MDG Technology feature guide

by SparxSystems Japan

MDG テクノロジー 機能ガイド

(2016/10/07 最終更新)

1　はじめに

このドキュメントでは、Enterprise Architect が提供する強力なカスタマイズ機能である「MDG テクノロジー」について説明します。この機能を利用することで、UML のような独自のモデリング言語(Domain Specific Modeling Language：DSM 言語)を定義することができます。

具体的には、以下のような内容を独自に定義することができます。

- 独自の要素やダイアグラム
  独自の要素やダイアグラム(図)を定義することができます。要素については、外見(表現)をカスタマイズすることができます。

- 要素間の関係
  Enterprise Architect の特徴的な機能であるクイックリンクについて、表示される選択肢を自由にカスタマイズできます。また、Enterprise Architect の「評価」の機能にルールを追加することで、独自のルールや制約条件に合致しているかどうかを確認することができます。

- 独自要素から生成されるソースコードの内容
  作成したモデルから出力されるソースコードの内容を定義することができます。これにより、作成したモデルから「スケルトン」以上の結果を得ることも可能です。

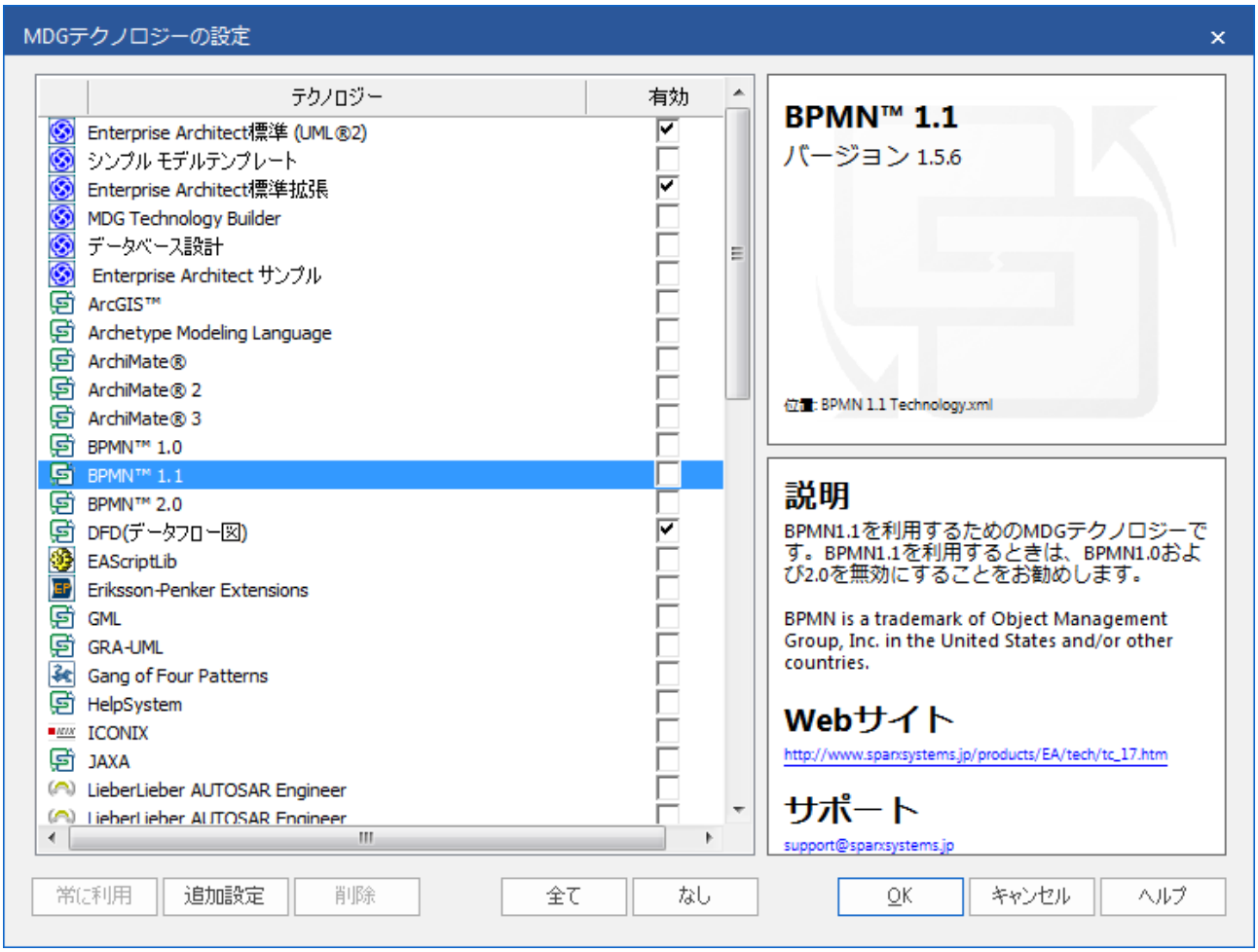
そのほか、Enterprise Architect でのモデルの作成を支援するために、以下の内容を定義することができます。

- 独自のパターン
  (要素間の関係を定義したもの)
- 独自のモデルパターン
  (パッケージ・ダイアグラム・要素を含む構成情報)
- 独自のヘルプ情報
- 独自のドキュメント生成ルール
- 独自のスクリプト
- 独自の検索ルール・ビュー
- 独自のプロパティ画面

また、MDG テクノロジーを利用することで、以下の支援機能を利用することができます。

- カスタマイズした全ての情報を単一の XML ファイルとして出力。そのファイルを配布することで、複雑な設定をせずに利用可能
- ネットワークドライブや Web サーバに MDG テクノロジーファイルを配置すれば、それぞれの利用者は自動的に最新の設定情報を取得可能

作成した MDG テクノロジーは、以下のような画面で利用の有無を選択することができます。
(「アドイン・拡張」リボン内の「MDG テクノロジー」パネルにある「設定」ボタンで開くことができます。)



なお、この画面で UML に関する情報を「無効」に設定することで、Enterprise Architect を独自に作成したモデル要素のみを表示させることも可能です。例えば、UML を無効にし、BPMN のみを有効にすれば、Enterprise Architect を BPMN 専用のモデリングツールとして活用できます。

以下、このドキュメントでは、このような独自のモデリング言語の定義を「独自モデル」と表現します。

なお、個別の機能の詳細についてはこのドキュメントでは触れません。ヘルプや PDF ドキュメントをご覧ください。PDF ドキュメントは、以下のページからダウンロードできます。
https://www.sparxsystems.jp/products/EA/ea_documents.htm

このドキュメントでは、Enterprise Architect 13.0 ビルド 1304 を利用しています。異なるバージョンの場合、操作方法が異なる場合があります。

2　独自モデルの定義の流れ

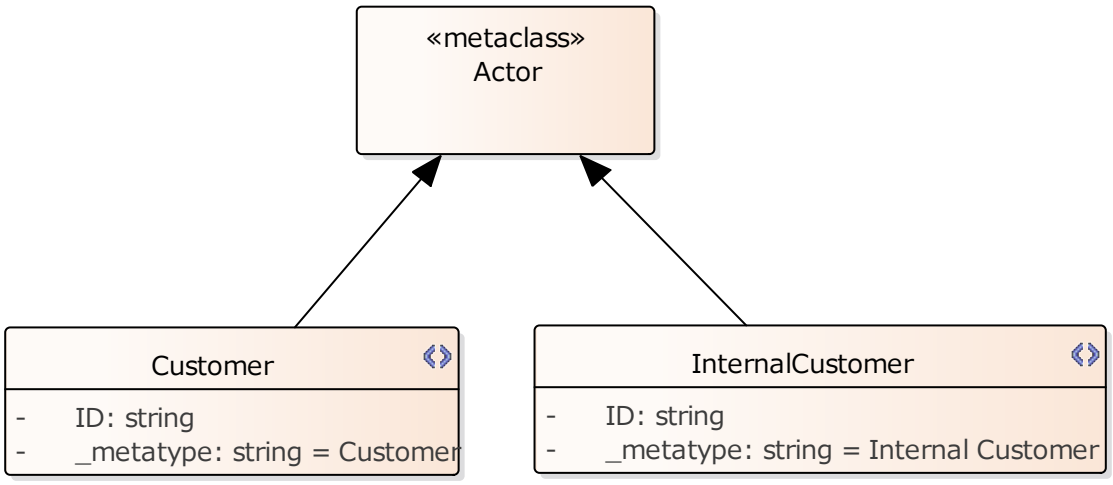
Enterprise Architect で MDG テクノロジーを利用して独自モデルを構築するまでの大まかな手順としては、次のようになります。

2.1　モデルの定義

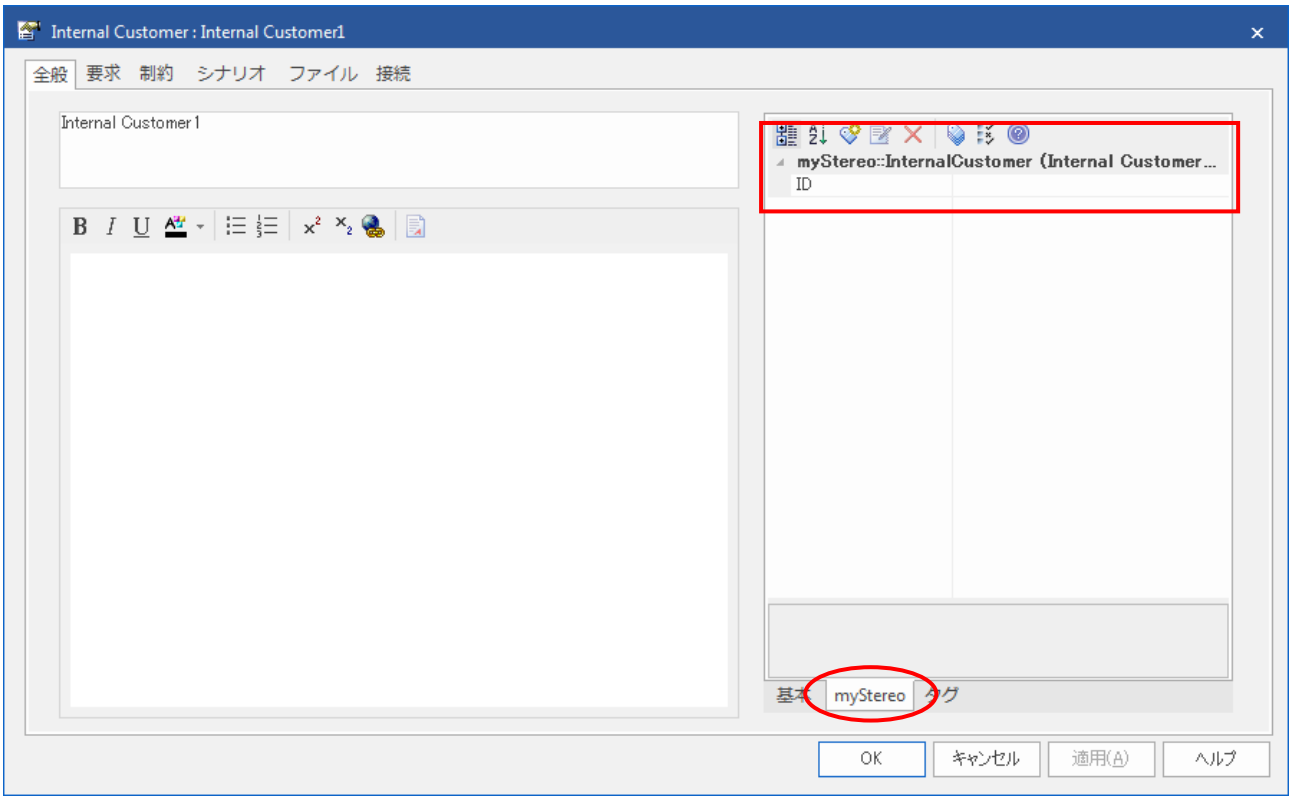
まず、独自モデルを利用してモデリングを行う場合に利用する要素や接続を定義します。この定義のためには、「UML プロファイルの定義」機能を利用します。UML プロファイルの定義では、UML の要素をベースに独自の要素や接続を定義します。この定義作業も Enterprise Architect で行います。

要素や接続には、関係するプロパティが定義されることが一般的です。ここでは、そのプロパティについても定義します。

以下の例は、アクター要素を元に拡張し、「Customer」「InternalStakeholder」という 2 つの要素を定義した例です。それぞれ、プロパティ「ID」を持つことを示しています。
(名前の先頭が_(下線)の属性は、EA 内部の処理のための情報を示します。)



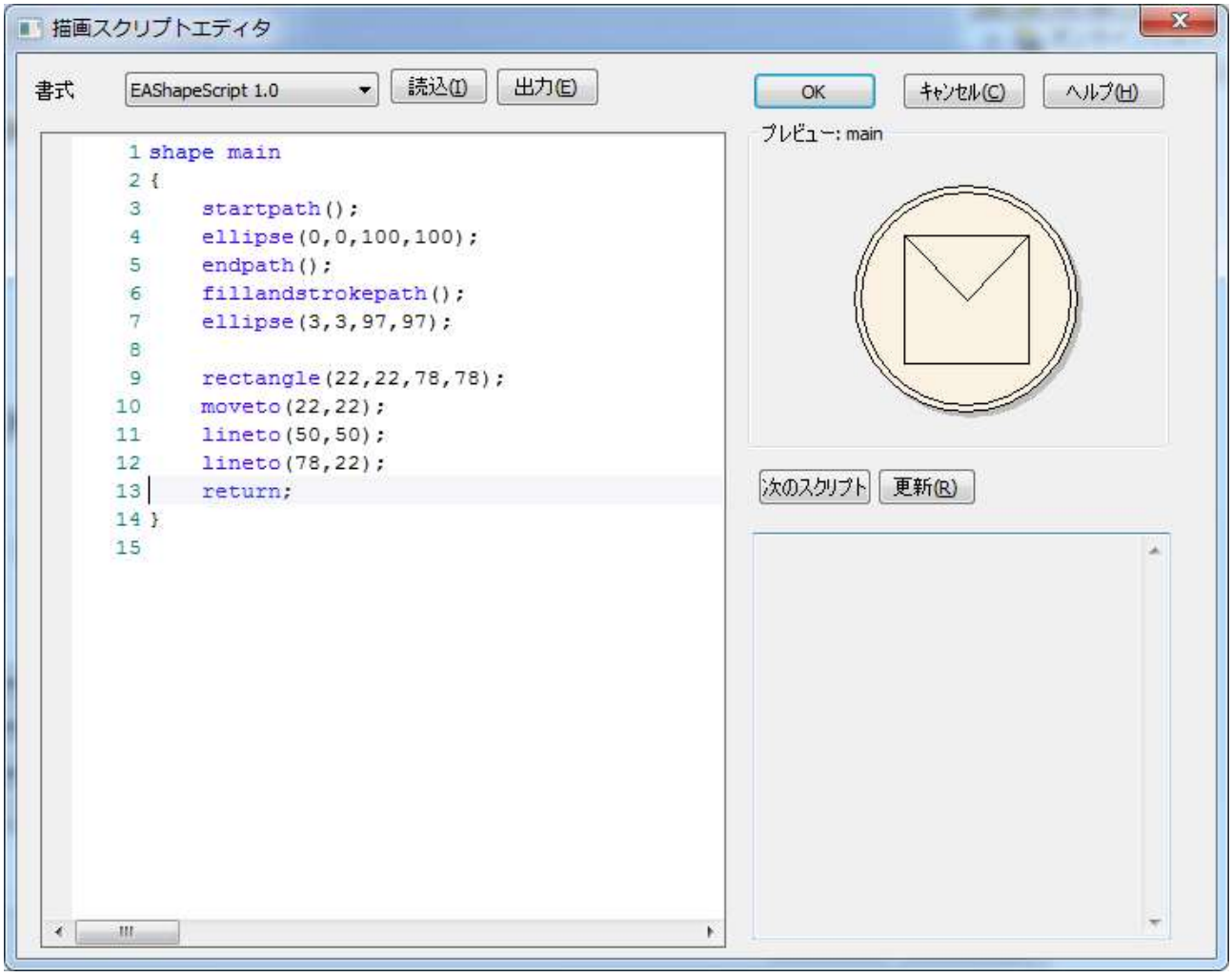
なお、ここで定義した要素をモデルに作成すると、プロパティは以下のようにプロパティ画面に表示され、参照・設定できます。



UML プロファイルについての概要は、PDF ドキュメント「UML プロファイル 機能ガイド」をご覧ください。

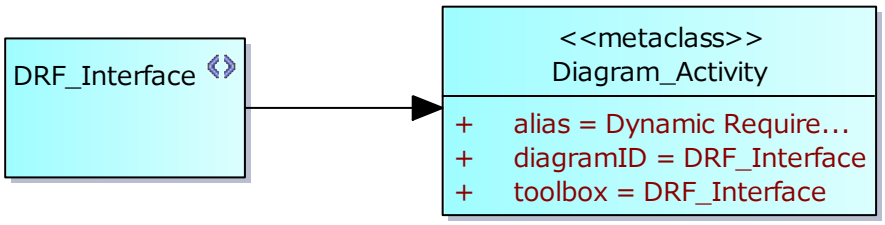
作成する要素や接続の外見を変更する場合には、「描画スクリプト」機能を活用します。描画スクリプト機能を利用すると、プロパティの値に応じて外見を変化させることもできます。

以下の図は、Enterprise Architect の描画スクリプトの定義画面です。



描画スクリプトの詳細やサンプルについては、ヘルプをご覧ください。

また、独自のダイアグラム(図)の種類を作成する場合にも UML プロファイルの定義で行います。具体的には、以下のようなモデルで定義します。



(DRF_Interface が追加するダイアグラムの種類。Diagram_Activity(アクティビティ図)を拡張することを示しています。)

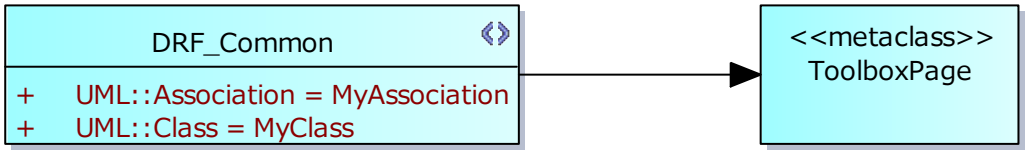
独自ダイアグラムの定義については、ヘルプの「ダイアグラムプロファイル」のページをご覧ください。

2.2　制約の定義

次に、作成した独自モデルが持つ「制約」に関する定義を行います。具体的には、「ツールボックスの定義」「クイックリンクの定義」「評価ルールの定義」になります。

ツールボックスの定義は、ダイアグラムを開いた際に表示される内容を定義します。これにより、ダイアグラムでモデルを作成する場合に利用する要素や接続を提示することができます。ツールボックスは、UML プロファイルを作成するだけで自動的に追加されますが、任意の項目・順番を指定することもできます。

ツールボックスの内容を任意に指定する場合に作成するモデルは次のような形になります。



(DRF_Common という名前のツールボックスを作成し、その中に 2 つの項目が含まれることを示しています。)

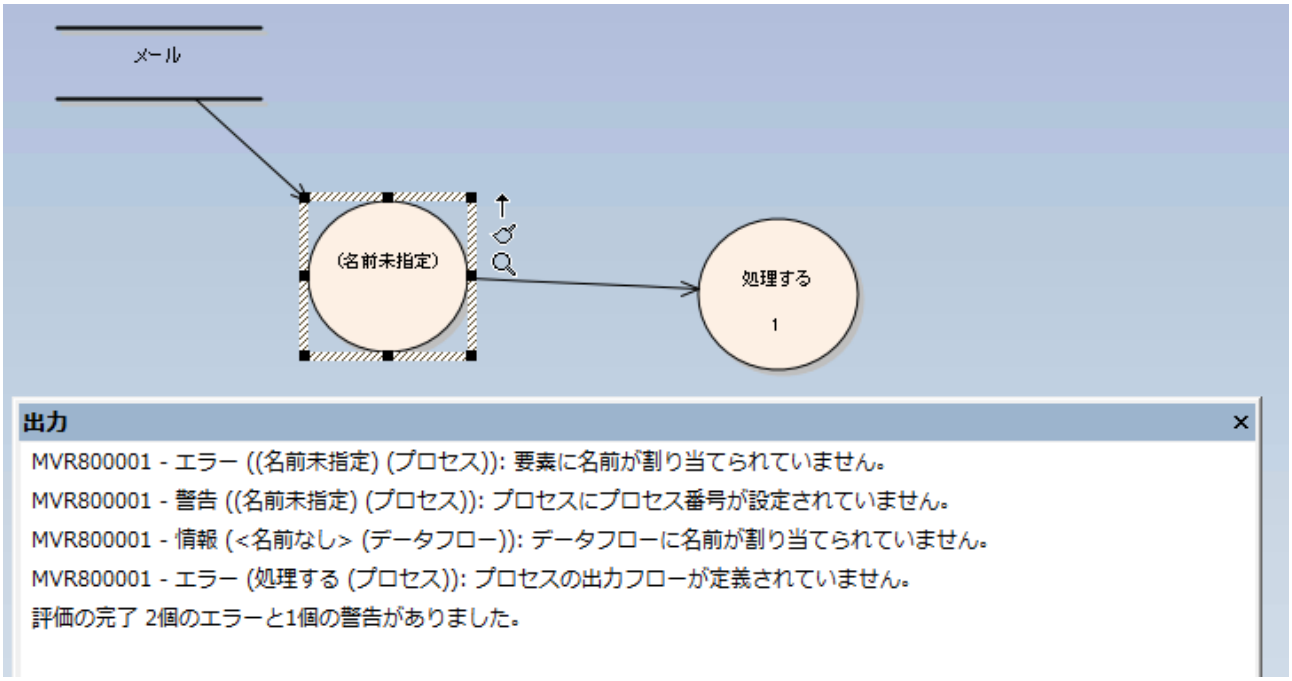
ツールボックスの定義については、ヘルプの「ツールボックスプロファイル」のページをご覧ください。

また、クイックリンクのルールをカスタマイズすることで、要素間の接続の種類を制約することができます。クイックリンクを利用することにより、効率的なモデリングが可能になります。クイックリンクの定義は CSV 形式で行います。詳細については、ヘルプの「クイックリンクプロファイル」のページをご覧ください。

ただし、現在の Enterprise Architect の動作としては、ツールボックスから接続を選択し、ルールに反する形で要素間を結びつけることも可能です。この対策の一つとして、「評価」のルールを追加し、独自モデルでのルールを検出するための仕組みを構築することができます。設計者は、随時「評価」の機能を実行することで、ルールに反していないかどうかを確認することができます。

あるいは、アドインの機能を利用することで、要素間に接続を作成した時に即時に検知(EA_OnPreNewConnector イベント)したり、ダイアグラムで利用できない要素を配置しようとした(EA_OnPreNewDiagramObject イベント)ときにエラーを表示したりするなどの動的な処理を行うことも可能です。

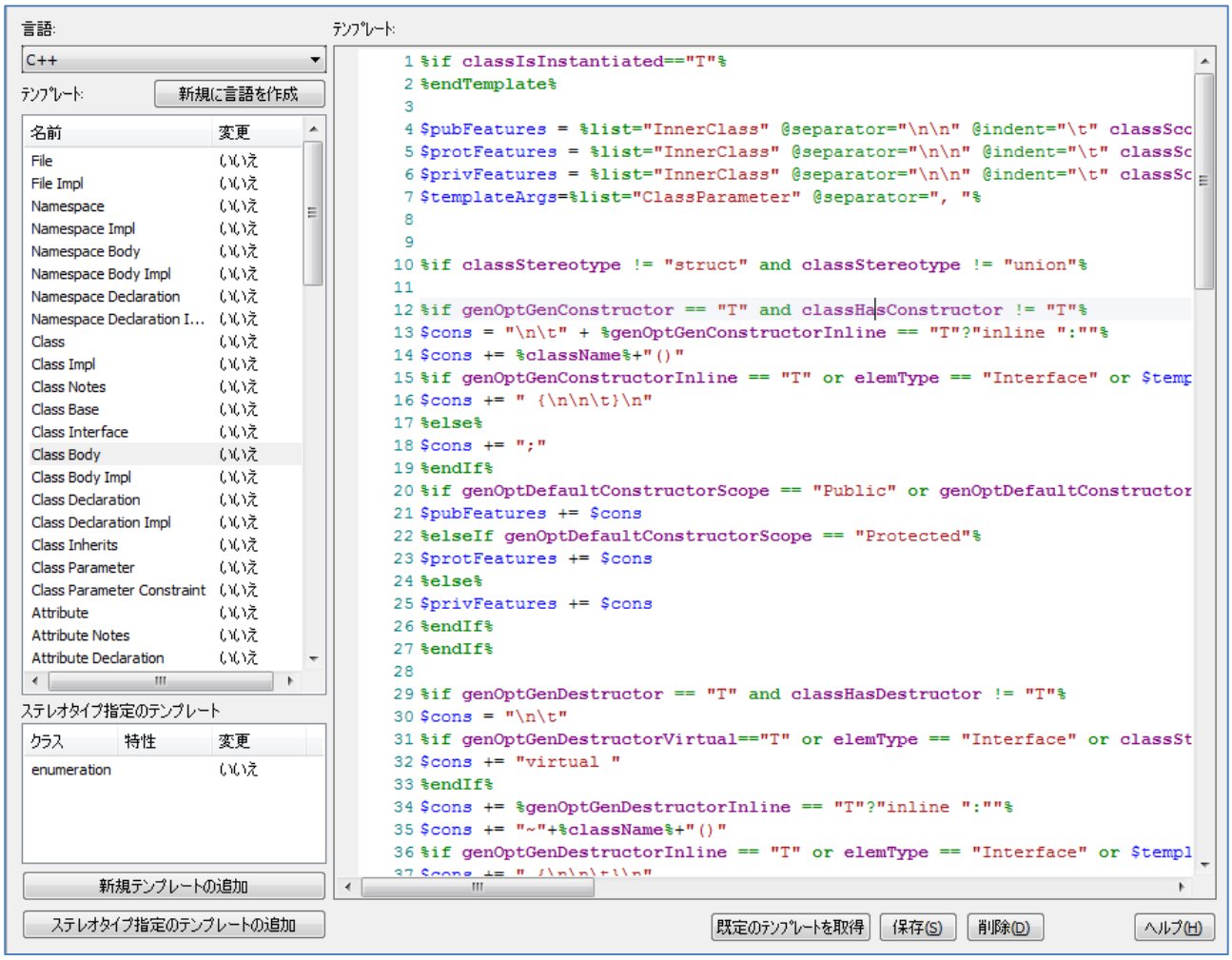
下の例は、DFD の拡張機能で、評価機能を実行した例です。評価結果の一覧に表示される内容をダブルクリックすると関係する要素に移動することができますので、確認・修正が容易です。



2.3　ソースコード生成の定義

最後に、作成した独自モデルからソースコードを生成する場合の出力結果を定義します。この場合には、「コード生成テンプレート」をカスタマイズします。なお、ソースコードが生成できるのは UML のクラス要素を拡張した要素のみとなりますので、ソースコードを生成する元となる要素は、クラス要素を「拡張」して定義してください。

ソースコード生成の定義のカスタマイズについては、PDF ドキュメントの「コードテンプレートフレームワーク(CTF) 機能ガイド」をご覧ください。以下の図は、コード生成テンプレートエディタの画面です。



なお、Enterprise Architect Suite の機能を利用すると、アクティビティ図・シーケンス図・ステートマシン図(を元にした独自のダイアグラム)からソースコードの実装を出力することができます。この機能の概要については、PDF ドキュメント「アクティビティ図・シーケンス図からのコード生成 機能ガイド」および「ステートマシン図からのコード生成 機能ガイド」をご覧ください。

3　支援機能の定義の流れ

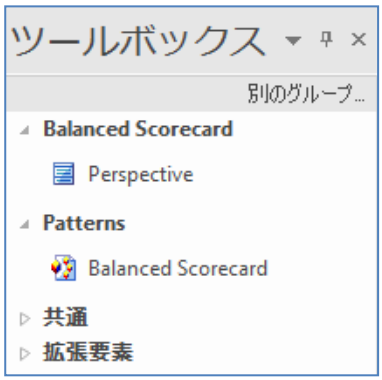
次に、独自モデルの作成を支援するために利用できるさまざまな機能の概要を説明します。

3.1　独自のパターン

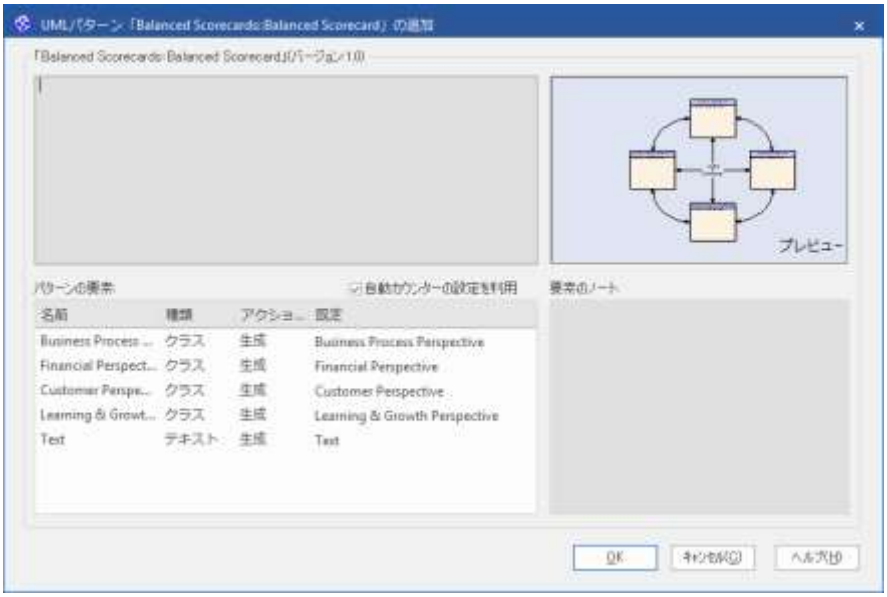
Enterprise Architect が提供する「UML パターン」の機能を利用すると、独自のパターンを定義することができます。パターンとは、要素と接続の組み合わせのテンプレートです。有名なパターンとしては「デザインパターン」があります。

このような、モデルに出現するパターン(部品)を定義し、ツールボックスに表示することが可能です。パターンを利用したい場合には、ツールボックスからドラッグ&ドロップします。

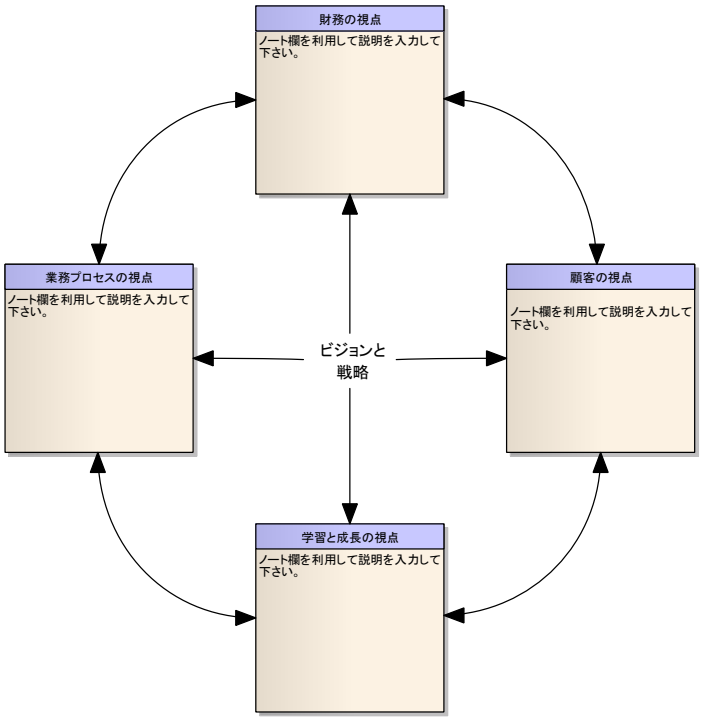
具体的な例として「戦略モデリング」テクノロジーの「バランススコアカード」では、「Balanced Scorecard」というパターンが定義されています。このパターンをダイアグラムにドラッグ&ドロップすると以下のようなダイアログが表示され、定義されたパターン(テンプレート構成)をダイアグラムに追加することができます。



ツールボックス



設定画面



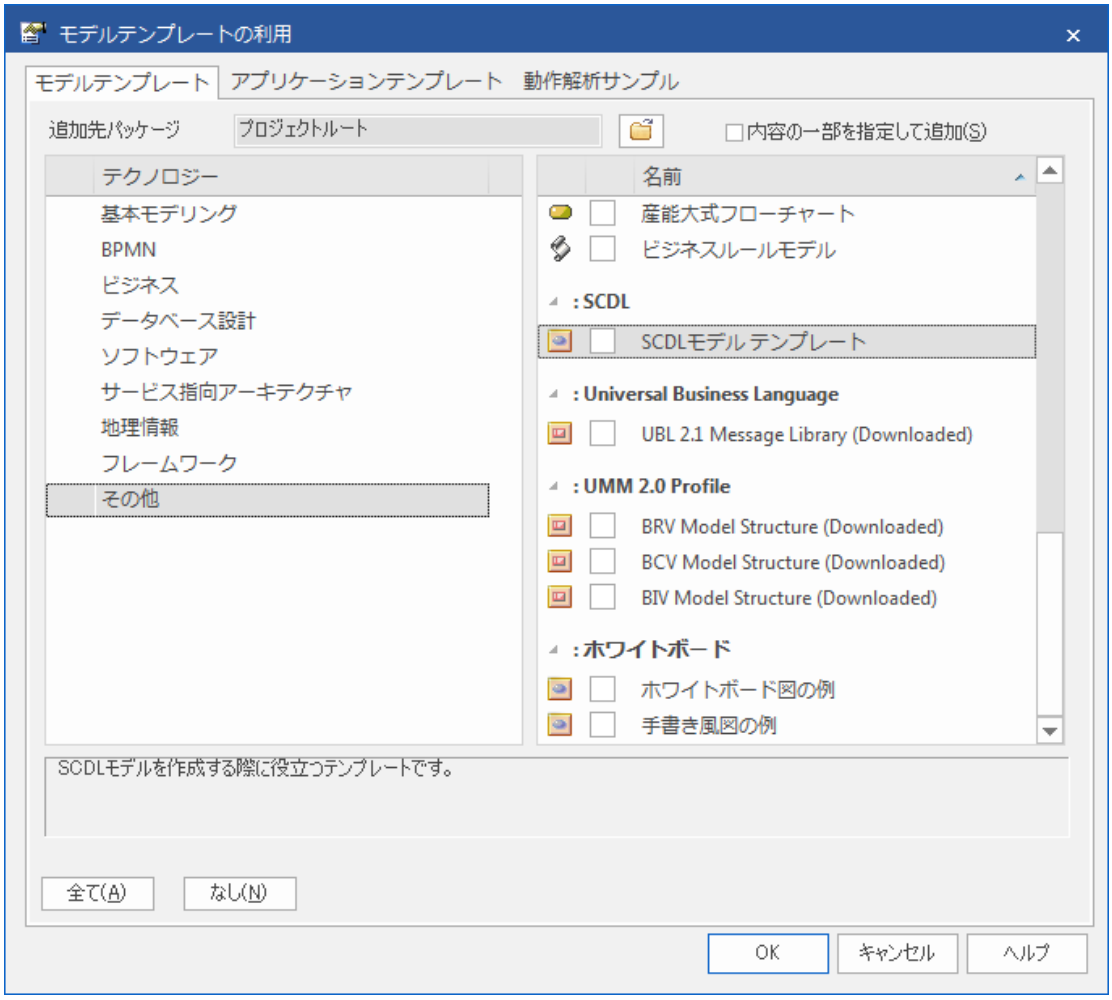
ドロップした結果を編集した例

3.2　独自のモデルテンプレート

3.1 章のパターンは、要素と接続のみで構成される内容になりますが、モデルテンプレートは、ダイアグラム・パッケージを含めたモデル構造のテンプレートになります。パッケージの階層構造を含めて、モデルを構築する場合の適切な構成をテンプレートとして利用することができます。

設計者は、「モデルテンプレートの利用」機能を実行して利用するモデルテンプレートを選択できます。これにより、特にモデリングの最初の段階において効率的に進めることができるようになります。

具体的な例として、独自に追加した SCDL の場合に、「SCDL モデル テンプレート」として新規にプロジェクトを始める際のテンプレートを追加することができます。



選択画面

Enterprise Architect が提供しているモデルテンプレートのデータは、Enterprise Architect のインストールディレクトリにある「ModelPatterns」ディレクトリに格納されています。それぞれのデータは Enterprise Architect のパッケージを XMI 形式で出力したものです。独自に追加する場合には、MDG テクノロジーでファイル名を指定する必要があります。

この、モデルテンプレートは、独自のパッケージ構成などを定義してチーム内で利用する場合にとても便利です。このモデルテンプレートのみを含む MDG テクノロジーファイルの簡単な作り方と利用方法を 6.2 章で説明しています。

3.3　独自のドキュメント生成ルール

独自モデルで作成した設計情報は、ソースコードとして出力するほか、DOCX, PDF, RTF(Word 互換)のドキュメントとして出力することができると便利です。

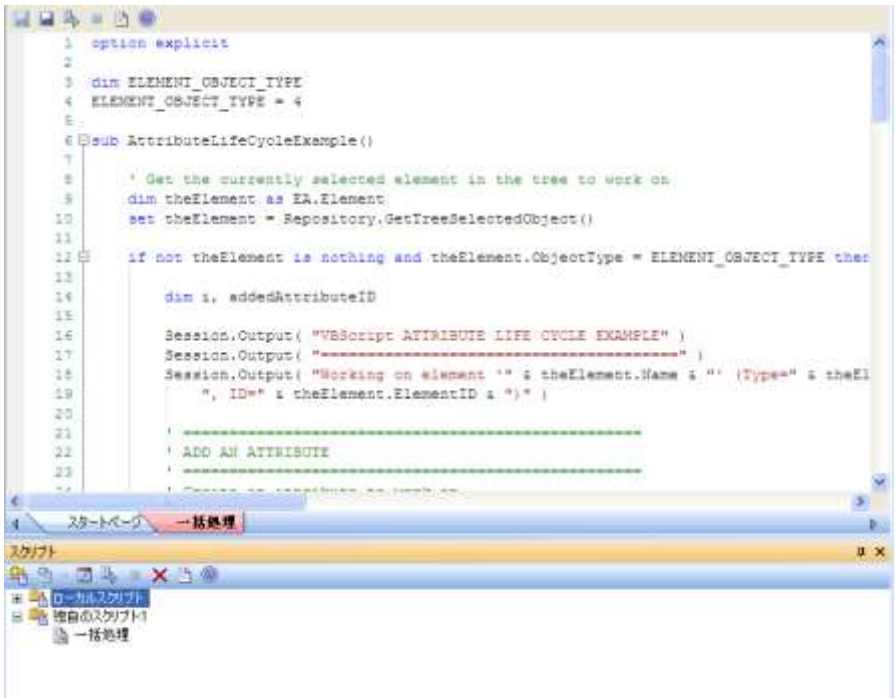
このドキュメントの出力内容は「ドキュメントのテンプレート」をカスタマイズすることで、自由に定義ができます。詳細は、PDF ドキュメント「ドキュメント出力機能 機能ガイド」をご覧ください。

3.4　独自のスクリプト

Enterprise Architect では、内部で API を利用してプログラムを作成することができます。例えば、全ての要素を対象に一括で処理を行うような場合に、要素の数が多い場合には手作業で処理を行うのは現実的ではありません。このような場合に、API を利用してスクリプトを定義して実行することができます。

定義したスクリプトも MDG テクノロジーの一部分になります。これにより、他の設計者も同じスクリプトを利用して作業を効率化することができます。

下の画像は、スクリプトエディタの画面です。スクリプトは、VBScript, JavaScript, Jscript のいずれかで作成できます。



定義したスクリプトは作成したプロジェクトファイルに格納されます。MDG テクノロジーのファイルを作成する際に、プロジェクトファイル内のスクリプトを出力することができます。

3.5　独自の検索ルール・ビュー

Enterprise Architect で独自に定義できる検索ルールやビューも効率的なモデリングを支援します。

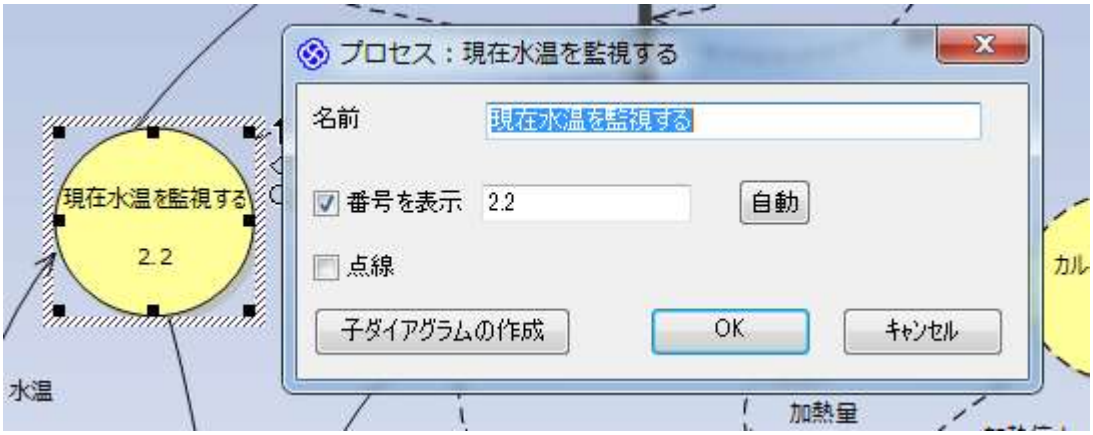
検索ルールは、モデル内の要素やダイアグラムを対象に、指定した条件で抽出することができます。例えば、指定した要素と独自モデルとして関係のある要素を抽出する場合などに活用できます。

ビューは、プロジェクトブラウザと同じように活用できますが、独自の条件で表示する内容を指定できます。例えば、独自モデルの制約を満たしていない要素を抽出して表示する等の対応が可能です。

独自のビューの定義については、ヘルプファイルの「ビューブラウザ」のページをご覧ください。MDG テクノロジーのファイルを作成する際に、プロジェクトファイル内のビューを出力することができます。

3.6　独自のプロパティ画面

独自モデルを作成しても、要素や接続をダブルクリックして表示されるプロパティ画面は Enterprise Architect 標準のままです。このプロパティ画面には、独自モデルで定義した専用のプロパティは表示されませんし、利用することのないプロパティも表示されます。下の画像は、DFD の要素をダブルクリックした場合に表示されるプロパティ画面です。



アドイン機能の EA_OnContextItemDoubleClicked イベントを利用することで、独自のプロパティ画面を作成することができます。ダブルクリックした要素の種類に応じてプロパティ画面を変えることもできます。
(このイベントの戻り値として True を返すと、Enterprise Architect 標準のプロパティダイアログは表示されなくなります。)

また、このような独自のプロパティ画面を作成することで、プロパティ内の制約や矛盾をチェックすることができます。例えば、ノート(コメント)を必ず入力しなければならない、というようなルールがある場合、独自プロパティ画面を閉じる前にチェックを行い、入力されていない場合にはエラーを表示する、という処理を実装できます。

4　MDG テクノロジーのサンプル

ここまで説明したような MDG テクノロジーを実際に利用した例は数多くあります。実際に、スパークスシステムズ ジャパンで提供している有償・無償のアドインはこの MDG テクノロジーの仕組みを利用しています。

無償の MDG テクノロジーの主なものと、独自に定義している内容をご紹介します。これらの MDG テクノロジーのうちのいくつかは、テクノロジーを定義している XML ファイル自身を参照することが

できますので、独自にファイルを作成する場合にも役立ちます。

- DFD(データフロー図)・PFD(プロセスフロー図)・BPMN
  ダイアグラム・要素(外見の変更を含む)・ツールボックス・クイックリンク定義・プロパティ画面(アドイン)・評価ルール(アドイン)
- 戦略モデリング
  ダイアグラム・要素(外見の変更を含む)・ツールボックス・クイックリンク定義・パターン

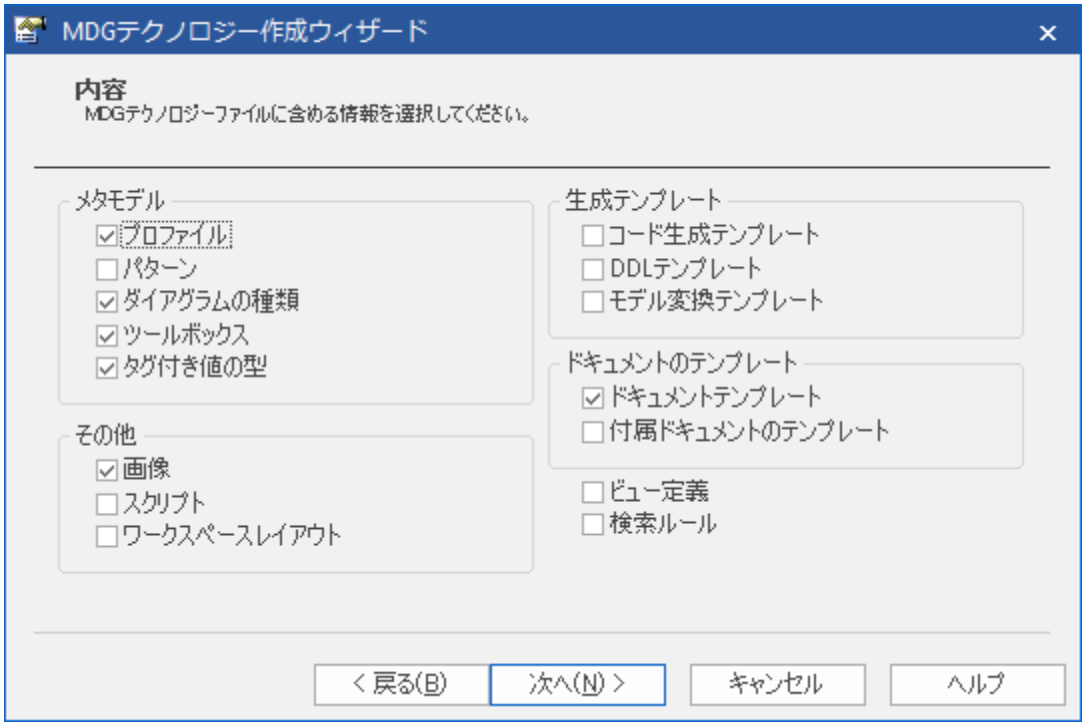
MDG テクノロジーのファイルは、Enterprise Architect のインストールディレクトリにある「MDGTechnologies」ディレクトリに格納されています。独自に作成した MDG テクノロジーファイルをこの位置に配置することで、Enterprise Architect で利用できるようになります。

(Windows Vista 以降の OS では、このディレクトリへの書き込みには管理者権限が必要です。「アドイン・拡張」リボン内の「MDG テクノロジー」パネルにある「独自拡張」ボタンを押すと表示されるメニューから「MDG テクノロジーファイルの読み込み」を実行することで、各ユーザーの権限でファイルを作成でき、Enterprise Architect から参照される位置に MDG テクノロジーファイルを配置(コピー)することができます。)

5　MDG テクノロジーの作成方法

MDG テクノロジーファイルを作成するには、2 章および 3 章で説明したさまざまな設定を 1 つの EAP ファイルに格納し、そのファイルで「MDG テクノロジー作成ウィザード」を実行します。

作成ウィザードでは、下記のような画面で対象のカスタマイズデータを指定することができます。



この MDG テクノロジー 作成ウィザードについての詳細は、ヘルプの「MDG テクノロジーの作成」

をご覧ください。

6　MDG テクノロジーの開発例

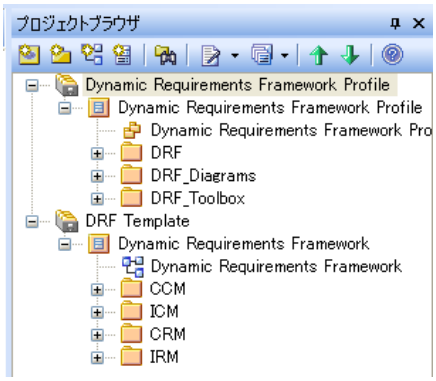
最後に、実際に独自の MDG テクノロジーを定義したサンプルを紹介します。

6.1　独自の要素やダイアグラムなどのサンプル

このサンプルは「Dynamic Requirements Framework」という要求を管理するためのフレームワークに関するプロファイルです。
このサンプルは以下の URL よりダウンロードできます。

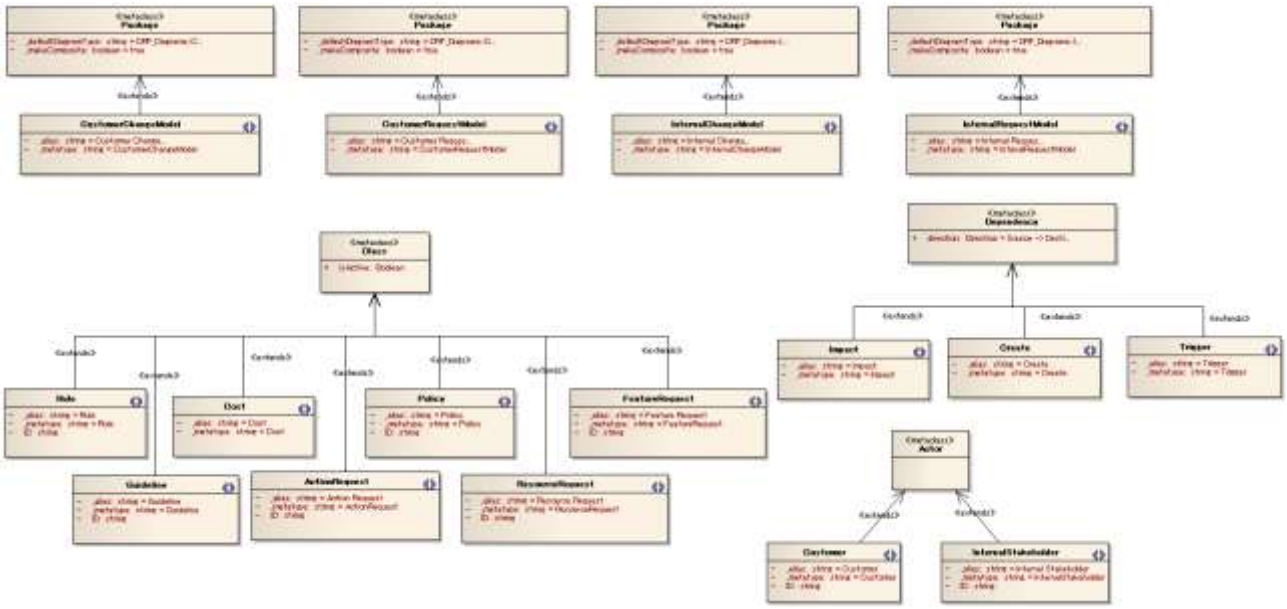
http://www.sparxsystems.com/downloads/whitepapers/assets/ea_framework_sample.zip

このサンプルに含まれる EAP ファイルは次のような構成になっています。



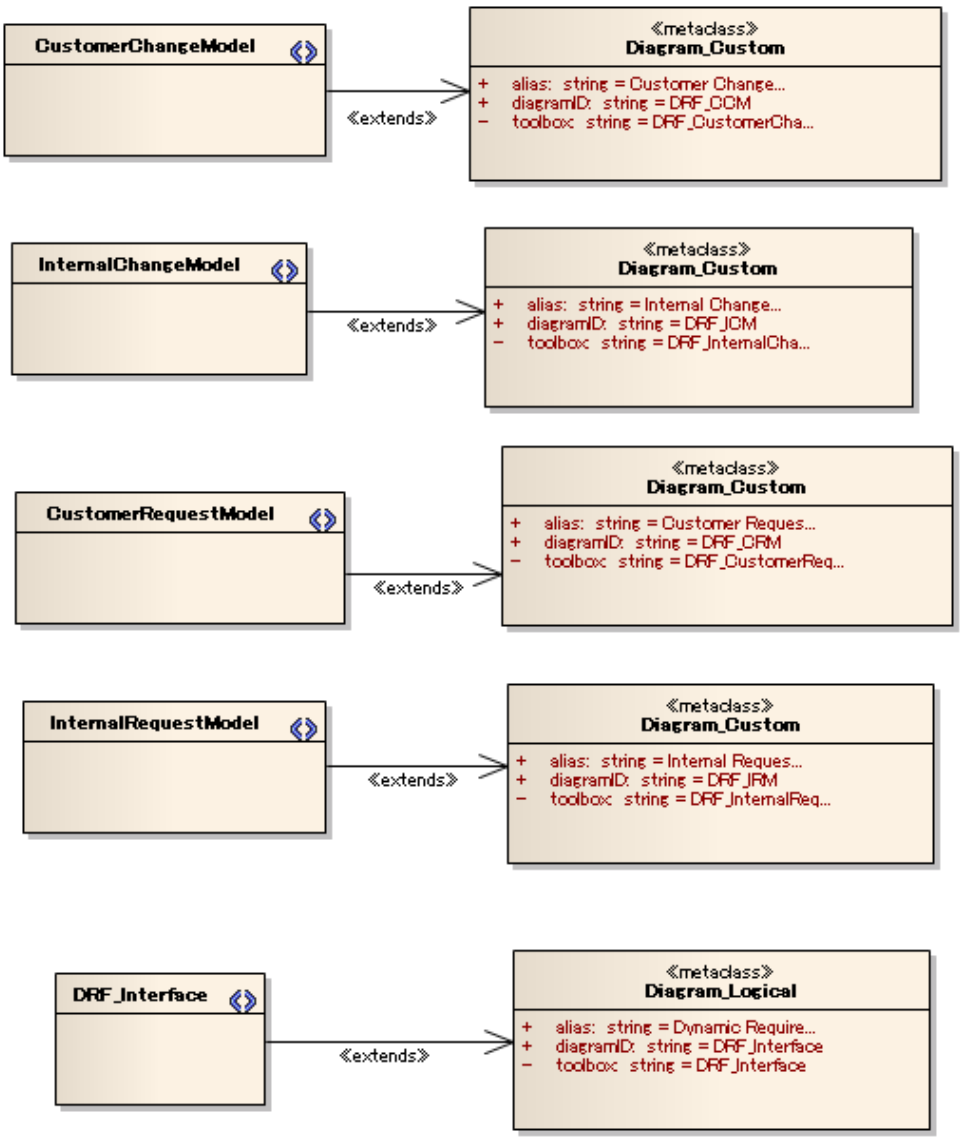
上の「Dynamic Requirements Framework Profile」プロジェクトルートに含まれる内容で、独自のモデルを定義しています。下の「DRF Template」プロジェクトルートは、モデルパターンとして提供するテンプレートです。

まず、「Dynamic Requirements Framework Profile」プロジェクトルートに含まれる「DRF」パッケージを確認します。このパッケージでは、UML プロファイルとして独自の要素を定義しています。定義してある内容は次の図の通りです。

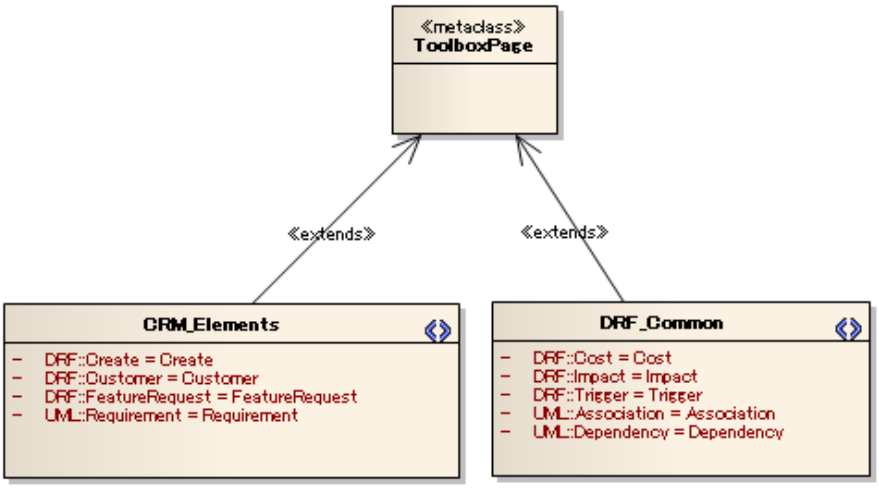


この定義のうち、クラスの右上に<>のマークがついているのが独自に追加する要素で、extends の関係でつながっているクラスが、Enterprise Architect がもつ、基底となるクラスです。

DRF_Diagram のパッケージでは、独自のダイアグラムの定義を行っています。左側が独自に追加するダイアグラムです。

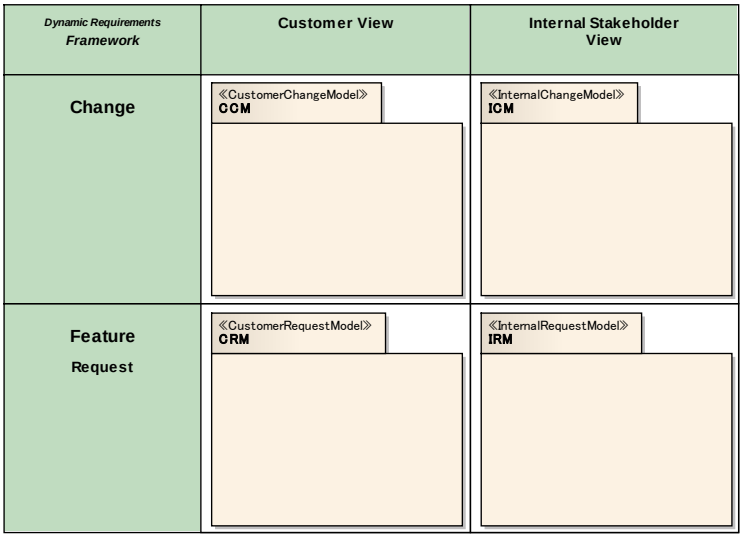


そして、DRF_Toolbox パッケージ内で、これらの独自要素と独自のダイアグラムとの関連づけを行っています。DRF_Toolbox パッケージ内には上記の 5 つのダイアグラムのそれぞれについて定義を行っているため、5 つのダイアグラムがあります。下の例はその 5 つのダイアグラムのうちの 1 つです。



今回のサンプルでは、独自のモデルの定義のほかに、テンプレートとなるモデルも定義しています。

それが、「DRF Template」プロジェクトルートです。このテンプレートに含まれるダイアグラムでは、次のような「目次」の画面を提供しています。



このようなテンプレートも作成・提供することで、独自モデルの活用の効率化を図ることができます。

ここまでのデータが作成できたら、MDG テクノロジーウィザードを利用して、MDG テクノロジーファイル(1 つの XML ファイル)を作成します。そして、その XML ファイルを所定の位置に配置すれば、利用可能になります。

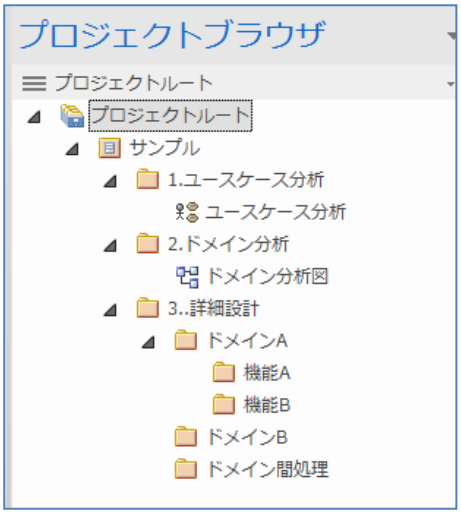
6.2 モデルテンプレートのみのサンプル

3.2 章で説明したモデルテンプレートは、チーム(複数人数)での継続的な設計の場合に活用すると、大きな効果を発揮します。具体的には、チームでの標準的なパッケージ構成をテンプレートとして定義しておくことで、新規にプロジェクトを作成した場合に、同じパッケージ構成で設計ができます。

同じパッケージ構成で設計ができることにより、例えばどのように UML で表現するのか、どこまで詳細に表現するのか、などの気になる点があれば、既存の他のプロジェクトファイルの同じ場所を見ることで、参考にすることができます。プロジェクトファイルごとにパッケージ構成がばらばらになっていると、このような場合に参照すべきパッケージ・ダイアグラムを見つけることは容易ではありません。

この、モデルテンプレートのみを含む MDG テクノロジーファイルは簡単に作成できます。作成方法と利用方法を紹介します。

まず、モデルテンプレートとして利用するパッケージ構成を Enterprise Architect を利用して作成します。空のプロジェクトファイルにパッケージやダイアグラムを追加しても良いですし、既存のプロジェクトファイルをコピーし、要素などの中身を削除して作成しても良いです。例えば、以下のようなパッケージ構成を定義したとします。



完成したら、テンプレートとして作成する最上位のパッケージ(上記の例の場合は「サンプル」のパッケージ)を右クリックし、「モデルの読み込みと出力」→「XMI ファイルへ出力」を実行し、内容を XMI ファイル(ファイルの拡張子は XML)として出力して下さい。

複数のテンプレートを利用可能にする場合には、この作業を繰り返して複数の XMI ファイルを作成して下さい。

次に、このモデルテンプレートを含む MDG テクノロジーファイルを作成します。メモ帳などのテキストエディタで、以下の内容を含むファイルを作成して下さい。

----ここから----

```
<?xml version="1.0" encoding="Shift_JIS"?>
<MDG.Technology version="1.0">
<Documentation id="MY_ID" name="MyTemplateName" alias="テンプレートグループ名" version="1.0" notes="全体の説明文"/>
<ModelTemplates>
<Model name="テンプレート名 1" description="テンプレート説明 1" location="TemplateFileName1.xml"/>
<Model name="テンプレート名 2" description="テンプレート説明 2" location="TemplateFileName2.xml"/>
</ModelTemplates>
</MDG.Technology>
```

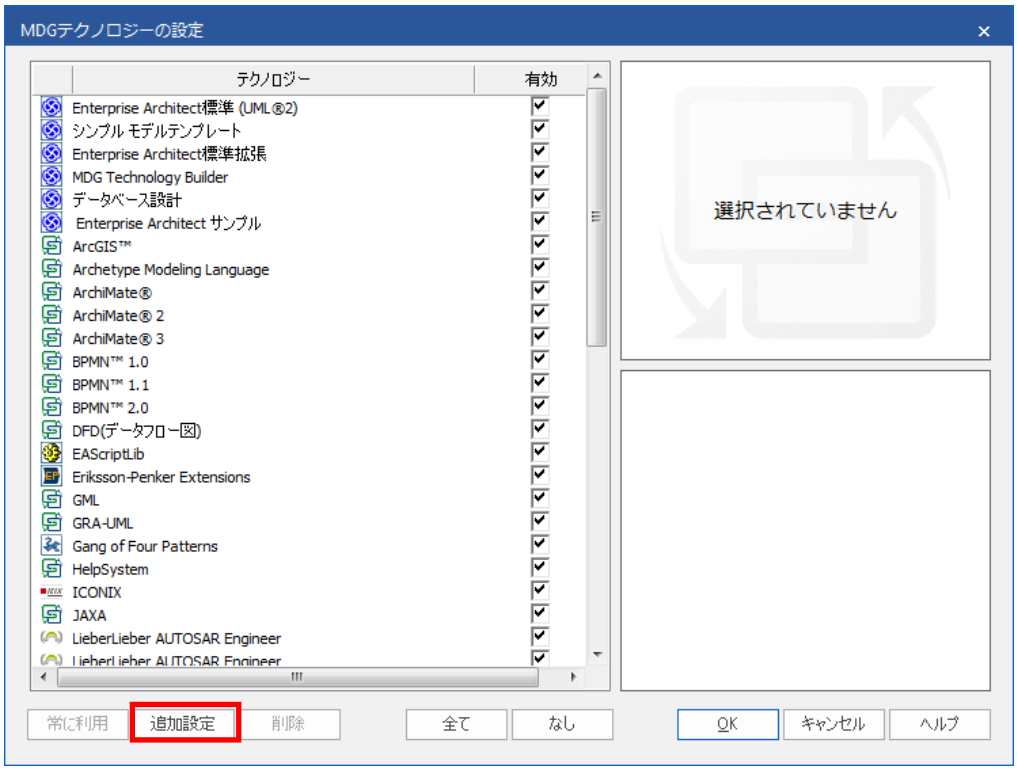
----ここまで----

このファイルで、赤字で記載されている部分については、必要に応じて内容を変更して下さい。最初の id と name は、アルファベットと数字のみで、先頭文字はアルファベットにして下さい。location の値は、先ほど出力したファイル名にしてください。必要に応じて、<Model name=で始まる行の数を増減させて下さい。完成したら、このファイルも拡張子 XML のファイルとして保存して下さい。

次に、このファイルと、先ほどの XMI ファイルをまとめて 1 つのフォルダに配置します。チームで

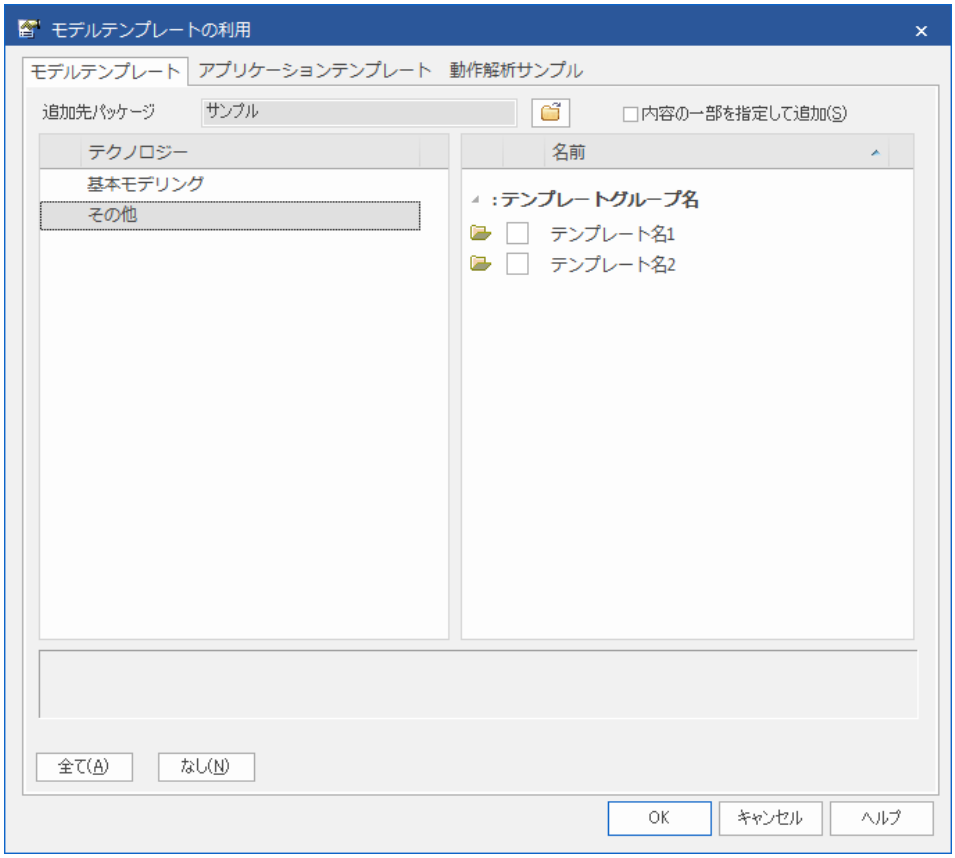
共有する場合には、ネットワークドライブの特定のフォルダに配置すると良いでしょう。

このモデルテンプレートを利用するためには、Enterprise Architect を起動し、「アドイン・拡張」リボン内の「MDG テクノロジー」パネルにある「設定」ボタンを押します。「MDG テクノロジーの設定」画面が表示されますので、画面左下にある「追加設定」のボタンを押してください。



すると、「MDG テクノロジー – 追加設定」画面が表示されますので、左下の「追加」ボタンを押し、サブメニューから「パスの追加」を選択します。そして、先ほどのファイルが格納されているフォルダを指定して下さい。

以上で、下の画像のように、モデルテンプレートの選択画面から、自分で作成したテンプレートを選択できるようになります。



改版履歴

2009/12/11 初版
2011/05/18 Enterprise Architect9.0 のリリースに伴い、内容を更新。
2011/12/21 Enterprise Architect9.2 のリリースに伴い、内容を更新。
2013/02/04 6.2 章を追加。
2014/01/30 説明の微調整・画像の差し替え。
2015/08/31 説明内容の一部を最新の内容に更新。
2015/12/01 Enterprise Architect12.1 のリリースに伴い、内容を更新。
2016/02/15 利用することができない情報が残っていた箇所を削除。
2016/10/07 Enterprise Architect12.1 のリリースに伴い、内容を更新。