

---

# **Software Repository - Evaluation Criteria & Dissemination**

**Revision: 0.2**

Author: Steven Newhouse

---

Open Middleware Infrastructure Institute  
Room 6005, Building 21 (Faraday)  
Highfield Campus  
University of Southampton  
Hampshire, SO17 1BJ  
Tel: +44 (0)2380 58787

---

---

<b>Revision No.</b>	<b>Author</b>	<b>Changes Made</b>	<b>Date Revised</b>
0.1	Steven Newhouse	Initial content – ready for review	20/5/05
0.2	Steven Newhouse	Comments from CLW & SC.	26/5/05

---

## Table of Contents

<i>1</i>	<i>Overview</i>	3
<i>2</i>	<i>Captured Information</i>	3
2.1	Product Details	3
2.2	Release Details	3
<i>3</i>	<i>Evaluation Criteria</i>	3
3.1	Packaging Structure	4
3.2	Documentation	4
3.3	Operation	5
3.4	Management	5
3.5	Identified Issues	6
3.6	Summary	6

---

## 1 Overview

This document specifies the three critical phases of the software repository process:

- The information that must be captured when a product is created within the repository and a specific release submitted to the repository.
- The assessment criteria that should be used to review the software contribution.
- How the product and release information, coupled with the evaluation results, are presented within the repository.

The submission and assessment workflow and its support within the OMII repository portal is detailed elsewhere.

## 2 Captured Information

In the following sections the fields marked \* are required. Text fields are a line <256 chars. Text areas are multi-lined with large text length (i.e. enough to hold a licence agreement).

### 2.1 Product Details

Information relating to the:

- Product Name (text field)\*
- Product Description (text area)\*
- Product URL (text field)\*
- Contributing Organisation's name (text field)\*
- Contributing Organisation's contact details & address (text area)\*
- Support Description (text area)
- Support URL (text field)
- Developer email alias (text field) (NB: An individual name and email contact can be obtained from the person who submitted the product/release)
- Discuss mailing alias (text field)
- Discuss mailing list subscription (text area)
- Announce mailing alias (text field)
- Announce mailing list subscription (text area)
- Creation Date

### 2.2 Release Details

Information relating to the:

- Version\* (structured)
- License text\* (text area)
- Description\* (text area)
- URL to obtain the software\* (text field)
- Documentation.URL (text field)
- Creation Date

## 3 Evaluation Criteria

The evaluation points covers several areas. The following key applies to the table:

- Criteria: The aspect under consideration
- Description: A description of the criteria
- Form: How the information is captured:
  - **Score:** A 0-5 radio button. 0: Not applicable. 1-5 numerical assessment. A high score is better (i.e. nearer to the ideal).
  - **Comment:** A one line comment to clarify the score
- Ideal: What must be achieved to get a score of 5

The table is broken into sections to indicate that a series of internal evaluation criteria are grouped under a public summary criteria (in italics).

### 3.1 Packaging

The OMII has developed a packaging specification to simplify the integration of different managed programme products into the distribution. While integration with this specification is not technically essential it should significantly reduce the integration burden on the internal OMII team and allow standard scripts to be developed.

Criteria	Description	Form	Ideal
<i>Integration With OMII Packaging</i>	<i>How well aligned is the product with the OMII's integration specification?</i>	S & C	<i>A high score indicates that the product could be easily integrated with the OMII's installers.</i>
Build	Support for the OMII specified build targets	S & C	Full compliance with the targets in the OMII integration document
<targets> ...	Support for the OMII specified install targets	S & C	Full compliance with the targets in the OMII integration document

### 3.2 Documentation

The OMII has developed a specification describing the documentation that it expects from its contributors. A high overall score here indicates that the user can be reasonably confident that the supporting documentation will answer the majority of their queries.

Criteria	Description	Form	Ideal
<i>Documentation</i>	<i>How comprehensive and useful is the provided documentation?</i>	S & C	<i>A high score indicates that the documentation provides sufficient depth and coverage to be useful for those trying to utilise the product for its main purpose/function.</i>
Introductory Docs.	A concise summary of the software	S & C	Information as to what the software does and how to quickly get started with it.
Pre-requisite Docs.	Information relating to environment required for running this software.	S & C	Is the environment required for this software well described?
Installation Docs.	Information on how to install the software.	S & C	Clear instructions on how to install the software.
User Docs.	Information on the API	S & C	Clear simple user manual with usage scenarios and sample code
Admin Docs.	Information on how to administer the software	S & C	Clear instructions on how to configure the software and maintain it in operation
Tutorials	Details how to use the software	S & C	Clear, simple step-by-step description how to use the software with code samples if appropriate

Functional Specification	Functional specification of the software	S & C	Clear, simple description of product's functionality
Implementation Specification	Implementation details of functional specification	S & C	This document should contain not only the implementation details but also the reasons.
Test Documents	Details of product testing carried out within the OMII container.	S & C	Details of the test plans, test code (e.g. JUnit) and results (with known issues) from running on various platforms and scenarios. Also describe how the user can repeat the same tests.

### 3.3 Technical

The OMII evaluator will use the provided documentation to try and use the software. Their success (or otherwise) in using the software will demonstrate if the contributed software provides useful functionality.

Criteria	Description	Form	Ideal
<i>Technical</i>	<i>An examination of the technical components of the software</i>	<i>S a&amp; C</i>	<i>Can the product be deployed and does it run successfully from the provided documentation?</i>
Pre-requisites	Software and environment changes necessary to support the installation of the software.	S & C	Are the pre-requisites accurately described and sufficient to install and run the software?
Deployment	Deployment of the product into the server or client environment.	S & C	How easy is the deployment of this software into the required environment?
Verification	Evaluating the correct operation of the product.	S & C	Is it clear how you can verify that the software has been successfully deployed and operating correctly, e.g. post-installation tests?
Stability	Determination as to the stability of the production.	S & C	Does the software run reliably under reasonable usage and are there tests to support this?
Scalability	An assessment of the scalability of the software.	S & C	How well does the software respond to high levels of utilisation and concurrent client activity and are there tests to support this?
Coding	An inspection of the code within the software	S & C	Does the software comply with OMII's coding guidelines?

### 3.4 Management

A decision to use software within a project is driven by many non-functional issues such as support, adoption, etc. This section characterises some of these issues.

Criteria	Description	Form	Ideal
----------	-------------	------	-------

<i>Management</i>	<i>Non-functional issues relating to the product</i>	S & C	Does the product appear to have a supported, sustainable future?
Support	The support model provided by the contributors.	S & C	Is the software supported through the community or dedicated resources?
Sustainability	Information about the products future.	S & C	Is there a roadmap with supporting funding defining the products technical future?
Standards		S & C	Does the software support mainstream specifications that are standards or becoming standards?

### 3.5 Identified Issues

Specific correctable issues will be identified during the course of the evaluation that will need to be rectified to improve the software. These should be identified in the Bugzilla (or equivalent) belonging to this product and references in the evaluation report:

- Short comment describing the issues
- Bug ID

The Bug ID should be coupled with the Bugzilla URL to provide a direct link to the external Bugzilla entry so progress can be reviewed. If the project does not provide a Bugzilla instance then the issues should be filed in the OMII Bugzilla enabling remedial work to be costed and prioritised.

Criteria	Description	Form	Ideal
Issues	List of outstanding issues	S & C	Describe not only the issues, but also some ideas for solving them.

### 3.6 Summary

A summary section provides an overall assessment of the software to be reached.

Criteria	Description	Form	Ideal
<i>Overall</i>	<i>An overall assessment of the product.</i>	S & C	<i>How far does this software product go towards meeting OMII's goal of distributing robust, reliable, well engineered, documented and supported software?</i>
<i>Public Comments</i>	<i>Public comments by the evaluator.</i>	Text	
Private Comments	Private comments by the evaluator which only be available to OMII & the contributor.	Text	