From a Specification Level PIM to a Design Level PIM in the Context of Software Radios

Juan Pablo Zamora
Francis Bordeleau

Collaborators:
Michel Barbeau
Jeff Smith
Outline

- **Context**
- Specification vs. Design
- Software Radio PIM Refinement
- UML to UML-RT Mapping Examples
- Conclusion
Context

Software Radio Definitions:

• “A software radio is a radio whose channel modulation waveforms are defined in software”

• “… radios that provide software control of a variety of modulation techniques, wide-band or narrow-band operation, communications security functions (such as hopping), and waveform requirements of current and evolving standards over a broad frequency range.”
Context

Software Radio Platform Independent Model:

- Developed by the Software Radio Special Interest Group (SWR DSIG) within the OMG.
- Based on The Software Communications Architecture (SCA)
  - Developed by the Modular Software-programmable Radio Consortium (MSRC) formed by Raytheon, ITT Industries, Rockwell Collins and BAE SYSTEMS
  - Funded by the Joint Tactical Radio System (JTRS) Joint Program Office (JPO).
- Main contributors includes MSRC members, Mercury Computer Systems and Carleton University
- Important constituent of the OMG D&C RFP submission
Model Driven Architecture

- Provides the infrastructure for PIM to PSM refinement.
- Sets the guidelines for model mapping and model refinement.
- Provides platform independence abstracting away platform specific issues and thus allowing for different platform specific implementations.
- Uses multiple views and levels of abstraction.
- Model Consistency.
UML-RT

- Removes ambiguity from a specification UML model
- Provides for executable models
- Increases architecture understandability
- Supported by Rational Rose RealTime
- Automated Code Generation with Rational Rose RealTime
Outline

- Context
- Specification vs. Design
- Software Radio PIM Refinement
- UML to UML-RT Mapping Examples
- Conclusion
Specification
 Goal: define main concepts to be understood and allow for different implementations.

Design
 Goal: be specific enough to allow for code generation.
Specification Elements and Relationships

SWR PIM
Core Framework
Control Overview
Class Diagram

Bi-directional Association
Aggregation
Generalization
Class

Core Framework
Base Application Overview
Class Diagram

Realization
Interfaces
Composition
Unidirectional Association
UML-RT Design Elements and Relationships

Core Framework Base Application Overview UML-RT Class Diagram

Core Framework Base Application Overview UML-RT Capsule Structure Diagram

- Protocol
- Composition
- Dependency
- Capsule
- Data Class
- Capsules
- Connector
- Ports
• Context
• Specification vs. Design
• **Software Radio PIM Refinement**
• UML to UML-RT Mapping Examples
• Conclusion
Model Driven Architecture
from Specification to Design level

PIM Refinement

PIM

Specification
UML-RT Design

PSM

UML-RT/PLATFORM Design

SW Radio Spec. UML
UML-RT Design

UML-RT-to-CORBA mapping

SW Radio UML/CORBA PSM
Outline

• Context
• Specification vs. Design
• Software Radio PIM Refinement
• UML to UML-RT Mapping Examples
• Conclusion
Realization of an Interface

Operations: single parameter, no return type and visibility (public only).

Mappings:
1. Active classes map to Capsules
2. Interfaces map to Protocol classes
3. Operation’s names in the Interface map to incoming signal’s names in the new protocol class.
4. Parameter’s type map to incoming signal’s type upon incoming signal creation from step 3.
5. The realize relationship map to a <<port>> composition relationship between the new capsule and the new protocol class.
Association between two classes (1)

Operations: single parameter, no return type and visibility (public, protected and private).

1. Active Classes map to Capsules.
2. Protected/private operations in the source class map to protected/private operations of the new capsules.
3. Association relationship map to a new protocol class and a composite relationship between the capsules and the protocol class to allow for port creation. The class for which the set of incoming signals were taken, map to the base role in the port definition. The second port plays the conjugated role in the port definition. The association also maps to a connector that binds the two capsules.
4. Operation names of public operations in the associated classes map to incoming/outgoing signal names in the new protocol class. Public operations of one class in the association map to the set of incoming signals, while public operations in the second class map to the set of outgoing signals in the protocol class definition.
5. Parameter’s type map to incoming/outgoing signal’s type upon signal creation from step 4.
1. Active Classes map to Capsules.
2. Association relationship map to a new protocol class and a composite relationship between the capsules and the protocol class to allow for port creation. The class for which the set of incoming signals were taken, map to the base role the port definition. The second port plays the conjugated role in the port definition. The association also maps to a connector that binds the two capsules.
3. Operations with two or more parameters map to a new data class.
4. Parameters names map to argument names. Parameters’ types map to argument types
5. Operation names of public operations in the associated classes map to incoming/outgoing signal names in the new protocol class.
6. Operations with two or more parameters (after data class creation in step 3) map to signal’s type
7. In operations with return type the operation name map to signal name in an opposite direction as used for step 5. The operation’s return type map to the signal’s type
Composition and Association between two classes

Operations: no parameters, no return type and visibility (public only).

UML Class Diagram

1. Active Classes map to Capsules.
2. Association relationship map to a new protocol class and a composite relationship between the capsules and the protocol class to allow for port creation. The class for which the set of incoming signals were taken, map to the base role in the port definition. The second port plays the conjugated role in the port definition. The association also maps to a connector that binds the two capsules.
3. Operation names of public operations in the associated classes map to incoming/outgoing signal names in the new protocol class.
4. Composite relationship between active classes map to composition relationship between capsules, and composition relationships from the container class to protocol classes used by the composite capsule to communicate with external components. The ports added are of the same port role (base/conjugate) as the port role defined for its composite capsule. Also the port is defined as relay port.
5. The association from step 2 and the composition from step 4 also map to connectors binding the container capsule’s port to the composite capsule’s port and the external capsule’s port.
Outline

- Context
- Specification vs. Design
- Software Radio PIM Refinement
- UML to UML-RT Mapping Examples
- Conclusion
Conclusions

- By applying UML to UML-RT Mappings we take our UML model from a Specification to a Design level (some refinement may be required).
- UML-RT Notation is specific enough to allow for code generation.
- If we “trust” the Mapping definitions and verify that they were followed in a PIM to PSM refinement, we can “trust” the resulting PSM.
- This work can set the basis for a Validation and Compliance Framework for PIM implementations.