

AWIPS-II

Software Development

Training Guidelines

OHD/HSEB
03/19/2008

Table of Contents

1.0 Background	2
2.0 Technical Approach	3
2.1 Training Focus Areas	3
2.2 Training Resource Types	4
2.3 General Approach	4
2.4 Individualized Approach	5
Appendix A. On-Line Classes	6
A.1 Commerce Learning Center	6
A.2 Other On-line Class Resources	8
Appendix B. Electronic Documentation	9
B.1 OHD Shared Drive Documentation	9
B.2 T08 Design Discussion Modules	10
B.3 TOT1 Training Modules	11
B.4 Specific Information from Software CTR PIP	11
Appendix C. Web Sites	14
Appendix D. Printed Documentation	16
Appendix E. Source Code Analysis	17

DRAFT

2.0 Technical Approach

The training plan must consider the media or manner in which training is delivered and must consider the topics on which different staff should be trained. This section discusses these two issues and presents general information.

Because of the evolving nature of the priorities of AWIPS-II development activities, it is expected that this approach will itself evolve as experience with the AWIPS-II training process is gained.

2.1 Training Focus Areas

The skills and knowledge required for developers to understand and develop software in the AWIPS-II environment is considerable. Because of the sheer volume of topics involved, it is unrealistic to expect that everyone become an “expert” in every topic. Therefore, training should be conducted on only those components relevant to an employee’s role in the development process. Suggested training needs for certain roles are presented below.

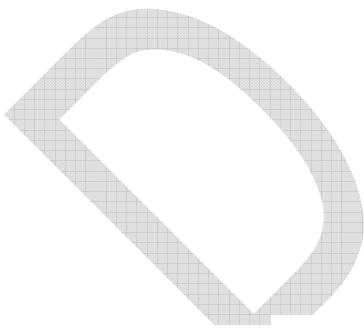
Some topics may be so general or pervasive that most or all developers need to understand the topic. Other topics may be specific to one feature of the AWIPS-II environment, such as graphical interfaces, so that only developers working on software providing these features need be knowledgeable.

This breakdown of suggested focus areas (i.e. developer roles) is summarized immediately below. It is expected that further refinement will be made in the future.

- a) All: Software Product Improvement Plan, XML
- b) Application Developers:
 - o Languages: Java, JavaScript
 - o Integrated Development Environment: ADE, Eclipse,
 - o Configuration Management: Subversion
- c) Application Integrators: JMS (ActiveMQ), ESB (Mule), TomCat, MicroEngine, JibX
- d) Database Access Development: Java DAO, Hibernate, Mule+Spring, Middlegen, Apache Derby; EDEX, HDF5
- e) Graphical Interface Development: CAVE, Eclipse RCP, SWT+JFace, SVG
- f) Configuration Management Lead: Eclipse, ANT, Subversion

Note that while most of these topics are COTS components, a few of these are special customized components assembled and/or tailored to the AWIPS needs by the Raytheon staff. These custom AWIPS-specific components include:

- ADE (AWIPS Development Environment)
- CAVE (Common AWIPS Visualization Environment)



2.4 Individualized Approach

The approach to training used by each person will be different. It will be dictated not just by what knowledge and experience the person already has, but also their expected role in AWIPS-II development.

Regardless, the recommended approach is to balance readings and courses with hands-on development. It is not possible to be proficient in AWIPS-II development without actually doing development, initially at the basic level and then progressing to more sophisticated tasks. The ultimate validation of a successful training plan for an individual is their demonstrated ability to complete tangible software development tasks that improve the baseline system. Understanding of the baseline system is necessary and valuable too, but it is only as relevant as to how that understanding can directly contribute to successful development task completion.

Training depends heavily on the expected role of the developer. The topics associated with the developer should be learned at a level which ensures proficiency in development tasks. The resources available for each topic should be investigated to determine which approach to take and at what level to begin the training. Some staff will be able to skip introductory level training, while others will need to start at the beginning.

Each employee should develop a general plan for the topics they think they need to learn and for the resources they should use to develop the plan. A strong degree of personal responsibility and independent time management is expected of all employees. Employees should document their planned topics and resources in a summary fashion and share them with the HSEB Project Area Leaders (PALs).

Appendix A. On-Line Classes

A.1 Commerce Learning Center

The following classes are available on the Department of Commerce Learning Center which can be accessed via:

- <http://www.wfm.noaa.gov/e-learning/index.html> (click on Commerce Learning Center @NOAA)
- <https://doc.learn.com/login.asp>

Before taking a class, review the course description and check the pre-requisites. The estimated number of minutes from the Learning Center are given in parentheses after the course name.

1) Java Introduction:

- a) Java Programming with J2SE 5: Getting Started with Java (120)
- b) Creating Classes in Java (novice) (120)
- c) Working with Classes in Java (novice) (160)
- d) Operators and Flow Control in Java (novice) (110)
- e) Exception Handling and Assertions (novice) (100)
- f) Java Utilities (novice) (190)

2) Java J2SE5 Programming:

- a) Java Programming with J2SE 5: Creating Classes in Java (175)
- b) Java Programming with J2SE 5: Working with Classes in Java (155)
- c) Java Programming with J2SE 5: Operators and Flow Control in Java (135)
- d) Java Programming with J2SE 5: Java Utilities (220)
- e) Java Programming with J2SE 5: Exception Handling and Assertions (130)
- f) Java Programming with J2SE 5: Basic GUI Development in Java (135)
- g) Java Programming with J2SE 5: Generics and Annotations (180)
- h) Java Programming with J2SE 5: Java Applets (115)
- i) Java Programming with J2SE 5: Java I/O (105)
- j) Java Programming with J2SE 5: Reference Types and Threading (140)
- k) Using Metadata in J2SE 5.0 (120)
- l) J2SE 5.0 Language Features (185)
- m) Library and Class Changes in J2SE 5.0 (220)

3) Java Scripting

- a) JavaScript: Language Basics (300)
- b) JavaScript: Scripting (160)

4) XML Coding:

- a) XML Language Basics (240)
 - b) XML APIs (100)
 - c) Style Sheets and Links (315)
 - d) Structuring XML with Schemas (85)
- 5) J2EE Programming:
- a) Developing J2EE Clients (120)
 - b) Packaging and Deploying J2EE Applications (100)
 - c) The J2EE Connector Architecture (120)
 - d) J2EE Architecture (260)
- 6) Web Services:
- a) Java Technologies For Web Services (160)
 - b) Java Web Service Clients (100)
 - c) Web Services Technologies (180)
 - d) Emerging Web Service Standards (140)
 - e) Web Services and Service-oriented Architecture (110)
 - f) Web Services Security (90)
- 7) Object Oriented Design:
- a) Object-Oriented Analysis and Design with UML: Fundamentals (160)
 - b) Object-Oriented Analysis and Design with UML: Analyzing the System (300)
 - c) Object-Oriented Analysis and Design with UML: Design and Implementation Issues (160)
 - d) Object-Oriented Analysis and Design with UML: Designing the System (180)
 - e) Object-Oriented Analysis and Design with UML: Exploring System Behavior (120)
 - f) Object-Oriented Analysis and Design with UML: Finding Classes (220)
 - g) Object-Oriented Analysis and Design with UML: System and Object Behavior (240)
 - h) Object-oriented Coding Best Practices (140)
- 8) Miscellaneous SOA/Java topics:
- a) Developing JSPs (140)
 - b) Developing Servlets (280)
 - c) Java 2 Security Features (180)
 - d) Java Database Connectivity (320)
 - e) RMI and the Java Message Service (310)
 - f) Using WSDL (160)
 - g) Using UDDI (110)
 - h) Using SOAP (110)

A.2 Other On-line Class Resources

This links to a NOAA site that is a Java resource list put together by FSL which includes PowerPoint presentations on many Java topics.

<http://www-ad.fsl.noaa.gov/ac/javazone/ClassMaterial.html>

DRAFT

Appendix B. Electronic Documentation

B.1 OHD Shared Drive Documentation

Assorted AWIPS documents are available on the OHD S: drive folders. These documents are contained in folders located under the following folder:

S:\OHD-11\AWIPS\AWIPS_II

The folders contain documents associated with specific AWIPS-II task orders, including:

- TO4_Docs
- TO5_Docs
- TO6_Docs
- TO7_Docs
- TO8_Docs
- TOT1_Docs

There is also a MiscDocs folder for miscellaneous documents. Another folder under TO6_Docs contains the source code and associated files for the first “official” delivery of the AWIPS ADE (AWIPS Development Environment) in the folder TO6_ADE1.0_release. The software delivered with TO8 is currently not maintained on the S: drive.

A few documents in these folders are worth special discussion:

- The file “AWP.PLN.SwPIP_03.00.PDF” in the MiscDocs folder is the Continuous Technology Refresh Product Improvement Plan (PIP), which describes the overall plan for the migration of software into the AWIPS-II environment. This document has a few portions which are excerpted below in the subsequent section because of their relevance to AWIPS-II training.
- The file “AWIPS II Migration Approach Final 22 June 2007.ppt” in the T07_Docs folder provides an overview presentation on the strategy for the migration into the new AWIPS-II architecture. Although some of this information is already outdated, this presentation provides helpful information.
- The folders associated with each of the task orders have powerpoint (.ppt) or adobe (.ppt) documents that were part of the developer training or support information supplied by Raytheon. The most important of these folders are those for TO8 and TOT1. A separate task order (TOT1) for training has been contracted and in February 2008, the first deliverables for this TO were received. These TOT1 documents are up-to-date with TO8, and are expected to be updated with “regular” TO delivery. Both of these folders are further discussed immediately below.

B.2 T08 Design Discussion Modules

The folder for T08

S:\OHD-11\AWIPS\AWIPS_II\T08_Docs

contains a document with all the (12) available training modules in one file:

- AWP.TRG.SWCTRT08.ADE-CAVE_All.pdf

[This document is also provided in a subfolders for task order 8 (S:\OHD-11\AWIPS\AWIPS_II\T08_Docs\Delivery CD\TechDocuments)]

The training modules are the primary source of specific design information on the AWIPS-II implementation of the SOA technologies. The modules in this aggregate document are:

- 1: Architecture [T04}
- 2: Installation, Build, and Regression Test {T04}
- 3: MicroEngine Scripting {T06}
- 4: Data Type Plug-In {T05}
- 5: Service Oriented Architecture {T04}
- 6: CAVE-Underlying Framework and Rendering {T04}
- 7: CAVE-User Interface {T04}
- 8: CAVE Visualization Plug-Ins {T04}
- 9: Installation/Deployment {T06}
- 10: CAVE Menu Creation {T06}
- 11: Localization {T06}
- 12: T08 ADE 1.0 Developer Updates {T08}

The TO noted in braces indicate when the module was last updated, prior to T08 except for module 12 which was first introduced in T08, to the best of our knowledge.

B.3 TOT1 Training Modules

Specific how-to information on certain important AWIPS-II development topics are provided in the folder:

S:\OHD-11\AWIPS\AWIPS_II\TOT1_Docs

The four documents are:

- AWP.TRG.SWCTRTOT1-01_UEngScript.pdf - Writing a MicroEngine Script
- AWP.TRG.SWCTRTOT1-02_UEngTask.pdf - Creating a MicroEngine Task
- AWP.TRG.SWCTRTOT1-03_EngTaskList.pdf - MicroEngine Tasks
- AWP.TRG.SWCTRTOT1-04_CodeExs.pdf - Code Examples

B.4 Specific Information from Software CTR PIP

There is information in the “Software Continuous Technology Refresh Product Improvement Plan, Version 3; Document No. AWP.PLN.SWPIP-03.00”; dated 22 June 2007, which is highly relevant to learning AWIPS-II. The pertinent sections regarding AWIPS-II software development are:

- Section 3. AWIPS II Architecture
- Section 6. AWIPS Software Migration
- Section 8. Training

Specific sub-sections within these sections are excerpted below:

Table 3-1. Open Source Project Usage in AWIPS II

Function Open Source Project
Software Build: ANT
Configuration Management (CM): Subversion + Trac
Enterprise Service Bus (ESB): Mule + Spring
Integrated Development Engineering (IDE): Eclipse
Logging: Log4j
Java Messaging Service (JMS) Broker: ActiveMQ
XML Reader: Commons Digester

Extract: Section 6.3:

FEWS and AWIPS II

Question: We've heard that "FEWS" will be included in the first release of AWIPS II. Is that true?

Response: FEWS ("Flood Early Warning System") is being pursued by the Hydro community as a "replacement" for RFS. Raytheon will execute a study task order to determine the feasibility of and approach to integrating FEWS and AWIPS II. No commitment has been made for or against FEWS inclusion in AWIPS II R1.0. However, it is unlikely that it will be included in AWIPS II R1.0 as it would require many elements to "fall into place" including additional functionality added to FEWS. It would also require agreements for re-baselining the SW CTR Project between NWS and Raytheon, and this might require a re-plan of TO sequencing (the main hydro TO is 10 but other tasks are being developed earlier). So, it is highly unlikely to occur.

Extract: 6.4 Documentation

Documentation provided with ADE 1.0 will be updated as needed throughout the migration (e.g., Javadoc, Tech Brief). Current documentation that Raytheon is responsible for also includes the AWIPS User Manual (UM), Systems Manager's Manual (SMM), System/Subsystem Design Description (SSDD), and Release Notes. This documentation will be redlined in appropriate sections with each task order, and the updated versions will be delivered with AWIPS II Release 1.0.

Updating other related documentation such as the AWIPS Integration Framework Manual (AIFM) will be the responsibility of the NWS.

Extract: 8.2 Developer Training Syllabus

The developer training syllabus is modular and supports the approach discussed in the previous section. It includes a “Foundation” course on Service Oriented Architectures; the NWS is contracting a third party to deliver this “Foundation” course.

The training syllabus is as follows:

- Foundation Course
- Script Development
 - Using script language
 - Extending script language
- Cave Plug-ins
 - Creating menus in CAVE
 - Using the localization pattern within CAVE
- SOA Plug-ins
 - Data ingest
 - Using Data access
 - Micro Engine extensions
 - Configuring Mule end points
- Using the Localization Pattern
 - Extending the meteo library

Appendix C. Web Sites

The NWS Training Center in Kansas City is facilitating training for AWIPS-II developers. They have established a web page listing resources for developers at:

http://www.nwstc.noaa.gov/AWIPS/ADE/ADE_resources.html

This web page has links to other web pages and supporting information. For convenience, many of these web pages are also listed in the table below.

ID	Website	Description
1	http://en.wikipedia.org/wiki/Service-oriented_architecture	SOA
2	http://service-architecture.com/	SOA
3	http://www.eclipse.org	Eclipse (IDE – Integrated Development Environment)
4	http://www.cs.umd.edu/class/fall2004/cmsc131/Eclipse Tutorial or https://eclipse-tutorial.dev.java.net/	Eclipse primer tutorials
6	http://mule.codehaus.org	Mule, Spring (ESB – Enterprise Service Buss)
8	http://www.activemq.org	ActiveMQ (Java Messaging Service (JMS) Broker)
9	http://xml.apache.org	Apache (web server)
10	http://java.sun.com/xml/tutorial_intro.html or http://www.w3schools.com/xml/default.asp or http://www.devshed.com/c/b/XML/	XML Internet Tutorial
11	http://subversion.tigris.org	Subversion CM
13	http://www.postgresql.org	POSTGRE RDBMS
14	http://www.jboss.org/products/jbosscache	JBossCache
15	http://www.mozilla.org/rhino	Rhino JS scripting
16	http://lucene.apache.org/java/docs	Reverse Indexing
17	http://jakarta.apache.org/commons/digester	Common_digester for XML configuration and scripting
18	http://directory.apache.org/subprojects/network/index.html	MINA
19	http://xml.apache.org/batik	Scalable Vector Graphic;

ID	Website	Description
	or http://www.w3schools.com/svg/default.asp	Batik SVG tools
20	http://hibernate.org	Hibernate relational to Object mapping
22	News://wiki.eclipse.org/index.php/Rich_Client_Platform	Eclipse rich Platform (RCP)
23	http://www-ad.fsl.noaa.gov/ac/javazone/ClassMaterial.html	Java – FSL powerpoint
24	http://www.programmingtutorials.com/java.aspx or http://java.sun.com/docs/books/tutorial/	Java Internet Tutorial
25	http://java.sun.com/j2se/1.5.0/docs/guide/jmx/tutorial/tutorialTOC.html	Java Internet Tutorial This is a fairly comprehensive tutorial on Java Management Extensions (Requires loading JMX env)
26	http://javascript.about.com/od/reference/a/js101.htm or http://www.programmingtutorials.com/javascript.aspx or http://javascript.about.com/od/learnjavascript/Learn_to_Program_with_Javascript.htm or JavaScript by example	JavaScript Internet Tutorial
27	http://www.jython.org/Project/index.html	Jython reference

Appendix D. Printed Documentation

There are countless books describing the technology that is contained within AWIPS-II. Listed in this section are books that have been purchased and are available within OHD/HSEB.

Software Development Methods:

- Implementing Lean Software Development; by Mary Poppendieck, Tom Poppendieck
- Agile and Iterative Development: A Manager's Guide; by Craig Larman
- Behind Closed Doors: Secrets of Great Management (Pragmatic Programmers); by Johanna Rothman, Esther Derby
- The Pragmatic Programmer: From JourneyMan to Master; by Andrew Hunt, David Thomas

Applied Software Development Methods:

- Pragmatic Project Automation: How to Build, Deploy, and Monitor Java Apps; by Mike Clark
- Pragmatic Unit Testing in Java with Junit; by Andrew Hunt, David Thomas
- Pragmatic Version Control: Using Subversion; by Mike Mason
- Agile Java Development with Spring, Hibernate and Eclipse; by Anil Hemrajani
- Spring in Action; by Craig Walls, Ryan Breidenbach

User Interface:

- The Definitive Guide to SWT and JFACE; by Robert Harris, Rob Warner

Enterprise Services:

- Enterprise Service Bus: Theory in Practice; by David Chappell

Eclipse:

- Eclipse: Building Commercial Quality Plug-ins; by Eric Clayberg, Dan Rubel
- Eclipse Rich Client Platform: Designing, Coding, and Packaging; by Jeff McAffer, Jean-Michel Lemieux
- Eclipse IDE Pocket Guide; by Ed Burnette

Java:

- Head First Java; By Kathy Sierra, Bert Bates
- Java Persistence with Hibernate; by Christian Bauer, Gavin King
- Java in a Nutshell, 5th Edition; by David Flanagan.

Appendix E. Source Code Analysis

This section is reserved for information derived from “reverse-engineering” analysis of the actual source code. As the documentation can never fully describe all aspects of the system operations, it is expected that review of the source code may sometimes be necessary. Lessons learned from this review can be added to this section as appropriate.

DRAFT