

软件资源库管理规范

Software Resource Management Specification

(TRUSTIE-SRMC V 2.0)

2009 年 10 月 10 日

发布声明

牵头单位：北京大学信息科学技术学院

参研单位：国防科技大学计算机学院

北京大学信息科学技术学院

中国科学院软件研究所

中创软件公司

执笔人：赵俊峰、谢冰、邹艳珍、李戈

版本号：2.0

发布时间：2009年10月10日

审核人：王怀民

版权声明

本技术规范得到国家 863 重点项目“高可信软件生产工具及集成环境”第一课题“可信的国家软件资源共享与协同生产环境”的资助，版权归“可信的国家软件资源共享与协同生产环境”课题组所有。

本规范在以下条件下可以自由传播：

- (1) 保持本规范的完整性（包括发布声明）；
- (2) 未经课题组许可，任何人不得以本规范盈利。

Trustie 技术规范



变更记录

变更版本	日期	A/M/D	原因与修改情况描述	修订人	审核人
V1.0	2009.5.4	M	第 4 章有关资源描述	赵俊峰	谢冰
V1.0	2009.5.4	A	第五章发布资源实体	赵俊峰	谢冰
V2.0	2009.10.10	A	第 4, 5 章有关可信证据管理部分, 增加附录 3 的示例	赵俊峰	谢冰

注: A - 增加 M - 修改 D - 删节



目 录

第 1 章	引言	1
1.1	背景.....	1
第 2 章	范围与引用	2
2.1	范围.....	2
2.2	引用.....	2
第 3 章	术语和定义	4
3.1	软件.....	4
3.2	软件产品.....	4
3.3	软件资源.....	4
3.4	软件构件.....	5
3.5	软件服务.....	5
3.6	软件工具.....	5
3.7	软件构架.....	5
3.8	软件质量.....	5
3.9	软件可信.....	5
3.10	软件可信性.....	5
3.11	软件可信属性.....	5
3.12	软件可信等级.....	6
3.13	软件可信证据.....	6
3.14	软件可信分级评估.....	6
3.15	信任.....	6
3.16	信誉.....	6
3.17	服务质量.....	6
第 4 章	软件资源库中资源的类型	7
第 5 章	软件资源管理的相关数据模型	8
5.1	用户描述信息.....	8
5.2	资源描述信息.....	9
5.3	软件资源分类描述信息.....	11
5.4	软件资源质量信息.....	15
5.5	用户信任关系信息.....	20
5.6	资源库统计信息.....	21
5.7	可信证据信息.....	22
5.8	其它：资源关系描述信息等.....	27
第 6 章	软件资源管理及使用过程	28
6.1	用户管理.....	28
6.1.1	用户注册.....	28
6.1.2	用户登录.....	29
6.1.3	用户信息修改.....	29

6.1.4	用户权限的控制.....	29
6.2	系统管理员.....	29
6.3	发布资源.....	30
6.3.1	发布资源基本信息.....	30
6.3.2	发布资源实体信息.....	31
6.3.3	发布质量信息.....	33
6.4	检索资源.....	37
6.4.1	简单检索.....	37
6.4.2	复杂检索.....	37
6.4.3	基于 Tag (标签) 检索.....	38
6.4.4	检索资源的结果显示.....	38
6.5	反馈评估.....	38
6.6	软件资源可信证据的管理.....	43
6.7	统计分析.....	44
6.8	我的资源库.....	46
6.9	帮助信息.....	47
6.10	中英文界面.....	47
附录 1: 软件常见许可证列表.....		48
附录 2: 软件资源库中英文术语对照表.....		53
附录 3: 软件资源可信证据示例.....		57

第1章 引言

1.1 背景

软件资源是软件产业、软件开发的基础。软件资源库通过对软件资源的收集、分类、存储和检索等支持，为软件开发提供资源管理的支持。

本规范给出了软件资源管理的相关技术术语、管理过程、描述软件资源的数据模型，以及相关软件资源发布、检索、分类等活动。本规范面向软件资源库的开发者和软件资源库的使用者。

本规范为“可信的国家软件资源共享与协同生产环境”中软件资源的管理提供了参考依据，该项工作由北京大学牵头负责，由国防科技大学、北京航空航天大学、中国科学院软件所、中创软件中间件有限公司等单位共同参与。

软件资源管理规范工作组从 2008 年 1 月份开始工作，一年来，工作组成员进行了大量的研究和广泛的交流讨论工作，最终形成了本参考规范 2.0 版。

本规范起草的主要成员包括：北京大学谢冰、赵俊峰、邹艳珍、李戈。

第2章 范围与引用

2.1 范围

本规范对软件资源的概念、软件资源的数据模型、软件资源管理的范围、软件资源管理的使用过程等进行了阐述。本规范给出了对软件资源进行有效管理的方法，适用于有需要软件资源管理的组织和个体。

2.2 引用

下列文件中的有关条款通过引用而成为本标准的条款。凡注日期或版次的引用文件，其后的任何修改（不包括勘误的内容）或修订版都不适用于本标准，但提倡使用本标准的各方探讨使用其最新版本的可能性。凡不注明日期或版次的引用文件，其最新版本适用于本标准。

北大西洋公约组织（NATO）的三个关于可复用软件构件的标准：

- Contel Corporation. Standard for the Development of Reusable Software Components. NATO contract number CO-5957-ADA, 1991;
- Contel Corporation. Standard for Management of a Reusable Software Component Library. NATO contract number CO-5957-ADA, 1991;
- Contel Corporation. Standard for Software Reuse Procedures. NATO contract number CO-5957-ADA, 1991.

IEEE 关于可复用软件构件的标准：

- IEEE Std 1420.1-1995, Reuse Library Interoperability Group (RIG), IEEE Standard for Information Technology—Software Reuse—Data Model for Reuse Library interoperability: Basic Interoperability Data Model (BIDM);
- IEEE Std 1420.1a-1996, Reuse Library Interoperability Group (RIG), Supplement to IEEE Standard for Information Technology—Software Reuse—Data Model for Reuse Library Interoperability: Asset Certification Framework;
- IEEE Std 1420.1b-1999, Reuse Library Interoperability Group (RIG),

IEEE Trial-Use Supplement to IEEE Standard for Information Technology—Software Reuse—Data Model for Reuse Library Interoperability: Intellectual Property Rights Framework;

- IEEE Std 1430-1996, Reuse Library Interoperability Group (RIG), IEEE Guide for Information Technology—Software Reuse—Concept of Operations for Interoperating reuse Libraries.

信息产业部颁布:

- 中国电子行业标准: SJ/T11373-2007 《软件构件管理 第1部分: 管理信息模型》标准。

其他参考文献:

- W.B. Frakes and T.P. Pole, Proteus: A Reuse Library System that Supports Multiple Representation Methods, ACM SIGIR Forum, vol. 24, pp. 43-55, 1990;
- Mikio A., Sanjiva W., Hiroshi M., Web Services Engineering: Promises and Challenges, Proceeding at International Conference on Software Engineering 2002, pp 647-648;
- UDDI.ORG, UDDI Version 2.0 Published Specification, 2001;
- UDDI.ORG, UDDI Version 3.0 Published Specification, 2002.

第3章 术语和定义

3.1 软件

信息处理系统的全部或部分程序、规程、规则和相关文档。

3.2 软件产品

一组计算机程序、规程以及可能有的相关文档和数据。又称为软件制品。

3.3 软件资源

1、IEEE 1517 标准定义资源为：

一个诸如设计、规约、源代码、文档或者测试过程指南等的项目（item），它被设计为可以在多种上下文环境中使用。

2、IEEE BIDM (Basic Interoperability Data Model , IEEE Standard 1420.1,1995) 定义资源为：

保存在复用库中的、复用者可能感兴趣的项，例如设计文档、规约、源代码、文档、测试方案等，或对复用者有潜在价值的任何其它信息单元。

3、OMG RAS Version 2.2 (Reusable Asset Specification, OMG Standard, November 2005) 定义资源为：

软件资源是一个软件制品的集合，用以解决一个或一类在软件开发周期中所遇到的问题。

4、信息产业部颁布的电子行业标准 SJ/T11373-2007《软件构件管理 第1部分：管理信息模型》标准定义软件资源为：

具有复用价值的软件制品，它包括软件需求、软件设计、软件代码、测试案例和文档等。软件资源可以用于创造新的软件产品以及相关的制品。

本规范中对软件资源的定义是：

软件资源是软件生存周期中具有复用价值的软件制品，它包括软件需求、软件设计、软件代码、测试案例和文档等。软件资产可以用于创造新的软件产品以及相关的制品。

3.4 软件构件

软件构件 (Software Component): 软件系统中具有相对独立功能、可以明确辨识、接口由契约指定、和语境有明显依赖关系、可独立部署、且多由第三方提供的可组装软件实体。(摘自《计算机大百科全书》)

- 复用者使用特定的构件实体进行系统组装, 这种构件实体一般为某种语言代码或特定格式二进制代码; 构件实体被嵌入在最终用户使用的应用系统中

3.5 软件服务

以服务的形式提供给复用者使用的独立的、分布式的软件服务。服用者使用该服务提供的服务接口进行系统组装, 服务本身同时还可以独立地为其他系统提供服务。典型的软件服务包括 Web services 等

3.6 软件工具

一类软件, 用来辅助计算机软件的开发、运行、维护、管理、支持等过程中的活动和任务。(摘自《计算机大百科全书》)

3.7 软件构架

软件构架 (Software Architecture): 软件系统的结构包含软件元素、软件元素外部可见的属性以及这些软件元素之间的关系; 软件系统的基本组织, 包含构件、构件之间、构件与环境之间的关系, 以及相关的设计与演化原则等。

3.8 软件质量

与表征软件能够满足规定或隐含需求能力有关的软件的所有特性。

3.9 软件可信

如果一个软件系统的行为总是与用户预期的行为和结果相一致, 则称该软件可信 (trustworthy)。

3.10 软件可信性

软件按照用户期望提供安全可靠服务的能力。

3.11 软件可信属性

用以描述和评价软件系统可信的一组属性。软件可信属性可被细化成多级子

属性。

3.12 软件可信等级

尺度中某一范围的值可用来按照明确或隐含的需求对软件可信进行分级评定，这些可信级别称为可信等级，又称为软件可信级别。软件的可信等级的评定与软件的应用领域、软件类别等密切相关。

3.13 软件可信证据

与软件相关的能够反映其某种可信属性的数据、文档或其他信息，称为软件可信证据。

3.14 软件可信分级评估

依据特定的已成文的软件可信评估准则，确定特定的软件产品是否达到某一特定可信等级的活动，称为软件可信分级评估，简称为软件可信评估。

3.15 信任

在软件资源库中，信任描述了用户（主体，信任者）对构件（客体，被信任者）在满足一定信任属性要求的前提下完成某个操作的信心。其中，信任属性定义建立在构件依赖于某种应用环境的行为约束方面，如可靠性、可用性、安全性等。

3.16 信誉

一个实体的信誉是其他实体在过去与之交互过程中对其反馈的综合评价。信誉是一种从系统中潜在的社会群体中导出的可量化信息，它具有综合性和统计性：信誉表达了在这个社会群体中从所有或多数成员角度出发的一致性观点，即信誉具有综合性。信誉不仅仅在于评价一次服务的完成情况，它建立在对实体多次行为的评估的基础上，即信誉具有统计性。

3.17 服务质量

服务质量 QoS (Quality of Service)是一个综合指标，用于衡量使用一个服务的满意程度，描述关于一个服务的某些性能特点。

第4章 软件资源库中资源的类型

软件资源的类型可以分为软件构件、软件服务、软件工具、软件构架和其他等。其中，

- 软件构件：以软件代码形态出现，可以嵌入到目标系统中的有机构成成分。
- 软件服务：以软件服务形态出现的软件运行实体，可以在目标系统中引用或调用的独立运行成分。
- 软件工具：在开发阶段使用，而在最终目标系统中无具体应用的软件系统。
- 软件构架：若干软件资源有机构成特定系统时的资源间相互连接关系。
- 其他：除去上述 4 种类型的资源，如软件知识、文档。

第5章 软件资源管理的相关数据模型

软件资源库中要存储、管理各种不同类型的软件资源，软件资源管理的相关数据模型包含了软件资源的各种元素及属性和各类元素之间的关联关系等相关定义，主要包含用户信息、资源描述信息、资源包描述信息、分类描述信息、资源质量信息/反馈信息、用户信任关系模型、资源库统计信息和其它诸如资源关系描述信息等信息。

5.1 用户描述信息

用户描述信息用于描述在资源库中注册的用户注册信息和由系统自动生成的信息。用户描述信息分为针对个人用户和企业用户两种类型的信息。

名称	类型	约束	备注说明	来源	
用户名 (Account Name)	String	必须提供	用户名由字母、数字和“_”组成	用户填写信息	
密码(Password)	String	必须提供	密码由字母和数字组成		
查询密码问题	String	可选			
查询密码答案	String	可选			
用户真实姓名(Real Name)	String	必须提供	仅个人用户填写		
企业名称(Enterprise Name)	String	必须提供	仅企业用户填写		
企业联系人(Enterprise Contactor)	String	必须提供	仅企业用户填写		
电子邮箱(E-mail)	String	必须提供			
联系电话(Telephone)	String	可选			
联系地址(Address)	String	可选			
传真(Fax)	String	可选			
ID	String	必须			系统自动生成
上次登录时间 (LastLoginTime)	Date	必须			
注册时间 (RegisterTime)	Date	必须			

5.2 资源描述信息

资源描述信息包含两部分信息，一部分为资源在资源库中描述信息，另一部分为资源包的描述信息。

其中资源信息包括如下信息：

名称	类型	约束	备注说明	来源
名称 (Name)	String	必须提供	资源的名称	用户填写信息
类型 (Type)	枚举型，包括了软件构件、软件服务、软件工具、软件构架和其他五类	必须提供	资源的类型	
描述 (Description)	String	必须提供	对资源的一般性解释	
标签 (Tag)	String	可选	可以有多个标签	
费用 (Cost)	枚举型，包括了 Open, Free 和 Commercial 三类	可选		
价格 (Price)	Float	可选	当费用型为 Commercial 或 Open 时，描述了用户为获得该资源所必须付出费用的数量（以美元为单位）	
产权证书	String	可选	当费用型为 Open, Free 时，描述用户必须遵守的产权保护证书	
限制 (Restriction)	String	可选	限制资源使用的因素；包括算法、编译器、可移植性、环境等	
认证方法 (Certification)	String	可选	描述了评价或认证资源的方法，包括评价或认证的结果，实施认证的日期，实施认证的人员等	
参考 (Reference)	String	可选	包含了资源介绍信息源的声明，例如 WSDL 等信息	
目标环境 (TargetEnvironment)	String	可选	资源开发所针对的计算机系统、操作系统、编译器等	
版本 (Version)	String	可选	资源对应的版本信息	
访问地址 (accessPoint)	String	可选	当资源为软件服务时，描述该服务的访问地址信息	

名称	类型	约束	备注说明	来源
ID (ID)	UUID	必须	资源的唯一标识	系统自动生成
传输大小 (TransferSize)	Integer	必须	描述了资源在进行传输的时候所需要的字节数	
最后修改日期 (LastUpdateDate)	Date	必须	描述资源最后修改的日期	
发布日期 (PublishDate)	Date	必须	描述资源发布的日期	

作为资源描述信息的另一部分资源包的描述信息，描述了一个资源包含多种相关文件的打包信息。目前拟对 OMG 组织的 RAS 标准规范进行适度扩展以描述该资源包信息。

2003 年 10 月，IBM/Rational、ComponentSource 等向 OMG 组织提交了可复用资源规范 RAS(Reusable Asset Specification)，规定了可复用软件资源的结构、内容和描述方法。RAS 提供了一种标准的方法用于打包和释放一系列相关的文件，这些文件或工作产物的集合被称为 RAS 资源。RAS 在装配，组织，存储和文档化资源方面提供了指导。

资源包的信息包括如下信息：

名称	类型	约束	备注说明	来源
需求文件	可以是任何类型文件	上传的文件应该是与所发布资源的需求相关的文件	用户可以在该部分上传与资源需求相关的所有文件，例如资源开发过程中的需求分析文档、用例图、用例说明文档等。	用户提供
设计文件	可以是任何类型文件	上传的文件应该是与所发布资源的设计相关的文件	用户可以在该部分上传与资源开发生命周期中设计阶段相关的所有文件，例如资源开发过程中的系统设计文档、资源设计说明文档、UML 类图等。	用户提供
资源实体文件	可以是任何类型文件	上传的文件应该是实现资源功能的实体文件，如果发布的是“软件构件”，用户必须需要上传资源的实体文件	用户可以在该部分上传与资源功能实现有关的所有实体文件，例如资源的源代码、二进制代码等。	用户提供

名称	类型	约束	备注说明	来源
测试文件	可以是任何类型文件	上传的文件应该是与所发布资源的测试相关的文件	用户可以在此部分上传与所发布资源的测试相关的所有文件，例如资源的测试用例、测试报告等。	用户提供
资源使用帮助文件	可以是任何类型文件	上传的文件应该是有助于所发布资源的使用的说明文件	用户可以在此部分上传资源的使用帮助文件，例如资源的使用说明文档、用户手册等。	用户提供
WSDL文件的地址	字符串	必须是 WSDL 文件的有效 URL，如果发布的“Web Service”，用户必须提供服务的 WSDL 文件的 URL	前半部分下拉菜单中是供选择的 WSDL 的 URL 的通信协议类型，包括 http/https/ftp 三种。后半部分是 WSDL 的不包括通信协议部分的 URL 地址：比如完整某 WSDL 文件的 URL 为 http://www.sei.pku.edu.cn/test.wsdl ，则填写内容为 www.sei.pku.edu.cn/test.wsdl	用户提供

5.3 软件资源分类描述信息

软件资源分类描述信息针对各个软件资源的特点，由资源发布者提交分类信息，以帮助用户更好地检索软件资源。软件资源分类信息包含了刻面分类、关键字分类、枚举分类三种不同分类方式的分类信息。分类信息可以描述软件构件、软件服务、软件工具和其他资源。如图 1 所示：

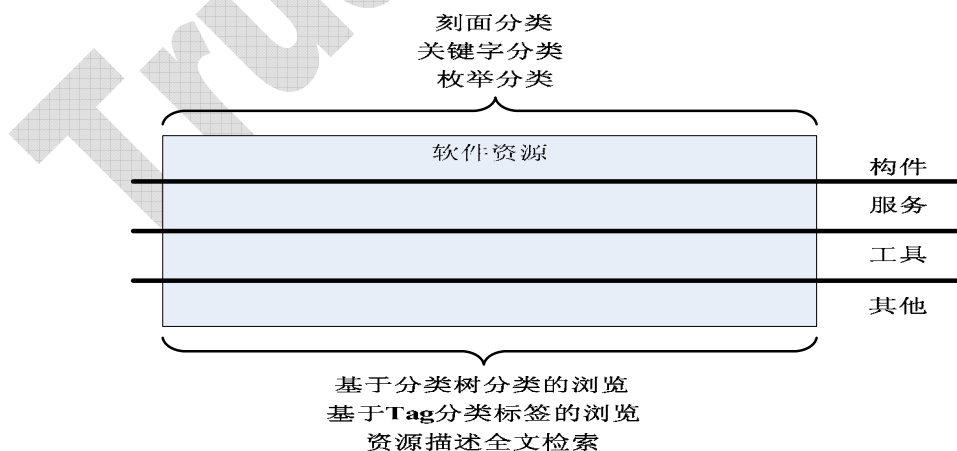


图 1 分类描述信息和应用

软件资源分类描述信息如下所示：

名称	类型	约束	备注说明	来源
名字(Name)	String	必须提供	资源的名称	用户填写信息
资源类型(Type)	枚举类型，分为构件、服务、构架、其他	必须提供		
资源描述关键词(Key Words)	String	必须提供	可以由多个词组成，以逗号分隔，用户可以任意填写，不受限	
相关刻面术语(Facet)		可选	从本库所给出的受限术语集合（树型结构）中选择与资源相关的关键词	
资源分类标签(Tag)	String	必须提供	根据用户输入的关键词生成的、帮助用户检索资源的 Web 标签	系统自动生成
资源分类树(Classification)	String	必须提供	用于引导用户对资源进行浏览的树型结构	

目前软件资源库所采用的系统默认刻面分类模式如下：

- pku_sei:facet
 - Application Domain
 - AI
 - Business Automation
 - POS
 - CAD
 - CAI
 - Mathematics
 - Tele-Instruction
 - CASE
 - CM
 - Component Library
 - OO Tools
 - Product Line
 - Program Understanding
 - Test Tools
 - CSCW
 - DSS
 - ERP
 - General
 - Geologic Engineering

-
- Engineering Geology
 - hydrogeology
 - MIS
 - Cabaret Management
 - Finance
 - Financial affairs
 - Application Environment
 - Database
 - DB2
 - Desktop DB
 - Access
 - Foxpro
 - XBASE
 - Informix
 - MS SQL Server
 - MySQL
 - Oracle
 - Sybase
 - OS
 - DOS
 - Linux
 - RedHat
 - UNIX
 - SOLARIS
 - WINDOWS
 - WINDOWS 2000
 - WINDOWS 3.X
 - WINDOWS 95
 - WINDOWS 98
 - WINDOWS NT
 - WINDOWS XP
 - Open GL
 - Functionality
 - Compiler
 - Graphics Language
 - Syntax
 - Text Language
 - lexical
 - symbol list
 - Database
 - Modeling
 - Multimedia
 - Graphics
 - 2D graphics

-
- 3D graphics
 - Speech
 - Video
 - Vision
 - Network
 - Output
 - Chinese Characters
 - System
 - IO
 - Keyboard
 - Mouse
 - Testing
 - Software Testing
 - User Interface
 - GUI
 - Java Interface
 - Text
 - Web
 - courseware
 - Math Expression
 - edlin
 - charater string
 - Level
 - Analysis
 - Code
 - Design
 - Test
 - Representation
 - Code
 - Binary Code
 - Byte Code
 - SourceCode
 - Graph
 - DFD
 - MSD
 - OO Diagram
 - JBOO Model
 - OMT
 - Rose Model
 - Programming Language
 - ASM
 - Basic
 - Visual Basic
 - C

- C++
 - Visual C++
- Java
- Pascal
 - Delphi
- Text
 - TXT
 - Word
- Type
 - ActiveX
 - COM/DCOM
 - CORBA
 - EJB
 - JadeBird

5.4 软件资源质量信息

以支持用户反馈为目标，描述软件资源的质量，依据度量方法评价软件资源质量，以帮助用户有效地选择软件资源。软件资源质量信息均有用户提交，该信息包含以下内容：

质量属性名称	定义	值域	取值方法
功能性	适 合 性	$0 \leq X \leq 1$	$X = 1 - A/B$ ($0 \leq X \leq 1$) A=在评价中检测有问题的功能数 B=系统声称可以实现的总功能数 5级：功能全部实现而完全正确 ($X=1$) 4级：极少的功能有问题 ($0.85 \leq X < 1$) 3级：较少的功能有问题 ($0.7 \leq X < 0.85$) 2级：较多的功能有问题 ($0.5 \leq X < 0.7$) 1级： 很多的功能有问题 ($0 \leq X < 0.5$)
	准 确 性	$0 \leq X \leq 1$	$X = 1 - A$ ($0 \leq X \leq 1$) A=软件产品没有提供所需精确度的功能数的比例 5级：准确 ($X=1$) 4级：比较准确 ($0.85 \leq X < 1$) 3级：基本准确 ($0.7 \leq X < 0.85$) 2级：部分准确 ($0.5 \leq X < 0.7$) 1级：不准确 ($0 \leq X < 0.5$)
	互 操 作	$0 \leq X \leq 1$	$X = 1 - A/B$ ($0 \leq X \leq 1$) A=软件产品不可以成功进行交互的系统数目

	性	力		B=软件产品声称可以进行交互的系统的数目 5级：好 (X=1) 4级：较好 (0.85≤X<1) 3级：中 (0.7≤X<0.85) 2级：较差 (0.5≤X<0.7) 1级：差 (0≤X<0.5)
	保密安全性	软件产品保护信息和数据的能力，以使未经授权的人员或系统不能阅读或修改这些信息 and 数据，而不拒绝授权人员或系统对它们的访问	{5, 4, 3, 2, 1}	5级：严格的用户权限管理 (X=5) 4级：用户权限检验 (X=4) 3级：用户权限检查 (X=3) 2级：区分用户权限 (X=2) 1级：不区分用户权限 (X=1)
可靠性	成熟性	测试充分性	$0 \leq X \leq 1$	X=A (0≤X≤1) A=测试过程对软件产品测试覆盖的充分度 5级：非常充分 (X=1) 4级：较充分 (0.85≤X<1) 3级：中 (0.7≤X<0.85) 2级：不充分 (0.5≤X<0.7) 1级：差 (0≤X<0.5)
		软件产品为避免由软件中错误而导致失效的能力	$0 \leq X$ ，单位：秒	X=失效的平均间隔时间
	容错性	在软件出现故障或者违反指定接口的情况下，软件产品维持规定的性能级别的能力	$0 \leq X \leq 1$	X=1-A/B (0≤X≤1) A=损坏发生的次数 B=软件失效的数目 5级：好 (X=1) 4级：较好 (0.85≤X<1) 3级：中 (0.7≤X<0.85) 2级：较差 (0.5≤X<0.7) 1级：差 (0≤X<0.5)
		易恢复性 (可用性)	在失效发生的情况下，软件产品重建规定的性能级别并恢复受直接影响的数据的能力	$0 \leq X \leq 1$

效率	时间特性	响应时间	$0 \leq X$, 单位: 秒	$T = (\text{获得结果的时间}) - (\text{完成命令输入的时间})$
		吞吐率	$0 \leq X$, 单位:	$X = A/T \quad (0 \leq X)$ A=完成的任务个数 T=观察的时间周期
	资源利用率	在规定条件下, 软件产品执行其功能时, 使用合适的资源数量和类别的能力	$0 \leq X$, 单位:	内存使用大小, CPU 使用率等
易用性	易理解性	软件产品用户能理解软件是否合适以及如何能将软件用于特定的任务和使用环境的能力	$0 \leq X \leq 1$	$X = 1 - A/B \quad (0 \leq X \leq 1)$ A=不能被理解的功能(或功能的类型)数 B=功能(或功能的类型)总数 5级: 好 ($X=1$) 4级: 较好 ($0.85 \leq X < 1$) 3级: 中 ($0.7 \leq X < 0.85$) 2级: 较差 ($0.5 \leq X < 0.7$) 1级: 差 ($0 \leq X < 0.5$)
	吸引力	软件产品吸引用户的能力	{5, 4, 3, 2, 1}	5级: 极具吸引力 ($X=5$) 4级: 具有吸引力 ($X=4$) 3级: 有吸引力 ($X=3$) 2级: 吸引力较差 ($X=2$) 1级: 吸引力差 ($X=1$)
	易操作性	软件产品使用户能操作和控制它的能力	$0 \leq X \leq 1$	$X = 1 - A/B \quad (0 \leq X \leq 1)$ A=用户发现不易操作的功能数 B=功能的总数 5级: 极易操作 ($X=1$) 4级: 较易操作 ($0.7 \leq X < 1$) 3级: 可操作 ($0.5 \leq X < 0.7$) 2级: 较难操作 ($0.3 \leq X < 0.5$) 1级: 难操作 ($0 \leq X < 0.3$)
	易学性	软件产品使用户能学习其应用的能力	$0 \leq X$, 单位: 秒	$T = \text{学习正确使用一项功能要用的平均时间} \quad (0 < T)$ 单位: 秒
维护性	易分析性	软件产品诊断软件中的缺陷或失效原因以及标识待修改部分的能力	$0 \leq X \leq 1$	$X = 1 - \sum A_i \quad (0 \leq X \leq 1)$ $A_i = \text{不利于软件产品分析的因素}$ 5级: 极易分析 ($X=1$) 4级: 较易分析 ($0.85 \leq X < 1$) 3级: 可分析 ($0.7 \leq X < 0.85$) 2级: 较难分析 ($0.5 \leq X < 0.7$)

				1级：难分析 ($0 \leq X < 0.5$)
	易 改 变 性	软件产品使指定的修改可以被实现的能力	$0 \leq X \leq 1$	$X = 1 - \sum A_i$ ($0 \leq X \leq 1$) A_i = 不利于软件产品修改的因素 5级：极易分析 ($X = 1$) 4级：较易分析 ($0.85 \leq X < 1$) 3级：可分析 ($0.7 \leq X < 0.85$) 2级：较难分析 ($0.5 \leq X < 0.7$) 1级：难分析 ($0 \leq X < 0.5$)
	稳 定 性	软件产品避免由于软件修改而造成意外结果的能力	$0 \leq X \leq 1$	$X = 1 - \sum A_i$ ($0 \leq X \leq 1$) A_i = 不利于软件产品分析的因素 5级：极易分析 ($X = 1$) 4级：较易分析 ($0.85 \leq X < 1$) 3级：可分析 ($0.7 \leq X < 0.85$) 2级：较难分析 ($0.5 \leq X < 0.7$) 1级：难分析 ($0 \leq X < 0.5$)
	易 测 试 性	内置测试功能的有效性	$0 \leq X \leq 1$	$X = A/B$ ($0 \leq X \leq 1$) A = 实际测试中使用了内置式测试方法或数据的测试用例数 B = 实际测试使用的测试用例数 5级：很有效 ($X = 1$) 4级：较有效 ($0.85 \leq X < 1$) 3级：有效 ($0.7 \leq X < 0.85$) 2级：效果较差 ($0.5 \leq X < 0.7$) 1级：无效 ($0 \leq X < 0.5$)
		软件产品使已修改软件能被确认的能力	$0 \leq X$, 单位:	$X = \text{Sum}(T) / N$ $0 \leq X$ T = 为确认是否已经解决所报告的失效而花费的测试时间。 N = 解决的失效总数
可 移 植 性	适 应 性	软件产品无需采用有别于为考虑该软件的为目的而准备的活动或手段就可能适应不同的指定环境的能力	$0 \leq X \leq 1$	$X = 1 - A/B$ ($0 \leq X \leq 1$) A = 在结合环境硬件进行运行测试时未能完成任务的运行功能数 B = 进行测试的功能总数 5级：好 ($X = 1$) 4级：较好 ($0.85 \leq X < 1$) 3级：中 ($0.7 \leq X < 0.85$) 2级：较差 ($0.5 \leq X < 0.7$) 1级：差 ($0 \leq X < 0.5$)
	易 安 装 性	软件产品在指定环境中被安装的能力	$0 \leq X \leq 1$	$X = A/B$ ($0 \leq X \leq 1$) A = 用户为自己的方便成功地改变安装操作的次数 B = 用户为自己的方便企图改变安装操作

			<p>的总次数</p> <p>5级：好 ($X=1$)</p> <p>4级：较好 ($0.85 \leq X < 1$)</p> <p>3级：中 ($0.7 \leq X < 0.85$)</p> <p>2级：较差 ($0.5 \leq X < 0.7$)</p> <p>1级：差 ($0 \leq X < 0.5$)</p>
共存性	软件产品在公共环境中同与其分享公共资源的其它独立软件共存的能力	$0 \leq X$	<p>$X=A/T$ ($0 \leq X$)</p> <p>A=在与其他软件共同运行时，用户遇到的限制或意外失效的次数</p> <p>T=与其他软件共同运行的时间</p> <p>5级：好</p> <p>4级：较好</p> <p>3级：中</p> <p>2级：较差</p> <p>1级：差</p>
易替换性	软件产品在同样环境下，替代另一个相同用途的指定软件产品的能力	$0 \leq X \leq 1$	<p>$X=A/B$ ($0 \leq X \leq 1$)</p> <p>A=在新版软件中产生类似结果而无需变换的功能数</p> <p>B=在替代软件中有类似功能并已测试过的功能数</p> <p>5级：极易替换 ($X=1$)</p> <p>4级：较易替换 ($0.85 \leq X < 1$)</p> <p>3级：可替换 ($0.7 \leq X < 0.85$)</p> <p>2级：较难替换 ($0.5 \leq X < 0.7$)</p> <p>1级：难替换 ($0 \leq X < 0.5$)</p>
有效性	软件产品在指定的使用条件下，使用户能准确和完整地获得规定目标的能力	$0 \leq X \leq 1$	<p>$X=A/B$ ($0 \leq X \leq 1$)</p> <p>A=软件产品已完成任务数</p> <p>B=软件产品的总任务数</p> <p>5级：极有效 ($X=1$)</p> <p>4级：较有效 ($0.85 \leq X < 1$)</p> <p>3级：有效 ($0.7 \leq X < 0.85$)</p> <p>2级：有效性较差 ($0.5 \leq X < 0.7$)</p> <p>1级：有效性差 ($0 \leq X < 0.5$)</p>
生产率	是在特定使用环境中用户为达到有效性而花费的资源	{5, 4, 3, 2, 1}	<p>5级：高 ($X=5$)</p> <p>4级：较高 ($X=4$)</p> <p>3级：中 ($X=3$)</p> <p>2级：较低 ($X=2$)</p> <p>1级：低 ($X=1$)</p>
安全性	在特定使用环境下对人、商业、软件、属性或环境产生危害的风险级别	$0 \leq X \leq 1$	<p>$X=1-A/B$ ($0 \leq X \leq 1$)</p> <p>A=发生损失的次数</p> <p>B=使用总次数</p> <p>5级：很安全 ($X=1$)</p> <p>4级：较安全 ($0.85 \leq X < 1$)</p>

			3级：安全 ($0.7 \leq X < 0.85$) 2级：较危险 ($0.5 \leq X < 0.7$) 1级：危险 ($0 \leq X < 0.5$)
满意度	软件产品在指定的使用条件下，使用户满意的能力	{5, 4, 3, 2, 1}	5级：很满意 (X=5) 4级：较满意 (X=4) 3级：满意 (X=3) 2级：不太满意 (X=2) 1级：不满意 (X=1)

5.5 用户信任关系信息

用于描述、建立和维护用户之间的信任关系，为软件资源的可信性评估提供支持。包括两部分：

- 用户信任关系的建立和维护：用户可以自定义其对不同反馈者的信任度，同时维护一个定制化的信赖/不信赖用户集合。
- 系统中不同反馈者的影响因子（可信度）分析：基于不同用户设定的信任关系，建立系统的信任关系网络，通过聚类分析算法，分析得到系统中不同反馈者的影响因子（可信度）。

用户信任关系信息包含以下信息：

名称	类型	约束	备注说明	来源
反馈者 ID (feedbackUserID)	String	必须提供		用户填写信息
信任度 (trustWeight)	Float	必须提供	0 到 1 之间的浮点数，值越大表示信任度越高	
描述 (trustDescription)	String	必须提供	一段文字，用于表达信任的依据等信息	
用户 ID(UserID)		可选		系统自动生成
信任关系 (trustRelationKey)		必须提供	信任关系的标识符	
是否有效(isValid)	布尔值	必须提供	为了便于检索建立的索引	
发布时间 (publishTime)		必须提供		
更新时间		必须提供		

名称	类型	约束	备注说明	来源
(updateTime)				

5.6 资源库统计信息

资源库统计信息用于统计资源库中资源的状况，帮助用户迅速了解资源库的概况，该信息包括：

1、库总体信息

- 用户总数
- 资源总数
- 资源包总数
- 用户反馈总次数
- 可获取 QoS 信息的 Service 个数
- QoS 信息获取数量
- 质量评估次数
- 可信评估模型个数
- 下载总次数
- 复用总次数
- 可信分级覆盖率（xx 个构件已被分级，每一级分别有多少个）
- 价格信息统计（free: xx 个、open: xx 个、business: xx 个）

2、考虑资源分类（构件、服务、工具、构架和其他）的信息

	已发布资源个数	累计被复用次数	可信分级统计	价格信息统计
构件		xx（按下载次数计数）	1 级: xx 个 2 级: xx 个 3 级: xx 个 4 级: xx 个 5 级: xx 个（按分级进行查看）	free: xx 个 open: xx 个 business: xx 个
服务		xx（按生成客户端次数计数）	1 级: xx 个 2 级: xx 个 3 级: xx 个 4 级: xx 个 5 级: xx 个（按分级进行查看）	free: xx 个 open: xx 个 business: xx 个
工具		xx（按下载次数计数）	1 级: xx 个 2 级: xx 个 3 级: xx 个 4 级: xx 个 5 级: xx 个（按分级进行查看）	free: xx 个 open: xx 个 business: xx 个

	已发布资源个数	累计被复用次数	可信分级统计	价格信息统计
				个

3、关于资源个体的统计

- 用户发布资源数
- 用户发布反馈次数
- 用户复用次数
- 资源 QoS 信息获取次数
- 资源质量评估次数
- 资源复用次数

5.7 可信证据信息

软件资源质量可信评估是基于相关证据的。这里称其为可信证据，即为确定软件质量可信度而提供证明的根据。可信证据是指具有特定形式，能够经过查证确定属实，并用来证明软件资源质量真实情况的一切事实。可信证据必须具备以下三个特征：

(1)客观性，即证据是不以人们的意志为转移的客观存在的事实。这是证据的本质特征。

(2)关联性，或称相关性，是指证据和需要证明的质量可信度间有一定的关系或联系。

(3)可获得性，是指证据是可以依照确定的程序收集、审查、判断、获得和检验的。

软件资源质量及可信评估证据框架是该机制的核心。这里称其为可信证据框架。本规范此部分遵循相应的《软件资源质量及可信评估 证据框架》。

用于评估软件资源可信性所需提交的可信证据信息分为开发阶段证据项、提交阶段证据项和使用阶段证据项，这三部分的证据信息及说明如下：

可信证据信息由资源发布者按照资源库所提供的可信证据项框架的相对应内容进行提交，之后由专家组对收集而来的软件资源的可信证据进行综合评审，最终给定软件资源的可信级别。可信证据的收集过程和评价过程详见 5.6 节。

1、开发阶段证据项由 8 项信息构成，具体说明如下：

分类	证据名称	结果	支撑附件	说明
开发阶段	过程管理规范	字符串 (300 字符)	软件资源过程管理文档	1) 本证据项取值为本软件资源开发阶段所采用的过程管理规范名称； 2) 所采用的过程管理规范可以是ISO9000 或 CMMI 等被业界普遍认同的标准或规范，也可以是开发单位自主制定的特定规范。 3) 本软件资源过程管理文档作为附件一并提交
	需求变更数	整型	需求变更记录文档	1) 本证据项取值为软件资源开发阶段需求项增加、删除和修改的总数 2) 需求变更记录作为附件一并提交。
	需求评审结论	字符串 (100 字符)	需求评审报告	1) 本证据项取值为需求评审结论简述 2) 需求评审报告作为附件一并提交
	设计方法	字符串 (100 字符)		1) 软件资源设计方法的描述
	设计评审结论	字符串 (100 字符)	设计评审报告	1) 本证据项取值为设计评审结论简述 2) 设计评审报告作为附件一并提交
	编码规范	字符串 (100 字符)	编码规范文档	1) 本证据项取值为编码规范简要说明 2) 编码规范涵盖编码风格规范和异常处理规范 3) 编码规范文档作为附件一并提交。
	测试缺陷趋势	{ 开始时间: 时间型; 结束时间: 时间型; 缺陷密度: 浮点型; }		1) 本证据项取值为的一组描述测试缺陷密度变化趋势的数据，按时间顺序，每个数据为一个三元组，包括开始时间、结束时间和测试缺陷密度。 2) 测试缺陷密度定义为：该时间段内测试发现代码缺陷总数与被测软件实体代码行数的比值。
	缺陷清除率	浮点型		1) 缺陷清除率定义为：软件资源开发阶段已清除的缺陷总数占应清除的缺陷总数的百分比。

分类	证据名称	结果	支撑附件	说明

2、提交阶段证据项由 12 项信息构成，具体说明如下：

分类	证据名称	结果	支撑附件	说明
测试结果	测试类型与结果	{ 测试类型： 字符串 (20 字符)； 测试结果： 字符串 (100 字符)； }	测试报告	1) 本证据项取值为一组关于测试类型及其测试结果的文字描述,包括测试类型描述和对应测试结果描述 2) 测试类型包括功能测试、性能测试、可靠性测试、安全性测试等; 3) 测试结果是对所进行的测试内容和测试结论的简短描述。如功能测试的测试结果可包括对功能测试的覆盖率和正确率的描述。 4) 该证据项源自于软件测试报告,测试报告作为附件一并提交。
	残留缺陷列表	字符串(100 字符)	测试报告	1) 该证据项是对残留缺陷数的统计值。 2) 残留缺陷列表作为附件一并提交,残留缺陷附件与测试报告可以是同一附件。
测试需求	测试准则	字符串(100 字符)	测试报告	1) 该证据项是对测试要求及其依据的简单描述。如: 对于安全软件,依据行业标准,必须通过单元测试且语句覆盖和分支覆盖需达到100%等。 2) 该证据项源自于软件测试报告,测试报告作为附件一并提交。
	测试需求	测试类型及其测试需求	测试报告	1) 该证据项是对测试需求的简单描述。如需通过测试来确认的质量特性。 2) 该证据项源自于软件测试报告,测试报告作为附件

分类	证据名称	结果	支撑附件	说明
				一并提交。
测试过程	测试方法	字符串(100字符)	测试报告	1) 该证据项是对所采用测试方法的简单描述 2) 该证据项源自于软件测试报告, 测试报告作为附件 一并提交。 3) 面向功能组件结构的测试方法: 白盒测试、黑箱测试、灰盒测试 4) 面向验证符合设计标准的测试方法: 功能测试性、边界测试、使用界面测试、可用性测试、安全型测试、试用版测试 5) 面向检验覆盖面的测试方法: 单元测试、系统测试或集成测试、压力测试、随机任意性测试、全程测试、回归测试、自动化测试
	测试人员	字符串(100字符)	测试报告	1) 该证据项是对测试组织和测试人员及其测试能力等级的描述; 2) 该证据项源自于软件测试报告, 测试报告作为附件 一并提交。
	测试工具	字符串(100字符)	测试报告	1) 该证据项是对采用的测试工具的描述。 2) 该证据项源自于软件测试报告, 测试报告作为附件 一并提交。
	测试环境	字符串(100字符)	测试报告	1) 该证据项是对各种测试环境的描述。 2) 该证据项源自于软件测试报告, 测试报告作为附件一并提交。
	测试过程记录	测试过程记录	测试报告	1) 该证据项是对测试过程追踪和记录的描述。 2) 该证据项源自于软件测试记录, 测试记录作为附件 一并提交。
分	死锁检测	整型	分析报告	1) 本证据项取值为未处理的死锁缺陷数

分类	证据名称	结果	支撑附件	说明
析证据				量； 2) 死锁缺陷包括同步类缺陷，如死锁；继承类缺陷，如方法重载缺陷；数据流缺陷，如空指针引用等。 3) 若有所采用的分析工具生动生成的分析报告，作为附件一并提交
	内存泄露检测		分析报告	1) 本证据项取值为未处理的内存泄漏缺陷数量； 2) 若有所采用的分析工具生动生成的分析报告，作为附件一并提交
	代码风格缺陷		分析报告	1) 本证据项取值为未处理的代码风格缺陷个数； 2) 若有所采用的分析工具生动生成的分析报告，作为附件一并提交

3、应用阶段证据项由 5 项信息构成，具体说明如下：

阶段	证据名称	结果	支撑附件	说明
使用阶段	使用规模	整型或浮点型	用户使用报告	1) 本证据项取值为资源被不同系统使用的次数或行业占有率。同一目标系统的多个拷贝只计算1次。可以通过用户合同、下载用户、下载次数等确定。 2) 若有用户提供的具体使用情况报告，作为附件一并提交
	应用领域	字符串（20 字符）		1) 本证据项取值为资源被使用的应用领域名称。应用领域将与相应的可信评估方法有关。此证据可由资源发布者或使用者填写。
	用户满意度	枚举型[很好、较好、中、较差、很差]	用户满意度详细说明	1) 本证据项取值为用户满意度枚举值，可由资源发布者或使用者填写。 2) 若用用户提供的更具体的满意度说明，作为附件一并提交。
	应用案例	字符串(100 字符)	详细应用情况介绍	1) 本证据项是对应用案例的简短描述，可由资源发布者或使用者填写。应用案例应能够反映资源的质量特性，以便可信评估。 2) 若有更为详细应用情况介绍，作为附件

阶段	证据名称	结果	支撑附件	说明
				一并提交。
	第三方评测论证情况	字符串(100字符)	第三方评测报告	1) 本证据项是对权威或专业第三方对本软件资源评价信息的简短描述,如关于奖励情况、评测情况、网上排名或被应用次数等的描述。 2) 与第三方评测相关的原始文档(获奖证书、评测报告等),作为附件一并提交。

可信证据的提交请参见 5.6 节。

5.8 其它：资源关系描述信息等

为描述若干软件资源构成一个特定系统（软件构架）和软件生产线系统的构成，以及提供系统的整体下载，这部分会扩展使用 RAS 机制，将资源关系描述和使用放在资源库外部。

需要提供资源关系描述机制和相关信息，在资源库中单独设立资源关系描述机制，允许用户描述和使用。

第6章 软件资源管理及使用过程

6.1 用户管理

用户管理用于管理用户注册信息，根据用户权限决定用户操作的合法性与否则。资源库用户使用本系统前应首先注册成为系统的合法用户，并且登录成功方可使用资源库提供的诸如发布资源、反馈评估等功能。用户管理分为用户注册、用户登录、用户信息修改、用户权限控制等功能。

6.1.1 用户注册

用户注册分为“个人用户”与“企业用户”注册。用户点击页面左上方或登录框下的注册“个人用户”或“企业用户”链接，进入注册页面：

个人用户注册时需填写如下信息：

- 用户名：登录时使用，5-8 位字符。
- 密码及确认密码：确保用户名不被他人使用，5-8 位字符。
- 查询密码问题及查询密码答案：忘记密码时给予提示，可不填。
- 用户真实姓名：请正确填写，在与用户联系时确认身份。
- 电子邮箱：忘记密码时向该邮箱发送密码。
- 联系电话：可为空。
- 联系地址：可为空。
- 传真：可为空。

企业用户注册时需填写如下信息：

- 用户名：登录时使用，5-8 位字符。
- 密码及确认密码：确保用户名不被他人使用，5-8 位字符。
- 查询密码问题及查询密码答案：忘记密码时给予提示，可不填。
- 企业名称：请正确填写企业的名称。
- 企业联系人：请正确填写，在企业用户联系时确认身份。
- 电子邮箱：忘记密码时向该邮箱发送密码。
- 联系电话：可为空。

- 联系地址：可为空。
- 传真：可为空。

填写完毕后点击“提交”，信息若通过验证则注册成功。若要重新填写信息，点击“重置”则清空输入的信息。

6.1.2 用户登录

注册成功后，只有登录资源库系统才能使用除查询外的各项功能。正确填写用户名和密码，即可登录。

若忘记密码，点击页面上的“忘记密码”。输入注册时使用的用户名，然后根据用户注册时填写的查询密码问题回答，若回答正确系统将把密码发送至注册时填写的电子邮箱。注册时没有填写查询密码问题的用户可以发送电子邮件给系统管理员。

6.1.3 用户信息修改

注册用户成功登录系统后，可以修改注册密码及其他个人信息。

- 修改密码：输入新密码。新密码将会在注销本次登录后生效。
- 修改个人信息：用户可以修改查询密码问题、查询密码答案、真实姓名、电子邮箱、联系电话、联系地址和传真。

6.1.4 用户权限的控制

用户权限控制用于保证用户在资源库中的操作是否合法，只有拥有合法的权限，用户才能在资源库中进行资源管理、资源检索、反馈信息发布等操作。

6.2 系统管理员

系统管理员实施更上层的管理功能，系统管理员具有普通用户的所有功能，除此而外，系统管理还具有以下功能：

- 用户停用与启用：管理员可以停用在资源库中已注册的用户账号，某个用户被停用后，该用户不能登录资源库，并提示用户其账号已经被停用，请用户与管理联系。但管理员仍然可以查看该用户的相关信息。当管理员认为有必要时，可以启用任一被停用的用户帐户信息，启用后，该用户可以重新登录资源库。

-
- 资源删除：当资源存在较大问题时、某个服务已被证明无限期停止服务时、资源被重复发布时，管理员均可删除资源信息。删除资源信息时，与资源信息有关的基本信息、分类信息、扩展属性信息、tag 信息、反馈信息、RAS 信息和 RAS 包均被删除。若资源是推荐资源，则同时在推荐资源列表中删除，并重新计算推荐资源列表。
 - 删除 Tag：管理员可以删除某一 Tag，删除时，会将所有与该 tag 相关的资源与 Tag 的关联记录的值清空。
 - 删除日志：管理员的任何删除操作，均被记录形成删除日志，以便在未来进行查询。

6.3 发布资源

用户可将构件、服务、工具、构架等资源，通过资源库进行发布，这样资源能被其它用户使用，实现资源的复用。

资源是资源库中被管理的最重要的信息。软件资源可以用于创造新的软件产品以及相关的制品。资源信息包括资源的基本信息、实体信息和质量信息。资源的基本信息描述资源基本属性的信息，包括资源名称、资源类型、分类信息、属性信息等，用于理解和搜索定位资源；资源实体信息则是与资源相关的资源实体的访问地址或下载地址，由于一个资源可能含多种相关文件（如需求文档、设计文档、测试文件等），资源库中提供了符合 OMG 标准的 RAS (Reusable Asset Specification) 规范的资源实体上传方式；资源的质量信息则是由资源的发布者或资源的使用者依据资源库提供的符合 ISO/IEC9126 标准的质量模板进行资源质量信息发布，用以帮助用户依据资源的质量进行资源的选择与复用。

6.3.1 发布资源基本信息

用户登陆后即可发布资源的基本信息，该信息包括以下内容：

- 资源名称：必填信息，一个资源必须有一个名称，名称不能为空。
- 资源类型：必填信息，一个资源必须有所属的类型，可从构件、服务、工具、构架或其他五种选项中进行选择。
- 资源描述：必填信息项，资源的相关描述，对资源的一般性解释。
- 资源的分类信息：可选信息。用于标记资源的特定分类信息（如工业、

产品或地理代码) 的名/值对的可选列表。这一信息项可用于资源的检索。

- **资源标签 (Tag):** 可选信息, 用以简明标识资源的标签信息, 该信息可填写多个, 用逗号进行分割, 这一信息可用于资源的检索。
- **资源的许可证:** 可选信息, 默认值为无, 用以描述所发布的资源必须遵守的产权保护证书, 该证书可从常用许可证列表中进行选择, 也可由用户自行定义, 该项信息也可为空。(常用许可证列表信息见附件 1)。
- **费用:** 可选信息, 默认值为无, 包括了 **Open, Free** 和 **Commercial** 三类, 可由发布者自行选择。
- **价格:** 可选信息, 默认值为无, 当费用型为 **Commercial** 或 **Open** 时, 描述了用户为获得该资源所必须付出费用的数量 (以美元为单位)。
- **参考:** 可选信息, 包含了资源介绍信息源的声明, 例如 **WSDL** 等信息
- **限制:** 可选信息, 由发布资源者填写资源限制使用的因素; 包括算法、编译器、可移植性、环境等。
- **目标环境:** 可选信息, 资源开发所针对的计算机系统、操作系统、编译器等。
- **访问地址:** 可选信息, 当资源为软件服务时, 描述该服务的访问地址信息。
- **版本:** 可选信息, 资源对应的版本信息。
- **认证方法:** 可选信息, 描述了评价或认证资源的方法, 包括评价或认证的结果, 实施认证的日期, 实施认证的人员等。

填写完上述信息后, 即可进行下一步, 发布资源实体信息。

6.3.2 发布资源实体信息

资源实体信息是由发布者将与资源相关的制品实体进行上传, 上传方式符合 **OMG** 的 **RAS** 的规范。

发布者可以选择 2 种方式进行资源实体信息的发布, 其一, 可以选择直接将资源实体所有相关文件按照 **RAS** 规范进行整体打包, 然后将打包后的符合 **RAS** 规范的文件进行上传; 其二, 可以分别将与资源实体相关的需求文件、设计文件、实体文件、测试文件、帮助文件进行上传, 由资源库统一将这些文件按照 **RAS**

规范进行打包后存储。

资源实体信息包括以下信息：

- **需求文件：** 可选信息，可以是任何类型文件，上传的文件应该是与所发布资源的需求相关的文件，用户可以在该部分上传与资源需求相关的所有文件，例如资源开发过程中的需求分析文档、用例图、用例说明文档等。用户可以选择将所有这些文件打包后一次上传，也可以通过点击“继续添加需求相关文件”的链接添加多个文件。
- **设计文件：** 可选信息，可以是任何类型文件，上传的文件应该是与所发布资源的设计相关的文件，用户可以在该部分上传与资源开发生命周期中设计阶段相关的所有文件，例如资源开发过程中的系统设计文档、资源设计说明文档、UML 类图等。用户可以选择将所有这些文件打包后一次上传，也可以通过点击“继续添加设计相关文件”的链接添加多个文件。
- **资源实体文件：** 可选信息，可以是任何类型文件。上传的文件应该是实现资源功能的实体文件，如果发布的是“软件构件”，用户必须需要上传资源的实体文件 用户可以在该部分上传与资源功能实现有关的所有实体文件，例如资源的源代码、二进制代码等。用户可以选择将所有这些文件打包后一次上传，也可以通过点击“继续添加实现的代码文件”的链接添加多个文件。
- **测试文件：** 可选信息，可以是任何类型文件 上传的文件应该是与所发布资源的测试相关的文件，用户可以在此部分上传与所发布资源的测试相关的所有文件，例如资源的测试用例、测试报告等。用户可以选择将所有这些文件打包后一次上传，也可以通过点击“继续添加测试相关文件”的链接添加多个文件。
- **资源使用帮助文件：** 可选信息，可以是任何类型文件，上传的文件应该是有助于所发布资源的使用的说明文件 用户可以在此部分上传资源的使用帮助文件，例如资源的使用说明文档、用户手册等。用户可以选择将所有这些文件打包后一次上传，也可以通过点击“继续帮组文件”的链接添加多个文件。

-
- WSDL 文件的地址：字符串 必须是 WSDL 文件的有效 URL，如果发布的“Web Service”，用户必须提供服务的 WSDL 文件的 URL，前半部分下拉菜单中是供选择的 WSDL 的 URL 的通信协议类型，包括 http/https/ftp 三种。后半部分是 WSDL 的不包括通信协议部分的 URL 地址：比如完整某 WSDL 文件的 URL 为 http://www.sei.pku.edu.cn/test.wsdl ， 则 填 写 内 容 为 www.sei.pku.edu.cn/test.wsdl。

填写完上述信息后，即可进行下一步，发布资源质量信息。

6.3.3 发布质量信息

随着资源库系统的发展，资源数量将会增大，这将对用户选择资源带来一定的困难。为此，由资源发布者发布的资源的质量信息，可以帮助用户快速有效地对众多的资源进行理解和评估，辅助用户选择最合适的资源。

目前，资源质量信息以 ISO/IEC9126 的质量模型为模板，由资源的发布者进行资源的发布。

资源质量信息的模板及各个属性的取值范围如下：

■ 功能性：

- 适合性：软件产品为指定的任务和用户目标提供一组合适的功能的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表适合性最差，5 级代表适合性最好，由发布者根据所发布资源的质量据实填写。
- 准确性：软件产品提供具有所需精确度的正确或相符的结果及效果的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表准确性最差，5 级代表准确性最好，由发布者根据所发布资源的质量据实填写。
- 互操作性：软件产品与一个或更多的规定系统进行交互的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表互操作性最差，5 级代表互操作性最好，由发布者根据所发布资源的质量据实填写。
- 保密安全性：软件产品保护信息和数据的能力，以使未经授权的人员

或系统不能阅读或修改这些信息和数据，而不拒绝授权人员或系统对它们的访问；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表保密安全性最差，5 级代表保密安全性最好，由发布者根据所发布资源的质量据实填写。

■ 可靠性

- 成熟性：软件产品为避免由软件中错误而导致失效的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表成熟性最差，5 级代表成熟性最好，由发布者根据所发布资源的质量据实填写。
- 容错性：在软件出现故障或者违反指定接口的情况下，软件产品维持规定的性能级别的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表容错性最差，5 级代表容错性最好，由发布者根据所发布资源的质量据实填写。
- 易恢复性(可用性)：在失效发生的情况下，软件产品重建规定的性能级别并恢复受直接影响的数据的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易恢复性(可用性)最差，5 级代表易恢复性(可用性)最好，由发布者根据所发布资源的质量据实填写。

■ 效率

- 时间特性：在规定条件下，软件产品执行其功能时，提供适当的响应和处理时间以及吞吐率的能力；该属性的取值为空或具体的软件资源的响应时间，单位为秒，由发布者根据所发布资源的质量据实填写。
- 资源利用性：在规定条件下，软件产品执行其功能时，使用合适的资源数量和类别的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表资源利用性最差，5 级代表资源利用性最好，由发布者根据所发布资源的质量据实填写。

■ 易用性

- 易理解性：软件产品使用户能理解软件是否合适以及如何能将软件用于特定的任务和使用环境的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易理解性最差，5 级代表易理解性最好，

由发布者根据所发布资源的质量据实填写。

- 吸引力：软件产品吸引用户的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表吸引力最差，5 级代表吸引力最好，由发布者根据所发布资源的质量据实填写。
- 易操作性：软件产品使用户能操作和控制它的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表保密易操作性最差，5 级代表易操作性最好，由发布者根据所发布资源的质量据实填写。
- 易学性：软件产品使用户能学习其应用的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易学性最差，5 级代表易学性最好，由发布者根据所发布资源的质量据实填写。

■ 维护性

- 易分析性：软件产品诊断软件中的缺陷或失效原因以及标识待修改部分的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易分析性最差，5 级代表易分析性最好，由发布者根据所发布资源的质量据实填写。
- 易改变性：软件产品使指定的修改可以被实现的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易改变性最差，5 级代表易改变性最好，由发布者根据所发布资源的质量据实填写。
- 稳定性：软件产品避免由于软件修改而造成意外结果的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表稳定性最差，5 级代表稳定性最好，由发布者根据所发布资源的质量据实填写。
- 易测试性：软件产品使已修改软件能被确认的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易测试性最差，5 级代表易测试性最好，由发布者根据所发布资源的质量据实填写。

■ 可移植性

- 适应性：软件产品无需采用有别于为考虑该软件的目的地而准备的活动或手段就可能适应不同的指定环境的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表适应性最差，5 级代表适应性最好，由发布者根据所发布资源的质量据实填写。

-
- 易安装性：软件产品在指定环境中被安装的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易安装性最差，5 级代表易安装性最好，由发布者根据所发布资源的质量据实填写。
 - 共存性：软件产品在公共环境中同与其分享公共资源的其它独立软件共存的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表共存性最差，5 级代表共存性最好，由发布者根据所发布资源的质量据实填写。
 - 易替换性：软件产品在同样环境下，替代另一个相同用途的指定软件产品的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易替换性最差，5 级代表易替换性最好，由发布者根据所发布资源的质量据实填写。
- 使用质量
- 有效性：软件产品在指定的使用条件下，使用户能准确和完整地获得规定目标的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表有效性最差，5 级代表有效性最好，由发布者根据所发布资源的质量据实填写。
 - 生产率：软件产品在指定的使用条件下，使用户能花费和有效性有关的合适资源数量的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表生产率最差，5 级代表生产率最好，由发布者根据所发布资源的质量据实填写。
 - 安全性：软件产品在指定使用条件下，获得可接受的损害人类、事务、软件、财产或环境风险级别的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表安全性最差，5 级代表安全性最好，由发布者根据所发布资源的质量据实填写。
 - 满意度：软件产品在指定的使用条件下，使用户满意的能力。该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表满意度最差，5 级代表满意度最好，由发布者根据所发布资源的质量据实填写。

填写完资源的基本信息、实体信息和质量信息后，即完成资源信息的发布，可以将信息提交到资源库进行存储。

6.4 检索资源

资源检索部分的主要功能，是帮助用户根据自己的需要查询、浏览、选取和提取资源库中的资源和其他实体信息。可以进行资源查询的用户有两类，一类是缺省的游客用户（guest），一类是注册登录用户。不同的用户查询资源的权限不同。

查询的基本步骤为：选择查询类型（简单或复杂）；填写查询条件；查询；查看详细的查询结果。

6.4.1 简单检索

简单检索是一种快捷简便地帮助用户进行资源检索的方式，用户对不同种类的资源按名称进行查询，简单检索需输入以下查询条件：

- 资源类型：用户进行资源检索时，需要选择所检索的资源类型，分别为构件、服务、工具、构架或其他类型，默认的资源类型为构件。
- 名称：要查询内容的名字。可以一次输入多个名称，多个名称之间以空格分隔（对于规范的查询仅支持单个名字，即包括空格在内的所有字符都将被视为一个规范名称），它们之间是“逻辑或”关系；每个名称中可以使用通配符“%”，即使通配符没有输入，也默认存在（如：输入“A”，相当于“A%”）。不输入名称则列出所有查询结果。

当上述检索条件输入完毕后，直接点击搜索按钮进行检索即可。

若上述信息不填写，则系统默认检索结果为资源库中的所有的资源。

6.4.2 复杂检索

相对于简单检索，复杂检索可以增加检索的约束条件，减小检索的范围，以便于更准确地定位检索目标。复杂检索可以通过资源库首页的高级检索链接直接进入。

复杂检索需输入以下查询条件：

- 资源类型：用户进行资源检索时，需要选择所检索的资源类型，分别为构件、服务、工具、构架或其他类型，默认的资源类型为构件。
- 名称：要查询内容的名字。可以一次输入多个名称，多个名称之间以空

格分隔（对于规范的查询仅支持单个名字，即包括空格在内的所有字符都将被视为一个规范名称），它们之间是“逻辑或”关系；每个名称中可以使用通配符“%”，即使通配符没有输入，也默认存在（如：输入“A”，相当于“A%”）。不输入名称则列出所有查询结果。

- 发布者：指定搜索某个发布者发布的资源；
- 标签（Tag）：可以按照资源所带的标签信息进行检索，可以同时输入多个标签，多个标签之间用逗号分割；
- 分类信息：从分类树选择某一分类术语进行资源的准确定位。
- 质量信息：可以依据质量模板填写所需检索的资源的质量约束，指定搜索资源的质量值必须不低于填写的值。

填写完上述信息后，直接点击提交按钮进行复杂检索即可。

若上述信息不填写，则系统默认检索结果为资源库中的所有的资源。

6.4.3 基于 Tag（标签）检索

为了更简便地进行资源的搜索，资源库提供基于 Tag（标签）检索，用户可以点击首页上的任一 Tag 进行资源的检索。资源库根据用户选择的标签，将查询结果列表返回给用户，用户可在列表中选择任一资源查看详情。

6.4.4 检索资源的结果显示

检索资源结果的显示按照资源发布的时间先后进行显示，且在一页中仅展示 5 个结果，若检索结果大于 5 个，则分页进行显示，用户可以点击分页的页码，转到相应的页面，再点击查询结果的名称，查看详细的检索结果。

检索结果为符合检索条件的资源列表，直接点击所显示的资源名称即可查看资源的详细信息，若需要提取资源实体信息，可以直接点击相应资源的下载地址即可提取相关的实体信息等，此时需要提供资源在本地存放的位置信息。

6.5 反馈评估

随着资源库系统的发展，资源数量将会增大，这将对用户选择资源带来更大的困难。于是我们建立了反馈评估子系统，允许用户发布使用资源后的评价，这样可以帮助用户快速有效地对众多的资源进行理解和评估，辅助用户选择最合

适的资源。

资源的反馈信息：用户在复用资源后，将对资源的总体满意程度、总体评价和基于 ISO/IEC9126 的软件质量模型的反馈模板中的诸多质量信息进行评价，该信息用以辅助资源提供商提高资源的质量，并且辅助用户更好的选择合适的软件资源。

反馈信息发布者需要提交的反馈信息如下：

- 总体满意程度：表明反馈信息发布者对于其所使用的资源的满意度的总体评价，即与用户的期望相符合的程度。该属性的取值为空或 1，2，3，4，5 这 5 个级别，1 级代表总体满意程度最差，5 级代表总体满意程度最好，由发布者根据其所使用资源后的满意程度据实填写。
- 总体评价：表明反馈信息发布者对于其所使用的资源的总体评价信息，该信息为文本信息，由用户根据其所使用资源后的体验据实详细填写。
- 功能性：
 - 适合性：软件产品为指定的任务和用户目标提供一组合适的功能的能力；该属性的取值为空或 1，2，3，4，5 这 5 个级别，1 级代表适合性最差，5 级代表适合性最好，由发布者根据所发布资源的质量据实填写。
 - 准确性：软件产品提供具有所需精确度的正确或相符的结果及效果的能力；该属性的取值为空或 1，2，3，4，5 这 5 个级别，1 级代表准确性最差，5 级代表准确性最好，由发布者根据所发布资源的质量据实填写。
 - 互操作性：软件产品与一个或更多的规定系统进行交互的能力；该属性的取值为空或 1，2，3，4，5 这 5 个级别，1 级代表互操作性最差，5 级代表互操作性最好，由发布者根据所发布资源的质量据实填写。
 - 保密安全性：软件产品保护信息和数据的能力，以使未经授权的人员或系统不能阅读或修改这些信息和数据，而不拒绝授权人员或系统对它们的访问；该属性的取值为空或 1，2，3，4，5 这 5 个级别，1 级代表保密安全性最差，5 级代表保密安全性最好，由发布者根据所

发布资源的质量据实填写。

■ 可靠性

- 成熟性：软件产品为避免由软件中错误而导致失效的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表成熟性最差，5 级代表成熟性最好，由发布者根据所发布资源的质量据实填写。
- 容错性：在软件出现故障或者违反指定接口的情况下，软件产品维持规定的性能级别的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表容错性最差，5 级代表容错性最好，由发布者根据所发布资源的质量据实填写。
- 易恢复性(可用性)：在失效发生的情况下，软件产品重建规定的性能级别并恢复受直接影响的数据的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易恢复性(可用性)最差，5 级代表易恢复性(可用性)最好，由发布者根据所发布资源的质量据实填写。

■ 效率

- 时间特性：在规定条件下，软件产品执行其功能时，提供适当的响应和处理时间以及吞吐率的能力；该属性的取值为空或具体的软件资源的响应时间，单位为秒，由发布者根据所发布资源的质量据实填写。
- 资源利用性：在规定条件下，软件产品执行其功能时，使用合适的资源数量和类别的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表资源利用性最差，5 级代表资源利用性最好，由发布者根据所发布资源的质量据实填写。

■ 易用性

- 易理解性：软件产品使用户能理解软件是否合适以及如何能将软件用于特定的任务和使用环境的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易理解性最差，5 级代表易理解性最好，由发布者根据所发布资源的质量据实填写。
- 吸引力：软件产品吸引用户的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表吸引力最差，5 级代表吸引力最好，由发

布者根据所发布资源的质量据实填写。

- 易操作性：软件产品使用户能操作和控制它的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表保密易操作性最差，5 级代表易操作性最好，由发布者根据所发布资源的质量据实填写。
- 易学性：软件产品使用户能学习其应用的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易学性最差，5 级代表易学性最好，由发布者根据所发布资源的质量据实填写。

■ 维护性

- 易分析性：软件产品诊断软件中的缺陷或失效原因以及标识待修改部分的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易分析性最差，5 级代表易分析性最好，由发布者根据所发布资源的质量据实填写。
- 易改变性：软件产品使指定的修改可以被实现的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易改变性最差，5 级代表易改变性最好，由发布者根据所发布资源的质量据实填写。
- 稳定性：软件产品避免由于软件修改而造成意外结果的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表稳定性最差，5 级代表稳定性最好，由发布者根据所发布资源的质量据实填写。
- 易测试性：软件产品使已修改软件能被确认的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易测试性最差，5 级代表易测试性最好，由发布者根据所发布资源的质量据实填写。

■ 可移植性

- 适应性：软件产品无需采用有别于为考虑该软件的目的地而准备的活动或手段就可能适应不同的指定环境的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表适应性最差，5 级代表适应性最好，由发布者根据所发布资源的质量据实填写。
- 易安装性：软件产品在指定环境中被安装的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易安装性最差，5 级代表易安装性最好，由发布者根据所发布资源的质量据实填写。

-
- 共存性：软件产品在公共环境中同与其分享公共资源的其它独立软件共存的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表共存性最差，5 级代表共存性最好，由发布者根据所发布资源的质量据实填写。
 - 易替换性：软件产品在同样环境下，替代另一个相同用途的指定软件产品的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表易替换性最差，5 级代表易替换性最好，由发布者根据所发布资源的质量据实填写。
- 使用质量
- 有效性：软件产品在指定的使用条件下，使用户能准确和完整地获得规定目标的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表有效性最差，5 级代表有效性最好，由发布者根据所发布资源的质量据实填写。
 - 生产率：软件产品在指定的使用条件下，使用户能花费和有效性有关的合适资源数量的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表生产率最差，5 级代表生产率最好，由发布者根据所发布资源的质量据实填写。
 - 安全性：软件产品在指定使用条件下，获得可接受的损害人类、事务、软件、财产或环境风险级别的能力；该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表安全性最差，5 级代表安全性最好，由发布者根据所发布资源的质量据实填写。
 - 满意度：软件产品在指定的使用条件下，使用户满意的能力。该属性的取值为空或 1, 2, 3, 4, 5 这 5 个级别，1 级代表满意度最差，5 级代表满意度最好，由发布者根据所发布资源的质量据实填写。

填写上述反馈信息后，可以将信息提交到资源库进行存储，有资源库根据各个反馈信息发布者发布的信息进行资源质量的评估，并按照系统默认的评估模板或用户定制的评估模板给出资源的最终评估结果。

6.6 软件资源可信证据的管理

软件资源库中可信证据是评价软件资源可信性的前提，软件资源库中可信证据包括了软件资源可信证据的提交与软件资源可信证据的查看和下载。

软件资源可信证据的提交是由资源的发布者或资源的使用者依据资源库提供的证据模板进行资源可信评估证据发布，用以支持可信分及评估。软件资源库按照软件资源可信证据框架，提供了与该框架所对应的软件可信证据信息提交的功能，具体实施如下：

1: 在软件资源库中下载 Excel 表格文件 `evidence.xls`，该文件用以用户填写软件资源可信证据项的相应信息；

2: 根据用户的所掌握的软件可信证据信息，在 `evidence.xls` 文件中写入相应信息，同时根据该文件中的超链接提示，将其他被引用文件与其所对应项建立关联关系；

3: 将用户所填写完毕的 `evidence.xls` 文件与所有引用的文件一起打成 `.zip` 包；

4: 在软件资源发布的第三阶段，即用户完成上传软件资源实体后，将前一步所生成的 `.zip` 包上传到资源库中，一个软件资源对应一个可信证据包，后面提交的会将前一次提交的覆盖。

查看和下载可信证据是提供给用户或专家进行可信证据查阅的功能，进入到所检索资源的详细信息页面后，在可信证据栏中，可以看到发布者上传的可信证据的浏览与下载链接。点击浏览，即会打开该资源的 `evidence.xls`，用以显示资源的证据详情。或点击下载，即会弹出对话框，提示用户下载 `evidence.xls` 文件与所有引用的文件一起打成 `.zip` 包。

软件资源可信评级过程则是由专家组依据可信证据、用户反馈信息、QoS 信息等共同评估软件资源的可信级别。

软件资源可信评级操作需 2 类人员协作完成，他们分别是专家组成员和系统管理员。下面分别对两类人员的操作进行介绍。

专家组成员

在软件资源发布者提交完可信证据包之后，评审专家组根据可信证据包的内容对软件资源进行可信评级，并将评审结果反馈到软件资源库中。其操作步骤如

下:

1. 专家组成员使用其用户名和密码登录到软件资源库中
2. 通过检索接口查找目标软件资源
3. 进入软件资源详情页面，在资源详情页面下方点击“增加反馈”链接
4. 进入资源反馈页面，在“总体满意度”上指定资源的可信级别
5. 提交

系统管理员

软件资源库在后台维护一个专家组成员列表，该列表记录了所有专家组成员的用户名。系统管理员根据列表中成员对资源的反馈确认资源的可信级别并提交。系统管理员具有两类提交操作：

- 批量提交
 1. 进入系统管理的可信评估管理页面
 2. 系统将所有专家成员提交的反馈列出
 3. 系统管理员经过确认，点击页面下方的“确认分级”按钮，完成评级流程
- 特定提交
 1. 通过检索接口查找目标软件资源
 2. 进入软件资源详情页面，在资源反馈部分，查看特定反馈
 3. 进入反馈查看详情页面，点击页面右侧的“确认分级”按钮，提交可信级别

经过系统管理员的确认和提交，软件资源的可信级别信息将显示在资源详情页面。

6.7 统计分析

统计分析用于统计资源库中资源、用户等数据，帮助用户更好地了解资源库，辅助用户更好地使用资源库。目前，统计分析信息分为 3 各部分：

- 资源库总体信息统计：显示软件资源库的资源、用户、资源的复用和下载情况，以及最新的资源发布情况等统计数据；

-
- 资源概况信息统计：资源概况中展示了各类软件资源的个数，并可以查看资源发布的统计趋势图等；
 - 可信评估信息统计：可信评估信息中展示了各个可信级别的资源总数、分布等统计数据。

其中，在资源库总体信息统计中，用户可以查阅的统计信息包括以下几个方面：

- 用户总数：统计资源库中用户的总个数。
- 资源总数：统计目前资源库中资源的总个数。
- 资源包总数：统计目前资源库中资源包的总个数。
- 下载总次数：统计资源库中资源被下载的总个数。
- 复用总次数：统计资源库中资源被复用的总个数。

为了更好地让用户了解资源发布的情况，统计分析还提供趋势图查询功能，目前有资源库概况统计趋势图查询和新增资源统计趋势图查询功能：

资源库概况统计趋势图查询：在资源概况统计中，提供资源库统计信息的趋势图查询功能，提供用户总数、资源总数和下载总次数统计信息的趋势图查询功能，用户点击进入上述选项后，输入想要查询的时间段，即可展现在特定时间段，用户总数、资源总数或下载总次数统计信息的趋势图。

新增资源统计趋势图查询：用户可以输入想要了解资源发布的时间段，点击新增构件、新增服务、新增工具、新增构架四个资源类型选项，即可显示相应的资源类型发布的趋势图。

资源概况信息统计信息中，用户可以查阅的统计信息包括以下几个方面：

- 资源分类信息及分布的趋势图：用户可以查询目前在资源库中各种不同类型的资源的个数以及累计被复用次数。同时，用户可以输入具体的时间段查看在某一时间段内，各种不同类型的资源的个数以及累计被复用次数。
- 按分类显示资源信息：用户可以在分类树中选择某个分类术语，依据该分类术语，系统会显示术语该分类的构件、服务、工具及构架的资源的个数以及累计被复用次数。同时，用户可以输入具体的时间段查看在某一时间段内，具有某一分类术语的各种不同类型的资源的个数以及累计

被复用次数。

- 价格信息统计： 用户可以根据资源费用的类别（分为 Free、 Open、 Commercial），查看构件、服务、工具、构架或其他资源的个数。

可信评估信息统计信息中，用户可以查阅的统计信息包括以下几个方面（注：此部分正在开发中）：

- 用户反馈总次数：统计资源库中所有用户发布的反馈信息数目，也可按照用户输入的具体的时间段查看在某一时间段内，用户反馈信息的发布趋势。
- 可获取 QoS 信息的 Service 个数：统计资源库中，可以获取 QoS 的服务个数，也可按照用户输入的具体的时间段查看在某一时间段内，可以获取 QoS 的服务的个数趋势。
- QoS 信息获取数量：统计资源库中，所有可以获取 QoS 信息的数量，也可按照用户输入的具体的时间段查看在某一时间段内，可以获取 QoS 信息的数量趋势。
- 质量评估次数：统计资源库中，所有资源质量评估次数，也可按照用户输入的具体的时间段查看在某一时间段内，资源质量评估次数的趋势。
- 可信分级覆盖率：按照可信分级（共有 6 级）的不同级别，查看不同级别的资源总个数，也可按照用户输入的具体的时间段查看在某一时间段内，不同级别的资源的发布趋势。
- 详细按资源类型可信分级信息：可以按照资源的类型，即构件、服务、工具和构架，按照可信的级别，统计资源库中，每个级别下不同种类资源的个数。

6.8 我的资源库

“我的资源库”用于统计资源库注册用户所发布的资源、下载的资源、浏览的资源以、发布的反馈信息等，便于注册用户掌握其所发布的信息。具体而言，用户可以通过“我的资源库”查看：

- “我发布的资源”：查看用户所有发布的资源，所列出的资源按照时间排序，用户可点击任一发布资源查看详情。

-
- “我下载过的资源”：查看用户所有下载过的资源，所列出的资源按照时间排序，用户可点击任一下载资源列表的任一资源查看详情。
 - “我浏览过的资源”：查看用户所有浏览过的资源，所列出的资源按照时间排序，用户可点击任一浏览资源列表的任一资源查看详情。
 - “对我发布的资源的反馈信息”：查看用户所有对其所发布的资源的反馈信息，所列出的反馈信息按反馈时间排序。用户可点击任一反馈信息列表的任一反馈信息查看详情。
 - “我发布过的反馈信息”：查看用户所有发布过的反馈信息，所列出的反馈信息按反馈时间排序。用户可点击任一反馈信息列表的任一反馈信息查看详情。

6.9 帮助信息

帮助信息起到帮助用户使用本系统并了解本系统的基本结构和基本功能的作用。帮助信息分为术语和使用指南两部分。

术语部分介绍了北京大学资源库系统中涉及到的专业术语。初次使用本系统的用户和在使用中遇到不理解的术语的用户可以在此处得到帮助，这里对系统中最核心的专业词汇作了解释。

使用指南介绍了本系统各部分的功能和用法。使用指南包括概述和资源发布、资源检索、分类法、反馈评估、统计信息、我的资源库六部分。每个部分均给出了辅助用户使用资源库的帮助信息。该部分的具体内容可以详见《资源库使用指南》。

6.10 中英文界面

软件资源库支持中英文界面，用户可以根据需要进入中文版的资源库或英文版的资源库。用户仅需点击资源库页面菜单条右上角的“中文版”或“English”即可切换到对应语言的界面。

附录 1：软件常见许可证列表

1、GPL-Compatible Free Software Licenses

GNU General Public License (GPL) version 3
GNU General Public License (GPL) version 2
GNU Lesser General Public License (LGPL) version 3
GNU Lesser General Public License (LGPL) version 2.1
GNU Affero General Public License (AGPL) version 3
Apache License, Version 2.0
Artistic License 2.0
Clarified Artistic License
Berkeley Database License (aka the Sleepycat Software Product License)
Boost Software License
Modified BSD license
CeCILL version 2
The Clear BSD License
Cryptix General License
eCos license version 2.0
Educational Community License 2.0
Eiffel Forum License, version 2
EU DataGrid Software License
Expat License
FreeBSD license
Freetype Project License
License of the iMatix Standard Function Library
Independent JPEG Group License
License of imlib2
Intel Open Source License
ISC License
Microsoft Public License (Ms-PL)
NCSA/University of Illinois Open Source License
License of Netscape Javascript
OpenLDAP License, Version 2.7
License of Perl 5 and below
Public Domain
License of Python 2.0.1, 2.1.1, and newer versions

License of Python 1.6a2 and earlier versions
License of Ruby
Standard ML of New Jersey Copyright License
License of Vim, Version 6.1 or later
W3C Software Notice and License
X11 License
XFree86 1.1 License
License of ZLib
Zope Public License, versions 2.0 and 2.1

2、 GPL-Incompatible Free Software Licenses



Affero General Public License version 1
Academic Free License, all versions through 3.0
Apache License, Version 1.1
Apache License, Version 1.0
Apple Public Source License (APSL), version 2
BitTorrent Open Source License
Original BSD license
Common Development and Distribution License (CDDL)
Common Public Attribution License 1.0 (CPAL)
Common Public License Version 1.0
Condor Public License
Eclipse Public License Version 1.0
IBM Public License, Version 1.0
Interbase Public License, Version 1.0
Jabber Open Source License, Version 1.0
LaTeX Project Public License 1.3a
LaTeX Project Public License 1.2
◀ Lucent Public License Version 1.02 (Plan 9 license)
Microsoft Reciprocal License (Ms-RL)
Mozilla Public License (MPL)
Netizen Open Source License (NOSL), Version 1.0
Netscape Public License (NPL), versions 1.0 and 1.1
Nokia Open Source License
Old OpenLDAP License, Version 2.3
Open Software License, all versions through 3.0
OpenSSL license
Phorum License, Version 2.0

PHP License, Version 3.01
License of Python 1.6b1 through 2.0 and 2.1
Q Public License (QPL), Version 1.0
RealNetworks Public Source License (RPSL), Version 1.0
Sun Industry Standards Source License 1.0
Sun Public License
License of xinetd
Zend License, Version 2.0
Zope Public License version 1

3、 Non-Free Software Licenses

Aladdin Free Public License
Apple Public Source License (APSL), version 1.x
Artistic License 1.0
AT&T Public License
eCos Public License, version 1.1
GPL for Computer Programs of the Public Administration
Hacktivism Enhanced-Source Software License Agreement (HESSLA)
Jahia Community Source License
Old license of ksh93
License of Lha
Microsoft Limited Public License (Ms-LPL)
Microsoft Limited Reciprocal License (Ms-LRL)
Microsoft Reference License
Microsoft's Shared Source CLI, C#, and Jscript License
Microsoft Windows Embedded CE 6.0 Shared Source License
NASA Open Source Agreement
Open Public License
License of PINE
Old Plan 9 license
Reciprocal Public License
Scilab license
SGI Free Software License B, version 1.1
Squeak license
Sun Community Source License
Sun Solaris Source Code (Foundation Release) License, Version 1.1
University of Utah Public License
YaST License

4、 Licenses For Documentation

■ Free Documentation Licenses

GNU Free Documentation License

Apple's Common Documentation License, Version 1.0

FreeBSD Documentation License

Open Publication License, Version 1.0

■ Non-Free Documentation Licenses

Open Content License, Version 1.0

Open Directory License (aka dmoz.org License)

5、 Licenses for Works of Practical Use Besides Software and Documentation

GNU General Public License

GNU Free Documentation License

Creative Commons Attribution 2.0 license (a.k.a. CC-BY)

Creative Commons Attribution-Sharealike 2.0 license (a.k.a. CC-BY-SA)

Design Science License (DSL)

Free Art License

6、 Licenses for Fonts

GNU General Public License

Arphic Public License

SIL Open Font License 1.1

7、 Licenses for Works of Opinion and Judgment

GNU Verbatim Copying License

Creative Commons Attribution-NoDerivs 3.0 license (a.k.a. CC-BY-ND)

Trustie技术规范

附录 2：软件资源库中英文术语对照表

- 软件资源库：Software Resource Library
- 用户信息：User Information
 - ◆ 用户：User
 - ◆ 用户 ID：User ID
 - ◆ 用户真实姓名 user real name
 - ◆ 密码：password
 - ◆ 邮箱：E-mail
 - ◆ 电话：Telephone Number
 - ◆ 传真：Fax
 - ◆ 联系方式：Contact
 - ◆ 企业名称：Corporate Name
 - ◆ 企业联系人 Corporate Contact Name
 - ◆ 企业联系方式：Corporate Contact Name
- 发布资源信息：User Information
 - ◆ 名称：Name
 - ◆ 类型：Type
 - ◆ 实体构件：Entity Components
 - ◆ 服务：Service
 - ◆ 构架：Architecture
 - ◆ 工具：Tools
 - ◆ 其他：Other
 - ◆ 描述：Description
 - ◆ 标签：Tag
 - ◆ 分类法：Classification
 - ◆ 软件许可证：Software License
 - ◆ 费用：fee
 - ◆ 开源：Open
 - ◆ 免费：Free
 - ◆ 商业：Commercial

-
- ◆ 引用：Reference
 - ◆ 约束：Limitation
 - ◆ 目标环境：Target Environment
 - ◆ 版本：Version
 - ◆ 认证方法：Authentication method
 - ◆ 访问地址：Access address
 - ◆ 上传需求文件 Upload requirement file
 - ◆ 上传设计文件 Upload design file
 - ◆ 上传测试文件 Upload test file
 - ◆ 上传资源实体 Upload resource entity
 - ◆ 上传资源使用帮助文件 Upload resource help file
 - ◆ 质量信息：Quality Information
 - 总体满意度：overall satisfaction
 - 总体评价：overall evaluation
 - 有效性：effectiveness
 - 满意度：satisfaction
 - 安全性：safety
 - 生产率：productivity
 - 功能性：Functionality
 - ◆ 保密安全性：Security
 - ◆ 适合性：Suitability
 - ◆ 准确性：Accuracy
 - ◆ 互操作性：Interoperability
 - 可靠性：Reliability
 - ◆ 易恢复性：Recoverability
 - ◆ 成熟性：Maturity
 - ◆ 容错性：Fault Tolerance
 - 效率：Efficiency
 - ◆ 时间特性：Time Behaviour
 - ◆ 资源利用率：Resource Behaviour
 - 易用性：Usability
 - ◆ 吸引力：acctractiveness

-
- ◆ 易学性: Learnability
 - ◆ 易理解性: Understandability
 - ◆ 易操作性: Operability
 - 维护性: Maintainability
 - ◆ 易改变性: Changeability
 - ◆ 易测试性: Testability
 - ◆ 稳定性: Stability
 - ◆ 易分析性: Analyzability
 - 可移植性: Portability
 - ◆ 适应性: Adaptability
 - ◆ 易替换性: Replaceability
 - ◆ 共存性: co-existence
 - ◆ 易安装性: Installability

■ 统计信息: Statistics

- ◆ 复用总次数 total times of reused
- ◆ 趋势图 Trend Chart
- ◆ 下载次数 downloads...
- ◆ 下载客户端 download client
- ◆ 下载排行榜 download ranking
- ◆ 下载时间 download time
- ◆ 下载资源 download resource
- ◆ 下载总次数 total downloads
- ◆ 新增服务 new service
- ◆ 新增工具 new tool
- ◆ 新增构架 new architecture
- ◆ 新增构件 new component
- ◆ 新增资源 new resource
- ◆ 资源包总数 Total Resource Bundle

Trustie技术规范

附录 3：软件资源可信证据示例

本示例给出了北京大学构件化应用服务器软件在软件资源库中根据可信证据模板所填写的相应证据项信息。其中，有关支撑附件仅在表中列出，具体文档不作附件。

1、开发阶段可信证据信息

分类	证据名称	结果	支撑附件
开发阶段	过程管理规范	PKUAS 开发小组过程管理规范	01PKUAS 研制说明书.doc
	需求变更数	2	02PKUAS-Container 需求变更记录文档.doc
	需求评审结论	对 J2EE 规范、EJB2.0 规范、EJB3.0 规范的钻研较深，基本全面掌握规范要求，对容器需求分析深刻，对四种 EJB 容器、容器管理机制、在线演化机制、容器定制、用户自定义连接子等需求功能界定清晰，对于容器功能的一般性要求和非功能属性的规定较明确。	03PKUAS-Container 需求评审报告.doc
	设计方法	详见设计文档说明	04EJB 容器和实例管理器设计文档.doc
	设计评审结论	新型构件化应用服务器 PKUAS 的容器设计清晰性、完整性、依从性、一致性、可行性、数据使用、功能性、接口、可维护性、可靠性、可追溯性都较良好，需求文档和设计文档齐全、编写规范、符合标准化要求。评审组同意通过设计评审。	05PKUAS-Container 设计评审报告.doc
	编码规范	良好的 Java 程序代码规范	06PKUAS 代码风格规范.doc
			06PKUAS 异常处理编码规范.doc
	测试缺陷趋势	{开始时间：2007-4-27 结束时间：2007-7-10 缺陷密度：0.0014 } {开始时间：2007-10-22 结束时间：2008-3-29 缺陷密度：0.007}	
缺陷清除率			

--	--	--	--

2、提交阶段可信证据信息

分类	证据名称	结果	支撑附件
测试结果	测试类型与结果	{测试类型: 功能测试 测试结果: 功能正确率 100%, 覆盖率 100%} {测试类型: 可靠性测试 测试结果: 动态集群提高了可靠性} {测试类型: 安全测试 测试结果: 安全模块提高了安全性}	07PKUAS 鉴定测试大纲.doc 07PKUAS 鉴定测试记录.doc
	残留缺陷数	0	07PKUAS 鉴定专家测试报告.doc
测试需求	测试准则	测试 PKUAS 应用服务器的 EJB2.0/3.0 规范功能兼容性, 根据 J2EE 5 规范要求, 必须通过 EJB2.0 及 EJB3.0 规范的 EJB 功能测试且功能覆盖率达 100%	07PKUAS 鉴定测试大纲.doc 07PKUAS 鉴定专家测试报告.doc
	测试需求	按软件工程标准的要求, 检查全部技术文档, 包括开发文档、用户手册和测试文档, 并对 PKUAS 功能性进行评估	07PKUAS 鉴定测试大纲.doc 07PKUAS 鉴定专家测试报告.doc
测试过程	测试方法	针对 PKUAS 的开发目标, 依据测试大纲, 围绕构件化应用服务器的功能、性能、可用性和健壮性等方面, 进行了现场测试, 并做详细记录。	07PKUAS 鉴定测试大纲.doc 07PKUAS 鉴定专家测试报告.doc
	测试人员	“新型构件化应用服务器”鉴定测试组	07PKUAS 鉴定专家测试报告.doc
	测试工具	Junit 测试框架	07PKUAS 鉴定测试大纲.doc
	测试环境	硬件环境: 两台 IBM HS20 8843 刀片服务器, 主要配置参数为: Intel Xeon 2.8 GHz, 1G DDR II 内存, 80G 硬盘(RAID, Ultra320 SCSI), 双 1000M 以太网控制器, 其中, 一台为数据库服务器, 另一台为	07PKUAS 鉴定测试大纲.doc

		每日构建服务器。 软件环境： Debian Linux 4.0 操作系统, Oracle 9i 和 Mysql 5.0 数据库 系统, JDK 环境为 Sun Hotspot 1.5.0_10 JVM, PKUAS2006。	
	测试过程记录	按模块分工至负责人, 分别详细 记录每个测试结果, 最后汇总至 PKUAS 鉴定测试记录文档。	07PKUAS 鉴定测试记 录.doc
分析证据	死锁检测		—
	内存泄漏检测		—
	代码风格缺陷检 测		—

3、应用阶段可信证据信息

分类	证据名称	结果	支撑附件
使用阶 段	使用规模	对于 J02nAS 合作版本, 自 2008 年 4 月以来, 下载次数为 2958 次	09PKUAS-Container 用户 J02nAS 使用报告.doc 09PKUAS 用户使用案例列 表.doc
	应用领域	PKUAS 已广泛应用于金融、交通、 政务、能源、电信等领域, 拥有 大中型客户逾百家, 产生了很好 的直接经济效益。	09PKUAS 用户使用案例列 表.doc
	用户满意度	很好	09PKUAS-Container 用户 J02nAS 使用报告.doc
	应用案例	(1) 整体转化为中和威 InterWeb、中创 InforWeb 等国 内知名产品 (2) 专利技术转让给东方通 TongWeb 等 (3) 联合研发了 863 中间件套 件“四方国件” (4) 与 AVAYA 和 Platform 等国 际知名公司的产品实现了集成 (5) PKUAS 的开源版本实现了与 国际著名开源中间件 JOnAS 的对	10PKUAS 详细应用情况介 绍.doc

		等合并，形成了新的中间件平台 J02nAS	
	第三方评测论证情况	2007 年通过鉴定，2008 年获得国家科技进步二等奖	2008 年国家技术发明二等奖 构件化应用服务器核心技术与应用. bmp
			11PKUAS 神州数码性能测试结果及证明. doc
			11PKUAS 总装备部武器装备论证研究中心测试结果及证明. doc

Trustie 技术规范