|  |  |
| --- | --- |
| **项目名称：** | **InWise 8.0** |
|  |  |
| **文档名称：** | **配置管理计划** |
|  |  |
| **文档编号：** | **CX08-06** |

|  |  |  |  |
| --- | --- | --- | --- |
| **拟制：** | 刘华先 | **日期：** | 2020.6.3 |
|  |  |  |  |
| **审核：** | 董金彩 | **日期：** | 2020.6.3 |
|  |  |  |  |
| **批准：** | 刘偊赜 | **日期：** | 2020.6.3 |

修订记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **版本** | **修订说明** | **修订者** | **修订日期** |
| 1 | 1.0 | 创建该文件 | 刘华先 | 2020.6.3 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目 录

[1 概述 5](#_Toc127609778)

[1.1 目的 5](#_Toc127609779)

[1.2 范围 5](#_Toc127609780)

[1.3 定义和术语 5](#_Toc127609781)

[2 项目配置管理资源 5](#_Toc127609782)

[2.1 角色和职责 5](#_Toc127609783)

[2.2 工具、环境和基础设施 5](#_Toc127609784)

[3 配置管理任务 6](#_Toc127609785)

[3.1 配置标识 6](#_Toc127609786)

[3.1.1 标识方法 6](#_Toc127609787)

[3.1.2 识别配置项 6](#_Toc127609788)

[3.1.3 选取和命名配置项 6](#_Toc127609789)

[3.1.4 项目基线 6](#_Toc127609790)

[3.2 配置和变更控制 6](#_Toc127609791)

[3.2.1 配置控制要求 6](#_Toc127609792)

[3.2.2 版本控制 6](#_Toc127609793)

[3.2.3 变更请求的处理和审批 6](#_Toc127609794)

[3.3 配置状态统计 6](#_Toc127609795)

[3.4 配置审核 7](#_Toc127609796)

[4 阶段里程碑和日程表（可考虑用project） 7](#_Toc127609797)

[5 培训 7](#_Toc127609798)

# 概述

## **目的**

由于在系统移植的过程中各种软件存在多个版本，为使系统稳定、流畅的运行，为防止软件出现版本冲突，因此需要规范统一软件版本。为了更好的为项目提供支持，特制定此计划。以更好的为项目服务。

## 范围

本文档适用于InWise 8.0系统开发项目，并对后续项目起参考作用，本文档将规范此次项目开发的一些工具及其他开发相关事件。

## 定义和术语

提供正确理解本文档所需的全部术语的定义。

GitLab：版本控制系统。

# 项目配置管理资源

## 角色和职责

对于任何一个管理流程来说，保证该流程正常运转的前提条件就是要有明确的角色、职责和权限的定义。特别是在引入了软件配置管理的工具之后，比较理想的状态就是：组织内的所有人员按照不同的角色的要求、根据系统赋予的权限来执行相应的动作。因此，在本文档中所说明的这个软件配置管理过程中主要涉及下列的角色和分工：

l 项目经理（Project Manager，PM）：

项目经理是整个软件研发活动的负责人，他配置管理的各项活动并控制它们的进程。其具体职责为以下几项：

制定和修改项目的组织结构和配置管理策略；

批准、发布配置管理计划；

决定项目起始基线和开发里程碑。

l 开发人员（Developer，DEV）：

开发人员的职责就是根据组织内确定的软件配置管理计划和相关规定，按照软件配置管理工具的使用模型来完成开发任务。

## 工具、环境和基础设施

1. 使用InWise 8.0 进行开发。
2. 系统开发过程中使用单独的ARM架构的PC。
3. 开发好的软件做成软件包需要上传到GitLab进行备份及保存。

# 配置管理任务

## 配置标识

### 标识方法

在系统移植过程中，无错误的软件包将使用和原软件包相同的命名方式，如果需要修改，则将软件包的SPEC文件进行修改，修改后软件包版本自动进行升级，软件包名字中统一含有ax7字样。所有的移植修改的软件包均上传到GitLab留作备份及后续项目开发时的参考。

### 识别配置项

项目中使用的版本控制及备份软件是GitLab，系统BUG提交及研发人员修改确认的软件是Bugzilla。

### 选取和命名配置项

项目中软件包修改的命名规则是在SPEC文件中进行修改，修改的规则如下：

1. 符合SPEC文件的书写语法。
2. 修改源码之后要进行changelog的编写。
3. 添加版本号需要在%{dist}之后加“.1”，如果已经存在“.x”，则将此版本号增加1。

### 项目基线

项目分成系统移植成功、beta1版本测试、beta2版本测试、beta3版本测试及RTM版本发布几个阶段。项目经理对以上阶段进行控制及授权。

## 配置和变更控制

### 配置控制要求

在系统软件移植的过程中，需要首先确定稳定的需要移植的软件版本，以CentOS 7.2系统使用的软件版本为基准，在Cent OS 系统中没有的软件包，使用Fedora 19的软件包。如果需要的软件包为以上两个系统中都不存在的软件包的时候，以客户需要的软件包的版本为基准在需要进行软件包版本修改或者软件包增加的时候需要向项目经理提交申请，并通知所有项目人员。

### 版本控制

在系统中使用项目起始定义的系统版本为准，将项目中的版本不同的软件进行冲突解决。

### 变更请求的处理和审批

系统软件版本修改或者增加的时候需要向项目经理提交申请，在部门经理审批后进行软件移植，并将新移植的软件上传到GitLab。

## 配置状态统计

此项目在进行各种变更及产品发布，需要进行记录，并向项目经理进行提交申请，申请需要填表记录。

## 配置审核

项目的配置审核需要在项目立项之后进行制定，需要项目组全体人员进行参加，系统中软件包如果需要升级或者增加的时候需要向项目经理提交申请，项目经理审核过后需要通知全体项目人员。

# 阶段里程碑和日程表（可考虑用project）

1. 配置管理需要对项目的各项工具及软件版本进行控制。
2. 软件版本进行变更的时候需要项目经理批准并在移植过后进行充分测试。
3. 在GitLab上建立系统软件包管理项目，用于保存根据项目需要修改的软件包。
4. 系统移植完成后进行测试，测试各个软件包的稳定性。
5. 测试完成后，由项目经理进行产品发布。

# 培训

为项目全体人员进行GitLab使用培训及上传、命名规则的培训。