



UML 2 Compressed™ with MagicDraw™ Workshop

Accelerate your UML 2.x project with this intense, interactive workshop that emphasizes pragmatic modeling principles and techniques, and shows how to apply them with MagicDraw™, an award winning modeling tool.

The Object Management Group first standardized the Unified Modeling Language™ (UML™) in 1997. Since that time the software industry has adopted UML as its primary language for specifying software-intensive systems in a wide variety of domains, ranging from health and finance to aerospace and telecommunications. Recently the OMG has adopted a major revision of UML, UML 2 (currently UML 2.1), which offers many advantages for software architects and systems engineers who are specifying complex systems, including the following:

- **Hierarchical decomposition of structures and support for component-based development.** UML 2 introduces a major new diagram type, Composite Structure diagrams, that includes new constructs (Parts, Ports and Connectors) which allow you to recursively decompose a a system-of-systems into systems, subsystems, components, sub-components, etc.
- **Hierarchical decomposition of behavior.** UML 2 enhances Activity and Sequence diagrams so that you can recursively decompose behaviors into sub-behaviors. For example, you can decompose Action Nodes into sub-Action Nodes, sub-sub-Action Nodes, etc.
- **Improved integration between structural and behavioral models.** When properly applied UML 2 Parts allow you to seamlessly integrate structural and behavioral diagrams. For example, the same EFI Part in a Composite Structure diagram for an Engine might also be reused for a swimlane partition in an Activity diagram for Activate Cruise Control.
- **Enhanced support for executable models.** UML 2 includes a fully integrated Action Semantics that enables executable models capable of driving simulations and automatically generating programming code.

THE PIVOTPOINT TRAINING ADVANTAGE

- **Authored and taught by experts.** All workshops are authored by PivotPoint's founder, Cris Kobryn, an internationally recognized expert in visual modeling languages and model-driven development technologies. (Cris chaired the international standardization teams for UML 1.1, UML 2.0 and SysML 1.0.) In addition, all PivotPoint instructors have 10+ years experience working with Model-Driven Development technologies.
- **Small, intense and interactive.** We limit workshop sizes, usually to a maximum of 12 participants. This ensures that participants get the individual attention that they need to learn quickly. Also, our workshops are intense and highly interactive with frequent work sessions, so you will learn from other participants as well as your instructor.
- **Proven principles and best practices.** Our workshops emphasize proven modeling principles and best practices that will work with all modeling tools that comply with the relevant standards. If you have already chosen a modeling tool, we can integrate optional tool training in your workshop. If you have not yet selected a modeling tool, we can help you select one that best meets your project and team needs.
- **Emphasis on pragmatic problem solving.** Our workshops emphasize the use of modeling technologies to solve tough, practical problems such as those you encounter on your day job. The bigger and more difficult the problem you choose for practice sessions, the more interesting the workshop will be for the participants and the instructor.
- **Flexible choice of venue.** Our workshops are available onsite at Client training facilities, which allow us to customize workshops to meet Client project or team needs, or at PivotPoint training facilities.

WHAT WILL YOU LEARN?

The following are the key learning objectives of this workshop:

- What is UML 2 and why do we model?
- UML 2's basic and advanced constructs for modeling structure and behavior
- How UML 2 can model large, complex systems
- How UML 2 can specify the full system lifecycle: requirements through testing
- Practical guidelines for specifying correct, clear, concise and consistent models
- How you can customize UML 2 for problem domains, such as aerospace, communications, and manufacturing
- How UML 2 can be used with object, component-based and structured analysis/design methods
- How to select tools and methods
- How to draw and execute UML 2 using a selected UML tool: MagicDraw™
- How to learn more about UML 2 modeling

WHO SHOULD PARTICIPATE?

System architects/engineers, software architects/engineers, project managers, and others who want to learn how the UML 2 language can improve how they architect, analyze, design, and manage systems will benefit from this workshop.

PREREQUISITES

Systems or software engineering experience in building large, complex systems. Experience using one or more structured analysis/design, object or component methods is desirable.

WORKSHOP AUTHOR & PRIMARY INSTRUCTOR



Cris Kobryn is the CEO and Founder of PivotPoint Technology Corporation, a company that specializes in Model-Driven Engineering Solutions™ for tough business and engineering problems. He is an internationally recognized expert in visual modeling and model-driven development, and has successfully applied these technologies to diverse industries ranging from aerospace-defense and telecom to financial services and manufacturing. Cris has global experience leading high-performance software development teams, and has architected custom applications and commercial products. He formerly held senior technical positions at Telelogic, EDS, MCI Systemhouse, Inference Corporation, and SAIC.

Cris chaired large international teams of vendors and users to specify the Unified Modeling Language (UML) 1.1 and 2.0 standards for software engineering, and the Systems Modeling Language (SysML) for systems engineering. In recognition of Cris's contributions to the UML the Object Management Group (OMG) presented him with its Distinguished Service Award, and in acknowledgement of his contributions to the SysML the International Council on Systems Engineering (INCOSE) presented him with its Outstanding Service Award. Cris is a contributing editor for *Software and Systems Modeling* journal, and a member of IEEE, INCOSE, ACM, and AAAI.

WORKSHOP SYLLABUS

The workshop syllabus, in a menu form that can be customized to meet your needs, is described at the end of this handout. NOTE: This workshop description and syllabus are subject to revision. Check www.PTCorp.com/training.htm for updates.

WORKSHOP SIZE

The number of workshop participants is restricted to maximize interactions with the instructor, especially during modeling lab sessions. Most workshops are restricted to 12 or fewer participants. Exceptions must be approved by the instructor.

COST, AVAILABILITY, AND FURTHER INFORMATION

This workshop is available at client sites, PivotPoint instructional facilities, or by web conferencing. Costs depend upon your choice of venue, duration, and the number of participants. For further information regarding the contents, availability, and cost of the workshop please email workshops@PTCorp.com or call +1-760-728-9747.

WORKSHOP MENU

All PivotPoint workshops include both structured presentations and interactive hands-on work sessions to reinforce learning principles and best practices. In addition, workshops can be customized for different project and team requirements.

- **3 day workshop** includes: *UML 2 – Basic, UML 2 – Intermediate, and either UML 2 – Advanced or UML 2 – Basic Modeling Tool.*
- **4 day workshop** includes: *UML 2 – Basic, UML 2 – Intermediate, UML 2 – Advanced, and UML 2 – Basic Modeling Tool.*
- **5 day workshop** includes: *UML 2 – Basic, UML 2 – Intermediate, UML 2 – Advanced, UML 2 – Basic Modeling Tool, and UML 2 Advanced Modeling Tool/Project Practicum.*

<p style="text-align: center;">UML 2 – BASIC [Workshop# LU101]</p> <p>Introduction</p> <ul style="list-style-type: none"> • Model-Driven Software Engineering • Basic concepts • Principles and best practices <p>UML 2 Quick Tour</p> <ul style="list-style-type: none"> • Language overview • Diagram walkthrough <p>Diagram Techniques</p> <ul style="list-style-type: none"> • Use Case diagrams • Class diagrams • Sequence diagrams <p>Lifecycle Phases</p> <ul style="list-style-type: none"> • Requirements • Analysis 	<p>Goals</p> <ul style="list-style-type: none"> • Understand the advantages of a Model-Driven Software Engineering approach • Comprehend the differences among architectural models, frameworks, processes and tools • Learn the basic concepts and principles for modeling complex systems with UML 2 • Understand how to specify a correct, complete, concise, and consistent model
<p style="text-align: center;">UML 2 – INTERMEDIATE [Module# LU102]</p> <p>Topics</p> <ul style="list-style-type: none"> • UML 2 as an Architecture Description Language • Architecture patterns and frameworks • Interface-Based Design • Service-Oriented Architectures • Verification & Validation techniques <p>Diagram Techniques</p> <ul style="list-style-type: none"> • Composite Structure diagrams • Activity diagrams • State Machine diagrams • Component diagrams • Deployment diagrams <p>Lifecycle Phases</p> <ul style="list-style-type: none"> • System Design • Construction • Testing 	<p>Goals</p> <ul style="list-style-type: none"> • Understand how to use UML 2 as an Architecture Description Language • Learn how to make your models more scalable • Understand how to improve the integrity and quality of your models • Learn how to apply UML 2 modeling techniques to the full system development life cycle • Understand how to verify and validate your models

<p style="text-align: center;">UML 2 – ADVANCED [Module# LU103; optional]</p> <p>Topics</p> <ul style="list-style-type: none"> • Advanced Structural Modeling • Advanced Behavioral Modeling • Customizing UML for domains and platforms • Executable models <p>Diagram Techniques</p> <ul style="list-style-type: none"> • Interaction Overview diagrams • Timing diagrams • Communication diagrams <p>Putting It All Together</p> <ul style="list-style-type: none"> • Process selection and customization • Tool selection and customization • Modeling futures • Modeling resources 	<p>Goals</p> <ul style="list-style-type: none"> • Learn advanced techniques to refine structural and behavioral models • Understand how to customize UML 2 for your work domain and target platform • Learn how to build executable models, which can drive simulations and generate code • Learn how to select model-driven processes and tools
<p style="text-align: center;">UML 2 – BASIC MODELING TOOL: MAGICDRAW™ [Module# LU111-MD]</p> <p>Selected Modeling Tool Tour</p> <ul style="list-style-type: none"> • Projects and diagrams • Generating documentation • Roundtrip engineering • Importing/exporting models • Model validation and metrics <p>Diagram Techniques</p> <ul style="list-style-type: none"> • Use Case diagrams • Class diagrams • Sequence diagrams • Composite Structure diagrams • Activity diagrams • State Machine diagrams • Other diagrams (TBD) 	<p>Goals</p> <ul style="list-style-type: none"> • Gain familiarity with the user interface and basic features of selected UML 2 modeling tool • Learn how to model most common UML 2 diagram types using selected tool • Understand the strengths and weaknesses of selected tool • Assess UML and XMI standards compliance for selected tool
<p style="text-align: center;">UML 2 – ADVANCED MODELING TOOL/ PROJECT PRACTICUM [Module# LU112; optional]</p> <p><i>[Participants may choose between advanced tool topics and a project practicum.]</i></p> <p>Advanced Tool Topics</p> <ul style="list-style-type: none"> • Team modeling • Roundtrip engineering • Document generation • Requirements-driven modeling • Pattern-based modeling • Customizing profiles <p>Project Practicum</p> <p><i>[This practicum provides an opportunity to apply modeling principles and best practices to solve a practical problem in a creative and supervised workshop environment. Participants can identify a problem in advance, or Instructor will work with participants to identify a practical problem.]</i></p>	<p>Goals</p> <ul style="list-style-type: none"> • Learn advanced tool techniques specifically requested by workshop participants • Solve specific project problems identified by workshop participants