Magic eDeveloper V10



Magic eDeveloper V10 コーディングサンプル Version 2 (コンポーネント編)



Enabling Business with Superior Technology

本書および添付サンプル(以下、本製品)の著作権は、マジックソフトウェアジャパン株式会社(MSJ)にあります。MSJ の書面による事前の許可なしでは、いかなる条件下でも、本製品のいかなる部分も、電子的、機械的、撮影、録音、その他のいかなる手段によっても、コピー、検索システムへの記憶、電送を行うことはできません。

本製品の内容につきましては、万全を期して作成していますが、万一誤りや不正確な記述があったとしても、MSE(Magic Software Enterprises Ltd.)および MSJ はいかなる責任、債務も負いません。本製品を使用した結果、または使用不可能な結果 生じた間接的、偶発的、副次的な損害(営利損失、業務中断、業務情報の損失などの損害も含む)に関し、事前に損害の可能性 が勧告されていた場合であっても、MSE および MSJ、その管理者、役員、従業員、代理人は、いかなる場合にも一切責任を負い ません。MSE および MSJ は、本製品の商業価値や特定の用途に対する適合性の保証を含め、明示的あるいは黙示的な保証は 一切していません。

本製品に記載の内容は、将来予告なしに変更することがあります。

サードパーティ各社商標の引用は、MSE および MSJ の製品に対する互換性に関しての情報提供のみを目的としてなされるものです。一般に、会社名、製品名は各社の商標または登録商標です。

本製品において、説明のためにサンプルとして引用されている会社名、製品名、住所、人物は、特に断り書きのないかぎり、すべて架空のものであり、実在のものについて言及するものではありません。

初版 2007年9月19日

マジックソフトウェア・ジャパン株式会社

目次

3コンポーネント	
3.1 コンポーネントの構成	4
3.2 コンポーネントファイルの配置	6
3.2.1 ディレクトリ構造	6
3.2.2 プロジェクトソースファイル	6
3.2.3 コンポーネントキャビネットファイル	7
3.2.4 コンポーネントファイルの指定	7
3.3 共通コンポーネントの内容	
3.3.1 ユーザ定義イベント	
3.3.2 モデルリポジトリ	
3.3.3 関連プログラム	
3.3.4 ユーザ定義関数	
uMsgBoxFull	
uYesNoDlg	
uTaskModeString	
3.4 データコンポーネントの内容	
3.4.1 モデルリポジトリ	
3.4.2 データリポジトリ	
3.4.3 プログラムリポジトリ	
3.5 受注コンポーネントの内容	
3.5.1 モデルリポジトリ	
3.5.2 データリポジトリ	
3.5.3 プログラムリポジトリ	
3.6 ホストコンポーネントの内容	
3.6.1 モデルリポジトリ	
3.6.2 データリポジトリ	
3.6.3 プログラムリポジトリ	
3.6.4 アプリケーション特性	
3.7 モジュール定義	
3.7.1 モジュールの表示	
3.7.2 モジュールのネスト	
3.7.3 モジュールの用途	
3.7.4 モジュールの追加	

3 コンポーネント

本書添付のサンプルアプリケーションでは、アプリケーションを機能的に分割してコンポーネント化しています。 本章では各コンポーネントについて簡単に説明します。

3.1 コンポーネントの構成

サンプルアプリケーションは次のようなコンポーネントに分割されています。

コンポーネント	プロジェクト名	目的
共通コンポーネント	ED_COMMON	どのアプリケーションでも非常によく使われる便利なユーザイベント、モ デル、ユーザ定義関数などを提供する。
データコンポーネント	ED_DATA	アプリケーションに固有なデータ定義(モデル、データソース)、およびそれに関連するプログラム(マスタメンテナンス、選択プログラム)を提供する。
受注コンポーネント	ED_JUCHU	受注業務に関連するプログラムを提供する。
ホストコンポーネント	ED_HOST	コンポーネント全体を統括し、ユーザメニューを提供する。

各コンポーネントの依存関係は下図の通りです。



即ち、ホストコンポーネントは受注コンポーネントに依存し、受注コンポーネントはデータコンポーネントに依存しています。また、すべてのコンポーネントは共通コンポーネントに依存しています。

共通コンポーネントは、アプリケーションでよく使われるオブジェクトのうち、特定のアプリケーションに依存しないものを集めたものです。このコンポーネントはどのアプリケーションにも応用できることになり、開発者にとっての財産となります。

データコンポーネントは、データベースに依存したデータ定義を行うものです。このコンポーネントは、受注コン ポーネントのほかに、仕入れ、在庫、その他具体的な業務を行うためのコンポーネント間で共有できるようにな ります。

ホストコンポーネントと受注コンポーネントとをあえて分けている理由は、拡張性を考慮してのことです。例えば、 このアプリケーションに在庫および仕入れ業務のためのコンポーネントを追加するとなると、次の図のような構 造とすることができます。図中、緑色のコンポーネントが追加となったコンポーネントです。



在庫および仕入れコンポーネントの追加方法 (共通コンポーネントの依存関係は省略してあります)

このように、受注コンポーネントの上にひとつホストコンポーネントをかぶせた形にすると、拡張性が高くなり、 パッケージでオプションコンポーネントとすることも容易になります。

3.2.1 ディレクトリ構造

サンプルアプリケーションである ZIP ファイルを解凍すると、下図のようなディレクトリ構造が現れます。この中に必要なファイルはすべて格納されています。

😂 C:¥Magic¥v10_coding_sar	nple2	app		
ファイル(E) 編集(E) 表示(V)	お気(こ入り(A) ツール(T)	ヘルプ(円)	
アドレス(D) 🛅 C:¥Magic¥v10_codir	ng_sar	nple2_app		🔺 🄁 移動
フォルダ	×	名前 🔺	種類	更新日時
🖃 🚞 v10_coding_sample2_app	~	COMPONENTS	ファイル フォルダ	2007/08/24 9:42
COMPONENTS	_	ED_COMMON	ファイル フォルダ	2007/08/24 9:43
🗉 🛅 ED_COMMON		🛅 ED_DATA	ファイル フォルダ	2007/08/24 9:49
🗉 🛅 ED_DATA		🛅 ED_HOST	ファイル フォルダ	2007/08/24 12:57
표 🛅 ED_HOST	=	🛅 ED_JUCHU	ファイル フォルダ	2007/08/24 10:11
🗉 🛅 ED_JUCHU	_	🛅 ENV	ファイル フォルダ	2007/08/22 10:49
🛅 ENV		🛅 Picture	ファイル フォルダ	2007/08/22 10:49
🛅 Picture		🛅 Text	ファイル フォルダ	2007/08/22 10:49
🛅 Text		(<u>`</u> ml	ファイル フォルダ	2007/08/22 10:49
🛅 xml	~			
<	>			

3.2.2 プロジェクトソースファイル

各コンポーネントのプロジェクトは、それぞれのプロジェクト名と同じ名前のサブディレクトリに格納されています。 例えば、共通コンポーネント ED_COMMON のプロジェクトソースファイルは、サブディレクトリ ED_COMMON の 下にあります。

😂 C:¥Magic¥v10_coding_sa	mple2	app¥ED_COMMON	
ファイル(E) 編集(E) 表示(⊻)	お気()	:入り(<u>A) ツール(T) ヘルプ(H)</u>	A
アドレス(D) 🛅 C:¥Magic¥v10_codi	ing_san	nple2_app¥ED_COMMON	🔺 🄁 移動
7 ช มชี	×	名前 🔺	サイズ 種類
🖃 🚞 v10_coding_sample2_app	~	🚞 Exports	ファイル
COMPONENTS		C Source	ファイル
🖃 🗁 ED_COMMON		📷 CIB_metadata	70 KB ファイル
🛅 Exports		ED_COMMON.edp	1 KB eDevel
🚞 Source		🖬 ED_COMMON.opt	1 KB OPT 7:
🗉 🚞 ED_DATA	_		
🗉 🚞 ED_HOST			
🗉 🛅 ED_JUCHU			
🛅 ENV			
🛅 Picture	~		
<	>	<	>

3.2.3 コンポーネントキャビネットファイル

各コンポーネントをキャビネットファイル化した ECF ファイルは、COMPONENTS ディレクトリにまとめて格納されています。

😂 C:¥Magic¥v10_coding_sa	mple2	_app¥COMPONEN	TS		×
ファイル(E) 編集(E) 表示(⊻)	お気(2	:入り(<u>A</u>) ツール(①	ヘルプ(田)		
アドレス(D) 🛅 C:¥Magic¥v10_codi	ing_san	ple2_app¥COMPONEN	ITS	💙 🄁 相	鋤
フォルダ	×	名前 🔺	サイズ	種類	更
🖃 🚞 v10_coding_sample2_app	~	D_COMMON.ecf	20 KB	eDeveloper Cabinet File	200
COMPONENTS		🗟 ED_COMMON.eci	5 KB	ECI ファイル	200
🗉 🛅 ED_COMMON		💓 ED_DATA.ecf	105 KB	eDeveloper Cabinet File	200
🗉 🚞 ED_DATA		🗟 ED_DATA.eci	5 KB	ECI ファイル	200
표 🚞 ED_HOST		💓 ED_HOST.ecf	56 KB	eDeveloper Cabinet File	200
🗉 🛅 ED_JUCHU	_	💓 ED_JUCHU.ecf	166 KB	eDeveloper Cabinet File	200
🛅 ENV		🗟 ED_JUCHU.eci	1 KB	ECI ファイル	200
🛅 Picture					
🛅 Text					
🛅 xml	~				
<	>	<			>

キャビネットファイルをプロジェクトファイルとは異なるディレクトリに格納してあるのは、単に管理上の容易さの ためです。共通コンポーネントなど、独立に別のアプリケーションでも使われる可能性があるコンポーネントの 場合に、ソースの位置とキャビネットファイルの位置とを分けたり、あるいは場合によっては、自社開発でないコ ンポーネントなどプロジェクトソースファイル自体持っていない場合もあることも考慮すると、キャビネットファイ ルおよびコンポーネントインターフェース定義ファイル (ECI ファイル)を決まった共通のディレクトリに入れておく と便利です。

3.2.4 コンポーネントファイルの指定

各プロジェクトが依存するコンポーネントはコンポーネントリポジトリに登録しますが、ここでコンポーネント特性 として、プロジェクトファイル名とキャビネットファイル名が設定されます。実行環境でアプリケーションをインストー ルする位置に自由度を与えるためには、ここには論理名を使って定義するのがベストです。

例えば、次の図は、ED_JUCHU 中のコンポーネントリポジトリでの ED_DATA コンポーネントのコンポーネント特性です。ここでは ED_ROOT という論理名を使ってプロジェクトファイル名とキャビネットファイル名を定義しています。

このようにしておけば、論理名 ED_ROOT (アプリケーションのルートディレクトリ名)が正しく設定されていれば、 コンポーネントのキャビネットファイルもプロジェクトファイルも正しく読み込まれるようになります。

ここではプロジェクトファイル名とキャビネットファイル名だけを使っていますが、コンポーネントのへ ルプファイルを使っている場合には、ヘルプファイル 欄も論理名 ED_ROOT を使って定義しておくべ きでしょう。

コンポーネ	ント特性		X
汎用特 ふ	性 読み込まれるコンポーネン 実行環境に依存します.	トの特性を定義します。これらの特性は開発環境、および	
	レビ^ッジ*ョン:	1.2	
	プロジェクトファ仙名:	XED ROOTXED DATA¥ED DATA.edp]
(キャビネットファ仙名:	XED ROOT%COMPONENTS¥ED DATA.ecf]
	☑ 即時有効		
~設定			_
<u>ਤ</u>	ここでは、読み込むコンポーネ	♡Nに対応するヘルプファイルを定義することができます。	
	^₩フ°ファイ₩ :		
	∿⊮フ°キー:		
		OK ++>tell	

ところで、コンポーネント特性のプロジェクト特性のこれらのファイル名は、Magic コンポーネントインターフェー スビルダを使ってコンポーネントインターフェイス定義ファイル(ECIファイル)を作成するときに、**コンポーネントと プロジェクトの設定** の画面で指定します (下図)。

Magic コンポーネントインターフェースビルダを開くに は、プロジェクト ED_DATA を開いている状態で、オ プション(O) → インターフェースビルダ(B) → Magic (D) を 選びます。



このウイザートは作成を支援します an eDeveloper Component Interface

<u>キャンセル</u> 〈 戻る(B) / 次へ(N) > (終了(E)

「ようこそ」画面が出るので、「次へ(N)」を押します。

「有効なコンポーネント」画面で、「新規(E)」 ボタンを 押します。 すでにコンポーネント定義を行っている場合には一 覧に ED_DATA があるはずですので、それを選んで 「次へ(N)」を押します。	Maeicコンボーネントインタフェースビルダ 有効なコンボーネント ここでは、既存のコンボーネントの修正や削球、または新規作成を行うことができ ます。
「コンポーネントとプロジェクトの設定」 画面で、コン ポーネント名を ED_DATA (任意の名前で構いません が、管理上の便利さのため、プロジェクト名と同じに しておきます) とし、プロジェクトファイル名、キャビネッ トファイル名を論理名 ED_ROOT を使って指定します。	7泊ゲセ州名: Data 新規(E) 評目除(D) キャンセル(G) く戻る(B) 次へ(M)> 終7(F) Magicコンホーネントとブロジェクトの設定 ここでコンボーネントとブロジェクトを設定することができます. ●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●

ここには、デフォルトではフルパス名でプロジェクトファイル名などが設定されていますが、論理名を使った名前に書き直してください。

ここで設定しておけば、その内容が ECI ファイルに反映され、それを読み込めばコンポーネント特性にも反映されるようになります。

3.3 共通コンポーネントの内容

共通コンポーネントは、アプリケーションでよく使われるオブジェクトのうち、特定のアプリケーションに依存しないものを集めたものです。これには次にようなものがあります。

- ユーザ定義イベント
- 共通モデル
- 関連プログラム
- ユーザ定義関数

3.3.1 ユーザ定義イベント

共通コンポーネントでは、どのアプリケーションでもよく使われるような、ユーザ定義イベントが定義されています。

メインプログラムを開いて、ユーザイベントテーブルを開いてください。下図のようなユーザ定義イベントが定義 されています。

#	名前	トリガ・タイフ。	11-12 11	い うメータ	強制終了	公開名	公開
1	ெ.ズーム	I=内部	λ°−₽(Σ)	0	E=編集	GU_ズーム	~
2	GU_照会	N≕なし		0	E=編集	GU_照会	~
3	GU_修正	N≕なし		0	E=編集	GU_修正	~
4	GU_登録	N≕なし		0	E=編集	GU_登録	~
5	GU_削除	N≕なし		0	E=編集	GU_削除	~
6	GU_出力	N≕なし		0	E=編集	GU_出力	~
7	GU_印刷	N≕なし		0	E=編集	GU_ED刷	~
8	GU_取消	N=なし		0	E=編集	GU_取消	~
9	GU_終了	N=なし		0	E=編集	GU_終了	~
10	GU_選択	N≕なし		0	E=編集	GU_選択	~
11	GU_検索	N=なし		0	E=編集	GU_検索	~
12	GU_実行N	N≕なし		0	N≕なし	GU_実行N	~
13	GU_実行E	N=なし		0	E=編集	GU_実行E	~
14	GU_実行C	N=なし		0	C=⊐ՆՒロ-∥	GU_実行C	~
15	GU_実行R	N≕なし		0	R=レコード更新前	GU_実行R	~

これらのユーザ定義イベントの利用目的は、だいたい次の表のような利用法を想定しています。

種類	イベント	目的
ズーム対応	GU_ズーム	ズームに対応したイベントハンドラを作成する時に、入力途中のデータを項目 に反映させるため、強制終了が「E=編集」である必要があるので、定義してい る。
タスクモード 変更ボタン用	GU_照会、 GU_修正、 GU_登録	「照会」「修正」「登録」ボタンを押した場合に発生させるイベント。
ー般機能 ボタン用	GU_削除、 GU_出力、 GUI_印刷 等/	「削除」「出力」「印刷」等々、よく使うボタンを押した場合に発行させるイベント。 マ

実行ボタン用	GU_実行 N、 GU_実行 E、 GU_実行 C、	汎用の「実行」ボタンを押した場合に発行させるイベント。4種類定義されてい るのは、「強制終了」の設定が異なるものを定義しているため。
	GU_美行 C、 GU_実行 R	

コンポーネントとして外部から利用してもらうために、それぞれのイベントには公開名が定義されており、公開 フラグをオンにしてあります。



- これらのユーザ定義イベントは、それ自体で上記のような機能を持っているものではありません。必ず、 これらのイベントに対応したイベントハンドラをプログラムで作成してやる必要があります。
- 強制終了 パラメータについては、ここでは説明を省略します。

3.3.2 モデルリポジトリ

共通コンポーネントのモデルリポジトリには、Magic のアプリケーションならばよく使うような、汎用のモデルが定義されています。(下図)

	ŧ	テルリポジトリ				$\mathbf{\times}$
Ц;		名前	752	型	フォルジ 公開名	^
	1	● 内部イベント ボタン	F=項目	A=文字		
	2	PB_取消	D=GUI表示形式	P=プッシュボタン	PB_取消	
	3	PB_終了	D=GUI表示形式	P=プッシュボタン	PB_終了	
	4	PB_取消(加-ズイベント)	D=GUI表示形式	P=プッシュボタン	PB_取消(ウロース	
	5	PB_取消終了	D=GUI表示形式	P=プッシュボタン	PB_取消終了	
	6	PB_OK	D=GUI表示形式	P=プッシュボタン	PB_OK	
	- 7	PB_選択	D=GUI表示形式	P=プッシュボタン	PB_選択	
	8	PB_ズーム	D=GUI表示形式	P=プッシュボタン	PB_ス*ーム	
	9	TB_取消	F=項目	A=文字	TB_取消	
	10	TB_終了	F=項目	A=文字	TB_終了	
	11	TB_取消(クローズイベント)	F=項目	A=文字	TB_取消(ウロース	
	12	TB_取消終了	F=項目	A=文字	TB_取消終了	
	13	TB_OK	F=項目	A=文字	ТВ_ОК	
	14	TB_選択	F=項目	A=文字	TB_選択	
	15	TB_ス*ーム	F=項目	A=文字	TB_ス°ーム	
	16	● ユーザイベントボタン	F=項目	A=文字		
	17	PB_U実行E	D=GUI表示形式	P=プッシュボタン	PB_U実行E	
	18	PB_U実行C	D=GUI表示形式	P=プッシュボタン	PB_U実行C	
	19	PB_U実行R	D=GUI表示形式	P=プッシュボタン	PB_U実行R	
	20	PB_U照会	D=GUI表示形式	P=プッシュボタン	PB_U照会	
	21	PB_U修正	D=GUI表示形式	P=プッシュボタン	PB_U修正	
	22	PB_U登録	D=GUI表示形式	P=プッシュボタン	PB_U登録	
	23	PB_U削除	D=GUI表示形式	P=プッシュボタン	PB_U肖·J除	
	24	PB_U出力	D=GUI表示形式	P=プッシュボタン	PB_U出力	
	25	PB_UED刷	D=GUI表示形式	P=プッシュボタン	PB_UED刷	
	26	PB_U取消	D=GUI表示形式	P=プッシュボタン	PB_U取消	
	27	PB_U終了	D=GUI表示形式	P=プッシュボタン	PB_U終了	
	28	PB_U選択	D=GUI表示形式	P=プッシュボタン	PB_U選択	
	29	PB_U検索	D=GUI表示形式	P=プッシュボタン	PB_U検索	~

これには次のようなものがあります。

内部イベント用 ボタン PB_終了、 PB_選択等 PB_選択等 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_選択、該 PB_認力 PB_ PB_ PB_ PB_ PB_ PB_ PB_ PB_ PB_ PB_	種類	モデル例	目的
	内部イベント用 ボタン	PB_取消、 PB_終了、 PB_選択 等	プッシュボタンのモデルであり、内部イベント(「キャンセル(C)」、「クローズ(C)」などが 実行イベント として設定されています。このモデルを利用すれば、プログラムロジックを書かずとも、フォーム上に貼り付けるだけで、取消、終了などの機能を持ったボタンを実現できます。

例: PB_終了 ボタンモデル

⊐)hr-W特性: 7°。	必ュボタン - PB_終★ 	🖾 ғ	デルリポジトリ		
区分(<u>C</u>) 全体(<u>A</u>)	#	名前	752	型
行う	L7°7 🔥	1	● 内部イベント ボタン	F=項目	A=文字
	PR 终了	2	PB_取消	D=GUI表示形式	P=プッシュボタン
古りていた。	\$\$7(&X)	3	PB_終了	D=GUI表示形式	🔹 P=プッシュボタン 🔵
型	[データ]	4	- PD_取消(加 ズイベント)	D=GUI表示形式	₽=プッシュボタン
す゛タンスタイル	P=プッシュボタン	5	PB_取消終了	D=GUI表示形式	P=プッシュボタン
デフォルトイメージ・ファ		6	PB_OK	D=GUI表示形式	P=プッシュボタン
(実行ハ*)	ንበ~ ኢ *(C)	7	PB_選択	D=GUI表示形式	P=プッシュボタン
美行工	C=1777 929	8	PB ズーム	D=GUI表示形式	P=プッシュボタン

種類	モデル例	目的
ユーザイベント用 ボタン	PB_U 照会、 PB_U 印刷、 PB_U 実行 E 等	プッシュボタンのモデルであり、共通コンポーネントで定義されたユー ザ定義イベントが 実行イベント として設定されています。利用する場 合には、フォーム上に貼り付けて、対応するイベントハンドラを作成す ることになります。

例: PB_U 実行 E ボタンモデル

コンhu-4時性: プッシュボタン - PB_Uミ×		<u>1</u> : × [🗴 🗧	デルリポジトリ		
区分© ±1↑	(±)		#	名前	752	型
ロモデル		^	16	ユーザイベントボタン	F=項目	A=文字
行り	1717		(17	PB_U実行E	D=GUI表示形式	P=7°-99±#°\$2
1)kn-1/2	PR II主行		18-	PD_U実行C	D=GUI表示形式	₽=プۥッシュボタン
書式	実行(&E)		19	PB_U実行R	D=GUI表示形式	P=プッシュボタン
型	[データ]		20	PB_U照会	D=GUI表示形式	P=プッシュボタン
ボタンスタイル	P=プッシュボタン		21	PB_U修正	D=GUI表示形式	P=プッシュボタン
73111-17-2577	11		22	PB_U登録	D=GUI表示形式	P=プッシュボタン
(実行ハント	GU_実行E		23	PB_U削除	D=GUI表示形式	P=プッシュボタン
表行力に	U=1777 919		24	PB_U出力	D=GUI表示形式	P=プッシュボタン

種類	モデル例	目的
ボタン項目	TB_終了、 TB_U 照会 等	上の内部イベント用ボタンあるいはユーザイベント用ボタンを GUI 表 示形式のスタイルとして参照している、文字型の項目モデルです。 プログラムでは、データビューでこのモデルを参照して変数を定義し、 その項目をフォームエディッタ上に貼り付けることにより、各種の機能 を持ったボタンを配置することができます。変数項目として定義してい るので、カーソルがパークします。

例: TB_終了 ボタン項目モデル

項目特性 A=文字		×	58 F	デルリポジトリ		
区分(<u>C</u>) 全体(<u>A</u>)			#	名前	572	型
ブラウザ形式 ゴラウザ形式モーゴル	E=Iディット E=Tディット	•	* 8	PB_X*-&	D=GUI表示形式	P=プッシュボタン
日田表示形式	P=7*a91**41	\sim	9	TB_取省	F=項目	A=文字
GUI表示形式テープト	P=7*>>1=*9)	21	(10	TB_終了	F=項目	A=文字
GUI出力形式	EFIT Yoh	1	7+-	TD_取消(カース・ヘー・ソー)		計文字
GUI出力形式テーブル	EFIFYYyh		12	TB_取消終了	F=項目	A=文字
テキスト形式	E=Iディット		13	TB_OK	F=項目	A=文字
⊡デフォルト/NULL			14	TB_選択	F=項目	A=文字
NULL值可	No		15	TB_ス°~ሬ	F=項目	A=文字
NULL計算他		= =	16	・ユーザイベントボタ	F=項目	A=文字
NULLAGICATION	No	-	17	PB_U実行E	D=GUI表示形式	P=プッシュボタン
デフォト値	終了(&X)		18	PB_U実行C	D=GUI表示形式	P=プッシュボタン
データベースデフォルト値			19	PB_U実行R	D=GUI表示形式	P=プッシュボタン
□格納			20	PB_U照会	D=GUI表示形式	P=プッシュボタン

種類	モデル例	目的
共通データ項目	ファイル名、 プログラム番号	どんなアプリケーションにでもよく使われるようなデータ項目を定義し ます。

例: ファイル名 項目モデル。ファイル名のモデルでは、ズームしたらファイルダイアログが表示されるよう、選 択プログラムが設定してあります。また、ドラッグアンドドロップも有効にしてあり、エクスプローラなどからドラッ グアンドドロップでファイル名を設定できるようにもなっています。

⊟tフ [*] ש ידייש	[デフォレト]	<u>^</u>	58 59	EDT_中サイズ(12pt) ERM CUIED副基本	0=GUI出力形式 0=GUI出力形式	E=Iディット E=コ+ーレ
□詳細			00 00	● 子の他	F=項日	- 74 G A=女字
者 式 型	255 A=文字		61	ファイル名	F=項目	A=文字
範囲		_	62	🗢 内部制御用 单	F=項目	A=文字
2入力		=	63	タスクモード	F=項目	A≕文字
(選択プログラム	3)	SEL_	64	インデックス番号	F=項目	N=数値
起動刊	B=前置		65	プログラム番号	F=項目	N=裝y(值
□表示			66	プログラム名	F=項目	∆=文字
11.7°画面	0		67	プログラムロ	F= 1百日	N=类灯(直
ツールチッファ	0		60		- 項日	N=选作(店
目動ヘルブ	U		00		[項目 [項目	いて安然に置
□ 791 ₽			63	×_1-2	F-4月日	A-X-F
フィラウザ、形式	EFIFYANN					
フッラウサッモジェイテーフッル	F=Tディϣト					
GUI表示形式	E=17°491					
GUI表示形式テーブ	E=Iディゥト)				
GUINET	E=T7 Amb					

種類	モデル例	目的							
その他の GUI 部品	TBL_交互色、 FRM_基本、 LBL_ラベル、 EDT_表示専用	ユーザインターフェースに一貫性を持たせるために、テーブル、フォー ム、ラベル(テキストコントロール)、エディットコントロールなど多用され るコントロールについて、いくつかのパターンをモデルとして用意しまし た。開発時にはモデルを選択するだけでよいので設定が容易になり 設定忘れなどがなくなるし、保守時にはモデルの変更だけでアプリケー ション全体のユーザインターフェースが一貫して変更されるようになり ます。							
例:TB_交互色:交	例: TB_交互色: 交互色背景のテーブルコントロールを定義します。								
コントロール特性: テーフドル	- TBL	×	🕄 E:	デルリポジトリ					
区分(C) 全体(A)			#	名前	<u> </u>	型			
□表示		~	42	TB_U検索	F=項目				
巴志利鱼の指定	」	- 11	43	● GUI⊐ントロール类頁	F=項目	A=文字			
交互表示色	101	- 11	44	TBL_交互色	D=GUI表示形式	T=テープル			
ツールチッフ。	0		45	TBL_交互色(位置100)	D=GUI表示形式	T=テーブル			
አንብዞ	W=Windows		46	FRM_基本	D=GUI表示形式	F=7#~&			
境界のスタイル	HE太線		47	FRM_選択画面フォーム	D=GUI表示形式	F=フォーム			
スクロールハット	Yes		48	FRM_スクリーン入力フォーム	D=GUI表示形式	F=フォーム			
区 切りり 11/5/1/2号/11/2万/2万/11	NO C-おり見 とついれついし	_	49	FRM_MDI7W	D=GUI表示形式	F=フォーム			
7171110782110 ハイライト行の色	い 日安Cコ/TUTW 3	_	50	LBL_5∿°⊯	D=GUI表示形式	S=Z\$Frigh			
下辺の間隔	▼ N=なし		51	EDT_表示専用 	D=GUI表示形式	E=Iディット			

3.3.3 関連プログラム

サンプルアプリケーションの共通コンポーネントでは、関連するプログラムとして、ファイル名モデルをズームした場合に、ファイルダイアログが表示されるよう、選択プログラムを用意しました。

その他のプログラムはテスト用であり、コンポーネントの外には公開されていません。

🖾 70	グラムリポジトリ			
ЦЦ.	名前	 公開名	外部	最終更新日
1 2	メインプログラム			2007/08/08
	SEL_ファイル			2007/08/07
5	TEST			
6	T/MsgBox			2007/07/31
7	T_SEL_ファイル			2007/08/07

3.3.4 ユーザ定義関数

共通コンポーネントには次のユーザ定義関数が定義されています。

3 4	<u> </u>	- メインプロ	グラム							×
デー	タビ=	⊾ー ロジック	フォーム							
	1 [∃ F=関数	u∎sgBoxFu	11		スコーフ*:	G=2*0-	バ戻り値: 2	CallDLL('user32.Mes	~
	2	項目	P=//°5%-ጵ	1	PI_Xot-91		A=文字	200		
	3	項目	P=//°5%-ጶ	2	PI_Xot=ジタイトル		A=文字	50		
	4	項目	P=//°5%-ጶ	3	PI_ポタンオプション		N=数值	2		
	5	項目	P=//°5%-ጶ	4	PI_アイコンオフ°ション		N=数値	2		
	6	項目	P=//°5%-ጶ	5	PI_デフォルトボタン		N=数值	1		
	7	項目	P=//°5%-ጶ	6	PI_その他		N=数值	8		
	8									
	9	項目	¥=変数	1	VS_X-JE-917551		N=数値	8		
	10									
	11	項目更新	∀=項目	G	VS_X-JE-9*759*	値:	1 IF	(PI_#*\$)#7*9a) (·	
	12 (∃ F=関数	uYesNoDlg			スコーフ*:	G=ク゛ロー	バ戻り値: 3	u∭sgBoxFull (PI_⊁∍ŧ	
	13	項目	P=//°5%-ጵ	7	PI_Xot-91		A=文字	200		
	14	項目	P=//°5%-ጶ	8	PI_\$/N		A=文字	50		
	15 (∃ F=関数	uTaskilode	Strin	ng	スコーフ*:	G=ኃ゙ロ-	バ戻り値: 4	IF (Stat(PI_タスクルベル	
	16	項目	P=/\°5%-%	9	₽I_\$ス\$レヘ*⊮		N=数值	N5		

uMsgBoxFull

名前	uMsgBoxFull					
目的	メッセージダイアログボック	スを表示します。				
パラメータ	PI_メッセージ	(文字型)メッセージ本文				
	PI_メッセージタイトル	(文字型)タイトルバーに表示する文字列				
	PI_ボタンオプション	(数値)表示するボタンの下図やラベルを指示するフラグ				
	PI_アイコンオプション	(数値)表示するアイコンの種類を指示するフラグ				
	PI_デフォルトボタン	(数値) デフォルトボタンを指示するフラグ				
	PI_その他	(数値)その他のオプションを指示するフラグ				
戻り値	(数値)押されたボタンに対応する番号					
補足	Windows の MessageBox APIを利用しています。オプションフラグは Windows API の値に 準拠して定義されています。					

ここで、各パラメータの値の意味は、以下の通りです。

パラメータ	値	意味
PI_ボタンオプション	0	ОК
	1	OK/キャンセル
	2	中止/再試行/無視
	3	はい/いいえ/キャンセル
	4	はい/いいえ
	5	再試行/キャンセル
	その他	(0と同じ)
PI_アイコンオプション	0	アイコンなし

	1	中止 (× マーク)
	2	疑問 (? マーク)
	3	警告(!マーク)
	4	情報 (iマーク)
	その他	0 と同じ
PI_デフォルトボタン	0	左から1番目のボタン
	1	左から2番目のボタン
	2	左から3番目のボタン
戻り値	1	OK ボタンが押された
	2	キャンセルボタンが押された。あるいは、ウィンドウの[X] ボタンで閉じられた。
	3	中止 ボタンが押された。
	4	再試行 ボタンが押された
	5	無視 ボタンが押された
	6	はい ボタンが押された
	7	いいえ ボタンが押された

uYesNoDlg

名前	uYesNoDlg		
目的	はい/いいえのメッセージダイアログボックスを表示します。		
パラメータ	PI_メッセージ	(文字型)メッセージ本文	
	PI_メッセージタイトル	(文字型)タイトルバーに表示する文字列	
戻り値	(論理型)はい が押された場合には True、その他の場合には False が返ります。		
補足	uMsgBoxFull の簡略版です。		

uTaskModeString

名前	uTaskModeString
目的	現在のタスクのタスクモードを文字列で返します。
パラメータ	(なし)
戻り値	(文字型)タスクのモードに応じ、「登録」「照会」「修正」のいずれかが返ります。

3.4 データコンポーネントの内容

データコンポーネント ED_DATA には、サンプルアプリケーションに固有なデータの定義と、それに関連するプログラムが定義されています。

3.4.1 モデルリポジトリ

モデルリポジトリには、アプリケーションに固有なデータ項目がモデルとして定義されています。これには、顧客コード、顧客名、商品コード等があります。

また、顧客コードや商品コードなどには、選択プログラムが設定されています。

さらに、選択肢が比較的少ないデータ項目(担当者コード、単位名など)には、データコントロールとしてコンボボックスモデルが定義されています。

😹 ED_DATA – Magic	影ED_DATA - Magic - 体験版							
ファイル(E) 編集(E) 表	示① プロジェクト	(<u>P</u>) オフジ	a)(<u>0</u>) 7	゙ンベッグ(<u>D</u>) ヘルフ [®] (<u>H</u>)				
🏝 🚖 🚖 🕨 📰 🗳	P 🔲 🔯 💷 🕸			8 3 8 8 1	l 😭 🖳 🔉 🛛) 🖪 🗊 🎰 🛛		9 B 6 / ?
項目特性 A=文字		×	68 H-	白山田豊富大田				
区分(C) 全体(A)			(CA) 12.	1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2				
	Testal k1		#	名前	クラス	型	「Jaly」公開名	<u>^</u>
	17 740411		1	 アブリケーショ: 	ン F=項目	A=文字		
方 集	12K0		2	顧客CD	F=項目	A=文字	顧客CD	
型	A=文字		3	顧客名	F=項目	A=文字	顧客名	
範囲		_	4	顧客名毛	F=項目	A=文字	顧客名	=
□入力		=	5	郵便番号	F=項目	A=文字	郵便番	
選択プログラム	4	SEL_	6	都道府県	F=項目	A=文字	都道府	
起動セード	BF前置		7	住所	F=項目	A=文字	住所	
日表示	0		8	電話番号	F=項目	A=文字	電話番	
^₩/ 回回 ツールチュコ ⁰	0		9	決算条件CD	F=項目	N=数值	決算条	
2 10/92 自動へいつ?	0		10	決算条件名	F=項目	A=文字	決算条	
□ スタイル	·		11	CMB_決済条件	D=GUI表示形式	C=コンボボックス	CMB_決	
フ*ラウザ*形式	Ε=Ιディット		12	決済条件CD(CMB)	F=項目	N=数值	決済条	
ブラウザ形式テーブル	EFIディット		13	担当者CD	F=項目	N=数值	担当者	
GUI表示形式	ΕΞΙΞΎγοΝ		14	CMB_担当者CD	D=GUI表示形式	C=コンホドホドックス	CMB_担	
GUI表示形式テーブル	EFIFYYON	~	15	担当者CD(CMB)	F=項目	N=数値	担当者	
	F-728. L		16	担当者名	F=項目	A=文字	担当者	
七7 あ フォーム / アントロール / 項目 / の名	ほうれた対するすべ	7/0/特性	17	出荷先CD	F=項目	N=数値	出荷先	
を再継承したり継承を解除	することができます		18	連絡先名	F=項目	A≕文字	連絡先	
			19	商品区分	F=項目	A≕文字	商品区	
			20	商品CD	F=項目	A=文字	商品CD	
ナビグータ 特性			21	商品名	F=項目	A=文字	商品名	~
開発モード: ED_DATA								

3.4.2 データリポジトリ

データリポジトリには、アプリケーションに固有なテーブルの定義がされています。これには、顧客マスタ、商品 マスタ、受注ヘッダ、受注詳細ヘッダなどが含まれます。

また、一時テーブルとして、メモリデータベースに 受注ヘッダ_TMP、受注明細_TMP が定義されています。 生産性と保守性の向上のために、それぞれのテーブルのカラムは、すべてモデルを参照して定義されています。

SED_DATA - Magic - 体験版						
ファイル(E) 編集(E) 表示(V) プロジェウト(P)	オフ [®] ション(<u>O</u>) デバッケ(<u>D</u>) ヘルフ	°(<u>H</u>)				
* · · · · · · · · · · · · · · · · · · ·		91 III. 🧆 III. III. III.	🗵 🕾 🗩	/ 🛶 🖦 🖦		1 9
抗特性 A=文字: 顧客CD ×	🛒 データリポジトリ					
区分(<u>C</u>) 全体(<u>A</u>)	# 「夕哉」 「おこか」	_1.4	≑ೇಟಗಿ∧ೇಟಾ	Dell b*	八明々	
⊡स्न*∦ 🔨	# 20m 7 7/7	-∧-□ 务注承是答理	ED SAMPLE2	24/02	ム開石	
tři 顧客CD … Ø 呼吁			ED_SAMPLE2		頭客マスタ	
	3 出荷先マスタ EDM &	出荷先マスタ	ED SAMPLE2		出荷先マスタ	-
査1% 12N0 型 ム=文字 ■	4 決済条件マスタ EDM ½		ED SAMPLE2		決済条件マスタ	_
	5 担当者マスタ EDM_1	旦当者マスタ	ED_SAMPLE2		担当者マスタ	
□入力	6 商品マスタ EDM_M	品マスタ	ED_SAMPLE2		商品マスタ	
選択プログラム 4 SEL_	7 単位マスタ EDM_9	単位マスタ	ED_SAMPLE2		単位マスタ	~
起動t-h' B-前直 口事子						_
	カラム インデジックス 外部キー					
ッールチッフ [*] 0	ま 名前	∓ ∓ °ll.	刑	方 垂	<u>^</u>	
自動147°0	1 顧客CD	2 顧客CD		12K0		
	2 顧客名	3 顧客名	A≕文字	50		
ノーフソリーカシェン、 ビーエアーイット フドラウサド田を式デーフド1F=Tデドオット	3 顧客名託	4 顧客名弐	A=文字	50		
GUI表示形式 E=Iディット	4 郵便番号	5 郵便番号	A=文字	###-####	ко	
0017 ≢ 🎞 🕮 🕂 🗖 🗖 – 🖓 – 1	5 都道府県	6 都道府県	A=文字	10		
モデル コットノイン・リークロローのタエージョイ・ウォナス・オッツ	6 住所 1	7 住所	A=文字	50		
フォームノコンロールノ項目の合モナルに入りるのタイ ての特性を再維承したり継承を解除することが	7 住所2	7 住所	A=文字	50		
できます.	8 電話番号	8 電話番号	A=文字	20K0		
		8 電話番号	A=文子	20K0	*	
			m_ 21111A	*	>	
 開発モード: ED_DATA					広境 挿2	λ

データリポジトリで定義されているテーブルの ER 図は次のページの図の通りです。



3.4.3 プログラムリポジトリ

下図は、データコンポーネントのプログラムリポジトリです。

3	プロ	グラムリボジトリ					
#		名前	Dallo [*]	公開名	外部	最終更新日	
	1	メインプログラム				2007/08/08	
	2	BT_サンプルデータ入力		BT_サンブルデータ入力	~	2007/08/10	
	3	OT_選択プログラム		OT_選択プログラム	~	2007/08/08	
	4	SEL_顧客選択		SEL_顧客選択	~	2007/08/08	
	5	SEL_出荷先選択		SEL_出荷先選択	~	2007/08/03	
	6	SEL_決算条件選択		SEL_決算条件選択	~	2007/08/03	
	- 7	SEL_担当者選択		SEL_担当者選択	~	2007/08/03	
	8	SEL_商品選択		SEL_商品選択	~	2007/08/03	
	9	SEL_課税区分選択		SEL_課税区分選択	~	2007/08/03	
	10	SEL_消費税選択		SEL_消費税選択	~	2007/08/03	
	11	SEL_受注選択		SEL_受注選択	~	2007/08/10	
	12						
	13	OM_顧客マスタ照会		OM_顧客マスタ照会	~	2007/08/08	
	14	0C_顧客マスタ入力		0C_顧客マスタ入力	~	2007/08/08	
	15	BD_顧客データ削除		BD_顧客データ削除	~	2007/08/10	
	16						
	17	OM_商品マスタ照会		OM_商品マスタ照会	~	2007/08/08	
	18	00_商品マスタ入力		0C_商品マスタ入力	~	2007/08/08	
	19	BD_商品データ削除		BD_商品データ削除	~	2007/08/08	
	20						
	21	OM_決済条件マスタ照会		OM_決済条件マスタ照会	~	2007/08/10	
	22	OM 担当者マスタ照会		OM 担当者マスタ照会	\checkmark	2007/08/08	
<							>

データコンポーネントのプログラムリポジトリには、次のような種類のプログラムが定義されています。

- 選択プログラム
- マスタ照会・修正プログラム
- サンプルデータ入力

それぞれについて、次ページ以下に簡単に説明します。

種類	プ	゜ログラム例	目的			
選択プログラム		EL_顧客選択	コードデータ(顧客コード、商品コードなど)でズームされたときに表示される、一覧選択プログラム。			
例: 顧客選択プロ 行うことができる。	コグラム ようにな	ム。この選択プログラムて なっています。	『は、顧客⊐ード、顧客名、顧客名読みを入力し	、てキー検索を		
	顧客選択					
	キー検索					
	顧客CD					
	顧客名					
	顧客名					
	顧客CD	顧客名	顧客名彩			
	0110	エムエスジャパンサービス	エムエスシ゛ゃハ゜ンサーヒ゛ス			
	0120	川田設備	カワタ*セツビ*			
	0130	アールエス産業	アールエスサンキ~ョウ エー・トスキロトンズンマン			
	0140	エニックス 川口文店 東三販売館 浦和古庄	1_978/777777			
	0160	東 かんしゅう 加速 10 文名 カナモト 戸田	17771011 1777777 17771011 1777777			
	0170	東リース 浦和支店	ビカ19リースウラワシテン			
	0180	技建	キ* か)			
	0190	アベル 桑名工場	アヘドルクワナコウシドョウ			
	0200	アベル 桑名支店	アヘドルクワナシテン			
	0210	前田製作所 コベツ名古屋 津支店	マエダ・セイザウジェッコへ、ウナコ、ヤウジテン			
	0220	コベツ三重	IV. YEI			
		<u> </u>				

種類	プログラム例	目的	
マスタ照会・修正 プログラム	OM_顧客マスタ照会	マスタテーブル(顧客マスタ、商品マスタな 登録などするためのプログラム。	よど)を照会、修正、
例: 顧客マスタ。下国	図では、顧客 0110 の顧客に	ついてデータを修正する修正画面も表示で	されています。
77(14年)編集(₽ オラ%ン@ ウィルウ@] 罰 夕 J J 秒 種 蛇 ■ ☎ ■ ☷ 閏 亜 ெ		
+─検索 顧客00	離客名		
顧客 0110 エムエス・ジャル エムエス・ジャル	住所 448-3003 愛知県 パンサービス 刈谷市ーツ木町竹下X83X *2	電話番号 決算条件/担当者 X48-284-XXXX 30 随時 X566-28-XX4X 8 長利 雄介	
1120 川田島価 カウドセンド 1130 アールエス度シ アールエス度シ マールエス マールエス マールエス マールエス マールエス マールエス マールエス マールエス マール マールエス マール マールエス マーシーム ローロ マール マールエス マール コンマート マーロ マール マールエス マーレ マール コンマーク フーレ マーシ マーロ マート マーロ マート マーム ロンマーク ス リ ロ マーレ マーク ス リ マーレ マーン フー ロ マーク ス リ マー マース コン マー マーン マー ロ マー マー マー マー マー マー マー マー マー マー マー マー マー	454-1002 愛知県 名古屋市中川区野田2X433 第名マスタ入力 第名マスタ入力 第名でスタ入力 第名名記: 江ムエスジレパンサービス 第名名記: 江ムエスジレパンサービス 第名名記: 江ムブン「パンサービス 第四音名記: 江ムブン「パンサービス 第四音名記: 江ムブン「パンサービス 第四音名記: 江ムブン「パンサービス 第四音名記: 江ムブン「パンサービス 第四音名記: 江ムブン「パンサービス 第四音音: 第四音一: 第四音一: 第四音一: 第四音一: 第四音一: 第四音一: 第四音一: 第四音一: 第四音一: 第四音: 第四音一: 第四音: 第四音一: 第四音: 第四音: 第四音: 第四音: 第四音: 第四音: 第四音: 第四音	X40-2X7-XX73 35 随時 X52-354-5XX4 6 長潮 雄介 54 30 随時 53 30 随時 54 6 長潮 雄介 53 30 随時 XX 6 長潮 雄介 2X 30 随時 6 長潮 雄介 XX 8 長潮 雄介 XX 30 随時 XX 8 長潮 雄介 XX 30 随時 XX 30 随時 XX 8 長潮 雄介 XX 30 随時 XX 8 長潮 雄介 XX 8 長潮 雄介 XX 8 長潮 雄介	
実行: ED_DATA			×) 太矮 挿入

種類	プログラム例	目的			
サンプルデータ入力	BT_サンプルデータ入力	テキストファイルのサンプルデータを、DBMS にインポートします。			
例: BT_サンプルデータ入力。このプログラムはバッチタスクなので、画面表示はありません。					

3.5 受注コンポーネントの内容

受注コンポーネント ED_JUCHU には、受注業務に関連するプログラムが定義されています。

3.5.1 モデルリポジトリ

受注コンポーネントには、モデルが定義されていません。モデルはすべてデータコンポーネント、および共通コンポーネントのものを参照しています。

3.5.2 データリポジトリ

データリポジトリには、XML データベースとして定義されてデータソースが3つ定義されていますが、現時点の サンプルアプリケーションでは利用されていません。

3	Ŧ	ータリポジトリ			
#		名前	データソース名	ティーダイィーズ フォルタッ	公開名 🔨
	1	EDT_soheader.dataroot	order.xml	Default XML Database	
	2	EDT_soheader.EDT_soheader	order.xml	Default XML Database	
	3	EDT_soheader.EDT_sodetail	order.xml	Default XML Database	
ľ	°1~	インデックス			
		a 🛃 dataroot	— #	名前 511 型	▲ た書
		senerated		1 ノート [*] ID 0 N=数値	12
		🖨 💽 EDT_soheader		2 親ID 0 N=数值	12
		📃 受注番号		3 generated 0 A=文字	25
		ーヨ タイプ			
		- 前空ったり			
		- 日 出荷先コード			
		- 📑 顧客発注番号			
		- 😑 出荷経由			
		[~

3.5.3 プログラムリポジトリ

受注コンポーネントのプログラムリポジトリには、次のようなプログラムが定義されています。

- 受注入力
- 受注取引画面参照
- 印刷
- 関連プログラム

以下にそれぞれについて簡単に見ていきます。

種類	プログラム例	目的
受注入力	OC_受注入力	受注データの登録、修正、照会、印刷等を行います。
	1.134 A1A	
受注番号 8000000011 9/7	◎ [[:在庫] ❤	取引日 07/04/26 納入子定日 07/04/27
顧客CD 0140 工:	ニックス 川口支店	出荷先印 0 エニックス 川口支店
11-9 (主所 476-XXX2)	ሳ አክ ባታ [*]	住所 【476-XXX2】
愛知県 東海市名和町二ノ上	EX 8 XXX	愛知県 東海市名和町二ノ上X8XXX
電話番号 X48-285-XX53	ファックス X52-6X3-88XX	電話番号 X48-285-XX53 ファックス X52-6X3-88XX
顧客発注番号		
担当者CD 6 安部	孝之	決算条件CD 30 未締めの翌末払い
# 商品CD 商品: 1 0.1002 パイー	名 販売数量 : ナッブルの缶詰 2 1	受注単位 課税区分CD 税処理CD単価 消費税金額 金額 納期日 ④ 1 1915志上分 1 149 15 313 07/04/27
2 01039 マヨ	ネーズ 2 1	(個 1課税売上分·1 373 37 783 07/04/27
		×
		全額 1,096
修正	受注検索(S) [印刷(P) 削除([) 修正(M) 照会(Q) 登錄(C) 確定(C) 取消(C) 終了(X)
実行: ED_JUCHU		
_		, ,

種類	プログラム例	目的
受注取引画面参照	OQ_受注取引画面参照	受注データの一覧表示をします。
	5.156.010	
受注番号 顧客発注番	号 顧客CD	取引日 01/01/01 納品予定日 01/01/01
受注番号 [9/7°]取引日	納入予定日 顧客CD 顧客名	▲ 😽 顧客 🗐 出荷先 📀 その他
SO00000001 I:在庫 07/04/01	I 07/04/05 0110 エムエスジャパン	サービス
S000000002 I:在庫 07/04/02	2 07/04/03 0200 アベル 桑名支店	
SUUUUUUUU3 I:在庫 U//U4/U5) U//U4/U6 U12U 川田設備) 07/04/11 0500 N°Uつ7	
S000000004 I: 在庫 07/04/10 S000000005 I· 左庫 07/04/10) 07/04/11 0630 パロス	油田支店 -
S000000006 I:在庫 07/04/15	i 07/04/16 0630 秋田総合リース	神岡支店
S000000007 I:在庫 07/04/06	3 07/04/07 1800 イケダ重機	電話番号 XX87-72-2X36
S000000008 I:在庫 07/04/14	4 07/04/15 0620 コベツ秋田 由利	支店 XX6-355-6X45
SO00000009 I:在庫 07/04/09	9 07/04/10 0500 ハ°ル⊐ス	
S000000010 I:在庫 07/04/26	3 07/04/27 1280 アサジュウ 守谷3	友店 🔽
# 商品CD 商品名	販売数量 受注単位	単価 課税区分CD 課税処理 消費税金額 金額 納期日
<u>- 01052 ホット:</u> 0.01001 4.551		524 1 課税売上分一 1:税抜 524 11,004 07/04/16
2 01031 4-712		4,555 1 計机元工力一 1:税扱 1,230 20,243 07/04/16
	(roBi	
	[[[]]	
」 実行: ED_JUCHU		照会 挿入
		, , , , , , , , , , , , , , , , ,

種類	プログラム例	目的
印刷	OP_受注報告書印刷	受注データを印刷する。

例:OP_受注報告書印刷。

パラメータ入力画面は次のとおりで、受注番号などで範囲付けできます。

受注報告書印刷	
受注番号(開始) S00000002	
受注番号(終了) S00000003	
☑プレビューを行う?	
<u>実行(E)</u> 取消(<u>C</u>)	

印刷結果は、次のようなものとなります。

			受注幸	贤告書				ベージ:1 日付: 2007/08/24 時刻: 16:26:04
受注番号 \$000000 顧客 0110 ⊐ 顧客発注番号LS001 担当者 8 森川	1001 取引日 エムエスジャル 由香	3 07/04/01 納入予定日 07/04/05 ペンサービス 出荷経由 横浜	出荷先 摘要 決算条修	0 エムエスジャパンサ- 重要 牛 30 末締めの翌末払い	ービス		取引タ	イプ I:在庫
N0 商品⊐-ト [°]	商品名		販売数量	課税区分	経理処理	単価	消費税金額	金額 納期日
1 01000	コーン		20	1課税売上分一般仕入	1:税抜	75	75	1,575 07/04/05
2 01003	レタス		20	1課税売上分一般仕入	1:税抜	188	188	3,948 07/04/05
						合計	263	5, 523
受注番号 S000000 隋客 0200 7 隋客発注番号 A B A O 担当者 6 安部	1002 取引日 Pベル 桑名支 0 2 孝之	3 07/04/02 納入予定日 07/04/03 店 出荷経由 神戸	出荷先 摘要 決算条(0 アベル 桑名支店 宅急便 牛 28 末締めの翌20払い			取引タ	イプ I:在庫
N0 商品⊐-ト [°]	商品名		販売数量	課税区分	経理処理	単価	消費税金額	金額 納期日
1 01026	チキン胸		20	1課税売上分一般仕入	1:税抜	431	431	9,051 07/04/03
2 01052	ホットココ	זי	20	1課税売上分一般仕入	1:税抜	524	524	11,004 07/04/03
3 01065	コンパック	オフィスツール	20	1課税売上分一般仕入	1:税抜	1, 249	1, 249	26, 229 07/04/03
						合計	2, 204	46, 284
受注番号 S000000 顧客 0120 川 顧客発注番号 担当者 6 安部	1003 取引日 川田設備 孝之	∃07/04/05 納入予定日07/04/0€ 出荷経由	出荷先 摘要 決算条(1 川田設備一東京 キ 35 末締めの翌々5払い	,		取引タ	イプ :在庫
N0商品コード	商品名		販売数量	課税区分	経理処理	単価	消費税金額	金額 納期日
1 01065	コンパック	·オフィスツール	20	1課税売上分一般仕入	1:税抜	1, 249	1, 249	26, 229 07/04/06
2 01065	コンパック	オフィスツール	20	1課税売上分一般仕入	1:税抜	1, 249	1, 249	26, 229 07/04/06
						合計	2, 498	52, 458
马达来日 \$000000	1004 取引日 ゜ルコス	3 07/04/10 納入予定日 07/04/11 出荷経由 会況	出荷先 摘要	0 パルコス 重要 + 1 P5m4			取引タ	イプ I:在庫
受け上留ち 3000000 顧客 0500 パ 顧客発注番号 P001 担当者 6 安部	孝之		決算条件	+ I M204				
受注曲号 3000000 加 顧客 0500 加 顧客発注番号P001 担当者 6 安部 N0 商品コート [®]	孝之 商品名		決算条件販売数量	+ 1 2010月 課税区分	経理処理	単価	消費税金額	金額 納期日
	孝之 商品名 2 スベシャルテ	" #" {\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\	決算条件 販売数量 20	+ 1 2020年 課税区分 1課税売上分一般仕入	経理処理 1∶税抜	単価 1,250	消費税金額 1,250	金額 納期日 26,250 07/04/1

3.6 ホストコンポーネントの内容

ホストコンポーネントの目的は、3.1 コンポーネントの構成 で説明したように、業務ごとに分割したコンポーネントを一つにまとめることにあり、実質的な内容としては、メニューを提供する機能があるだけです。

3.6.1 モデルリポジトリ

ホストコンポーネントのモデルリポジトリでは何も定義されていません。

3.6.2 データリポジトリ

ホストコンポーネントのデータリポジトリでは、メニュー定義用のデータを格納するテーブルが定義されています。 このテーブルはメモリデータベースに作成されるもので、データはアプリケーションが起動されるたびに、テキス トファイルから読み込んできます。

8	8	Ŧ	一刻	ポジトリ							×
	ţ		名前		ディーダ	ソース名 デー	<u> </u>	7#I	が「公開名」		~
		1	EDM_	XII-	EDM_	,×⊐⊐≻ Mer	mory				
L											~
ſ	カラ	L	心	デックス 外部キー							_
	[#		名前	ŧ≓°∥			型	古書	^	
			1	×I⊐ − ID	62	ED_COMMON.	У Да HID	N=数值	4		
			2	親メニューID	62	ED_COMMON.	メニューID	N≕数值	4		
			3	メニュー名	63	ED_COMMON.	メニュー名	A=文字	30		
			4	公開ブログラム名	, 60	ED_COMMON.	ブログラム名	A=文字	30		

メニュー定義のテキストファイルは、ENV ディレクトリの EDM_メニュー.TXT です。

このファイルの内容は、階層的に定義されているメニューのデータで、1レコードが1メニュー項目に対応します。 各レコードでは、ノードID、親ノードID、メニューラベル、および選択時に呼び出すプログラムの公開名が定義さ れています。

1	0 受注入力	
2	1 受注入力	0C_受注入力
3	1 受注取引画面参照	0Q_受注取引画面参照
4	1 受注報告書印刷	OP_受注報告書印刷
10	0 マスタメンテナンス	
11	10 顧客マスタ照会	OM_顧客マスタ照会
12	10 商品マスタ照会	OM_商品マスタ照会
13	10 決済条件マスタ照会	OM_決済条件マスタ照会
14	10 担当者マスタ照会	OM_担当者マスタ照会
15	10 単位マスタ照会	OM_単位マスタ照会
16	10 課税区分マスタ照会	OM_課税区分マスタ照会
17	10 消費税マスタ照会	OM_消費税マスタ照会
20	0 印刷	
21	20 受注確認書印刷	OP_受注確認書印刷
22	20 受注報告書印刷	OP_受注報告書印刷
90	0 スペシャル	

91	90 サンプルデータ入力	BT_サンプルデータ入力
92	90 メニュー照会	OM_メニュー照会

3.6.3 プログラムリポジトリ

ホストコンポーネントのプログラムリポジトリでは、メニューを実現するプログラム、およびメニューデータの保守のためのプログラムが定義されています。

新日	^
10.0 /0.0	
08/08	l
08/22	
08/22	
08/22	
'08/10	
'08/22	
	-
	08/22 08/22 08/22 08/10 08/22

メニュープログラムは、下図のような外観を持ったオンラインタスクです。

メイン
受注入力
マスタメンテナンス
<u> </u>
スペシャル
終了(<u>X</u>)

ここでのプッシュボタンは、実は境界もスクロールバー、タイトルもないテーブルコントロールに表示されるレコードデータで、表示にエディットコントロールの代わりにプッシュボタンコントロールを使っているものです。

メニューに対応するプログラムを呼び出すときには、コール 公開名 コマンドを使い、メニューレコードに定義されている公開プログラム名を参照して呼び出しを行います。

💙 ५८१	:لا_00 - 6	コー表示					
データビ	ビュー ロジッ	クフォーム					
1	🖂 T=937	P=前					~
2	2 34	P=フ*ロク*ラム 5	BD_メニュー定義削	除 - 「4 #85-4 b]			
3	ן בן דביעייוג ו ⊐ דביעייוג	P=7.49.56 3	◎ BU_八/J - EUM_×L 11 宇行F	U0_7%~%J	77-74 9=#7*44-		
5	ים ב-וי אר ז דיים לים	I=If 4	し、それて 【公開プログラム	名=''	A-1 / 5-7/ //		
6	3 3-16	P=7°ロケッシー 6	00_メニュー表示				
7	לים ר	E=Else \	/es				
8	3 (14	N=公開名	公開プログラム名	[0 /*5X-5]	戻り値: ???	条件 Yes	
9	у <u>сво</u>	NEEnd	}				
	ール公開フロ	グラム					
	- 公開ブログ· 呼び 場合 ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・	ラム名 出すプログラムの は、キャビネット の ⁵ 弘名:	公開名を指定してくだ ・ファイル名も指定して ③ 公開ブロ	さい. プログラム ください. グラム名	が異なるキャビネッ	ットファイルにある	,
	ተየር ሱነ	07771104	U				

3.6.4 アプリケーション特性

ホストコンポーネントのアプリケーション特性では、一箇所だけ変更を行っています。それは、Enter キーを次項目 イベントにマップするために、独自のキーボード定義ファイルを使っていることです。

実行用キーボード書	引付: C:¥Magic¥v10_	coding_samp	ple2_app¥ENV¥act_rnt.jpn	X
開発(<u>S</u>) 実行	ĵ(<u>R</u>)			
# 内部へい	ŀ + -	२२-५२ 🗸	A	
265 次項目	Down	1		
266 次項目	Left	1		
267 次項目	Rght	1	- キーボード状態	
268 次項目	Tab	0	#1. tri	
269 次項目	Enter		***	
270 次文字	Rght	0	#2: なし	
271 式(E)	Ctrl+E	0 🗧	= #3: tal.	
272 式定義(E	E) Ctrl+E	0		-1
273 式表示	Ctrl+H	0 💽		
<u></u>	ОК	++>tell		

このキーボード定義ファイルは、ENV ディレクトリの下の act_rnt.jpn です。

アフリケーション特性	
スタートアップ(S) 外部参照ファイル(E) セキュリティ(R) 外部参照ファイル指定 このプロジェクトがアプリケーションの イルを指定します.コンボーネントとして 効になります.	メインとなる場合に参照する外部ファ てオーブンされる場合は、この設定は無
プリンタ属性ファイル: HTMLスタイルファイル: アプリケーション基本色定義ファイル: 内部基本色定義ファイル:	
アプリケーションフォント定義ファイル: 内部フォント定義ファイル:	
実行用キーボード割付ファイル:	XED ROOTXENV¥act rnt.jpn

3.7 モジュール定義

モジュール定義というのは、V10 で新しく導入されたもので、コンポーネントなどの関連するプロジェクトをモジュールとして参照の定義を記録しておく機能です。

3.7.1 モジュールの表示

モジュール定義は、ナビゲータからモジュールペインを開くことによって、見ることができます。

例えば、データコンポーネント ED_DATA プロジェクト に定義されているモジュールを見るには、ナビゲータ を表示した状態で、ナビゲータのコンボボックスから **モジュール**を選択します。



すると、モジュールの依存関係が表示されます。この 場合、ED_DATA が、ED_COMMON コンポーネントに 依存していることを表しています。



3.7.2 モジュールのネスト

モジュールはネストして定義することも可能です。すなわち、ED_DATA プロジェクトは ED_COMMON プロジェクトに依存していますが、ED_JUCHU プロジェクトが ED_DATA コンポーネントに依存しているので、ED_JUCHU \rightarrow ED_DATA \rightarrow ED_COMMON という依存関係があることがわかります。

このような場合に、ED_JUCHU プロジェクトで、ED_DATA プロジェクトをモ ジュールとして追加すると、モジュールペインには、ED_DATA → ED_COMMON という依存関係も自動的に取り込まれ、表示されるように なります。(右図)

また、一番トップの ED_HOST プロジェクトは、ED_JUCHU プロジェクトに 依存しているので、ED_HOST プロジェクトを開いてモジュールペインを表 示させてみれば、

 $\mathsf{ED}_\mathsf{HOST} \to \mathsf{ED}_\mathsf{JUCHU} \to \mathsf{ED}_\mathsf{DATA} \to \mathsf{ED}_\mathsf{COMMON}$







3.7.3 モジュールの用途

モジュールにより、プロジェクト間の依存関係を定義することはわかりましたが、実際的にはモジュールを定義することによりどういった利点があるのでしょうか?

まず、第一には、プロジェクトのオープンの手間が若干省略されます。モジュールペインを表示している状態で、 依存しているモジュールのノードをマウスでダブルクリックすると、そのプロジェクトが開きます。モジュールの依 存関係が定義されているならば、再帰的に全モジュールの依存関係が表示されるので、複数のコンポーネント からなるアプリケーションの場合に、いちばんトップのプロジェクト(サンプルでは ED_HOST)だけを最初に開い ておけば、あとはモジュールペインからコンポーネントのプロジェクトを開くことができます。

第二には、デバッグの際にソースコードが表示されることが、大きな利点として挙げられます。デバッグを行って いる場合、コンポーネントの中のプログラムについては、通常はソースコードを表示することができません。従っ て、ステップ実行やブレークなどを入れて実行の流れを制御観察することが困難になります。しかし、モジュー ルとしてコンポーネントのプロジェクトが定義されていたならば、Magic Studio はコンポーネントの中にまで入っ て実行制御やブレイクポイントを行うことが可能になります。

3.7.4 モジュールの追加

サンプルプログラムでは予めモジュール定義がされていましたが、モジュール定義は自動的に登録されるもの ではなく、あくまで、開発者が自分でモジュールの追加を行う必要があります。

モジュールの追加は、次のように行います。これは、ED_DATA プロジェクトモジュールが定義されていない状態で、ED_COMMON プロジェクトをモジュールとして追加する場合の例です。

まず、モジュール ペインを開きます。 前述のように、ナビゲータを表示し ている状態で、コンボボックスから **モジュール**を選択します。

次に、ポップアップメニューから、モ ジュール追加 メニューを選択しま す。

JC^J^~9	×
ሃቱ°୬°ኑሃ	~
りず*ジトり 	
ארי אריק לעק על געריק גרי ארי ארי	
972-7 E9°a-1	
📲 権利 (0)	
🗐 אבבא (2)	
🚠 ጋጋቱ "- ትጋኑ (1)	



プロジェクトを選択 ダイアログが開 くので、モジュールとして追加した いプロジェクトファイルを指定し、 OK を押します。

プロジェクトを選択					? 🛛
ファイルの場所型:	COMMON		*	G 🕫 🖻 🖽	
最近使ったファイル	Exports Source ED_COMMON.edp				
6 77.01-97					
ک ۲۲ ۴キ۱۶۷					
ער אד דארב אד					
マイ ネットワーク	ファイル名(<u>N</u>):	ED_COMMON.edp		~	■【◎】
	ファイルの種類(工):	eDeveloper ንግንንጉንትን	ファイル(*.edp)	~	キャンセル
ታ <u></u> ኒ		×			
€୬°⊐−₩		*			
😑 遶 [ED_DATA]					
É	ED_COMMON	N			

モジュールが追加されました。

ー般的には、コンポーネントとして使っているプロジェクトをモジュールとして定義しますが、コンポーネントリポ ジトリにコンポーネントとして使うプロジェクトを登録しても、自動的にモジュールとして定義されるわけではあり ません。

また逆に、コンポーネントリポジトリに取り込まれていないプロジェクトであっても、モジュールとして追加することができます。これは、コンポーネントリポジトリには登録しないで、コール公開プログラムにより呼び出すコンポーネントを使う場合に便利です。

Magic eDeveloper V10



Magic eDeveloper V10 コーディングサンプル Version 2 (コンポーネント編)

Copyright $\ensuremath{\textcircled{O}}$ 2007, Magic Software Japan K.K., All rights reserved.

第1版 2007年9月19日

発行〒151-0053 東京都渋谷区代々木三丁目二十五番地三号
あいおい損保新宿ビル 14 階
マジック ソフトウェア・ジャパン (株)
http://www.magicsoftware.co.jp/