Module 16

Introduction to BEA Ed.Lab

At the end of this module, you will be able to:

✓ Install a new course using Ed.Lab

✓ Execute commands against Ed.Lab

✓ Explain the components that implement Ed.Lab

Road Map

1. **Getting Started with Ed.Lab**

    – Install a course

    – Set up a lab exercise

    – Continue a previous lab exercise

    – Continue your current work

2. Ed.Lab Internals

What Is Ed.Lab?

▶ Ed.Lab:

- Is an extensible Java framework used to install and execute labs and demos within BEA courses

- Allows students and instructors to concentrate on a lab's learning objectives, not on its infrastructure

- Maximizes a student's time to complete an exercise

- Avoids course-specific setup instructions and scripts

- Supports both Windows and Linux

▶ Ed.Lab consists of the following components:

- Course Installer

- Lab Framework

Install a Course

▶ Execute the native script `install.cmd` or `install.sh` on the student media provided with the course, such as a CD.

▶ Complete the following steps:

1. Supply a path to which the course will be installed.

2. Provide the path of your BEA HOME, such as `c:\bea`.

3. Provide the path of your WEBLOGIC 9.0 HOME, such as `c:\bea\weblogic90`.

4. Provide the path of an existing Java 5 JDK, such as `c:\bea\jdk150_03`.

Installation Steps

▶ The Course Installer performs the following steps:

1. Extract the course contents to the supplied path.

2. Update variables for BEA, WEBLOGIC 9.0, and JDK paths within the following files:

   ▶ `bin/prompt.cmd`

   ▶ `bin/prompt.sh`

   ▶ `edlab/edlab.properties`

   ▶ `*/pointbase.ini`

3. Send the command `setup_course` to the Lab Framework for any custom installation steps.

▶ A log file is also generated for debugging purposes.

Reference Variables

▶ The following variables are used throughout the course to refer to common files or directories.

Variable	Description	Examples
BEA_HOME	BEA root installation directory	`C:\bea` `D:\bea90`
WEBLOGIC_HOME	WebLogic product installation directory	`C:\bea\weblogic90` `D:\bea90\weblogic90`
JAVA_HOME	Java 5 JDK installation directory	`C:\bea\jdk150_03` `D:\java\jdk150_03`
STUDENT	Course installation directory	`C:\student\mycourse` `D:\mycourse`
CURRENT_LAB	Resource directory associated with the lab you are currently working on	`<STUDENT>\labs\lab01` `<STUDENT>\labs\lab11`

Lab Framework

▶ The Ed.Lab Lab Framework automates the initialization of resources and infrastructure for a specific lab:

  – Create and update WebLogic Server domains

  – Launch domain administration servers

  – Launch database servers

  – Update databases

  – Update application source

  – Deploy J2EE applications

▶ The Lab Framework maintains a backup copy of each lab to allow students to restore previous work.

Start a New Lab

▶ To begin work on an exercise, perform the following steps:

1. Execute `bin/prompt.cmd` (or `.sh`).

2. Change directories to the location associated with the lab: `cd lab01`.

3. Type the following command: `ant setup_exercise`.

▶ Lab resources, such as application source, will be placed under the `work` directory.

▶ Any work from a prior lab will be backed up.

Continue an Existing Lab

▶ To continue working on a previously started exercise, perform the following steps:

1. Execute `bin/prompt.cmd` (or `.sh`).

2. Change directories to the location associated with the lab: `cd lab01`.

3. Type the following command: `ant continue_exercise`.

▶ Backup resources will be restored to the `work` directory.

Start a Lab Solution

► To begin working on the complete solution for a given lab, perform the following steps:

1. Execute `bin/prompt.cmd` (or `.sh`).

2. Change directories to the location associated with the lab: `cd lab01`.

3. Type the following command: `ant setup_solution`.

Continue Your Current Work

► While working on a lab, some situation may result in the shut down of your servers or databases, such as a machine restart.

► To continue working on the current exercise, perform the following steps:

1. Execute `bin/prompt.cmd` (or `.sh`).

2. Change directories to `work`:
   `cd ../work`.

3. Type the following command:
   `ant continue_work`.

Start a New Demo

▶ To initialize and begin presenting a course demonstration, perform the following steps:

1. Execute `bin/prompt.cmd` (or `.sh`).

2. Change directories to the location associated with the demo:
   `cd ../demos/demo01`.

3. Type the following command:
   `ant setup_demo`.

Additional Documentation

▶ You can find additional Ed.Lab documentation at the location `<STUDENT>/edlab/docs.`

Section Review

In this section, we learned how to:

- ✓ Explain the capabilities of Ed.Lab

- ✓ Install a course

- ✓ Start a new lab

- ✓ Continue an existing lab

Road Map

1. Getting Started with Ed.Lab

2. **Ed.Lab Internals (Optional)**
   - Course Layout
   - Dispatcher
   - `edlab.properties`

Course Layout

▶ The layout of a typical Ed.Lab course is shown below:

Directory Tree	Description
student	
mycourse	Course root location, selected during installation
bin	Native scripts, such as `prompt.cmd`
demos	Resources for any course demonstrations
edlab	Lab Framework scripts and configuration files
install	Resources to initialize during course installation
labs	Resources for any course labs
work	Workspace containing files for the current lab or demo

Lab Framework Implementation
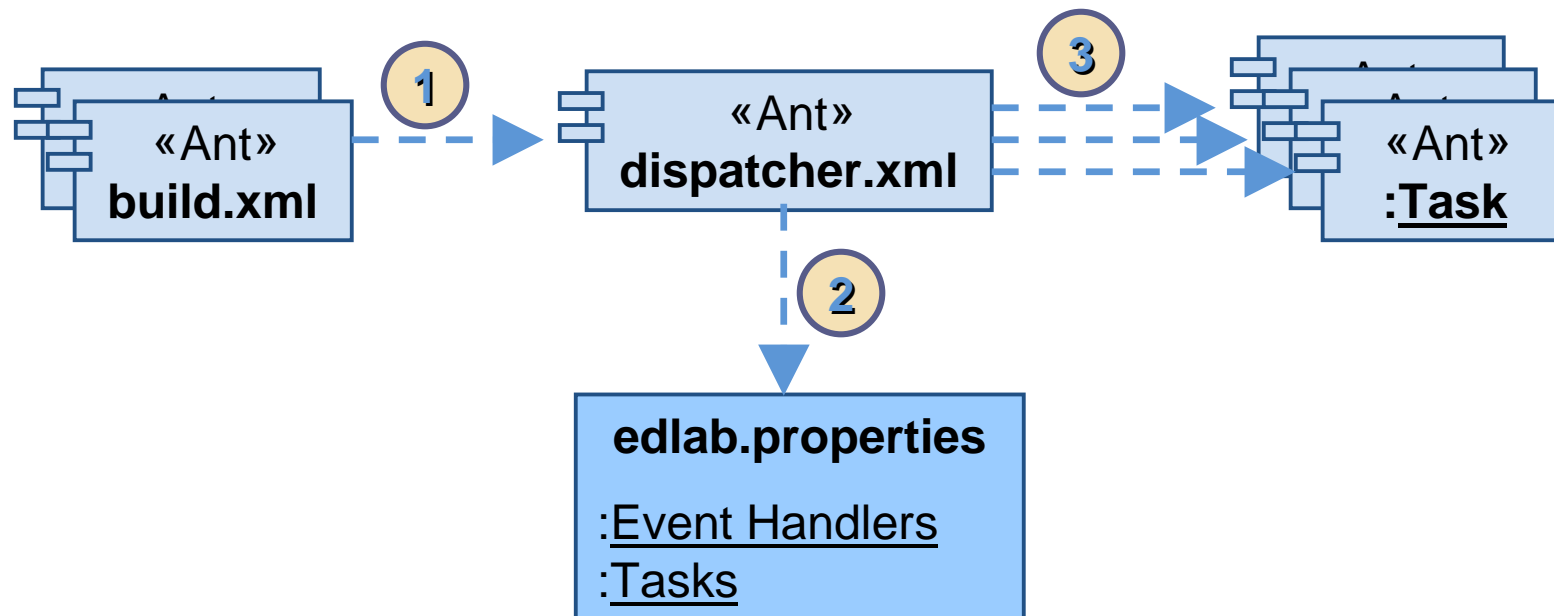
Lab Framework Layout

| Directory Tree	Description
⊟ 📁 edlab	**Lab Framework root directory**
⊞ 📁 common	**Resources (apps, etc.) common to all labs**
⊞ 📁 docs	**Ed.Lab documentation**
⊞ 📁 logs	**Lab Framework log files**
⊟ 📁 tasks	
⊞ 📁 app	**Application init and deployment scripts**
⊞ 📁 db	**Database start and update scripts**
⊞ 📁 domain	**Domain initialization and update scripts**
⊞ 📁 misc	**Miscellaneous scripts**
⊟ 📁 server	
📄 start_server.xml	**Server start script**
📄 update_server.xml	**Server update script**
📄 currentwork.properties	**Lists the current lab**
📄 dispatcher.xml	**Dispatcher Ant script**
📄 edlab.properties	**Default Lab Framework configuration**

Lab Framework Configuration…

▶ The *Dispatcher* determines how to process a request, or *Event*, using the file `edlab.properties`.

▶ You can also configure individual *Task* scripts using this file.

▶ A default `edlab.properties` is under the `edlab` course directory.

▶ An individual lab or demo may have its own copy of `edlab.properties`.

▶ Properties in this file will be merged with the default configuration file.
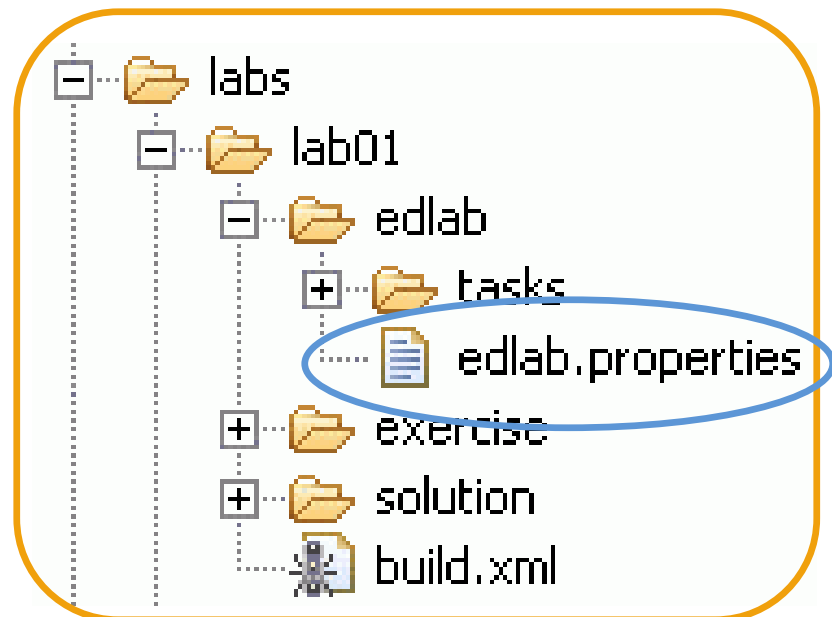
Sample Configuration

```
jdkhome=d\:\\bea90\\jdk150_03
beahome=d\:\\bea90
wlhome=d\:\\bea90\\weblogic90

event.setup_exercise=handler2
event.continue_exercise=handler3

event.setup_exercise.path=exercise

handler.handler2=domainbackupwork,appbackupwork,setcurrentlab,
                 initdomain,startdb,startserver

task.domainbackupwork=tasks/domain/backup_work.xml
task.appbackupwork=tasks/app/backup_work.xml
task.setcurrentlab=tasks/misc/set_current_lab.xml
task.initdomain=tasks/domain/init_domain.xml
task.startdb=tasks/db/start_db.xml

task.startdb.db1.url=jdbc:pointbase:server://localhost:9093/demo
...
```

Initialized by Installer

Assign handlers to events

Assign tasks to handlers

Register task Ant scripts

1
011

Section Review

In this section, we learned how to:

✓ Describe the components that make up the Lab Framework

✓ Understand the purpose and contents of `edlab.properties`

© 2006 BEA Systems, Inc.

Module Review

In this module, we learned how to:

✓ Explain the features of BEA Ed.Lab

✓ Use the Course Installer

✓ Use the Lab Framework