



# Meister and Mojo 7.0 Feature Summary

A technical review of the new features of OpenMake Meister and Mojo 7.0

# Table of Contents

---

<i>Table of Contents</i> .....	<i>ii</i>
<i>Openmake Meister and Mojo 7.0 Introduction</i> .....	<i>1</i>
<i>Openmake Meister and Mojo Workflow Features</i> .....	<i>2</i>
<b>Lifecycle Management and Control, Centralization and Collaboration</b> .....	<b>2</b>
<b>ALM Tool Integration</b> .....	<b>3</b>
<b>Enhanced Build Metrics Reporting</b> .....	<b>4</b>
Bill of Material Reports .....	4
Build Difference Report .....	4
Workflow Summary .....	6
Workflow Activity Detail.....	7
Build Summary Performance Graph .....	8
<b>IDE Integration</b> .....	<b>8</b>
<i>Openmake Meister 7.0 Feature Summary</i> .....	<i>9</i>
<b>Workflows to Replace "Projects"</b> .....	<b>9</b>
<b>Public and Private Workflows</b> .....	<b>9</b>
<b>IDE Integration</b> .....	<b>10</b>
<i>Additional Information</i> .....	<i>10</i>
<i>Company Overview</i> .....	<i>10</i>

## **Openmake Meister and Mojo 7.0 Introduction**

---

Openmake 7.0 has evolved into two separate products, Mojo and Meister. Mojo will be offered as a free workflow management solution with the option to purchase support (referred to as Mojo Max) and Meister will be offered as a complete build and workflow solution for purchase. All Workflow scheduling and Application Lifecycle features will be provided as the free download.

Meister will include all of the features of Openmake 6.41, including the Workflow Scheduling and Application Lifecycle components, with a new look and design based on the Eclipse RCP.

Openmake Mojo and Meister will provide workflow functionality for managing the build to release process. Workflow functionality centralizes the control of software development from development to release. Workflow streamlines software delivery activities from development to deployment providing a centralized "dashboard" to developers and production control administrators. Workflow allows diverse IT teams to collaborate and synchronize their build to release activities. It provides centralized access across multiple development projects and development platforms with global reporting and tracking.

With Meister and Mojo Workflow you configure the steps in your lifecycle process to execute in a service flow order from integrating with SCM tools, to executing builds, testing and deployment. It optimizes hardware resources by allowing you to define the steps of your lifecycle process to execute across multiple machines while managing the step dependencies between remote locations. Real-time monitoring and centralized logging for all lifecycle activities are centralized providing an easy to use method of managing the lifecycle process.

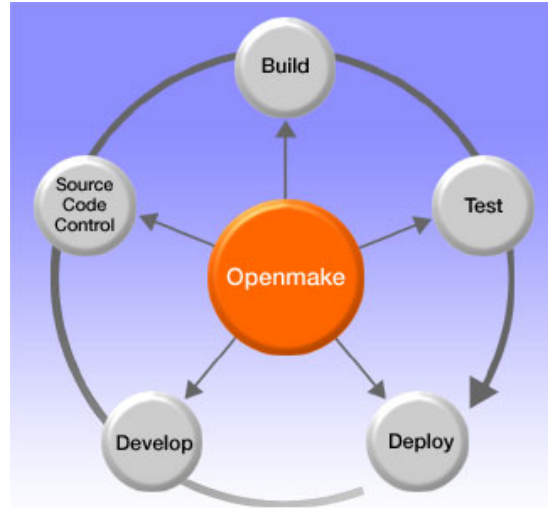
Openmake Meister and Mojo Workflow provides a flexible development platform allowing developers to use existing solutions while still providing a high level of audit and traceability.

## Openmake Meister and Mojo Workflow Features

---

Openmake Meister and Mojo Workflow automates and brings together the people and procedures that normally manage the lifecycle process manually. The advantage of workflow management is in the cohesive approach it brings to standardizing Application Lifecycle processes. The effort of moving source code developed by the application team to executables ready for release to production requires the efforts of individuals from many IT departments. Various tools are used to complete the effort such as versioning tools, compilers, testing solutions and deployment tools. Workflow provides benefits in three main areas –

- Lifecycle management control, centralization and collaboration,
- Interoperability with other application lifecycle tools
- Metrics Reporting



### Lifecycle Management and Control, Centralization and Collaboration

Workflow provides a central location for defining the lifecycle management activities. This is completed by using a container called a Workflow. Workflows are the containers that define the individual steps of the application lifecycle process. A complete Development to Release process can be defined within a single Workflow. A Workflow will contain one to many Activities that form the entire lifecycle process. Each activity within a Workflow represents a single step of the lifecycle process. Each Workflow Activity can be chained as a dependency to another Workflow Activity. Each Workflow Activity can be defined to execute on a unique remote machine and can span multiple development projects.

All Workflow Activity is managed using real-time centralized logging. All activities executed within a Workflow is logged to a centralized location and displayed in real-time mode.

Build Job	Build Step	Status	Result	Build Machine	User	Build Label	Complete Time	Start Time
METALWORK...	Generate Build...	Completed	Successful	WOODY	tracy	METALWORK...	2006-03-10 14...	2006-03-10 14...
METALWORK...	Dependency A...	Completed	Successful	WOODY	tracy	METALWORK...	2006-03-10 14...	2006-03-10 14...
METALWORK...	Copy Theme F...	Running		WOODY	tracy	METALWORK...	2006-03-10 14...	2006-03-10 14...
METALWORK...	Launch Metal...	Pending		WOODY	tracy	METALWORK...		
METALWORK...	Build Job com...	Pending		WOODY	tracy	METALWORK...		
METALWORK...	Set Classpath ...	Completed	Successful	WOODY	tracy	METALWORK...	2006-03-10 14...	2006-03-10 14...
METALWORK...	Ant Javac of m...	Completed	Successful	WOODY	tracy	METALWORK...	2006-03-10 14...	2006-03-10 14...
METALWORK...	Ant Jar of met...	Completed	Successful	WOODY	tracy	METALWORK...	2006-03-10 14...	2006-03-10 14...
METALWORK...	om complete f...	Completed	Successful	WOODY	tracy	METALWORK...	2006-03-10 14...	2006-03-10 14...

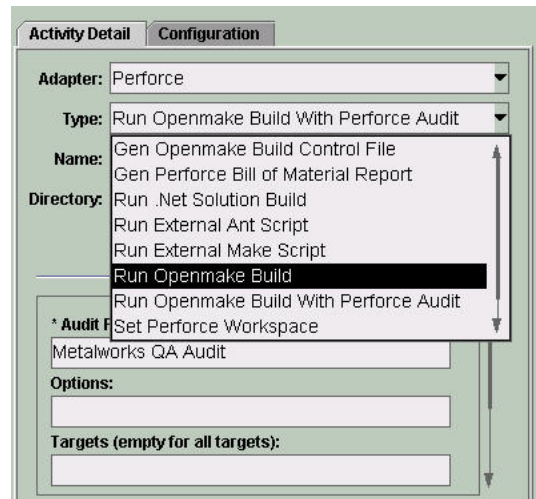
Real Time Build Monitoring

Workflows can be scheduled to run at predefined times.

## ALM Tool Integration

Mojo and Meister provides ALM plug-ins for interoperability with other ALM tools. Plug-ins are provided as point to point integrations, certified by the ALM vendor, and ready for use. Custom plug-ins can be written as needed. The integration with the following ALM vendors is provided out of the box with Meister and Mojo's ALM plug-ins.

- Accurev
- ClearCase
- CVS
- CA Harvest
- Perforce
- MKS Source Integrity
- Borland StarTeam
- Subversion
- Serena PVCS
- Microsoft SourceSafe
- JUnit
- Serena Mover
- CA Unicenter Software Delivery



ALM Activities

## **Enhanced Build Metrics Reporting**

Openmake Meister and Mojo provides enhanced Build Metrics Reporting. Build Metrics Reports provides an easy view of what occurred in the lifecycle process with a focus on the "build" activities. This includes what was checked out for a build, what was changed in a build, build durations and potential build performance problem areas.

## **Bill of Material Reports**

Bill of Material Report provides a listing of the files that existed before the build occurred. A Bill of Material report can show either what source code was retrieved from the SCM tool prior to the build execution or what source code existed in the build directory during the build.

## **Build Difference Report**

A Build Difference Report shows what changed between the current build and the last build executed. This information is helpful in determining what was changed between the builds.

## Difference Report

Toggle All Rows

Type	File	Version	Timestamp	Size	MDS
+ Changed	METALWORKS DEV-My Computer-2006-01-01 12_00_01-bom data		2006-02-28 21:08:06	12061	28719fb5c
+ Changed	METALWORKS DEV-My Computer-2006-01-01 12_00_01-diff data		2006-02-28 21:08:08	12195	9a4c0cd6f
Added	METALWORKS DEV-My Computer-2006-01-01 12_00_02-bom data		2006-02-28 21:31:35	12443	a74ca1f6c
Added	METALWORKS DEV-My Computer-2006-01-01 12_00_02-diff data		2006-02-28 21:31:37	6851	820a3c2d8
- Changed	Metalworks\AquaMetalTheme.java	\main\dev_dev\CHECKEDOUT from \main\dev_dev\ Rule: CHECKEDOUT	2006-02-28 21:25:19	2479	5ee744b2f
	28-Feb.21:25	Steve checkout version "Metalworks\AquaMetalTheme.java" from \main\dev_dev\i (reserved)			
	28-Feb.15:23	Steve create version "Metalworks\AquaMetalTheme.java@0\main\IntegrationStream\1"			
	28-Feb.15:21	Steve create version "Metalworks\AquaMetalTheme.java@0\main\IntegrationStream\0"			
	28-Feb.15:21	Steve create branch "Metalworks\AquaMetalTheme.java@0\main\IntegrationStream"			
	28-Feb.15:03	Steve create version "Metalworks\AquaMetalTheme.java@0\main\dev_dev\1"			
	28-Feb.15:03	Steve create version "Metalworks\AquaMetalTheme.java@0\main\dev_dev\0"			
	28-Feb.15:03	Steve create branch "Metalworks\AquaMetalTheme.java@0\main\dev_dev"			
	28-Feb.15:03	Steve create version "Metalworks\AquaMetalTheme.java@0\main\0"			
	28-Feb.15:03	Steve create branch "Metalworks\AquaMetalTheme.java@0\main"			
	28-Feb.15:03	Steve create file element "Metalworks\AquaMetalTheme.java@0"			
*****					
<<< file 1: Metalworks\AquaMetalTheme.java@0\main\dev_dev\1					
>>> file 2: Metalworks\AquaMetalTheme.java					
*****					
-----[changed 1-56]-----[changed to 1-56]-----					
/*   test+					
* 0 (#) AquaMetalTheme.java 1.4 99+  +					
*   /*+					
* Copyright (c) 1998, 1999 by Sun Mic+ * 0 (#) AquaMetalTheme.java 1.4 99+					
*   /*+					
* Sun grants you ("Licensee") a non-+ * Copyright (c) 1998, 1999 by Sun Mic+					
* modify and redistribute this softwa+ * +					
* provided that i) this copyright not+ * Sun grants you ("Licensee") a non-+e					
* the software; and ii) Licensee does+ * modify and redistribute this softwa+					
* which is disparaging to Sun.   * provided that i) this copyright not+					
*   * the software; and ii) Licensee does+					
* This software is provided "AS IS," + * which is disparaging to Sun.+					
* EXPRESS OR IMPLIED CONDITIONS, REPR+ * +					
* IMPLIED WARRANTY OF MERCHANTABILITY+ * This software is provided "AS IS," +					
* NON-INFRINGEMENT, ARE HEREBY EXCLUD+ * EXPRESS OR IMPLIED CONDITIONS, REPR+					
* LIABLE FOR ANY DAMAGES SUFFERED BY + * IMPLIED WARRANTY OF MERCHANTABILITY+					
* OR DISTRIBUTING THE SOFTWARE OR ITS+ * NON-INFRINGEMENT, ARE HEREBY EXCLUD+					
* LICENSORS BE LIABLE FOR ANY LOST RE+ * LIABLE FOR ANY DAMAGES SUFFERED BY +					
* INDIRECT, SPECIAL, CONSEQUENTIAL, I+ * OR DISTRIBUTING THE					

## Workflow Summary

Workflow Summary shows the Build Label, build results and the duration of the Workflow.

### Build Job Summary

<u>BuildJob</u>	<u>Build Label</u>	<u>Build Number</u>	<u>Result</u>	<u>Start</u>	<u>Duration (in Secs)</u>
METALWORKS DEV	METALWORKS DEV 2006-03-02 09_57_04		Success	2006-03-02 09:57:27	33
METALWORKS DEV	METALWORKS DEV 2006-03-02 09_58_55		Success	2006-03-02 09:59:06	23
METALWORKS DEV	METALWORKS DEV 2006-03-02 20_33_04		Success	2006-03-02 20:33:30	41
METALWORKS DEV	METALWORKS DEV 2006-03-02 20_40_26		Success	2006-03-02 20:43:53	242
METALWORKS DEV	METALWORKS DEV 2006-03-03 08_58_38		Success	2006-03-03 08:59:04	49
METALWORKS DEV	METALWORKS DEV 2006-03-03 09_30_45		Success	2006-03-03 09:31:10	38
METALWORKS QA	METALWORKS QA 2006-03-07 14_22_10		Success	2006-03-07 14:22:37	38
METALWORKS QA	METALWORKS QA 2006-03-07 14_24_58	Build 457	Success	2006-03-07 14:25:18	46
METALWORKS QA	METALWORKS QA 2006-03-07 16_34_22	Build 459	Success	2006-03-07 16:34:33	22
METALWORKS QA	METALWORKS QA 2006-03-07 16_35_43	Build 460	Failed	2006-03-07 16:35:54	16
METALWORKS QA	METALWORKS QA 2006-03-08 12_46_23	Build 464	Success	2006-03-08 12:46:36	24
METALWORKS QA	METALWORKS QA 2006-03-08 15_45_39	Build 465	Success	2006-03-08 15:45:53	36
METALWORKS QA	METALWORKS QA 2006-03-09 11_49_00	Build 466	Success	2006-03-09 11:49:27	36
METALWORKS QA	METALWORKS QA 2006-03-09 12_33_04	Build 467	Success	2006-03-09 12:33:30	35
METALWORKS QA	METALWORKS QA 2006-03-09 12_34_01	Build 468	Success	2006-03-09 12:34:13	22
METALWORKS QA	METALWORKS QA 2006-03-10 07_43_27	Build 469	Success	2006-03-10 07:43:56	34
METALWORKS QA	METALWORKS QA 2006-03-10 07_45_34	Build 470	Success	2006-03-10 07:45:45	21
METALWORKS QA	METALWORKS QA 2006-03-10 08_55_46	Build 471	Success	2006-03-10 08:56:00	29

## Workflow Activity Detail

The Workflow Activity Detail is extremely useful in viewing the duration of each Lifecycle activity. An activity could be the retrieving of source from an SCM tool to the calling of a particular compiler in a Build Step. This report can expose problem areas of the Workflow where performance is an issue.

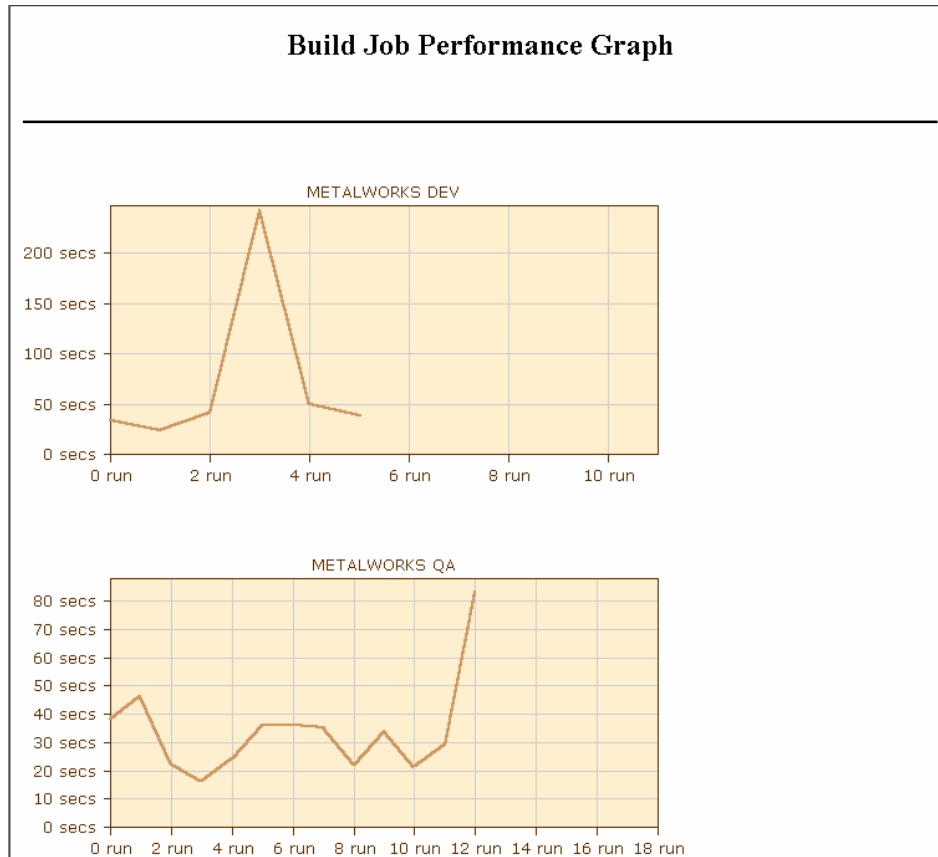
### METALWORKS QA Activity Detail

---

<u>Run</u>	<u>Build Label</u>	<u>Build Number</u>	<u>Result</u>	<u>Start</u>	<u>Duration (in Secs)</u>
1	METALWORKS QA 2006-03-07 14_22_10		Success	2006-03-07 14:22:37	38
2	METALWORKS QA 2006-03-07 14_24_58	Build 457	Success	2006-03-07 14:25:18	46
3	METALWORKS QA 2006-03-07 16_34_22	Build 459	Success	2006-03-07 16:34:33	22
4	METALWORKS QA 2006-03-07 16_35_43	Build 460	Failed	2006-03-07 16:35:54	16
5	METALWORKS QA 2006-03-08 12_46_23	Build 464	Success	2006-03-08 12:46:36	24
6	METALWORKS QA 2006-03-08 15_45_39	Build 465	Success	2006-03-08 15:45:53	36
7	METALWORKS QA 2006-03-09 11_49_00	Build 466	Success	2006-03-09 11:49:27	36
8	METALWORKS QA 2006-03-09 12_33_04	Build 467	Success	2006-03-09 12:33:30	35
9	METALWORKS QA 2006-03-09 12_34_01	Build 468	Success	2006-03-09 12:34:13	22
10	METALWORKS QA 2006-03-10 07_43_27	Build 469	Success	2006-03-10 07:43:56	34
11	METALWORKS QA 2006-03-10 07_45_34	Build 470	Success	2006-03-10 07:45:45	21
12	METALWORKS QA 2006-03-10 08_55_46	Build 471	Success	2006-03-10 08:56:00	29
13	METALWORKS QA 2006-03-10 14_40_20	Build 472	Success	2006-03-10 14:41:02	83
14	METALWORKS QA 2006-03-11 17_12_48	Build 473	Success	2006-03-11 17:13:16	145
15	METALWORKS QA 2006-03-11 17_27_46	Build 474	Success	2006-03-11 17:27:57	19
16	METALWORKS QA 2006-03-11 17_31_24	Build 475	Success	2006-03-11 17:31:35	25
17	METALWORKS QA 2006-03-11 17_35_08	Build 476	Success	2006-03-11 17:35:20	22
18	METALWORKS QA 2006-03-11 17_41_09	Build 477	Success	2006-03-11 17:41:31	38

## Build Summary Performance Graph

The Build Summary Performance Graph shows the Workflow Summary Report in line graph format. This gives you an easy view of how long each Workflow is taking over time and allows you to quickly monitor Workflow performance.



## IDE Integration

Openmake Meister and Mojo will include an Eclipse Plug-in allowing Eclipse developers to download Mojo from sites such as Eclipse Plug-in Central. The Plug-in will allow developers to execute Mojo from within their current Eclipse IDE, or IBM RAD. For .Net developers, an executable will be provided so that Mojo can be launched from within the Visual Studio Framework.

## **Openmake Meister 7.0 Feature Summary**

---

Openmake Meister includes the core functionality of the Openmake 6.41.1 product as defined below:

- Build Configuration Management (Build Types and Rules)
- Dependency Management (Search Paths and TGTs)
- Eclipse, IBM-RAD, and .Net build integrations and Plug-ins
- User and Group Administration
- Advanced LDAP security
- Impact Analysis
- Build Audit Reporting
- Real-Time Build Monitoring
- Project and Build Type Reporting and Metrics
- Mojo ALM processing, tool integration, reporting and scheduling

Meister 7.0 includes a new Graphical User interface based on the Eclipse RCP.

### **Workflows to Replace "Projects"**

With Openmake Meister, the concept of an Openmake "Project" will be retired and replaced by a Workflow. TGT files, Search Paths, and setup build options will be contained within a Workflow. The Workflow will be the primary object to contain all Build information. The Workflow will be used in both Openmake Mojo and Openmake Meister. A Workflow in Openmake Meister will have additional features not available in Openmake Mojo. These features will include:

- Build Types
- Search Paths
- TGT definition
- Setup Build Options – Impact Analysis, Build Audit, Dependency Management

### **Public and Private Workflows**

Only Openmake Meister will provide the ability to secure Workflows and restrict Workflows to particular users or Groups.

## IDE Integration

Openmake Meister will include an Eclipse Plug-in allowing Eclipse developers to execute the compile and dependency management features of Openmake Meister from inside Eclipse, IBM-RAD or .Net/Visual Studio.

## Additional Information

---

For more information on OpenMake Meister and OpenMake Mojo visit [www.OpenMakeSoftware.com](http://www.OpenMakeSoftware.com) for technical whitepapers and datasheets.

## Company Overview

---

OpenMake Software is a premiere provider of Enterprise Build Management software and consulting services dedicated to assisting customers with the implementation of an enterprise build process. OpenMake Software has specialized in the design and implementation of reliable and repeatable application build processes for Global 2000 organizations since 1995.

### **North American Headquarters and Worldwide Sales**

Openmake Software  
213 W. Institute Place #404  
Chicago, IL 60610

Ph: 800-359-8049 or 312-440-9545