

## MARTE : 新しい組込み UML 標準

—アーキテクチャ先導型コデザイン・パラダイムへ



オブジェクトテクノロジー研究所  
鎌田 博樹

1. DSML : ドメイン特化モデリング言語の世界
  2. 組込みモデリング標準 : SPT から MARTE へ
  3. 大規模国家プロジェクトによる研究開発
  4. MARTE の概要
  5. MARTE の構成要素
  6. MARTE をサポートするツール
- むすび

### はじめに

OMG は、9 月下旬に開催したフロリダ州ジャクソンビルにおける TC で、UML Profile for Modeling and Analysis of Real-time & Embedded Systems (MARTE) という新しい仕様を採択した。CORBA 以来の 5 文字略語で、なかなか覚えられそうもない名前だが、要するに、「組込み UML」の標準 (拡張プロファイル) である。UML の組込み向けプロファイルとしては、Schedulability, Performance, and Time (SPT) や Quality of Service and Fault Tolerance (QoS & FT) があり、なぜ新しく MARTE が必要だったかを理解するのは簡単ではない。

筆者も分かっていなかった。そこで、今回正式採択されたのを機会に、周辺情報を含めて勉強することにしたのだが、この異例の新標準が、急速に発展してきた組込みシステム分野の特異性を反映したもので、そのキープレーヤー、関連する研究開発/標準化のプロジェクトの動向と合わせて考えてみると、非常に興味深いものであることが理解できた。仕様書そのものは 600 ページを超えるので、関心のある方は (必要に応じて) お読みいただくとして、ここでは MARTE をめぐる組込み MDE の動向を中心にレポートしてみたい。

### 1. DSML : ドメイン特化モデリング言語の世界

ソフトウェアはじつに問題の多い技術分野だが、最大の魅力は、機械や電気から人間まで、あらゆるものを情報として扱うことが出来る自在性と、可能性であろう。中でも、設計 (モデル) からそのまま完全な実装をつくり

出すことが出来るという性質が最も重要で、そんなことができる工学は他にはない。しかし、魔法のようなことを実現するには、魔法ではなく、モデルの正確性と詳細性を保証する人間の努力の積み重ねしかない。

モデルと実装の距離は、着実に縮まってきた。それによってモデルの対象も大きくなっている。製造業の世界では、組込みソフトウェアから半導体、ついには自動車や航空機、ロボットなどのシステム全体もモデルの対象に含まれるようになってきている。ソフトウェアと非ソフトウェアモデルとのインタフェースも出来てきた。もはや、モデルのパワーを疑う余地はない。一足飛びには進まないが、前進していることが重要である。そしてこの段階では、先発と後発の差が生じやすい。ツールを含めた環境が整備され、実証済みのモデルが出回るよう

### PROMARTE チームのメンバー (\* OMG MARTE 仕様の提案企業)

#### ツールベンダー

ARTISAN Software Tools \*  
International Business machines (IBM) \*  
Mentor Graphics Corp.  
Softeam \*  
Telelogic AB (I-Logix\*)  
Tri Pacific Software  
No Magic  
The Mathworks

#### 産業ユーザー

Alcatel \*  
France Telecom  
Lockheed Martin \*  
Thales \*

#### 大学・公的研究機関

Carleton University  
Commissariat a l'Energie Atomique (CEA)  
ESEO  
ENSIETA  
INRIA  
NSA from Lyon  
Software Engineering Institute (SEI/CMU)  
Universidad de Cantabria

になると、後発でもキャッチアップは容易になるだろう。先発企業が初期の優位に驕っていればの話だ。

モデルが十分に精確であるかどうかは「言語」に依存する。しかし、システムの様々な側面を捉えて表現する言語は一つや二つでは済まない。言語化の考え方、用途、環境などによって異なった言語が必要になるところは、実装言語と変わりはない。モデル駆動では、複数の言語が絡んでくるのが少なくないし、システムが複雑になれば不可避でもある。

OMGのような標準化団体でさえ、モデリング言語が1種類でよいということはない。大きく汎用モデリング言語(GPML)とドメイン特化モデリング言語(DSML)に分かれる。ともに複数あり、これからも増えるだろう。同じくUMLの拡張プロファイルだが、SysMLは前者、MARTEは後者に属する。SysMLの例のように、UMLメタモデルを使った拡張はうまく機能するので、「新しい」言語でもUMLとの一貫性は保たれ、何よりもUMLツールを使う利点がある。

DSMLは、「特定のアプリケーションドメインに特化したモデリング言語」である。汎用の言語だけを使ってメタモデルを扱っていくと、プロジェクト、ドメイン、開発組織、業界などによって様々なメタモデルが乱立し、モデルベースの大規模開発やインテグレーションなどすぐに不可能となる。オブジェクト指向が失敗しやすい最大の原因がこのメタモデルである。だからメタモデルの範囲を限定するのに越したことはない。

DSMLの設計は難しい。ドメイン(のシステム化)についての知識だけでなく、言語設計についての(人並み外れた)知識が必要となるからである。力があれば造れないことはないが、メンテナンスも考えれば、少なくとも業界で標準にすることを考えたほうがいいし、業界の標準としても、他の業界、分野(例えば電子商取引や会計モデル)との整合性を考えないといけない。

DSMLでは、ドメイン特有のコンセプトを言語構成物として使用する。「コンポネント」や「ノード」「クラス」といった汎用的概念ではなく、「メモリ」や「タスク」といった用語を使用する。

OMGにおいてDSMLを開発する方法は基本的に2つあり、UMLではなくUMLのメタモデルであるMOFを使って新しい言語を開発する方法(重量級の拡張)とUMLメタモデルを使う方法(軽量級の拡張)の2種類がある。

自由度が大きいのは、もちろん前者だが成功へのハードルはより大きくなる。実装環境とのシームレスな連携など、確実なメリットがあり問題の回避が可能でない限り、手を出すことはないだろう。モデリング言語が成功するには、表現力と簡潔性、他の技術との相互運用性(インタフェース)、ツールのサポートなど多くの条件が必要だが、MARTEはUMLメタモデルをそのまま利用しているので多くの問題を回避できている。

## 2. 組込みモデリング標準：SPTからMARTEへ

UMLが標準として採択されて、今年で10年。実用的な組込みシステムの設計現場に普及を始めてからも優に5年以上は経過しており、実時間性と資源制約性が厳しく要求されるこのシステム分野において、UMLの表現力と応用力の限界を探る多くの経験が重ねられてきた。

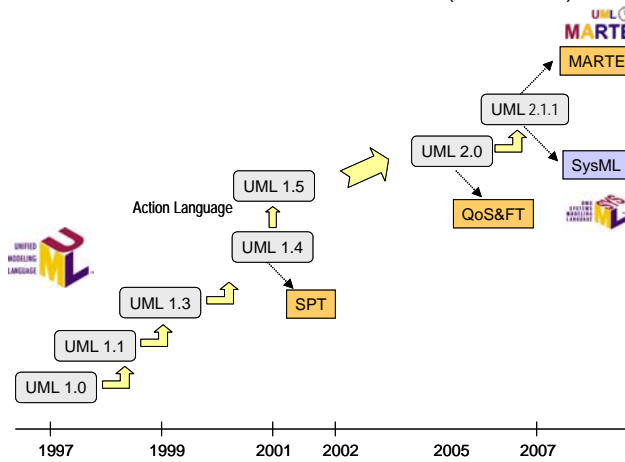
組込みシステムのモデリングでは、定量的性質の表現が鍵を握る。実装に依存しないエンジニアリングモデルの表現では、Quality of Service (QoS)というコンセプトを使う。QoSとは機能的性能(サービス)がどのように提供されるかを仕様化するもので、スループットや容量、応答時間、デッドラインなど、通常は定量的なものとなる。UMLじたいはQoSプロパティの表現力がない。それでは設計したものが分析に使えない。

フランスCEA/LIST(原子力委員会付属ソフトウェア研究機関)のフランソワ・テリエによれば、UML1.xの限界とは、第一に時間と資源に関する量的概念が表現できないこと。第二に意味定義の厳密性が弱いことも、利用範囲を狭くする。第三に、タスクやセマフォのような、リアルタイムOSレベルに関連した構成物(artifacts)を使ってモデルを構築するのに特別な構成物を必要とする。

これらをもって、RTESにおけるUMLの「致命的な欠陥」を言いたてる議論もあったが、UMLの中にこうした問題を解決するためのメカニズム(とくに拡張機構)が用意されており、それらを使うことで、基本的なモデリング概念を追加したり、あるいは別な言語を考えだしたりする必要もないことが認められた。

OMGにおけるRTES対応の努力は、したがってUMLの拡張メカニズムを使って、RTESのモデリングに必要なコンセプトを表現し、そしてスケジューラビリティおよびパフォーマンスに関するプロパティを分析する機能を拡張することが目標とされた。最初の成果といえるのは

図1：UMLの展開と組込み分野への拡張(1997-2007)



SPT v.1.1(2005)と QoS & FT である。

SPT は、UML 1.4 を拡張したもので、リソースと非機能的プロパティ(Non Functional Properties , NFP)のモデリングのための総称的フレームワークを導入した。時間や同時性の表現に関しては十分な機能を提供したと言われている。分析に関しては、スケジューラビリティ分析では主に RMS (Rate- Monotonic Analysis)をサポート、パフォーマンス分析では待ち行列関連の技法をサポートしている。

しかし、結果的に SPT は急速に発展する RTES のニーズに十分応えるものではなかった。SPTの限界に関して、サブプロファイル間の一貫性の欠如や意味の曖昧さ、状態機械ベースの分析ができないなど、当初から多くの指摘があり、それは FTF (Finalization Task Force)のレポートでも指摘されている。何よりも問題だったのは、簡単ではあったが構造的に柔軟性を欠いていたことで、柔軟性はあっても分析に必要な基本的コンセプトを定義していない QoS & FT との組合せでは、使いにくかったということであろう。柔軟性と簡明性は両立が難しいが、よりバランスのとれた仕様が求められた。

SPT の枠組みの上に機能を追加する“2.0”ではなく、新しい枠組みを定義する新規の RFP (Request for Proposals) を発行することが決まったのはそのためである (OMG の数多い標準の中でも、スコープ自体を変更した例は、非常に少ない)。

### 3. 大規模国家プロジェクトによる研究開発

組込みモデリングのための新しい標準の開発の最前線となったのは、航空・宇宙・自動車など、高度な安全性や、生産や保守・更新との連携が求められ、複合的な

リアルタイム/組込みシステム (習慣に従って、以下これを RTES とする) を扱う産業である。

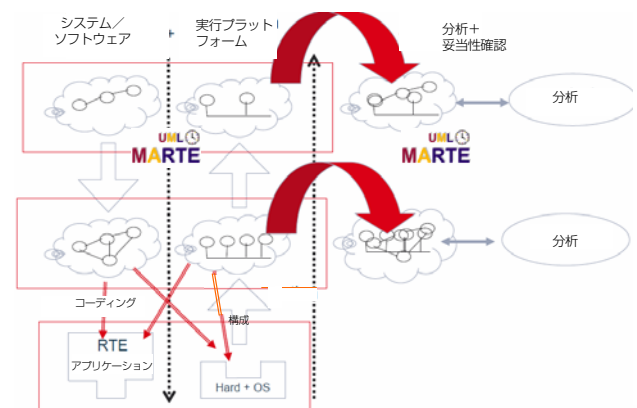
産業技術の根幹をなすだけに、EU 政府ともいえる欧州委員会(EC)や米国の国防総省、NASA では、戦略的観点から毎年 1,000 億円単位の資金援助を行ってきた。米国と EU は、航空機産業の覇権をめぐる激しい競争を展開しており、彼らは「組込みソフトウェア」というよりは、21 世紀型製造業 (ものづくり!) のテクノロジー・バリューチェーンを構想し、その基盤をなすモデル駆動エンジニアリング(MDE)に戦略的投資を行ってきたのである。

その結果、ハードウェアを含めた組込みシステムのモデリングのための UML の拡張、MDE を構成する他の技術群との連携、高度なツールによる自動化の支援、といった方向性が明確となった。

欧米における組込み MDE 関連のプロジェクトは、コンポーネントベース(CBSE)やハードウェア/ソフトウェア・コデザイン(HW&SW)を実現するための、様々なレベルのプロジェクトとサブプロジェクトから成っている。OMG の UML Profile for QoS & FT (2005)や SysML (2006)、SAE の AADL (2006-draft)、Autosar (2006)といった標準は、そうした「産官学プロジェクト」の成果である。そのあたりの関係については (複雑かつ重要なので) 別の機会に譲りたいが、ともかく、MARTE はそうした新しい MDE パラダイムを支えるべきものとしてスタートした。例えば、フランスでは、CARROLL の Protes というプロジェクトが中心となって推進されている。

CARROLL は、タレス社、CEA、INRIA による MDE のための共同イニシアティブで、分散、リアルタイム、オンボード、モバイルといった性質を持つ複雑なシステム

図2：MARTE が目指す開発環境



を対象としている。これまで、MOTORS（モデル変換）、ICE（RTE ミドルウェア）、Mutation（テスト生成）などのプロジェクトを立上げているが、MARTE を担当させるためにさらに Protes というサブプロジェクトを発足させた。参加したのは、CEA-List、INRIA（とくに DaRT、ESPRESSO、AOSTE）、タレス（TRT、DAE、TCF）などである。

CEA-List では、Accord および MAST (Modeling and Analysis Suit for Real-Time Applications) プロジェクトを通じて SPT の言語的拡張を実現し、必要とされるモデリング機能とオープンなツールによるサポートを開発・実証していった。

標準化のために、特に PROMARTE というコンソーシアムが編成された。これが標準仕様の作成・提案の中心である。PROMARTE のメンバーは、ユーザー企業、ツールベンダー、大学・研究機関の 3 者で構成されている（メンバーは別表の通り）。

読者は、これまでの話の中で、ツールベンダーの動きに触れていないことに不審をもたれるかもしれない。ベンダーが参加していないのではなく、脇役であるためである。もちろん MDE においてツールは不可欠な要素であり、重視されてもいるのだが、組込みシステムの世界では「ものづくり」のニーズが主導で動いており、ベンダーが前面に出る意味がない。反対に、エアバスやボーイング、ダイムラーや PCG（とそれぞれを支援する政府）にとっては、MDE は戦略的技術分野であるがゆえに最先端に立とうとするわけである。それに、プロファイルレベルでの拡張にモデリングツールが対応するのは、あっけないほど簡単であり、SysML や MARTE にしても、無償ダウンロードで済んでしまう。現在のツール化の焦点は、組込みソフトウェア自体を扱うものではなく、ハードウェアを含めたアーキテクチャ設計や安全管理、ライフサイクル管理などの連携に関わるものである。

**4. MARTE の概要**

MARTE に求められた新たな要件は、ほぼ以下のように要約できる。

1. ソフトウェアだけでなくハードウェアの側面もモデル化できること
2. MDA アプローチをサポートするため、プラットフォームおよびプラットフォーム非依存、プラットフォーム依存のモデルを表現できること

3. UML for QoS & FT に準拠すること
4. リアルタイム的制約だけでなく、消費電力や記憶容量など、他の組込み QoS 特性も記述できること
5. 組込み、リアクティブ、制御/コマンド、および集約的なデータフロー計算システムのモデル化
6. コンポーネントベースのアーキテクチャのモデリングと分析
7. Asynchronous/Causal, Synchronous/Clocked および Real/Continuous の各モードのモデリング

MARTE の RFP は 2005 年 2 月に発行された。SPT が正式化された時には、別の標準化をスタートさせていたことになる。2005 年 11 月には第一次提案がなされ、これをもとに最終案に向けての作業が進められ、今回の採択となったものである。その間に欧州を中心に、産業レベルやアカデミックレベルで何度もワークショップやセミナーが開催され、MARTE への意見の集約と合意形成が図られていった。こうした進め方は欧州の得意とするところである。

**MDA における位置**

MARTE はモデル駆動開発(サイクル)における構成物のユーザーとプロバイダーのために開発された。つまり、モデルの設計者と分析者、そして方法論開発者と実行プラットフォームのプロバイダーである。

このプロファイルを使うことにより、以下が達成されることが期待されている。

- RTES におけるハードウェアとソフトウェアの 2 つの側面を扱う共通の方法を提供し、2 つの開発者の間のコミュニケーションを改善する
- 仕様化、設計、検証、コード生成等に使われるツール間の相互運用を可能にする。
- リアルタイム/組込みシステムの機能を、ハード/ソフト両面の特性を考慮しつつ定量的に予測するために使われるモデルの構築を促進する

図 3: UML 仕様との関係

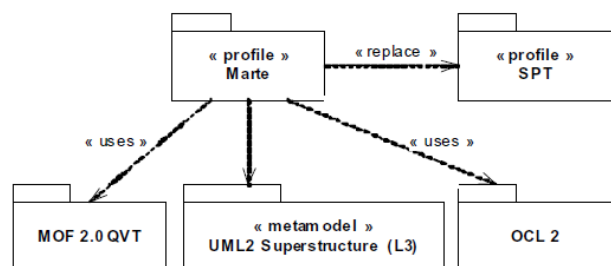
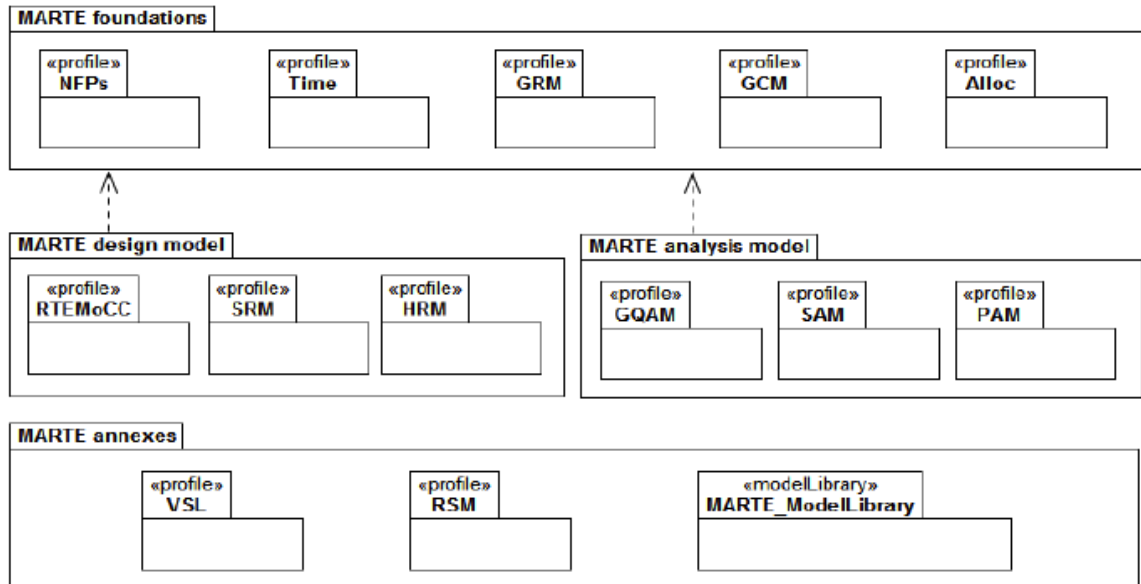


図 4: MARTE の全体構成



MARTE は「組込みソフトウェア」だけではなく、ハードウェアを含めた「組込みシステム」のモデル駆動開発を対象としている。両方の仕様と設計を扱うことが重要な点である。RTE 分析は、とりあえずスケジューラビリティとパフォーマンスを扱う。OMG 標準体系との関係では、UML 2.0、SysML、QoS&FT をサポートする。UML2 スーパーストラクチャのメタモデルをプロファイルし、既存の SPT プロファイルに置き換える。OCL2 と MOF2 QVT を使用する。

OMG の RTE 関連標準としては、いずれも UML のプロファイルとして、QoS&FT、SoC (Software on Chip)、SE (SysML)があり、また進行中の Executable UML Foundation、AADL プロファイルがある。QoS&FT は、MARTE の NFP (Non Functional Properties)パッケージとして扱う。SoC は MARTE よりも限定された領域を対象としている。またミドルウェアとして、MARTE のターゲット・ミドルウェアである RT-CORBA がある。RT-CORBA ベースのアーキテクチャは、MARTE で正確に記述できる。SysML との関係では、アロケーション (割当) の概念を特殊化するとともに、フロー関連の概念についてはそのまま再利用している。MARTE と SysML のチームはそれぞれメンバーも重複しており、今後も密な連携が図られることになろう。

MARTE のアーキテクチャ

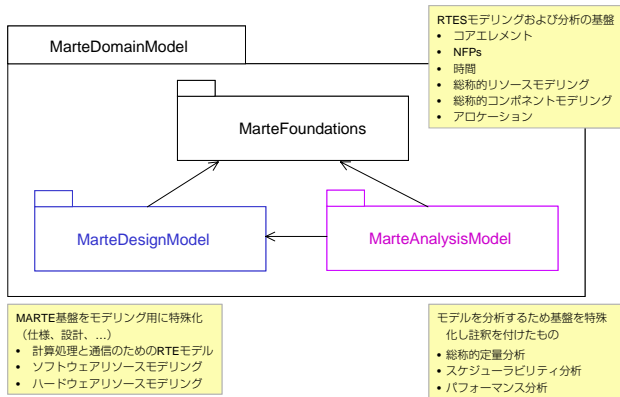
プロファイルは2つの主要な対象、

- 1) RTES の機能のモデリング
- 2) システムプロパティの分析をサポートする、アプリケーションモデルへの注釈表記

を中心に構成されている。デザインモデル・パッケージと分析モデル・パッケージがそれぞれ対応する。どちらにも関係する時間と同時的な資源の使用に関するものは、ファウンデーション・パッケージに収められている。4 番目のパッケージには、MARTE で定義された補助的なプロファイルや、モデラーが RTE アプリケーションを表現する際に使うことができる、定義済みモデルライブラリが含まれている。図は MARTE のハイレベル・アーキテクチャを示したものである。

- NFPs = Non-Functional Properties : 非機能的プロパティ
- GRM = Generic Resource Modeling : 総称的リソースモデリング
- GCM = Generic Component Model : 総称的コンポーネントモデル
- Alloc = Allocation modeling : 配分モデル
- RTEMoCC = RTE Model of Computation & Communication : RTE 計算・通信モデル
- SRM = Software Resource Modeling : ソフトウェアリソース・モデリング

図 5 : MARTE 主要部分の内容



- HRM = Hardware Resource Modeling : ハードウェアリソース・モデリング
- GQAM = Generic Quantitative Analysis Modeling : 総称的定量分析モデリング
- SAM = Schedulability Analysis Modeling : スケジューラビリティ分析モデリング
- PAM = Performance Analysis Modeling : パフォーマンス分析モデリング
- VSL = Value Specification Language : 値仕様化言語
- RSM = Repetitive Structure Modelling : 反復的構造モデリング

5. MARTE の構成要素

以下、主要なモデル要素についてみていこう。

ファウンデーション

タイムモデルは、UML2 の SimpleTime パッケージの機能を発展させたものである。時間に関連するすべての事物は明示的にクロックを参照する。時間は物理的時間に限定されない。また、配分およびクロック不確定性をサポートする。MARTE のタイムモデルは、時間に関連した 3 種類のコンセプトと意味をカバーしている。すなわち、

図 6 : 基盤モデル

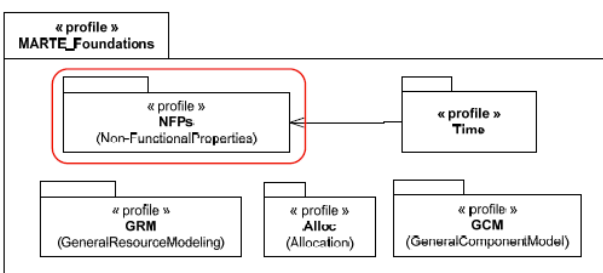
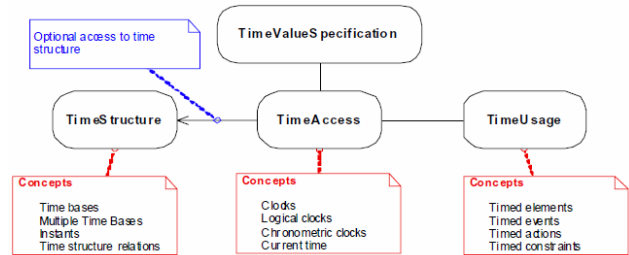


図 7 : タイムモデルが対象とするコンセプト



因果/時相タイムモード：先行性、依存性に関係したもの

クロック/同期タイムモデル：同時性、時分割

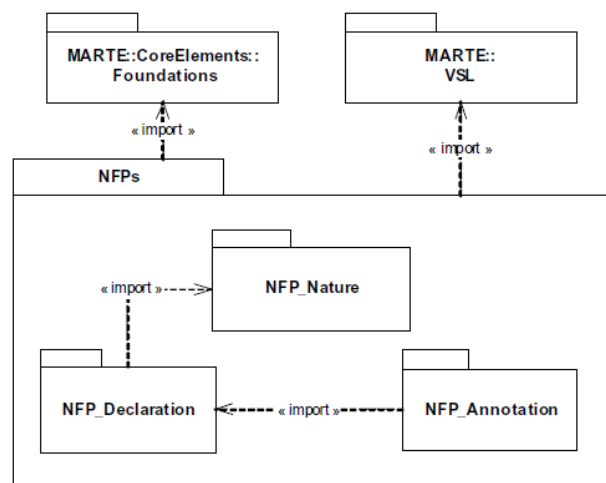
物理的/リアルタイム・タイムモデル：継続時間を正確にモデル化

時間は、いくつかのクロック(clocks)から構成される。クロックは、順序づけられた瞬間(instants)の集合で、単位(units)で持続時間を指定する。

NFP モデルは、非機能的プロパティを対象とする。MARTE は、SPT および QoS&FT にあったアイデアのいくつかを形式化している。SPT の Tag Value Language および時間に関連した値、QoS&FT のプロパティ識別子である。

MARTE は、NFP のために新たに Value Specification Language (VSL)という言語を導入した。これは 1) NFP の宣言、2) 値の仕様記述、3) NFP を使った注釈メカニズムの 3つの部分から構成される。まず NFP を定義、値、測定単位、統計的性質などを明確化する。次に値の仕様(シンタックス)と表記法を定義し、最後にタグ付値、制約などを使った注釈メカニズムを示すものである。

図 8 : NFP モデリングパッケージの構造



NFP/VSL はかなり重要な拡張であり、UML や SysML においても有用であると考えられている。

MARTE の目的の一つは、コンポーネントベース・アプローチをサポートすることであったが、新しいコンセプトを導入するのではなく、UML2 や SysML、Spirit、AADL、Lightweight-CORBA、EAST-ADL2、Autosar など既存のコンポーネントモデルを扱えるようにすることにフォーカスしている。主として UML の構造化クラスに依拠し、その上に SysML のブロックを追加している (atomic および non-atomic フローポート)。

総称的なリソース(Resource)のモデリングに関しては、定義・使用にNFPを持つことが出来る Service を提供する。Resource のタイプは、Storage、Synchronization、Concurrency、Communication、Timing、Computing、DeviceResources など、かなり豊富に提供されている。資源の共有やスケジューリング戦略、特別な使用 (メモリ消費、計算時間、エネルギーなど) を指定することが出来る。

ソフトウェア・リソースのモデリングは、RTE 用の新しい API 標準ではなく、既存の RTE ソフトウェア実行 API に統一的な手段を提供したものである。ハードウェア・リソースのモデリングでは、論理的ビュー (機能モデリング) と物理的ビューを定義する。

例：

Logical: HwStorage, HwCommunication, HwTiming...

Physical: HwLayout, HwPower...

アロケーション (配分) は、アプリケーション要素を実行資源に割当てるもの。SysML のアロケーションと整合している。NFP の制約を課すことが出来る。

図 9：総称的リソースモデリング

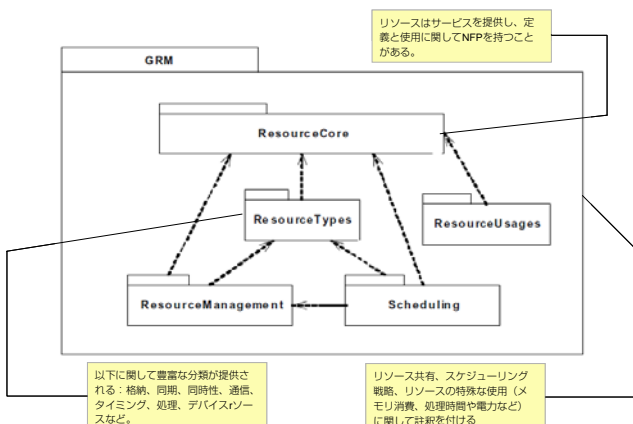
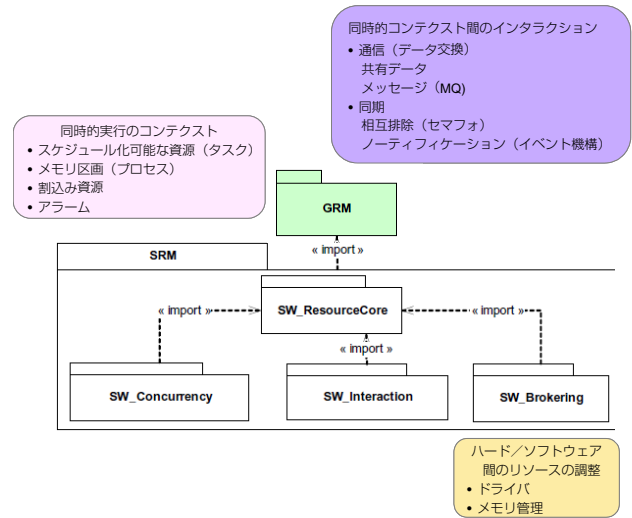


図 10：ソフトウェアリソースモデル



デザインモデル

計算処理および通信のリアルタイムモデルは、リアルタイム機能の定性的側面をモデル化するハイレベルのコンセプトを提供するものとして、UML2 の Active Objects を汎化した Real-Time Unit (RTUnit)、Passive Objects を汎化した Protected Passive Unit (PPUnit)があり、定量的側面をモデル化するものとして、提供された振舞いに結びついた Message Queue のサイズとポリシーを記述する Real-Time Behavior (RtBehavior)、UML の Action、Message、Signalなどを拡張した Real-Time Feature (RTF)、UML の Connector を拡張した Real-Time Connector (RteConnector)がある。

分析モデル：GQAM、SAM、PAM

分析については、総称的定量分析モデル (General Quantitative Analysis Model, GQAM)、スケジューリング分析モデル (Scheduling Analysis Model, SAM)、パフォーマンス分析モデル (Performance Analysis Model, PAM)の3つがある。

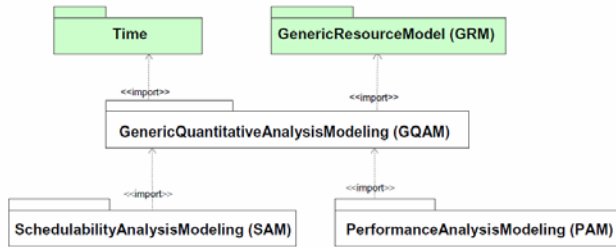
GQAM は UML2 をベースに、SPT を修正、拡張している。とくにタイミングに関しては、オーバーヘッド、応答時間、タイミング要件を導入した。図 xx は、GQAM の依存性とアーキテクチャを示している。また、モデルベース分析の処理スキーマを説明したものが図 xx である。GQAM：分析サブプロファイルで共通に使用されるコンセプト

SAM：スケジューラビリティ分析技法をサポート

PAM：パフォーマンス分析技法をサポート

SAM は、ハイレベルな分析構成物を提供する (感度分

図 11 : GQAM のアーキテクチャと依存性



析、パラメータ分析、時間制約や時間予測の監視など)。RMA などほとんどのスケジューラビリティ分析技法をサポートする。

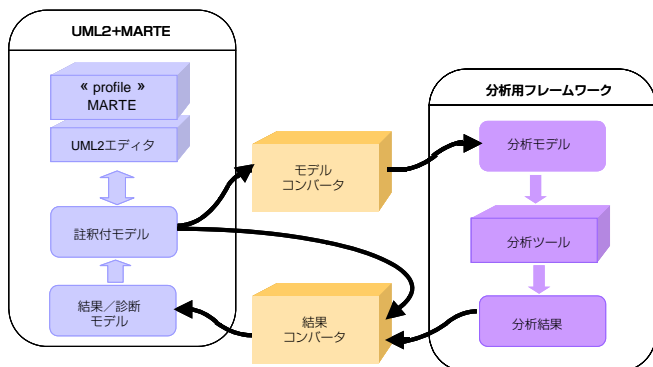
PAM は、ワークロード、振舞い、リソースに分けられるが、一部の GQAM ステレオタイプを特殊化し、あるいは再利用している。キューング・ネットワーク、ペトリネットなど、ほとんどのパフォーマンス分析技法をサポートする。UML+MARTE モデルは必ず、構造のビュー（アーキテクチャと配置）と振舞いのビュー（主要なパフォーマンスシナリオ）を含む。

6. MARTE をサポートするツール

MARTE は、既存の UML モデリングツールにプロファイルを追加することで使用できる。OMG の MARTE 情報ページは、ツールに関する情報も載せているが、これまでに、Rational Software Architect 7.0 向けのアドイン・プロファイルとして MARTE profile for RSA 7 (v.1.0.0)が、タレス社から提供されているほか、オープンソース Papyrus 4 UML 向けの Papyrus for MARTE もある。ほどなく、ARTiSAN、Softteam、Telelogic-I-Logix、No Magic などの組込みモデリングツールからも提供されることになる。

<http://www.papyrusuml.org/>

図 12 : モデルベース分析の処理の流れ



むすび

蛇足だが、Marte (マルテ) はスペイン語とイタリア語で火星を意味する。5 文字略語をあえて使ったのは、太陽系の 4 番目の惑星という意味があったからだろうか。

MARTE は高度な組込みシステム開発のニーズから生まれた言語であり、標準的なコンセプトと表記法を提供する。しかし、UML 同様、分析やモデル化の方法については規定していないから、ユーザーは方法論とツールを決めてから使う必要がある。

航空機産業では、MARTE と AADL (Architecture Analysis & Design Language)を併用する試みが活発に行われている。AADL は自動車・航空機産業の国際的団体である SAE (Society of Automotive Engineers)の標準であるが、モデル駆動エンジニアリング (SAE やカーネギーメロン大学 SEI ではモデルベース・エンジニアリング=MBE と称している)の有力な標準として普及しつつある。AADL も組込み/リアルタイム・システムのハードウェア、ソフトウェア・アーキテクチャのモデリングに使われる。AADL のアーキテクチャ・モデルは、UML/MARTE に変換されて設計ドキュメント、分析、コード生成などに使われる。

AADL は、AADL Meta Model & XMI、AADL UML Profile など OMG 標準と連携を持っており、航空機、航空電子、自動車、鉄道、ロボット、軍事システムなど、様々な分野で多くのプロジェクトが進行している。MARTE はそうした国際的な動きの結果であると同時に、推進力ともなると期待されている。(了)

参考 :

OMG の公式情報は以下を参照

<http://www.omgmarTE.org>

A UML Profile for MARTE, Beta 1, August 2007 OMG Adopted Specification, OMG Document #: ptc/07-08-04

MARTE Tutorial: An OMG UML profile to develop Real-Time and Embedded systems, Sébastien Demathieu, 13th SDL Forum, Sep 2007, Paris

MARTE: A new OMG standard for Modeling and Analysis Real-Time and Embedded systems, Laurent Rioux, FDL '07 Forum on Specification & Design Languages

MARTE: An OMG profile for Modeling and Analysis Real-Time and Embedded systems, Madeleine Faugere, Ph.D., SAE Meeting-Anchorage, October 2007

A General Structure for the Analysis Framework of the UML MARTE Profile, Huáscar Espinoza<sup>1</sup>, Hubert Dubois, Julio Medinab, and Sébastien Gérard

Annotating UML Models with Non-Functional Properties for Quantitative Analysis, by Huascar Espinoza, Hubert Dubois, Sébastien Gérard, Julio Medina, et. al.

MARTE: A New Standard for Modeling and Analysis of Real-Time and Embedded Systems, by Dr. Sébastien Gérard, Dr. Julio Medina, Pr. Dorina Petriu

A Co-modeling Methodology for the Integration, of Real-time Architecture Models, by Isabelle Perseil, Laurent Pautet, ICECCS07- Auckland, NZ Workshop «UML&AADL'2007 »

Languages and Tools for Hybrid Systems, Design, by Luca P. Carloni, Roberto Passerone, Alessandro Pinto<sup>3</sup>and Alberto L. Sangiovanni-Vincentelli

Domain-Specific Modeling Languages and UML Profiles – An Introduction, by Bran Selic, 2006 IBM Corp.

Managing Complexity of Automotive Electronics Using the EAST-ADL, UML & AADL Workshop, July 14, 2007, by P. Cuenot, DJ Chen, S. Gérard, H. Lönn, M.-O. Reiser, D. Servat, C.-J. Sjöstedt, R. Tavakoli, M. Törngren, M. Weber

The Modelling Approach in ATESSST –Overview, ATESSST Report 2007

System Modeling for Automotive Embedded Applications, by Henrik Lönn, Volvo Technology Corp. MDD for Distributed Embedded Real-Time Systems, September 2006

How ADLs concepts can help in adapting the CORBA Component Model to Real-Time Embedded Software Design, by Sylvain Robert, Ansgar Radermacher, Vincent Seignole, Sébastien Gérard, Virginie Watine, Stéphane Ménoret and François Terrier,

Workshop on Architecture Description Languages WADL'2004, Toulouse, France

EAST-ADL – An Architecture Description Language, by Vincent De Bruyne and Françoise Simonot, Workshop on Architecture Description Languages WADL'2004

SysML に関しては OMG 情報を参照

<http://www.omgsysml.org/>

日本語→

<http://www.otij.org/omginfo/technology/primer/sysml.html>

レポート『SysML とシステム工学支援環境の現状と動向』オブジェクトテクノロジー研究所、2007 年

<http://www.otij.org/research/index.html>