Software Configuration Management
Course Catalog

Tropaion Inc
280 Sheffield St
P.O. Box 1230
Mountainside, NJ 07092
(888) CONFIG1
(908) 654-3871 fax
config1@tropaion.com
## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Course Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Practical Software Configuration Management (CM201)</td>
<td>2</td>
</tr>
<tr>
<td>Practical Software Release Management (CM301)</td>
<td>3</td>
</tr>
<tr>
<td>Practical Project Metrics and Software Measurement (CM302)</td>
<td>4</td>
</tr>
<tr>
<td>Adopting a Useful Software CM Plan (CM303)</td>
<td>5</td>
</tr>
<tr>
<td>Planning for the Capability Maturity Model (CM304)</td>
<td>6</td>
</tr>
<tr>
<td>Sablime® User Training (CM401 &amp; CM401L)</td>
<td>7</td>
</tr>
<tr>
<td>Sablime® Administrator Training (CM402 &amp; CM402L)</td>
<td>8</td>
</tr>
<tr>
<td>Sablime® Advanced Administrator Training (CM403L)</td>
<td>9</td>
</tr>
<tr>
<td>Sablime® Software Release Management (CM404L)</td>
<td>10</td>
</tr>
<tr>
<td>Lucent® nmake - Makefile-Based Product Build Tool (CM421)</td>
<td>11</td>
</tr>
<tr>
<td>Lucent® nmake - Makefile-Based Product Build Tool with Lab (CM425L)</td>
<td>12</td>
</tr>
<tr>
<td>Lucent® nmake – Advanced Makefile-Based Product Build Tool with Lab (CM426L)</td>
<td>13</td>
</tr>
<tr>
<td>PBT™ - Product Build Tool Training (CM121)</td>
<td>14</td>
</tr>
<tr>
<td>Pricing</td>
<td>15</td>
</tr>
</tbody>
</table>

Lucent & Sablime are registered marks of Lucent Technologies, Inc.
ClearCase is a registered mark of Rational Software Corporation.
All other trademarks are the property of their respective owners.
Practical Software Configuration Management (CM201)

This two (2) day workshop is for anyone involved in a software development environment.

1. **Introduction to the Seminar**
   - Scope and Purpose.
   - Audience.
   - Topics.
   - Why SCM is important to the organization.
   - Brief history.

2. **Basic Concepts of SCM.**
   - Definitions and general overview.
   - The four disciplines of product development.
   - Development paradigms and models.
   - The SCM “Triangle.”
   - Ancillary components to SCM.
   - Configuration Items (Artifacts).

3. **Version Control / Change Control**
   - Definitions and the differences between the two.
   - VC and CC tools to accomplish the task.
   - Managing with tasks, not files.

4. **Manufacturing.**
   - Definition of “Product Manufacturing.”
   - Issues that directly impact the ability to manufacture your product.
   - Directory (Build Node) design.
   - Manufacturing tools

5. **Product Build.**
   - What does it mean to “build” product.
   - General strategies.
   - What to build.
   - Build Tools.

6. **Metrics Collection**
   - Introduction to the “Devil's Playground.”
   - The classic Feedback Control Mechanism.
   - Good metrics and bad metrics.

7. **Adopting an SCM Solution.**
   - Opportunity vs. Responsibility.
   - Factors impacting a successful transition to SCM.
   - SCM Adoption Phases.
   - Understanding the costs and savings.
   - Mitigating the risk factors.
Practical Software Release Management (CM301)

This intermediate-level Software CM Seminar is designed to provide attendees with an overview of the knowledge base for designing and implementing a detailed Product Build strategy to support developers, testers, beta users, first-office applications, and general-availability customers. While many projects have recognized the need for implementing some type of version/change control system in addition to employing standard code manufacturing tools like make, nmake, etc., few, if any, have successfully addressed the enormous task of managing a comprehensive product build support environment. Extending the general concepts and issues presented in “CM201, Practical Software CM,” the CM-301 Seminar identifies the numerous issues all projects must address within this mandatory component of Software CM and provides simple, effective, and understandable solutions.

Duration: 1 Days
Prerequisite: CM201 Practical Software CM
Audience: Change Management Organization, Product Build Administrators, Release Coordinators, and Project Management [in addition, developers and testers will find much of the Seminar useful, especially in medium- and large-scale projects.]

Contents:
- Review of General Release Management Issues from CM201
- Product Build Administration and the Change Management Organization
- Product Build Modeling in UNIX using Sablime®, ClearCase®, etc.
- Finite State Building versus Pseudo-Random Task Selection Techniques
- Forward-Flow Task Management Model of Product Build
- Managing Logical and Physical Task Dependency Issues
- Task Severities, Known Deficiency Lists, and Release Readiness
- Official Build Nodes and Trial Build Support
- Master and Incremental Product Build Strategies
- Product Build With and Without ViewPath Capabilities
- Special Issues in Multi-Platform Development Environments
- Special Issues in Multi-Site Development Environments
- Product Building for the Development Team
- Product Building for Single or Multiple Test Teams
- Product Building for External Delivery
- Optimum Product Build Scheduling
- Versioning Schemes and Release Identification
- Quick-Fix and Emergency Release Management
- Constructing Product Releases from Independent Systems or Components
- Implementing Cost-effective Release Distribution Models
- Historical Release Retention and Recovery
- Tool Support for Efficient Product Building and Release Management
Practical Project Metrics and Software Measurement (CM302)

This intermediate-level Software CM Seminar is designed to provide projects with the groundwork necessary to implement an accurate, useful metrics measurement and reporting capability. Demonstrating in an understandable way that three-quarters of all metrics gathered on projects are either ineffective or completely useless, the CM-302 Seminar defines the characteristics of sound metrics analysis and practical software measurement. Based on the original concepts raised in “CM201 – Practical Software CM,” and replete with examples (both good and bad!) attendees will learn how to secure support for, design, integrate, measure, process, and analyze a realistic metrics and software measurement program. With an effective measurement program in place, project management should realize a marked improvement in both overall cost control and on-time product delivery.

Duration: 1 Day
Prerequisite: CM201 Practical Software CM
Audience: Change Management Organization, Project Management, Financial Officers, and Quality Assurance Personnel

Contents:
- Why We Are Compelled to Measure
- Review of the Good and the Bad of Metrics Measurement
- Overview of the Eight Principles of Good Software Measurement
- Detailed Review of the Three Phases of Metrics Program Implementation
- Identifying Useful Project Issues for Measurement
- Selecting Appropriate Measurement Techniques
- Tailoring, Collecting, Processing, and Analyzing Metrics Data
- Setting, Altering, and Using Thresholds in Software Measurement
- Understanding Estimation, Feasibility, and Performance Analysis
- Implementing Metric Measurements Using Common Software CM Tools
- Metrics and the Project’s Task Management Life Cycle
- Gateway Management Using Discrete-Time Metrics
- Roles Project Members Play in the Software Measurement Program
Adopting a Useful Software CM Plan (CM303)

This intermediate-level Software CM Seminar is designed to provide projects with insight into the numerous factors that must be considered when planning any computer program development and maintenance activities. Taking into consideration the essential concepts of sound Software CM practices, the CM-303 Seminar presents a strategy for designing, writing, and distributing a simple, elegant, and practical Software CM Plan that can be used effectively by all members of the project, whether or not they are familiar with the defining issues and inherent problems in all software development models. Sample Software CM Plans are provided on diskette and techniques for adapting these templates to individual projects of various sizes are discussed.

Duration: 1 Day  
Prerequisite: CM201 Practical Software CM
Audience: Change Management Organization and Project Management [in addition, all projects should encourage representative participation from the development, test, and quality assurance communities because of the impact a CM Plan will have on those disciplines]

Contents: Discussion of Common Terms and Acronyms Applicable to a CM Plan  
Software CM Disciplines in Software Management  
Baselines, Processes, and Tool Sets  
When and How to Plan and Implement Software CM  
Responsible Parties for the Software CM Plan Adoption Activities  
Selecting a Style for Your Software CM Plan  
Identifying Sections for Your Software CM Plan  
Relationship Between the Software CM Plan and Selected Software CM Tools  
Roles, Gateways, Life Cycles, and Task States in Your Software CM Plan  
Software CM Plan for Traditional Product-Line Management  
Software CM Plan for Prototype Development  
Software CM Plan for Critical Software Projects  
Software CM Plan for Inherited Software Projects
Planning for the Capability Maturity Model (CM304)

This intermediate-level Software CM Seminar is designed to provide projects with a basic understanding of the CMU/SEI’s Capability Maturity Model which provides them with guideposts to measure their transformation from a chaotic, immature, unrepeatable software process to that of a structured, well-managed, confident development paradigm. For those projects considering CMM Certification, the CM-304 Seminar will introduce them to the five Maturity Levels and the key practices necessary to achieve each level. Alternatively, exposure to the Capability Maturity Model can also be used by projects internally to assess their own software processes, promote software process improvements, and evaluate various software development capabilities.

**Duration:** 1 Day  
**Prerequisite:** CM201 Practical Software CM  
**Audience:** Configuration Management Organization, Project Management, Company Management, and Quality Assurance Personnel  

**Contents:**  
Overview of the Capability Maturity Model  
Interpreting the CMM  
The Five Maturity Levels and Software Process Capabilities  
Key Process Areas and Goals  
Common Features and Implementation  
Key Practices and Activities  
The CMM and Your Project’s Development Life Cycle  
Similarities and Differences Between Software CM and CMM  
Preparing for CMM Certification
Sablime® User Training (CM401 & CM401L)

TARGET AUDIENCE: Team members on new or active projects who are using Sablime® for the first time, who have never had a formal Sablime user course, or who need a refresher course. The Sablime® workshops provide valuable, practical SCM knowledge and tool-based expertise for managers, developers, hardware engineers, document writers, product builders, integration and system testers, quality assurance staff, distribution personnel, and any other project members who plan to use Sablime® effectively for total project control and management.

Note that this course is completely rewritten and updated for Sablime® 6.x.

PREREQUISITES: Familiarity with the UNIX environment (as a user); familiarity with the Korn Shell (ksh programming knowledge is a plus).

LENGTH: CM401 - 2 days
         CM401L - 3 days

DESCRIPTION: This workshop will teach Sablime® users about: general issues involved in Configuration Management; each project member’s role in the overall SCM plan; the differences in hardware, software (firmware), and document processes; an overview of Sablime®, its history and its features; Sablime® terminology; the Sablime® databases and how Sablime® stores information; interacting with the Sablime® command interface; setting up to use Sablime® and tuning the user environment; creating Sablime® groups to simplify user tasks; the complete MR lifecycle subsystem; how and when to use each of the Sablime® MR management commands; external MR management (when applicable); Web Sablime®; and using the Sablime® information retrieval system effectively and efficiently (the report and query commands). In addition this workshop will teach users about: the design and use of project directory structures; Sablime® source database details; the complete source management and source file control subsystem (i.e., for source code, documents, test streams, action items, memos, etc.); how and when to use each Sablime® source management command; the Sablime® underlying version control systems (SCCS and SBCS); retrieving files with the sget and getversion commands; how and where products are built, including developer (private), unit test, integration/system test, and official nodes; build tools for source code and documents (focusing on nmake with ViewPath capabilities); product version and release management; MR dependency issues and how they relate to builds; and how to interact directly with the Sablime® database with ksh and other tools to obtain useful information quickly.

The lab version of the course (CM401L) includes a number of practical exercises that reinforce critical skills.
Sablime® Administrator Training (CM402 & CM402L)

TARGET AUDIENCE: Administrators on new or active projects who are planning to use or are using Sablime® as their software configuration management tool and want to accurately, effectively, and efficiently install Sablime®, customize Sablime® to specific project needs, maintain Sablime®, aid in designing the project's official directory structure, maintain the standard development environment, assist in or direct the product build and delivery functions, and comfortably providing general program counseling to users.

Note that this course is completely rewritten and updated for Sablime® 6.x.

PREREQUISITES: CM401(L) Sablime® User Training or equivalent plus familiarity with the UNIX environment and some programming experience. Familiarity with SCCS and/or SBCS is essential. In software environments, nmake and product build familiarity is extremely useful (prospective administrators should strongly consider workshop CM421 or CM425L).

LENGTH:      CM402 - 4 days
             CM402L - 5 days

DESCRIPTION: This workshop presents attendees with a systematic introduction to the duties and responsibilities of the Sablime® administrator (SDBA) with regard to the acquisition, installation, customization, maintenance, and daily administration of the Sablime® Configuration Management tool. Integral to these classes is a discussion of sound CM principles and practices, woven into both the overview and subsequent detailed discussions of the numerous Sablime® features and capabilities. Primary topics include 1) the foundations of CM and your product development direction, 2) CM, the Sablime® SDBA, and the role they play in successful product development, 3) Sablime® acquisition and installation, 4) User management, 5) Product creation, 6) generic design, creation, source instantiation, and on-going maintenance activities, 7) a tour of the Sablime® Databases and data management, 8) essential tool customization techniques, including setup scripts, screen design and layout, entry field implementation, popup selection windows, team/state management, etc., 9) source code (file) and directory management, 10) miscellaneous administrative procedures, including audits, cleanup, error handling, etc., and 11) advanced customization techniques, including data collection design, metrics issues, pre- and post-action triggers, command wrappers, etc., 12) instructions for installing and managing the Sablime® web interface.

The lab version of the course (CM402L) includes a number of practical exercises that reinforce critical skills.
**Sablime® Advanced Administrator Training (CM403L)**

**TARGET AUDIENCE:** Administrators on larger or more complex projects planning to use advanced Sablime® features.

*Note that this course is completely rewritten and updated for Sablime® 6.x.*

**PREREQUISITES:** CM402(L) Sablime® User Training. Note that this workshop builds on the concepts and techniques taught on CM402(L).

**LENGTH:** CM403L - 2 days

**DESCRIPTION:** This workshop is a logical extension to CM-402 and CM-402L. Participants are introduced to the advanced Sablime® features and attendant Configuration Management practices and principles. These classes present up to a half-day session on WebSablime and the various installation issues involved, including a basic description of web site vocabulary, design, and management topics so that new Sablime® administrators (SDBAs) can work with their UNIX Administration counterparts effectively in getting WebSablime operational.

Additional primary topics include:

1. Advanced generic management techniques (i.e., using lightweight, shadow, mono-thread, mosaic-of, and phased generics)
2. Installing Sablime®’s multi-machine feature and the management of subsequent multi-site issues
3. Implementation of Sablime®’s External MR Management capabilities for automated project-to-project communications
4. Advanced database corruption issues and corrective actions by the DBA,
5. Administrative tool development suggestions for monitoring and maintaining the Sablime® Databases.
Sablime® Software Release Management (CM404L)

This workshop is designed to provide attendees with a extensive, hands-on practical knowledge base for designing and implementing a detailed Product Build strategy to support developers, testers, beta users, first-office applications, and general-availability customers in a Sablime® environment. While many projects have recognized the need for implementing some type of version/change control system in addition to employing standard code manufacturing tools like make, nmake, etc., few, if any, have successfully addressed the enormous task of managing a comprehensive product build support environment. Extending the general concepts and issues presented in “CM201, Practical Software CM,” the CM-404L workshop identifies the numerous issues all projects must address within this mandatory component of Software CM and provides simple, effective, and understandable solutions.

Duration: 3 Days
Prerequisite: CM201 Practical Software CM and CM401 Sablime® User Training
Audience: Change Management Organization, Product Build Administrators, Release Coordinators, and Project Management [in addition, developers and testers will find much of the Seminar useful, especially in medium- and large-scale projects.]

Contents:
- Review of General Release Management Issues from CM201
- Product Build Administration and the Change Management Organization
- Product Build Modeling in UNIX using Sablime®, ClearCase®, etc.
- Finite State Building versus Pseudo-Random Task Selection Techniques
- Forward-Flow Task Management Model of Product Build
- Managing Logical and Physical Task Dependency Issues
- Task Severities, Known Deficiency Lists, and Release Readiness
- Official Build Nodes and Trial Build Support
- Master and Incremental Product Build Strategies
- Product Build With and Without ViewPath Capabilities
- Special Issues in Multi-Platform Development Environments
- Special Issues in Multi-Site Development Environments
- Product Building for the Development Team
- Product Building for Single or Multiple Test Teams
- Product Building for External Delivery
- Optimum Product Build Scheduling
- Versioning Schemes and Release Identification
- Quick-Fix and Emergency Release Management
- Constructing Product Releases from Independent Systems or Components
- Implementing Cost-effective Release Distribution Models
- Historical Release Retention and Recovery
- Tool Support for Efficient Product Building and Release Management
Lucent® nmake - Makefile-Based Product Build Tool (CM421)

TARGET AUDIENCE: Users and administrators on new or active projects that are planning to implement nmake for controlling the project's product build process with this development-level tool. The workshop is geared for developers, testers, and nmake administrators/counselors.

PREREQUISITES: Solid familiarity with the UNIX environment (as a user). Korn Shell (ksh) programming is indispensable. Familiarity with the UNIX make tool is helpful but not required.

DESCRIPTION: This 3-day intensive workshop provides attendees with a general overview of the Lucent® nmake makefile-based product build tool and will help them to develop efficient and useful makefiles. Topics include: an overview of nmake; background and history of the tool; a comparison of nmake and make; the product structure; nmake command line options; makefile definitions and conventions; nmake statefiles and object files; assertions; assertion operators; common actions; nmake metarules, base rules, and global rules; USE rules; programming with nmake directives; global makefiles; nmake variables (including definitions, assignment, expansion, automatic, state, and editing); atoms and binding; and nmake viewpathing.

Attendees will also be provided with guidelines for writing makefiles by reviewing sets of working project makefiles.
Lucent® nmake - Makefile-Based Product Build Tool with Lab (CM425L)

TARGET AUDIENCE: Users and administrators on new or active projects that are planning to implement Lucent® nmake for controlling the project's product build process with this development-level tool. The workshop is geared for developers, testers, and nmake administrators/counselors.

PREREQUISITES: CM201 and familiarity with the UNIX environment plus programming experience (C/C++/Java/etc.). Familiarity with the UNIX make tool is helpful but not required.

DESCRIPTION: This 3-day intensive workshop with lab provides attendees with a general overview of the Lucent® nmake makefile-based product build tool and will help them to develop efficient and useful makefiles. Topics include: an overview of nmake; background and history of the tool; a comparison of nmake and make; the product structure; nmake command line options; makefile definitions and conventions; nmake statefiles and object files; assertions; assertion operators; common actions; nmake metarules, base rules, and global rules; USE rules; programming with nmake directives; global makefiles; nmake variables (including definitions, assignment, expansion, automatic, state, and editing); atoms and binding; and nmake viewpathing.
**Lucent® nmake – Advanced Makefile-Based Product Build Tool with Lab (CM426L)**

**TARGET AUDIENCE:** Users and administrators of larger or more complex projects that are nmake-savvy who wish to learn advanced techniques for implementing exceptional code manufacturing paradigms into their development environment.

**PREREQUISITES:** CM421 or CM425L.

**DESCRIPTION:** This 2-day intensive workshop with lab presents attendees with a systematic, CM-inspired capstone to the CM-421 or CM-425L classes covering the most powerful product manufacturing model available by expanding the independent Makefile constructs learned in the earlier classes into a multi-Makefile, multi-tasking production mode.

Primary topics include:

1) Using `nmake` in multi-platform development and production environments
2) Advanced Assertion design and implementation (the ":TGEN:" assertion operator for multi-platform production environments)
3) Advanced variable editing
4) Advanced "special atoms"
5) Survey of `nmake`'s "Global Rules" engine, "makerules.mk;"
6) The `probe` utility and probe files
7) Adding new "scan rules" to the nmake repertoire,
8) Using `nmake` and `coshell` for a multi-thread, multi-processor manufacturing capability..
PBT™ - Product Build Tool Training (CM121)

TARGET AUDIENCE: New or active projects who are planning to standardize their product build and delivery environment and would like to automate the process with PBT™, the Product Build tool. The PBT™ workshop is ideal for Sablime® administrators, integration and system testers, product build teams, and/or product delivery staff. and want to have administrators who can accurately, effectively, and efficiently install Sablime®, customize Sablime® to specific project needs, maintain Sablime®, aid in designing the project's official directory structure, maintain the standard development environment, assist in or direct the product build and delivery functions, and are comfortable providing general program counseling to users.

PREREQUISITES: CM401 (Sablime® User Training). Solid familiarity with the UNIX environment (as a user). Korn Shell (ksh) programming is indispensable. Familiarity with SCCS and/or SBCS is helpful. In software environments, nmake familiarity is useful (prospective product builders should consider workshop CM421).

DESCRIPTION: This 2-day workshop is designed to provide attendees with the knowledge to Implement a complete product build methodology for their projects. Class topics include: an Introduction to the PBT™ including build philosophy, build node design, PBT™ implementation, and the pb driver files; interfacing Sablime®, nmake, and PEI; controlling and coordinating release builds and distribution; software vs. document "builds;" point releases; build authentication, history, and re-creation; Incremental build methodology; viewpathing and development synchronization; multi-generic, multi-release, and multi-site build methodology; and build scheduling for development and test (integration, system, etc.).
## Public Course Pricing

<table>
<thead>
<tr>
<th>COURSE</th>
<th>Price per Student*</th>
<th>Days</th>
</tr>
</thead>
<tbody>
<tr>
<td>CM121</td>
<td>745.00</td>
<td>2</td>
</tr>
<tr>
<td>CM201</td>
<td>745.00</td>
<td>2</td>
</tr>
<tr>
<td>CM301</td>
<td>395.00</td>
<td>1</td>
</tr>
<tr>
<td>CM302</td>
<td>395.00</td>
<td>1</td>
</tr>
<tr>
<td>CM303</td>
<td>395.00</td>
<td>1</td>
</tr>
<tr>
<td>CM304</td>
<td>395.00</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>COURSE</th>
<th>Price per Student*</th>
<th>Days</th>
</tr>
</thead>
<tbody>
<tr>
<td>CM401</td>
<td>745.00</td>
<td>2</td>
</tr>
<tr>
<td>CM401L</td>
<td>1,295.00</td>
<td>3</td>
</tr>
<tr>
<td>CM402</td>
<td>1,445.00</td>
<td>4</td>
</tr>
<tr>
<td>CM402L</td>
<td>1,995.00</td>
<td>5</td>
</tr>
<tr>
<td>CM403L</td>
<td>945.00</td>
<td>2</td>
</tr>
<tr>
<td>CM404L</td>
<td>1,295.00</td>
<td>3</td>
</tr>
<tr>
<td>CM421</td>
<td>1,095.00</td>
<td>3</td>
</tr>
<tr>
<td>CM425L</td>
<td>1,295.00</td>
<td>3</td>
</tr>
<tr>
<td>CM426L</td>
<td>945.00</td>
<td>2</td>
</tr>
</tbody>
</table>

* Discounts available for companies sending more than two students to a public course on the same purchase order.

## On-Site Course Pricing

<table>
<thead>
<tr>
<th>COURSE</th>
<th>Price**</th>
<th>Days</th>
<th>Max Students</th>
</tr>
</thead>
<tbody>
<tr>
<td>CM121</td>
<td>7,000.00</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>CM201</td>
<td>7,000.00</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>CM301</td>
<td>3,500.00</td>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>CM302</td>
<td>3,500.00</td>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>CM303</td>
<td>3,500.00</td>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>CM304</td>
<td>3,500.00</td>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>CM401</td>
<td>7,000.00</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>CM401L</td>
<td>12,495.00</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>CM402</td>
<td>14,000.00</td>
<td>4</td>
<td>20</td>
</tr>
<tr>
<td>CM402L</td>
<td>19,495.00</td>
<td>5</td>
<td>12</td>
</tr>
<tr>
<td>CM403L</td>
<td>8,995.00</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>CM404L</td>
<td>12,495.00</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>CM421</td>
<td>10,500.00</td>
<td>3</td>
<td>20</td>
</tr>
<tr>
<td>CM425L</td>
<td>12,495.00</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>CM426L</td>
<td>8,995.00</td>
<td>2</td>
<td>12</td>
</tr>
</tbody>
</table>

** Reasonable travel and lodging cost for instructor will be added to total price. Discounts available for companies scheduling more than 1 on-site course on the same purchase order.