

# メタモデルを変更する場合の考察

ドメイン固有モデリング言語を実開発で使用するには、ライフサイクルを通じた修正やメンテナンスを避けることはできない。少なくとも、その製品系列が開発・メンテナンスされ続ける数年間は。

MetaEdit+のリポジトリには、DSM ソリューションを共有するためのポリシーが備わっていて、自動的に言語の新しいバージョンで、全てのユーザーのモデルを更新することができる。あるいはDSMの定義を個々のファイル（またはファイル群）で共有可能にして、各モデリング担当者が個々のモデリング環境にインポートすることもできる。これは新しいバージョンがデプロイされたときに、メタモデルの古いバージョンで作成されたモデルインスタンスの情報が失われないようにするために、極めて重要な機能。

## 1. 概要

MetaEdit+の特徴のひとつに、簡単にモデリング環境を更新できることがあります。（一から独自でモデリング環境を作っている場合は多少の変更を加えるにも多大な時間とコストがかかってしまいますが）メタモデルが変更されるとモデルは自動的にアップデートされ（ドメインが固定される必要が無く、モデリング環境を自在に拡張できる）、また変更されたモデル部品は簡単に全てのモデルに反映させることができます。しかも以下で紹介する仕組みを介してメタモデルとモデルの両方を簡単かつ安全に更新・変更することができます。

## 2. メタモデルを変更した場合

MetaEdit+は、メタモデルの変更により作成済みのモデルが使えなくならないように設計されています。

基本的に、メタモデル部品の追加を行っても既存のモデルに対する影響はありません。また、メタモデルからメタモデル部品を削除した場合やルールやコンストレインツの変更がされた場合でもモデル上の実体が勝手に削除されないようになっており、それらは自由に参照や編集が可能になっています。

しかし、変更があった後では既存のモデルは修正される必要が有るでしょう。

プロパティに関しては、追加・削除に関わらず、全ての変更が即座に反映されます。この場合も注意が必要です。

その為、変更後のメタモデル環境に従ってモデルが修正されるような仕組みが必要です。

## 3. メタモデルの変更をモデリング担当者に伝える方法（自動的に漏れがなく）

メタモデル環境に変更が有った場合、その規則に従うようにモデルは修正されるべきです。従って、メタモデル環境構築担当者は、メタモデルを変更したときに、新しいメタモデルのルールとそれに従って既存のモデルを修正する必要が有ることを、全ての開発者に通知する必要があります。また、既存のモデルが新しいメタモデルで表現できるように考慮して、メタモデルを変更する必要があります。例えば削除されたモデル部品の

実体を置き換え可能なモデル部品が、新しいメタモデル環境の中に含まれている必要があります。

メタモデルの変更後、各開発者はモデルをチェックし修正することになるわけですが、これを手作業で行うことは効率的にも確実性の面でも問題があるでしょう。この作業を自動的に漏れが無いように行える2つの方法を示します。

1つ目の方法は、修正や削除されたメタモデル部品のシンボルを、削除されたことが分かるように変更することです。(例えば、大きなエクスクラメーションマークをつける等)

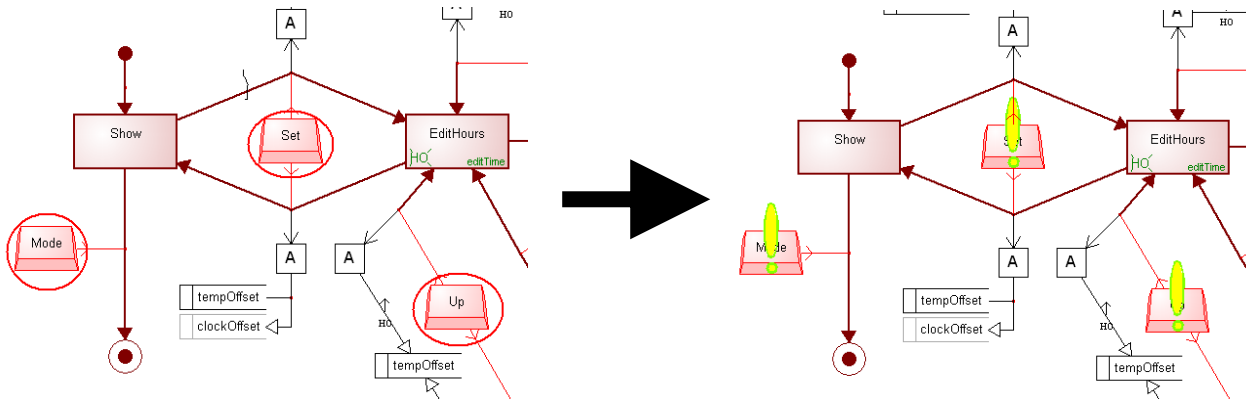


図) 左側の赤い丸で囲まれたオブジェクトが削除された場合の表示例

メタモデル部品を削除しても、その実体はデータベースには残ります。これらに対してグラフツールのようなモデリング環境からアクセスすることは出来なくなりますが、オブジェクトツールやタイプブラウザーのようなメタモデル構築環境からはアクセス可能で、シンボルの編集も行えます。そのためメタモデル環境で削除するメタモデル部品のシンボルを修正しておくことで、各開発者は変更が必要な箇所を判別できるようになります。

さらに、"\_Obsolete Types" など削除専用のグラフタイプを追加することで、不要となるメタモデル部品を登録することでモデリングとモデル環境構築のリポジトリ(データベース)が異なっているような場合でも、確実に変更を反映させることが出来るでしょう。

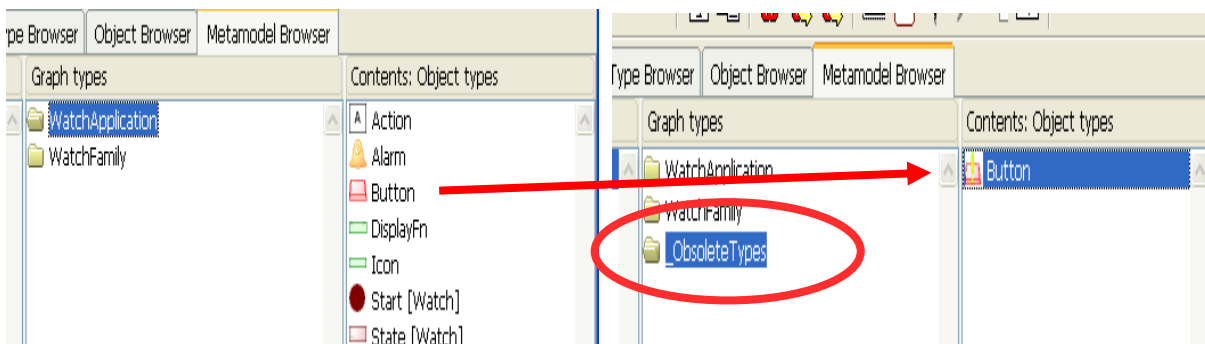


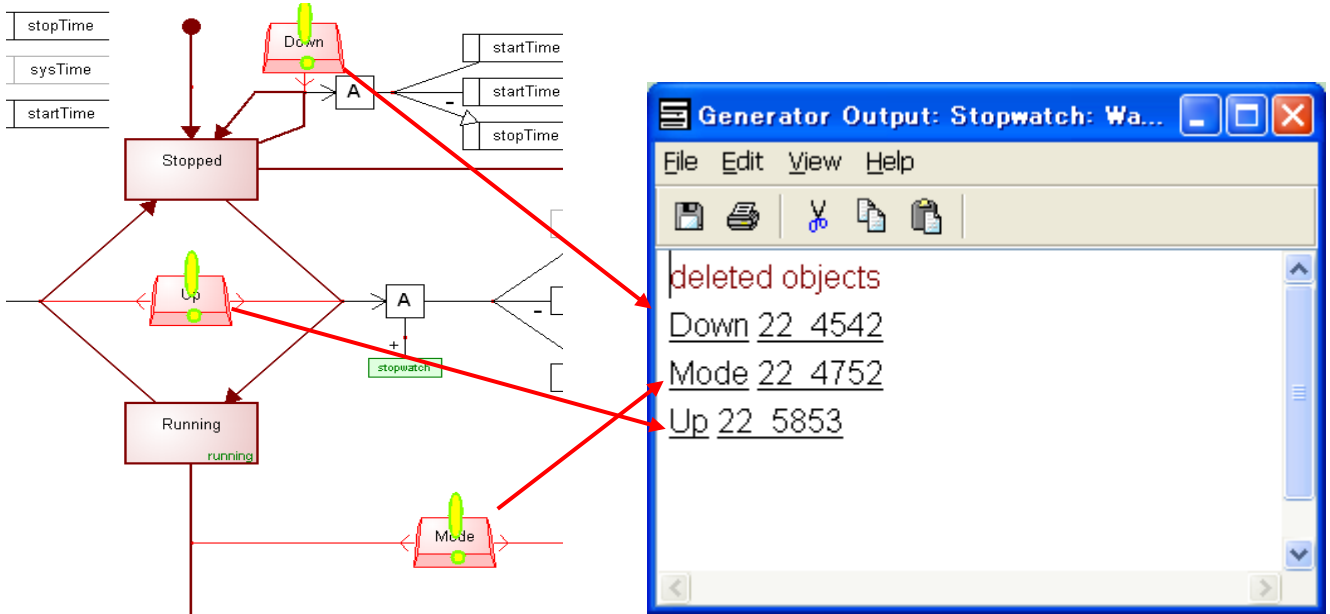
図) 削除したいオブジェクトを別のグラフタイプに移動させる

メタモデルとこのグラフタイプと一緒に配布する事で、各開発者はアップデートされたシンボルを確実に読み込むことになる為、修正忘れがなくなるでしょう。

※ モデリング環境に新たなグラフタイプが増えることとなりますが、グラフタイプの名前が"\_"から始まって

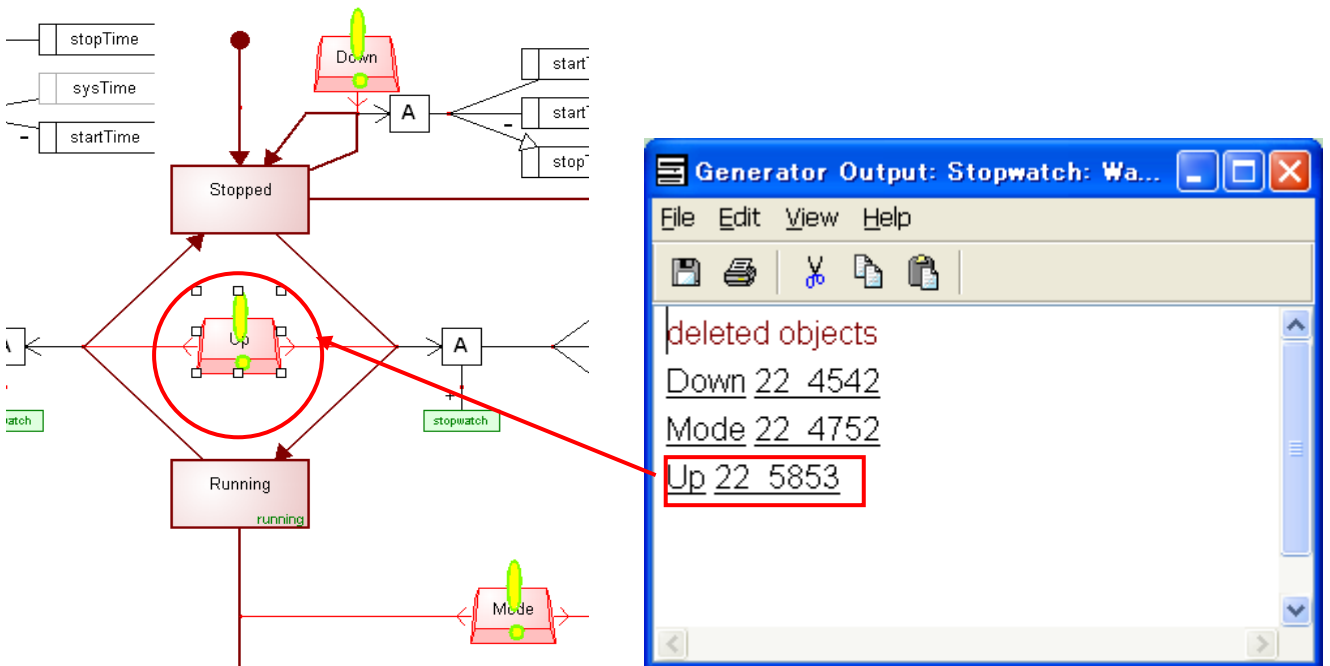
いる事で抽象タイプとして扱われる為に、不用意にグラフが生成されることはありません。  
(\_ObsoleteTypes でグラフを作成しようとするとエラーが発生します)

2つ目の方法は、修正や削除されたメタモデル部品をリストアップするジェネレータを作成することです。  
各開発者には、作成済みの各モデルに対してジェネレータを実行してもらい、リストアップされたモデル部品  
を修正してもらいます。



削除されたオブジェクトをジェネレータでリストアップした例

ジェネレータの実行結果には、モデル部品へのリンクが張られており、それをクリックする事でモデル上の  
モデル部品を表示することが可能です。各開発者は表示されたモデル部品をそのまま修正することが出来ます。



ジェネレータ上のリストをダブルクリックすると対象オブジェクトがハイライトされる

#### 4. 不要な部品をモデル環境から完全に削除する方法

最終的にメタモデル環境構築担当者は、タイプマネージャーで"Delete Unselected Types"の機能を使用する事で、メタモデル部品をデータベースから完全に削除できます。この場合、同じデータベース内のモデルでそのメタモデル部品が使用されていれば、これを削除することが出来ません。これによって、メタモデル環境構築担当者は削除したいメタモデル部品が使用されているか否かを確認することができます。

以上