**Version DESCRIPTION DOCUMENT**

**For**

**META PHase 1B SEPT 2011 Demo**

**Software Release**

Contract No.: HR0011-10-C-0108

**Sept 8, 2011**

**Prepared for:**

Defense Advanced Research Projects Agency  
Tactical Technology Office (TTO)

**Prepared by:**

BAE Systems

6 New England Executive Park

Burlington, MA 01803

(781) 273-3388

# Table Of Contents

Table Of Contents i

1 Introduction 1

1.1 System Overview 1

2 Version Description 2

2.1 Inventory of Materials Released 2

2.2 Inventory of CSCI ConTENTS 2

2.3 BUILD / Development TOOLS 2

2.4 INSTALLATION INSTRUCTIONS 3

2.4.1 Extracting the software from the CD 3

2.4.2 Preparing the SpringSource Integrated Development Environment 3

APPENDIX A – META Development Repository Contents 5

# Introduction

## System Overview

The goal of the META program is to reduce development cycle time for complex cyber physical systems (particularly aerospace and defense systems such as aircraft, rotocraft, and ground vehicles) by a factor of 5x over current cycle times. The approach to this goal is to apply innovative technology employing model-based approaches for revolutionizing design and verification processes currently used throughout the industry. The objective of this process is to apply innovative technical approaches and then define and develop the new infrastructure needed and make this technology available to the industry.

# Version Description

## Inventory of Materials Released

**Table 3.1-1 Released Media**

|  |  |  |
| --- | --- | --- |
| **Media** | **Identity** | **Description of Content** |
| CDROM | META trunk revision #1695 | META software development repository as of SVN version #1695 |

## Inventory of CSCI ConTENTS

A complete top level file listing of release media is provided in Appendix A.

## BUILD / Development TOOLS

The table below describes the external tools required to build and run the META tools. Software that requires a purchased or evaluation license is noted in the Source column.

WinZip is required to extract the contents of the CD with the META source files.

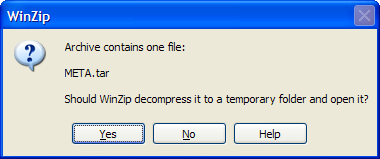
**Table 3.4-1 Build / Development Tools**

|  |  |  |
| --- | --- | --- |
| **Item** | **Purpose** | **Source** |
| ANT V1.8.2 | Build Scripts | http://ant.apache.org/bindownload.cgi |
| CLISP V2.4.4 | Lisp Compiler | http://sourceforge.net/projects/clisp/ |
| Cygwin V1.7.9-1(including Octave but not X11) | Linux environment emulator for Windows | http://www.cygwin.com/ |
| Magic Draw V17 SP 1 | UML Design Toolset | No Magic (licensed) Evaluation version available at: http:// www.magicdraw.com |
| SpringSource Tool Suite V2.6.1 | Java Development Environment | http://www.springsource.com/landing/best-development-tool-enterprise-java |
| Java JDK V6 Update 22 (or higher) | Java Compiler / Runtime Environment | http://www.oracle.com/technetwork/java/javase/downloads/index.html |
| Matlab 7.11.0 R2010(b) | Algorithm development environment | The MathWorks (licensed) http://www.mathworks.com/products/matlab/tryit.html |
| PuTTY V0.6 | Terminal Emulator | http://www.putty.org/ |
| ProEngineer Wildfire V5.0 | Computer Aided Design / Computer Aided Manufacturing | PTC (Licensed) http://www.ptc.com |
| WinZip V1.1 (or greater?) | File compressing utility | Licensed. http://www.winzip.com/ |

## INSTALLATION INSTRUCTIONS

### Extracting the software from the CD

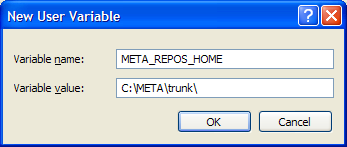
The following instructions can be used to extract the software from the delivery CD.

1. Create a C:\META folder on the computer
2. Insert the CD into the CD drive of the computer
3. Navigate to the CD drives folder and right click the “trunk.zip” file and select “WinZip -> Extract to..”. This will produce the following dialog.
4. Select the “No” button and the WinZip Extract dialog will appear.
5. Navigate to the C:\META folder and select the “Extract” button. The contents will be extracted the META folder and the dialog will close.
6. Verify the C:\META\trunk folder exists.

### Preparing the SpringSource Integrated Development Environment

The following instructions can be used to create a project within the SpringSource Tool Suite on the Windows platform. This is required before building the META tools.

1. Create the folder “C:\tmp” if it does not already exist,
2. Set the environment variable “META\_REPOS\_HOME” to the location of the META source (C:\META\trunk\ ). Environment variables can be set from the Start Menu’s Control Panel -> System icon. When the dialog opens, select the “Advanced” tab and then select the “Environment Variables” button. When the Environment Variables dialog opens, select the “New” button under the User Variables section. Populate the dialog as shown below. Select OK on all 3 dialogs to complete the action.



1. Open a Command Prompt window and “cd” to “C:\META\trunk\ArrowManualArtifacts” folder of the software release. Enter the following command to add Java archives to the Maven repository.

*mvn install*

1. Startup the Spring Source Tool suite from the Start menu. When prompted, set the workspace to a desired location (E.g., C:\META\_workspace).
2. When the Welcome window displays, close it.
3. Navigate to Windows->Preferences->Maven->Installations and Add an installation for apache-maven-3.0.3. Note, by default STS uses the Maven version it comes packaged with. META requires version 3.0.3 be used by STS.
4. Navigate to Windows->Preferences->Java->Installed JREs. If jdk1.6.0\_22 does not exist, add it and select it. Note: Java installations can be found under C:\Program Files\Java.
5. Select File->Import, select Maven->Existing Maven Projects, then click Next,
6. Navigate to C:\META\trunk folder and select subfolder “arrow-mvn-init”, click OK, then Finish.
7. Using the same procedure in the previous two steps, import subfolder “arrow-mvn-all”
8. In the STS Package Explorer window, right click “arrow-mvn-init” then select Run As->Maven Install.
9. Repeat the above step for “arrow-mvn-all”. The install may take several minutes to complete.

# APPENDIX A – META Development Repository Contents

Each CD contains 1 file name “trunk.zip”. This file extracts into a folder named “trunk” containing the following files and subfolders:

**build\_and\_run\_Arrow.sh**

Shell script for building and running Arrow Web Services.

**copyright.txt**

Contains the copyright notice for the content of the directory.

**readme.txt**

Quick start directions on getting set up and running.

**./AmilExtern**

This folder contains the current META development baseline for the AMIL external library functionality.

**./AmilLib**

This folder contains the current META development baseline for the AMIL library functionality.

**./AmilProlog**

This folder contains the current META development baseline for the ProLog AMIL query utility.

**./ArrowManualArtifacts**

This folder contains java archives of 3rd party software not supported by Maven automatic import.

**./ArrowWebServices**

This folder contains the current META development baseline for the ARRoW web services functionality.

**./ComponentModelLibrary**

This folder contains the current META development baseline for the ARRoW CML Server.

**./DebugSysMLPlugin**

This folder contains source code for debugging the SysML Magic Draw Plugin.

**./GalileoWrapper**

This folder contains the current META development baseline for the Galileo Wrapper functionality.

**./MIT**

This folder contains qualitative model source code and data files to support the MIT Reach Set Analysis tool.

**./Meta-Windchill**

This folder contains obsolete functionality used to support previous demos.

**./QRG**

This folder contains an installation of the ZGraph library suite which provides functionality for displaying and editing graphs.

**./Servers**

This folder contains a build environment installation of the Apache Tomcat JavaServlet and JavaServer Pages development suite.

.**/TestData**

This folder is currently empty.

**./arrow-mvn-all**

This folder contains a SpringSource project definition with a Maven Project Object Models (POM) file defining all other projects that make up the META development suite.

**./arrow-mvn-init**

This folder contains a SpringSource project definition with the top level Maven Project Object Models (POM) file for the META development suite.

**./dashboard**

This folder contains the current META development baseline for the ARRoW dashboard.

**./docs**

This folder contains the current META development baseline for the ARRoW documentation.

**./draw**

This folder contains PARC software.

**./envisioner**

This folder contains qualitative model source code and data files to support the Parc Envisioner tool.

**./galileo**

This folder contains experimental code used to test Galileo prototype functionality.

**./json**

This folder contains the a version of the json library with bug fixes for META.

**./lang**

This folder contains the current baseline of the AMIL Graph source code.

**./mdplugin**

This folder contains the Magic Draw plugin which is used to publish ARRoW requirements from Magic Draw into AMIL.

**./metrics**

This folder contains the current baseline of the metrics framework.

**./models**

This folder contains a repository of miscellaneous model data files and simulation executables from various modeling tools.

**./ontologies**

This folder contains the current META development baseline for the ARRoW ontology development.

**./ontology**

This directory contains a first pass at creating an Arrow

ontology.

**./proe-plugin**

This folder contains an obsolete PRO-E support tool.

**./proe-webservice**

This folder contains the source for a PRO-E server that supports geometrical queries.

**./proe-ws-client**

This folder contains an obsolete PRO-E Client used in previous demos.

**./semneo**

This directory contains testing software for ontology work.

**./tinkerpop**

This directory contains supporting software for ontology work.

**./tools**

This folder contains a release of the Multi-Parametric Toolbox for MatLab (./mpt) and a release of the Visual Swing for Eclipse plugin(./vs4e). MPT is utilized by the Reach Set Analysis MatLab scripts. VS4E is used by the Magic Draw Plugin.

**./tuProlog**

This folder contains the current a test bed baseline used in support of the Galileo functionality. It contains build environment installations of JavaAssist, a class library for editing byte codes in Java and TuProlog a Java base light weight ProLog engine.

**./uml**

This folder contains the active Magic Draw development documents for the current baseline.

**./videos**

This folder contains videos used in the demos.

**./webflow\_sandbox**

This folder contains experimental development work.

**./webservice\_sandbox**

This folder contains experimental development work.