

# HP Application Lifecycle Intelligence

适用于 HP ALM Platform

软件版本： 2.6

---

## 用户指南

文档发行日期： 2012 年 8 月

软件发行日期： 2012 年 6 月



## 法律声明

### 担保

HP 产品和服务的唯一担保已在此类产品和服务随附的明示担保声明中提出。此处的任何内容均不构成额外担保。HP 不会为此处出现的技术或编辑错误或遗漏承担任何责任。

此处所含信息如有更改，恕不另行通知。

### 受限权利声明

机密计算机软件。必须拥有 HP 授予的有效许可证，方可拥有、使用或复制本软件。按照 FAR 12.211 和 12.212，并根据供应商的标准商业许可的规定，商业计算机软件、计算机软件文档与商品技术数据授权给美国政府使用。

### 版权声明

© 版权所有 2003 - 2012 Hewlett-Packard Development Company, L.P.

### 商标声明

Adobe™ 是 Adobe Systems Incorporated 的商标。

Microsoft® 和 Windows® 是 Microsoft Corporation 在美国的注册商标。

UNIX® 是 The Open Group 的注册商标。

## 文档更新

此文档的标题页包含以下标识信息：

- 软件版本号，用于指示软件版本。
- 文档发行日期，该日期将在每次更新文档时更改。
- 软件发布日期，用于指示该版本软件的发布日期。

要检查是否有最新的更新，或者验证是否正在使用最新版本的文档，请访问：

<http://h20230.www2.hp.com/selfsolve/manuals>

需要注册 HP Passport 才能登录此站点。要注册 HP Passport ID，请访问：

<http://h20229.www2.hp.com/passport-registration.html>

或单击“HP Passport”登录页面上的“**New users - please register**”链接。

此外，如果订阅了相应的产品支持服务，则还会收到更新的版本或新版本。有关详细信息，请与您的 HP 销售代表联系。

## 支持

请访问 HP 软件支持网站:

<http://www.hp.com/go/hpsoftwaresupport>

此网站提供了联系信息, 以及有关 HP Software 提供的产品、服务和支持的详细信息。

HP 软件联机支持提供客户自助解决功能。通过该联机支持, 可快速高效地访问用于管理业务的各种交互式技术支持工具。作为尊贵的支持客户, 您可以通过该支持网站获得下列支持:

- 搜索感兴趣的知识文档
- 提交并跟踪支持案例和改进请求
- 下载软件修补程序
- 管理支持合同
- 查找 HP 支持联系人
- 查看有关可用服务的信息
- 参与其他软件客户的讨论
- 研究和注册软件培训

大多数提供支持的区域都要求您注册为 HP Passport 用户再登录, 很多区域还要求用户提供支持合同。要注册 HP Passport ID, 请访问:

<http://h20229.www2.hp.com/passport-registration.html>

要查找有关访问级别的详细信息, 请访问:

[http://h20230.www2.hp.com/new\\_access\\_levels.jsp](http://h20230.www2.hp.com/new_access_levels.jsp)

## 目录

用户指南 .....	1
目录 .....	5
欢迎使用 Application Lifecycle Intelligence .....	7
本指南的组织方式 .....	9
设置 ALI .....	10
ALI 先决条件 .....	11
启用 ALI 扩展 .....	11
ALI 迁移 .....	11
SCM 系统集成 .....	12
支持的 SCM 系统 .....	13
SCM 代理程序 .....	13
配置存储库 .....	14
添加和编辑存储库 .....	15
设置外部存储库查看器 .....	16
设置分支和实施 .....	17
设置分支 .....	18
设置提交模式 .....	19
设置变更检测 .....	21
内部版本系统集成 .....	22
支持的内部版本系统 .....	23
Hudson/Jenkins 安装 .....	23
HP ALI Hudson/Jenkins 插件安装 .....	23
HP ALI Hudson/Jenkins 插件配置 .....	24
添加内部版本服务器 .....	25
添加内部版本配置 .....	25
从内部版本配置重用 SCM 配置 .....	26
设置内部版本配置缺陷筛选 .....	26
自定义 ALI 项目列表 .....	26
代码问题跟踪 .....	27
配置代码问题跟踪 .....	27

设置内部版本服务器检测 .....	28
Force.com 集成 .....	29
先决条件 .....	30
项目部署、测试和报告生成 .....	30
Hudson/Jenkins Force.com 配置 .....	31
监控 SCM 变更和可跟踪性 .....	33
查看代码变更表 .....	34
代码变更详细信息 .....	34
查看变更影响报告 .....	35
生成项目报告 .....	35
生成图 .....	35
监控开发活动 .....	37
开发活动 - 发布 .....	38
开发活动 - 缺陷 .....	38
开发活动 - 需求 .....	38
监控缺陷详细信息中的开发活动 .....	38
内部版本-测试可跟踪性 .....	40
查看内部版本-测试可跟踪性 .....	41
监控内部版本活动 .....	42
查看内部版本 .....	43
内部版本详细信息 .....	43
内部版本报告 .....	45
生成内部版本图 .....	45

## 欢迎使用 Application Lifecycle Intelligence

Application Lifecycle Intelligence (ALI) 是一组用于提供完整 ALM 可跟踪性的功能、报告和度量。ALI 使 ALM 干系人可以做出明智的决策。

ALI 利用源代码管理 (SCM) 和内部版本管理集成扩展了 HP ALM Platform 模型, 建立了 ALM 实体 (发布、需求、缺陷、测试集、测试运行…)、代码变更、内部版本、单元测试结果与代码覆盖率分析之间的链接。这将 HP ALM 转换到提供有关业务活动和开发活动的信息的中央记录系统。ALI 还提供了内部版本测试可跟踪性, 使您可以将内部版本与测试关联并跟踪这些关联。

### 控制提交到 SCM 的变更

使用 ALI, 可以定义用来控制已提交变更的 SCM 策略。实施 SCM 策略有助于确保开发者遵循规定的准则和最佳实践。团队领导人可以确定开发者执行了正确的功能, 并将必需的元数据添加到提交的变更集。策略实施还在稳定期间提供了宝贵的帮助, 因为可以轻松确保开发者仅修复严重的缺陷或完全锁定发布的代码库。

有关详细信息, 请参见 [“设置分支和实施” \(第 17 页\)](#)。

### 开发者、QA 管理员和项目经理可以轻松查看:

- 执行需求或缺陷的代码变更。
- 特定内部版本中执行的代码变更。
- 特定时间段、发布、内部版本中具体执行的内容或单个贡献者具体执行的内容。
- 受影响的需求和缺陷。
- 与特定需求关联的变更量。
- 变更集中的逐行差异。
- 与需求和缺陷都不关联的变更。
- 内部版本和质量度量。
- 有关代码覆盖率、测试结果和变更量的汇总信息。
- 内部版本测试可跟踪性。

有关详细信息, 请参见以下部分:

- [“监控 SCM 变更和可跟踪性” \(第 33 页\)](#)
- [“监控开发活动” \(第 37 页\)](#)
- [“内部版本-测试可跟踪性” \(第 40 页\)](#)
- [“内部版本详细信息” \(第 43 页\)](#)

Application Lifecycle Intelligence 通过将 ALM 需求、测试和缺陷与常见的开放式源代码和商业 SCM、内部版本系统、单元测试及代码覆盖率分析工具相集成, 支持此 ALM 可跟踪性。

### ALI 2.6 提供与以下对象的集成:

- **SCM 系统**

- Subversion (SVN)
- Concurrent Versions System (CVS)
- Microsoft Team Foundation Server (TFS)
- Perforce
- Git

集成是独立于 SCM 客户端的；开发者可以从其当前 SCM 客户端提交变更，可以是命令行实用工具也可以是类似使用 Tasktop 插件的 Eclipse 的 IDE。有关详细信息，请参见 [“支持的 SCM 系统”](#)（第 13 页）。

- **内部版本系统**

- Jenkins
- Hudson

ALI 为内部版本服务器提供自动提取内部版本信息和度量的插件，并将它们嵌入到 ALM 模型中。有关详细信息，请参见 [“内部版本系统集成”](#)（第 22 页）。

- **单元测试**

- JUnit
- TestNG
- NUnit

- **代码覆盖率分析**

- Cobertura
- NCover

---

**注：**本指南介绍如何将 Application Lifecycle Intelligence 与 ALM 结合使用。有关使用 ALM 的详细信息，请参考《HP Application Lifecycle Management 用户指南》。

---



## 本指南的组织方式

本指南包含以下章节：

章节	描述
第 1 章 <a href="#">“设置 ALI”</a> (第 10 页)	使用 Application Lifecycle Intelligence 的先决条件和配置信息。
第 2 章 <a href="#">“SCM 系统集成”</a> (第 12 页)	将 Application Lifecycle Intelligence 与 SCM 系统集成。
第 3 章 <a href="#">“内部版本系统集成”</a> (第 22 页)	将 Application Lifecycle Intelligence 与内部版本系统集成。
第 4 章 <a href="#">“Force.com 集成”</a> (第 29 页)	将 Application Lifecycle Intelligence 与 Force.com 集成。
第 5 章 <a href="#">“监控 SCM 变更和可跟踪性”</a> (第 33 页)	如何查看变更集数据。
第 6 章 <a href="#">“监控开发活动”</a> (第 37 页)	如何查看开发活动。
第 7 章 <a href="#">“内部版本-测试可跟踪性”</a> (第 40 页)	如何将内部版本与测试关联并跟踪这些关联。
第 8 章 <a href="#">“监控内部版本活动”</a> (第 42 页)	如何查看内部版本数据。

# 第 1 章

---

## 设置 ALI

此章节描述使用 Application Lifecycle Intelligence (ALI) 的安装要求和设置过程。

如果从更早版本的 ALI 升级，请参见 [“ALI 迁移”](#) (第 11 页)。

此章节包括以下内容：

ALI 先决条件 .....	11
启用 ALI 扩展 .....	11
ALI 迁移 .....	11

## ALI 先决条件

ALI 需要与之集成的 SCM 和内部版本系统。有关详细信息，请参见[“支持的 SCM 系统”](#)（第 13 页）和[“支持的内部版本系统”](#)（第 23 页）。

## 启用 ALI 扩展

在 ALM 站点管理中，必须为需要 ALI 的每个 ALM 项目启用 ALI 扩展。可以在现有项目上启用扩展，也可以在创建新项目时启用。

启用 ALI 扩展之前，请考虑以下方面：

- 启用扩展后，无法禁用项目扩展。
- 如果项目处于活动状态，则必须先停用项目再启用扩展。启用扩展后，可以重新激活项目。
- 为项目启用扩展可能需要一些时间。

有关在 ALM 项目中启用扩展的详细信息，请参考《HP Application Lifecycle Management 管理员指南》。

## ALI 迁移

### 迁移到 ALI 2.6

如果从以前版本升级：

- 按照[迁移到 ALI 2.0](#) 下列出的步骤操作。
- 在“代码变更”模块中，所有用户都必须添加**消息**列。迁移后，此列不再显示。

### 迁移到 ALI 2.0

**Perforce 迁移：** 如果从 Application Lifecycle Intelligence 版本 1.1 升级到版本 2.0，则迁移由 ALM 处理的 Perforce 数据需要从 Perforce 中删除当前数据后再重新加载这些数据。

**要从 Perforce SCM 迁移所有数据，请对每个 Perforce 存储库执行以下步骤：**

1. 在“管理”模块中，选择 **SCM 存储库** 并选择 Perforce 存储库。
2. 从“存储库”菜单选择**清理**，然后选择最近提交的删除日期。
3. 转到**分支**选项卡，对于每个分支：
  - a. 选择分支详细信息。
  - b. 删除**上次变更读取值**。
4. 从**变更检测**选项卡启动同步。

## 第 2 章

---

### SCM 系统集成

此章节包括以下内容:

支持的 SCM 系统 .....	13
配置存储库 .....	14
设置分支和实施 .....	17
设置提交模式 .....	19
设置变更检测 .....	21

## 支持的 SCM 系统

Application Lifecycle Intelligence (ALI) 本机支持 Subversion、Perforce、Git 和 CVS 系统。除非需要使用 Push 机制的提交消息策略实施, 否则无需额外安装就可从这些系统加载变更集。如果 Push 机制是必需的, 则需要在 SCM 系统上安装 SCM 代理程序。有关详细信息, 请参见[“SCM 代理程序”\(第 13 页\)](#)。

本机还支持 TFS 系统, 但要获取完整 TFS 系统支持, 则必须安装 TFS 代理程序 Web Service。其中某个 Web Service 设计用于计算更改的行数。如果未安装该服务, 则“更改的行数”字段将始终显示“0”。

**备注:** ALI 2.6 支持通过 SvnBridge 的以前版本的 TFS 系统方法, 但此方法在当前版本的 ALI 中已弃用。请改为使用 TFS 系统的本机支持。

### 支持的 SCM 系统

ALI 提供对以下 SCM 系统的支持:

- Subversion 版本: 1.6.\*、1.7\*(在 1.6.11、1.6.16、1.7.0、1.7.3 上测试)
- CVS 版本: 1.11.\*、1.12.\*(在 1.11.22、1.11.23、1.12.13 上测试)
- TFS 版本: Team Foundation Server 2010(在 TFS 2010 上测试, 仅限 Windows)
- Perforce 版本: 2010.2(在 2010.2/334844 上测试)
- Git 版本: 1.7.9\*、1.7.10\*(在 1.7.8.5、1.7.9.4、1.7.10.0 上测试, 仅限 Linux)

SCM 代理程序支持部署到以下操作系统:

- Red Hat Enterprise Linux 6.x(32 位、64 位)
- SuSE Linux Enterprise 11.x(32 位、64 位)
- Windows 2008 Server(32 位、64 位)
- Windows 2008 R2 Server(64 位)

#### 先决条件:

MS PowerShell 2.0 或更高版本必须安装在 Windows 上并且可以运行脚本。Linux/Unix 代理程序脚本使用 BASH。BASH 版本必须是 4.0 或更高版本才能支持推送到多个存储库。

#### 限制:

- **Git:** ALI 不支持 Git 更改提交历史记录的功能。如果在 Git 存储库中更改了提交历史记录, 则必须为所有受影响的分支重新加载 ALI 跟踪的整个提交历史记录。要重新加载提交历史记录, 请删除这些分支再重新添加。

## SCM 代理程序

代理程序是与 Push 机制必需的相应 SCM 系统相关的应用程序的名称。代理程序是安装在 SCM 服务器上配置用于侦听 SCM 系统的一组脚本或专有应用程序。当变更提交到配置的存储库和分支时, 代理程序会检查策略, 如果允许提交, 则将变更集推送到 ALM 服务器。

有关配置 SCM 代理程序的详细信息, 请参见 `readme.txt` 文件, 该文件位于主 ALI 分布存档内的特定代理程序存档中。

有关使用 SCM 代理程序的更多详细信息, 请参见[“设置变更检测”\(第 21 页\)](#)。

**SVN Agent for Linux**

agents\scm-integration\unix-linux\scm-agent-subversion.tgz

**SVN Agent for Windows**

agents\scm-integration\windows\scm-agent-subversion.zip

**CVS\_Linux**

agents\scm-integration\unix-linux\scm-agent-cvs.tgz

**CVS\_Windows**

agents\scm-integration\windows\scm-agent-cvs.zip

**TFS 代理程序安装过程**

agents\scm-integration\windows\scm-agent-tfs.zip

**Perforce Agent Linux**

agents\scm-integration\unix-linux\scm-agent-perforce.tgz

**Perforce Agent Windows**

agents\scm-integration\windows\scm-agent-perforce.zip

**Git Agent for Linux**

agents\scm-integration\unix-linux\scm-agent-git.tgz

**Git Agent for Windows**

agents\scm-integration\windows\scm-agent-git.zip

## 配置存储库

要启用 Application Lifecycle Intelligence (ALI) 以从 SCM 系统加载代码变更并启动工作项(需求/缺陷)、代码变更和已定义发布的自动可跟踪性, 必须配置 SCM 存储库。

此部分包括以下内容:

添加和编辑存储库 .....	15
设置外部存储库查看器 .....	16

## 添加和编辑存储库

ALI 支持 Subversion、CVS、TFS、Perforce 和 Git。有关详细信息，请参见[“支持的 SCM 系统”](#)（第 13 页）。

### 要添加或编辑存储库：

1. 使用管理员特权登录到 ALM。在 ALM 侧栏上的“管理”下方，选择 **SCM 存储库** 以打开“存储库”页。
2. 要添加新存储库，请单击**新建存储库**。从下拉列表选择 SCM 类型并单击**确定**。
3. 在新窗口中，根据需要进行输入存储库的名称、位置、用户名和密码。

**注：**位置是指存储库的完整路径，比如 `http://host/svn/repo` 等。如果不确定此值，请咨询系统管理员。

**Perforce:** ALI 存储库相当于 Depot。存储库的位置应包括 Perforce 服务器的主机名和端口以及 Depot 名称。格式应类似于主机:端口//depot 名称。

**Subversion:** 存储库位置必须指向实际的存储库根。分支稍后用于指定存储库内的路径。如果 SVN URL 有未知根，则使用“`svn info <URL>`”命令查找根。

**TFS:** ALI 存储库相当于 TFS Project Collection。存储库的位置应包括 TFS 服务器的主机名和端口以及 Project Collection 的路径。格式应为 `http://主机:端口//project collection` 路径。

**CVS:** 在别名属性中输入完整路径：`CVSROOT`（例如：`pserver:用户名:密码@主机/cvsrepo`），与内部版本系统中配置的完全一致。`CVSRoot` 属性是 `pserver` 协议所必需的。

**Git:** 存储库位置应是整个 Git URL。支持的协议包括 `http(s)`、`Git` 和 `ssh`。例如：`https://github.com/hp/ali.git`。对于使用私钥的 `ssh` 身份验证，请使用安全密钥区域。

如果配置了 GitHub 存储库，则应选择“GitHub view diff/file link templates”属性以使用本机 GitHub 文件查看和差异查看 Web 工具。

4. 如果需要修改任何存储库属性，请选择属性并单击**编辑属性**按钮。在**编辑属性**窗口中，添加值并单击**确定**。

**CVS:** CVS 存储库需要 CVS 协议 `pserver` 和**初始日期**才能开始加载变更集。还应指定 CVS 根，但这不是必需的。

**TFS:** 如果计数行服务 URL 不是默认值（存储库协议://存储库主机:9080/AliTfsServices/Service/CountLinesService.svc），则创建分支时将出现警告，且所有加载的变更集都以 0 条更改的行设置存储。

**Git:** 必须设置必需的属性“`directory for clone`”。这是应创建远程存储库镜像的绝对路径。该路径应位于 ALM 存储库文件夹下。每个位置都需要单独的文件夹。如果更改存储库位置，则应更改克隆目录属性或删除文件夹的内容。

5. 单击**提交**。ALI 先测试存储库的连接再将其添加到列表。

可以在添加存储库时设置变更检测和提交模式选项。

有关详细信息，请参见[“设置变更检测”](#)（第 21 页）和[“设置提交模式”](#)（第 19 页）。

## 设置外部存储库查看器

ALI 提供内置存储库查看器，这样无需额外配置便可查看文件的差异和详细信息。还可以使用首选存储库浏览器，比如 ViewVC。

以下属性在配置 SCM 存储库时输入，可从现有存储库的详细信息选项卡进行更改。

**Template for diff links** —— 在存储库查看系统（比如 ViewVC）中指向给定文件的差异视图的链接的模板。允许您在 UI 中创建源自 ALI 变更集表的链接。变更集中的每个文件都包含显示差异视图（当前变更集的修订与先前修订的差异）的链接。模板可以包含在运行时可根据 UI 中当前所选变更集的上下文而扩展的标记/变量。

可替换/扩展的标记/变量：

`${filePath}` ... 存储库中文件的路径。

`${revision}` ... 当前变更集中的修订。

`${fromRevision}` ... 与当前变更集中的修订相对的先前修订。

`${fromFilePath}` ... 如果是复制/移动，则为其源位置。

**备注：(TFS)** 要查看 TFS 中存储的两个文件的差异，预期地址为：

```
http://TFSServer:8080/tfs/_COLLECTION_/web/diff.aspx?opath={originalItemPath}
&ocs={originalItemChangeset}&mpath=
{modifiedItemPath}&mcs={modifiedItemChangeset}
```

其中：

- TFSServer: TFS 服务器的地址。
- \_COLLECTION\_: 请求的文件项目是其成员的集合的名称。
- opath: 原始源文件的完整路径（包括“\$”符号和项目名称），比如 `$/ali-replica/alik/pom.xml`
- ocs: TFS 原始变更集 ID
- mpath: 已修改的源文件的完整路径（包括“\$”符号和项目名称），比如 `$/ali-replica/alik/pom.xml`
- mcs: TFS 已修改的变更集 ID

**备注：(Git)** Git 差异链接模板包括以下变量：

`${blobId}` —— Blob SHA

`${commitId}` —— 提交 SHA

`${prevBlobId}` —— 文件的以前版本的 Blob SHA

`${prevCommitId}` —— 父提交的提交 SHA

`${branchName}` —— 分支的名称

`${filePath}` —— Git 存储库中的文件路径（不以“/”开头）



**Template for file links** —— 在存储库查看系统(比如 ViewVC) 中指向给定文件的文件视图的链接的模板。允许您在 UI 中创建源自 ALI 变更集表的链接。变更集中的每个文件都包含显示文件视图(当前变更集范围内给定修订的文件文本内容) 的链接。模板可以包含在运行时可根据 UI 中当前所选变更集的上下文而扩展的标记/变量。

可替换/扩展的标记/变量:

`${filePath}` ... 存储库中文件的路径。

`${revision}` ... 当前变更集中的修订。

**备注: (TFS)** 要查看 TFS 中存储的文件的源, 预期地址为:

```
http://TFSServer:8080/tfs/_COLLECTION_/web/view.aspx?&path={itemPath}&cs={itemChangeset}
```

其中:

- TFSserver: TFS 服务器的地址。
- \_COLLECTION\_: 请求的文件项目是其成员的集合的名称。
- path: 源文件的完整路径(包括“\$”符号和项目名称), 比如 `$/alireplica/alik/pom.xml`
- cs: TFS 变更集 ID

**备注: (Git)** Git 文件链接模板包括以下变量:

`${blobId}` —— Blob SHA

`${commitId}` —— 提交 SHA

`${branchName}` —— 分支的名称

`${filePath}` —— Git 存储库中的文件路径(不以“/”开头)

## 设置分支和实施

分支是从任何现有存储库的**分支**选项卡设置的。分支是存储库中独立于其他行存在的单独行。

分支需要与发布关联, 并且可以将变更检测选项设置为将变更以变更集的形式显示在代码变更表中。

还可以设置提交模式, 以便允许或禁止用户提交分支变更, 但满足指定条件时除外。

当为 SCM 存储库配置了代理程序时, 可随时进行策略实施。

有关代理程序的详细信息, 请参见 [“SCM 代理程序”\(第 13 页\)](#)。

此部分包括以下内容:

**设置分支** .....18

## 设置分支

### 要添加新分支：

1. 转到**分支**选项卡并单击**添加**。
  - a. 提供分支的路径名称，还可选择提供分支和上次变更读取属性值。

**CVS:** 分支名称仅对某些 CVS 存储库有意义。不要为 SVN 或通过 SvnBridge 的 TFS 填充此属性。对于 SVN，“上次变更读取”字段是上次修订；对于 CVS，则是读取变更集的日期时间。如果必须从给定分支读取所有变更集，则将这些字段留空。

**Perforce:** 如果使用 Perforce，则设置不包含 Depot 名称的分支路径。例如，如果分支位于 `//depot/HelloWorld/releases/release-1.0/...` 中，则路径应为 `/HelloWorld/releases/release-1.0`。即使已经指定分支，也不要设置参数 `branch`。

**TFS:**

使用不包含“\$”的项目路径作为分支路径。例如，如果项目位于“\$/TestApp”中，则分支路径应为“/TestApp”。

分支路径必须仅包含项目名称。不支持包含子文件夹的路径。

**Git:** 如果使用 Git，则始终将分支路径设为“/”。字段名应设置为实际 Git 分支名称。仅使用“master”格式的简单分支名称，而不是“refs/head/master”。
2. 单击**提交**以测试链接并添加分支。

### 要将分支与发布关联：

1. 单击分支路径或**详细信息**按钮，打开“SCM 分支详细信息”窗口。
2. 单击**添加**按钮并选中下拉菜单中必需发布的复选按钮。
3. 默认情况下，开始日期和结束日期取自发布，可通过单击日期字段并输入必需日期来更改这些设置。

给定时间段内分支的变更集与指定发布关联。当选择指定发布时，这些变更集显示在“代码变更”模块中。可以为分支指定更多分布，但通常应在不同的时间段内。

### 要为想要实施提交的分支指定签入策略：

**备注：**此功能不适用于 Git 存储库。设置将被忽略。

1. 单击 **SCM 分支详细信息** 窗口左侧窗格中的**实施**。
2. 选择以下选项：
  - **签入策略**选项卡上的**提交消息必须与已定义模式匹配**，导致提交消息与预定义模式不匹配的提交被代理程序拒绝。
  - **变更集引用需求**，强制每次提交都必须引用现有需求。
    - **具有需求类型或优先级为**，从下拉菜单中选择类型和/或优先级，以便提交仅允许与特定需求类型或优先级关联的变更。
  - **变更集引用缺陷**，强制每次提交都必须引用现有缺陷。

- **严重度为**，选择所需的严重级别，以仅允许与特定严重度关联的变更。
  - **将此注释添加到系统消息中**，发送包含所应用注释的自定义系统消息，通知用户提交被阻止。
3. 单击**确定**。

**注：** 签入策略功能需要安装代理程序才能运行。有关详细信息，请参见[“SCM 代理程序”](#)（第 13 页）。

**要为想要实施提交的分支指定锁定策略：**

**备注：** 此功能不适用于 Git 存储库。设置将被忽略。

1. 在**锁定策略**选项卡上，可以管理与分支关联的锁定策略。
2. 选择以下选项：
  - **不允许提交，以下情况除外**，禁止除以下例外的所有到给定分支的提交：
    - 即使阻止了分支，也可以通过应用允许提交的 SCM 用户的用户名来指定允许提交到分支的用户列表。
    - 通过单击**添加缺陷**按钮并提供缺陷的 ID 来创建允许提交变更的缺陷列表。通过单击**删除缺陷**从表中删除缺陷。
  - **将此注释添加到系统消息中**，发送包含所应用注释的自定义系统消息，通知用户提交被阻止。
3. 单击**确定**。

## 设置提交模式

检测和维护变更集与需求/缺陷之间的可跟踪性是基于用户（开发者）在提交 SCM 系统变更时提供的提交消息进行的。ALI 提供适应用户提供的各种允许模式的可自定义机制。其目的是传输在提交变更集时所提供的提交消息内的需求或缺陷 ID。

可通过在**提交模式**视图中设置新存储库或单击现有存储库的**提交模式**选项卡来访问提交模式设置。

可以基本或高级模式定义模式。

**基本：**

1. 通过单击**添加**按钮，然后输入值并单击**确定**来为缺陷或需求添加关键字。提供的关键字将提交标记为与缺陷或需求相关。

可通过选择关键字并单击**删除**按钮来删除该关键字。

2. 通过输入 ID 号将前缀 ID 添加到模式。单击单选按钮使其变为可选。
3. 单击**更多选项**按钮可添加以下选项：
  - **包括默认 Tasktop 提交模式** —— 默认情况下打开。当 ALM 中集成了 Taskop 插件时，Tasktop 将根据 ALI 可识别的模式生成默认提交消息。
  - **区分大小写的提交消息** —— 默认情况下关闭。想要对提交消息实施大小写区分时打开。
  - **多个缺陷或需求分隔标记** —— 当提交消息包括多个缺陷或需求时，此标记被识别为分隔符。

- **提交消息中的关键字位置** —— 仅提交消息的开头或消息内任何位置的关键字将被识别。
- **用户提交消息分隔符** —— 后跟缺陷或需求关键字的多条用户提交消息之间的分隔符。
  - **提交消息中的选项示例:**

fixing defect #100, #101:fix caching and enhance functionality

*fixing defect* = 缺陷关键字

*#100* = 缺陷 ID 前缀和缺陷 ID

“,” = 多个缺陷分隔符

“:” = 用户提交消息分隔符

*fix caching and enhance functionality* = 用户提交消息

4. 退出选项卡将提交配置。

**高级:**

1. 添加或更改默认代码模式。
2. 通过单击**对现有提交进行测试**来测试已修改的代码。
  - a. **区分大小写**对消息实施大小写区分。
3. 可选择通过在字段中输入自定义提交消息来测试该消息。
4. 退出选项卡将提交配置。

**恢复默认值**按钮删除任何变更并替换默认关键字。

**下面是高级配置的几个基本示例:**

1. 示例 1

- a. 模式

```
[fixing] REGEX('defects?') IDLIST(DEFECT) | [implementing] REGEX('requirements?') IDLIST(REQ) ):TEXT
```

- b. 示例消息:

```
"fixing defect #56721:something really serious was fixed"
```

```
"defects #57893,#61432:division by zero"
```

```
"requirement #1:domains"
```

2. 示例 2

- a. 模式

```
(UNTIL(RE '(BUG)|(REQ))#') (IDLIST(DEFECT lead='(BUG)?#?' sep=',') | IDLIST(REQ lead='(REQ)?#?' sep=',') ) {0,} [TEXT]
```

- b. 示例消息:

```
"This commit fixes BUG#1,#2 and implements REQ#4,REQ#5 making the product faster (resolving BUG#7)."
```

- c. 此模式与任何找到的“BUG#”和“REQ#”模式的所有输入和提取匹配。此类开放模式可能不适合实施常用策略，但是当旧存储库中的数据以“只读”模式加载（比如，用于报告）时，此模式将非常有用。

### 3. 示例 3

#### a. Tasktop 模式

```
(LISTITEM(' Bug Status') - WORD IDLIST(DEFECT lead='DEF' sep=' ') |  
Incomplete - WORD IDLIST(REQ lead='REQ' sep=' ')) :TEXT
```

#### b. 匹配的默认 Tasktop 消息如下：

```
"OPEN - task DEF10:http://host:9090/qcbin;DEFAULT;ALI_DEV-  
DEF10"
```

```
"Incomplete - task REQ42:http://host:9090/qcbin;DEFAULT;ALI_  
DEV-REQ42"
```

## 设置变更检测

ALI 检测预配置的 SCM 存储库上的变更、将有关变更集和关联文件的信息加载到 ALM 服务器并自动创建已加载代码变更与工作项（需求、缺陷）之间的可跟踪性。ALI 支持两种检测机制：轮询机制和 Push 机制。

打开轮询机制时，ALI 定期检查给定 SCM 存储库中是否有新变更，如果有，则加载它们。对于轮询机制，不需要在 SCM 系统端进行任何配置，因此该方法可以在不运行 SCM 代理程序的情况下运行。轮询机制与“变更检测”选项卡上的选项从 *SCM 读取变更* 对应。

Push 机制将 SCM 代理程序连接到 SCM 系统。代理程序侦听 SCM 系统，并在变更提交到预配置的存储库和分支时，检查策略，如果允许提交，则将变更集推送到 ALM 服务器。Push 机制与“变更检测”选项卡上的选项接收 *SCM 代理程序传输的变更* 对应。

可通过在**变更检测**视图中设置新存储库或单击现有存储库的**变更检测**选项卡来访问变更检测设置。

有两个检测选项可用：

### 从 SCM 读取变更

- 此选项允许设置以秒为单位的读取间隔，以及通过单击**重新计划**按钮并选择时间和日期来设置运行计划。
- 可通过单击**同步**按钮实施即时读取。

**注：**可使用“工具”菜单下的 ALM **任务管理器** 工具监控同步过程。

### 接收 SCM 代理程序传输的变更

- 如果已经设置 SCM 代理程序，则此选项将立即处理变更。有关详细信息，请参见 [“支持的 SCM 系统”（第 13 页）](#)。

## 第 3 章

---

### 内部版本系统集成

内部版本是软件开发的关键可交付产品。Application Lifecycle Intelligence 跟踪内部版本的信息及其与其他 ALM 实体的关系，以便提供可跟踪性。

模型提供以下方面的信息：

- 内部版本产物（内部版本流程产生的二进制文件）。
- 内部版本内容（组件、包、文件...）。
- 内部版本中包含的新变更（自上个内部版本以来的新增内容——组成内部版本的变更集）。
- 执行的工作项（已在内部版本中执行/修改的需求、缺陷和故事）。
- 内部版本与发布之间的关系。
- 内部版本与测试之间的关系。

与内部版本系统集成是允许您测量代码变更对软件可交付产品和内部版本产物的影响的工具。内部版本允许您报告已执行的新代码以及以关键度量增量表示的对已交付项目的影响（例如，某个提交导致测试结果下降 5%）。

此章节包括以下内容：

支持的内部版本系统 .....	23
添加内部版本服务器 .....	25
添加内部版本配置 .....	25
从内部版本配置重用 SCM 配置 .....	26
设置内部版本配置缺陷筛选 .....	26
自定义 ALI 项目列表 .....	26
代码问题跟踪 .....	27
配置代码问题跟踪 .....	27
设置内部版本服务器检测 .....	28

## 支持的内部版本系统

ALI 提供对以下内部版本系统的支持：

- Jenkins —— 长期支持发布 —— 1.424.6、1.447.1
- Hudson —— 最新产品版本 —— 2.1.2、2.2.0

ALI 为内部版本服务器提供自动提取内部版本信息和度量的插件，并将它们嵌入到 ALM 模型中。

ALI 支持以下开发度量工具：

- 单元测试
  - JUnit —— 在与支持的内部版本系统捆绑的版本上测试。
  - TestNG —— 在 Hudson 0.8 和 Jenkins 0.32 上测试
  - NUnit —— 在 Hudson 0.10、Jenkins 0.14 以及 NUnit 框架 2.5.10 版本上测试
- 代码覆盖率分析
  - Cobertura —— 在 Hudson 1.1 和 Jenkins 1.3 上测试
  - Ncover —— 在 Hudson 0.3、Jenkins 0.3 以及 Ncover 3.4.18.6937 x86 (试用版) 版本上测试

## Hudson/Jenkins 安装

如果在使用 Hudson/Jenkins 内部版本服务器，则必须安装以下额外组件：

- **对于 Perforce、TFS 或 Git：**支持使用 SCM 的 Hudson/Jenkins 插件。可以从 <http://hudson-ci.org> 或 <http://jenkins-ci.org> 下载。这些仅是 Perforce、TFS 和 Git 所必需的，可在公共 Hudson/Jenkins 插件存储库中找到。默认情况下支持 SVN 和 CVS。
- HP ALI Hudson/Jenkins 插件(位于 ALI 捆绑包中的 `ali-hudson-plugin.hpi` —— `agents\build-integration\hudson\`)。此插件支持 Hudson 和 Jenkins 系统。
- **对于 Perforce、TFS 或 Git：**位于 ALI 捆绑包中的特定 SCM 的额外 HP ALI Hudson/Jenkins 插件。

有关如何安装适用于 Hudson/Jenkins 的 ALI 插件的详细信息，请参见下面的“[HP ALI Hudson/Jenkins 插件安装](#)” (第 23 页)。

## HP ALI Hudson/Jenkins 插件安装

要安装 HP ALI Hudson/Jenkins 插件：

1. 在 Hudson/Jenkins 中，打开插件管理器，并单击**高级**选项卡。
2. 在**上传插件**部分中，浏览到位于 ALI 捆绑包中的 `ali-hudson-plugin.hpi`，然后单击**上传**。
  - **TFS：**如果由 Hudson/Jenkins 构建的项目源代码存储在 TFS 中，则除了基本插件 `ali-hudson-plugin.hpi` 外，还必须安装 `ali-hudson-tfs-plugin.hpi`。

**警告：** `ali-hudson-tfs-plugin.hpi` 要求预安装从 Hudson/Jenkins 下载的

Hudson/Jenkins TFS 插件。有关更多详细信息，请参见 [“Hudson/Jenkins 安装” \(第 23 页\)](#)。

- **Perforce:** 如果由 Hudson/Jenkins 构建的项目源代码存储在 Perforce 中，则除了基本插件 ali-hudson-plugin.hpi 外，还必须安装 ali-hudson-perforce-plugin.hpi。

**警告:** ali-hudson-perforce-plugin.hpi 要求预安装从 Hudson/Jenkins 下载的 Hudson/Jenkins Perforce 插件。有关更多详细信息，请参见 [“Hudson/Jenkins 安装” \(第 23 页\)](#)。

- **Git:** 如果由 Hudson/Jenkins 构建的项目源代码存储在 Git 中，则除了基本插件 ali-hudson-plugin.hpi 外，还必须安装 ali-hudson-git-plugin.hpi。

**警告:** ali-hudson-git-plugin.hpi 要求预安装从 Hudson/Jenkins 下载的 Hudson/Jenkins Git 插件。有关更多详细信息，请参见 [“Hudson/Jenkins 安装” \(第 23 页\)](#)。

3. 上载插件后，重新启动 Hudson/Jenkins 服务器以实施更改。

#### 备注:

- 如果使用 CVS 或 TFS SCM，请参见 [“添加和编辑存储库” \(第 15 页\)](#) 了解 ALM 中 SCM 的更多配置。
- 安装的插件应列在**插件管理器**的 **已安装** 选项卡中。
- ALI 插件安装正确时，**Ali Integration (ALI 集成)** 链接应显示在 Hudson/Jenkins 左侧菜单中。单击此链接将显示插件提供的功能。

有关如何安装和使用 Hudson/Jenkins 插件的更多详细信息，请参见 Hudson/Jenkins 系统文档。

## HP ALI Hudson/Jenkins 插件配置

ALI Hudson 插件具有可从全局配置系统访问的全局配置和可从特定作业访问的作业范围配置。有关详细描述，请转到 Hudson/Jenkins 服务器上的 ALI 集成插件，此处描述了每个属性。

### 要设置全局配置:

从**系统管理 - 系统设置**中选择 **ALI Integration (ALI 集成)** 部分。可以配置以下选项:

- *Include the credentials in the SCM configuration (在 SCM 配置中包括凭据)* —— 指定是否应在 SCM 存储库描述符中包括用户名和密码。如果启用此安全模式，则用户必须对要列出的凭据具有“扩展的读取”权限。

**警告:** 启用此选项时务必非常小心，因为与内部版本配置关联的 SCM 存储库的凭据以纯文本形式显示在 REST 端点上。

- *Update build information in HP ALM (在 HP ALM 中更新内部版本信息)* —— 选中时，启动内部版本时会立即将有关所执行内部版本的信息发送到 HP ALM 服务器，该内部版本结束



时再次发送这些信息。

**注：**在全局配置中指定的所有属性都可以稍后被特定作业的属性替代。要为给定内部版本服务器配置 Push 机制时必须配置这些属性，但如果要使用 Push 机制并且还要为各个作业指定 ALM 属性，则必须启用此选项。

#### 要为特定作业设置配置：

从给定作业左侧菜单上的“配置”链接，选择 **ALI Integration (ALI 集成)** 选项。如果要为作业启用 ALI 集成，则必须打开此选项。

- *Test sources mapping pattern (测试源映射模式)* —— 使您可以根据实际测试结果确定测试源位置。有关详细信息（包括示例），请参见 ALI 集成插件帮助。
- *HP ALM configuration (HP ALM 配置)* —— 覆盖全局 ALI 配置属性 —— 在 HP ALM 中更新给定内部版本作业的内部版本信息。
- *NCover* —— 与 .NET 配置的 NCover 代码覆盖率相关。*NCover report XMLs (NCover 报告 XML)* 指定生成的原始 XML 报告文件，比如 `myproject/target/coverage-reports/*.xml`。文件集的基本目录是根工作区。
- *Force.com* —— 与 Force.com 集成相关。有关详细信息，请参见 [“Force.com 集成”](#)（第 29 页）。

## 添加内部版本服务器

#### 要添加内部版本服务器：

1. 使用管理员特权登录到 ALM。在 ALM 侧栏上的“管理”下方，选择**内部版本服务器**以打开“服务器”页。
2. 单击**新建服务器**，然后从下拉列表中选择“内部版本服务器配置”并单击**确定**
3. 在“新建内部版本服务器”对话框中，根据需要进行服务器的名称、位置、服务器描述、用户名和密码。

**注：**位置是指服务器的完整路径，比如 `http://xx.xx.xxx.xxx:xxxx`。如果不确定此值，请咨询系统管理员。

4. 选择**变更检测**选项卡并执行以下某项操作：
  - 根据需要选择**从内部版本服务器读取变更**并提供间隔或计划运行时间。
  - 根据需要选择**接收内部版本服务器代理程序传输的内部版本**。有关详细信息，请参见“内部版本服务器检测”。
5. 单击**提交**以添加服务器。

## 添加内部版本配置

#### 要添加内部版本配置：

1. 转到“内部版本配置”选项卡并单击**新建内部版本配置**。
2. 从显示的列表中选择“内部版本配置”，然后单击**确定**以打开详细信息窗口。

**注：**此列表包括受管理服务器的未使用配置。

**警告：**此列表仅包含启用了 ALI 集成的 Hudson/Jenkins 作业的配置。

3. 单击**发布**下拉菜单并选择关联的发布。
4. 单击**内部版本配置**下拉菜单并选择内部版本类型。此值允许您在“开发”选项卡下的“内部版本”模块中筛选内部版本。
5. 执行以下某项操作：
  - 启用配置 —— 如果启用了内部版本配置，将从内部版本系统加载新内部版本。
  - 将配置设为默认值 —— 默认内部版本配置用于计算为关联的发布显示的统计信息。

## 从内部版本配置重用 SCM 配置

现有内部版本配置均由 ALM 存储，可将 SCM 分支和存储库应用于新创建的配置。

对于内部版本系统上的特定内部版本配置，SCM 相关信息通常已指定（SCM 存储库、分支、凭据）。ALI 可从内部版本系统重用特定内部版本配置的这一信息，以简化 ALI 中另一个 SCM 系统的配置。

### 要从特定内部版本配置应用现有 SCM 配置：

1. 单击“管理”模块**内部版本服务器**页上的某个服务器。
2. 选择**内部版本配置**选项卡。
3. 选择配置并单击它以打开**内部版本配置详细信息**弹出窗口。
4. 选择 **SCM** 选项卡以显示可用的存储库和分支。
5. 选择必需的存储库和/或分支并单击 + 按钮将其应用到 SCM 配置。

存储库或分支位置旁的状态消息提供以前是否定义了存储库或分支的信息。

**注：**单击详细信息按钮将打开选择的存储库或分支详细信息弹出窗口。

## 设置内部版本配置缺陷筛选

对于此内部版本配置，此筛选设置会缩小与内部版本报告缺陷趋势图上所示与发布相关的打开缺陷和新缺陷的输入集范围。不要在此筛选中指定缺陷状态和目标发布。

### 要对特定内部版本配置应用缺陷筛选：

1. 单击“管理”模块**内部版本服务器**页上的某个服务器。
2. 选择**内部版本配置**选项卡。
3. 单击配置名称以打开**内部版本配置详细信息**窗口。
4. 选择**缺陷筛选**。
5. 单击**更改缺陷筛选**以打开筛选缺陷窗口。
6. 添加或删除某个筛选条件并单击**确定**应用该条件。

### 要从特定内部版本配置删除缺陷筛选：

单击**清除缺陷筛选**，并单击**是**确认删除。

## 自定义 ALI 项目列表

在 ALM “项目自定义”中，可以在“项目列表”页自定义以下默认 ALI 项目列表：

ALM 项目列表	描述
ALI 内部版本类别	定义 <b>内部版本类别</b> 字段中的可用值。
ALI 已关闭状态	定义 <b>状态</b> 字段值的子集。当缺陷的 <b>状态</b> 字段设置为此列表中定义的某个值时，ALI 将该缺陷视为已关闭。只能在缺陷被视为已关闭时设置 <b>关闭于内部版本</b> 字段的值。
ALI 代码问题定义	定义 ALI 用于将缺陷标记为 <b>代码问题</b> 的值。 有关代码问题的更多详细信息，请参见 <a href="#">“代码问题跟踪”</a> （第 27 页）。
ALI QA 状态	定义内部版本的 <b>QA 状态</b> 字段中的可用值。
ALI 报告的严重度	定义指向代码问题的链接可用的值、“需求”或“缺陷”模块中的“开发活动”选项卡以及“内部版本详细信息”中显示的值。这些值必须与“严重度”项目列表中的值相同或是其子集。

有关自定义项目列表的更多详细信息，请参考《HP Application Lifecycle Management 管理员指南》。

## 代码问题跟踪

ALI 使您可以跟踪**代码问题**。作为代码变更结果创建新缺陷时（例如，作为代码分析或测试过程的一部分），ALI 可以根据所设置的条件将该缺陷标记为代码问题。代码问题跟踪为代码问题缺陷添加从特定代码变更关联的需求、缺陷或内部版本到该缺陷的链接。

跟踪代码问题缺陷使您可以：

- 识别与特定需求关联的代码变更所产生的缺陷
- 识别与特定缺陷关联的代码变更所产生的新缺陷
- 识别特定内部版本中检测到的缺陷

在项目中配置代码问题跟踪后，可以查看从相关“需求”、“缺陷”或“内部版本详细信息”对话框到代码问题缺陷的链接。

有关配置代码问题跟踪的详细信息，请参见[“配置代码问题跟踪”](#)（第 27 页）。

## 配置代码问题跟踪

此部分描述如何在 ALM 项目中配置代码问题跟踪。要了解有关代码问题的详细信息，请参见[“代码问题跟踪”](#)（第 27 页）。

为了将缺陷识别为需求、缺陷或内部版本的代码问题，该缺陷必须满足以下条件：

- 缺陷不得被视为已关闭。这意味着缺陷**状态**字段不得设为 **ALI 已关闭状态**项目列表中定义的某个值。
- 缺陷的**优先级**字段不得设为 **ALI 报告的严重度**项目列表中定义的某个值。

- 必须为代码问题跟踪定义用户定义字段( 请参见下文) 。此字段必须在值定义为 **ALI 代码问题定义**项目列表中的子项的缺陷中设置。
- 必须适用以下某项：
  - 对于需求或缺陷：缺陷必须是与该缺陷或需求关联的代码变更导致的。
  - 对于内部版本：缺陷必须已在该内部版本中检测到。

**要配置代码问题跟踪：**

1. 在“项目自定义”的“项目实体”页中，定义要用于代码问题跟踪的用户定义字段。有关创建用户定义字段的更多详细信息，请参考《HP Application Lifecycle Management 用户指南》。
2. 在“项目列表”页中，将用户定义字段的**名称**字段值添加为 **ALI 代码问题定义**项目列表中的唯一列表项。例如，BG\_USER\_01。
3. 将新的用户定义字段的值添加为 **ALI 代码问题定义**项目列表中的子项。

## 设置内部版本服务器检测

Application Lifecycle Intelligence 检测内部版本服务器上的新内部版本，并加载新状态信息、关联的代码变更以及( 可选) 单元测试结果和代码覆盖率。自动在内部版本、代码变更和工作项( 缺陷、需求) 之间创建可跟踪性。与 SCM 集成一样，ALI 支持两种检测机制：轮询机制和 Push 机制。

可通过在“变更检测”视图中设置新内部版本服务器或单击现有内部版本服务器的“变更检测”选项卡来访问变更检测设置。

**轮询：**

使用轮询机制选项，ALI 定期检查给定内部版本服务器中的新内部版本，如果它们尚不存在于 ALM 服务器上，则加载它们。

使用“变更检测”选项卡上的**从内部版本服务器读取变更**选项激活轮询机制。可以计划运行、指定时间段来检查新内部版本或根据需要使用同步命令加载变更。

**Push：**

Push 机制的运行方式与轮询机制相反。在内部版本服务器上的 ALI 集成插件中选择了在 *HP ALM 中更新内部版本信息*选项时( 有关详细信息，请参见[“HP ALI Hudson/Jenkins 插件配置”](#)( 第 24 页) )，该插件将侦听该内部版本服务器。新内部版本启动时，它会将该内部版本的信息推送到 ALM 服务器。该内部版本完成时，ALM 服务器上的状态和其他内部版本相关信息将更新。

Push 机制通过“变更检测”选项卡上的**接收内部版本服务器代理程序传输的内部版本**选项激活。

## 第 4 章

---

### Force.com 集成

此集成允许开发 Force.com 平台的团队利用 ALI 为标准开发提供的所有功能。尽管所有源代码都在云中存储、编译和测试，但 ALI 建立了代码、工作项（需求和缺陷）、内部版本与内部版本度量（测试结果和覆盖率）之间的可跟踪性。

**注：**Force.com 集成在 Force.com 版本 API 22.0 上测试。

此章节包括以下内容：

先决条件 .....	30
项目部署、测试和报告生成 .....	30
Hudson/Jenkins Force.com 配置 .....	31

## 先决条件

Force.com 与 ALI 之间的集成需要以下条件:

- Force.com 源代码存储在 SCM 系统中。有关 HP ALI 支持的 SCM 系统的完整列表, 请参见 [“支持的 SCM 系统”](#) (第 13 页)。
- Hudson 或 Jenkins
  - 支持使用 SCM(默认情况下支持 SVN 和 CVS) 的 Hudson/Jenkins 插件。
  - HP ALI Hudson 插件 (ali-bundle.zip/build/Hudson/ali-hudson-plugin.hpi)。  
有关详细信息, 请参见 [“HP ALI Hudson/Jenkins 插件安装”](#) (第 23 页)。
- 配置内部版本管理服务器以将源代码部署到集成/临时环境。
- Apache Ant(从 <http://ant.apache.org/> 下载)。
- HP force-deploy-task 位于 ALI 存档中( /tools/force-deploy-task/force-deploy-task-bundle.zip, 将 zip 文件的内容解压缩到 ant\_install\_dir/lib)。

## 项目部署、测试和报告生成

源代码的部署、测试及以下报告生成通过特殊 Ant 任务 *HP force-deploy-task* 得以确保。为了正常运行, 必须在 Force.com 项目的根文件夹下创建 Ant 构建脚本 “build.xml” (如果不存在), 其内容与下面的示例类似:

1. 以下示例将源代码部署到配置的 Force.com 环境并运行所有测试。由于将运行所有测试, 因此报告应包括整个项目的代码覆盖率。

```
<project name="Sample usage of force-deploy-task" default="
deployAndTestAndReport " basedir=".">
  <target name="deployAndTestAndReport">
    <taskdef name="sfdeploy"
classname="com.claimvantage.force.ant.DeployWithXmlReportTask"/>
    <delete dir="test-report-xml" quiet="true"/>
    <sfdeploy
username="username to force.com environment"
password="password to force.com environment"
serverurl="force.com server URL"
deployRoot="path to source directory"
runalltests="true"
reportDir=" test-report-xml " />
  </target>
</project>
```

2. 以下示例将源代码部署到配置的 Force.com 环境并仅运行与给定模式匹配的测试。在这种情况下, 将不会为 ALM 提供完整代码覆盖率。

```
<project name="Sample usage of force-deploy-task" default="
deployAndTestAndReport " basedir=".">
  <target name="deployAndTestAndReport">
    <taskdef name="sfdeploy"
classname="com.claimvantage.force.ant.DeployWithXmlReportTask"/>
```

```
<delete dir="test-report-xml" quiet="true"/>
<sfdeploy
  username="username to force.com environment"
  password="password to force.com environment"
  serverurl="force.com server URL"
  deployRoot="path to source directory"
  runalltests="false"
  reportDir=" test-report-xml ">
  <!-- Run only tests with file names that match this pattern -->
  <batchtest>
    <fileset dir="src/classes">
      <include name="*Test.cls"/>
    </fileset>
  </batchtest>
</target>
</project>
```

HP force-deploy-task 的描述( 在定义为 sfdeploy 的示例中) :

- 属性 *username* —— force.com 环境的登录名
- 属性 *password* —— force.com 环境的密码
- 属性 *serverurl* —— force.com 环境的登录页的 URL
- 属性 *deployRoot* —— 源代码目录( 文件夹类、触发器等所在的位置) 的路径
- 属性 *runalltests* —— (true/false) 如果为 true, 则启动所有测试并报告项目的代码覆盖率; 如果为 false, 则仅启动由 batchtest 元素指定的测试( 不提供代码覆盖率)
- 属性 *reportDir* —— 将存储所有报告的目录
- 元素 *batchtest* —— 指定应启动哪些测试的元素( 仅在 runalltests=true 时起作用)
  - 元素 *fileset* —— 要运行的测试的文件集
    - 属性 *dir* —— 测试所在的目录
    - 元素 *include*
      - 属性 *name* —— 要运行的测试的类名模式

## Hudson/Jenkins Force.com 配置

1. 创建自由样式的作业并根据需要配置 SCM 和构建触发器
2. 添加构建步骤 *Invoke Ant* ( 调用 *Ant*) 并指定应启动的目标( 如示例 *deployAndTestAndReport*) 。
3. 在 *Post-build Actions* ( 构建后操作) 部分中, 按照以下图像配置:

Publish JUnit test result report ?

Test report XMLs  ?

Fileset 'includes' setting that specifies the generated raw XML report files, such as 'myproject/target/test-reports/\*.xml'. Basedir of the fileset is [the workspace root](#).

Retain long standard output/error ?

ALI Integration ?

Test sources mapping pattern  ▼

Regex pattern for locating test sources, e.g. (.\*)?target/surefire-reports/.\*.xml//\${1}/src/test/java

4. 在测试报告 XML 中, 将 test-report-xml 字符串替换为实际的报告路径 (force-deploy-task 中的属性 reportDir)。

5. 在测试源映射模式中, 将 src 替换为实际的源目录路径。

上述配置对于大多数情况已足够, 但如果:

- force-deploy-task 在从主 Ant 脚本调用的分布式 Ant 脚本中定义, 则在 Ali Integration/Advanced (ALI 集成/高级) 中配置报告目录 (Ant 脚本中属性 reportDir 的值)。
- 源代码目录 (包括文件夹类、触发器等) 不在直接位于工作区根目录下的 src 目录中, 则配置项目根目录。

配置示例:

**Force.com**

Report directory	<input type="text" value="test-report-xml"/>
	<small>Relative path to directory where reports are generated by HP force-deploy-task Ant task (attribute reportDir). Consider <a href="#">workspace root</a> as basedir.</small>
Project root	<input type="text" value="src"/>
	<small>Relative path to directory with source codes (consider <a href="#">workspace root</a> as basedir). Default value is src.</small>



## 第 5 章

---

### 监控 SCM 变更和可跟踪性

设置存储库、分支和实施规则后，Application Lifecycle Intelligence (ALI) 将汇总数据并在 ALM 的“代码变更”模块中显示“变更集”。此模块提供有关从源代码视角查看项目内所发生状况的信息，并维护源代码、需求/缺陷与发布之间的可跟踪性。

报告功能是有有效项目管理的一个重要部分，ALI 使用涵盖代码变更实体的额外报告扩展了 HP ALM Platform 报告。

可以使用项目报告和图（摘要、进度和趋势）查看并比较团队和开发者的开发活动。此外，ALI 还提供了变更影响报告来描述所执行变更对缺陷和需求的影响。

此章节包括以下内容：

查看代码变更表 .....	34
代码变更详细信息 .....	34
查看变更影响报告 .....	35
生成项目报告 .....	35
生成图 .....	35

## 查看代码变更表

代码变更模块在 ALM 侧栏上的“开发”下提供。

初始显示列出选定发布和给定时间段的所有代码变更。这可以使用表顶部的下拉菜单进行修改和筛选。

日期时间	开发者	文件	行数	消息	状态消息
5/26/2011 12:49:56 PM	bill	1	3	defect #37: use standard mechanism for asynchronicity	
5/26/2011 10:48:16 AM	john	1	16	defect #40: make name/summary clickable too	Defect 40 does not es
5/26/2011 10:36:01 AM	david	1	8	requirement #1: documentation update	
5/26/2011 8:48:00 AM	luke	2	27	defect #41: add missing readme	Defect 41 does not es
5/25/2011 6:37:00 PM	bill	1	2	Incomplete - task REQ2712: Portal Semantics http://	Requirement 2,712 d
5/25/2011 6:34:20 PM	bill	1	2	Incomplete - task REQ2709: Reporting DB Schema http://	Requirement 2,709 d
5/25/2011 6:31:14 PM	bill	1	8	Incomplete - task REQ2709: Reporting DB Schema http://	Requirement 2,709 d
5/25/2011 6:08:47 PM	john	4	26	defect #37: creation time of the report, minor layout modifications	
5/25/2011 5:14:53 PM	david	3	9	requirement #1: update of documentation	
5/25/2011 5:02:42 PM	robert	3	13	Requirement #2: Added missing resource.	
5/25/2011 4:54:28 PM	robert	1	1	fixing defect #41: column selection not persisted	Defect 41 does not es
5/25/2011 4:42:50 PM	luke	6	135	defect #38: enforcement and push support more projects	
5/25/2011 4:37:35 PM	luke	1	1	requirement #3: new documentation	

表显示变更的日期/时间、执行变更的开发者、受影响的文件数和更改的行数以及由开发者提供的提交消息。当提交消息与预定义模式不对应时，状态消息将显示错误或警告。可通过启用查看菜单下的筛选来修改和筛选该表中的列。

黑色指示存在变更集的关联需求。

绿色指示存在变更集的关联缺陷。

红色指示既不存在与变更集关联的缺陷也不存在关联的需求。

除了发布和时间，还可以显示只与需求或缺陷关联的数据以及未分配的变更。


**注：**未分配的变更是与需求和缺陷都不相关的提交。这对于处理计划外变更的团队管理员是非常有价值的信息。

可通过单击筛选按钮  访问高级筛选。

启用查看菜单下的“信息面板”可在该表下方打开面板，其中包括查看变更集内的关联文件、需求和缺陷的选项卡。还可以在此面板中查看提交消息或状态消息。

## 代码变更详细信息

要查看代码变更详细信息：

- 双击“代码变更”或单击“代码变更详细信息”按钮  以在新窗口中打开代码变更。
  - 除了基本属性外，还可以看到关联的文件的列表及其状态，并且还可以比较与文件的以前版本的差异。
  - 如果变更集与需求或缺陷关联，则可通过单击左侧菜单中的“需求”或“缺陷”菜单项提供这些链接。

## 查看变更影响报告

ALI 不仅报告文件和模块的相关变更影响，还报告分配给开发者的工作项（需求和缺陷）。它提供有关代码覆盖率、测试结果、变更量以及提交受影响工作项的人员的汇总信息。

### 要查看变更影响报告：

1. 选择时间段并单击**查看报告**。  
 基于该时间段的报告将在新窗口中显示。  
 报告显示在选定时间段内需求和缺陷已发生的变更。  
 报告右上角的图显示以 KLOC（千行代码）为单位的项目的工时分布。
2. 单击**显示未分配的变更**以显示未分配的变更。
3. 单击**打印此报告** 以将报告发送到打印设备。
4. 单击**通过电子邮件发送此报告**可打开电子邮件客户端并将报告的 URL 发送到提供的地址。

## 生成项目报告

ALI 扩展提供了一组已添加到标准 ALM 报告选择中的预定义报告模板。

预定义模板包括：

- ALI - 代码变更模板
- ALI - 缺陷概述模板
- ALI - 需求概述模板
- ALI - 内部版本模板

使用这些模板可创建与代码变更和内部版本相关的项目报告。

有关生成项目报告的详细信息，请参见《HP Application Lifecycle Management 用户指南》。

## 生成图

使用 ALI 扩展可为代码变更生成图。

可以针对代码变更生成以下图：

图	描述
代码变更进度图	<p>显示 ALM 项目中在某一时间段内的特定点积累的代码变更数。</p> <p>指定沿 x 轴显示的时间间隔，以及 ALM 对数据分组所依据的需求信息。指定是要查看与开发者相关的代码变更计数、更改的文件数总和还是更改的行数总和。</p>
代码变更摘要图	<p>显示当前 ALM 项目中的代码变更数。</p> <p>指定沿 x 轴显示的数据类型，以及 ALM 对数据分</p>

	组所依据的代码变更信息。
代码变更趋势图	<p>针对显示的每个时间间隔，显示 ALM 项目中特定代码变更字段的变更历史记录。</p> <p>指定要查看其变更数的字段以及与要查看数据对应的时间段。</p>

要生成图，请执行以下某项操作：

#### 1. 启动图向导。

图向导指导您完成在创建图和配置其设置中涉及的所有步骤。可以从“分析视图”模块或在“代码变更”模块中工作时启动图向导。

##### a. 在“分析视图”模块中。

在 ALM 侧栏上的“控制面板”下方，选择**分析视图**，右键单击文件夹，然后选择**图向导**。

##### b. 在“代码变更”模块中。

选择**分析 > 图 > 图向导**。

##### c. 在“图向导”窗口中，更改**代码变更**的实体名称。列出了三种图类型。选择相应的图并按照向导说明操作。

#### 2. 在“代码变更”模块中创建预定义图。

选择**分析 > 图**并选择以下某个预定义图：

- 代码变更 - 摘要图 - 按‘开发者’分组
- 代码变更 - 摘要图 - 按‘开发者’分组的‘文件数’总和
- 代码变更 - 摘要图 - 按‘开发者’分组的‘行数’总和
- 代码变更 - 进度图 - 按‘开发者’分组
- 代码变更 - 趋势图 - 按‘开发者’分组

#### 3. 在分析视图中创建图。

在 ALM 侧栏上的“控制面板”下方，选择**分析视图**。右键单击文件夹，选择**新建图**。

#### 4. 在“分析视图”模块的“分析菜单”选项卡中创建预定义图。

在 **代码变更主分析菜单**文件夹下，有以下预定义图可用：

- 代码变更 - 摘要图 - 按‘开发者’分组
- 代码变更 - 摘要图 - 按‘开发者’分组的‘文件数’总和
- 代码变更 - 摘要图 - 按‘开发者’分组的‘行数’总和
- 代码变更 - 进度图 - 按‘开发者’分组
- 代码变更 - 趋势图 - 按‘开发者’分组

有关生成图的详细信息，请参考《HP Application Lifecycle Management 用户指南》。

## 第 6 章

---

### 监控开发活动

设置内部版本服务器和配置后, Application Lifecycle Intelligence (ALI) 将在 ALM 的“发布”、“缺陷”和“需求”模块中显示“开发活动”选项卡。此选项卡显示有关从内部版本和开发者视角查看项目内所发生状况的信息。

选项卡内的链接提供选定内部版本的详细信息和统计信息。

选项卡底部的表显示每个开发者的内部版本中的变更。

单击**更改**可查看其他配置的内部版本的活动。

此章节包括以下内容:

开发活动 - 发布 .....	38
开发活动 - 缺陷 .....	38
开发活动 - 需求 .....	38
监控缺陷详细信息中的开发活动 .....	38

## 开发活动 - 发布

如果存在与发布关联的开发活动，选项卡上将显示绿色星形图标。单击选项卡可查看活动。

选项卡中显示了包含成功率、构建时间、修复时间的“内部版本状态”以及以 KLOC（千行代码）为单位的“基本代码统计信息”，使用户可以清楚地看到内部版本的进展。单击选项卡中的这些链接可查看详细信息。

单击**更改**可选择查看非默认的其他内部版本配置。

单击**配置内部版本**可更改配置。

有关详细信息，请参见[“添加内部版本配置”（第 25 页）](#)。

选项卡底部的表显示与发布关联的开发者列表以及每个开发者的贡献级别。

## 开发活动 - 缺陷

如果存在与缺陷关联的开发活动，选项卡上将显示绿色星形图标。单击选项卡可查看活动。

选项卡显示需求的包括单元测试和覆盖率的“开发度量”及“缺陷实施”。单击“单元测试”链接可查看**缺陷单元测试报告**，单击“覆盖率”链接可查看**缺陷覆盖率报告**。

单击“缺陷实施”下方的链接可查看内部版本详细信息或特定代码变更。

选项卡底部的表显示与缺陷关联的“代码变更”和“活动开发者”。

单击代码变更中的消息链接将转到“缺陷详细信息”页。

单击列标题可对表进行排序或筛选。

## 开发活动 - 需求

如果存在与需求关联的开发活动，选项卡上将显示绿色星形图标。单击选项卡可查看活动。

选项卡显示需求的“开发度量”和“需求实施”。单击“单元测试”链接可查看**需求单元测试报告**，单击“覆盖率”链接可查看**需求覆盖率报告**。

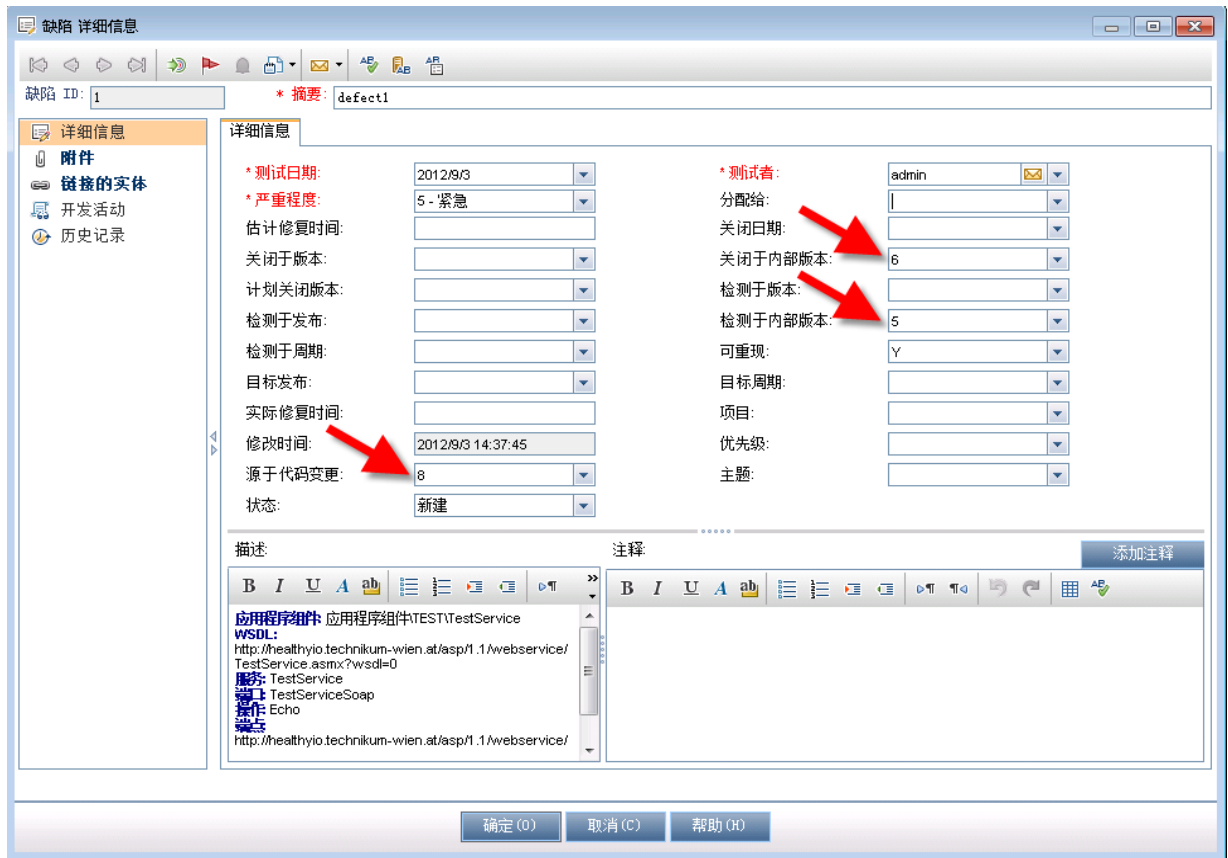
选项卡底部的表显示与需求关联的特定“代码变更”和“活动开发者”。

单击代码变更中的消息链接将转到“代码变更详细信息”页。

单击列标题可对表进行排序或筛选。

## 监控缺陷详细信息中的开发活动

Application Lifecycle Intelligence 在“缺陷详细信息”页引入了三个新值。



**检测于内部版本** —— 指示检测到给定缺陷的内部版本。

**关闭于内部版本** —— 指示已关闭给定缺陷的内部版本。

**源于代码变更** —— 指示导致此缺陷的变更集。

如果已配置支持与 HP ALM Platform 集成的静态代码分析工具，将自动填充这些值。如果未配置，可以使用下拉列表手动填充。

这些字段中的值将填充“内部版本详细信息”页中的缺陷选项卡并显示在内部版本摘要报告中。有关详细信息，请参见“[内部版本详细信息](#)”（第 43 页）和“[内部版本报告](#)”（第 45 页）。

## 第 7 章

---

### 内部版本-测试可跟踪性

使用“内部版本-测试可跟踪性”可将内部版本与测试关联并跟踪这些关联。

此章节包括以下内容：

查看内部版本-测试可跟踪性 .....	41
---------------------	----



## 查看内部版本-测试可跟踪性

作为 QA 工程师或管理员，内部版本-测试可跟踪性使您可以创建并跟踪特定内部版本与特定测试集和运行之间的关联。

运行测试之前，请先定义测试集。然后可以定义要用于测试该测试集的内部版本。这意味着即将在选定内部版本上对所有测试进行测试。基于您定义的测试与内部版本之间的这一关联，可以从 QA 的角度监控内部版本的质量。可以计划并审阅特定内部版本的测试的测试结果。对于特定内部版本，可以审阅由 QA 工程师执行的测试集和测试运行的结果。

可以在以下位置设置和查看内部版本与测试之间的关联的详细信息：

位置	内部版本-测试可跟踪性
“内部版本”模块 > 内部版本报告	在“内部版本详细信息”对话框中，单击 <b>查看报告</b> 可查看内部版本变更报告。
“内部版本”模块 > “测试”选项卡	在“内部版本”模块窗口或“内部版本详细信息”对话框中，单击 <b>测试</b> 选项卡。
“测试实验室”模块 > “测试集详细信息”对话框	<b>测试内部版本</b> 字段显示与选定测试集关联的内部版本号。选择要与测试集关联的内部版本。
“测试实验室”模块 > “手动运行器”对话框	<b>测试内部版本</b> 字段显示与选定运行关联的内部版本号。
“测试实验室”模块 > “运行详细信息”对话框	<b>测试内部版本</b> 字段显示与选定运行关联的内部版本号。

有关“测试实验室”模块的更多详细信息，请参考《HP Application Lifecycle Management 用户指南》。

## 第 8 章

---

### 监控内部版本活动

设置内部版本系统后，Application Lifecycle Intelligence 会监控内部版本结果以及相关代码和执行的工作项。

在“内部版本”模块中，可以查看“快速”、“夜间”或“集成”内部版本的状态和结果，并在报告错误时执行操作。根据内部版本系统配置，您可以监控单元测试的结果、成功或失败的测试数、代码覆盖率、源代码和缺陷统计信息。

此外，还提供了内部版本的特定项目报告和图。有关详细信息，请参见 [“内部版本报告”](#)（第 45 页）和 [“生成内部版本图”](#)（第 45 页）。

此章节包括以下内容：

查看内部版本 .....	43
内部版本详细信息 .....	43
内部版本报告 .....	45
生成内部版本图 .....	45

## 查看内部版本

登录到启用了 ALI 扩展的项目时，**内部版本**模块将可用。

默认视图显示选定发布和给定时间段的所有内部版本。这可以使用表顶部的下拉菜单进行修改和筛选。

日期/时间	数字	内部版本状态	标签	QA 状态	单元测试	覆盖率	持续时间	内部版本报告	内部版本系统
2012/8/23 12:54:53	57						1 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/23 12:57:13	58						7 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/23 13:01:00	59				100%		36 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/23 13:02:55	60				100%		24 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/23 13:38:12	61				100%		29 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/23 13:41:32	62				100%		24 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/23 13:47:38	63				100%		24 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/23 15:31:42	64				100%		47 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/24 14:25:43	65				75%		1 分钟 16 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/24 14:36:06	66				100%		25 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/24 14:48:53	67				100%		25 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/8/27 11:17:01	66				100%		1 分钟 14 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/9/10 15:00:48	69				75%		1 分钟 25 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>
2012/9/10 15:08:23	70	警告			100%		23 秒	<a href="#">查看报告</a>	<a href="#">查看详细信息</a>

该表显示内部版本的日期/时间以及为内部版本记录的所有其他信息。可通过启用**查看**菜单下的筛选来修改和筛选该表中的列。

“内部版本状态”字段按下面的颜色显示每个状态：

- 绿色 = 成功。指示成功的内部版本。
- 黄色 = 警告。指示内部版本有问题。
- 红色 = 失败。指示内部版本失败。

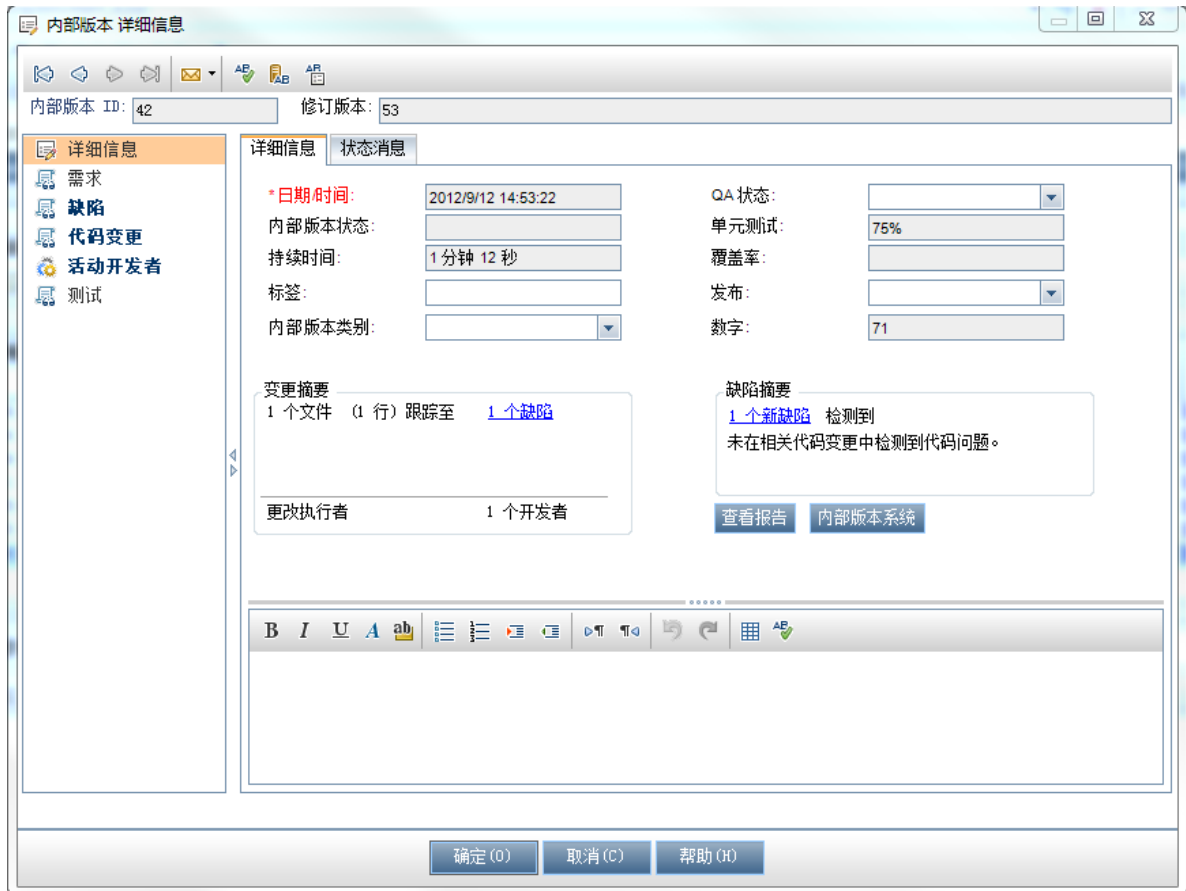
可通过单击筛选按钮  访问高级筛选。


在“查看”菜单下，选择**信息面板**可在该表下方显示面板，其中包括查看该内部版本的需求、缺陷、代码变更、测试、分布和状态消息的选项卡。

有关查看单个内部版本的更多详细信息的消息，请参见 [“内部版本详细信息”](#)（第 43 页）。

## 内部版本详细信息

使用“内部版本详细信息”对话框可查看有关选定内部版本的详细信息。



访问	在“内部版本”模块中，双击内部版本，或选择内部版本并单击 <b>内部版本详细信息</b> 按钮  。
另请参见	<a href="#">“监控内部版本活动”</a> （第 42 页）

以下 UI 元素可用：

UI 元素	描述
详细信息	<p>列出内部版本字段。</p> <p>单击<b>查看报告</b>可生成内部版本变更报告。有关详细信息，请参见 <a href="#">“内部版本报告”</a>（第 45 页）。</p> <p>单击<b>内部版本系统</b>可生成整个内部版本系统的视图。</p> <p>单击代码问题链接可查看相关代码问题缺陷的详细信息。有关详细信息，请参见 <a href="#">“代码问题跟踪”</a>（第 27 页）。</p>
“需求”选项卡	<p>列出与内部版本关联的需求。</p> <p>单击列标题可对列表排序。</p> <p>单击需求可打开“需求详细信息”页。</p>

<b>“缺陷”选项卡</b>	<p>列出与内部版本关联的缺陷。</p> <p>选择选项卡可查看“检测到的缺陷”、“关闭的缺陷”和“交付的修复”。选项卡上的绿色星形图标指示网格中有新条目。</p> <p>单击列标题可对列表排序。</p> <p>单击缺陷摘要中的链接可打开“缺陷详细信息”对话框。</p>
<b>“代码变更”选项卡</b>	<p>列出与内部版本关联的代码变更。</p> <p>单击列标题可对列表排序。</p> <p>单击代码变更消息可打开“代码变更详细信息”对话框。</p>
<b>“活动开发者”选项卡</b>	<p>列出与内部版本关联的开发者以及贡献级别。</p> <p>单击列标题可对列表排序。</p>
<b>“测试”选项卡</b>	<p>列出与内部版本关联的测试运行的详细信息。选项卡上的绿色星形图标指示该选项卡已填充。</p> <p>可以查看所有测试运行的状态。单击测试计数可查看所有测试实例的列表。</p> <p>摘要框的最后一行显示运行的状态以及按状态分组的运行计数。</p>

## 内部版本报告

**内部版本摘要报告**提供给定时间段和内部版本类别的内部版本的概述视图。该图合并 SCM 活动的信息并从选定发布打开缺陷。

**要查看内部版本摘要报告：**

1. 从 Application Lifecycle Intelligence 的“开发”模块中选择“内部版本”。
2. 单击**查看报告**按钮以在新窗口中生成摘要报告。

**特定内部版本报告**显示与该内部版本关联的需求、已交付缺陷、新缺陷和已关闭缺陷。

**要查看特定内部版本报告：**

1. 在摘要报告上单击该内部版本的链接，或打开内部版本详细信息页并单击**查看报告**按钮。
2. 单击**显示未分配的变更**链接以显示未分配的变更。

**内部版本变更报告**提供受选定内部版本影响的需求以及已由选定内部版本更改、关闭或创建的缺陷的概述。

**要查看内部版本变更报告：**

1. 从开发模块的**内部版本**页，选择一个内部版本并打开**内部版本详细信息**页。
2. 单击**查看报告**以在新窗口中生成变更报告。

## 生成内部版本图

使用 ALI 扩展可为内部版本生成图。

可以生成以下图：

图	描述
内部版本摘要图	显示当前 ALM 项目中的内部版本数。 指定沿 x 轴显示的数据类型，以及 ALM 对数据分组所依据的内部版本信息。
内部版本趋势图	针对显示的每个时间间隔，显示 ALM 项目中特定内部版本字段的内部版本历史记录。 指定要查看其内部版本数的字段以及与要查看数据对应的时间段。

要生成图，请执行以下某项操作：

#### 1. 启动图向导。

图向导指导您完成在创建图和配置其设置中涉及的所有步骤。可以从“分析视图”模块或在“内部版本”模块中工作时启动图向导。

- 在“分析视图”模块中。

在 ALM 侧栏上的“控制面板”下方，选择**分析视图**，右键单击文件夹，然后选择**图向导**。

- 在“内部版本”模块中。

选择**分析 > 图 > 图向导**。

- 在“图向导”窗口中，更改内部版本的实体名称。列出了两种图类型。选择相应的图并按照向导说明操作。

#### 2. 创建预定义图。

在“内部版本”模块中工作时，将生成临时图以分析模块数据。

选择**分析 > 图**并选择以下某个预定义图：

- 内部版本 - 摘要图 - 按 ‘内部版本配置’。
- 内部版本 - 摘要图 - 按 ‘部版本状态’ 分组的 ‘内部版本配置’。
- 内部版本 - 摘要图 - 按 ‘QA 状态’。
- 内部版本 - 趋势图 - 按 ‘内部版本配置’ 分组。

#### 3. 在分析视图中创建图。

在 ALM 侧栏上的“控制面板”下方，选择**分析视图**。右键单击文件夹，选择**新建图**。

#### 4. 在“分析视图”模块的“分析菜单”选项卡中创建预定义图。

在“内部版本主分析菜单”文件夹下，有以下预定义图可用：

- 内部版本 - 摘要图 - 按 ‘内部版本类别’。
- 内部版本 - 摘要图 - 按 ‘内部版本状态’ 分组的 ‘内部版本类别’。
- 内部版本 - 摘要图 - 按 ‘QA 状态’。
- 内部版本 - 趋势图 - 按 ‘内部版本类别’ 分组。

有关生成图的详细信息，请参考《HP Application Lifecycle Management 用户指南》。

此外，还为内部版本提供了特定项目报告。有关详细信息，请参见 [“生成项目报告”](#)（第 35 页）。