

PeopleSoft®

EnterpriseOne 8.9
Web クライアント
PeopleBook

2003 年 9 月

PeopleSoft EnterpriseOne 8.9
Webクライアント PeopleBook
SKU AC89JDW0309

Copyright 2003 PeopleSoft, Inc. All rights reserved.

本書に含まれるすべての内容は、PeopleSoft, Inc. (以下、「ピープルソフト」) が財産権を有する機密情報です。すべての内容は著作権法により保護されており、該当するピープルソフトとの機密保持契約の対象となります。本書のいかなる部分も、ピープルソフトの書面による事前の許可なく複製、コピー、転載することを禁じます。これには電子媒体、画像、複写物、その他あらゆる記録手段を含みます。

本書の内容は予告なく変更される場合があります。ピープルソフトは本書の内容の正確性について責任を負いません。本書で見つかった誤りは書面にてピープルソフトまでお知らせください。

本書に記載されているソフトウェアは著作権によって保護されており、このソフトウェアの使用許諾契約書に基づいてのみ使用が許諾されます。この使用許諾契約書には、開示情報を含むソフトウェアと本書の使用条件が記載されていますのでよくお読みください。

PeopleSoft、PeopleTools、PS/nVision、PeopleCode、PeopleBooks、PeopleTalk、Vantiveはピープルソフトの登録商標です。Pure Internet Architecture、Intelligent Context Manager、The Real-Time Enterpriseはピープルソフトの商標です。その他すべての会社名および製品名は、それぞれの所有者の商標である場合があります。ここに含まれている内容は予告なく変更されることがあります。

オープンソースの開示

この製品には、Apache Software Foundation (<http://www.apache.org/>) が開発したソフトウェアが含まれています。Copyright (c) 1999-2000 The Apache Software Foundation. All rights reserved. このソフトウェアは「現状のまま」提供されるものとし、特定の目的に対する商品性および適格性の黙示保証を含む、いかなる明示または黙示の保証も行いません。Apache Software Foundationおよびその供給業者は、損害の発生原因を問わず、責任の根拠が契約、厳格責任、不法行為(過失および故意を含む)のいずれであっても、また損害の可能性が事前に知らされていたとしても、このソフトウェアの使用によって生じたいかなる直接的損害、間接的損害、付随的損害、特別損害、懲罰的損害、結果的損害に関しても一切責任を負いません。これらの損害には、商品またはサービスの代用調達、使用機会の喪失、データまたは利益の損失、事業の中断が含まれますがこれらに限らないものとします。

ピープルソフトは、いかなるオープンソースまたはシェアウェアのソフトウェアおよび文書の使用または頒布に関しても一切責任を負わず、これらのソフトウェアや文書の使用によって生じたいかなる損害についても保証しません。

目次

アプリケーション・ユーザー・インターフェイス	1
Web クライアントへのアクセス.....	1
タスク・エクスプローラ.....	2
略式コマンド.....	4
お気に入り用タスク・ビュー.....	5
ロール・ベース用タスク・ビュー.....	6
ユーザー・オプション.....	6
タスク・エクスプローラの設定.....	7
タスク・エクスプローラのトラブル・シューティング.....	9
J.D. Edwards Web アプリケーション/レポート.....	9
アプリケーション・ショートカット.....	10
Web アプリケーション・フォーム.....	11
グリッド.....	14
メディア・オブジェクト添付.....	19
Web アプリケーション設計	24
ユーザー中心の設計ガイドライン.....	24
ユーザーの定義.....	24
ユーザーの目標と業務目標の定義.....	25
ユーザー・タイプ別のソリューション.....	25
ユーザー・タスクの定義.....	27
使用事例の作成.....	29
有用性の設計.....	29
情報のアーキテクチャ.....	31
ワイヤ・フレームの作成.....	31
基本的な設計ガイドライン.....	32
ナビゲーションの定義.....	32
検索インターフェイスの定義.....	34
HTML と Windows の機能上の相違点.....	35
外観の相違.....	36
動作の相違点.....	36
プラットフォームの互換性.....	43
Web 用アプリケーションの設計.....	43
フォーム設計ツール(FDA)を使用した Web アプリケーションの設計.....	45
モバイル用アプリケーションの設計.....	62

モバイル・デバイスのランタイム・アーキテクチャ.....	62
モバイル・デバイスの設計方針.....	62
HTML デバイスとモバイル・デバイスの機能上の相違点.....	63
マルチ・アプリケーション・フレームワーク	65
MAF コンポーネントの実装.....	65
Web クライアントの構成	70
ログインの構成.....	70
匿名ユーザー.....	70
直接ログイン.....	72
基本認証.....	73

アプリケーション・ユーザー・インターフェイス

EnterpriseOne の大部分のアプリケーションは Web クライアントで使用することができます。プラットフォームが異なるため、インターフェイスに違いが見られる場合もありますが、アプリケーションには Windows バージョンと同じ機能があります。

Web ベースのアプリケーションは JAS サーバーと連動しています。JAS サーバーは、EnterpriseOne を実行するときにシステム管理者により設定されます。ただし、ほとんどの場合、Web と Windows 版のそれぞれのアプリケーションは同じデータベース・テーブルを参照するため、システムによっては、どちらのユーザーも同じシステムを同時に使用でき、すべての変更をリアルタイムで確認することができます。さらに、JAS サーバーはモバイル機器との通信が可能なため、モバイル機器から EnterpriseOne アプリケーションにアクセスすることもできます。

システム管理者のセキュリティ設定またはポータルが実行されているかどうかによって、ユーザーは、Web クライアントに直接ログインすることも、ポータルを通してログインすることもできます。どちらの場合も、ユーザーはタスク・エクスプローラを通して EnterpriseOne アプリケーションにアクセスできます。Web クライアントに直接ログインするユーザーは、タスク・エクスプローラにはスタンドアロンのインターフェイスとしてアクセスしますが、ポータルを通してログインするユーザーには、ポータル・コンポーネントを通してタスク・エクスプローラにアクセスします。タスク・エクスプローラは、Windows バージョンの Solution Explorer にあたるもので、同じ機能を提供します。

Web クライアントへのアクセス

Web で EnterpriseOne ソフトウェアにアクセスすることは、社内ネットワークでファイルやアプリケーションにアクセスするのと似ています。通常、コンピュータを起動したり会社のネットワーク・ディレクトリにアクセスする場合、ネットワークへのアクセス権があるユーザーであることをシステムが検証するためにログインする必要があります。ログインした後は、ネットワークのファイルやアプリケーションにアクセスできます。同じように、EnterpriseOne アプリケーションを実行するには、EnterpriseOne ソフトウェアへのゲートウェイとなる Web クライアントにログインする必要があります。

Web クライアントにログインするには、社内イントラネットにアクセスでき、コンピュータに Web ブラウザがインストールされており、システム管理者によってアカウントが設定されていることが必要です。モバイル・デバイスを使用している場合、アカウントのみ設定されればログインできます。システム管理者はアカウントが設定すると、次にユーザーID を作成してパスワードを割り当てます。

通常、Web クライアントの起動時にユーザーID とパスワードを入力します。ただし、システム管理者は、ログイン・プロセスをバイパスしたように見えるように設定することも可能です。ただし、セキュリティ上の理由で、ほとんどのシステム管理者はユーザーに手作業でログインするよう設定しています。

Web クライアントにログインすると、タスク・エクスプローラが表示されます。タスク・エクスプローラでは、EnterpriseOne アプリケーションやレポート、その他の機能にアクセスできます。

Enterprise One Web クライアントへのログインは、社内ネットワークへのログインに似ていますが、ログオフは異なります。ほとんどの場合、ネットワーク接続からログオフする場合は、コンピュータの電源を切るだけで済みます。ただし、Web クライアントにログインした場合は、作業が完了した時点でログオフする必要があります。つまり、Web ブラウザやコンピュータやモバイル・デバイスの電源をオフにしなければなりません。Web ブラウザを閉じてコンピュータの電源を切る前に、ログオフしてください。

注:

タスク・エクスプローラにはポータルからもアクセスできます。この場合、すでにポータルにログインしているため、タスク・エクスプローラにログインする必要はありません。同じように、ポータルからログオフする必要はありますが、タスク・エクスプローラからログオフする必要はありません。

▶ Web クライアントからログオフするには

タスク・エクスプローラで、[ログアウト]をクリックします。

▶ Web クライアントにログインするには

1. Web ブラウザを起動して EnterpriseOne Web ログイン画面を表示してください。

システムがどのように Web クライアントにサインオンするかは、システム管理者によって決定されます。たとえば URL を使う場合、アイコンをクリックする場合、またはブラウザのお気に入りフォルダを使ってサインオンする場合などがあります。Web ログイン画面を表示できない場合は、システム管理者に連絡してください。

2. 次のフィールドに入力します。

- User ID (ユーザーID)
- Password (パスワード)

3. システム管理者によって特定の環境にログインするよう指定されている場合は、[ログイン詳細]をクリックして次のフィールドに値を入力してください。

- Environment (環境)

4. [ログイン]をクリックします。

タスク・エクスプローラが表示されます。

タスク・エクスプローラ

タスク・エクスプローラは Web ブラウザで実行され、EnterpriseOne アプリケーションやレポート、その他の機能にアクセスできます。Web クライアントにログインすると、ワークステーションまたはモバイル・デバイスでタスク・エクスプローラが自動的に表示されます。ポータルからもタスク・エクスプローラ・コンポーネントを通してアクセスできます。

タスク・エクスプローラに直接アクセスする場合(ポータルからアクセスしない場合)、右上に[ログアウト]ボタンが表示されます。タスク・エクスプローラでの作業が終わったら、[ログアウト]をクリックします。

タスク・エクスプローラの上部には、ハイパーリンクやボタン、ドロップダウン・メニューのあるバーがあります。タスク・エクスプローラの主な部分にはツリーが表示され、実行する特定のアプリケーションに移動することができます。ツリーには、アプリケーションの他にも、ショートカットやリンクなどのオブジェクトが含まれます。このため、ツリーのすべてのオブジェクトはタスクと呼ばれます。ツリーにあるフォルダやアプリケーション(レポート)、リンク、ショートカットは、すべてタスクです。フォルダをクリックするたびに、ツリー・レベルやツリー変更のビューを展開したりすることができます。

主なタスクのまとまりを、タスク・ビューと呼びます。ログインすると、タスク・エクスプローラにはデフォルトのタスク・ビューが表示されます(定義したビューが1つの場合)。そうでない場合は、OneWorldメニュー・タスク・ビューが表示されます。タスク・ビューにショートカットが含まれていない限り、1つのタスク・ビューから別のタスク・ビューへは移動できません。通常、タスク・ビュー・フィールドを使ってタスク・ビューを切り替えますが、システム管理者の設定によって、タスク・ビュー・フィールドが表示されない場合があります。

タスク・ビューとツリーを展開したときに表示される新しいビューとは別のものです。ツリー・ビューは変更されても、新しいタスク・ビューに位置するわけではありません。タスク・エクスプローラに関するガイドでは、ツリー・ビューをビューとして変更する方法が説明されています。

システム管理者は、Solution Explorer とセキュリティ設定アプリケーションを使って、ユーザー・アカウントを管理してタスク・エクスプローラを設定します。

参照

- タスク・ビューの設定については『Solution Explorer』ガイドの初期設定に関するトピック
- エンド・ユーザーが使用できるタスク・エクスプローラの機能の設定については『システム・アドミニストレーション』ガイドの「Solution Explorer セキュリティ」

▶ デフォルトのタスク・ビューを変更するには

1. タスク・エクスプローラで、[タスク・ビュー]フィールドのドロップダウン・メニューからデフォルトのタスク・ビューとして使用するタスク・ビューを選択します。
2. [デフォルトとして保存]をクリックします。

▶ タスク・エクスプローラ内で移動するには

1. タスク・エクスプローラで、[タスク・ビュー]フィールドのドロップ・ダウン・リストから[タスク・ビュー]を選択します。
ツリーには、タスク・ビューのツリーのルート構造が表示されます。
2. フォルダ・アイコンのあるタスクをクリックしてください。
ツリーには、クリックしたタスクの下にのみタスクが表示されます。タスク・エクスプローラは、現在のビュー名を前のビュー名に追加し、前のビュー名をハイパーリンクにします。タスク・ビュー構造を辿っていくと、タスク・ビューにより、現在のビュー名がタイトルに追加され、前のビュー名がハイパーリンクにされていきます。この表示により、現在のビュー表示の履歴がわかり、過去のビューに戻ることができます。
3. 現在のビューから前のビューに戻るには、前のビューのハイパーリンクをクリックしてください。

注:

前のタスク・ビューに戻る場合は、ブラウザの[戻る]ボタンは使わないでください。

略式コマンド

略式コマンドは、特定のタスク(フォルダやアプリケーション、レポート)に直接アクセスできます。略式コマンドを使うと、メニューやアプリケーション間を簡単に移動できます。略式コマンドには、次の項目があります。

- デモ・データと共に出荷される略語、または環境に合わせて定義する略語。たとえば、“BV”と入力すると、バッチ・バージョン・アプリケーションにアクセスしてレポートを実行できます。
- タスク ID。
- プログラム名。

略式コマンドを使用するには、[ツールバー]に略式コマンドを入力して、[Enter]キーを押します。

システム管理者によるアカウント設定によって、略式コマンドのセキュリティを表示したり変更したりできない場合があります。

タスク・エクスプローラでは略式コマンドを使用して、タスク・ビュー、EnterpriseOne アプリケーションなどを実行できます。タスク・ビューを指定するには、“TV:”に続けて内部タスク ID を入力します。たとえば、“TV:98”と入力すると、お気に入りタスク・ビューにアクセスできます。

[略式コマンド]フィールドを使用するとメニューにもアクセスできます。タスク・ビューはメニューおよび個々のタスクから構成されます。タスク・エクスプローラには特別なフォーマットはありません。アプリケーション開発者が使いやすいようにアプリケーションをグループ化する方法を提供しています。メニューにアクセスする場合は、タスク・ビュー内の特定の場所にアクセスします。メニューにアクセスするには ID を入力します。たとえば“G0”と入力すると、基本設定メニューにアクセスできます。

アプリケーションを実行するには、アプリケーションのプログラム番号を入力してください。フォームを指定するには、アプリケーション ID、|、フォーム ID の順に入力してください。たとえば、“P01012 | W01012”と入力すると、〈住所録の処理〉アプリケーションで〈住所録の処理〉が表示されます。開くフォームのバージョンを指定する場合は、プログラム名、|、バージョン番号の順に入力します。(例：“P01012|W01012B|ZJDE0003”)

特定の内部タスクやメニュー、アプリケーション ID についてはシステム管理者に問い合わせてください。

注:

略式コマンドのないアプリケーションもあります。

▶ アプリケーションまたはレポートを実行するには

1. タスク・エクスプローラで、実行するアプリケーションまたはレポートに移動します。
2. 処理オプションやバージョンを定義せずにアプリケーションやレポートを実行するには、レポートまたはアプリケーションをクリックします。

アプリケーションが実行されます。バージョンを選択せずにレポートを実行すると、〈バッチ・バージョンの処理〉が表示され、実行するバージョンを選択できます。

3. アプリケーションの処理オプションやバージョンを選択するには、タスクの横にある緑色の矢印にカーソルを置いて、[処理オプション...]または[バージョン...]を選択してください。

メニュー・オプションが斜体で表示されている場合は、アプリケーションにはそのオプションが使用できないことを表します。

処理オプションまたはバージョンを選択すると、アプリケーションが実行されます。

4. 処理オプションやバージョンを選択したり、レポートのデータ選択パラメータを指定するには、タスクの横にある緑色の矢印の上にカーソルを置いて、次のオプションのうち1つを選択してください。

- 処理オプション
- バージョン
- データ選択
- データ選択および処理オプション

メニュー・オプションが斜体で表示されている場合は、レポートにはそのオプションが使用できないことを表します。

オプションを設定すると、〈バッチ・バージョンの処理〉が表示され、実行するバージョンを選択できます。次に、〈バージョン・プロンプト〉フォームが表示されます。プロンプトを選択して[投入]をクリックし、プリンタを選択してレポートを処理してください。

参照

- アプリケーションの実行については『基本操作』ガイドの「対話型バージョン・アプリケーション」
- レポートの実行については『基本操作』ガイドの「レポート用バッチ・バージョン」
- 処理オプションの使用については『基本操作』ガイドの「処理オプション」

お気に入り用タスク・ビュー

お気に入りタスク・ビューには、別のタスクへのリンクを保存することができます。最も頻繁に使用するタスクは、お気に入り用タスク・ビューに保存すると便利です。1つの場所からこれらのタスクにすばやく簡単にアクセスできます。

お気に入りのタスクを識別するお気に入りを作成し、これらのお気に入りを、タスクを識別する子として整理することができます。お気に入りタスク・ビューまたはビューに加えた変更は、他のユーザーには表示されません。同じ環境でログインした場合、Solution Explorer (Windows バージョンのタスク・エクスプローラ)を使用すると、同じリストがアプリケーションに表示されます。

システム管理者によるアカウント設定によって、お気に入りタスク・ビューのセキュリティを表示したり変更することができない場合があります。

▶ お気に入りタスク・ビューにアクセスするには

タスク・エクスプローラで、[タスク・ビュー]フィールドのドロップ・ダウン・リストから[お気に入り]を選択します。

▶ お気に入りタスク・ビューにタスクを追加するには

1. タスク・エクスプローラで、お気に入りタスク・ビューに追加するタスクへ移動します。
2. タスクの横にある矢印にカーソルを置いて[送る...]を選択します。

ルール・ベース用タスク・ビュー

システム管理者は、必要なタスクにアクセスしやすいように、別バージョンのタスク・ビューを作成できます。ルールは、カテゴリとグループ・ユーザーに分けるための方法です。システム管理者により、アカウント作成時にルールが割り当てられます。ルール・ベース用タスク・ビューでは、特定のジョブを実行するのに必要なアプリケーションやレポートのリストが表示されます。

▶ ルールベース用タスク・ビューにアクセスするには

1. タスク・エクスプローラで、[ビュー]メニューから[タスク・ビュー] - [エンドユーザー・タスク] (ルールベースのタスク・ビューの例)を選択します。

表示されるタスク・ビューは、デフォルト・ルールに有効なビューです。

2. ルールを変更してルールベース用タスク・ビューの異なる選択を表示するには、[ルール]をクリックしてルールを選択してください。

表示されるルールは、システム管理者によってアクセスを許可されたルールです。

ユーザー・オプション

タスク・エクスプローラで、[ビュー]メニューから[ユーザー・オプション]を選択すると、〈ユーザー・デフォルトの改訂〉フォームが表示されます。次のリストでは、〈ユーザー・デフォルトの改訂〉の各ボタンで行う設定について説明します。

ボタン	説明
ユーザー・プロフィールの改訂	〈ユーザー・プロフィールの改訂〉プログラム(P0092)にアクセスします。プロフィールはシステム管理者のみが変更できます。
ローカル出力の表示	EnterpriseOne ソフトウェアを実行中のマシンの印刷待ち行列フォルダにアクセスします。
投入済みレポート	〈サーバーの処理〉プログラム(P989116)にアクセスします。ここでは、投入済みレポートジョブの状況の確認、優先順位の変更、レポート出力の処理、およびエラーの再表示ができます。
メニューの改訂	タスク・エクスプローラでは使用できません。
パスワードの変更	〈ユーザー・セキュリティ〉プログラム(P98OWSEC)にアクセスすると、パスワードを変更できます。
デフォルト・プリンタ	〈プリンタ〉プログラム(P98616)にアクセスします。デフォルト・プリンタの設定は、システム管理者だけが変更できます。

参照

- ユーザー・プロフィールの変更については『システム・アドミニストレーション』ガイドの「ユーザー・プロフィール」
- PrintQueue ディレクトリへのアクセスについては『基本操作』ガイドの「レポートの出力を表示するには」
- 『エンタープライズ・レポート・ライティング』ガイドの「レポートの投入」
- 『Web アプリケーション設計』ガイドの「パスワードを変更するには」

- デフォルト・プリンタの設定変更については『エンタープライズ・レポート・ライティング』ガイドの「プリンタ定義プログラムの処理」

▶ パスワードを変更するには

1. タスク・エクスプローラで[ユーザー・オプション]を選択します。
2. 〈ユーザー・デフォルトの改訂〉で、[パスワードの変更]をクリックします。
3. 〈ユーザー・パスワードの改訂〉で、次のフィールドに値を入力して[OK]をクリックします。
 - 旧パスワード
 - 新パスワード
 - 新パスワード(確認)

▶ 1 度実行したレポートを再度オンライン表示するには

レポートをオンラインで表示するにはレポート・バージョンを実行してください。

1. タスク・エクスプローラで[ユーザー・オプション]を選択します。
2. 〈ユーザー・デフォルトの改訂〉で、[ローカル出力の表示]をクリックします。
3. [ファイルを開く]で、ファイルを選択して[開く]をクリックします。

PDF 形式のレポートが表示されます。エラー・ログなどの表示ログ・ファイルも表示できます。これを実行するには、[ファイル・タイプ]フィールドに UBE ログ・ファイルを指定します。

参照

- レポートの実行については『エンタープライズ・レポート・ライティング』ガイドの「レポートの投入」

タスク・エクスプローラの設定

システム管理者は、次のタスクを実行してタスク・エクスプローラを設定および管理します。

- Windows のワークステーションにおける Solution Explorer を設定する
- 正しいテーブルを参照するようにタスク・エクスプローラを設定する
- Solution Explorer を使ってロールを作成する
- Solution Explorer を使ってタスク・ビューを作成する
- セキュリティ・ワークベンチを使ってセキュリティを設定する

Windows ユーザーが OneWorld エクスプローラを使ってアプリケーションにアクセスしている場合でも、タスク・エクスプローラのロールやタスク・ビューだけは Solution Explorer で設定できます。システム管理者は、Windows ユーザーが OneWorld エクスプローラを実行している場合、タスク・エクスプローラがさまざまな OneWorld メニュー・テーブル(F008x)を参照するように設定します。Windows ユーザーが Solution Explorer を実行している場合は、タスク・エクスプローラがさまざまな Solution Explorer テーブル(F900x)を参照するように設定します。EnterpriseOne、OneWorld Xe、または ERP 8.0 を使用している場合は、Solution Explorer とその関連テーブルを使用することをお勧めします。システム

管理者が OneWorld® エクスプローラ・テーブルを使用している場合は、ユーザーがアクセスできるタスク・ビューは 1 つ (OneWorld メニュー) だけです。

テーブルを切り替えるために、2 つのテンプレート・ファイル (taskexplorer.properties.owmenu および taskexplorer.properties.activera) が提供されています。

システム管理者は、taskexplorer.properties の内容を taskexplorer.properties.owmenu で一時変更して、OneWorld メニュー・テーブルを参照するように設定します。システム管理者は、taskexplorer.properties の内容を taskexplorer.properties.activera で一時変更して、Solution Explorer テーブルを参照するように設定します。デフォルトでは、taskexplorer.properties は、Solution Explorer テーブルを参照するように設定されています。

ロールベースのタスク・ビューは、Solution Explorer およびタスク・エクスプローラの最も便利なツールの 1 つです。ロールベースのタスク・ビューを設定するには、最初にロールを設定してください。ユーザーの必要に応じて、タスク・ビューを変更することができます。Solution Explorer を使用するとロールやタスク・ビューを設定できます。ロールとタスク・ビューの設定については『Solution Explorer』ガイドを参照してください。これらの機能は、Solution Explorer テーブルに依存するため、OneWorld メニュー・テーブルを参照する場合は、ロールや複数タスク・ビューを利用できません。

セキュリティ・ワークベンチには、Solution Explorer (タスク・エクスプローラも同じ) セキュリティ・オプションに関するフォームがあります。タスク・エクスプローラでは、エクスプローラ、お気に入り、略式コマンド/検索のみ使用できます。ユーザーにアクセス権のないタスク・ビューを表示させることができます。ユーザーにお気に入りリストを表示させたり変更させたりすることもできます。略式コマンド機能にアクセスさせることもできます。([検索] はタスク・エクスプローラでは現在使用できません。) ロールや個々のユーザーには異なるセキュリティを設定することができます。

システムでは、OneWorld テーブルが設定されている場合は UDC タイプの略式コマンド(00/FP)にある UDC コードが参照されます。Solution Explorer テーブルが設定されている場合は UDC タイプ ActivEra 略式コマンド(H90/FP)の UDC コードが参照されます。

参照

- ユーザーにタスク・エクスプローラ・タスク・ビューやお気に入りタスク・ビュー、略式コマンドへのアクセスを許可する方法については『システム・アドミニストレーション』ガイドの「Solution Explorer セキュリティ」
- ロールの作成と管理については『システム・アドミニストレーション』ガイドの「ユーザー・ロールの設定」
- タスク・ビューの作成と変更については『Solution Explorer』ガイドの「タスク・ビューの設定」
- タスク・ビューに含めるタスクの変更とタスクにロールを適用する方法については、『Solution Explorer』ガイドの「タスクの設定」

▶ 正しいテーブルを参照するようにタスク・エクスプローラを設定するには

1. WebSphere の J.D.・Edwards インスタンスを停止します。
2. Web サーバーで、`¥$.¥webclient¥taskexplorer¥`に移動します。

taskexplorer.properties ファイルは、タスク・エクスプローラの実行方法を決定します。これには、使用するテーブル・セットも含まれます。デフォルトでは、タスク・エクスプローラは Solution Explorer テーブルを参照するように設定されています。

注意

下記のように指示された場合を除いて、taskexplorer.properties は変更しないでください。

3. taskexplorer.properties を taskexplorer.properties.def に名前変更します。
4. タスク・エクスプローラで OneWorld メニュー・テーブルを参照するように設定するには、taskexplorer.properties.owmenu を taskexplorer.properties に名前変更します。
5. タスク・エクスプローラで Solution Explorer テーブルを参照するように設定するには、taskexplorer.properties.activera を taskexplorer.properties に名前変更します。
6. WebSphere の EnterpriseOne インスタンスを開始します。

タスク・エクスプローラのトラブル・シューティング

エンド・ユーザーとしてタスク・エクスプローラでの作業中に問題が発生した場合は、IT サポートまたはシステム管理者に連絡してください。

システム管理者は、タスク・エクスプローラが表示されていない場合、次のログを検討してください。

- JAS.log を検討して、すべてのドライバが登録されていることを確認してください。SQL (または同等の) 例外がないか調べて、データベースの接続性にエラーがないかどうか確認してください。
- JAS.ini ファイルで DEBUG=TRUE を設定し、JASDEBUG.log で API アクティビティにエラーがないかどうか確認してください。
- STDOUT.log、STDERR.log、STDOUT.log を検討してください。これらのログには、エントリ・ポイント・フォームのルックアップに関連するエラーなどの、タスク・エクスプローラに固有な情報が含まれています。

ログのロケーションは JAS.ini ファイルで指定されます。

J.D.・Edwards Web アプリケーション/レポート

J.D.・Edwards では、さまざまなアプリケーション、レポート、オブジェクトを提供しています。通常、これらのオブジェクトにはタスク・エクスプローラからアクセスします。

▶ アプリケーションまたはレポートを実行するには

1. タスク・エクスプローラで、実行するアプリケーションまたはレポートに移動します。
2. 処理オプションやバージョンを定義せずにアプリケーションやレポートを実行するには、レポートまたはアプリケーションをクリックします。

アプリケーションが実行されます。バージョンを選択せずにレポートを実行すると、〈バッチ・バージョンの処理〉が表示され、実行するバージョンを選択できます。

3. アプリケーションの処理オプションやバージョンを選択するには、タスクの横にある緑色の矢印にカーソルを置いて、[処理オプション...]または[バージョン...]を選択してください。

メニュー・オプションが斜体で表示されている場合は、アプリケーションにはそのオプションが使用できないことを表します。

処理オプションまたはバージョンを選択すると、アプリケーションが実行されます。

4. 処理オプションやバージョンを選択したり、レポートのデータ選択パラメータを指定するには、タスクの横にある緑色の矢印の上にカーソルを置いて、次のオプションのうち 1 つを選択してください。

- 処理オプション
- バージョン
- データ選択
- データ順序

メニュー・オプションが斜体で表示されている場合は、レポートにはそのオプションが使用できないことを表します。

オプションを設定すると、〈バッチ・バージョンの処理〉が表示され、実行するバージョンを選択できます。次に、〈バージョン・プロンプト〉フォームが表示されます。プロンプトを選択して [投入] をクリックし、プリンタを選択してレポートを処理してください。

参照

- アプリケーションの実行については『基本操作』ガイドの「対話型バージョン・アプリケーション」
- レポートの実行については『基本操作』ガイドの「レポート用バッチ・バージョン」
- 処理オプションの使用については『基本操作』ガイドの「処理オプション」

アプリケーション・ショートカット

Web クライアントでの作業中に、使用しているアプリケーションやフォームのショートカットを送信することができます。受信者は、電子メールのショートカットをダブルクリックすると、そのアプリケーションやフォームにアクセスできます。

注:

ショートカットを送信して Windows および Web クライアントのアプリケーションを起動することができます。ショートカットを送信する前に、ショートカットの受信者が Windows または Web のどちらのクライアント・バージョンを使用しているか確認してください。ただし、ショートカットの優先情報は Windows クライアント上でのみ設定できます。

▶ アプリケーション・フォームにショートカットを送信するには

1. Web アプリケーションを起動して送信するフォームにアクセスします。
2. [ツール] をクリックして [ショートカットの送信] を選択します。
3. 〈ショートカットの送信〉で、次のフィールドに値を入力します。
 - 住所番号/ユーザー/ロール/配布リスト

注:

配布リストに含まれているユーザーにショートカットを送信する場合は、[配布リスト]オプションをクリックして配布リストにある住所番号を選択します。[配布リスト]オプションを選択せずに住所番号を入力すると、ショートカットは配布リストの住所番号に送られるだけでリストのユーザーには送信されません。

- メールボックス
メッセージを送信するメールボックス/待ち行列を選択します。
 - 件名
[件名]行に件名を入力します。
4. メッセージにショートカットを添付する場合、フォームの一番下にある大きなフィールドに入力してください。
 5. [OK]をクリックし、ショートカットを送信します。
受信者は、ワーク・センターまたはサードパーティの電子メールを通して、ショートカットを受信します。

Web アプリケーション・フォーム

J.D.・Edwards ERP アプリケーションでは対話型のフォームを使用できます。アプリケーションを実行すると、デフォルトのフォームが表示されます。必要に応じてその他のフォームが表示されます。一度に表示されるフォームは1つだけです。フォーム間を移動する場合は、ブラウザの[戻る]([進む])ボタンではなくフォームのボタンを使用してください。

J.D.・Edwards のフォームは次のコンポーネントから構成されています。

- タイトル・バー
- ボタン
- ラジオ・ボタン/チェックボックス
- ハイパーリンク
- タブ
- フィールド
- グリッド
- ツリー・コントロール

タイトル・バー

タイトル・バーはアプリケーションの一番上に表示されます。左側にはアプリケーションの名前を表示し、右側にはヘルプ・ボタンが3つあります。これらのボタンを押すと、アプリケーションのバージョン・フォーム、フォームやアプリケーションのオンライン・ヘルプ、各フォーム・コンポーネントのオンライン・ヘルプにアクセスできます。

ボタン/メニュー

ボタンは、ツールバーとタイトル・バーの両方にあります。フォームの中央に表示されることもあります。ボタンをクリックすると、フォームに入力した情報を処理してフォームを閉じたり、その他のプログラムを実行するなどの操作が行われます。

右下のボタンにある矢印は、ボタンがメニューであることを示しています。ボタンをクリックすると、メニューが表示されます。

通常、フォームに入力した情報は、ボタン([OK]や[投入])をクリックするまで処理されません。

主な操作ボタンには次のものがあります。フォームによっては表示されないボタンもあります。

- 選択 - グリッドでローを選んで[Select]をクリックすると、そのレコードに対して、詳細フォームを含む別のフォームを開く、レポートを起動するなどの特定のアクションが実行されます。
- 検索 - 検索基準のあるフォームでフィールドに値を入力したら、[検索]をクリックして検索結果を表示するグリッドに値を入力します。
- 追加 - [追加]をクリックしてテーブルに新しいローを追加します。通常、新しいフォームが表示されてレコードを追加できるようになります。
- 削除 - 削除するレコードをグリッドで選択し、[削除]をクリックしてテーブルから削除します。
- コピー - コピーするレコードをグリッドで選択し、[コピー]をクリックして選択したレコードに基づいて新しいレコードを作成します。
- OK - フォームのフィールドに値を入力したら、[OK]をクリックして情報を処理します。
- キャンセル - [キャンセル]をクリックすると、フォームが閉じます。他の操作ボタン([OK]や[検索])ではなく[キャンセル]をクリックすると、フォームに入力したデータは破棄されます。
- 閉じる - [閉じる]をクリックすると、フォームが閉じます。アプリケーションのデフォルトのフォームで作業中の場合は、[閉じる]ボタンをクリックするとアプリケーションが閉じます。

主な操作メニューには次のものがあります。フォームによっては表示されないメニューもあります。

- ロー - 選択したグリッドのレコードに適用するオプションが含まれます。
- フォーム - フォーム固有のオプション(例:フォームの添付ファイルの表示)の他にも、アプリケーション内での別のフォームへのリンクを含みます。
- ツール - J.D.Edwards のほとんどのアプリケーションに標準のオプションが含まれています。たとえば、ユーザー・オプション、ジョブやレポートの投入機能、グリッドの内容のスプレッドシートへのエクスポートなどがあります。

フィールド/グリッド

フィールドとグリッドにデータを入力すると、データが表示されます。グリッドはテーブルのような外観で、いくつかのカラムに情報が表示されます。各ローには異なるレコードが表示されます。フィールドには一度に1つのデータしか表示されません。使用不可(グレー表示されている)フィールドでは、表示されるデータを変更できません。また、ドロップダウンの矢印があるフィールドもあります。この矢印をクリックすると、選択するオプションが表示されます。

フィールドをクリックすると、フィールドの右側にボタンが表示されることがあります。ボタンのアイコンにはフラッシュライト、電卓、カレンダーがあります。このボタンはビジュアル・アシストと呼ばれます。ビジュアル・アシストをクリックすると、フィールドに入力する値を検索することができます。また、システム情報を検索するフォームが表示されます。たとえばシステムにログインすると、選択する環境がいくつか表示されます。ビジュアル・アシストをクリックしてリストから環境を選択すれば、手作業の選択

や入力によるミスを防ぐことができます。電卓の絵の場合は、値を計算してフィールドに結果を入力してくれます。カレンダーの絵の場合は、カレンダーから日付を選んでフィールドに入力してくれます。

フォームによっては、すべてのフィールドに入力しなくてもよい場合があります。たとえば多くのフォームでは、情報を検索することができ、フィールドに入力したデータを使って検索を絞り込んでいきます。特定のフィールドに基づいて検索を絞り込まない場合は、フィールドに"*"と入力します。*は、そのフィールドに対してすべての値が有効であることを示します。

ツリー・コントロール

ツリー・コントロールは 2 つの画面のように見えます。片方は、ファイルやアプリケーションなど、オブジェクトの階層構造を表示します。もう片方は、階層のオブジェクトに応じて表示されます。

▶ フォームでヘルプを表示するには

すべてのヘルプ・ボタンは J.D.・Edwards Web フォームでは右上の隅に表示されます。ボタンの上にカーソルを置くと、名前が表示されます。

1. ID やアプリケーション、ソフトウェア・バージョンなどの情報を表示するには、[J.D.・Edwards 情報]をクリックしてください。
2. 現在のアプリケーションの使用方法については、[ヘルプ]をクリックしてください。
3. 特定のフィールドについての情報を表示するには、[項目ヘルプ]をクリックしてください。
カーソルの横に疑問符マーク(?)が表示されます。
4. フィールドをクリックすると情報が表示されます。
フィールドのどの番号もクリックできます。
5. 終了したら、[項目ヘルプ]を再度クリックしてください。
カーソルの横の疑問符が表示されなくなったら、フォームの使用を続行できます。

Web アプリケーション・フォームでのドラッグ・アンド・ドロップ

親/子フォームには、ドラッグ・アンド・ドロップ機能を使用してフォーム内で項目を並べ替える機能が用意されています。たとえば、〈ビジネスユニット構造ツリー表示〉プログラム(P0006A)では、ビジネスユニットを別の組織レベルにドラッグ・アンド・ドロップして組織構造を改訂することができます。

▶ Web の親/子フォームで項目をドラッグ・アンド・ドロップするには

1. 〈ビジネスユニット構造ツリー表示〉プログラム(P0006A)などの親/子フォームで、移動するフォルダまたは項目のノードをクリックします。
2. [ドラッグ]ボタンをクリックします。
移動するノードの隣にカーソル・アイコンが表示されます。
3. 移動するフォルダまたは項目のノードをクリックしてドラッグ項目を置きます。
4. [移動]ボタンをクリックします。
ドラッグ項目が移動先フォルダに配置されます。

参照

- 親/子フォームの機能については『基本操作』ガイドの「親/子フォーム」

グリッド

多くのフォームにはグリッドがあります。グリッドにはデータが表示されます。アプリケーションによっては、グリッドのデータを追加、変更、削除することができます。

グリッドのあるアプリケーションを実行すると、最初はグリッドは空です。検索を実行してグリッドに値を入力していきます。検索を実行するには、[検索]をクリックします。ほとんどの場合、検索基準を設定せずに[検索]をクリックすると、ビジネス・ビューのデータをすべて表示するものと仮定されます。最初に検索基準を指定しないと検索を実行できない場合もあります。

J.D. Edwards では、さまざまな検索基準の定義を提供しています。ほとんどのグリッドには、カラム見出しの上にブランクのフィールド付きのローがあります。このローは、QBE と呼ばれています。QBE フィールドにデータを入力してから[検索]をクリックすると、入力したデータに基づいて検索が絞られます。たとえば、従業員情報をリストするアプリケーションでは、[名字]カラムでフィールドに“Abbot”と入力すると、システムからは名字が Abbot という従業員のみ返されます。他のカラムにデータを入力してさらに検索を絞ることもできます。グリッドに「市」というカラムがある場合、市の名前を入力すると、名字が Abbot で、指定した市に住む従業員のデータが返されます。

QBE の他にも、多くのフォームにはグリッドの上にフィールドが含まれています。これらのフィールドにデータを入力すると、検索を絞ることができます。さらに、ラジオ・ボタンやチェックボックスを使って検索を絞ることのできるフォームもあります。たとえば受注オーダーを処理する場合、フォームのチェックボックスを選択すると、未完了のオーダーを除外することができます。

QBE とフィールドでは、通常ワイルドカード(*)を使用できます。最初の例で、名字の最初の文字が M の従業員をすべて表示するには、[名字]カラムに“M*”と入力します。また、多くのフィールドには、フラッシュライト、電卓、またはカレンダーなどのビジュアル・アシストが組み込まれています。ビジュアル・アシストをクリックすると、有効な値を検索したり計算したりしてくれます。

どの場合にも、検索基準を追加することができます。つまり、QBE およびグリッドの上のフィールドに検索基準を入力してその他のオプションを選択すると、これらすべての検索基準に基づいて値が返されます。フィールドが無効の(グレー表示されている)場合は、検索を絞り込むことができません。

システム管理者の設定によって、グリッドには一度に数件のデータしか取り込まれないこともあります。現在表示されているよりも多くのデータが存在する場合、グリッドの一番上にロー・カウンタと上下矢印が表示されます。上下矢印を使うと、さらにデータが取り込まれたり、前に取り込まれたグリッドの設定に戻ることができます。カウンタにより、リストのどのレコードを参照しているかがわかります。たとえば、最初にグリッドを読み込むと、カウンタには「レコード 1 - 10」と表示されます。下矢印をクリックすると、次の 10 件が読み込まれ、「レコード 11 - 20」と表示されます。

グリッドの特定のローを処理するには、ローの左にあるラジオボタンまたはチェックボックスを選択してください。グリッドのローがハイライトされ、選択されていることがわかります。ローをクリックしてデータを処理するためのオプションのリストを表示します。ローの添付も表示します。グリッドでチェックボックスまたはラジオ・ボタンが使用されている場合、複数のローを選択できます。ただしその場合、1 つのローを選択する場合と異なり、[ロー]メニューでは同じオプションを選ぶことができない場合があります。

グリッドにはデータのみを表示するものもあります。データを入力できるグリッドもありますが、別のフォームを表示する必要があります。ただし、グリッドが小さい場合は、グリッドにデータを直接入力できる場合もあります。この場合グリッドには、データを手作業で読み込むか、Microsoft Excel スプレッドシートの内容をインポートして読み込むことができます。このようにするためには、スプレッドシー

トで指定した範囲がグリッドのカラムと完全一致する必要があります。ほとんどのグリッドでは、レコードを Microsoft Excel または Word ファイルにエクスポートすることができます。

グリッドは外観もカスタマイズできます。グリッドを最大化して画面のほとんどに表示することもできます。カラムの色も変更でき、カラム・テキストに色やフォーマットを適用することもできます。カラムの表示順序やソート順序、カラム幅も定義できます。1つのフォームに複数のグリッド・フォーマットを作成して異なる表示方法でデータを表示することもできます。

▶ **グリッドの内容を Microsoft Excel や Word にエクスポートするには**

1. グリッドのあるアプリケーションを起動して、[検索]をクリックしてグリッドのレコードを読み込みます。
2. 次のいずれかを実行します。
 - グリッドのレコードを Excel にエクスポートするには、[ツール]をクリックして[Excel にエクスポート]をクリックしてください。
 - グリッドのレコードを Word にエクスポートするには、[ツール]をクリックして[Word にエクスポート]をクリックしてください。
3. エクスポート・アシスタントを使って、範囲の最初または最後のセルをクリックし、エクスポートするデータの範囲を指定します。

たとえば、グリッドに4つのカラムと3つのローがあり、最初の3つのカラムと最初の2つのローのみエクスポートする場合、最初のローの最初のセルをクリックしてから2番目のローにある3番目のセルをクリックします。

セルをビューに読み込むには、グリッドのスクロール・バーを使用してください。一度に読み込まれるローは数件です。グリッドの一番上にある上下矢印を使うと、さらにローのデータが取り込まれたり、前に取り込まれたローの設定に戻ることができます。

違うセルをクリックした場合は、[選択のリセット]をクリックしてください。

4. [続行]をクリックします。
選択したグリッドの内容が、適切なファイル・タイプにエクスポートされて表示されます。

▶ **Microsoft Excel スプレッドシートをグリッドにインポートするには**

1. データを直接入力できるグリッドのあるアプリケーションを実行します。
2. [ツール]をクリックして[Excel からインポート]を選択します。
3. [Excel からのインポート]で、インポートする Excel スプレッドシートを検索するには[検索]をクリックします。
4. 次のフィールドに入力します。
 - インポートするワークシート
インポートするデータを含むワークシートの名前を入力します。
 - 開始セル・カラム
インポートするデータの範囲の最初のセルのカラム文字を入力します。
 - 開始セル・ロー

インポートするデータの範囲の最初のセルのロー番号を入力します。

- 終了セル・カラム

インポートするデータの範囲の最後のセルのカラム文字を入力します。

- 終了セル・ロー

インポートするデータの範囲の最後のセルのロー番号を入力します。

5. [インポート]をクリックします。

選択したデータがインポートされます。

▶ グリッド・フォーマットを作成するには

デフォルトのフォーマットを再び使用するには、新規フォーマットを保存する前に変更前のグリッドを保存しておきます。変更前に保存しなかった場合は、新規フォーマットを削除し、アプリケーションを終了してアプリケーションに再びアクセスすると、デフォルトのグリッドが表示されます。

1. 新しいグリッド・フォーマットを作成するアプリケーションを実行して、[グリッドのカスタマイズ]をクリックしてください。
2. [グリッド・フォーマットの選択]で、[作成]をクリックします。
3. [グリッド・フォーマット名]で、フォーマットの名前を入力します。
4. 必要に応じて残りのオプションにも値を入力して、[OK]をクリックします。
5. モバイル・デバイスのデフォルト・フォーマットとしてグリッド・フォーマットを使用するには、作成したフォーマットをクリックして[モバイル・デバイスのデフォルト]を選択します。
6. [閉じる]をクリックします。

▶ モバイル・デバイスのグリッド・フォーマットを作成するには

1. Web クライアントを使用して、モバイル・デバイスのグリッド・フォーマットを作成するアプリケーションを実行します。
2. [グリッドのカスタマイズ]をクリックして、モバイル・デバイスに使用するフォーマットを選択します。

フォーマットをまだ作成していない場合は、ここで作成してください。Web ベースのフォーマットを作成する場合にも、同じステップに従ってください。ただし、表示するカラムやカラム幅、カラムやテキストの色などを指定する場合に、ほとんどのモバイル・デバイスで提供されるスペースや色オプションは限られています。

3. [モバイル・デバイスのデフォルト]をクリックして[閉じる]をクリックしてください。

▶ グリッド・フォーマットを適用するには

グリッドのあるフォームで、[グリッドのカスタマイズ]の横にあるドロップダウン・リストからグリッドを選択してください。

グリッド・フォーマットを適用する前に、グリッドを作成しておいてください。

▶ グリッド・フォーマットを変更するには

1. 変更するグリッド・フォーマットを含むアプリケーションを実行して、[グリッドのカスタマイズ]をクリックします。
2. [グリッド・フォーマットの選択]で、変更するグリッド・フォーマットを選択して[変更]をクリックしてください。
3. グリッド・フォーマットの名前を変更するには、[グリッド・フォーマット名]に新しい名前を入力します。
4. 必要に応じてグリッドの要素を変更し、[OK]をクリックします。
5. [閉じる]をクリックします。

▶ グリッド・フォーマットを削除するには

1. グリッドのあるフォームで、[グリッドのカスタマイズ]をクリックします。
2. [グリッド・フォーマットの選択]で、削除するグリッド・フォーマットを選択して[削除]をクリックしてください。
グリッド・フォーマットがリストから表示されなくなります。
3. [閉じる]をクリックします。

▶ グリッド・カラムを表示/非表示するには

1. 変更するグリッドを含むアプリケーションを実行します。
2. [グリッドのカスタマイズ]をクリックして、新規フォーマットを作成するか、変更する既存のフォーマットを選択します。
3. [グリッドのカスタマイズ]で、[表示および順序]セクションまでスクロールします。
4. グリッドにカラムを表示しないようにするには、[表示および順序]リストを選択して、左矢印をクリックします。
5. グリッドにカラムを表示させるには、[使用可能なカラム]リストからカラムを選択して右矢印をクリックします。
6. 上下矢印を使用して、グリッドでカラムを表示する順序を変更します。
7. 完了したら、[OK]をクリックして[閉じる]をクリックします。

▶ グリッド・カラムを再設定するには

1. 変更するグリッドを含むアプリケーションを実行します。
2. [グリッドのカスタマイズ]をクリックして、新規フォーマットを作成するか、変更する既存のフォーマットを選択します。
3. [グリッドのカスタマイズ]で、[表示および順序]セクションまでスクロールします。
4. [表示および順序]リストで、カラム名をクリックして上下矢印を使用してリストを上下に移動します。

リストではカラムは上から下に表示されます。つまり、リストの1番目のカラムはグリッドで最初に表示されます。2番目のカラムは2番目に表示されます。

5. ステップ 4 を繰り返して、他のカラムも移動してください。
6. 完了したら、[OK]をクリックして[閉じる]をクリックします。

▶ グリッドの色とフォントを設定するには

1. 変更するグリッドを含むアプリケーションを実行します。
2. [グリッドのカスタマイズ]をクリックして、新規フォーマットを作成するか、変更する既存のフォーマットを選択します。
3. [グリッドのカスタマイズ]で、[表示および順序]セクションまでスクロールします。
4. [表示および順序]リストでカラム名をクリックします。
[選択したカラム]フィールドにカラム名が表示されます。
5. 背景色をカラムに適用するには、[カラムの色]の下にあるパレットで色をクリックしてください。
選択した色の 16 進数の値が[カラムの色]フィールドに表示されます。
6. 色をカラムのテキストに適用するには、[テキストの色]の下にあるパレットで色をクリックしてください。
選択した色の 16 進数の値が[テキストの色]フィールドに表示されます。
7. カラムのテキストにフォント・スタイル(太字や斜体)を適用するには、適用するスタイルを[テキスト・オプション]リストでクリックしてください。
8. [スタイルの更新]をクリックします。
[選択したカラム]フィールドが更新され、どのように見えるか示されます。[表示および順序]リストでは、カラム名の横に「+」が表示されます。ユーザー定義フォーマットがカラムに適用されることを示します。
9. ステップの 4 から 8 を繰り返して、その他のカラムのフォーマットにも適用します。
10. 完了したら、[OK]をクリックして[閉じる]をクリックします。

▶ グリッド・カラム幅を変更するには

1. 変更するグリッドを含むアプリケーションを実行します。
2. [グリッドのカスタマイズ]をクリックして、新規フォーマットを作成するか、変更する既存のフォーマットを選択します。
3. [グリッドのカスタマイズ]で、[表示および順序]セクションまでスクロールします。
4. [表示および順序]リストでカラム名をクリックします。
[選択したカラム]フィールドにカラム名が表示されます。
5. [カラムの幅]フィールドに数値(%)を入力します。
この値はカラムに割り当てるスペースの割合です。カラムが基づくデータ項目に定義された幅に基いて割り当てられます。入力できる値は 25 から 400 までです。
6. 完了したら、[OK]をクリックして[閉じる]をクリックします。

▶ グリッドのソート順序を変更するには

1. 変更するグリッドを含むアプリケーションを実行します。
2. [グリッドのカスタマイズ]をクリックして、新規フォーマットを作成するか、変更する既存のフォーマットを選択します。
3. [グリッドのカスタマイズ]で、[データ順序設定]セクションまでスクロールします。
4. カラムをソートする場合、[使用可能なカラム]でカラム名をクリックして右矢印をクリックします。
[使用可能なカラム]から[順序づけられたカラム]にカラム名が移動されます。

カラムはグリッドに含まれる必要があります。カラム名はソートする前に[表示および順序]に表示されていないとなりません。

5. カラムをソートする場合、[順序づけられたカラム]でカラム名をクリックして左矢印をクリックします。
[順序づけられたカラム]から[使用可能なカラム]にカラム名が移動されます。
6. ソート順序を再編成するには、[順序づけられたカラム]の下にある上下矢印を使用して、カラム名を再編成してください。

最初にリストの一番上のカラムでソートされ、上から2番目のカラムで次にソートされます。

7. カラムを昇順でソートする場合は、[順序づけられたカラム]で名前をクリックして[昇順]をクリックします。
[昇順]を空白のままにすると、カラムは降順でソートされます。

昇順でソートされるカラム名の横に A という文字が表示されます。降順でソートされるカラム名の横には D という文字が表示されます。

8. 完了したら、[OK]をクリックして[閉じる]をクリックします。

メディア・オブジェクト添付

メディア・オブジェクトおよびイメージ機能を使用すると、現在紙ベースで配布している情報も含め、アプリケーションに情報を添付することができます。たとえば、テキストを使用してある仕訳入力に関する特殊な状況を説明することができます。メディア・オブジェクト機能を使用すると、EnterpriseOne アプリケーション、フォームやロー、およびオブジェクト・ライブラリアンのオブジェクトに情報を添付することができます。メディア・オブジェクトで使用できるイメージ機能により、さらに効率よく情報を保存することができます。

また、OneWorld 中のグリッドのローやフォーム自体にオブジェクトをリンクさせることができます。グリッドのローやフォームに添付できるオブジェクトには次のタイプがあります。

テキスト メディア・オブジェクトには文書作成機能があり、添付を作成することができます。たとえば、テキスト添付により、フォームの記入方法やレコードに関する追加情報を添付することができます。

イメージ Windows のイメージ・ファイルには、BMP、.GIF、およびJPG があります。このようなファイルは、電子ファイルから作成された場合と紙の伝票をスキャナで入力した画像の場合があります。添付できるイメージは、システム管理者により最初にイメージ待ち行列に添付されます。

OLE メディア・オブジェクトには、OLE 標準に合ったファイルを使用できます。OLE を使用すると、異なるプログラム間でのリンクを作成できます。このリンクを使用して、あるプログラムのオブジェクトを別のプログラムに保存することができます。J.D.・Edwards のソフトウェアは、OLE オブジェクトを添付するのに必要なリンクを提供します。

OLE メディア・オブジェクトは、ベース・フォーム・レベルで添付します。フォームのベース・レベルにオブジェクトを添付する際には、オブジェクトはフォーム上に表示される 1 データではなくフォーム自体に添付されます。OLE ファイルはグリッドのローまたはフォームにリンクすることができますが、ファイル自体は別のディレクトリに保存されます。OLE がリンクするアプリケーションについて持つファイル情報は、リンクするファイルへのパス情報だけです。

使用できるのは、OLE オブジェクトとして正しく登録してインストールしたものに限りです。

URL/ファイル メディア・オブジェクトとして、Web ページの URL またはその他のファイルへのリンクを作成することもできます。フォームで URL メディア・オブジェクトをコントロール・オブジェクトに添付すると、その Web ページがフォームの一部として表示されます。URL をフォームまたはオブジェクト・ライブラリアンのオブジェクトに添付すると、そのメディア・オブジェクトはその URL へのリンクとなります。ファイルはイメージ待ち行列に常駐しているものを使用することも、ローカルやネットワーク・ファイルを添付することもできます。

メディア・オブジェクトをフォームに添付してフォーム上で異なるデータにアクセスする場合、添付ファイルが使用できないことがあります。たとえば、オーダー番号 2002 に関するデータを含む詳細フォームにメディア・オブジェクトを添付すると、オーダー番号 3003 のデータにアクセスしたときに表示される詳細フォームには、この添付ファイルは表示されません。ベース・フォーム(上記の場合は、詳細フォーム)は同じでも、フォームに関連付けられているデータのオーダー番号により異なるためです。オーダー番号は添付ファイルが保管されているロケーションへのキーになっています。

フォーム・レベルでファイルが添付されているフォームには、ステータス・バーの右にペーパークリップ・アイコンが表示されます。ベース・フォーム・レベルで OLE オブジェクトが添付されたフォームでは、ステータス・バーの右に文書アイコンで表示されます。

ユーザーが検索を行ってグリッド・ローにレコードを表示させた時点では、そのレコードに対して添付ファイルが存在するかどうかは表示されません。レコードに添付ファイルがあるかどうかを確認するには、添付を検索する必要があります。システムでは、現在グリッドにロードされているレコードの添付のみ検索されます。[検索]ボタンをクリックすると、フォームにより添付ファイルの参照状況が再設定されます。グリッド・レコードの添付を表示するには、再度検索を実行してください。

メディア・オブジェクト・ビューアを使用するとオブジェクトに既に添付されているメディア・オブジェクトを調べたり、新しいメディア・オブジェクトを追加したりできます。メディア・オブジェクトのワークスペース・ビューアは、2 つのパネルに分割されています。左のパネルはアイコン・パネルで、右のパネルはビューア・パネルです。以前にレコードに添付されたファイルはアイコン・パネルに表示されます。添付を表示するには、添付アイコンをクリックすると、ビューア・パネルに内容が表示されます。オブジェクトには、複数の添付を含めることができます。

テンプレートを使用すると、頻繁に使用するメディア・オブジェクト用のフォーマットを定義しておくことができます。テンプレートにはイメージやショートカットなど、添付オブジェクトを含むこともできます。たとえば、メモ用のレターヘッドや標準フォームを作成することができます。テンプレートはシステム管理者によって作成されます。

参照

- Web クライアントでメディア・オブジェクトを添付する方法については『Web アプリケーション設計』ガイドの「メディア・オブジェクトの添付」
- テキスト・テンプレートの作成については『基本操作』ガイドの「テンプレートの処理」

- メディア・オブジェクトを処理するためのフォームの設定については『Form Design Aid(フォーム設計ツール)』ガイドの「Processing Media Objects(メディア・オブジェクトの処理)」
- メディア・オブジェクトを使用可能にする方法については『システム・アドミニストレーション』ガイドの「メディア・オブジェクトおよびイメージング」

▶ **添付を表示するには**

1. グリッドのあるアプリケーションを実行して、[検索]をクリックしてグリッドにレコードを入力します。
2. 拡大鏡とペーパークリップのあるボタンをクリックします。(ボタンはカラム見出しの左にあります。)添付ファイルのあるレコードのローには、グリッドの左にペーパークリップが表示されます。リフレッシュや検索、前のレコード設定のビューに戻る場合は、そのつど添付を検索する必要があります。
3. ペーパークリップ・アイコンの付いたボタンをクリックして、ローの添付を表示します。
4. メディア・オブジェクト・ビューアで、表示するメディア・オブジェクトに対応する左カラムにあるオブジェクトをクリックします。
5. 完了したら[キャンセル]をクリックします。

▶ **テキストを添付するには**

1. グリッドのあるアプリケーションを実行して、[検索]をクリックします。
2. 添付を追加するローで、グリッドの左にあるボタンをクリックします。
3. <メディア・オブジェクト>で、[テキスト]をクリックします。
4. ビューア・パネルでテキストを入力します。ビューア・パネルの上部にあるフォーマット・ツールを使用すると、テキストのフォーマットを調整できます。
5. [戻る]をクリックすると、アプリケーションに戻ります。

▶ **テンプレートを使用してテキストを添付するには**

1. グリッドのあるアプリケーションを実行して、[検索]をクリックします。
2. 添付を追加するローで、グリッドの左にあるボタンをクリックします。
3. <メディア・オブジェクト>で、[テキスト]をクリックします。システムにより、新しいブランクのテキスト・オブジェクトが作成されます。
4. [テンプレート]をクリックします。
5. <メディア・オブジェクト・テンプレートの処理>で、[検索]をクリックしてグリッドを読み込みます。
6. 使用するテンプレートを選択して[選択]をクリックします。テンプレートがメディア・オブジェクトのワークスペースに表示されます。
7. ビューア・パネルでテキストを入力します。

ビューア・パネルの上部にあるフォーマット・ツールを使用すると、注テキストのフォーマットを調整できます。

8. アプリケーションに戻るには、[戻る]をクリックします。

▶ **イメージを添付するには**

1. グリッドのあるアプリケーションを実行して、[検索]をクリックします。
2. 添付を追加するローで、グリッドの左にあるボタンをクリックします。
3. 〈メディア・オブジェクト〉で、[イメージ]をクリックします。
4. プルダウン・メニューから[イメージ]を選択します。
5. 添付するイメージをクリックします。
イメージをプレビューするには、[プレビュー]をクリックしてイメージをクリックします。
6. [追加]をクリックします。
7. アプリケーションに戻るには、[戻る]をクリックします。

▶ **URL またはファイルを添付するには**

1. グリッドのあるアプリケーションを実行して、[検索]をクリックします。
2. 添付を追加するローで、グリッドの左にあるボタンをクリックします。
3. 〈メディア・オブジェクト〉で、[URL/ファイル]をクリックします。
4. イメージ待ち行列に常駐しているファイルを添付するには、次の手順に従ってください。
 - a. [待ち行列]をクリックします。
 - b. プルダウン・メニューから[イメージ]を選択します。
 - c. 添付するファイルをクリックします。
ファイルをプレビューするには、[プレビュー]をクリックしてファイルをクリックします。
5. 別のロケーションまたは URL にあるファイルを添付するには、次の手順に従ってください。
 - a. [URL/ファイル]を選択します。
 - b. URL 名またはファイル名とロケーションを入力して、[参照]をクリックしてファイルを検索します。
 - c. ファイルをプレビューするには、[URL/ファイルのプレビュー]をクリックしてください。
6. [追加]をクリックします。
7. アプリケーションに戻るには、[戻る]をクリックします。

▶ **OLE オブジェクトを添付するには**

1. グリッドのあるアプリケーションを実行して、[検索]をクリックします。
2. 添付を追加するローで、グリッドの左にあるボタンをクリックします。
3. 〈メディア・オブジェクト〉で、[OLE]をクリックします。

4. 新規オブジェクトを作成するには、〈オブジェクトの挿入〉で[オブジェクトの種類]を1つ選択します。

ワークステーションまたはネットワークにインストールされている機能によって、システムごとに選択項目が異なります。

5. オブジェクトを作成します。
6. 既存のオブジェクトを添付するには、[ファイルから作成]を選択して、システムから添付するオブジェクトを検索して[OK]をクリックします。

添付オブジェクトを新規に作成するか既存のオブジェクトを添付するかによって、オブジェクトに関連付けられたアプリケーションがビューア・パネルに表示されるときに、空白のワークスペースが表示されるか既存のオブジェクトが表示されるかという違いがあります。

メニュー・バーには、埋め込まれたアプリケーションのメニュー・バーが表示されます。たとえば Excel ワークシートを選択する場合、メニュー・バーに Excel 用メニューが表示されます。

7. アプリケーションに戻るには、[戻る]をクリックします。

▶ 添付ファイルを削除するには

注:

オブジェクトから添付を削除する場合は、オブジェクトとメディア・オブジェクトとの接続を解除します。オブジェクトの添付を削除しても、メディア・オブジェクトそのものは削除されません。

1. グリッドのあるアプリケーションを実行して、[検索]をクリックしてグリッドにレコードを入力します。
2. 拡大鏡とペーパークリップのあるボタンとをクリックします。(ボタンはカラム見出しの左にあります。)

添付ファイルのあるレコードのローには、グリッドの左にペーパークリップが表示されます。

リフレッシュや検索、前のレコード設定のビューに戻る場合は、そのつど添付を検索する必要があります。

3. ペーパークリップ・アイコンの付いたボタンをクリックして、ローの添付を表示します。
4. メディア・オブジェクト・ビューアで、削除するメディア・オブジェクトに対応する左カラムにあるオブジェクトをクリックします。

5. [削除]をクリックします。
メディア・オブジェクトのアイコンが表示されなくなります。

6. 完了したら[キャンセル]をクリックします。

Web アプリケーション設計

この項では、J.D.・Edwards ツールセットを使用して Web クライアントで使用する EnterpriseOne のアプリケーションを設計および生成する方法について説明します。さらに、HTML クライアントと Windows クライアントの設計上の類似点と相違点、より便利な Web アプリケーションの設計に役立つ豊富な実務上のガイドライン、ツール、およびテクニックについても説明します。この項は、アプリケーション開発者を対象としています。

ここで説明するガイドラインは、弊社社員、ビジネス・パートナー、およびカスタマーを対象としています。

参照

- 『開発ツール』ガイド
- 『開発スタンダード: アプリケーション設計』ガイド
- 『J.D.・Edwards Portal (J.D.・Edwards ポータル)』ガイド

ユーザー中心の設計ガイドライン

使用しやすい設計を行うには、ユーザー中心の設計(UCD)テクニックに従う必要があります。ユーザー中心の設計とは、初期段階から実際のユーザーと頻繁に対話する機会を設けることによって、そのフィードバックや情報をアプリケーション設計に反映させることを意味します。アプリケーション開発を開始する前に、ユーザーと共に概念上、経験上から設計を見直すことを通じて、目標や業務目的を明確化し、検証する必要があります。また、チームのメンバー全員が、その対象ユーザーと、各ユーザーがアプリケーションに期待している成果が何であるかを理解する必要があります。

ユーザー分析チェックリスト

- ユーザーの定義
- ユーザーの目標と業務目標の定義
- ユーザー・タスクの定義
- 使用事例の作成
- 有用性評価の実施

ユーザーの定義

ユーザーの処理能力を拡張するアプリケーションを設計する前に、対象となるユーザーを把握する必要があります。したがって、アプリケーションを使用する予定のユーザー・タイプ(エンド・ユーザー、パワー・ユーザーなど)ごとに、ユーザー・プロファイルを開発します。ユーザー・プロファイル情報は、Web での調査、インタビュー、タスク分析、現状調査、フォーカス・グループおよび市場や競合他社の分析など、さまざまなソースから収集できます。主として、次のような質問を設ける必要があります。

- 対象ユーザー(B2B、あるいは電子的協業を行う仕入先や購買担当者など)
- 対象ユーザーの役職(電子調達管理者、購買担当者など)
- 対象ユーザーのスキル・レベル(初級の Web ユーザー、サプライチェーン・ビジネスのエキスパートなど)

- 対象ユーザー固有のニーズ(リモートでの使用、モバイルでの使用など)
- 対象ユーザーが使用するコンピュータのタイプ(デスクトップ、携帯機器、画面解像度やサイズ、Web ブラウザなど)
- 対象ユーザーの作業環境(製造現場、小部屋、オフィスなど)

ユーザーの目標と業務目標の定義

目標からタスク、アクションに至るまでのユーザーの動きを把握する必要があります。単に「短時間で正確に時間入力を行う」ことを目標としているユーザーもいれば、増収やサポート提供コストの削減などを業務目標としていることも考えられます。製品設計で成功を収めるには、ユーザーと業務の目標を両方とも把握する必要があります。ユーザー目標は、コンテキスト・ベースの照会、ユーザーへのインタビュー、および調査を通じて定義できます。

主として、次のような質問を設ける必要があります。

- アプリケーションの使用におけるユーザー個人の目標(奨励策を満たすためにできるだけ迅速にデータ入力を行うなど)
- 対象ユーザーが達成すべき主な目標
- 業務全体の目標(日次の受注オーダー入力の増加など)

ユーザー・タイプ別のソリューション

どのような企業でも、エンタープライズ・アプリケーション・スイートへのアクセス権を持っているのは、従業員のごく一部です。そして、ほとんどの企業では、テクノロジーへの投資効率を高め、全社的に一貫した情報伝達を行うことによるメリットを得ることを望んでいます。ただし、これらの目標を達成するために、従業員全員に必要なソフトウェアをすべて搭載したデスクトップ・マシンを支給することはできません。社内イントラネットが普及した要因の1つは、情報配布の合理化が低コストで実現できたことによります。企業は従来のような紙ベースの制約から解放されることで、社内の情報フローを飛躍的に改善できることに気づいています。

ただし、イントラネットに基づく情報のほとんどは比較的変動しないものです。これに対して、エンタープライズ・システム内の情報は、新規トランザクションが発生するたびに変化していきます。企業は、社内イントラネットによる配布の容易さと、エンタープライズ・アプリケーションのリアルタイム情報による精度と処理能力を結合させようとしています。

このソフトウェアでは、社内イントラネットを活用し、エンタープライズ・アプリケーションへのアクセス性を高めることができます。ブラウザ搭載デバイスにより、企業の情報リソースにリアルタイム・ウィンドウを提供できます。テクノロジー上のオーバーヘッドは小さいため、企業は情報フローに関与できるユーザー数を増やしつ、次の目標を達成できます。

- より広範囲の受け手に対する企業情報の迅速な提供
- 散在する業務や分散サイトの集約
- 目標、優先順位および戦略のよりよい伝達
- あらゆる組織レベルで情報の可用性を高めることによる意思決定の改善

ユーザーの範囲

エンタープライズ・アプリケーションの使用方法は、組織のユーザーごとに異なります。つまり、継続的なアクセスと、堅固なクライアント環境の機能やデスクトップ・ツール全体を必要とするユーザーも

いれば、状況を検討して簡単なトランザクションを入力できるだけでよいというユーザーもいます。ユーザーのスキルと業務要件に応じ、ニーズの範囲に基づいてエンタープライズ・アプリケーション全体でさまざまなテクノロジー要件を事前に想定できます。

エンタープライズ・アプリケーションに対するユーザーの要件は、組織における役割(ロール)に応じて異なります。分析担当のユーザーは、複数のデスクトップ・ツールを活用して企業情報を解析し、パッケージ化します。処理担当のユーザーは、迅速に利用でき、簡単にアクセスできる情報を必要とします。

分析担当のユーザー

このユーザーは、情報の収集、分析、再パッケージ化を行って、他の従業員や組織の管理者に配布します。この種のユーザーは、エンタープライズ・アプリケーション、スプレッドシート、パブリッシング・ツールなど、所定のデスクトップ・ツールを使用して、社内のさまざまな情報を収集し、意味を持った情報へととりまとめます。

クライアント/サーバー・モードでは、この種のユーザーが社内の情報フローをメンテナンスする上で必要とする、完全に統合化された OLE ベースのデスクトップを提供します。Windows クライアントには、個人の生産性アプリケーションとエンタープライズ・アプリケーションの処理ツール、およびそれぞれのデータ・リソースがまとめられています。この後、分析担当のユーザーは情報を統合し、部門別の観点から分析し、所要量計画、重役による意思決定、マーケティング、購買など、複数の業務分野に関するコンテキストで再パッケージ化できます。

処理担当のユーザー

もう一端を担う、このタイプのユーザーは、担当業務を実施できるように情報の検討を行います。このクラスのユーザーは組織ごとに異なり、受注入力担当者、作業現場スタッフ、重役などが含まれます。通常、処理担当のユーザーはシステムを使用してオーダー状況や在庫状況の検討などを行います。以降の社内フローに向けて情報を再パッケージ化してパブリッシュすることはありません。ブラウザ・モードではポイント・アンド・クリック・インターフェイスで必要なアクセスが得られるため、トレーニングも最小限に抑えられます。同時に、クライアントのオーバーヘッドが少ないため、必要なソフトウェアをすべてインストールしたクライアント・ワークステーションが装備できず、これまで情報フローから外されていたユーザーにもアクセス権を拡張できます。これらのユーザーには、標準的な TCP/IP ネットワーク経由でブラウザ搭載デバイスへアプリケーションをプッシュできます。この場合、コードはクライアント側には格納されないため、このタイプの全ユーザーに対してはソフトウェアのメンテナンスやアップグレードが一元化されるというメリットもあります。

エンタープライズ・アプリケーションの考慮事項

ユーザーと同様に、エンタープライズ・アプリケーションも、情報フロー内の役割に応じて、クライアント/サーバー・モードに適合する場合とブラウザ・モードに適合する場合があります。さまざまなソースからの情報を分析し、操作するツールを提供するアプリケーションには、クライアント/サーバー実装が適しています。クライアント/サーバー・モードには、必要な対話型処理とグラフィカル・サポートを提供するのみでなく、クライアントのプロセッサを専用に使用できるというメリットがあります。次に例を示します。

- モデリングとプロトタイプ化
- 予算処理と予測処理
- トランザクション別または得意先タイプ別のカスタム・インターフェイスを必要とする大量のトランザクション

共有情報の伝達と配布、および標準的なトランザクション入力処理のサポートを行うアプリケーションには、ブラウザ・モードが適しています。次に示すタイプのアプリケーションは、クライアントのリソースを妨げることなく情報を表示するという大きなメリットがあるため、ブラウザ・モードが理想的です。

- 情報収集およびプレゼンテーション・アプリケーション
- 照会ベースのセルフサービス・アプリケーション
- 反復的な標準トランザクション・アプリケーション

企業がリアルタイムで変化するビジネスに対応するには、コンピューティング・モードとアプリケーション環境が相互に補い合う形で適合している場合でも、クライアント/サーバー・ベースのアクセスと Web ベースのアクセスの両方のモードを使用可能にする必要があります。このソフトウェアには両方のモードがアーキテクチャに組み込まれているため、必要に応じてデプロイメントできます。企業では、ビジネス・データとビジネス・プロセスの一貫性を保ちつつ、両方のモードを混在させることができます。エンタープライズ・アプリケーションにクライアント/サーバーおよび Web ベースのアクセスを提供することで、EnterpriseOne は社内のあらゆるユーザーのニーズを満たすことができます。

クライアント/サーバー・モードとブラウザ・モードが 1 つのソフトウェア・ソリューションにあるため、ユーザーのニーズや必要なビジネス・プロセスの情報要件を満たすことができます。企業の情報リソースにアクセスするユーザー数を増やすことにより、次のような大きなメリットが得られます。

分散しているビジネスユニットの統合	組織全体で単一の一貫した情報ソースにアクセスするユーザー数が増加します。
プロセスの合理化	従来の紙ベースのプロセスを簡単に自動化できます。
より効率的な意思決定サイクル	決定事項が、エンタープライズ・アプリケーションにアクセスしている限られた数のユーザーにプッシュされるとは限りません。

この 2 つのモードには違いがあるにもかかわらず、単一のコンピューティング・ソリューションにクライアント/サーバー・ソリューションとブラウザ・ソリューションを結合する方向に移行しつつあります。各モードをデプロイメントする場合と方法は、ビジネス・ニーズとそのソリューションが持つ柔軟性によって異なります。

インターネットとクライアント/サーバー・テクノロジーの併用を考慮する際には、次の 2 つの根本的な課題にぶつかります。

- Web ベースのソリューションが最適の業務分野と、クライアント/サーバー・ソリューションが最適の業務分野を特定する必要があります。
- クライアント/サーバー・モードとブラウザ・モードの両方に対応するソリューションを実装する必要があります。

ユーザー・タスクの定義

ユーザー・タスクの分析では、ユーザーが目標を達成するためにシステムで実行するすべてのタスクを定義します。タスク分析の実施には、次のようなメリットがあります。

- 新製品に関するアイデアの生成
- 製品に組み込む主機能の識別
- 既に対象範囲が決定された製品に対するユーザーインターフェイスの設計
- 既に稼動している製品の有用性の改善

機能中心ではなくユーザー中心の設計アプローチを導入するには、ユーザーのタスクとそのタスクが実行されるコンテキストを把握する必要があります。目標の場合と同様に、ユーザー・タスクの分析もコンテキスト・ベースの照会、インタビュー、および調査を通じて収集できます。主として、次のような質問を設ける必要があります。

- ユーザーがシステムを使って実行する必要のあるタスク
- タスクのうち不可欠な側面と重要な側面
- タスクの実行順序(つまり、タスクのワークフロー)
- 現行の環境上の制約と問題点
- システムの使用によって改善可能な現行の作業プロセスの内容

タスク分析の例

この例では、タスク・ベースでのユーザー中心の設計の作成モデルとして、電子調達用セルフサービス・アプリケーションを使用します。タスク分析の例は、次のとおりです。

- 市場のニーズの変化に関する警告をみるために顧客がアプリケーションにアクセスします。
- 仕入先は警告に適宜応答します。
- 仕入先は、在庫不足データなど、買い手/売り手市場における重要なデータをジャストインタイムで受け取ります。
- 仕入先は、特定のオーダー要求を検索します。
- 仕入先は、出荷トラッキングのために現行のスケジュールを参照します。

次の例は、ユーザーが顧客または仕入先としてシステムで実行する全タスクを示しています。

#	顧客側のタスク	仕入先側のタスク
1.	ユーザー・プロフィールの変更	ユーザー・プロフィールの変更
2.	新規ユーザーID またはパスワードの要求	新規ユーザーID またはパスワードの要求
3.	エンタープライズからの日次警告の監視(初回)	エンタープライズからの日次警告の監視
4.	優先順位付きのスケジュールと処理の表示	却下されたオーダーなど、優先順位付き要求の表示
5.	日次配送スケジュールと遅延配送スケジュールの表示	日次納入スケジュールと遅延納入スケジュールの通知
6.	仕入先に対する見積要求の送信	購買担当者からの見積要求の表示
7.	仕入先からの回答の表示	購買担当者からの見積要求に対する回答
8.	仕入先からの特定のオーダーまたはオーダー項目の状況の検索	購買担当者に関する特定のオーダーまたは項目の状況の検索
9.	出荷残状況のトラッキング	納入のトラッキング
10.	未出荷オーダーの出荷要求	出荷要求に対する回答
11.	未払請求書の検討	期限超過請求書の検討
12.	受注オーダーの作成	受注オーダーの表示

13.	受注オーダー状況の表示	受注オーダー状況に対する応答
14.	仕入先在庫の検討	購買担当者在庫の検討

使用事例の作成

使用事例は、ユーザー・タスク・モデルを作成する方法です。使用事例の設計目的は、Web アプリケーション設計の基になるナビゲーション構造を理解しやすいように、ユーザー・タスク・モデルをフローチャート形式で作成することです。使用事例では、既存オーダーの状況のクエリーなど、システムで実行する必要があるタスクを記述します。使用事例アプローチを使用すると、システムの境界を定義し、明確な使用事例モデルがない場合にしばしば発生し得る対象範囲の拡大を防止できます。

対象ユーザーと使用事例を識別して定義する過程で、設計者はアプリケーションの対象範囲、つまり、アプリケーションで実行できることと実行できないことを定義します。

使用事例には、次のようなメリットがあります。

- 通信メカニズムの理解が容易になる。
- 要件の不備を防止できる(要件のトレース時)。
- 抽象レベル(低い修正コスト)におけるシステムの処理内容の簡潔なサマリーが得られる。
- カスタマーの言語を使用できる。

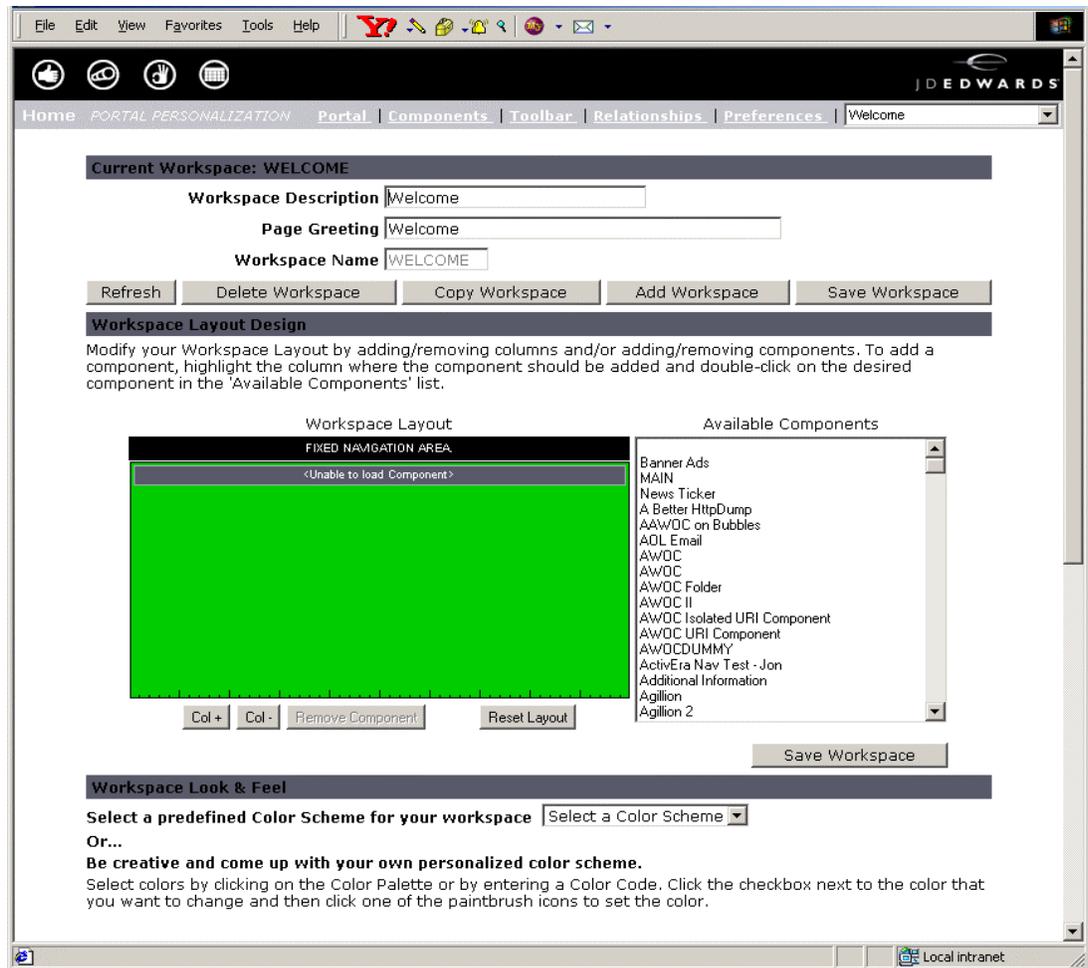
有用性の設計

J.D.・Edwards ソフトウェア・アプリケーションは、ユーザー中心の設計の基本理念に基づいて設計するだけでは不十分です。アプリケーション開発が進行するにつれて、エンド・ユーザーと共に設計の妥当性をチェックし、検証する必要があります。有用性の設計には、設計サイクル中に繰り返し実施する設計の有用性評価が含まれます。ユーザー中心の設計と有用性の設計を開発サイクルに取り込むことには、次のようなメリットがあります。

- ユーザーの効率とパフォーマンスの向上
- 運用効率と生産性の向上
- ユーザー経験の取込み
- 顧客満足度の向上
- 売上の増加と顧客との関係確立
- 全体の開発コストの削減
- トレーニングおよびサポート・コストの削減

ケース・スタディでは、有用性評価を実施した後に、設計者がフォームをどのように変更したかと、その理由について説明します。

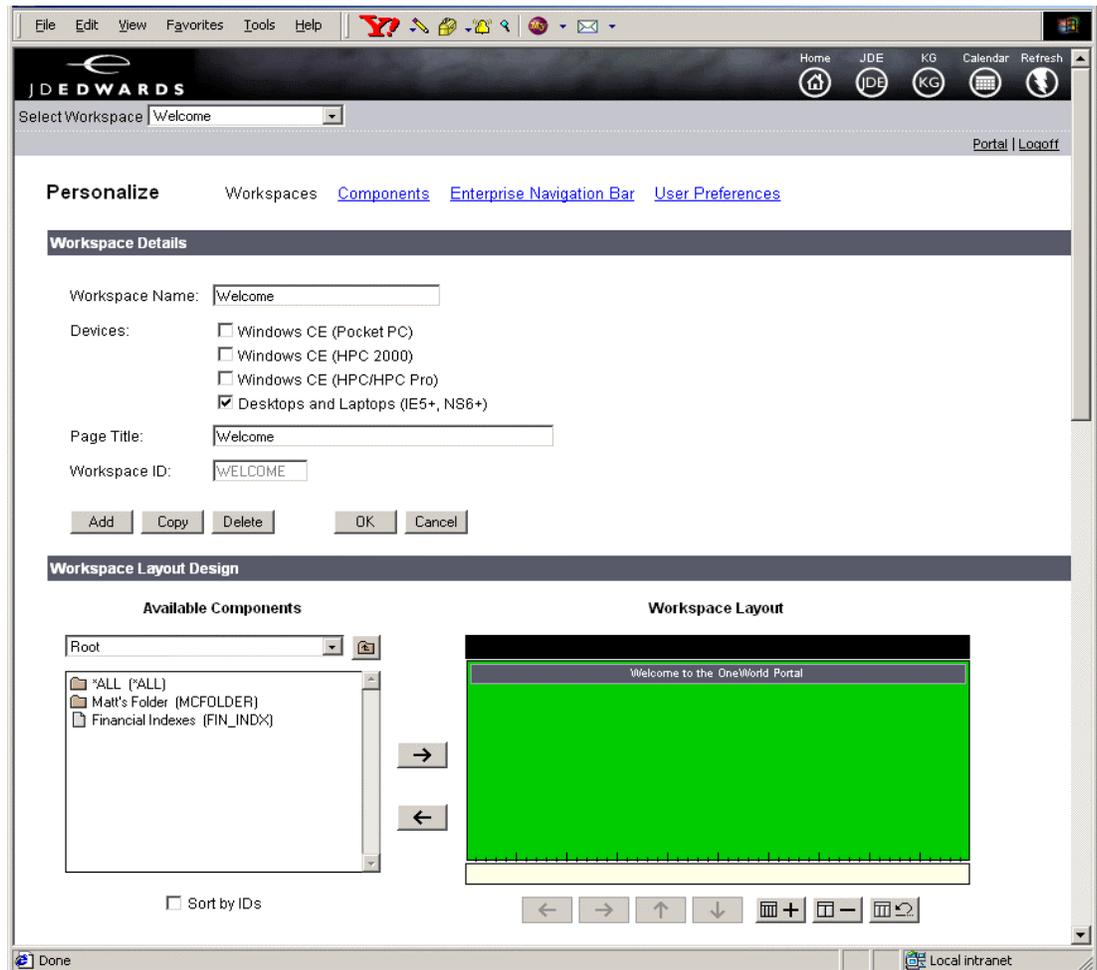
次のフォームは、開発者が最初に設計したものです。



評価に応えるために、設計者は有用性を高めるようなフォームの設計変更を模索しました。その結果、次のように変更することになりました。

- ブランドを強調する
- 各国のユーザーに不快感を与えない図形アイコンを使用する
- よりわかりやすいラベルと見出しを組み込む
- ユーザー・タスク・フローに基づいて、ユーザー定義の画面レイアウトを増やせるようにする
- わかりやすいラベルと見出しを組み込むことで、画面上のテキストの量を減らす
- 視覚的にすっきり見えるように整理する
- 情報を整理して適切にグループ化する

次のフォームは、設計者がインターフェイスに変更を加えた後のものです。



情報のアーキテクチャ

ほとんどの Web アプリケーションのインターフェイスはフォーム設計ツール(FDA)を使って作成されています。

ワイヤ・フレームの作成

Web アプリケーション設計に使用されるテクニックの 1 つが、Web 画面設計の初期モデルを記述するためにワイヤ・フレーム図を作成するプロセスです。ワイヤ・フレーム図は画面の内容を示すプレースホルダとなり、Web アプリケーションのアーキテクチャの概要を示す青写真の役割を果たします。開発サイクルにワイヤ・フレームを取り込むことで、Web アプリケーション作成プロセスの所要時間を短縮することができます。

使用事例には、次のようなメリットがあります。

- 画面間でのナビゲーション・フローを概念化する
- Web アプリケーションのコンテンツを明確にする

- Web アプリケーションの概要を提示する
- 概念的な設計プロトタイプを開発する
- 対話型領域におけるタスク分散方法に関する問題を明確化する
- ユーザーにとってシステムがどの程度複雑になるかを提示する(画面間での移動が多すぎる場合は、連結や簡素化が可能かどうかの検討が必要)
- ソフトウェア全体の編成を把握する強力な手段として役立つ
- 既存の設計にある潜在的な問題と、それに対して可能な解決策を明確化する

基本的な設計ガイドライン

FDA ツールの制約の範囲内で作業する際には、Web アプリケーション設計に関する次のガイドラインに従ってください。

- タスクに対するユーザーのニーズに基づいて情報を系統立てる。ユーザー・タスクの流れを表すレイアウトにする必要があります。
- 重要で使用頻度の高い情報を最初に表示する。
- 水平方向のスクロールは避ける(ユーザーはこの操作を好みません)。
- 小見出しを使用して、情報をセクション単位でグループ化する。
- 指示やラベルには、冗長な文章や句の使用を避ける。必要な処理の説明には動詞と能動態を使用します。ユーザー・インターフェイスは、操作内容が必要最小限でわかりやすいものにする必要があります。

ナビゲーションの定義

使用事例とワイヤ・フレーム図に基づいて、Web アプリケーションのナビゲーション・モデルを定義します。現在のナビゲーション方式では、ユーザーは一方向にしか移動できません。つまり、今開いているフォームを閉じると直前のフォームに戻ります。ただし、Web アプリケーションには[戻る]ボタンが組み込まれているため、ナビゲーション方式の設計にはさらなる柔軟性を必要とします。ユーザーが Web アプリケーションで目的を達成できなければ、ブラウザの[戻る]ボタンをクリックしてアプリケーションを終了することになります。

基本的なナビゲーションのガイドライン

次のリストは、Web アプリケーションと Web サイトを対象として適切なナビゲーションを設計するための、10 のガイドラインを示しています。ナビゲーションの設計時には、次のガイドラインに従ってください。

- 覚えやすいこと
- アプリケーション全体で一貫していること
- ユーザーにフィードバックを提供すること
- ユーザーのコンテキスト(エラー・メッセージなど)で表示すること
- 代替ナビゲーション手段を提供すること
- ユーザーにとって有効であること

- わかりやすい視覚的メッセージを提供すること
- わかりやすく目立つラベルを提供すること
- アプリケーションに適したものであること
- ユーザーの目標や行為をサポートすること

ナビゲーション方式

Web 設計者にとって重要な課題の 1 つは、Web フォームでの情報編成です。ナビゲーション方式は、ユーザーが必要な情報にすばやく簡単にアクセスできるものにする必要があります。設計目的に応じて、次のいずれかのタイプのナビゲーション方式を使用する必要があります。

- タスクベース

ナビゲーションは、ユーザーが処理するタスクの流れに基づきます。Web アプリケーションには、この方式が最も有効です。タスク・エクスプローラは、タスクベース・ナビゲーションの一例です。タスクベースのナビゲーションは参照パスをたどりますが、階層レベルが多くなりすぎないようにする必要があります。ユーザーが特定のタスクに到達するまでに移動する階層レベルは、4~6 レベルに抑えてください。

セルフサービス・アプリケーションには、タスク・ベースのナビゲーション・モデルが使用されています。ユーザーは〈Enterprise Portal (エンタープライズ・ポータル)〉を介してワークスペースにアクセスし、そこから特定のタスクにアクセスすることができます。

- 順次

このナビゲーション方式では、構造内で一度に 1 ステップずつ直線的に移動します。このタイプのナビゲーションは、ダイアログに基づくチュートリアル形式の設計に使用します。

- 情報

非直線的な設計です。この場合、ユーザーはハイパーテキスト・モデルを使用してページをジャンプしたりスキップすることができ、情報はカテゴリ別に編成されます。特定のナビゲーション・パスがないため、各種の情報を参照できるという点でより大きな柔軟性が必要になります。このタイプのナビゲーション方式は、Yahoo!などのサイトで使用されています。ポータルのワークスペースにも、このナビゲーション方式が使用されています。

リンク

カスタマイズされた多数の HTML コードを含むセルフサービス・アプリケーションなど、HTML の外観と操作性を持つアプリケーションを設計する場合は、ナビゲーション・メカニズムとしてリンクを使用する必要があります。グリッドやタブ、コントロールを使用して Windows ベースのアプリケーションを設計する場合は、ツールバーやメニュー・バーなど、標準のナビゲーション・スキームを使用すると便利です。

[検索]、[選択]、[OK]、[キャンセル]、[閉じる]などのボタンは、エンドユーザーが使い慣れたものであり、標準的なランタイム処理を提供します。ボタンはカスタマイズせずにできるだけこれらの標準のボタンを使用してください。

Web アプリケーションでナビゲーション・リンクを設計する場合は、次のガイドラインに従ってください。

- タスクの記述には能動態を使用する(勘定科目情報の表示など)。

- タスクの完了時にユーザーが別のパスに進まないようにする(カタログの表示など)。
- ユーザーが作業内容を保存してアプリケーションを終了できるように、わかりやすい場所に[閉じる]ボタンを配置する。
- ユーザーが追加情報にアクセスできるようにするにはハイパーリンクを使用する。グリッドでは、[選択]ボタンによる2段階の選択操作の代わりにハイパーリンクを使用します。
- わかりやすいハイパーリンクを提供する。表示される情報を明確に示します。たとえば、「旅行」ハイパーリンクで旅行代理店情報を表示するのは避けます。
- リンクには1語または短い語句(2~3語以内)を使用する。
- 長い文や段落にリンクを貼らない。
- リンクがないラベルには下線を付けない。ユーザーがリンクと単に強調されている他の用語を区別できるようにします。
- 未使用のリンクは下線付きの青色で表示する。ただし、ポータルナビゲーション・バーの場合は、色が競合する可能性があります。この場合は、コントラストがはっきりした色のリンクを使用してください。このような色は、リンクが見つらなくなるように設計する必要があります。

使用済みのリンクは下線付きの紫色で表示する。

現在選択しているリンクはカーソル位置に赤色で表示する。

- 電子メール・アドレスのリンクは、適切な電子メールのラベルを付けて下線付きの青色で表示する(電子メールのヘルプデスクなど)。
- 詳細を示す場合、「ここをクリック」を使用しない。文中の最も関連の深い語にリンクを貼ります。

正	受注残
誤	受注残:ここをクリック

検索インターフェイスの定義

検索機能は、Web ユーザーにとって単純で手軽なものにする必要があります。検索機能は、Web アプリケーションの最も重要なユーザー・インターフェイス要素の1つです。有用性調査は、ユーザーのうち Web 上で必要な情報を検索できるのは 64%にすぎないことを示しています。これは、検索アプリケーションが適切に設計されていないことが原因です。検索機能を設計する際には、次のガイドラインに従ってください。

上級検索

上級検索機能は、特定の情報を検索する場合に使用します。上級検索機能を使用すると、1つまたは複数の情報フィールドを使用して検索範囲を絞り込むことができます。上級検索機能を設計する際には、次のガイドラインに従ってください。

- 上級ユーザー向けには、標準的な QBE 検索機能に代わる上級検索オプションを提供する。
- メイン・フォームと〈単純検索〉フォームに、上級検索機能用のハイパーリンクを提供する。
- 検索のデフォルトである〈contains〉を使用して、QBE 機能を置き換える。
- 上級検索ページから単純検索に移動する手段を提供する。

- 検索条件を論理的にグループ化し、さまざまな検索オプションを明確に示す。
- 結果を表示するページで、さらに絞り込み検索ができるようにする。
- フォントの幅が狭く入力可能な文字数が増えるため、テキスト入力ボックスには Arial フォントを使用する。

検索結果の表示

検索結果の表示については、次のガイドラインに従ってください。

- 検索結果は検索を実行するページに表示する。
- 必要に応じて検索範囲を絞り込んで再検索できるように[リフレッシュ]ボタンを配置する。
- 検索結果セクションのタイトルには、目立つように太字のラベル[結果]を配置する。
- 検索結果は 10 レコード単位で表示する。
- 検索機能を使って特定のセクション表示に戻れるようにする。
- アクセスしたページやリンクに戻れるようにナビゲーションを用意する(1 2 3 4 など)。
- [Forward(進む)]と[Back(戻る)]ではなく、[Next(次へ)]と[Previous(前へ)]アイコンを使用する。
- レコードが見つからない場合、左上隅など目立つ場所に「レコードが見つかりません」というメッセージを表示する。

検索エラーの修正

絞り込み検索機能が使用しやすいように、次のガイドラインに従ってください。

- エラー・メッセージは、検索範囲を絞り込んだり検索パラメータを訂正できるように、検索した同じフォーム上に表示する。
- エラー・メッセージは、明確で具体的で特定の内容を示すものにする。
- レコードが見つからない場合、新しく検索できるようにする。
- 検索範囲を絞り込めるようにヒントを提供する。

HTML と Windows の機能上の相違点

EnterpriseOne には、次の 3 通りの対話型クライアントが用意されています。

- Microsoft Windows Full Client
- Windows Terminal Server Client
- HTML Client

これらのクライアントを組み合わせることで、エンタープライズ・ソリューションを包括的にサポートできます。すべてのクライアント・ソリューションで使用される共通コンポーネントは、EnterpriseOne の仕様です。すべてのクライアント・ソリューションは同じ仕様にアクセスするため、開発者はすべてのアプリケーションとレポートに「一度作成すればどこでも実行できる (Write Once, Run Everywhere)」という戦略を採用できます。

それでも、HTML クライアントの動作は、Windows クライアントとは若干異なります。フォーム設計ツールを使用して HTML アプリケーションを単独で、または Windows アプリケーションと共に生成するユーザーは、同じアプリケーションを両方の環境で効率的に機能するように設計するために、このような相違点を理解する必要があります。

Web アプリケーションを作成するには、Windows アプリケーションの場合と同様に、開発ツール・セットを使用します。Web アプリケーションを生成するには、次の手順で操作します。

1. フォーム設計ツール(FDA)でアプリケーションを保存する
2. Web サーバーに対してアプリケーションを生成する
3. アプリケーションをメニューに添付して実行モードを指定する
4. アプリケーションを実行する

現在、Web アプリケーションの生成処理には、手動で行う部分が比較的多く含まれています。今後は、このプロセスの一部が自動化される予定です。

外観の相違

Windows クライアントは Windows アプリケーションとして動作し、Windows プラットフォームで提供されるウィンドウ、フレーム、ダイアログ・ボックス、メニュー、ツールバーなどの GUI コンポーネントを使用します。

逆に、HTML クライアントは Web ブラウザで動作し、ブラウザに搭載された HTML 標準の共通コントロールを使用します。これらの共通コントロールは、Windows の GUI コンポーネントとは外観や操作方法が異なります。

将来的には、HTML 表示にはほとんどの場合に絶対位置が使用されるようになります。その結果、システムではコントロールがフォーム設計ツール(FDA)で定義されたとおりの位置に、FDA で設計されたとおりのサイズで表示されます。ブラウザと Microsoft Windows アプリケーションではフォントの表示が異なるため、プラットフォームによっては同じフォントの表示サイズが異なる場合があります。そこで、アプリケーション開発者はコントロールが Windows クライアントにも HTML クライアントにも十分な大きさであることを確認する必要があります。

絶対位置ルールの例外は、グリッド・コントロールの表示です。ブラウザの画面スペースを活用できるように、通常、HTML クライアントではグリッドがフォームと同じ幅で表示されます。つまり、グリッドの右境界がフォームの右境界まで拡大されます。グリッドの右にあるコントロールはグリッドの下に表示され、グリッドの左にあるコントロールはその位置に表示されます。

通常、グリッドはフォーム中央での操作となるため、グリッド・コントロールの左右にはコントロールを配置しないことをお勧めします。

動作の相違点

Windows クライアントはフィールドベースです。たとえば、ユーザーが[Tab]キーを使用してフィールドから移動すると、即時にデータ検証ルーチンが実行されます。また、データ・フィールドに関連する説明とエラー状況が直ちに表示されます。

逆に、HTML クライアントはページベースです。たとえば、ユーザーがフィールドからタブ移動しても、通常、その部分のデータが直ちに処理されることはありません。そのフィールドに対するイベントがすぐに起動されることはなく、イベントはキューに置かれます。ページ全体が JAS サーバーに送られると、キューに置かれたすべてのイベントがサーバー上で実行され、すべてのコントロールが

Windows クライアントのコントロール処理と同じ方法で処理されます。ページをサーバーに送信する処理を、ポスティングと呼びます。

JAS サーバーは、ユーザーから入力を含むページを受信すると、すべての入力データを入力された順に処理します。この処理中に、エンタープライズ・サーバーを使用してビジネス関数が処理され、データベース・クエリーなどが実行されます。この例では、データ検証ルーチンが処理され、関連する記述がエンタープライズ・サーバーからロードされます。その後、処理済みのページがブラウザに戻されます。データの送信(ポスティング)からブラウザへの戻りまでの処理全体を、ラウンドトリップと呼びます。この時点で、ユーザーはすべてのデータ・フィールドに関連する記述と、それぞれのエラー状況を確認できます。

JAS サーバーによる更新の処理方法

HTML クライアントのランタイム・エンジンによって、システムからのオープン・フォームの仮想イメージが管理されます。JAS サーバーは、ブラウザから送信されたページを受信すると、最初にそのユーザー・セッションに適切なクライアントの仮想イメージを検索します。次に、フォーム上の全データを次の順序で処理します。次の順序には、すべてのサーバー処理が含まれているとは限りません。あくまでもイベントの順序の概要を示しています。

1. Control is entered(コントロールへの入力)イベントが実行されます。
2. コントロールに変更があると、ブラウザから送信されたデータが、JAS サーバー上で実行中の仮想クライアントのフォーム・コントロールにコピーされます。
3. コントロール値に変更があった場合は、次のイベントが実行されます。
 - Control is exited(コントロールの終了)
 - Control is exited and changed inline(コントロールの終了とインラインでの変更)
 - Control is exited and changed asynchronously(コントロールの終了と非同期による変更)
4. コントロールが検証され、フォーマットされます。
5. 変更があったすべてのコントロールについて、ステップ 1 から 4 が繰り返されます。
6. QBE またはグリッド・セルに変更があった場合は、次のイベントとランタイム・ロジックが実行されます。
 - Column is entered(カラムの入力)
 - Column is exited(カラムの終了)
 - Column is exited and changed inline(カラムの終了とインラインでの変更)
 - Column is exited and changed asynchronously(カラムの終了と非同期での変更)
 - The grid cell is validated and formatted(グリッド・セルの検証とフォーマット)
 - Row is entered(ローの入力)
 - Row is exited(ローの終了)
 - Row is exited and changed inline(ローの終了とインラインでの変更)
 - Row is exited and changed asynchronously(ローの終了と非同期での変更)
7. 変更があったすべてのグリッド・セルとローについて、ステップ 6 が繰り返されます。
8. OK Button Clicked(OK ボタンのクリック)など、このページを送信した Button Clicked(ボタンのクリック)イベントが実行されます。

更新と入力処理をトリガーするアクション

次のユーザー操作によって、ブラウザから Web サーバーへの更新がトリガーされます。

- [OK]や[キャンセル]などのツールバー・ボタンのクリック
- フォーム・エグジット、ロー・エグジットまたはツール・エグジットのクリック
- Selection Changed(選択変更)イベントでのイベント・ルールが含まれているラジオ・ボタンまたはチェックボックスのクリック
- プッシュ・ボタンまたはビットマップのクリック
- クリック可能なスタティック・テキスト、グリッド・セルまたはテキスト・ブロック・コントロールなど、ハイパーリンクのクリック
- テキスト・フィールドやグリッド・セルのビジュアル・アシストのクリック
- グリッド・ローのペーパー・クリップ・イメージのクリック
- タブ・ページの切替え
- グリッドの[前へ]と[次へ]リンクのクリック
- 1 行のグリッドの[編集]ボタンのクリック
- 編集可能グリッドの初回のクリック
- 親子コントロール内のツリー・コントロールまたはツリーの次のような操作の実行
 - ツリー・ノードの展開
 - ツリー・ノードの縮小
 - ツリー・ノードのドラッグ・アンド・ドロップ
 - Double-Click Leaf Node(リーフ・ノードのダブルクリック)イベントにイベント・ルールが含まれている場合の、リーフ・ノードのダブルクリック
- [Tools/Refresh(ツール/リフレッシュ)]メニューのクリック

更新をトリガーするレコードは、入力処理によって管理されます。これらの入力処理イベントとは異なります。事実、入力処理イベントに ER イベントを添付する必要があるとは限りません。入力処理は、更新されたレコードを特定の順序で解決し、発生した変更内容を判断するために特定の手掛かりを検索します。入力プロセッサは検出したイベントを処理するため、グリッドに対するプログラムによる変更はユーザーが変更内容を入力した場合と同様に解釈されます。次のリストは、入力プロセッサでイベントが管理される順序、変更の検出方法、トリガーされる処理を示しています。

1. [ショートカット]または[戻る]ボタンがクリックされると、プロセッサはその処理を実行して終了します(それ以上の処理は実行されません)。
2. 該当する場合は、プロセッサが QBE イベントを処理します。
3. グリッド、PC イベント、ツリーを除き、ユーザーがアクセスまたは入力したコントロール・イベントがプロセッサで処理されます。システムでは、JavaScript を使用して次の条件が検出されます。
 - テキスト・フィールド – プロセッサは、ユーザーがアクセスしたすべてのコントロールに「entered(入力)」または「exited(終了)」イベントを送信します。また、ユーザーがアクセスしたコントロールに「changed(変更)」イベントが送信されます。
 - チェックボックス – プロセッサによって変更後の状態が検索され、該当する場合は「changed」イベントが送信されます。

- ラジオ・ボタン – プロセッサによって「clicked(クリック)」イベントのみが送信されます(変更はコントロール自体で検出されます)。
 - コンボ・ボックス – プロセッサによって選択が検討され、該当する場合はイベントが送信されます。
4. グリッド・イベントまたは親子イベントがプロセッサによって処理されます。
- グリッド・イベント – JavaScript ではユーザーのアクセス先は示されないため、グリッドの内容全体が送信されます。プロセッサによって、グリッド内の以前の値が現行の値と比較されます。変更があった場合は、グリッドに「cell edit(セル編集)」イベントが送信されます。このイベントによって、grid focus(グリッドのフォーカス)、row entered(ローの入力)、column entered(カラムの入力)、続いて column exited/change(カラムの終了/変更)、row exited/changed(ローの終了/変更)の各イベントが起動されます。これらのイベントは、処理時に上から下へ、左から右へと順に発生します。システムでは、自動更新にもこれと同じロジックが使用されますが、一度に送信されるローが少数です。
また、プロセッサでは、ユーザーがチェックボックスを使用してオンにしたローも選択されます。
 - 親子イベント – システムでは、PC コントロール用に次の相互排他イベントの 1 つを送信できます。
 - Cancel drag drop(ドラッグ・ドロップのキャンセル)(ドラッグが開始されたが、次の操作がドロップではなかった場合)
 - Row selections(ローの選択)
 - Next/Prev page(次/前ページ)
 - Expand/Collapse node(ノードの展開/縮小)
 - Drag(ドラッグ)
 - Move(移動)
 - Copy(コピー)
5. ツリー処理イベント – システムでは、ツリー・コントロール用に次の相互排他イベントの 1 つを送信できます。
- Row selections
 - Next/Prev page
 - Expand/Collapse node
 - Double-click node(ノードのダブルクリック)
6. クリック処理イベント – システムでは、次の相互排他イベントの 1 つを送信できます。
- Customize Grid(グリッドのカスタマイズ)
 - Export/Import(エクスポート/インポート)
 - Refresh(リフレッシュ)
 - Load Form(フォームのロード)(リンクまたはメニューからオープン)
 - Grid Prev/Next page(グリッドの前/次ページ)
 - Form/Row/Report/View Menu Exit(フォーム/ロー/レポート/ビュー・メニューの終了)
 - Clicked Hyper Link column on grid(グリッド上のハイパーリンク・カラムのクリック)

- Tab Page Clicked(タブ・ページのクリック)
- Grid Tab Clicked(グリッド・タブのクリック)
- Grid Next Line(グリッドの次行)(1 行の編集)
- QBE/Cell Visual Assist(QBE/セルのビジュアル・アシスト)
- Row double-clicked(ローのダブルクリック)(選択)
- Row Header double-clicked(ロー見出しのダブルクリック)(MO)
- Embedded MO event(MO イベントの埋込み)
- Check attachments(添付の確認)
- Textblock hyperlink clicked(テキストブロック・ハイパーリンクのクリック)

7. Button clicked — この種のイベントには、すべての標準ボタンが含まれます。

更新をトリガーしないアクション

次のイベントが発生しても、更新が自動的にトリガーされることはないため、ラウンドトリップは発生しません。

- [Tab]キーを使用したテキスト・フィールドからの移動
- [Tab]キーを使用したグリッド・セルからの移動
- [Tab]キーを使用した QBE セルからの移動
- ツリー内のツリー・ノードの選択

ラウンドトリップの管理

ラウンドトリップはネットワーク・リソースを使用するため、パフォーマンスを低下させます。ラウンドトリップが多すぎると、システムのスケーラビリティとパフォーマンスが低下します。また、ラウンドトリップはユーザーの作業効率も低下させます。ラウンドトリップが発生した場合、ユーザーはサーバーでページが処理されてブラウザに戻されるまで待つ必要があります。ページが戻されると、ブラウザに表示されます。ラウンドトリップはパフォーマンスを低下させますが、データの正確さが重要な場合には(予想されていても)許容できます。

このような影響があるため、必要以上のラウンドトリップを回避することが Web ベース・アプリケーションの重要な設計原則となります。ただし、更新がパフォーマンスに及ぼす影響を考慮し、代替ソリューションを排除した後も、アプリケーションで何らかのイベントを直ちに発生させる必要がある場合は、次のプロセスを使用してこの目標を達成できます。

更新をトリガーするイベントについて、イベント・プロパティ HTML Post を設定します。このプロパティは、次のイベントに使用できます。

- [Tab]キーを使用したテキスト・エディタからの移動: Control Is Exited(コントロールの終了)、Control Is Exited and Changed Inline(コントロールの終了とインラインでの変更)、Control Is Exited and Changed Asynch(コントロールの終了と非同期での変更)
- [Tab]キーを使用したグリッド・セル・エディタからの移動: Grid Column is Exited(グリッド・カラムの終了)、Grid Column Is Exited and Changed Inline(グリッド・カラムの終了とインラインでの変更)、Grid Column is Exited and Changed Asynch(グリッド・カラムの終了と非同期での変更)

- ツリー・ノードの選択: Tree Node Is Selected(ツリー・ノードの選択)、Tree Node Level Is Changed(ツリー・ノード・レベルの変更)

EnterpriseOne アプリケーションには、前述のソリューションをお勧めします。Xe では、既に必要なイベントのイベント・プロパティを設定している場合は、ジェネレータでイベント・ルールをスキャンしなくてもよいように、フォーム・プロパティ HTML Auto Refresh もオフにする必要があります。

グリッドの選択

Windows クライアントでは、ユーザーはローをクリックすることでグリッド・ローを選択します。選択されたローはハイライトされます。複数選択グリッドの場合は、ユーザーが別のローを選択すると、同時に[Shift]または[Ctrl]キーを押さない限り、その前に選択されたローが自動的に選択解除されます。

HTML クライアントでは、グリッド・ローを選択してもハイライトされません。単一選択グリッドの場合は、各グリッド・ローの隣にラジオ・ボタンが表示されます。このラジオ・ボタンをクリックして、別のグリッド・ローを選択します。複数選択グリッドの場合は、代わりに各グリッド・ローの隣にチェックボックスが表示されます。

注:

HTML クライアントでは、複数選択グリッド内で新規のローを選択しても、その前のグリッド・ローが自動的に選択解除されることはありません。

グリッド・ローの添付

Windows では、グリッド・ローに添付ファイルがあっても、デフォルトではペーパークリップ・アイコンは表示されません。カーソルをロー見出し上に動かすと、ランタイム・バッチ・プログラムによって、そのローに添付ファイルが存在するかどうかを検索されます。添付ファイルが存在する場合は、ペーパー・クリップが表示されます。HTML クライアントでは、添付ファイルの有無に関係なく、最初はずべてのグリッド・ローに何も表示されません。ロー見出しのペーパークリップ・アイコンをクリックすると、ランタイム・バッチ・プログラムによって、そのローに添付ファイルが存在するかどうかを検索されます。特定のローに添付ファイルが存在する場合は、ペーパークリップ・アイコンが表示されます。

タブの順序

HTML クライアントでは、タブ順序がサポートされますが、Windows クライアントの場合とは若干の違いがあります。HTML クライアントはブラウザ内で管理されるため、ブラウザのタブ順序を無効にすることはできません。ブラウザでは最初はページがタブ付きで表示されますが、タブ順序の終わりに達しても、フォームの最初のコントロールに折り返すことはありません。代わりに、フォームの外側にある専用 URL アドレス・ボックスに停止マークが表示されます。

グリッドの機能

Windows クライアントでは、グリッド上での[ズーム]、[最大化/復元]、[グラフ]、[印刷]がサポートされます。これらの機能は、HTML クライアントでは使用できません。

オートコンプリート編集

HTML クライアントでは、ブラウザから提供されるオートコンプリート機能を使用します。

クライアント専用のビジネス関数

クライアント専用のビジネス関数は、Web ではサポートされません。すべてのビジネス関数は、サーバーにマップされます。クライアント/サーバーのビジネス関数またはサーバー専用のビジネス関数では、クライアント専用のビジネス関数をコールしないでください。

ただし、これには 1 つ例外があります。フォーム・インターコネクトがあるために、NER にクライアント専用のマークのみが付いている場合（ローカル・ハード・ディスクへのアクセスや Windows のみのコールを行わない場合など）、NER は Web 上で正常に機能します。これらの NER はクライアント専用にしておいてください。これらの NER の生成時には、C のコードの代わりに Java クラスが生成されます。ここでは、このタイプの NER を Interpretive（解釈）NER と呼びます。

要約すると、サーバー上でクライアント専用 NER が実行されると、自動的に Interpretive NER が使用されます。Interpretive NER については、次のガイドラインを考慮してください。

- Interpretive NER をコールできるのは、アプリケーション ER または他の Interpretive NER からのみです。
- Interpretive NER は、UBE、クライアント/サーバーNER、サーバー専用 NER からはコールしないでください。
- フォーム・インターコネクトが含まれる場合は、クライアント専用として指定する。クライアント専用指定を使って、Interpretive NER を使用すべきかどうかが決まります。

複数行の編集と自動更新

デフォルトでは、HTML クライアントには編集可能グリッド行が 1 行だけ表示されます。特定のアプリケーション用に、複数の編集行を表示できます。この機能は、大量データの入力アプリケーション向けにお勧めします。複数行の編集機能は、フォーム・レベルの機能です。この機能は、フォーム設計ツールで変更できます。

フォームが複数行の編集用に設定されている場合は、ランタイム・バッチ・プログラムによって自動更新機能も提供されます。ユーザーがグリッドにデータを入力すると、入力された最初の 2 行が自動的に JAS サーバーに送信されて処理されます。サーバーでは、*Grid Row is Exited and Changed-Inline*（グリッド・ローの終了とインラインでの変更）や *Grid Row is Exited and Changed-Asynch*（グリッド・ローの終了と非同期的変更）などのイベントが処理されます。Web サーバーによって計算されたデータは、自動的に Web ブラウザに戻されます。たとえば、関連する説明は、最初の 2 つのグリッド行に更新されます。それと同時に、ユーザーはデータ行の入力を継続できます。この機能によって、データ入力アプリケーションのパフォーマンスが大幅に向上します。

注:

Row is Exited And Changed-Inline（ローの終了とインラインでの変更）イベントは、ユーザーが次のローに入力するまで処理されません。

プラットフォームの互換性

Web アプリケーションの設計時には、Web ブラウザの互換性の問題に注意する必要があります。各ブラウザによる HTML タグの解釈は少しずつ異なります。テーブル、フォーム、リンク、配置の各タグの機能も、ブラウザごとに異なります。Web アプリケーションの設計時には、次のガイドラインに従ってください。

- 該当する Web ブラウザのバージョンを対象として設計を行います。現在、Web アプリケーションは、Internet Explorer 5.5 以上と Netscape 6.2 以上のリリースで動作します。
- 1 つのブラウザでしかサポートされないタグは使用しないでください。
- リリース前に、複数のブラウザとプラットフォームで設計をテストします。

対象ユーザーのニーズに適した画面解像度を選択する必要があります。OneWorld Xe の場合、フォーム設計ツール(FDA)のデフォルトのフォーム・ガイド・サイズは 640×480 ピクセルです。15 インチ・モニタを使用するほとんどのユーザーは、800×600、17 インチ・モニタを使用するユーザーは 1024×768 の画面解像度を使用しています。製造業や流通業における B2B ユーザーは、通常、施設では 800×600 の画面解像度を持つ 15 インチ・モニタを使用しています。

Windows CE またはポケット PC 用のアプリケーションを設計する場合は、[Forms Guide(フォーム・ガイド)]メニューから 240×320 ピクセルのオプションを選択する必要があります。

Web 用アプリケーションの設計

フォーム設計ツール(FDA)を使って Web(HTML)用の対話型アプリケーションを開発する場合は、設計およびコーディング中にいくつかの設計方針を検討する必要があります。ここでは、これらの戦略について説明します。

通常、HTML Web アプリケーションの外観と機能は、Windows アプリケーションとは異なります。設計要件は、両者の相違点を具体的に示すものです。たとえば、従業員、カスタマー、サプライヤーのセルフサービス・アプリケーションの場合、このソフトウェア以外のユーザーがアプリケーションとフォーム間でナビゲート可能であることが要件の 1 つでした。この要件は、このソフトウェア独自のナビゲーション機能と見なされていたすべてのメニュー・エグジットとツールバー・エグジットを削除することで完結しています。メニュー・エグジットとツールバー・エグジットを置き換えることで、設計ではユーザーが HTML ベースのアプリケーションで一般的なハイパーリンクを使用して、アプリケーションとフォームの間でナビゲートできます。

Windows プラットフォームと HTML プラットフォームの両方に使用できるアプリケーションを設計する際に、両者に異なる外観を持たせたい場合は、フォーム・モードの使用が必要になることがあります。フォーム・モードについては、『開発ツール』ガイドを参照してください。

新規アプリケーションへのアクセス・メニューを作成する場合は、特定のフォーム・モードでフォームにアクセスできます。アプリケーションが Web 専用で、FDA でフォーム・モードを使用しなかった場合は、これらのいずれかのモードでアプリケーション (HTML シリアル化オブジェクト) を生成できます。ただし、Web 専用アプリケーションで、特定のモードで処理する必要のある他のアプリケーションへのフォーム・インターコネクトを提供する場合は、その必須モードで新規アプリケーションを設計して生成する必要があります。この要件は、フォーム・インターコネクトが処理される時に、実行モードが継承されることによるものです。つまり、子フォームには、コール元フォームのモードと同じモードが強制的に使用されます。

注:

メニューのエントリ・ポイントからアクセスされるフォームは、すべて同じモードで生成する必要があります。

Windows アプリケーションの作成時と同様に、適切なタイプのフォームを選択することが設計時の重要な考慮事項です。適切なフォーム・タイプを使用しないと、実行時の効率とユーザーの満足度が低下します。たとえば、テーブルの更新をサポートするフォーム・タイプ(見出しなし詳細や見出し詳細など)を表示専用で使用すると、ユーザーがクリックして各グリッド・ローにアクセスした時点でフォームがリフレッシュされます(つまり、Web サーバーへのラウンドトリップが発生します)。

また、Web 用のアプリケーションを設計する際には、パフォーマンスも重要な考慮事項となります。Web 環境には、パフォーマンスに関して Windows 環境に存在するタイプの問題のみでなく、コーディング前に考慮すべき問題もあります。Web 用のアプリケーションを設計する際には、次の要件を考慮してください。

- 即時処理を必要とするロジックを含むイベント・ルールを使用しないことで、不要なラウンドトリップをできるだけ減らします。
- ビジネス・ロジックをユーザー・インターフェイス・ロジックから分離します。たとえば、フォーム上のユーザー・インターフェイスを制御するには、イベント・ルールのみをアプリケーションから直接使用する必要があります。ただし、計算やデータベース・アクセスなどのビジネス・プロセスをカプセル化するビジネス・ロジック・コンポーネント(ビジネス関数)や、他のタイプのビジネス・ロジックを作成し、これらのコンポーネントにアプリケーションからアクセスする必要があります。これにより、アプリケーションの管理が容易になります。イベント・ルール内のロジックを減らすと、プロセッサに対するアプリケーションの依存度も低くなるため、アプリケーションのパフォーマンスが向上します。
- データベースへのアクセス・パフォーマンスを考慮します。テーブルには必ず適切なインデックスを設定します。データベースへのアクセスには、できるだけインデックスを使用します。3 層アーキテクチャを持つシステムに適切な設計方法は、アクセスする個々のコンポーネント(ビジネス関数)へのデータベース・アクセスのロジックを、アプリケーションから分離することです。
- Grid Record is Fetched(グリッド・レコードのフェッチ)または Write Grid Line - Before(グリッド線の作成-前)イベントでは、グリッド線の表示を抑制する設計に注意してください。検索条件に基づいてフェッチされるグリッド・レコードのほとんどまたはすべてをフォームで抑制する場合は、ほとんどのグリッド・レコードを抑制しなくてもよいように、より限定的な検索条件を工夫する必要があります。
- Page At a Time(1 度に 1 ページ)処理はオフにしないでください。フェッチされた合計レコード数が必要な場合は、ビジネス関数から JDB をコールすることで、この目標を達成できます。Page at a time 処理は、パフォーマンスを大幅に向上させます。
- Web サーバーからエンタープライズ・サーバーへのラウンドトリップは気にしないでください。通常、Web サーバーとエンタープライズ・サーバー間のリンクは高速です。また、どちらのサーバーでもスマート・キャッシュが実行されます。

フォーム設計ツール(FDA)を使用した Web アプリケーションの設計

ここでは、フォーム設計ツール(FDA)を使って Web アプリケーションを作成するためのヒントとテクニックについて説明します。

参照

- 設計後のフォームを生成する方法の詳細については、使用しているプラットフォーム用の『Java サーバー・インストール』ガイドの「シリアライズ・オブジェクトの生成」を参照してください。

複数モードによるフォームの設計

制御モードを使用して、複数のインターフェイスを持つアプリケーションを開発できます。これにより、同じアプリケーションの複数バージョンを管理する手間が省けます。ベースとなる 1 つのアプリケーションを作成し、異なるインターフェイス用アプリケーションに修正するためにモードを使用できます。各モードでは、フォーム上のコントロールを有効化したり非表示にすることができます。各モードでは、コントロール、カラム、メニュー・エグジットの可視性と有効化/無効化のプロパティのみ変更できます。非表示フィールドを表示にすると、現行モードでのみ表示されます。他のプロパティは全モードに共通です。すべてのフィールドが有効化され、すべてのフォームが表示されます。

制御モードは Windows ランタイム・エンジンでは認識されず、HTML ランタイム・エンジンでのみ認識されます。モード 1 はデフォルトのモードです。アプリケーションをメニューに添付して実行できます。このメニューにより、1 つのアプリケーションをさまざまなモードで実行できます。Web 上でアプリケーションを実行すると、デフォルトではモード 1 で実行され、指定した場合は別のモードで実行されます。Windows メニューにあるアプリケーションを添付すると、Windows ランタイム・エンジンでは指定したモードが無視され、アプリケーションはモード 1 で実行されます。アプリケーション全体で一貫したモードを使用してください。フォームを Web 対応バージョンで作成するには、eGenerator を使用して Java と HTML で生成します。このジェネレータを使用すると、1 つまたは複数モード用のフォームを同時に生成できます。

メニューおよびツールバー・エグジットの非表示化

フォーム・エグジットとロー・エグジット、および他のすべてのタイプのメニューおよびツールバー・エグジット(選択、キャンセル、OK または削除)をアプリケーションで非表示にする場合は、モード 2 または 3 で非表示オプションを選択します。モード 2 で非表示オプションを選択すると、エグジットはモード 2 でのみ非表示になります。モード 1 と 3 の場合、エグジットは引き続き表示されます。

▶ メニューおよびツールバー・エグジットを非表示にするには

1. 〈Form Design(フォーム設計)〉で、モード 2 または 3 を選択し、作業中のフォーム上で [Form(フォーム)]メニューから [Menu/Toolbar Exit(メニュー/ツールバー・エグジット)]を選択します。
2. 非表示にするエグジットを選択し、[Select(選択)]をクリックします。
3. [State(状態)]グループ・ボックスで [Hidden(非表示)]をクリックし、[OK]をクリックします。モード 2 または 3(作業中のモード)で生成されたエグジットは、フォームに表示されなくなります。

In-Your-Face-Errors の有効化

In-Your-Face-Errors プロパティは、HTML プラットフォームでのみ使用可能なフォーム・レベルのプロパティです。通常、システムでは、アプリケーション・エラーを示すために、アプリケーションの右上部分にある Errors and Warnings (エラーと警告) ハイパーリンクがハイライトされます。ただし、In-Your-Face-Errors プロパティを有効化すると、アプリケーション・エラーが Web ページに表示されません。

▶ In-Your-Face-Errors を有効化するには

1. [Form Design] で、次のどちらかのタスクを行います。
 - 新規フォームを作成します。
 - 既存のフォームを選択し、[Form] メニューから [Form Properties (フォーム・プロパティ)] を選びます。
2. <Form Properties> で、[Enable In-Your-Face-Errors Display (In-Your-Face-Errors 表示の有効化)] をクリックし、[OK] をクリックします。

画面サイズに合わせたフォームのサイズ設定

対象ユーザーのブラウザのサイズに従ったフォームのサイズ設定が必要になる場合があります。たとえば、カスタマー・セルフサービス・アプリケーションの場合、ほとんどのユーザーは 640 × 480 の画面解像度のみで実行します。

適切なサイズ設定のガイドを作成するために、[Form Properties] ダイアログで [Form Guide (フォーム・ガイド)] 値を設定するように選択できます。これにより、FDA で青い基準線を適切に調整できます。FDA では、サイズが規定されるのではなく、単に設計者がオブジェクトを青い基準線の範囲内に保つ参考になります。

▶ 画面サイズに合わせてフォームのサイズを設定するには

1. [Form Design] で、次のどちらかのタスクを行います。
 - 新規フォームを作成します。
 - 既存のフォームを選択し、[Form] メニューから [Form Properties] を選びます。
2. <Find/Browse Form Properties (フォーム・プロパティの検索/表示)> で、[Form Guide] ドロップダウン・メニューから使用する画面サイズを選択し、[OK] をクリックします。

グリッド・ロー・セレクタの非表示化

Hide Grid Row Selector プロパティは、HTML プラットフォームで使用可能なグリッド・レベルのプロパティです。グリッド上にグリッド・ロー・セレクタを表示しない場合は、[Grid Properties (グリッドのプロパティ)] ダイアログ・ボックスで [Hide Row Selector (ロー・セレクタの非表示)] オプションを選択します。

▶ グリッド・ロー・セクタを非表示にするには

1. 〈Form Design〉で、作業中のフォーム上でグリッドをクリックし、[Edit(編集)]メニューから [Item Properties(項目のプロパティ)]を選びます。
2. 〈Grid Properties〉の[HTML Properties(HTML のプロパティ)]セクションで、[Hide Row Selector]をクリックし、[OK]をクリックします。

代替グリッド・ロー・フォーマットの表示

Use Alternate Grid Row Format プロパティは、HTML プラットフォームで使用可能なグリッド・レベルのプロパティです。このプロパティを使用すると、グリッドの外観を制御できます。

このオプションを使用するには、[Row Format(ロー・フォーマット)]ボタンをクリックして、特定の HTML タグを指定する必要があります。〈HTML Alternative Grid Row Format(HTML の代替グリッド・ロー・フォーマット)〉ダイアログに表示される指示にしたがって、グリッド・ローの HTML タグを作成してください。これらの HTML タグによって、各グリッド・ローの外観が制御されます。また、グリッド・レコードとハイパーリンクについて、メディア・オブジェクト・イメージまたはテキスト・タイプのオブジェクトを出力するように選択することもできます。

このオプションは、更新不可のグリッドでのみサポートされます。

▶ 代替グリッド・ロー・フォーマットを表示するには

1. 〈Form Design〉で、作業中のフォーム上でグリッドをクリックし、[Edit]メニューから [Item Properties]を選びます。
2. 〈Grid Properties〉の[HTML Properties]セクションで、[Use Alternate Grid Row Format(代替グリッド・ロー・フォーマットの使用)]をクリックし、[OK]をクリックします。
3. 〈HTML Alternative Grid Row Format〉で、必要に応じて、各グリッド・ローの外観の制御に必要な HTML タグを指定して文字列を入力します。
4. [OK]をクリックします。
5. 〈Grid Properties〉で [OK]をクリックします。

複数行編集機能を使用したページ・リフレッシュの制御

Allow Multi-line Edit プロパティは、HTML プラットフォームで使用可能なグリッド・レベルのプロパティです。このプロパティを適用できるのは、ユーザーにグリッド・レコードの編集を許可するグリッドのみです。このオプションをオンにすると、HTML のランタイム・バッチ・プログラムにより、アクティブなグリッド・ローのみでなく、各セル(すべてのロー)に Editable Text Box(編集可能テキスト・ボックス)コントロールを含むグリッド・ローが表示されます。また、ユーザーがグリッド・ローから移動するたびに、システムによってページがリフレッシュされることがなくなります。代わりに、Grid Row イベントの処理が遅延され、自動送信を使用して 3~5 つのローのグループ単位でのみ処理されます。

▶ 複数行編集機能を使用してページ・リフレッシュを制御するには

1. 〈Form Design〉で、作業中のフォーム上でグリッドをクリックし、[Edit]メニューから [Item Properties]を選択します。
2. 〈Grid Properties〉の[HTML Properties]セクションで、[Allow Multi-Line Edit(複数行編集の許可)]をクリックし、[OK]をクリックします。

グリッド・レコードの各ページのグリッド・ロー数の制御

Grid Row Count プロパティは、HTML プラットフォームで使用可能なグリッド・レベルのプロパティです。このオプションにゼロ以外の値を設定すると、HTML のランタイム・バッチ・プログラムでは指定した数のグリッド・ローが表示されます。デフォルト(値 0)では、グリッドにページごとに 10 個のレコードが表示されます。

▶ グリッド・レコードの各ページのグリッド・ロー数を制御するには

1. 〈Form Design〉で、作業中のフォーム上でグリッドをクリックし、[Edit]メニューから[Item Properties]を選択します。
2. 〈Grid Properties〉の[HTML Properties]セクションで、[Grid Row Count(グリッド・ロー数)]フィールドにページごとに表示するグリッド・ロー数を入力し、[OK]をクリックします。

グリッド・セルへのチェックボックスの表示

Check Box プロパティは、HTML プラットフォームで使用可能なグリッド・カラム・レベルのプロパティです。このプロパティによって、オン/オフ状態情報を提供する特定のカラムのグリッド・セルにチェックボックスが表示されます。

このオプションを使用するには、Checked および Un-Checked 状態の値を指定する特定の値を入力する必要があります。実行時に、システムではカラムの GC 変数の値が検出され、その値がオン状態またはオフ状態のチェックボックスの表示用に指定した設定と相互参照されます。

▶ グリッド・セルにチェックボックスを表示するには

1. 〈Form Design〉で、作業中のフォーム上でグリッドをクリックし、[Edit]メニューから[Item Properties]を選びます。
2. 〈Grid Properties〉の[Grid Columns(グリッド・カラム)]セクションで、グリッド・カラムを選択して[Grid Column Properties(グリッド・カラムのプロパティ)]をクリックします。
3. 〈Grid Column Properties〉の[Display Style(表示スタイル)]セクションで、[Check Box(チェックボックス)]をクリックします。
4. [Values]をクリックします。
5. 〈Grid Column CheckBox Values(グリッド・カラムのチェックボックスの値)〉で、[Checked(オン)]および[Un-Checked(オフ)]フィールドに数値を入力し、[OK]をクリックします。
ここでは、GC 変数に設定した値に対応する値を入力する必要があります。
6. 〈Grid Column Properties〉で[OK]をクリックします。
7. 〈Grid Properties〉で[OK]をクリックします。

グリッド・セルへのハイパーリンクの表示

Clickable プロパティは、HTML および Windows プラットフォームで使用可能なグリッド・カラム・レベルのプロパティです。このプロパティによって、特定のカラムのグリッド・セルにテキストをハイパーリンク(クリック可能なテキスト)として表示できます。そのカラムのグリッド・セル内で、ブランク以外の値はすべてハイパーリンクとして表示されます。

Grid Column Clicked(グリッド・カラムのクリック)イベントで、ハイパーリンクがクリックされた時に実行されるロジックを入力します。クリックされたグリッド・ローには GC 値を使用できません。

▶ グリッド・セルにハイパーリンクを表示するには

1. 〈Form Design〉で、作業中のフォーム上でグリッドをクリックし、[Edit]メニューから[Item Properties]を選択します。
2. 〈Grid Properties〉の[Grid Columns]セクションで、グリッド・カラムを選択して[Grid Column Properties]をクリックします。
3. 〈Grid Column Properties〉の[Attributes(属性)]セクションで、[Clickable(クリック可能)]をクリックし、[OK]をクリックします。

Grid Column Clicked イベントを使用して、このグリッド・カラム内でハイパーリンクがクリックされた時に発生する動作を定義します。

4. 〈Grid Properties〉で[OK]をクリックします。

フォーム、グループ・ボックス、またはタブ・コントロールへのハイパーリンクの表示

フォーム、グループ・ボックス、またはタブ・コントロールに配置するハイパーリンクを作成するには、テキスト・ブロック・コントロールを使用する方法と、スタティック・テキスト・コントロールを使用する方法があります。通常、ハイパーリンクのテキスト形式に高度な設定が必要な場合は、テキスト・ブロック・コントロールを使用し、それ以外の場合は、設定項目の少ないスタティック・テキスト・コントロールを使用します。

テキスト・ブロック・コントロールを使用してテキストの内容をハイパーリンクとして表示するには、[Text Block Control Properties(テキスト・ブロック・コントロールのプロパティ)]ダイアログで、ハイパーリンクのテキストを保持するセグメントを追加し、[Clickable]オプションをオンにします。[Text Clicked]イベントに、ハイパーリンクがクリックされた時に処理されるロジックを入力する必要があります。

スタティック・テキスト・コントロールを使用してテキストの内容をハイパーリンクとして表示するには、[Static Text properties(スタティック・テキストのプロパティ)]ダイアログで、表示するハイパーリンクのテキストを割り当てて、[Clickable]オプションをオンにします。[Text Clicked]イベントに、ハイパーリンクがクリックされた時に処理されるロジックを入力する必要があります。クリック可能なテキスト・セグメントのフォントと色は一時変更できません。

▶ テキスト・ブロック・コントロールを使用してハイパーリンクを表示するには

1. 〈Form Design〉で、作業中のフォーム上で、反映させるテキスト・ボックス・コントロールを選択し、[Edit]メニューから[Item Properties]を選択します。
2. 〈Text Block Control Properties〉で、[Add Segment(セグメントの追加)]をクリックし、ハイパーリンクとして使用するテキストを入力します。

Text Clicked イベントを使用して、このハイパーリンクがクリックされた時に発生する動作を定義します。

3. [Segment Information(セグメント情報)]セクションで[Clickable]をクリックし、[OK]をクリックします。

▶ スタティック・テキスト・コントロールを使用してハイパーリンクを表示するには

1. 〈Form Design〉で、作業中のフォーム上で、反映させるスタティック・テキスト・コントロールを選択し、[Edit]メニューから[Item Properties]を選択します。
2. 〈Static Text Properties(スタティック・テキストのプロパティ)〉で、[Static Text(スタティック・テキスト)]フィールドにテキストを入力します。
Text Clicked イベントを使用して、このハイパーリンクがクリックされた時に発生する動作を定義します。
3. [Attributes]セクションで[Clickable]をクリックし、[OK]をクリックします。

フォームへのカスタム HTML タグの挿入

フォームに独自の HTML を挿入して、HTML の外観をカスタマイズすることができます。そのためには、テキスト・ブロック・コントロールを使用します。

テキスト・ブロック・コントロールを使用すると、HTML プラットフォームで既に生成されている機能を拡張できます。HTML プラットフォームでは、そこに含まれるテキストが生成時にフォームの HTML に追加されます。また、実行時にコントロール内のテキストを追加、削除、更新できます。このコントロールによって、HTML クライアントの機能を大幅に拡張できます。

テキスト・ブロック・コントロールにテキストを入力するには、FDA でコントロール内のテキスト・セグメントを定義する方法と、実行時にシステム関数を介してテキスト・セグメントを挿入する方法があります。

FDA を使用したカスタム HTML の挿入

フォームにテキスト・コントロール・ブロックを配置し、必要に応じてタグを保持するセグメントを挿入します。データによって変動するセグメントと、スタティック HTML タグを含むセグメントは分離する必要があります。データによって変動するセグメントは Dialog is Initialized(ダイアログの初期化)、Grid Record is Fetched、または Update Segment()システム関数を呼び出してその他の適切なイベントで設定することができます。

▶ FDA を使用してカスタム HTML を挿入するには

1. 〈Form Design〉で、[Insert(挿入)]メニューから[Text Control Block(テキスト・コントロール・ブロック)]を選択します。
2. コントロールを配置するフォームをクリックします。
3. 新規コントロールをクリックし、[Edit]メニューから[Item Properties]を選択します。
4. [Add Segment]をクリックし、フォームに追加する HTML テキストを入力します。
5. セグメントをクリック可能にするには、[Clickable]をクリックします。

このオプションをオンにすると、ユーザーがクリックした時に Text Clicked イベントを実行するハイパーリンクとしてセグメントが生成されます。

6. セグメントのカスケード・スタイルシート(CSS)設定を一時変更するには、次の手順で操作します。
 - a. [Use Default Font and Color(デフォルトのフォントと色を使用)]をオフにします。
 - b. [Font and Color(フォントと色)]をクリックします。

- c. 〈Font(フォント)〉で、使用するフォント、フォント特性および色を指定して[OK]をクリックします。

このオプションは多用しないことをお勧めします。

7. [OK]をクリックします。

実行時のカスタム HTML の挿入

次のシステム関数を使用すると、テキスト・ブロック・コントロールを ER から操作できます。

関数	パラメータ	備考
Add Segment	Text Control Text Font Clickable(true/false) SegmentId - 追加するセグメントについて戻される固有のセグメント ID	
Get Last Clicked Segment	Text Control SegmentId - ユーザーが最後にクリックしたセグメントについて戻されるセグメント ID	このシステム関数をテキスト・コントロールの Text Clicked イベントに使用して、どのセグメントがクリックされたかを判別します。
Get Segment Information	Text Control Text - セグメント ID で渡される、戻されたテキスト Clickable - セグメント ID で渡される、戻されたクリック可能フラグ SegmentId - 照会対象となるセグメントのセグメント ID	
Remove Segment	Text Control SegmentId - 削除対象となるセグメントのセグメント ID	
Update Segment	Text Control Text Font Clickable SegmentId - 更新対象となるセグメントのセグメント ID	

パラメータ Clickable および SegmentId は、Integer 型の変数となります。

上級機能

フォントと色の適用(指定した場合)、およびクリック可能イベントの処理に必要なタグの追加を除き、このコントロールに入力されたテキストは、生成時にフォームの HTML にフィルタリングなしで追加されます。したがって、HTML の生成に使用されている CSS(カスケード・スタイルシート)スキームを十分に理解する必要があります。中の HTML タグに適切なクラス名を使用することで、テキスト・ブロック・コントロールの外観がフォームと同じになり、カスタマーが制御用の CSS フォームを変更すると、フォームの残りの部分の外観が変化します。また、開発者は、必要でない限りテキスト・セグメントのフォントと色を定義する必要がなくなります。テキスト・セグメントのフォントと色については、フォームの CSS 定義が無視されるため、常にフォームの残りの部分とは異なる表示状態になります。

注:

現在、FDA は適切な CSS 設定を判断するために Web サーバーとやりとりしないため、FDA 内のテキスト・ブロック・コントロールは正常にフォーマットされません。適切にフォーマットされたコントロールを表示するには、そのフォームを生成してから HTML プラットフォーム上で表示します。

次の表は、共通の CSS タグを示しています。

クラス名	適用対象	備考
Padded	テーブル	ページ幅に合わせて埋め込まれたテーブルを作成します。
Border	汎用	
NoBorder	汎用	
WideTable	汎用	幅 = 100%
TallTable	汎用	高さ = 100%
TallAndWideTable	汎用	幅 = 高さ = 100%
Grid	テーブル	
MainHeading	汎用	
SectionHeading	汎用	
GroupHeading	汎用	
SubHeading	汎用	
FieldLabel	汎用	
RaisedBorders	汎用	
BlackBorders	汎用	
ClearBorders	汎用	
QBECell	汎用	
GridHeaderCell	汎用	
GridCell	汎用	

クラス名	適用対象	備考
InYourFaceError	汎用	
InYourFaceWarning	汎用	
ToolBarText	汎用	
GroupBox	汎用	
GroupBoxHeader	汎用	
FormLabel	汎用	
FormAboveGrid	汎用	

次の HTML タグは、テキストが正常にフォーマットされるという保証付きで使用できるように、カスタム・スタイル・タグと共に指定されています。

タグ	備考
HR1	
HR2	
BODY	テキスト・ブロック・コントロールによって前後のタグ、クラス、またはその両方が入力されていない未フォーマットのテキストには、基になるベース・スタイルが必要です。
INPUT	
SELECT	
A	
TABLE	
TABLE TD	

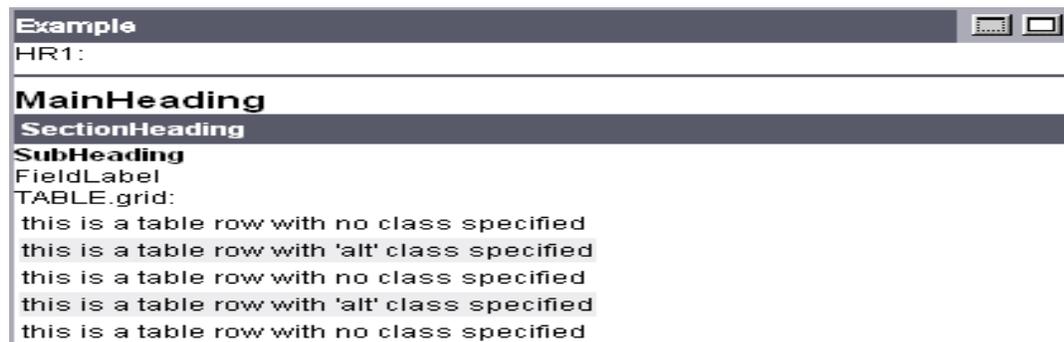
次のサンプル HTML コードでは、コードの後に示す図が生成されます。

```

Text in text block control :
<HR1>
HR1:
</HR1>
<HR class=MainHeading>
MainHeading
</HR>
<HR class=SectionHeading>
SectionHeading
</HR>
<HR class=SubHeading>
SubHeading
</HR>

```

```
<span class=FieldLabel>
SubHeading
TABLE.grid:
</span>
<TABLE class=Grid>
<TR><TD >
this is a table row with no class specified
</TD></TR>
<TR class=alt><TD >
this is a table row with no class specified
</TD></TR>
<TR><TD >
this is a table row with no class specified
</TD></TR>
<TR class=alt><TD >
this is a table row with no class specified
</TD></TR>
<TR><TD >
this is a table row with no class specified
</TD></TR>
```



フォームへのイメージの挿入

ビットマップ・コントロールを使用すると、フォーム上にさまざまなイメージを配置できます。イメージには、スタティック・イメージとクリック可能イメージがあります。Button Clicked イベントを使用して、イメージがクリックされた時に発生する動作を定義します。

▶ フォームにイメージを挿入するには

1. <Form Design>で、[Insert]メニューから[Bitmap(ビットマップ)]を選択します。
2. コントロールを配置するフォームをクリックします。

3. <Bitmap Properties(ビットマップ・プロパティ)>で、[Find]をクリックし、ビットマップ・コントロール用のファイルを検索します。
選択したビットマップが<Bitmap Properties>フォームに表示されます。
4. 次のフィールドに入力します。
 - [Tool Tip Text(ツール・ヒント・テキスト)]
このテキストは、ユーザーがマウス・ポインタをビットマップ上に動かすと、ポップアップ・ヒントと同様に表示されます。
5. 次の1つまたは複数の属性をクリックします。
 - Clickable
[Clickable]オプションをオンにすると、ビットマップに対する Button Clicked イベントが有効化されます。
 - Maintain Aspect Ratio(縦横比を固定する)
[Maintain Aspect Ratio]オプションをオンにすると、ビットマップの縦横比はサイズを変更しても変わりません。これにより、サイズ変更してもイメージが歪まなくなります。

フォームへのドロップダウン・リストの挿入

フォームにドロップダウン・リストを表示するには、フォームにコンボ・ボックス・コントロールを配置する必要があります。次に、データ辞書項目をコンボ・ボックス・コントロールの右にドラッグ・アンド・ドロップします。この操作を完了すると、<Combo Box Properties(コンボ・ボックスのプロパティ)>ダイアログに接続先のデータ辞書項目が表示されます。

ドロップダウン・リストの内容は、データ辞書項目が関連付けられている UDC 値と記述からのみ導出されます。リストは、データ辞書項目の編集ルール・プロパティに基づくものになります。コンボ・ボックスをユーザー定義コードのみ入力可能にすることにより、ユーザーが指定したコンボ・ボックスの値がシステムによって正しく解釈されます。

▶ フォームにドロップダウン・リストを挿入するには

1. <Form Design>で、[Insert]メニューから[Combo Box(コンボ・ボックス)]を選択します。
2. コントロールを配置するフォームをクリックします。
3. [View(表示)]メニューから、[Data Dictionary Browser(データ辞書ブラウザ)]を選択します。
4. <Data Dictionary Browser>で、コンボ・ボックス・コントロールに使用するデータ辞書項目を検索します。
5. コンボ・ボックス・コントロールを選択し、<Data Dictionary Browser>からデータ辞書項目をドラッグして、コンボ・ボックス・コントロールの真上にドロップします。
6. コンボ・ボックス・コントロールを選択し、[Edit]メニューから[Item Properties]を選択します。
7. データ辞書項目がコンボ・ボックスに関連付けられたことを確認し、[OK]をクリックします。

グリッドのカスタム選択とソートの実行

[Find]メニュー・エグジットの Button Clicked イベント内で、Clear Select()システム関数を使用してから Set Selection()システム関数を実行すると、グリッドの選択条件をカスタマイズできます。また、Clear Sequencing()、Set Sequencing()の順にシステム関数を実行し、グリッドの順序設定をカスタマイズするように選択することもできます。

Set Selection()または Set Sequencing()システム関数を使用すると、カスタム・グリッド選択条件を実行するためにフォーム上で非表示のフィルタ・フィールドを使用する必要がありません。また、Set Selection()関数では、生成される SQL 文中の「Or」および「And」文をサポートできます。

Set Selection()システム関数は、フォームのビジネス・ビューに存在するフィールドに依存しません。そのため、ビジネス・ビューと同じテーブルに属していれば、実際にはビジネス・ビューの外部にあるフィールドの選択を実行できます。

Web クライアントでのイベント

HTML 環境では、テーブル更新の関係で 2 つのソフトウェア・イベントは正常に動作しません。コントロールの表示/非表示やグリッドの有効化/無効化には、Control Is Exited イベントを使用しないことをお勧めします。Control Is Exited イベントで Hide/Show または Enable/Disable Grid システム関数を使用すると、システムではイベントを直ちに処理できるように、ページをリフレッシュする JavaScript が自動的に生成されます。この方法を使用すると、ラウンドトリップが発生するため注意してください。

次の表は、データ送信 (ポスティング) をトリガーするイベントなど、すべてのソフトウェア・イベントに対する HTML クライアントの応答を示しています。

オブジェクト・タイプ	イベント	更新のトリガー	遅延
ビットマップ	Button Clicked	Yes	No
チェックボックス	Selection Changed	No。ただし、ER が存在する場合は更新がトリガーされます。	—
コンボ・ボックス	Control is Exited	No	Yes
コンボ・ボックス	Control is Entered	No	Yes。更新前にフォームの状態に対して実行されます。
ドキュメント・オブジェクト	Media Objects – Form def	— (フォーム初期化中にサーバー上でトリガーされます)	—
ドキュメント・オブジェクト	Media Objects – Row def	— ([Find] ボタンの処理中にサーバー上でトリガーされます)	—
編集	Control Exited/Changed—Asynch	No。ただし、イベント・フラグ HTML Post がオンになっている場合は更新がトリガーされます。	Yes
編集	Control Exited/Changed—Inline	No。ただし、イベント・フラグ HTML Post がオンになっている場合は更新がトリガーされます。	Yes

オブジェクト・タイプ	イベント	更新のトリガー	遅延
編集	Control is Entered	No	Yes。更新前にフォームの状態に対して実行されます。
編集	Control is Exited	No。ただし、イベント・フラグ HTML Post がオンになっている場合は更新がトリガーされます。	Yes
編集	Post Visual Assist Clicked	－（イベント Visual Assist Button Clicked の終了直後にサーバー上でトリガーされます）	－
編集	Visual Assist Button Clicked	Yes	No
フォーム	Add Record to DB – After	－（[OK]ボタンの処理中にサーバー上でトリガーされます）	－
フォーム	Add Record to DB – Before	－（[OK]ボタンの処理中にサーバー上でトリガーされます）	－
フォーム	Clear Screen After Add	－（フォーム初期化中にサーバー上でトリガーされます）	－
フォーム	Clear Screen Before Add	－（フォーム初期化中にサーバー上でトリガーされます）	－
フォーム	Dialog is Initialized	－（フォーム初期化中にサーバー上でトリガーされます）	－
フォーム	Director Back Button Clicked	N/A (HTML クライアント上では使用できません)	N/A
フォーム	Director Cancel Button Clicked	N/A (HTML クライアント上では使用できません)	N/A
フォーム	Director Next Button Clicked	N/A (HTML クライアント上では使用できません)	N/A
フォーム	End Dialog	－（フォームのクローズ中にサーバー上でトリガーされます）	－
フォーム	Grid Record is Fetched	－（[Find]ボタンの処理中にサーバー上でトリガーされます）	－
フォーム	Last Grid Record Has Been Read	－（[Find]ボタンの処理中にサーバー上でトリガーされます）	－
フォーム	Post Commit	－（[Find]ボタンの処理中にサーバー上でトリガーされます）	－
フォーム	Post Dialog is Initialized	－（フォーム初期化中にサーバー上でトリガーされます）	－

オブジェクト・タイプ	イベント	更新のトリガー	遅延
フォーム	Update Record to DB – After	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
フォーム	Update Record to DB – Before	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
フォーム	Write Grid Line–After	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
フォーム	Write Grid Line–Before	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	Add Grid Rec to DB – After	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	Add Grid Rec to DB – Before	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	Add Last Entry Row to Grid	– ([Find]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	All Grid Recs Added to DB	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	All Grid Recs Deleted from DB	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	All Grid Recs Updated to DB	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	Delete Grid Rec from DB– After	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	Delete Grid Rec from DB– Before	– ([OK]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	Delete Grid Rec Verify– After	– ([Delete]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	Delete Grid Rec Verify– Before	– ([Delete]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	Double Click on Row Header	Yes	No
グリッド	Get Custom Grid Row	– ([Find]ボタンの処理中にサーバー上でトリガーされます)	–
グリッド	Kill Focus on Grid	No	Yes

オブジェクト・タイプ	イベント	更新のトリガー	遅延
グリッド	Row Exit & Changed - Asynch	Yes。グリッドのフラグ [Allow Multi-line Edit] がオンになっている場合、フォームはリフレッシュされずにバックグラウンドで更新が行われ、実際には更新された場合でも更新されなかったように見えます。	No
グリッド	Row Exit & Changed - Inline	Yes。グリッドのフラグ [Allow Multi-line Edit] がオンになっている場合、フォームはリフレッシュされずにバックグラウンドで更新が行われ、実際には更新された場合でも更新されなかったように見えます。	No
グリッド	Row is Entered	No	Yes。更新前にフォームの状態に対して実行されます。
グリッド	Row is Exited	Yes。グリッドのフラグ [Allow Multi-line Edit] がオンになっている場合、フォームはリフレッシュされずにバックグラウンドで更新が行われ、実際には更新された場合でも更新されなかったように見えます。	No
グリッド	Set Focus on Grid	随時。ユーザーが編集可能グリッド内でセルをクリックすると、更新が行われます。ユーザーが編集可能グリッドにタブ移動すると、イベントが遅延されます。	随時。ユーザーが編集可能グリッド内でセルをクリックすると、更新が行われます。ユーザーが編集可能グリッドにタブ移動すると、イベントが遅延されます。
グリッド	Update Grid Rec to DB - After	- ([OK] ボタンの処理中にサーバー上でトリガーされます)	-
グリッド	Update Grid Rec to DB - Before	- ([OK] ボタンの処理中にサーバー上でトリガーされます)	-
グリッド・カラム	Col Exited & Changed - Asynch	No。ただし、イベント・フラグ HTML Post がオンになっている場合は更新がトリガーされます。	Yes
グリッド・カラム	Col Exited & Changed - Inline	No。ただし、イベント・フラグ HTML Post がオンになっている場合は更新がトリガーされます。	Yes
グリッド・カラム	Col is Exited	No。ただし、イベント・フラグ HTML Post がオンになっている場合は更新がトリガーされます。	Yes
グリッド・カラム	Grid Column Clicked	Yes	No
グリッド・カラム	Post Visual Assist Clicked	- (イベント Visual Assist Button Clicked の終了直後にサーバー上でトリガーされます)	-

オブジェクト・タイプ	イベント	更新のトリガー	遅延
グリッド・カラム	Visual Assist Button Clicked	Yes	No
ハイパー項目	Button Clicked	Yes	No
ハイパー項目	Post Button Click – Asynch	— (ハイパー項目のイベント Button Clicked が発生した直後にサーバー上でトリガーされます)	—
ハイパー項目	Post Button Clicked	— (ハイパー項目のイベント Button Clicked が発生した直後にサーバー上でトリガーされます)	—
親/子	Kill Focus On Control	No	Yes
親/子	Set Focus on Control	No	Yes
親/子	Tree – Begin Drag Operation	Yes	No
親/子	Tree – Cancel Drag Drop	Yes	No
親/子	Tree – Drag Over Node	Yes	No
親/子	Tree – End Drag Drop Operation	Yes	No
親/子	Tree Node Is Collapsing	Yes	No
親/子	Tree Node Is Deleted	Yes	No
親/子	Tree Node Is Expanding	Yes	No
親/子	Tree Node Selection Change	No。ただし、イベント・フラグ HTML Post がオンになっている場合は更新がトリガーされます。	Yes
親/子	Tree–Node Level Changed	No。ただし、イベント・フラグ HTML Post がオンになっている場合は更新がトリガーされます。	Yes
親/子 – グリッド	All Grid Recs Deleted from DB	— ([OK]ボタンの処理中にサーバー上でトリガーされます)	—
親/子 – グリッド	Delete Grid Rec from DB– After	— ([OK]ボタンの処理中にサーバー上でトリガーされます)	—
親/子 – グリッド	Delete Grid Rec from DB– Before	— ([OK]ボタンの処理中にサーバー上でトリガーされます)	—
親/子 – グリッド	Delete Grid Rec Verify– After	— ([Delete]ボタンの処理中にサーバー上でトリガーされます)	—
親/子 – グリッド	Delete Grid Rec Verify– Before	— ([Delete]ボタンの処理中にサーバー上でトリガーされます)	—

オブジェクト・タイプ	イベント	更新のトリガー	遅延
親/子 - グリッド	Double Click on Row Header	Yes	No
親/子 - グリッド	Get Custom Tree Node	- ([Find]ボタンの処理中にサーバー上でトリガーされます)	-
プッシュボタン	Button Clicked	Yes	No
プッシュボタン	Post Button Clicked	- (プッシュボタンのイベント Button Clicked が発生した直後にサーバー上でトリガーされます)	-
ラジオ・ボタン	Selection Changed	No。ただし、ER が存在する場合は更新がトリガーされます。	-
スタティック・コントロール	Text Clicked	Yes	No
タブ・ページ	Tab Page is Initialized	- (タブ・ページが初めて選択されると、タブ・ページのイベント Tab Page is Selected が処理される前にサーバー上でトリガーされます)	-
タブ・ページ	Tab Page is Selected	Yes	No
テキスト・ブロック	Text Clicked	Yes	No
ツリー・コントロール	Double Click on Leaf Node	No。ただし、ER が存在する場合は更新がトリガーされます。	-
ツリー・コントロール	Get Custom Tree Node	- (ツリー・コントロールのイベント Tree Node Is Expanding の発生中にサーバー上で処理されます)	-
ツリー・コントロール	Kill Focus On Tree	No	Yes
ツリー・コントロール	Set Focus On Tree	No	Yes
ツリー・コントロール	Tree Node Is Collapsing	Yes	No
ツリー・コントロール	Tree Node Is Deleted	Yes	No
ツリー・コントロール	Tree Node Is Expanding	Yes	No
ツリー・コントロール	Tree Node Selected	No。ただし、イベント・フラグ HTML Post がオンになっている場合は更新がトリガーされます。	Yes

モバイル用アプリケーションの設計

SP15 以降の HTML クライアントでは、HTML クライアント・アーキテクチャとの高度な互換性を備えた Windows CE プラットフォームに準拠するモバイル・デバイスが全面的にサポートされます。現在、すべての Windows CE デバイスでは、各種バージョンの Internet Explorer(IE)ブラウザを介して HTML が適切にサポートされています。

EnterpriseOne アプリケーションでは、モバイル・デバイスについて次のアプリケーションのサポートが用意されています。

- IE 4.x ブラウザを実行する Windows CE 3.0 デバイス
この種のデバイスには、HP Jornada 720、NEC Mobile Pro™ 790 などがあります。バージョン 3.0 は Windows CE の最新バージョンを表し、IE 4.x ブラウザによる HTML4 スペックのサポートが用意されているため、Windows CE 2.11 よりも 3.0 準拠のデバイスをお勧めします。IE 4.x は、モバイル・デバイス上で現在使用可能なブラウザの中では最も豊富な機能を備えており、DTML と JavaScript に対する適切なサポートも組み込まれています。
- IE 3.01 を実行する Windows CE 2.11 SP1 以上
HTML クライアントでは、この種のデバイスがサポートされます。ただし、この種のデバイスは後継の Windows CE 3.0 よりも古いテクノロジーを表すため、お勧めしません。
- Pocket PC 2000 を実行する Windows CE 3.0
この種のデバイスには、Compaq IPAQ、HP Jornada 540、Casio Cassiopeia などがあります。この種のデバイスには、IE 3.01 ブラウザに基づく HTML アクセス機能も用意されています。HTML クライアントでは、これらのデバイスが全面的にサポートされます。

モバイル・デバイス・アーキテクチャでは、サポート対象プラットフォーム上で実行されるすべての対話型アプリケーションがサポートされます。また、既存のアーキテクチャが活用されており、独自アプリケーションの開発を変更する必要はありません。現在、モバイル・サポートを利用できるのは接続ユーザーのみで、ブラウザ経由でネットワークに接続していないユーザーはサポート対象外です。Palm Computing プラットフォームには、適切なブラウザ・サポートが欠けているため、このサポートは利用できません。

モバイル・デバイスのランタイム・アーキテクチャ

HTML クライアントでモバイル・デバイスのサポートを提供するアーキテクチャは、デスクトップの場合と同じです。HTML4 準拠のブラウザの場合、出力の生成も実際には変わりません。ただし、IE 3.01 を実行する Windows CE ブラウザ用の出力が生成されます。

ユーザーが最初に JAS サーバーにサインオンすると、クライアント・ブラウザと Web サーバー間のセッションが確立されます。このセッションでは、クライアント・プラットフォームとブラウザに関する情報が保持されます。システムでは、この情報に基づいて HTML 出力が生成されます。

モバイル・デバイスの設計方針

モバイル・デバイス用のアプリケーションを開発する場合の設計方針は、通常の Web ベース・アプリケーションの場合と同じです。ただし、これらのデバイスではフォームの画面解像度に制約があることに注意してください。フォームの画面解像度には、Windows CE 3.0 デバイスの場合は 640×480、Pocket PC の場合は 240×320 という制約があります。次のリストは、その他に考慮する必要のある設計方針を示しています。

- ブラウザベースのアプリケーションの場合、水平方向のスクロールよりも垂直方向のスクロールの方が一般的です。
- アプリケーションが使用しやすいように、Pocket PC インターフェイスの場合は、コントロールを上から下へと重ねて表示することを考慮します。ユーザーが過度のスクロールを行わないように、グリッドのカラム数を制限できます。
- SP16.1 以降では、HTML4 準拠のブラウザ上でのグリッド・タブがサポートされます。ただし、Pocket PC 上ではグリッド・タブはサポートされません。グリッド・タブには、モバイル・デバイスのデフォルトというオプションがあります。このオプションを使用すると、既存のアプリケーションについて Pocket PC に表示されるグリッドのカラム数を制限できます。

▶ モバイル・デバイスに特定のグリッド・フォーマットを使用するには

1. 編集可能グリッドが表示されている HTML フォーム上で、[グリッドのカスタマイズ]をクリックします。
2. <グリッドのカスタマイズ>で、[使用可能なフォーマット]リストからフォーマットを選択します。
3. グリッド・フォーマットをモバイル・デバイスのデフォルト・フォーマットにするには、[モバイル・デバイスのデフォルト]をクリックします。
4. [閉じる]をクリックします。

HTML デバイスとモバイル・デバイスの機能上の相違点

モバイル・デバイスと HTML では、エラーの管理方法が異なります。また、モバイル・デバイスと端末に表示される Web ブラウザでは、ユーザー操作も異なっています。

イベント処理

HTML クライアントとモバイル・デバイスの場合、フォーム・イベントの管理方法は同じです。ただし、Pocket PC の場合は、ユーザーが関連フィールド内のデータを変更するまで、コントロールに対する Control Is Exited イベントは処理されません。デスクトップ上の IE 5.x や Windows CE 3.0 上の IE 4.x など、HTML4 準拠のブラウザでは、ユーザーが [Tab] キーを使用して移動するすべてのコントロールが JavaScript を使用して追跡されます。この情報は Web サーバーに送信され、Control Is Exited イベントの実行に使用されます。ただし、Pocket PC 上の IE 3.01 ブラウザの場合は、フォームの仮想イメージが仮想クライアントによって保管され、このイメージを使用して変更対象フィールドが比較されます。したがって、Control Is Exited とその関連イベント (Control is Entered、Control is Exited、Changed など) は、フィールドが修正される場合にのみ処理されます。アプリケーションのロジックは、特定のフィールドが修正されない場合にも、Control Is Exited の処理に依存しないようにする必要があります。

有用性

モバイル・デバイス上の場合、ブラウザ上での DHTML と JavaScript のサポートが限られているため、これらのデバイス上で実行されるアプリケーションの外観は、デスクトップの IE 5.x ブラウザほど豊富ではありません。次のリストは、重要な相違点を示しています。

- モバイル・デバイスには、グリッドのスクロール・バーのサポートは存在しません。
- Pocket PC 上の HTML クライアントの場合は、キーボードのサポートに制約があります。

- Pocket PC には、複数行のグリッド編集機能がありません。そのため、これらのデバイスは大量のデータ入力には適しません。
- メディア・オブジェクトの RTF 編集と OLE オブジェクトのサポートは、存在しません。
- モバイル・デバイスでは、エクスポート/インポート機能はサポートされません。
- モバイル・デバイスでは、テキスト・ブロック・コントロールのサポートに制約があります。アプリケーションに、DHTML/JavaScript に大幅に依存するテキスト・ブロック・コントロールがあると、Pocket PC プラットフォーム上では正常に動作しない場合があります。
- 投入済みレポートの PDF ファイルの表示サポートは存在しません。

マルチ・アプリケーション・フレームワーク

エンドユーザーは、しばしば同時に複数のアプリケーションで作業する機能を必要とします。これが Web クライアント・ユーザーの場合は、アプリケーションの起動に Multiple Application Framework (MAF)を使用する必要があります。MAF により、同時に複数のブラウザ・セッションが使用可能になり、ランタイム・データを適切なアプリケーションに一致させることができます。

MAF は、呼出し元(タスク・エクスプローラやポータルなど)と呼出し先アプリケーションの間で、コーディネータの役割を果たします。MAF Launcher を使用することで、言語の不一致や他の操作ハンドルの間わず、MAF はソースからターゲットへのデータ転送を容易にします。MAF Launcher は、ソースであるポータルおよびタスク・エクスプローラのアンカーを、ターゲットである EnterpriseOne アプリケーション用のリソース・バンドルを含む Java オブジェクトと一致させています。

MAF を介してアプリケーションを起動すると、ソースとターゲットの間でのデータ交換が容易になります。また、ユーザーがソースを含むブラウザを閉じると、後でそのソースを含む新規ブラウザをターゲット・ブラウザから起動できます。ユーザーがブラウザを閉じてターゲットを終了すると、対応するポイントが MAF により削除されます。このポイントは、キュー内のソース・オブジェクトを指し、ソースからターゲットへのリンクを維持するものです。

システムを活用するには、MAF を介してアプリケーションを起動する必要があります。

注:

アプリケーションは、MAF コード構造ではコンポーネントとして参照されます。

MAF コンポーネントの実装

MAF コンポーネントを実装するには、2 つの抽象クラス MAFComponent および JASMAFComponentInstance を拡張します。コンポーネント・インスタンスが MAFComponent のコンテキストの外で作成されることはないため、MAFComponent の実装ではインナー・クラスとして JASMAFComponentInstance の実装を含めることができます。また、特定のコンポーネントに関するコードをすべて 1 か所にまとめておくことができます。

MAFComponent の拡張

次のコードにより、MAFComponentObject が拡張されます。この例は、言語には関係しません (newMAFComponent メソッドについては後述します)。

```
import com.jdedwards.base.maf.*;
import com.jdedwards.base.maf.jas.*;
import java.util.*;

public class MyMAFComponent extends MAFComponent
{
```

```

public String getId()
{
    return "JDE_MAF_EXAMPLE";
}

public String getDescription(Locale locale)
{
    //This is not language aware but is just an example
    return "MAF Example Component";
}
...
}

```

最初のメソッドは、このコンポーネント・タイプの固有 ID です。2 番目のメソッドは記述を戻します。このメソッドがロケールを提供することに注意してください。このメソッドを使用すると、ISO 互換の言語コードを指定したり、リソース・バンドルから情報を取り込むことができます。未定義の言語が必要な場合は、ISO コードを参照して EnterpriseOne の言語コードに変換する必要があります。

JASMAFComponentInstance の拡張

次のコードは、上記の例の newComponentInstance メソッドと、JASMAFComponentInstance クラスの実装です。この例では、JASMAFComponentInstance の実装は MAFComponent の実装のインナー・クラスです。

```

import com.jdedwards.base.maf.*;
import com.jdedwards.base.maf.jas.*;
import java.util.*;

public class MyMAFComponent extends MAFComponent
{
    ...

    public MAFComponentInstance newMAFComponentInstance(HttpSession session, String uniqueId, String launcher)
    {
        return new MyMAFComponentInstance(uniqueId, launcher, this);
    }

    private class MyMAFComponentInstance extends JASMAFComponentInstance
    {
        public MyMAFComponentInstance(String uniqueId, String launcher, MAFComponent component)
        {
            super (uniqueId, launcher, component);
        }
    }
}

```

```

public String getData(HttpServletRequest req, HttpServletResponse resp)
{
    return "This is my component's data";
}

public String getDescription (Locale locale)
{
    return "My component instance: "+getUniqueId();
}
}
}
}

```

MAFComponent の登録

MAF コンポーネントを使用する前に、動的 MAFComponentFactory クラスに登録する必要があります。問題の MAFComponent は、そのコンポーネントが実際に処理される前に必要になるため、WebApp の初期化中にコンポーネントを登録する必要があります。次のコード行により、上記の MAFComponent がインスタンス化されて登録されます。

```

MAFComponent comp = new MyMAFComponent();
MAFComponentFactory.register(comp);

```

MAFComponentInfo の取得と使用

インスタンスの初期化中と破棄中を除き、MAF Launcher からオブジェクトに MAFComponentInfo を提供する必要があります。MAFComponentInfo は、属性の HttpServletRequest オブジェクトとなります。このオブジェクトを取得するには、次のコマンドを使用します。

```

MAFComponentInfo info = request.getAttribute(MAFConstants.ATTRIB_COMPONENT_INFO);

```

MAF Close リンクの生成

MAF には、インスタンス内のすべてを閉じるためのリンクが用意されています。これには、Open Components Map からのコンポーネントの削除と、MAFComponentInstance の破棄メソッドの処理が含まれます。このリンクを生成するには (JavaScript を使用可能)、次のコードを使用します。

```

out.println("<a href=" + info.getCloseAnchor() + ">Close</a>");

```

クローズに他の機能も追加したい場合は、破棄メソッドを実装する必要があります。

Closed Launcher との再接続

コンポーネントは、関連付けられた MAFComponentInfo オブジェクト内の情報に基づいて、Launcher にリンクを提供できます。この機能は任意であり、コンポーネントはタグを生成する前にアクティブ・リンクを検索する必要があります。次のコードにより、Launcher 再接続リンクが設定されます。

```

out.println("<a href=" + info.getLauncherAnchor() + ">Goto " + info.getLauncherDesc(locale) + "</a>");

```

固有 ID の取得

MAF システムのバックボーンとなるのは MAF 固有 ID です。特定のセッションで、この固有 ID を使用して、実際に処理されているコンポーネントのインスタンスを判断することができます。固有 ID は、次のコードを使用して取得できます。

```
String uid = info.getUniqueId();
```

ループスルーの実行

MAF の起動後は、外部ウィンドウを使用しやすいようにコンポーネントを監視する必要があります。このループスルーは、次のどちらかの方法で実行できます。最初の方法は最も単純なテクニックで、MAFComponentInfo オブジェクトで wrapURL メソッドを使用します。このテクニックは柔軟ですが、処理できるのはスタティック・リンクのみです。wrapURL メソッドを次のように使用します。

```
Map params = new HashMap();
params.add("myParam", {"oh me", "oh my"});
out.println("<a href=" + info.wrapURL(params) + ">my url</a>");
```

このコードにより、次の例のようなタグが生成されます。

```
<a href="http://jde.com/jde/servlet/com.jdedwards.base.maf.jas.JASMAFServlet?...&myParam=oh+me&myParam=oh+my">my url</a>
```

指定したパラメータは、アンカー内の URL の末尾に表示されることに注意してください。これらのパラメータは、MAFComponentInstance の getData メソッドを使用してコンポーネントに戻されます。

2 番目のテクニックでは、MAFComponentInfo で提供される getMAFLauncherParameterBean を使用します。この方法では、オブジェクト・フォームにコンポーネントの URL とパラメータを提供します。このオブジェクトを使用してスタティック URL を作成できますが、その長所はフォーム投入用のリンクを生成できることにあります。次のコマンドを考えてみます。

```
MAFLauncherParameterBean mlpb = info.getMAFLauncherParameter(new HashMap());
out.println("<FORM action=" + launcherParamBean.getURL() + ">");

Map m = launcherParamBean.getParameterMap();
Iterator iterator = m.keySet().iterator();
while(iterator.hasNext())
{
    String paramName = (String)iterator.next();
    String[] paramValues = (String[])m.get(paramName);
    for (int x = 0; x < paramValues.length; x++)
    {
        out.println("<INPUT type=" + "hidden" + " name=" + paramName + " value=" + paramValues[x] + ">");
    }
}

out.println("Name: <input type=" + "text" + " name=" + "name">");
```

```
out.println("</FORM>");
```

このコードで生成されるフォームには、ユーザーが入力できる Name プロパティが組み込まれ、MAF に必要なパラメータもすべて非表示フィールドとして組み込まれます。

getMAFLauncherParameterBean の方が wrapURL メソッドに比べるとコードは長くなりますが、柔軟性が大幅に向上します。これらのパラメータは、MAFComponentInstances の getData メソッドがアクセスされるときに要求に含まれます。MAF は単純なパラメータ・パススルー・システムを使用して、適切な要求をコンポーネントに転送します。

Web クライアントの構成

Web クライアントに必要な構成作業のほとんどは、JAS サーバーのセットアップ時に完了しています。`.ini` ファイルの設定とシステム・パフォーマンスの調整については、JAS のドキュメンテーションを参照してください。

また、システム・ログインの構成方法には複数の選択肢があります。ポータルも使用している場合は、ログイン構成が両方のシステムに影響します。

ログインの構成

〈Security Workbench(セキュリティ・ワークベンチ)〉を使用してシステムへのアクセスとその機能を設定すると、アクセス権を持つユーザーは各自のユーザーID とパスワードを使用してログインします。翻訳されたバージョンがある場合は、ログイン・フォームの使用言語を定義することができます。JAS.ini ファイルの `OWWEB` セクションで、`InitialLanguageCode` を、使用する言語を表す ISO コードに設定します。この設定によって、ユーザーがログインするまでローカルの使用言語が一時変更されません。

Cookie を使用すると、ローカル・ワークステーションでユーザーのログイン詳細(パスワードなど)を記録できます。これにより、そのユーザーに対するログイン処理が合理化されます。また、匿名ユーザー・アカウントを作成して、ユーザーに匿名によるログインを許可することも可能です。

必要な場合は、標準ログイン機能を使用する代わりに、合理化された基本認証手順を使用できるようにシステムを構成できます。

システムでは、ユーザーのログイン方法に応じて、ユーザー・パスワードの有効期限を示すのみでなく、期限切れが近いことを通知することもできます。ユーザーは、通知を受け取った時点でパスワードを変更することを選ぶか、警告を無視することができます。その時点でパスワードを変更すると、システムではユーザーがログアウトされ、新規パスワードでログインされます。

次のログイン方法では、パスワード条件通知はサポートされていません。

- 匿名
- パラメータ・ベース
- 直接(Cookie 経由)

パスワードの期限切れが近いことを示す通知は、デフォルトで自動的に発行されます。この通知を実行しない場合は、JAS.ini パラメータ `DisablePasswordAboutToExpire` ([LOGIN]セクション)を `TRUE` に設定してください。

匿名ユーザー

システム・インスタンスを実行中の JAS サーバーが許容するように構成されており、ローカルの JAS.ini ファイルが適切に構成されていると、匿名ユーザーのログインを受け入れることができます。匿名ログイン用に、ワークスペースとコンポーネントへのアクセス権を別個のユーザーとして設定します。匿名ユーザーは*PUBLIC グループに属しません。匿名ユーザーを設定することを選んだ場合は、システム・オブジェクトへのアクセス権を明示的に付与する必要があります。

匿名ユーザーを受け入れるように構成されているインスタンスの動作は、通常のインスタンスとは異なります。この種のインスタンスをユーザーが参照すると、ログインが要求される代わりに初期の〈Welcome(ようこそ)〉フォームが表示されます。匿名ユーザー・アカウントにアクセス権が付与されているオブジェクトを表示できます。その場合、ユーザーがユーザー・オプションの変更などのコマンド操作を実行することはできませんが、アカウントを持っていれば通常どおりログインするよう選ぶことができます。ログイン後のユーザーは、そのアカウントに権限が付与されているすべてのオブジェクトとコマンド機能に対してアクセス権を持つことになります。

匿名ユーザー・アカウントは有効なアカウントであり、ユーザーはこの種のアカウントを使用して、通常のアカウントを使用する場合と同じ方法でシステムにログインできます。匿名ユーザー・アカウントには、通常のアカウントの場合と同様に、コマンド機能や他の機能へのアクセス権を付与できます。匿名アカウントを使用してログインしたユーザーは、そのアカウントに付与されているコマンド操作と機能にアクセスできます。

▶ 匿名ユーザー・ログインを受け入れるように JAS サーバー上のシステム・インスタンスを構成するには

1. 匿名ユーザーとして使用するユーザー・アカウントを作成します。
2. 次のように、環境に該当する設定をコマンド行パラメータに追加します。
 - Tomcat/WebSphere 3.5 の JVM については、次の設定を追加します。
 - Danon.user.oid=USR12345
USR12345 は、匿名ユーザー・アカウントに明示的に設定する固有のユーザーID です。
 - Danon.user.pwd=PASS12345
PASS12345 は、匿名ユーザー・アカウントに使用するパスワードです。
 - Danon.user.env=ENV12345
ENV12345 は、匿名ログインを許可するシステム・インスタンスです。

Tomcat を使用する場合は、これらのパラメータを Project|Project Properties|Run|VM パラメータに追加してください。

- WebSphere 4.0 の場合は、Web サーバー用に次のランタイム変数を追加します。

名前	値
anon.user.oid	匿名ユーザー・アカウントに明示的に設定する固有のユーザーID
anon.user.pwd	匿名ユーザー・アカウントに使用するパスワード
anon.user.env	匿名ログインを許可するシステム・インスタンス

3. コマンド行引数を環境に合わせて更新します。
 - a. WebSphere 3.5 を使用している場合は、WebSphere Administrative Console を使用して [Application Server(アプリケーション・サーバー)]メニュー・レベルにアクセスし、[General(一般)]タブで[Command Line Arguments(コマンド行引数)]を次のように更新します。

-Danon.user.oid=[USR] -Danon.user.pwd=[PWD] -Danon.user.env=[ENV]

- b. WebSphere 4 を使用している場合は、次の手順で操作します。
 - i. [Application Server] の [JVM Settings (JVM 設定)] タブに移動します。
 - ii. [System Properties (システム・プロパティ)] セクションに、anon.user.oid、anon.user.pwd および anon.user.env 項目とそれぞれの値を追加します。
 - iii. [Add (追加)]、[Apply (適用)] を順にクリックします。
コマンド行引数が更新されます。
4. JAS.ini ファイルの [OWWEB] セクションに次の 1 行を追加します。

```
AnonAccess=true
```
5. ユーザーがポータルの標準ログイン・プロセスを起動できるように、JAS.ini ファイルの [PORTALCONFIGURATION] セクションに次の 1 行を追加してハイパーリンクを作成します。

```
ShowSignin=TRUE
```

この機能を使用可能にすると、[Workspace Navigation Bar (ワークスペース・ナビゲーション・バー)] にログイン用ハイパーリンクが表示されます。匿名でログインしたユーザーは、このハイパーリンクを使用してポータルの標準ログイン・プロセスを起動できます。この方法では、ポータル・アカウントを持つユーザーは、匿名でログインしてから、通常のユーザーとしてログインできます。
6. 変更結果が有効になるように、サーバー上でアプリケーション・サーバー・インスタンスを再起動します。

直接ログイン

Cookie を使用すると、ユーザーのユーザー名、環境、ロール、言語、さらにパスワードを記録して、そのユーザーが次回にログインするときのログイン・プロセスを合理化することができます。

ユーザー名、環境、ロール、および言語を記録する(ただし、パスワードは記録しない)ようにシステムを構成するには、JAS.ini ファイルの SECURITY セクションで次の 1 行を追加または編集します。

```
UseLogonCookie=TRUE
```

Cookie の期限が切れる前にユーザーがログオンすると、パスワードのみを求めるプロンプトが表示されます。

パスワードを含めてユーザー関連情報をすべて記録するようにシステムを構成するには、JAS.ini ファイルの SECURITY セクションで次の 1 行を追加または編集します。

```
UseLogonCookie=DIRECT
```

Cookie の期限が切れる前にユーザーがログオンすると、ログイン・プロセスはユーザーに対して透過的に実行され、システムが直接起動するようになります。この設定では、パスワードの記録に標準的な暗号鍵が使用されます。ただし、パスワードを記述する際には独自の暗号鍵を使用できます。暗号鍵を設定するには、JAS.ini ファイルの LOGIN セクションで次の 1 行を追加または編集します。

```
PassKey=[alphanumeric key]
```

また、Cookie の有効期限を設定することも可能です。Cookie の有効期限が切れると、その情報が削除されます。Cookie の有効期限が切れた後にログインするユーザーは、ログイン情報を再入力

する必要があります。Cookieの有効期限を設定するには、JAS.iniファイルのSECURITYセクションで次の1行を追加または編集します。

CookieLifeTime=[time in days]

基本認証

システムは、デフォルトのログイン・プロセスの代わりに汎用のWebログイン・プロセスを使用するように構成することができます。基本認証を使用するとログイン・プロセスは合理化されますが、ユーザーはロールや環境を指定できません。そのため、基本認証が有効になっている場合は、ユーザー全員がデフォルト・ロールでデフォルト環境にログインすることになります。

一般的に、基本認証はポータルにログインするサードパーティ製品で使用されます。基本認証を使用しているサードパーティ製品は、ポータルに対して基本認証ヘッダーを発行する必要があります。

基本認証を有効にするには、サーバーのJAS.iniファイルの該当するセクションで次の各行を追加または編集します。

セクション	設 定
SECURITY	SSOEnabled=TRUE
OWWEB	DefaultEnvironment=[an environment]
OWWEB	DefaultRole=[a role]

