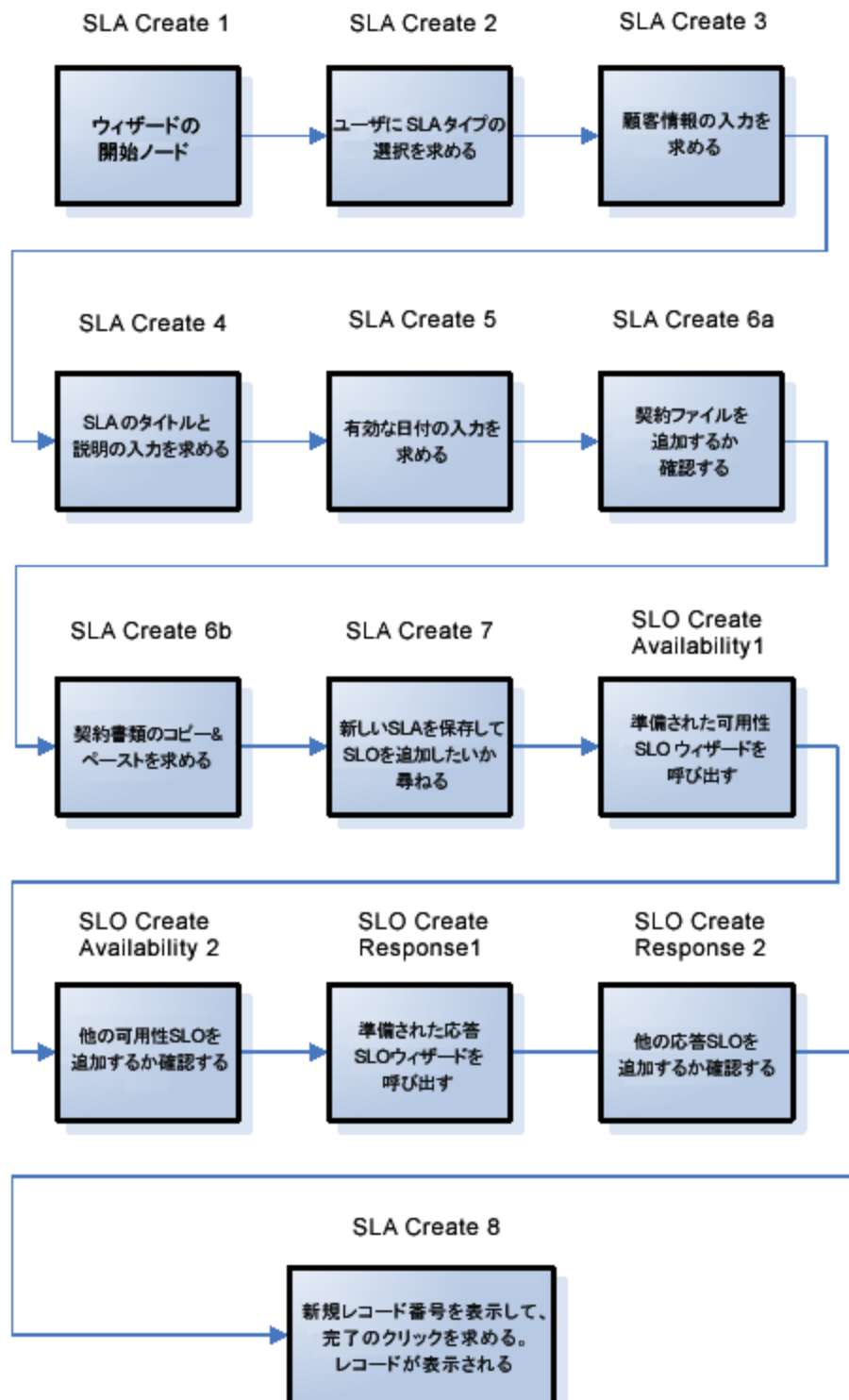
- SLA タイプ (サービスまたは顧客)
- 顧客
- タイトルと説明
- SLA 有効期間

さらに、これは必須ではありませんが、契約文書から SLA レコードにコピー& ペーストを実行したいというニーズがユーザの間にあります。そこで、ウィザードでパネルを表示し、このような情報の入力をユーザに求めることにします。

各ウィザードパネルの名前に連番を付ける (SLA Create 1 や SLA Create 2 など) ことで、これらの個々のウィザードパネルが、エンドユーザに表示される SLA 作成ウィザードの構成要素であることを明確にします。

以下のフローチャートでは、計画されたウィザードパネルと、それぞれが実行する機能を示します。フローチャートの各ボックスはまた、SLA ウィザードの構築で説明した、SLA ウィザードを構築するのに必要な手順に対応しています。

SLA ウィザードパネルの名前と機能



SLA ウィザードの構築

この例では、ウィザードツールを使用して、SLA の作成ウィザードの各パネルを設計します。

ほとんどのパネルは、ユーザに対して情報の入力を求める機能があり、したがって、フォームデザイナーを使用して、ウィザードパネル内に表示されるサブフォーマット用のフォームコントロールを設計することも必要です。

以下に示す各概要の手順は、前のセクションのフローチャートにあるボックスに対応しています。これらの手順を、ウィザードツールおよびフォームデザイナーを使用して行う必要のある特定のタスクと共に、以下に一覧表示します。

[「ステップ 1: ウィザードの開始ノードを定義する」\(38 ページ\)](#)

[「ステップ 2: ユーザに SLA タイプの選択を求める」\(40 ページ\)](#)

[「ステップ 3: 顧客情報の入力を求める」\(41 ページ\)](#)

[「ステップ 4: SLA のタイトルと説明の入力を求める」\(46 ページ\)](#)

[「ステップ 5: 有効期間の入力を求める」\(49 ページ\)](#)

[「ステップ 6: 契約文書について質問する」\(53 ページ\)](#)

[「ステップ 7: 契約の追加を求める」\(57 ページ\)](#)

[「ステップ 8: 新しい SLA を保存して SLO の追加を求める」\(60 ページ\)](#)

[「ステップ 9: 可用性 SLO ウィザードを呼び出す」\(66 ページ\)](#)

[「ステップ 10: 可用性 SLO の追加を求める」\(68 ページ\)](#)

[「ステップ 11: 応答 SLO ウィザードを呼び出す」\(71 ページ\)](#)

[「ステップ 12: 応答 SLO の追加を求める」\(73 ページ\)](#)

[「ステップ 13: 新しい SLA 番号とタイトルの表示」\(77 ページ\)](#)

[「ステップ 14: 新規 SLA ウィザードを呼び出すメニューの設定」\(81 ページ\)](#)

ステップ 1: ウィザードの開始ノードを定義する

この最初のタスクでは、新規 SLA の作成ウィザードの実行時に表示される、最初のウィザードパネルを作成します。ウィザードツールを使用して、最初のウィザードパネルを定義します。

SLA タイプの入力を求めるウィザードを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「SLA Create 1」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このウィザードが一連のウィザードパネルの中で最初なので、このチェックボックスを [true] に設定します。
概要	次のように入力します： このウィザードは SLA レコードを作成し、関連する SLO の作成方法をユーザに案内します。
ウィンドウタイトル	次のように入力します： 新規 SLA の作成
タイトル	次のように入力します： 新規 SLA の作成

5. [ファイル選択]タブを選択し、[**\$L.file の選択基準**]サブタブを選択して、次のフィールドに入力します。

フィールド	値
レコードの作成	このオプションを選択すると、ウィザードはレコードを作成します。レコードタイプを[対象タイプ]フィールドに入力または選択する必要があります。これが \$L.file になります。
対象タイプ (レコードの作成用)	[フィル]をクリックして、作成されるレコードのタイプを選択し、[sla]を選択します。

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
ユーザ入力の要求	このフィールドを選択すると、ユーザに情報の入力を求めます。
表示するサブフォーマット	表示するフォーマットのサブフォーマット名を入力します。次のように入力します： sla.create.type

7. [次のウィザード]タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します： SLACreate 2

フィールド	値
条件	次に何が起こるかを決定する条件です。この場合、順序通り次のウィザードを入力できます。そして評価する式は "true" です。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します： true

8. 完了したら、[保存]をクリックします。

最初のウィザードパネルの作成が完了しました。次に、パネルの外観や機能を改良するため、フォームデザイナーを使用してこのウィザードパネルに表示されるサブフォーマットを作成します。

ステップ 2: ユーザに SLA タイプの選択を求める

このステップでは、最初のウィザードパネルに手を加えることにより、ユーザに SLA タイプ(サービスまたは顧客)の選択を求めます。

フォームデザイナーを使用して、ウィザードパネルに表示されるサブフォーマットを作成します。

新規SLAの作成

SLAタイプを選択します。選択可能なタイプは、サービスと顧客です。サービスSLAはサービスに関連しており、顧客SLAは特定の契約に関連しています。

SLAタイプ

< 前へ 次へ > 終了 キャンセル

このタスクでは、フォームデザイナーを使用して、ユーザに SLA タイプの選択を求めるフォームコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、上記の画面がエンドユーザに表示されます。

SLA タイプウィザード用の新しいフォームを作成するには：

1. フォームデザイナーを起動します。
2. [フォーム]フィールドに「sla.create.type」と入力します。
ベストプラクティス: フォーム名はサブフォーマットの表示エントリと正確に一致する必要があるので、[使用方法]タブの表示するサブフォーマットをコピーして、[フォーム]フィールドに貼り付けることができます。これにより、スペルの誤りを回避できます。
3. [新規]をクリックします。
4. フォームウィザードの使用については[いいえ]をクリックします。

5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標が 0 (ゼロ) であることを確認します。

プロパティ	値
ラップラベル	
Caption	次のように入力します: SLA タイプ を選択します。選択可能なタイプは、サービスと顧客です。サービス SLA はサービス CI に関連しており、顧客 SLA は特定の契約に関連しています。
Visible	選択します。
ラベル	
Caption	次のように入力します: SLA タイプ :
Visible	選択します。
コンボボックス	コンボボックスは、通常のラベルボックスの右側に配置し、ラベルに関連した選択の一覧を含めます。
Display List	2 つのエントリを入力します: <ul style="list-style-type: none"> ■ サービス ■ 顧客
Input	次のように入力します: type
Mandatory	選択します。
Select Only	選択します。
Value List	2 つのエントリを入力します: <ul style="list-style-type: none"> ■ service ■ customer
Visible	選択します。

6. [OK]を2回クリックして、新規フォームを保存し終了します。

これで、新規 SLA の作成ウィザードで最初のパネルに使用するフォームの作成が完了しました。

ステップ 3: 顧客情報の入力を求める

このステップでは、顧客、サービス契約、サービス時間の入力をユーザに求めるパネルを作成します。

注: 開始地点として、作成したばかりのレコード SLA Create 1 を使用し、以下の情報を反映するようにフィールドを修正します。修正を開始した後は、忘れずに[追加]ボタンをクリックして、SLA

Create 1 レコードを上書きしないようにすることが非常に重要です。

以下の 2 つのタスクを実行して、顧客情報を収集するウィザードパネルを設計します。

顧客情報を収集するウィザードの作成

顧客情報を収集するフォームの作成

タスク 1: 顧客情報を収集するウィザードの作成

このタスクでは、ユーザに表示される次のウィザードパネルを作成します。このウィザードパネルは、顧客、サービス契約、およびサービス時間の入力をユーザに求めます。

顧客情報を収集するウィザードを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「SLA Create 2」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードパネルは、顧客名、サービス契約、およびサービス時間の入力を求めます。
ウィンドウタイトル	次のように入力します: 新規 SLA の作成 - 顧客情報
タイトル	次のように入力します: 新規 SLA の作成 - 顧客情報

5. [ファイル選択]タブを選択し、[\$L.file の選択基準]サブタブを選択して、次のフィールドに入力します。

フィールド	値
渡された \$L.file	このオプションを選択すると、\$L.file 変数が前のウィザードから渡される必要があります。

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

7. [アクション]タブの[フォーマットコントロール / プロセス名]サブタブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行されるアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)]を選択します。

8. [次のウィザード]タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します： SLACreate 3
条件	次に何が起こるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します： true

9. 完了したら、[保存]をクリックします。

顧客情報のウィザードパネルを作成しました。次は、新しいウィザードフォームをフォームデザイナーで作成します。

タスク 2: 顧客情報を収集するフォームの作成

このタスクでは、フォームデザイナーを使用して、ユーザに基本的な顧客情報の入力を求めるコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、次の画面がエンドユーザに表示されます。

新規SLAの作成 - 顧客情報

顧客の選択:

可能な場合は、サービス契約を選択します:

可能な場合は、サービス時間を選択します:

< 前へ 次へ > 終了 キャンセル

顧客情報を収集するための新しい SLA フォームを作成するには:

1. フォームデザイナーを起動します。
2. [フォーム]フィールドに「**sla.create.customer**」と入力します。
ベストプラクティス: フォーム名はサブフォーマットの表示エントリと正確に一致する必要がありますので、[使用方法]タブの表示するサブフォーマットをコピーして、[フォーム]フィールドに貼り付けることができます。これにより、スペルの誤りを回避できます。
3. [新規]をクリックします。
4. フォームウィザードの使用については[いいえ]をクリックします。
5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標が 0 (ゼロ) であることを確認します。

プロパティ	値
ラベル	
Caption	次のように入力します: 顧客の選択:
Visible	選択します。

プロパティ	値
コンボフィル	
Combo Button Visible	選択します。
Input	customer
Mandatory	選択します。
Select Only	選択します。
Third Button Visible	選択します。
Value List	次のように入力します: \$lo.colist
Visible	選択します。
ラベル	
Caption	次のように入力します: 可能な場合は、サービス契約を選択します:
Visible	選択します。
コンボフィル	
Combo Button Visible	選択します。
Fill Button Visible	選択します。
Input	次のように入力します: service.contract
Select Only	選択します。
Third Button Visible	選択します。
Value List Condition	次のように入力します: select("contract.id","servicecontract","provider",[customer])
Visible	選択します。
ラベル	
Caption	次のように入力します: 可能な場合は、サービス時間を選択します:
Visible	選択します。
コンボフィル	

プロパティ	値
Combo Button Visible	選択します。
Fill Button Visible	選択します。
Input	次のように入力します: service.hours
Select Only	選択します。
Third Button Visible	選択します。
Value List	次のように入力します: \$G.calendars
Visible	選択します。
Display List Condition	select("name","servicecontract","provider",[customer])

6. [OK]を2回クリックして、新規フォームを保存し終了します。

SLA ウィザードに必要なフォームの1つの作成を完了しました。

ステップ 4: SLA のタイトルと説明の入力を求める

ステップ 3 では、ユーザに SLA のタイトルと説明の入力を求める、次のウィザードの手順を作成します。

以下の2つのタスクを実行して、SLA のタイトルと説明の入力をユーザに求めるウィザードパネルを設計します。

[SLA の説明用ウィザードの作成](#)

[SLA の説明用フォームの作成](#)

タスク 1: SLA の説明用ウィザードの作成

このタスクでは、ユーザに新しい SLA のタイトルと説明の入力を求めるウィザードパネルを作成します。

SLA のタイトルと説明用のウィザードパネルを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「**SLA Create 3**」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードパネルは、ユーザに SLA のタイトルと説明の入力を求めます。
ウィンドウタイトル	次のように入力します: 新規 SLA の作成 - 説明
タイトル	次のように入力します: 新規 SLA の作成 - 説明

5. [ファイル選択]タブを選択し、[\$L.file の選択基準]サブタブを選択して、次のフィールドに入力します。

フィールド	値
渡された \$L.file	このオプションを選択すると、\$L.file 変数が前のウィザードから渡される必要があります。

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
ユーザ入力の要求	このフィールドを選択すると、ユーザに情報の入力を求めます。
表示するサブフォーマット	表示するフォーマットのサブフォーマット名を入力します。次のように入力します: sla.create.description

7. [アクション] タブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)]を選択します。

8. [次のウィザード]タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します: SLACreate 4
条件	次に何が起こるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します: true

9. 完了したら、[保存]をクリックします。

新しいウィザードパネルを作成しました。次は、新しいウィザードフォームをフォームデザイナーで作成します。

タスク 2: SLA の説明用フォームの作成

このタスクでは、フォームデザイナーを使用して、ユーザにタイトルと説明の入力を求めるコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、次の画面がエンドユーザに表示されます。

新規SLAの作成 - 説明

タイトルを入力します。

詳細な説明を入力します。

◀ 前へ

次へ >

終了

キャンセル

SLA の説明用ウィザードを表示する新しいフォームを作成するには:

1. フォームデザイナーを起動します。
2. [フォーム]フィールドに「**sla.create.description**」と入力します。

ベストプラクティス: フォーム名はサブフォーマットの表示エントリと正確に一致する必要がありますので、[使用方法]タブの表示するサブフォーマットをコピーして、[フォーム]フィールドに貼り付けることができます。これにより、スペルの誤りを回避できます。

3. [新規]をクリックします。
4. フォームウィザードの使用については[いいえ]をクリックします。
5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標が 0 (ゼロ) であることを確認します。

プロパティ	値
ラベル	
Caption	次のように入力します: タイトルを入力します。
Visible	選択します。
テキスト	
Input	次のように入力します: title
Visible	選択します。
ラベル	
Caption	次のように入力します: 詳細な説明を入力します。
Visible	選択します。
テキスト領域	
Input	次のように入力します: description
Visible	選択します。

6. [OK]を2回クリックして、新規フォームを保存し終了します。

SLA ウィザードに必要なフォームの1つを作成しました。

ステップ 5: 有効期間の入力を求める

このステップでは、ユーザに SLA の開始日と終了日の入力を求める、ウィザードの手順を作成します。

以下の2つのタスクを実行して、SLA の開始日と終了日の入力をユーザに求めるウィザードパネルを設計します。

[SLA 有効期間のウィザードの作成](#)

[SLA 有効期間フォームの作成](#)

タスク 1: SLA 有効期間のウィザードの作成

このタスクでは、ユーザに有効な開始日と終了日の入力を求めるウィザードを作成します。

有効な開始日と終了日を入力するウィザードを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「SLA Create 4」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードパネルは、ユーザに SLA の開始日と終了日の入力を求めます。
ウィンドウタイトル	次のように入力します: 新規 SLA の作成 - 有効期間
タイトル	次のように入力します: 新規 SLA の作成 - 有効期間

5. [ファイル選択]タブを選択し、[\$L.file の選択基準]サブタブを選択して、次のフィールドに入力します。

フィールド	値
渡された \$L.file	このオプションを選択すると、\$L.file 変数が前のウィザードから渡される必要があります。

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
ユーザ入力の要求	このフィールドを選択すると、ユーザに情報の入力を求めます。
表示するサブフォーマット	表示するフォーマットのサブフォーマット名を入力します。次のように入力します: sla.create.validfromto

7. [アクション]タブの[フォーマットコントロール/ プロセス名]サブタブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)]を選択します。

8. [次のウィザード]タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します： SLACreate 5
条件	次に何が起こるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します： true

9. 完了したら、[保存]をクリックします。

有効な開始日および終了日のウィザードパネルを作成しました。次は、新しいウィザードフォームをフォームデザイナーで作成します。

タスク 2: SLA の有効期間用フォームの作成

このタスクでは、フォームデザイナーを使用して、ユーザーに SLA の有効期間指定を求めるフォームコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、次の画面がエンドユーザーに表示されます。

新規SLAの作成 - 有効期間

SLAの発効日:

SLAの失効日:

< 前へ 次へ > 終了 キャンセル

SLA の有効期間 ウィザード用の新しいフォームを作成するには:

1. フォームデザイナーを起動します。
2. [フォーム]フィールドに「**sla.create.validfromto**」と入力します。
ベストプラクティス: フォーム名はサブフォーマットの表示エントリと正確に一致する必要があるの
で、[使用方法]タブの表示するサブフォーマットをコピーして、[フォーム]フィールドに貼り付けるこ
とができます。これにより、スペルの誤りを回避できます。
3. **[新規]**をクリックします。
4. フォームウィザードの使用については**[いいえ]**をクリックします。
5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標
が 0 (ゼロ)であることを確認します。

プロパティ	値
ラベル	
Caption	次のように入力します: SLA の発効日:
Visible	選択します。
コンボフィル	

プロパティ	値
Fill Button Visible	選択します。
Input	start
Mandatory	選択します。
Visible	選択します。
ラベル	
Caption	次のように入力します: SLA の失効日:
Visible	選択します。
コンボフィル	
Fill Button Visible	選択します。
Input	次のように入力します: expiration
Mandatory	選択します。
Visible	選択します。

6. [OK]を2回クリックして、新規フォームを保存し終了します。

SLA ウィザードに必要なフォームの1つの作成を終了しました。

ステップ 6: 契約文書について質問する

このステップでは、契約文書からテキストをコピーするかどうかを尋ねるウィザードパネルを作成します。

以下の2つのタスクを実行して、SLAに関連した契約およびその他の文書についてユーザに尋ねるウィザードパネルを設計します。

[契約について質問するウィザードの作成](#)

[契約について質問するフォームの作成](#)

タスク 1: 契約について質問するウィザードの作成

このタスクでは、契約文書からテキストをこの SLA にコピーするかどうかを尋ねるウィザードパネルを作成します。

パネルを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「SLA Create 5」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードは、契約文書からテキストを SLA にコピーするか、または SLA にその他の文書を添付するかをユーザーに尋ねます。
ウィンドウタイトル	次のように入力します: 新規 SLA の作成 - 契約文書のコピー
タイトル	次のように入力します: 新規 SLA の作成 - 契約文書のコピー

5. [ファイル選択]タブを選択し、次のフィールドに入力します。

フィールド / サブタブ	値
[初期式]サブタブ	次のように入力します: \$L.agreement.docs=false
[\$L.file の選択基準]サブタブ	[渡された \$L.file]オプションを選択すると、ウィザードに対して \$L.file 変数が前のウィザードから渡される必要があることを示します。

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
ユーザ入力の要求	このフィールドを選択すると、ユーザーに情報の入力を求めます。
表示するサブフォーマット	表示するフォーマットのサブフォーマット名を入力します。次のように入力します: sla.create.ask.agreements

7. [アクション] タブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)]を選択します。

8. [次のウィザード]タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します： SLA Create 6a
条件	次に何が起こるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します： \$L.agreement.docs=true
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します： SLA Create 7
条件	次のように入力します： true

9. 完了したら、[保存]をクリックします。

添付ファイルについてユーザに尋ねるウィザードパネルの作成が完了しました。次は、フォームデザイナーで対応する新しいウィザードフォームを作成します。

タスク 2: 契約を追加するためのフォームの作成

このタスクでは、フォームデザイナーを使用して、SLA に関連した契約文書からテキストのコピーと貼り付けをユーザに求めるフォームコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、次の画面がエンドユーザに表示されます。

新規SLAの作成 - 契約文書

このSLAに関連する契約文書からテキストをコピーして貼り付けます:

< 前へ 次へ > 終了 キャンセル

契約文書のテキストの追加をユーザに求めるウィザードを表示する新しいフォームを作成するには:

1. フォームデザイナーを起動します。
2. [フォーム]フィールドに「**sla.create.add.agreements**」と入力します。
ベストプラクティス: フォーム名はサブフォーマットの表示エントリと正確に一致する必要があるの
で、[使用方法]タブの表示するサブフォーマットをコピーして、[フォーム]フィールドに貼り付けるこ
とができます。これにより、スペルの誤りを回避できます。
3. [新規]をクリックします。
4. フォームウィザードの使用については[いいえ]をクリックします。
5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標
が 0 (ゼロ) であることを確認します。

プロパティ	値
ラベル	
Caption	次のように入力します: この SLA に関連する契約文書からテキストをコピーして貼り付けます:
Visible	選択します。
テキスト領域	

プロパティ	値
Input	次のように入力します: agreements
Visible	選択します。

6. [OK]を2回クリックして、新規フォームを保存し終了します。

SLA ウィザードに必要なフォームの1つの作成を終了しました。

ステップ 7: 契約の追加を求める

このステップでは、ユーザに SLA に関連した契約の追加を求めるウィザードパネルを作成します(前のパネルで[はい]と答えた場合)。ユーザはエントリフィールドにテキストをコピーして貼り付けることで契約を追加します。

以下の2つのタスクを実行して、ユーザに契約の追加を求めるウィザードパネルを設計します。

[契約を追加するためのウィザードの作成](#)

[契約を追加するためのフォームの作成](#)

タスク 1: 契約を追加するためのウィザードの作成

このタスクでは、ユーザに SLA に関連した契約からテキストのコピーと貼り付けを行うよう求めるウィザードパネルを作成します(前のパネルで[はい]と答えた場合)。

契約文書を添付するようユーザに求めるウィザードパネルを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「SLA Create 6a」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードは、契約文書からテキストをコピーして貼り付けするようユーザに求めます。
ウィンドウタイトル	次のように入力します:

フィールド	値
	新規 SLA の作成 - 契約文書
タイトル	次のように入力します: 新規 SLA の作成 - 契約文書

5. [ファイル選択]タブを選択し、[\$L.file の選択基準]サブタブを選択して、次のフィールドに入力します。

フィールド	値
渡された \$L.file	このオプションを選択すると、\$L.file 変数が前のウィザードから渡される必要があります。

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
ユーザ入力の要求	このフィールドを選択すると、ユーザに情報の入力を求めます。
表示するサブフォーマット	表示するフォーマットのサブフォーマット名を入力します。次のように入力します: sla.create.add.agreements

7. [アクション]タブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)]を選択します。

8. [次のウィザード]タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します: SLA Create 7
条件	次に何が起こるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを

フィールド	値
	指定できます。次のように入力します: true

9. 完了したら、[保存]をクリックします。

契約文書を添付するようユーザーに求めるウィザードパネルを作成しました。次は、新しいウィザードフォームをフォームデザイナーで作成します。

タスク 2: 契約を追加するためのフォームの作成

このタスクでは、フォームデザイナーを使用して、SLAに関連した契約文書からテキストのコピーと貼り付けをユーザーに求めるフォームコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、次の画面がエンドユーザーに表示されます。

契約文書のテキストの追加をユーザーに求めるウィザードを表示する新しいフォームを作成するには:

1. フォームデザイナーを起動します。
2. [フォーム]フィールドに「**sla.create.add.agreements**」と入力します。
ベストプラクティス: フォーム名はサブフォーマットの表示エントリと正確に一致する必要があるの
で、[使用方法]タブの表示するサブフォーマットをコピーして、[フォーム]フィールドに貼り付けるこ
とができます。これにより、スペルの誤りを回避できます。
3. [新規]をクリックします。
4. フォームウィザードの使用については[いいえ]をクリックします。
5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標
が 0 (ゼロ) であることを確認します。

プロパティ	値
ラベル	
Caption	次のように入力します： この SLA に関連する契約文書からテキストをコピーして貼り付けます：
Visible	選択します。
テキスト領域	
Input	次のように入力します： agreements
Visible	選択します。

6. [OK]を2回クリックして、新規フォームを保存し終了します。

SLA ウィザードに必要なフォームの1つの作成を終了しました。

ステップ 8: 新しい SLA を保存して SLO の追加を求める

この時点で、新しい SLA に必要なすべての情報が集まりました。次に実行するタスクは、文書エンジンでプロセスを作成し、新しい SLA をデータベースに保存することです。

さらに、SLA に SLO を追加するかどうかを尋ねます。

以下の3つのタスクを実行してこのステップを完了します。

[新しい SLA を保存するプロセスの作成](#)

[SLO の要件を確認するためのウィザードの作成](#)

[SLO の要件を確認するためのフォームの作成](#)

タスク 1: 新しい SLA を保存するプロセスの作成

このタスクでは、いま作成した新しい SLA レコードを保存するプロセスを作成します。

新しい SLA レコードを保存するプロセスを作成するには：

1. [カスタマイズ]>[文書エンジン]>[プロセス]をクリックします。[プロセス定義]フォームがオープンします。
2. [プロセス名]フィールドに、「sla.save.from.wizard」と入力します。
3. [RAD]タブで、[RAD 呼び出しの前に評価される式]セクションの最初の2行に次の式を入力します。

`$L.action="add"`

4. [RAD] タブの[RAD アプリケーション]セクションに、呼び出される RAD アプリケーションとして「se.view.engine」と入力し、それを実行する条件として「true」と入力します。次に、以下の

パラメータ名と値を入力します:

パラメータ名	パラメータ値
file	次のように入力します: \$L.file
description	\$L.action

5. [RAD]タブの[ポスト RAD 式]セクションに、次のように入力します。**\$L.new.sla=agreement.id in \$L.file**
6. [追加]をクリックします。
7. 完了したら、[OK]をクリックします。

新しい SLA レコードをデータベースに保存するプロセスを作成しました。

タスク 2: SLO の要件を確認するためのウィザードの作成

このタスクでは、保存プロセスを呼び出し、次に SLO を追加するかどうかをユーザーに尋ねるサブフォーマットを作成し、そして SLO の要件を確認するウィザードを作成します。

保存プロセスを呼び出して SLO の要件を確認するウィザードを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「**SLA Create 7**」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードパネルでは、ユーザーが可用性 SLO または応答 SLO を作成するかどうかを尋ねます。
ウィンドウタイトル	次のように入力します: 新規 SLA の作成 - SLO の要件の確認
タイトル	次のように入力します: 新規 SLA の作成 - SLO の要件の確認

5. [ファイル選択]タブを選択し、次のフィールドに入力します。

フィールド / サブタブ	値
[初期式]サブタブ	次のように入力します: <code>\$L.create.response.slo=false</code> <code>\$L.create.avail.slo=false</code>
[\$L.file の選択基準]サブタブ	[渡された \$L.file]オプションを選択すると、ウィザードに対して \$L.file 変数が前のウィザードから渡される必要があることを示します。

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
ユーザ入力の要求	このフィールドを選択すると、ユーザに情報の入力を求めます。
表示するサブフォーマット	表示するフォーマットのサブフォーマット名を入力します。次のように入力します: <code>sla.create.ask.slos</code>

7. [アクション] タブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)]を選択します。
[フォーマットコントロール/プロセス名]サブタブ	ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。
プロセス名	このウィザードが完了後にプロセスを実行できることを指定するため、以下のように入力します。 <code>sla.save.from.wizard</code>
ウィザード変数	<code>\$L.create.response.slo</code> <code>\$L.crate.avail.slo</code> <code>\$L.new.sla</code>

8. [次のウィザード]タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します: SLA Create Response 1
条件	次に何が起こるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します: \$L.create.response.slo=true
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します: SLA Create Availability 1
条件	次のように入力します: \$L.create.avail.slo=true
ウィザード名	次のように入力します: SLA Create 8
条件	次のように入力します: true

9. 完了したら、[保存]をクリックします。

保存プロセスとSLOの要件を確認するウィザードパネルを作成しました。次は、新しいウィザードフォームをフォームデザイナーで作成します。

タスク 3: SLO の要件を確認するためのフォームの作成

このタスクでは、フォームデザイナーを使用して、SLAに関連したサービスレベル目標を作成するかどうかをユーザに質問する、フォームコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、次の画面がエンドユーザに表示されます。

新規SLAの作成 - SLOの要件の確認

ヘルプデスクの応答時間に基づいてサービスレベル目標を作成しますか？

☐ はい ☒ いいえ

サーバーの可用性に基づいてサービスレベル目標を作成しますか？

☐ はい ☒ いいえ

< 前へ 次へ > 終了 キャンセル

保存プロセスを呼び出して SLO の要件を確認するウィザード用の新しいフォームを作成するには：

1. フォームデザイナーを起動します。
2. [フォーム]フィールドに「**sla.create.ask.slos**」と入力します。
ベストプラクティス：フォーム名はサブフォーマットの表示エントリと正確に一致する必要がありますので、[使用方法]タブの表示するサブフォーマットをコピーして、[フォーム]フィールドに貼り付けることができます。これにより、スペルの誤りを回避できます。
3. [新規]をクリックします。
4. フォームウィザードの使用については[いいえ]をクリックします。
5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標が 0 (ゼロ) であることを確認します。

プロパティ	値
グループ	
Caption	次のように入力します： ヘルプデスクの応答時間に基づいてサービスレベル目標を作成しますか？
Visible	選択します。
ラジオボタン	
Caption	次のように入力します： はい
Input	次のように入力します： \$L.create.response.slo

プロパティ	値
値	次のように入力します: true
Visible	選択します。
ラジオボタン	
Caption	次のように入力します: いいえ
Input	次のように入力します: \$L.create.response.slo
値	次のように入力します: false
Visible	選択します。
グループ	
Caption	次のように入力します: サーバーの可用性に基づいてサービスレベル目標を作成しますか?
Visible	選択します。
ラジオボタン	
Caption	次のように入力します: はい
Input	次のように入力します: \$L.create.avail.slo
値	次のように入力します: true
Visible	選択します。
ラジオボタン	
Caption	次のように入力します: いいえ
Input	次のように入力します: \$L.create.avail.slo
値	次のように入力します:

プロパティ	値
	false
Visible	選択します。

6. [OK]を2回クリックして、新規フォームを保存し終了します。

SLA ウィザードに必要なフォームの1つの作成を終了しました。

ステップ 9: 可用性 SLO ウィザードを呼び出す

次のステップでは、可用性 SLO が必要だと答えたユーザのために、既存の設定済みの可用性ウィザードを呼び出すウィザードパネルを作成します。

このウィザードパネルは、別のウィザードを呼び出すバックグラウンドタスクを実行するだけなので、エンドユーザに表示する画面はありません。したがって、このステップではフォームデザイナーの使用を必要とするタスクはありません。

タスク: SLO を呼び出すウィザードの作成

このタスクでは、既存の設定済みの可用性 SLO ウィザードを呼び出すウィザードを作成します。

既存の設定済みの可用性 SLO ウィザードを呼び出すウィザードを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「SLA Create Availability 1」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードは既存のウィザードを呼び出して、可用性 SLO を作成します。
ウィンドウタイトル	次のように入力します: 可用性 SLO の作成
タイトル	次のように入力します: 可用性 SLO の作成

5. [ファイル選択]タブを選択し、[\$L.file の選択基準]サブタブを選択して、次のフィールドに入力します。

フィールド	値
渡された \$L.file	オン

6. **【使用方法】**タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
表示をスキップ	このパネルにはユーザに表示するフォームが含まれていないため、このオプションを選択します。

7. **【アクション】**タブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)] を選択します。
[フォーマットコントロール/プロセス名]サブタブ	ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。
プロセス名	このウィザードが完了後にプロセスを実行できることを指定するため、以下のように入力します。 slo.add.avail.sla

8. **【次のウィザード】**タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します： SLA Create Availability 2
条件	次に何が起きるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します： true

9. 完了したら、**【保存】**をクリックします。

既存の設定済みの可用性 SLO ウィザードを呼び出すウィザードパネルを作成しました。このウィザードには新しいフォームは必要ありません。

ステップ 10: 可用性 SLO の追加を求める

次のステップでは、ユーザが別の可用性 SLO を追加するかどうかを尋ねます。追加する場合、このパネルは設定済みの可用性 SLO ウィザードを再度呼び出す必要があります。可用性 SLO を追加しない場合、ユーザが応答 SLO を追加することを示した場合は、設定済みの応答 SLO を呼び出し、それ以外の場合は最後のウィザード画面に移動する必要があります。

以下の 2 つのタスクを実行して、別の可用性 SLO を求めるウィザードパネルを設計します。

[追加の可用性 SLO のウィザードの作成](#)

[追加の可用性 SLO のフォームの作成](#)

タスク 1: 追加の可用性 SLO のウィザードの作成

このタスクでは、ウィザードを作成してユーザが追加の可用性 SLO を作成できるようにします。

ウィザードを作成して、ユーザが追加の可用性 SLO を作成できるようにするには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「SLA Create Availability 2」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードパネルでは、ユーザが別の可用性 SLO を作成するかどうかを尋ねます。
ウィンドウタイトル	次のように入力します: 別の可用性 SLO を作成する
タイトル	次のように入力します: 別の可用性 SLO を作成する

5. [ファイル選択]タブを選択し、[\$L.file の選択基準]サブタブを選択して、次のフィールドに入力します。

フィールド	値
初期式	\$L.create.avail.slo=false

フィールド	値
\$L.file なし (typecheck を使用)	このオプションを選択すると、保持ファイルとして動作する typecheck ファイルを初期化します。

6. **【使用方法】**タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
ユーザ入力の要求	このフィールドを選択すると、ユーザに情報の入力を求めます。
表示するサブフォーマット	表示するフォーマットのサブフォーマット名を入力します。次のように入力します: sla.create.ask.other.avail
[前へ] ボタンだけをオフにする	オン

7. **【アクション】** タブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)] を選択します。
ウィザード変数	\$L.create.avail.slo

8. **【次のウィザード】**タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します: SLA Create Availability 1
条件	次に何が起こるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します: \$L.create.avail.slo=true
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します:

フィールド	値
	SLA Create 8
条件	true

9. 完了したら、[保存]をクリックします。

ユーザが追加の可用性 SLO を作成できるようなウィザードを作成しました。次は、新しいウィザードフォームをフォームデザイナーで作成します。

タスク 2: 追加の可用性 SLO のフォームの作成

このタスクでは、フォームデザイナーを使用して、ユーザが追加のサービスレベル目標レコードを作成するかどうかを質問する、フォームコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、次の画面がエンドユーザに表示されます。

別の可用性SLOを作成する

別の可用性SLOレコードを作成しますか？

☒ はい ☐ いいえ

< 前へ

次へ >

終了

キャンセル

追加の可用性 SLO を作成するウィザードに新しいフォームを作成するには：

1. フォームデザイナーを起動します。
2. [フォーム]フィールドに「sla.create.ask.other.avail」と入力します。
ベストプラクティス: フォーム名はサブフォーマットの表示エントリと正確に一致する必要があるの
で、[使用方法]タブの表示するサブフォーマットをコピーして、[フォーム]フィールドに貼り付けるこ
とができます。これにより、スペルの誤りを回避できます。
3. [新規]をクリックします。
4. フォームウィザードの使用については[いいえ]をクリックします。
5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標
が 0 (ゼロ) であることを確認します。

プロパティ	値
グループ	
Caption	次のように入力します: 別の可用性 SLO レコードを作成しますか?
Visible	選択します。
ラジオボタン	
Caption	次のように入力します: はい
Input	次のように入力します: \$L.create.avail.slo
値	次のように入力します: true
Visible	選択します。
ラジオボタン	
Caption	次のように入力します: いいえ
Input	次のように入力します: \$L.create.avail.slo
値	次のように入力します: false
Visible	選択します。

6. [OK]を2回クリックして、新規フォームを保存し終了します。

SLA ウィザードに必要なフォームの1つの作成を終了しました。

ステップ 11: 応答 SLO ウィザードを呼び出す

次のステップでは、応答 SLO を追加すると答えたユーザーのために、既存の設定済みの応答ウィザードを呼び出すウィザードパネルを作成します。

このウィザードパネルは、別のウィザードを呼び出すバックグラウンドタスクを実行するだけなので、エンドユーザーに表示する画面はありません。したがって、このステップではフォームデザイナーの使用を必要とするタスクはありません。

タスク: 応答 SLO を呼び出すウィザードの作成

このタスクでは、既存の設定済みの応答 SLO ウィザードを呼び出すウィザードを作成します。

既存の設定済みの応答 SLO ウィザードを呼び出すウィザードを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「SLA Create Response 1」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードは、応答 SLO を作成する設定済みのウィザードを呼び出します。
ウィンドウタイトル	次のように入力します: 応答 SLO の作成
タイトル	次のように入力します: 応答 SLO の作成

5. [ファイル選択]タブを選択し、[\$L.file の選択基準]サブタブを選択して、次のフィールドに入力します。

フィールド	値
渡された \$L.file	オン

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
表示をスキップ	このパネルにはユーザに表示するフォームが含まれていないため、このオプションを選択します。

7. [アクション] タブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レ

フィールド	値
	コードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)]を選択します。
[フォーマットコントロール/プロセス名]サブタブ	ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。
プロセス名	このウィザードが完了後にプロセスを実行できることを指定するため、以下のように入力します。 slo.add.resp.sla

8. [次のウィザード]タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します： SLA Create Response 2
条件	次に何が起こるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します： true

9. 完了したら、[保存]をクリックします。

既存の設定済みの応答 SLO ウィザードを呼び出すウィザードを作成しました。このウィザードには新しいフォームは必要ありません。

ステップ 12: 応答 SLO の追加を求める

次のステップでは、ユーザが別の応答 SLO を追加するかどうかを尋ねます。追加する場合、このパネルは設定済みの可用性 SLO ウィザードを再度呼び出す必要があります。これ以上応答 SLO を追加しない場合は、最後のウィザード画面を表示する必要があります。

以下の 2 つのタスクを実行して、別の応答 SLO を求めるウィザードパネルを設計します。

[追加の応答 SLO のウィザードの作成](#)

[追加の応答 SLO のフォームの作成](#)

タスク 1: 追加の応答 SLO のウィザードの作成

このタスクでは、ウィザードを作成してユーザが追加の応答 SLO を作成できるようにします。

ウィザードを作成して、ユーザが追加の応答 SLO を作成できるようにするには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。

3. [ウィザード名]フィールドで「SLA Create Response 2」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードパネルでは、ユーザが別の応答 SLO を追加するかどうかを尋ねます。
ウィンドウタイトル	次のように入力します: 別の応答 SLO を作成する
タイトル	次のように入力します: 別の応答 SLO を作成する

5. [ファイル選択]タブを選択し、[\$L.file の選択基準]サブタブを選択して、次のフィールドに入力します。

フィールド	値
初期式	\$L.create.response.slo=false
\$L.file なし (typecheck を使用)	このオプションを選択すると、保持ファイルとして動作する typecheck ファイルを初期化します。

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
表示するサブフォーマット	表示するフォーマットのサブフォーマット名を入力します。次のように入力します: sla.create.ask.other.response
ユーザ入力の要求	オン
[前へ]ボタンだけをオフにする	オン

7. [アクション]タブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)]を選択します。
ウィザード変数	\$L.create.response.slo \$L.crate.avail.slo

8. [次のウィザード]タブを選択し、次のフィールドに入力します。

フィールド	値
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します: SLA Create Response 1
条件	次に何が起こるかを決定する条件です。今の場合、一連のウィザードの中で次のウィザードの名前と、trueに評価される式を入力できます。代わりに、ユーザのアクションまたは選択に基づいて、別のウィザードを指定できます。次のように入力します: \$L.create.response.slo=true
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します: SLA Create Availability 1
条件	次のように入力します: \$L.create.avail.slo=true
ウィザード名	これは、一連のウィザードの次のウィザードの名前です。次のように入力します: SLA Create 8
条件	次のように入力します: true

9. 完了したら、[保存]をクリックします。

ユーザが追加の応答 SLOを作成できるようなウィザードを作成しました。次は、新しいウィザードフォームをフォームデザイナーで作成します。

タスク 2: 追加の応答 SLO のフォームの作成

このタスクでは、フォームデザイナーを使用して、ユーザが追加のサービスレベル目標レコードを作成するかどうかを質問する、フォームコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、次の画面がエンドユーザに表示されます。

別の応答SLOを作成する

別の応答SLOレコードを作成しますか?

☒ はい ☐ いいえ

< 前へ

次へ >

終了

キャンセル

追加の応答 SLO を作成するウィザードに新しいフォームを作成するには：

1. フォームデザイナーを起動します。
2. [フォーム] フィールドに「sla.create.ask.other.response」と入力します。
ベストプラクティス: フォーム名はサブフォーマットの表示エントリと正確に一致する必要があるの
で、[使用方法] タブの表示するサブフォーマットをコピーして、[フォーム] フィールドに貼り付けるこ
とができます。これにより、スペルの誤りを回避できます。
3. [新規] をクリックします。
4. フォームウィザードの使用については[いいえ] をクリックします。
5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標
が 0 (ゼロ) であることを確認します。

プロパティ	値
グループ	
Caption	次のように入力します： 別の応答 SLO レコードを作成しますか？
Visible	選択します。
ラジオボタン	

プロパティ	値
Caption	次のように入力します: はい
Input	次のように入力します: \$L.create.response.slo
値	次のように入力します: true
Visible	選択します。
ラジオボタン	
Caption	次のように入力します: いいえ
Input	次のように入力します: \$L.create.response.slo
値	次のように入力します: false
Visible	選択します。

6. [OK]を2回クリックして、新規フォームを保存し終了します。

SLA ウィザードに必要なフォームの1つを作成しました。

ステップ 13: 新しい SLA 番号とタイトルの表示

これが最後のウィザードのステップです。新しく作成した SLA のレコード番号とタイトルを表示し、ユーザが[終了]ボタンをクリックしたらレコードの詳細を表示します。

以下のタスクを完了してこのステップを実行します。

[完了した SLA 用のウィザードの作成](#)

[完了した SLA 用のフォームの作成](#)

タスク 1: 完了した SLA のウィザードの作成

このタスクでは、SLA レコード番号および新しく作成した SLA レコードのタイトルをユーザに表示するウィザードパネルを作成します。

完了した SLA 用のウィザードを作成するには:

1. [カスタマイズ]>[ウィザード]をクリックします。[ウィザード情報]フォームがオープンします。
2. [ウィザード情報]タブを選択します。
3. [ウィザード名]フィールドで「**SLA Create 8**」と入力し、[追加]をクリックします。

注意: 既存の SLA レコードを使用して、新しい SLA ウィザードパネルを作成する場合、[保存]ではなく[追加]をクリックします。[保存]をクリックすると、追加しようとしている新規のレコードで既存のレコードが置き換えられます。

4. [ウィザード情報]タブをクリックし、次のフィールドに入力します。

フィールド	値
開始ノード	このオプションはオフのままにしておきます。
概要	次のように入力します: このウィザードはユーザに表示される最後のパネルで、レコード番号および作成された新規 SLA の名前を表示します。
ウィンドウタイトル	次のように入力します: 新規 SLA の作成 - 完了
タイトル	次のように入力します: 新規 SLA の作成 - 完了

5. [ファイル選択]タブを選択し、次のフィールドに入力します。

フィールド	値
[初期式]サブタブ	次のように入力します:
[\$L.file の選択基準]サブタブ	
レコードの選択 [対象タイプ]	このオプションを選択し、[フィル]をクリックして、レコードタイプとして[sla]を選択します。
使用するクエリ	次のように入力します: "agreement.id="+\$L.new.sla 注:それぞれの "agreement.id" の最後に二重引用符 (") を忘れずに含めてください。二重引用符はクエリの一部です。

6. [使用方法]タブを選択し、次のフィールドに入力して、ウィザードの実行時に HP Service Manager が実行するアクションを決定します。

フィールド	値
ユーザ入力の要求	このフィールドを選択すると、ユーザに情報の入力を求めます。
表示するサブフォーマット	表示するフォーマットのサブフォーマット名を入力します。次のように入力します: sla.create.finished

フィールド	値
[終了]オプションを有効にする	これが一連のウィザードパネルの最後のウィザードなので、このオプションを選択すると、このウィザードパネルに[終了]ボタンが表示されます。
[次へ]および[前へ]ボタンをオフにする	チェックボックスを選択します。

7. **【アクション】** タブを選択して、ユーザがウィザード内で[次へ]、[キャンセル]、または[前へ]をクリックしたときのアクションを定義します。以下のフィールドに値を指定します。

フィールド	値
アクションの実行対象	ここでは、実行するアクションによって影響を受けるレコードを指定できます。これは、現在のファイル、選択したリスト全体、または選択した各レコードに対して実行されるアクションのいずれかです。 [現在のファイル (\$L.file)]を選択します。
完了時にレコードを表示	このオプションを選択 (true に設定) すると、ウィザードの完了時にユーザに対してレコードが表示されます。

8. 完了したら、**【保存】**をクリックします。

SLA レコード番号および新しく作成した SLA レコードのタイトルを表示する最後のウィザードパネルを作成しました。次は、このウィザードの手順のフォーマットをフォームデザイナーで作成します。

タスク 2: 完了した SLA のフォームの作成

このタスクでは、フォームデザイナーを使用して、作成された新しい SLA レコードのレコード番号とタイトルを表示するフォームコントロールを作成します。これらのフォームコントロールと、ウィザードレコードで選択したウィンドウタイトル、プロンプト、ボタンなどの選択項目が組み合わされて、次の画面がエンドユーザに表示されます。

新規SLAの作成 - 完了

SLAレコード

190

XYZ Co. General Support Agreement

は正常に作成されました。[終了]をクリックしてレコードを表示します。

< 前へ

次へ >

終了

キャンセル

完了した SLA 用の新しいフォームを作成するには:

1. フォームデザイナーを起動します。
2. [フォーム]フィールドに「**sla.create.finished**」と入力します。
ベストプラクティス: フォーム名はサブフォーマットの表示エントリと正確に一致する必要があるの
で、[使用方法]タブの表示するサブフォーマットをコピーして、[フォーム]フィールドに貼り付けるこ
とができます。これにより、スペルの誤りを回避できます。
3. [新規]をクリックします。
4. フォームウィザードの使用については[いいえ]をクリックします。
5. このフォーム用に以下を作成します。新しいフォームの設計を開始する場合、X および Y 座標
が 0 (ゼロ)であることを確認します。

プロパティ	値
ラベル	
Caption	次のように入力します: SLA レコード
Visible	選択します。
テキスト	
Input	次のように入力します: agreement.id
Visible	選択します。
テキスト	
Input	次のように入力します: title
Visible	選択します。
ラベル	
Caption	次のように入力します: は正常に作成されました。[終了]をクリックしてレコードを表示しま す。
Visible	選択します。

6. [OK]を2回クリックして、新規フォームを保存し終了します。

SLA ウィザードに表示される、最後のウィザード画面の作成が完了しました。

ステップ 14: 新規 SLA ウィザードを呼び出すメニューの設定

このステップでは、新しいウィザードと、[サービスレベル管理]>[サービスレベル契約]>[新規 SLA の作成]というメニューコマンドを関連付けます。これにより、初期設定の[サービスレベル契約]入力フォームではなく、新しいウィザードの最初のパネルが表示されるようになります。

新しいウィザードを呼び出すようメニューを設定するには:

1. [カスタマイズ]>[カスタマイズツール]>[メニュー]をクリックします。
2. [メニュー名]フィールドに、次のように入力します:「SLA」と入力し、[検索]をクリックします。
3. オプション 5 – Service Level Agreements の行をクリックします。
4. [アプリケーション]列の、「thread.start」を「wizard.run」に置き換えます。
5. [パラメータ値]列の、「sla.add.object」を「SLA Create 1」に入力して置き換えます。
6. [OK]をクリックして変更を保存します。

[サービスレベル管理]>[サービスレベル契約]>[新規 SLA の作成]をクリックして、メニューの変更が反映され、ウィザードが期待どおりにオープンして実行されることを確認します。

第 5 章

カスタムウィザードのトラブルシューティング

ウィザードの実行時に発生する可能性のある問題のトラブルシューティングは、このセクションのトピックで確認します。エラーログは、トラブルシューティングに役立ちます。

以下のトピックについて解説しています。

- [ボタンオプションが正しく表示されていません](#)
- [ファイル変数がウィザードフローに渡されていません](#)
- [フォーマットが見つかりません](#)
- [JavaScript の構文エラー](#)
- [クエリは正しい結果を返せませんでした](#)
- [単一レコードではなく、複数のレコードがクエリによって返されました](#)
- [RAD 式の構文エラー](#)
- [レコードは正しくソートされていません](#)
- [ウィザードが停止し、「続行できませんでした」というエラーが表示されます](#)
- [トレースの実行](#)

ボタンオプションが正しく表示されていません

ボタンオプションが表示されていないか、または想定した形で表示されていない場合は、[使用方法]タブを確認して正しいチェックボックスが有効になっていることを確認します。

また、次のことを確認します。

- 最初に表示されるパネルには、[前へ]ボタンがないこと。
- 最後に表示されるパネルには、[終了]ボタンがあること。
- [前へ]ボタンが想定どおりに表示されていない場合、[ウィザード情報]タブの[開始ノード]オプションを選択したかどうかを確認します。

ヒント: [ウィザード情報]タブの[開始ノード]オプションをオンにする必要があるのは、最初のウィザードパネルだけです。

- [終了]ボタンとして[次へ]ボタンを選択しないでください。そうすると、ウィザードは実行を停止します。ベストプラクティスとして、ウィザードワークフローの最後のパネルでは、[次へ]ボタンを無効にして、[終了]ボタンを有効にします。
- [使用方法]タブのオプションのいずれかが、誤って抑制されていないかどうかを確認します。

ファイル変数がウィザードフローに渡されていません

ウィザードは実行を停止します。画面の上部には次のエラーメッセージが表示され、ユーザは呼び出し元のアプリケーションに戻ります。

ウィザードは続行できませんでした。ファイル変数が渡されませんでした。

原因

[ファイル選択]タブの[\$L.file の選択基準]サブタブで、オプション "渡された\$L.file" が選択されましたが、ファイル変数は渡されませんでした。

解決方法

wizard.run を正しいファイルパラメータを渡して呼び出すか、または[ファイル選択]タブの[\$L.file の選択基準]タブで別のオプションを選択します。たとえば、[レコードの作成 対象タイプ]を選択します。

フォーマットが見つかりません

ウィザードは実行を停止します。画面の上部には次のエラーメッセージが表示され、ユーザは呼び出し元のアプリケーションに戻ります。

フォーマット "test" が見つかりません (display,show.rio)

原因

[使用方法]タブの[ウィザードの使用 方法]セクションの「ユーザ入力 の要求」オプションを使用中に、[使用方法]タブの[表示するサブフォーマット]フィールドに無効なフォーマット名が入力されました。

解決方法

[使用方法]タブの[表示するサブフォーマット]フィールドに有効なフォーマット名を入力して、有効なサブフォーマットが表示されるようにします。または[表示をスキップ] オプションを選択して、フォームを表示しようとするのをバイパスします。

JavaScript の構文エラー

ウィザードは実行を停止します。画面の上部には次のエラーメッセージが表示され、ユーザは呼び出し元のアプリケーションに戻ります。

プロセスパネル <パネル名> (RAD wizard.run) の 5 行目でエラーが発生しました。(wizard.run,<パネル名>)

プロセスパネル init.expr (RAD wizard.run) の 5 行目でエラーが発生しました。(wizard.run,init.expr)

式を評価できません (wizard.run,init.expr)

スクリプト <不明> 1 行目:エラー 参照エラー:test は char 1 で定義されていません

アプリケーションの回復 不可能なエラー:パネル call.rad.1 上の se.call.process

アプリケーションの回復 不可能なエラー:パネル init.expr 上の wizard.run

原因

[JavaScript]タブに無効な構文が入力されました。RAD アプリケーションとパネルによって、構文エラーが発生した[JavaScript]タブのヒントが示されます以下のテーブルを参照して、どこを修正すれば良いかを判断します。

RAD アプリケーション / パネル	修正箇所
wizard.run,init.expr	[ファイル選択]タブの[JavaScript]サブタブ。
wizard.run,select.file.setup	[ファイル選択]タブの[\$L.file の選択基準]サブタブのクエリステートメント。
wizard.run,run.expressions	[アクション]タブの[式]サブタブ。
wizard.run,exit.cancel	[キャンセル式]タブの[キャンセル時に実行される JavaScript]サブタブ。

解決方法

JavaScript 式の正しい構文については、『JavaScript Programmer's Guide』で解説しています。JavaScript を使用する場合、変数は使用する前に定義する必要があります。例：

```
var <name>=new String()
```

フィールドに値を割り当てる正しい構文は次のとおりです。

```
Table.field1=value
```

2つのフィールドの値を互いに比較するには、次の構文を使用します。

```
Table1.field1==table2.field2
```

クエリは正しい結果を返せませんでした

[ファイル選択]タブにクエリを入力する場合、フィールド名などのすべての文字列を引用符で囲む必要があります。文字列が引用符で囲まれていない場合、クエリは正しい結果を返すことができません。

例：

```
"agreement.id="+G.new.sla
```

または

```
"logical.name=\""+str($group.name.new)+"\""
```

単一レコードではなく、複数のレコードがクエリによって返されました

ウィザードのステップによっては、[使用方法]タブに入力されたクエリから返されるレコードは単一である必要があります。代わりに、真偽判定検索が実行されました。

原因

無効なクエリ構文が使用されました。例えば、`agreement.id in $L.file=168` です。クエリステートメントにはファイル変数を含めないでください。

解決方法

正しいクエリ構文を使用します。例えば、`"agreement.id="168`。

RAD 式の構文エラー

ウィザードは実行を停止します。画面の上部には次のエラーメッセージが表示され、ユーザは呼び出し元のアプリケーションに戻ります。

プロセスパネル <パネル名> (RAD wizard.run) の 2 でエラーが発生しました。

(wizard.run, <パネル名>)

プロセスパネル init.expr (RAD wizard.run) の 2 行目でエラーが発生しました。

(wizard.run, init.expr)

式を評価できません (wizard.run, init.expr)

Bad arg (2) oper = (wizard.run, init.expr)

Bad arg (3) oper index (wizard.run, init.expr)

アプリケーションの回復不可能なエラー: パネル call.rad.1 上の se.call.process

アプリケーションの回復不可能なエラー: パネル init.expr 上の wizard.run

原因

これらのエラーは、ウィザードパネルの構文エラーが原因です。パネル名は、問題が発生した場所のヒントになります。

正しくない引数 (x) oper 演算子 <y> ステートメントの構文が適切でない、または見つからない、またはステートメントに渡されるはずのパラメータが無効であることを示しています。

正しくない引数 (1 または 2) oper <演算子>: 引数 1 は、<演算子> の左側に問題があることを示しています。引数 2 は、<演算子> の右側に問題があることを示しています。例えば、演算子は次のように定義できます。=、<、>、~=、~<、~>、など。

正しくない引数 (1 または 2) oper <in> 引数 1 は、ステートメントの "in" の左側に問題があることを示しています。引数が 2 の場合、ステートメントの "in" の右側に問題があります。

正しくない引数 (x) oper <関数> 関数に渡されたパラメータの数が無効であることを示しています。

次のテーブルは、ウィザードパネル内のエラーが発生した箇所および修正すべき箇所を見つけるのに役立ちます。

RAD アプリケーション / パネル	修正箇所
wizard.run, init.expr	[ファイル選択] タブの [初期式] サブタブ。
wizard.run, select.file.setup	[ファイル選択] タブの [\$L.file の選択基準] サブタブのクエリステートメント。
wizard.run, get.selection.records	[使用方法] タブの [\$L.file の選択基準] のクエリステートメント。
wizard.run, run.expressions	[アクション] タブの [式] サブタブ。
wizard.run, decide.restart	[アクション] タブの [パネルを再起動する条件] フィールド。
wizard.run, get.message	[メッセージ] タブの [条件] フィールド。
wizard.run, setup.wizard.variables	[変数] タブのウィザード変数。

RAD アプリケーション / パネル	修正箇所
wizard.run, find.next.wizard	[次のウィザード] タブの[条件] フィールド。
wizard.run, exit.cancel	[キャンセル式] タブの[キャンセル時に実行される式] サブタブ。

解決方法

RAD 式の正しい構文については、『Service Manager オンラインヘルプ』の「システム言語」トピックスを参照してください。Windows クライアントを使用している場合は、[ヘルプ]>[ヘルプの内容]をクリックしてこのヘルプを参照できます。Web クライアントを使用している場合は、[F1]を押します。

フィールドに値を割り当てる正しい構文は次のとおりです。

```
<field> in $L.file=value
```

2つのフィールドの値を互いに比較するには、次の構文を使用します。

```
<field1> in $L.file=<field2> in $L.file
```

フィールド名と変数を含む、連結ステートメントは次のように入力する必要があります。

"agreement.id=" + \$G.test \$G.test は数値またはブール値

"agreement.id =\" + \$G.test + "\" \$G.test は文字

"agreement.id='\" + \$G.test + '\"' \$G.test は日付 / 時刻

レコードは正しくソートされていません

ウィザードの設定によっては、選択されたレコードのリストが、特定のフィールドの値によってソートする必要がある場合があります。レコードのリストを表示する場合、代わりに固有キーでソートされます。

原因

クエリに必要なフィールドの種類 (dbdict 名) が無効です。

解決方法

有効なソートフィールドを、レコードタイプ dbdict に入力します。

ウィザードが停止し、「続行できませんでした」というエラーが表示されます

ウィザードは、考えられる構成エラーを含む、さまざまな理由から実行を停止します。発生する可能性があるエラーは以下のとおりです。

エラー	原因 / 解決方法
レコードタイプ "test" が作成できません。	<p>原因: 入力されたレコードタイプは、dbdict テーブルにレコードとして存在しません。</p> <p>解決方法: 有効なレコードタイプまたは dbdict 名を入力します。</p>

エラー	原因 / 解決方法
レコードタイプ "test" から選択できません。	<p>原因: 入力されたレコードタイプは、dbdict テーブルにレコードとして存在しません。</p> <p>解決方法: 有効なレコードタイプまたは dbdict 名を入力します。</p>
レコードタイプ "slas" が初期化できません。	<p>原因: [使用方法] タブの、[レコードのクエリ 対象タイプ] に無効なレコードタイプが入力されました。</p> <p>解決方法: 有効なレコードタイプまたは dbdict 名を入力します。</p>
無効な選択クエリ: agreement=\$G.test	<p>原因: [使用方法] タブで、レコードのクエリステートメントに無効なフィールド名が使用されています。</p> <p>解決方法: ステートメントの正しいフィールド名を特定し、使用します。</p>
ウィザード "test" が存在しません。システム管理者に連絡します。	<p>原因: [次のウィザード] タブに無効なウィザード名が入力されました。存在しないウィザードが true に評価されると、ウィザードワークフローは無限ループに陥り、セッションスレッドは強制終了されます。</p> <p>解決方法: [次のウィザード] タブに有効なウィザード名を入力します。</p>
<p>表示アプリケーションは画面を見つけることができませんでした。</p> <p>Query=screen.id="wizard.test.me" and language="ENG"</p> <p>アプリケーションの回復不可能 ★Mismatch Translation★ エラー: パネル call.rad.1 上の se.call.process★Mismatch Translation★</p> <p>表示アプリケーションエラー終了が選択されました。\$L.ds.ids=L.ds.desc=</p>	<p>原因: [使用方法] タブで無効な表示画面名が選択されました。</p> <p>解決方法: [使用方法] タブで、有効な表示画面名を入力するか、デフォルトのオプションとしてフィールドを空白のままにしておきます。</p>

トレースの実行

エラーメッセージが表示される場合は、以下を使用してトレースを行い、表示されているエラーの詳細を取得します。ウィザードフローが想定したパスを通らない場合にもトレースを実行できます。

トレースを実行するには:

1. **RTM:3** および **debugdbquery:999** パラメータを、Service Manager の **sm.ini** ファイルに入力します。
2. 新規のクライアント接続を開始します。

完全な詳細については、「*Diagnostics and Tuning (診断とチューニング)*」ホワイトペーパーの指示を参照してください。

3. トレース内で、`wizard.run` RAD アプリケーションの最初の呼び出しを検索します。
4. 次に、ウィザードのフローを追跡し、ウィザードテーブルに対する `dbquery` ステートメントを確認して、どのウィザードが次に呼ばれるかを確認します。
5. エラーメッセージが表示される場合は、エラーメッセージに対するトレースを追跡するか、またはトレースを追跡してどのパスでアプリケーションがウィザードフローを通過したかを把握します。
6. ウィザードパネルの、JavaScript 式の `print` ステートメントを使用して、追加情報をログファイルに書き出します。例えば、次に実行するウィザードに影響を与える変数の内容、またはこの時点で実行するウィザードのステートメントなどです。

注：ウィザードを実働環境に移す前に、これらのデバッグステートメントは忘れずに削除してください。

