

ITX Training Catalog

Effective August 1, 2017

COURSE: ITX Fundamentals (4 days)

Purpose: IBM Transformation Extender provides real-time transformation of complex data, including EDI X12, HIPAA, EDIFACT, and SWIFT. Through a combination of lectures and hands-on lab exercises, students learn how to use the IBM Transformation Extender Design Studio to define, transform, and route business objects and complex flat file data.

Specific topics covered by the course include an overview of the IBM Transformation Extender product and its features, creating and deploying maps, and working with a resource registry. Hands-on exercises give students practical experience with skills such as optimizing maps, handling and logging errors, defining data, and indexing output.

Audience: This course is designed for programmers and analysts who need to implement data integration and transformation solutions using IBM Transformation Extender.

Prerequisites: Before taking this course, students should have a basic understanding of data formats and processing logic and have hands-on experience with Microsoft Windows.

Objectives: After completing this course, students should be able to:

- Define workspaces and projects using the Eclipse Workbench
- Use the Type Designer to define data structures, data properties, and data components as type trees
- Validate type trees to ensure that they accurately describe the data
- Transform flat files using the Map Designer
- Use type design and mapping techniques to improve validation and throughput
- Use the Resource Registry to improve portability between operating environments
- Implement error handling
- Optimize map performance

Unit Topics

1. **IBM Transformation Extender product overview**
 - a. Product overview
 - b. Concepts and components overview
 - c. Design environment
 - d. Runtime services
 - e. Other components
 - f. Developing and deploying solutions
2. **Eclipse workbench fundamentals**
 - a. Use the Eclipse Workbench.
 - b. Create a workspace.
 - c. Create a project.
 - d. Import file into a project.
3. **Defining the data**

- a. Define an object as an item, group, or category.
 - b. Create a type tree.
 - c. Add a new type.
 - d. Explain and use type inheritance.
 - e. Explain and use type property propagation.
- 4. Defining properties**
 - a. Describe properties associated with an item, group, or category.
 - b. Explain the difference between explicit and implicit groups.
 - c. Define a delimiter and explain the various delimiter locations.
 - d. Define syntax properties.
- 5. Defining components**
 - a. Describe a component.
 - b. Create component lists.
 - c. Define component ranges.
 - d. Analyze a type tree for structure and logical consistency.
- 6. Validating data definitions**
 - a. Use the Map Designer to create a validation map.
 - b. Change map settings to generate a trace file.
 - c. Use a trace file to identify problems in data and data definitions.
- 7. Type Tree Importer**
 - a. Know what kind of data can be defined by using Type Tree Importer.
 - b. Build type tree by using Type Tree Importer.
- 8. Building additional validation into type trees**
 - a. Use restriction lists to validate input.
 - b. Use component rules to ensure that data values meet certain criteria for acceptance.
- 9. More about mapping**
 - a. Identify and write map rules.
 - b. Use Index to create output.
 - c. Use functions in map rules.
- 10. Functional maps**
 - a. Explain when to use a functional map.
 - b. Create a call to a functional map.
 - c. Use the Functional Map Wizard to create a functional map.
 - d. Describe the components and properties of a functional map.
 - e. Explain how the Map Designer evaluates functional maps.
 - f. Use the Map Debug option to troubleshoot map output.
 - g. Use the Map Organizer to track and resolve unresolved rules.
- 11. More efficient mapping**
 - a. Use functions to restrict the data that is passed to the functional map.

- b. Define and create a partition.
- c. Assign identifier component attributes.
- d. Create summary data in an output file.
- e. Add an output card to generate a second output file.

12. Error handling

- a. Describe error handling and recovery.
- b. Map invalid data.
- c. Modify map settings to generate a map audit log.

13. Resource Registry

- a. Use the Resource Registry to process various resources based on the active environment.

14. Database Interface Designer basics

- a. Specify the databases to use for a source or target.
- b. Define query statements.
- c. Generate type trees for queries or tables.

15. Map optimization

- a. Map Profiler
- b. Map settings
- c. Input card settings
- d. Output card settings

16. Map deployment

- a. Build maps for target platform
- b. Identify other Transformation Extender files that effect map runtime
- c. Create map reports

COURSE: ITX Advanced Techniques (3 days)

Purpose: This course provides the concepts and techniques needed to optimize and improve the performance of IBM Transformation Extender maps and type trees and to solve complex transformation problems.

- The course focuses on:
 - Type tree optimization
 - Map optimization
 - Performance optimization

Audience: This course is designed for programmers and analysts who require deeper understanding of implementing data integration and transformation solutions using IBM Transformation Extender.

Pre-requisites: Before taking this course, students should complete *IBM Transformation Extender Fundamentals*, and have the following experience:

- Solid understanding of data formats and processing logic
- Proficiency in using Microsoft Windows
- Actual experience with ITX mapping and type tree design, preferably 3 months.

Objectives: After completing this course, you should be able to:

- Explain and implement type tree design techniques to improve throughput and simplify mapping
- Identify recommended practices for mapping rules
- Use advanced functions in mapping rules to solve complex transformation problems
- Use router maps to select and run a map based on data content and variable processing requirements
- Use Internet and utility adapters to package, retrieve, and route data
- Identify modifications that can be made to type trees and maps to optimize performance
- Create custom reports from the map audit log

Unit Topics

1. **Advanced topics overview**
 - a. Defining data for optimal performance
 - b. Mapping for best possible performance
 - c. Map settings that affect performance
 - d. Card settings that affect performance
 - e. Performance issues when mapping with databases
2. **Art of the type tree**
 - a. Describe how data is validated
 - b. Use control-break logic to define data
 - c. Use partitions, choice groups, and unordered groups to simplify mapping
 - d. Modify type trees to optimize data throughput

3. **Art of the mapping rule**
 - a. Identify best practices for mapping rules
 - b. Use the Map Debugger to identify problems
 - c. Use advanced functions in mapping rules to solve complex transformation problems
4. **Router maps**
 - a. Use a router map to select and run a map based on content in the source data stream.
5. **Using Internet and utility adapters**
 - a. Use the Internet and utility adapters to package, retrieve, and route data
6. **Optimizing performance**
 - a. Identify modifications that can be made to type trees and maps to optimize performance
7. **Mapping the audit log**
 - a. Describe the structure of the audit log type tree
 - b. Use the audit log and audit log type tree to identify problems and create custom reports
8. **Map and type tree utility commands**
 - a. Access the core functionality of the Design Studio command line
 - b. Use Design Studio command-line utilities to:
 - c. Analyze type trees
 - d. Export and import type trees
 - e. Compile maps
 - f. Export and import maps

COURSE: ITX Using XML (2 days)

Purpose: This course teaches students how to use the IBM Transformation Extender Design Studio to define and transform complex business objects into any format using Extensible Markup Language (XML) data. Through lectures and hands-on lab exercises, students learn how IBM Transformation Extender transforms data through the use of XML schemas, type trees, and mapping. Topics covered include using XML document type definition (DTD) and XML schema importers to generate type trees, using the Map Designer to map XML data, and using native XML schemas instead of type trees in a map. In this course, emphasis is placed on mapping project implementation using the Transformation Extender Design Studio to validate and map XML data.

The hands-on exercises reinforce the lectures, and provide students with practical experience working with skills such as mapping data, using XML schemas, problem determination, and using the WebSphere Transformation Extender utilities for XML.

Audience: This course is designed for programmers and analysts who implement XML data integration and transformation solutions using IBM Transformation Extender.

Prerequisites: Before taking this course, students should complete course WE210 or VE210, *IBM WebSphere Transformation Extender V8.2 Fundamentals*. Students should also be able to describe the XML elements, attributes, data types, and structure when given an XML schema or document.

Objectives: After completing this course, students should be able to:

- Generate type trees from XML schemas using the WebSphere Transformation Extender XML schema importer
- Describe the differences between Xerces-based and WebSphere Transformation Extender classic type trees and validation
- Describe the structure of an XML type tree generated by the WebSphere Transformation Extender XML schema importer
- Transform XML files using the Map Designer
- Use the trace file and map debugger to identify problems
- Use an XML schema instead of a type tree in a map
- Use the Any-2-XML utility to create an XML schema from a type tree

*** This course assumes a basic understanding of XML data and XML schemas.**

Course Topics

1. XML Review

- a. Describe the XML structure and components of an XML document
- b. Describe the XML structure, components and attributes of an XML Schema Definition (XSD)

- c. Describe the structure and components of an XML DTD
 - d. Identify the difference between valid and well-formed XML
 - e. Describe the IBM Transformation Extender Design Studio features that are specific to XML data
 - f. Find additional information about XML
- 2. Using the IBM Transformation Extender XML schema importer**
 - a. Generate a type tree from an XML schema document
 - b. Describe the differences between Xerces-based and IBM Transformation Extender classic type trees and validation
 - c. Identify limitations and XML parsing error messages
 - d. Analyze the importer error logs and messages
- 3. Understanding XML schema type trees**
 - a. Identify the major components of an XML schema type tree
 - b. Describe the structure of the type tree
 - c. Describe the type properties for XML
 - d. Describe the differences between Xerces-based and IBM Transformation Extender classic type trees
- 4. Mapping XML Document components**
 - a. Map an XML document
 - b. View run results as XML
 - c. Describe the XML-specific functions
- 5. Error handling**
 - a. Enable XML document verification in a map
 - b. Analyze a trace file for XML data
 - c. Correct an XML file that contains errors
- 6. Mapping schema components**
 - a. Describe types created by the XML schema importer for XML attributes
 - b. Map XML data that contains substitution groups
 - c. Generate trees with XML XSDL hints
 - d. Add XSDL hints to a map
 - e. Add namespaces to output
- 7. Using native XML schemas in a map**
 - a. Select an XML schema as the type in an input or output card
 - b. Identify the top-level component in the schema for mapping
 - c. Map using an XML schema instead of a type tree
- 8. Utilities for XML**
 - a. Describe the utility commands for XML
 - b. Use the Any-2-XML utility to create an XML schema from a type tree and map the data to XML

COURSE: ITX System Design (2 days)

Purpose: The IBM Transformation Extender Design course is designed to provide a fundamental understanding and hands-on experience using WebSphere Transformation Extender to design and automate the execution of systems of maps based on time and event triggers using the Launcher.

Audience: The intended audience is programmers and analysts who need to implement data integration and transformation solutions using WebSphere Transformation Extender Launcher.

Prerequisites: Students must possess a basic understanding of data formats, databases and processing logic. Completion of the IBM Transformation Extender Fundamentals course is required.

Objectives: After completing this course, you should be able to:

- Use the WebSphere Transformation Extender Integration Flow Designer to define systems
- Prepare systems to run
- Use Launcher tools for controlling and monitoring system
- Use the Resource Registry to improve system portability

Course Topics

1. ITX Overview

- a. Describe the components of IBM Transformation Extender
- b. Describe the functions of the IBM Transformation Extender Management Tools

2. Integration Flow Designer Fundamentals

- a. Create a new system definition file and system
- b. Display collapsed and expanded views of a system definition file
- c. Select and move objects in the system window

3. Defining system components

- a. Add map components to a system
- b. Modify map component characteristics
- c. Create new map components
- d. Add document links to map and card components

4. Defining subsystem components

- a. Add subsystems to a system
- b. Change subsystem characteristics

5. Systems and servers

- a. Describe system server configurations
- b. Create server definitions
- c. Import and export server definitions
- d. Assign servers to systems

6. Defining execution modes and settings

- a. Identify system execution modes
- b. Assign an execution mode to a system

- c. Select the execution mode settings

7. Preparing systems to run

- a. Use the Launcher Administration tool to configure the Launcher
- b. Analyze a system for consistency
- c. Generate command files from the Integration Flow Designer
- d. Generate Launcher files from the Integration Flow Designer
- e. Create a system deploy script
- f. Deploy systems using a deploy script

8. Running systems with the Launcher

- a. Start the Launcher
- b. Set up the Management Console
- c. Start the system
- d. Troubleshoot the Launcher and system

9. Analyzing system performance

- a. Use IBM Transformation Extender Management Console to display system statistics and status
- b. Use the IBM Transformation Extender Launcher Monitor to display the map execution summary and take snapshots of Launcher activity
- c. Display map execution log using IBM Transformation Extender Snapshot Viewer

10. Using Resource Registry with the Launcher

- a. Use the Resource Registry to define resource alias names and values for virtual servers

11. Advanced concepts

- a. Use Event Coordination to control system triggering
- b. Use wildcards in input and output cards for variable names
- c. Modify map settings to allow multiple map instances
- d. Configure the Launcher to allow separate system process execution and control

12. Launcher utility commands

- a. Use the System deploy utility to build map and deploy Launcher systems from a command line
- b. Use Launcher control utility commands for UNIX and Windows environments

COURSE: ITX System Management (2 days)

Purpose: The IBM Transformation Extender System Management course is designed to provide a fundamental understanding and hands-on experience using WebSphere Transformation Extender to deploy, monitor, manage, and troubleshoot WebSphere Transformation Extender systems running in a production environment with the Launcher.

Audience: The intended audience is composed of operators and system administrators who need to manage ITX Launcher systems in a production environment.

Prerequisites: Students must possess a basic understanding of Windows operating systems and services. Students must have attended the IBM Transformation Extender Fundamentals course and the IBM Transformation Extender System Design course.

Objectives: After completing this course, you should be able to:

- Describe the WebSphere Transformation Extender Launcher architecture
- Use the Launcher management tools and log files to configure, monitor, and manage the Launcher on Windows and UNIX-based platforms
- Use Launcher administrator tools and commands to capture system runtime details

1. Launcher fundamentals

- a. Describe the IBM Transformation Extender components
- b. Describe the Launcher architecture
- c. Verify application port availability for the Launcher and Launcher processes
- d. Use the Launcher Administration application to configure the Launcher

2. Launcher management

- a. Start and stop the Launcher in a Windows environment
- b. Use the Management Console to enable Debug
- c. Use the Launcher log files generated by the Management Console to identify problems

3. Preparing to deploy to production

- a. Describe the tasks that should be completed before systems are deployed to the production environment
- b. Monitor system performance

4. Performance tuning

- a. Describe IBM Transformation Extender initialization file parameters
- b. Modify IBM Transformation Extender initialization file parameters to improve Launcher performance

5. Deploying to production

- a. Deploy to a production server
- b. Configure the Launcher for production

- c. Remotely manage the Launcher using the Management Console and Launcher Monitor
- d. Use the IBM Transformation Extender utility commands to control and acquire Launcher status

6. Managing Launcher systems from a command line

- a. Use the IBM Transformation Extender Utility Commands to start, stop, pause and restart the IBM Transformation Extender in a Windows environment
- b. Use the IBM Transformation Extender Utility Commands to start, stop, pause and restart the IBM Transformation Extender in a UNIX environment

7. Troubleshooting

- a. Describe change control options for managing system source
- b. Use log files, utility commands, and scripts to isolate problems in a running system