

DANSE Tools Training

Target Audience

- Qualified systems engineers who are working on Systems of Systems
- Technology providers who want more information on SoS tools
- Researchers advancing the state of the Systems Engineering art

Course Duration

- 5 sessions of 2 hours each – 10 hours of instruction
- Delivered via webinar

Instructional Methods

Method	Target
Lecture	80%
Discussion	None
Short Exercises (instructor-led)	20%
Major Exercise	None

Topical Outline

TOPIC		INSTRUCTOR	DURATION
Session 1	SoS Modeling		Day 1
1.1 Course Introduction	<ul style="list-style-type: none"> • Methodology overview • Solution methods and tools overview • Course topics to be covered • Introduction to architecture frameworks and MBSE 	HCODE (Honour) ✓	0:30
1.2 Rhapsody (overview) (tool 1.1)	<ul style="list-style-type: none"> • How to use Rhapsody to create an SoS UPDM model • Typical UPDM views to create • Views needed for DANSE simulation • Typical information from other tools – Modelica, etc. 	EADS (Sanduka) ✓	0:30
1.3 DANSE modeling extension profiles (tool 1.6)	<ul style="list-style-type: none"> • Contents of the extension profiles • How to access and use them 	OFFIS (Etzien) ✓	0:10
1.4 GCSL Editor (tool 1.7)	<ul style="list-style-type: none"> • GCSL constructs available • How to create a formal GCSL statement from a natural language statement • How to load the Rhapsody plug-in • How to create GCSL statements using the GCSL Editor 	OFFIS (Etzien) ✓	0:40

TOPIC		INSTRUCTOR	DURATION
Session 2 Architecture Changes			Day 2
2.1 Architecture Patterns (tool 1.3)	<ul style="list-style-type: none"> Structure of an architecture pattern How to mine patterns and load them into the library How to search the library to find a useful pattern How to import the pattern into a Rhapsody model 	LU (Kalawsky) ✓	0:40
2.2 Architecture Optimization Workbench (tool 1.4)	<ul style="list-style-type: none"> Functions available in the workbench How to load the Rhapsody plug-in How to modify the UPDM (SysML) model to indicate varying parameters for optimization How to structure Excel spreadsheets to provide the values for the varying parameters How to define the bounds on the optimization Running the optimization Evaluating the results and making selections, including multi-optimization cases How to use the back-annotated models in Rhapsody 	IBM (Shindin) ✓	0:40
2.3 Architecture Generation (tool 1.5)	<ul style="list-style-type: none"> Two major concepts for architecture generation: generation of new architectures, and generation of architecture transformation steps toward a future Defining transformation rules; sources for rules Graph grammar constructs available How to load the Rhapsody plug-in How to create a transformation rule using the Rhapsody plug-in Executing the architecture generation using GROOVE Typical results and what to do with them 	OFFIS (Etzien) ✓	0:40
Session 3 DANSE Tool-Net			Day 3
3.1 Tool-Net Semantic Mediation Container (tool 4.1)	<ul style="list-style-type: none"> Logical connections among DANSE tools Users categories of the tool-net: Not just engineers. Model sharing concepts of the DANSE tool-net – the Semantic Mediation Structure of the Tool-Net, using Jazz and OSLC, scalability and collaboration over the internet. Configuring the semantic mediation container Model sharing: editing model parameters directly in the Tool-Net Extending sharing of black-box components: The FMU objects in the DANSE tools 	IBM (Shani)✓	0:30
3.2 Protégé ontology editor (tool 4.2)	<ul style="list-style-type: none"> What is an ontology? How is it used for Semantic Mediation in the Tool-Net? Using Protégé to creating and edit ontologies 	IBM (Shani) ✓	0:10

TOPIC	INSTRUCTOR	DURATION
3.3 SMC client SDK (tool 4.4) <ul style="list-style-type: none"> Types of external software that might use Tool-Net Software interfaces to the Tool-Net Using and example of the Software Development Toolkit with the Eclipse IDE. 	IBM (Shani) ✓	0:05
3.4 Model sharing tools from SPRINT (with FMU support) <ul style="list-style-type: none"> Rhapsody SysML FMU exporter - how to load the Rhapsody plug-in, how to define and export FMUs from Rhapsody into the Tool-Net Modelica w/ System Modeler TestCast tool for test generation 	IBM (Shani) ✓	0:15
3.5 UPDM and NAF Model Sharing <ul style="list-style-type: none"> UPDM Model sharing: how to export models into the Tool-Net from Rhapsody NAF Model sharing: how to import model contents from the Tool-Net into System Architect 	SODIUS (Albert) ✓	0:15
3.6 MDWorkbench mediation rules editor (tool 4.3) <ul style="list-style-type: none"> What are mediation rules? The role of mediation rules in the Tool-Net semantic mediation structure Creating and editing mediation rules 	SODIUS (Albert) ✓	0:15
Session 4 Joint Simulation and Analysis		Day 4
4.1 DESYRE joint simulation (tool 3.4) (incl. Synthesis for diagnosis and prognosis, tool 3.6) <ul style="list-style-type: none"> Overview of DESYRE capabilities Prerequisite model forms; constraints on the SoS and CS models Overview of the Functional Mockup Interface (FMI) standard and Functional Mockup Units (FMU) Purpose of FMU export in the DANSE tools Typical FMU creation: Dymola Simulink (from Carmeq) How to load models into DESYRE How to define the simulation configuration How to define the trace configuration to be monitored during simulation Executing the joint simulation Typical results; forms of output; how to read and interpret the results 	ALES (Marazza) ✓	0:45
4.2 Monitor Synthesis <ul style="list-style-type: none"> Overview and Purpose of Contract-based run-time verification Contract-based Run-time Paradigm Contract-based Run-time Analysis Tool 	ALES (Marazza) ✓	0:15
4.3 PLASMA statistical model checking (tool 3.3) <ul style="list-style-type: none"> Overview of statistical model checking: PLASMA and GCSL; DANSE achievements Capabilities of the PLASMA software Connecting PLASMA to other tools User interface; how to define a simulation run 	INRIA (Boyer)	1:00

TOPIC	INSTRUCTOR	DURATION
<ul style="list-style-type: none"> Executing the simulation model for checking Typical results; forms of output, how to read and interpret the results 		
Session 5	Modeling Extensions	Day 5
5.1 Abstraction methods (tool 2.1) <ul style="list-style-type: none"> Concepts of abstraction – when to use it, what purpose does it serve? Types of model abstraction How to abstract by analysis of complex models How to abstract from specification How to abstract from observation – statistical learning Typical forms of the results 	OFFIS (Langner) ✓	0:30
5.2 State-Based timing analysis (tool 3.7) <ul style="list-style-type: none"> Motivation for Timing Analysis State-of-the Art Considered Input Model Compositional Timing Analysis using Abstraction and Composition Operations Handling Dynamicity: Impact Analysis Approach 	OFFIS (Gezgin) ✓	0:20
5.3 Contracts analysis <ul style="list-style-type: none"> Compatibility, dominance, and satisfaction of contracts How to check contracts characteristics without the time expense of full simulation Contract Analysis tool as part of DESYRE 	ALES (Senni) ✓	0:20