

## Training: Micro Focus CMS350 - Configuration Management System Advanced



### TRAINING GOALS:

This five day course enables students to understand advanced concepts and acquire Universal CMDB (UCMDB) development skills. Advanced topics are discussed and implementation methods and techniques are demonstrated using hands-on exercises. Jython, Java, and Web Services are introduced in the context of UCMDB.

The hands-on lab environment uses UCMDB version 10.2

Upon successful completion of this course, you should be able to:

- Use multiple tenants using the Multitenancy feature
- Integrate multiple Configuration Management Databases (CMDBs) and explain the integration components and process
- Use advanced modelling techniques and change the composite CI for the Universal CMDB (UCMDB) browser and Configuration Manager
- Explain the reconciliation process and use identification rules
- Develop new adapters: discovery and integration
- Use UCMDB data in your adapters
- Develop your own Jython scripts for your adapter
- Develop your own Generic database adapter and federate data into UCMDB or populate UCMDB with data
- Develop your own Java integration adapter
- Use the UCMDB API for Java adapters
- Develop your own XML push adapters
- Explain the Web Services push adapter
- Develop Java code using the UCMDB API
- Use the UCMDB Java API for querying and populating the UCMDB
- Develop your own Web Services client

### Audience/Job Roles

This course is recommended for:

- Business Technology Optimization (BTO) architects leveraging the power of the integrated CMDB across the BTO/BSM portfolio. –

- Those in charge of Configuration Management and the documentation and storage of business services and their related assets and relationships.
- Project managers, application modellers, discovery engineers, and UCMDb implementation consultants.
- Integration specialists implementing data federation across the different business silos and software product families

## CONSPECT:

- Course Overview
  - Contents of the course
  - Goals of the course
  - Recognizing fellow participants
  - The class agenda
  - Prerequisites
- CMS End-to-end Scenario
  - Define CMDB and Configuration Management System (CMS)
  - Describe the relationships between Service Asset and Configuration Management (SACM), Information
  - Technology Asset Management (ITAM), and CMS
  - List CMS use cases • Explain different integration methods
  - Explain the different ways to consume data
- Multi Tenancy
  - Describe multitenancy
  - List multitenancy use cases
  - Describe multitenancy architecture
  - Explain the differences between a tenant and a customer
  - Use multitenancy management tools
- Integrating Multiple CMDBs
  - Describe multiple CMDB synchronization
  - Explain the need for multiple CMDB synchronization
  - List multiple UCMDb components
  - Describe the CMDB adapter
  - Explain what the Global ID is
  - Describe what a push back ID is
  - Name the uses of Push and Federation in CMD synchronization
  - Describe the CLIP solution

- Name the UCMDB synchronization limitations
- Advanced Modeling
  - Describe the class model
  - Build complex relationships
  - Explain the principle of modeling for change management
  - Explain the principle of modeling for availability management
  - Explain modeling for UCMDB browser/CM
  - Explain URM functions
- Reconciliation
  - Describe the Reconciliation Engine in UCMDB 10
  - List reconciliation processes
  - Explain the history of reconciliation
  - Name reconciliation types
  - Use identification rules
- Adapter Development and Writing
  - Explain the need for development writing
  - Describe the Adapter development lifecycle
  - List the Adapter components
  - List the ways to access the CMDB data
  - Explain the differences between adapter types
  - Name the best practice in Adapter development
  - Describe remote process execution
- Developing Jython Adapters
  - Explain Python and Jython
  - Describe Python basics
  - Explain the Jython adapter structure
  - Explain the ObjectStateHolder
  - Describe a framework instance
  - Name OOTB UCMDB libraries
- Developing Generic DB Adapters
  - Describe the Generic Database Adapter (GDBA)
  - Describe Java Hibernate
  - Name the preparation for adapter steps
  - Recognize the mapping tags
  - List the differences between Federation and Population
  - List GDBA limitations

- Developing Java Adapters
  - Describe Java adapters
  - Describe the Federation Framework
  - List the federation framework capabilities
  - Use Java adapter resources
  - Recognize Java adapter structure
  - Use Java adapter coding
  - Use logging and debugging
- Developing Generic Push Adapters
  - Describe the Generic push Adapter
  - Differentiate between XML and the Web Services push adapter
  - Prepare the push adapter
  - Use push adapter Jython Script writing
  - Use the differential synchronization technique
  - List Generic push adapter best practices
- Using Data Flow Management API
  - List Data Flow Management (DFM) API use cases
  - Describe the DFM API
  - Use the DFM API documentation and access the Web Services Description Language (WSDL) file
  - Describe the DFM code general structure
  - Use DFM code examples
- Using Data Flow Management API
  - Name UCMDB API use cases
  - Describe the UCMDB API
  - Describe the UCMDB API flow
  - Use the UCMDB API general structure
  - Use UCMDB API code samples
- Using the CMDB Web API
  - Name uses for the Web Services API
  - Describe the UCMDB Web Services API
  - Access the Web Services API
  - Use the UCMDB API general structure
  - Explain UCMDB update using Web Services
  - Use code samples
- Hardening CMS
  - Describe UCMDB security threats

- Define basic security measurements
- Explain the role of reverse proxy and demilitarized zone (DMZ)
- Describe MSSQL recommended practice
- Explain the Confidential Manager system role
- Describe saving sensitive data in the memory
- Explain HA and hardening
- Performance Tuning and Sizing
  - Recognize UCMDB needs
  - Explore organization examples
  - Learn about use cases
  - Define server and probe specifications
  - Maintain a high performing UCMDB
- Appendix A: High Availability Deployment
  - Describe the purpose of High Availability (HA)
  - Explain typical HA architecture
  - Explain the principles of data flow
  - List the supported products
  - Describe HA configuration steps

## REQUIREMENTS:

- Successful completion of UCMDB120 UCMDB 10.x Essentials and UD120 Universal Discovery 10.x Essentials
- At least 6 months experience with UCMDB & Universal Discovery.
- Prior experience with scripting language such as Python, Perl, etc.

## Difficulty level



## CERTIFICATE:

The participants will obtain certificates signed by Micro Focus (course completion).

## TRAINER:

Authorized Micro Focus Trainer