Harmony/SE:
A SysML Based Systems Engineering Process

Hans-Peter Hoffmann, Ph.D
Chief Systems Methodologist
hoffmape@us.ibm.com
Model-Based Systems Engineering

- Integrated System / Embedded Software Development Process Harmony
- Fundamentals of Model-Based Systems Engineering (Harmony/SE)
  - Essential SysML Artifacts
  - Service Request-Driven Modeling Approach
- Task Flow and Work Products in Harmony/SE
- Systems Engineering Handoff to Hardware and Software Development
Integrated System / Software Development Process
Model-Driven Development of Embedded Systems

System Changes

- Stakeholder Requirements
- Requirements Models, System Use Cases
- Executable Use Case Models
- Architectural Analysis Model(s), System Architecture Model
- System Architecture Baseline
- Software Implementation Model

Requirements Analysis

System Functional Analysis

Design Synthesis

System Changes

Requirements & Test Scenarios

Embedded SW Engineering Harmony/ESW

System Acceptance

SW Analysis & Design

SW Implementation & Unit Test

Module Integration & Test

Requirements Repository *

* Configuration Controlled Knowledge of the System Under Development:
  - Requirements Documentation
  - Requirements Traceability
  - Design Documentation
  - Test Definitions

© Telelogic AB

Innovation 2008 TELELOGIC USER GROUP CONFERENCE
Model-Based Systems Engineering

- Integrated System / Embedded Software Development Process *Harmony*
- *Fundamentals of Model-Based Systems Engineering (Harmony/SE)*
  - Essential SysML Artifacts
  - Service Request-Driven Modeling Approach
- Task Flow and Work Products in *Harmony/SE*
- Systems Engineering Handoff to Hardware and Software Development
Essential SysML Artifacts for Model-Based Systems Engineering

- **SysML Diagram**
  - **Structural Diagrams**
    - Block Definition Diagram
    - Internal Block Diagram
    - Parametric Diagram
  - **Behavioral Diagrams**
    - Use Case Diagram
    - Sequence Diagram
    - Activity Diagram
    - Statechart Diagram
  - **Requirements Diagram**

- UML 2.1
- SysML 1.0
- UML4SysML

© Telelogic AB
SysML Artifacts in Harmony/SE

Requirements Diagram

- Taxonomy of Requirements
- Relationship between Model Elements and Requirements

Use Case Diagram

- Definition of System Scope
- Grouping of Requirements into Use Cases

Activity Diagram

- Functional Flow in Use Case / Block(s)

Structure Diagrams

- Block Definition Diagram: Structural Elements (Blocks) and their Relationship
- Internal Block Diagram: Realization of System Structure

Parametric Diagram

- Parametric Relationship between System Properties

Sequence Diagram

- Message Interactions between Nodes

Statechart Diagram

- State-based Behavior of Block

Parametric Diagram Example:

- «Attribute» mass
- «Attribute» force
- «Attribute» acceleration

Example of Parametric Relationship:

- force = mass * acceleration
- mass: Kg
- force: Newtons
- acceleration: MetersPerSec^2
Model-Based Systems Engineering (*Harmony/SE* )

Artifact Relationships in Requirements Analysis / System Functional Analysis
Model-Based Systems Engineering

- Integrated System / Embedded Software Development Process *Harmony*
- **Fundamentals of Model-Based Systems Engineering (Harmony/SE)**
  - Essential SysML Artifacts
  - Service Request-Driven Modeling Approach
- Task Flow and Work Products in *Harmony/SE*
- Systems Engineering Handoff to Hardware and Software Development
Service Request-Driven Modeling Approach

Step 1: Define Communication Nodes

Step 2: Describe Inter-Nodal Communication in a Sequence Diagram

Step 3: Allocate Service Requests and Operations

Step 4: Define Ports and Interfaces

Communication described through Operational Contracts (OpCon), i.e.
- asynchronous Service Requests via SysML Service Ports followed by
- provided services at the receiving part (state/mode changes or operations)
Model-Based Systems Engineering

- Integrated System / Embedded Software Development Process *Harmony*
- Fundamentals of Model-Based Systems Engineering (Harmony/SE)
  - Essential SysML Artifacts
  - Service Request-Driven Modeling Approach
- Task Flow and Work Products in *Harmony/SE*
- Systems Engineering Handoff to Hardware and Software Development
Key Objectives of the Model-Based Systems Engineering Process *Harmony/SE*

- Identify / derive required system functionality
- Identify associated system modes and states
- Allocate system functionality / modes to a physical architecture
Model-Based Systems Engineering (Harmony/SE)

Requirements Analysis

In the Requirements Analysis phase, the focus is on the analysis of the process inputs.

Stakeholder requirements are translated into system requirements that define
- what the system must do (functional requirements) and
- how well it must perform (quality of service requirements).

Once the requirements are sufficiently understood they are grouped into Use Cases.
Model-Based Systems Engineering (*Harmony/SE*)
System Functional Analysis

In the *System Functional Analysis* phase, the focus is on the translation of the functional requirements into a coherent description of system functions (*Operations*).

Each use case is translated into a model and the underlying requirements verified and validated through *model execution*. 
System Functional Analysis
Derivation of Use Case Scenarios from a UC Black-Box Activity Diagram
Use Case Model Verification and Validation Through Model Execution
Example: Security System Use Case *ControlEntry* (ref. *Rhapsody® Deskbook* Rel. 3.0)
Model-Based Systems Engineering (*Harmony/SE*)

Design Synthesis

The focus of the *Design Synthesis* phase is on the development of a system architecture capable of performing the required functions within the limits of the prescribed performance constraints.
Architectural Design
Allocation of System-Level Operations to Subsystems (Decomposition Level 1)
Detailed Architectural Design
Decomposition of UC Black-Box Scenarios and Definition of Ports / Interfaces
Architecture Model Verification and Validation Through Model Execution
Example: Security System (ref. Rhapsody® Deskbook Rel. 3.0)
Model-Based Systems Engineering

- Integrated System / Embedded Software Development Process *Harmony*
- Fundamentals of Model-Based Systems Engineering (Harmony/SE)
  - Essential SysML Artifacts
  - Service Request-Driven Modeling Approach
- Task Flow and Work Products in Harmony/SE
- Systems Engineering Handoff to Hardware and Software Development
In a model-driven development the key artifact of the handoff from systems engineering to the subsequent system development is the *baselined* executable model.

This model is the repository from which specification documents (e.g. HW/SW Requirements Specifications, ICDs, …) are generated.

Scope and content of the hand-off is dependent on the characteristics of the project and the organizational structure systems engineering is embedded.
Systems Engineering Handoff to Hardware and Software Development

- If the System under Development (SuD) is all software, systems engineering may stop at the functional analysis level. In this case, the hand-off will be executable use case models.

- From the organizational point of view, if there is a separation between systems engineering and subsystems engineering, then systems engineering may stop at the first level of system architecture decomposition. In this case the hand-off will be composed of relevant executable subsystem models.

- If systems engineers hand-off their specifications directly to HW/SW development, then the hand-off will be the respective executable HW and/or SW component models.
The hand-off packages typically are composed of:

- Baseline executable configuration item (CI) model(s)
- Definition of CI-allocated operations including links to the associated system functional and performance requirements
- Definition of CI ports and logical interfaces
- Definition of CI behavior, captured in a statechart diagram
- Test scenarios, derived from system-level use case scenarios
- CI-allocated non-functional requirements
Model-Based Systems Engineering

Integrated Systems / Embedded Software Development Process Harmony

Requirements Analysis
- System Functional Analysis
- Design Synthesis

System Changes
- Stakeholder Requirements
- Systems Engineering Harmony/SE
- Embedded SW Engineering Harmony/ESW
- SW Analysis & Design
- Module Integration & Test
- SW Implementation & Unit Test
- System Acceptance

Model / Requirements Repository
- Stakeholder Requirements
- System Requirements
- Use Case Model
- System Functional Analysis (Use Case-Based)
- System Use Cases
- System Operations
- System Requirements
- Architectural Analysis Model(s)
- UC Activity Diagram(s) (Black-Box)
- Use Case Scenarios (Black-Box)
- System Architecture Model (Baseline)
- Scenarios (White-Box)
- Logical ICDs
- HW/SW Req Specs incl. Test Scenarios

Revised for each Level of System Architecture Decomposition

Architectural Analysis (Trade Study)

Architectural Concept

Architecture Design

Allocated System Operations

Detailed Architecture Design

HW/SW Design

Links providing traceability to original requirements