**北京航空航天大学计算机学院**

**硕士学位论文文献综述**

**论文题目：**适航审定中软件配置管理关键技术研究

**专 业**：计算机技术

**研究方向**：软件适航技术

**研 究 生**：王鑫冶

**学 号**：SY1506325

**指导教师**：曹庆华 教授

**北京航空航天大学计算机学院**

**2016年12月20日**

目 录

[摘 要 2](#_Toc471636776)

[一、论文选题的背景与意义 4](#_Toc471636777)

[二、国内外研究现状分析 5](#_Toc471636778)

[2.1软件适航方面 5](#_Toc471636779)

[2.2软件配置管理 10](#_Toc471636783)

[2.3符合适航标准的软件配置管理过程 13](#_Toc471636786)

[三、关键技术和方法 16](#_Toc471636790)

[3.1软件过程建模 16](#_Toc471636791)

[3.2软件过程建模的方法 16](#_Toc471636792)

[**主要参考文献** 20](#_Toc471636796)

摘 要

适航软件的适航性审查工作是目前国家民用大飞机项目的重要组成部分，在传统的适航审定过程中，审定人员通过对研制单位提交的审定材料进行人工审查，审查标准主要靠评审人员对标准文件的理解和评审人员的日常经验和阅历，不同评审人员在进行相同的评审活动时可能得出的结论有差异，甚至得出相反的结论，这将导致评审结果的不确定性，有争议性。对适航标准进行研究，做出有利于提高适航软件领域开发与审定的方法与工具具有重大意义。

**关键词：适航；软件配置管理；软件过程建模**

Abstract

Review job of Airworthiness software currently is an important part of the national civil aircraft project. In the traditional airworthiness certification process,reviewers review projects by reviewing documents submitted by devolepers manully, review standards mainly by the review of staff on the understanding of the standard documents and the experience of the reviewers, the different conclusions that may be reached when different reviewers conduct the same assessment activities, and even draw the opposite conclusion, which will lead to uncertainty and controversial in the assessment results. It is of great significance to study the standards of airworthiness do178c and to develop methods and tools to improve the level of validation of airworthiness software.

Key words: airworthiness;software configuration management; software process modeling

# 一、论文选题的背景与意义

本课题来源于工信部的民用飞机专项科研“航空发动机电子控制系统适航审定关键技术”中的子课题“航空发动机电子控制器软件和硬件适航审定技术研究”。针对大型客机涡扇发动机电子控制系统软件的适航审定问题，开展相关的基础技术研究。

由于国外长期对航空发动机实施技术封锁，国内航空发动机研制相对国外具有较大差距，与之配套的适航审定经验不足，给航空发动机适航审定带来很大困难。在传统的适航审定过程中，审定人员对研制单位提交的审定材料进行人工审查，审查标准主要靠评审人员对标准文件的理解和评审人员的日常经验和阅历，不同评审人员在进行相同的评审活动时可能得出的结论有差异，甚至得出相反的结论，这将导致评审结果的不确定性，有争议性。而且这种审查方式需要耗费大量的时间、人力，工作效率低、周期长、成本高、质量控制难以把握。

为了辅助评审人员对部分目标进行审定，本文深入研究适航软件研发与审定中生命周期数据与过程、目标之间的关系，通过对符合Do-178c标准的适航软件开发过程的配置管理过程进行建模，将软件生命周期数据相互间关系及变动规则用模型表达，并利用生成的模型开发适航审定辅助工具，半自动的对研制材料进行初步审定。

我国航空发动机制造技术还处于较为艰难的发展阶段，发动机相应的适航研究也不是特别的深入。研究Do178c标准及其管理工具对于我国的航空工业来说有重要意义，可以推进我国航空发动机控制软件审查过程的规范化。

# 二、国内外研究现状分析

## 2.1软件适航方面

对于民用飞机在设计过程中如何满足民航当局的适航标准要求，国外发展较早，已经形成了较为完善的理论体系，提出了相应的适航符合性验证流程。我国由于民机研制起步较晚，民机安全性设计落后，在适航符合性验证方面的研究才刚刚起步，现行方法大都由国外标准直接翻译而来，适用性较差。 民用飞机需要经过民航标准体系的适航认证，Do-178c 标准是针对机载软件适航认证的一个重要标准。由于 Do-178c 是美国 RTCA 和欧洲 EUROCAE 两个组织共同制定的，所以在国外应用的已经比较成熟了。在中国基本是从 2005 年之后才开始传播，现在还处于起步阶段。目前国内已经引进并在学习该标准，但是还不能很好的理解应用它。

冯亚冬提出了一种适航认证的 FADEC 软件自动测试平台，主要是通过以 DO-178B 标准中软件测试过程要求为基础，从平台需求分析、技术论证、工具选型、接口定义、界面集成等五个阶段，详细讨论了 FADEC 系统控制软件上自动化测试平台的实现方法。

邢 亮，牟 明对Do178C目标及评审过程进行了研究，着重提出了每个过程中应实现的目标，详细分析了目标之间的关系，目标的分布与差异。以适航标准和指导性文件为依据，结合机载软件开发过程监控背景，对阶段介入评审进行机载软件开发过程评估关注的目标和数据进行了总结。

### 2.1.1我国适航认证体系

我国适航审定体系主要包括：航空法及条例、适航规章、管理程序、咨询通告、双边文件等。整个适航审定体系的构架和内容，都与美国联邦航空局（FAA） 基本相同。具体包括：13部中国民用航空规章、23个适航管理程序，23个咨询通报。其中和发动机有关的适航标准目前颁布了3部，分别为：CCAR33, CCAR34, CCAR35。其中，《航空发动机适航规定》（CCAR-33）是参考美国联邦航空条例FAR-33 第11修正案制定的，2011年1 月中国民用航空总局依据《中华人民共和国民用航空法》第三十四条，对《航空发动机适航标准》进行再一次修订，记为CCAR-33R2 部。现行的中国民航适航法规体系如图1所示：

****图1 中国民航适航法规体系

其中：

* 民用航空产品和零部件合格审定规定（CCAR-21）：主要针对民用航空产品和零部件合格审定规定。
* 《航空发动机适航规定》（CCAR-33R2）：主要针对民用航空发动机的适航标准。

在CCAR-33R2第33.28条第g节规定了对发动机控制系统软件的基本要求：申请人必须通过经中国民用航空局批准的方法设计、实现和验证所有相关软件，将软件错误存在的可能性减至最小，并符合其实施功能的关键性要求。然而针对发动机控制系统软件明确的适航审定方法却并没有给出，这势必对于执行适航审定机构造成没有实际审定依据的困难，因此，研究发动机控软件适航审定的基本体系，给出明确的适航审定依据就成为适航审定的基础。

### 2.1.2国际适航认证体系

我国在发动机控制系统软件适航审定方面还很不完善，因此，就很有必要学习国外比较成熟的适航审定体系，如FAA等国际通行的适航审定体系，再结合我国发动机控制软件实际研发情况，给出适合我国发动机控制软件适航审定的基本体系。

其中，有关航空发动机系统安全性评估的依据文件是FAR25.1309。其工业界最新的指导性标准ARP4761（民用飞机机载系统和设备安全性评估过程的指南和方法）、ARP4754A（民用飞机和系统开发指南）均为 SAE 颁发。系统安全性评估的方法，影响着软件等级的确定，进而影响着审定活动中一些重要输入，例如，决定控制类别，安全相关的派生需求或一些软件架构的反馈信息也会传送到系统安全评估过程，从而可能对软件级别发生改变。了解系统安全性评估的过程和方法能够帮助审定方更加准确地开展审定活动。因此，研究过程中将重点研读以上法规、咨询通告和标准中对于航空发动机控制软件安全性要求方面的内容，包括安全性设计的原则、符合性验证方法、故障条件与类别、发动机控制系统失效的定义、研制错误考虑等方面的阐述。在FAA最新的适航审定咨询通告AC20-115C(2013.7.19)中，给出了明确的基于DO-178C标准进行适航审定的要求。而在此之前的咨询通告AC20-115B ( 1993.1.11 -2013.7.19废除)，给出了明确的基于DO-178B标准进行适航审定的要求。

在以上所提标准中，发动机控制系统软件的安全等级都是要求满足A级要求。下表给出了DO-178C对不同软件等级要求目标的差异和描述。

表2 DO-178C不同软件等级的目标差异

|  |  |  |  |
| --- | --- | --- | --- |
| 级别 | 失效条件 | 目标数量 | 独立实现的目标 |
| A级 | 灾难性的 | 71（66） | 20(25) |
| B级 | 危害性的 | 69（65） | 18(14) |
| C级 | 严重的 | 62（56） | 2(2) |
| D级 | 不严重的 | 26（28） | 2(2) |
| E级 | 没有影响的 | 0（0） | 0（0） |

### 2.1.3 适航审定流程

对于航空机载软件的研制和适航审定，FAA确定了DO-178C、Order 8110.49chg1标准，其中，从适航符合性角度，DO-178C是基础标准，它定义了开发过程需要满足的与相应安全等级相适应的目标；Order8110.49 chg1则属于操作标准，定义了如何实际操作来审查一个项目的研制过程是否满足DO-178C的相应目标要求。DO-178C标准从软件安全性角度，按照机载软件的安全级别对开发过程和活动提出了一系列目标要求；后者则从符合性审查角度规定了具体的审查阶段、活动、涉及的数据和相应的审查规则等。

Order 8110.49 chg1针对DO-178C定义的开发阶段和相应的目标，规定适航符合性包括四次评审：针对软件计划阶段的符合性审查（SOI#1）、针对软件开发阶段的符合性审查（SOI#2）、针对软件验证阶段的符合性审查（SOI#3）和针对最终适航认证阶段的符合性审查（SOI#4），如表1所示。根据DO-178C标准，软件计划过程针对软件开发全过程涉及的活动和标准进行计划；软件开发过程是一个综合过程，包括开发、验证、配置管理、质量保证、审定联络；软件验证过程综合使用审查、评审、测试、分析等对软件制品进行验证。下表总结了这四个符合性审查阶段关注的目标和涉及的生命周期数据：

|  |  |  |  |
| --- | --- | --- | --- |
| 审查阶段 | 审查目标 | 合格判定准则 | 审查涉及数据 |
| 计划审查(SO#1) | 检查开发计划和标准是否为满足DO-178C目标要求定义了有效的开发和验证活动。 | 1. 开发计划和标准经过单位内部评审，且评审所发现问题已解决。 2. 开发计划和标准经过了质量保证部门评审，且评审所发现问题已解决。 3. 开发计划和标准得到了批准，并在配置管理中受控。 4. DO-178B/C中表格A-1（即计划阶段应满足的目标列表）所列目标得到了满足。 | 软件合格审定计划、软件开发计划、软件验证计划、软件验证结果、软配置管理计划、软件质量保证计划、软件需求、设计和编码标准、工具认证计划（如果涉及）、软件质量保证记录。 |
| 开发审查(SO#2) | 检查研制方是否按照相应计划开展了开发活动。 | 1. 形成了高层需求文档，经过了评审，并可追踪到系统需求。 2. 定义了软件体系结构，经过了评审和分析。 3. 形成了低层需求文档，经过了评审，并可追踪到高层需求。 4. 实现低层需求的代码可追踪到低层需求，并经过了评审。 | 软件需求、设计和编码标准、软件需求、设计描述、源代码、软件验证过程、软件验证结果、软件生命周期环境配置项、问题报告、软件配置管理记录、软件质量保证记录。 |
| 验证审查(SO#3) | 检查验证计划和过程的有效性和实现情况、是否完成了所有相关的软件配置管理和质量保证任务、是否软件需求、设计、代码和集成都进行了验证；是否软件验证过程达到了DO-178C关于覆盖的要求。 | 1. 开发阶段数据完整，经过了评审，并在配置管理中受控。 2. 测试用例和测试过程形成了文档，经过了评审，并在配置管理中受控。 3. 实施和执行了测试用例和过程。 4. 软件测试环境形成了文档，并受控。 | 软件需求、设计描述、源代码、目标代码、软件验证案例与过程、软件验证结果、软件生命周期环境配置项（含测试环境）、问题报告、软件配置管理记录、软件质量保证记录、软件工具认证数据。 |
| 最终审查(SO#4) | 检查最终产品是否满足DO-178B/C的目标要求；是否所有的开发、验证、质量保证、配置管理和认证联络活动完整执行；是否完成了符合性审查；检查最终的软件配置项SCI和软件项目总结报告SAS。 | 1. 软件符合性审查已完成，所有问题已解决。 2. SAS和SCI已完成且经过了评审。 3. 所有的软件生命周期数据都已完整、得到了批准，并在配置管理中受控。 | 软件验证结果、软件生命周期环境配置项、软件配置项、问题报告、软件配置管理记录、软件质量保证记录、软件项目总结报告SAS。 |

表1 符合性审查目标、准则和数据要求

## 2.2软件配置管理

### 2.2.1软件配置管理概念

配置：“配置”是在技术文档中明确说明并最终组成软件产品的功能或物理属性。“配置”包括了即将受控的所有产品特性,其内容及相关文档,软件版本, 变更文档,软件运行的支持数据,以及其他一切保证软件一致性的组成要素。

配置项：配置项是受配置管理控制和管理的基本单位,受控软件被划分为各类配置项,配置管理工作都是围绕配置项来进行。配置项是逻辑上组成软件系统的各组成部分。

基线：在配置管理系统中,基线就是一个配置项或一组配置项在其生命周期 的不同时间点上通过正式评审而进入正式受控的一种状态,而这个过程被称为“基线化”。基线是被评审过的一个或多个软件配置项。

软件配置管理（SCM——Software Configuration Mangement）是ISO9001和CMM Level2中的重要组成元素,它在软件产品开发的生命周期中,提供了结构 化、有序化、产品化的管理软件工程方法,是软件开发和维护的基础。软件配置管理是项目运作的一个支撑平台,它贯穿需求、分析、设计、实现、测试、等整个软件生存周期。

配置管理过程是对处于不断演化、完善过程中的软件产品的管理过程。其最 终目标是实现软件产品的完整性、一致性、可控性,使产品极大程度地与用户需 求相吻合。它通过控制、记录、追踪对软件的修改和每个修改生成的软件组成部 件来实现对软件产品的管理功能。

### 2.2.2软件配置管理关键活动

1）配置项识别

定义:Pressman对于SCI给出了一个比较简单的定义:软件过程的输出信息 可以分为三个主要类别 :

计算机程序（源代码和可执行程序）

描述计算机程序的文档（针对技术开发者和用户）

数据（包含在程序内部或外部）

这些项包含了所有在软件过程中产生的信息,总称为软件配置项。配置项的识别是配置管理活动的基础,也是制定配置管理计划的重要内容。

2）工作空间管理

在引入了软件配置管理工具之后,所有开发人员都会被要求把工作成果存放 到由软件配置管理工具所管理的配置库中去,或是直接工作在软件配置管理工具 提供的环境之下。所以为了让每个开发人员和各个开发团队能更好的分工合作, 同时又互不干扰,对工作空间的管理和维护也成为了软件配置管理的一个重要的 活动。

通常把整个配置库视为一个统一的工作空间,然后再根据需要把它划分为个 人（私有）、团队（集成）和全组（公共）这三类工作空间分支,从而更好 地支持将来可能出现的并行开发需求。

3）版本控制

版本控制是软件配置管理的核心功能。所有置于配置库中的元素都应自动予以版本的标识,并保证版本命名的唯一性。版本在生成过程中,自动依照设定的使用模型自动分支、演进。除了系统自动记录的版本信息以外,为了配合软件开发流程的各个阶段,还需要定义、收集一些元数据来记录版本的辅助信息和规范开发流程,并为今后对软件过程的度量做好准备。当然如果选用工具支持的话,这些辅助数据将能直接统计出过程数据,从而方便用户软件过程改进活动的进行。

4）变更控制

在对配置项的描述中,本文引入了基线的概念。从对于基线的定义中 可以发现,基线是和变更控制紧密相连的。也就是说在对各个做出了识别, 并且利用工具对它们进行了版本管理之后,如何保证它们在复杂多变的开发过程中真正处于受控状态,并在任何情况下都能迅速地恢复到任一历史状态就成为了

软件配置管理的另一重要任务。因此,变更控制就是通过结合人的规程和自动化工具,以提供一个变化控制的机制。

5）状态报告

配置状态报告就是根据配置项操作数据库中的记录来向管理者报告软件开 发活动的进展情况。这样的报告应该是定期进行,并尽量通过工具自动生 成,用数据库中的客观数据来真实地反映各配置项情况。 配置状态报告应根据报告应着重反映当前基线配置项的状态,以作为开发进 度报告的参照。同时也能从中根据开发人员对配置项的操作记录来对开发团队的 工作关系作一定分析。

6）配置审计

配置审计的主要作用是作为变更控制的补充手段,来确保某一变更需求己被 切实实现。在某些情况下,它被作为正式的技术复审的一部分,但当软件配置管理是一个正式的活动时,该活动由SQA(Software Quality Assurance)人员单独执行。

总之,软件配置管理的对象是软件研发活动中的全部开发资产。所有这一切 都应作为配置项纳入管理计划统一进行管理,从而能够保证及时对所有软件开发 资源进行维护和集成。

## 2.3符合适航标准的软件配置管理过程

### 2.3.1适航软件配置管理过程的目标：

a．每个配置项和它的后续版本都明确标示，因此，这是建立控制和配置项的参考依据。

b．基线被定义为进一步的软件生命周期过程的活动，并允许参考，控制以及在配置项中的可追踪。

c．问题报告进程记录记中包括不遵守软件计划和标准的进程，软件生命周期过程的产出记录不足的问题，记录反常行为的软件产品，以确保这些问题的解决。

d．变更控制提供记录，评估，决议，批准在整个软件生命周期的变化。

e．变更审查以确保对问题和变化进行评估，批准，或拒绝，变更的实施批准，并提供反馈意见问题报告，并在软件规划过程中定义变更控制方法，通过向受影响的进程。、

f．状态统计提供软件生命周期中配置的身份证明，基线，问题报告，变化控制等数据。

g．归档和检索，确保该软件的生命周期与软件产品相关的数据可以在需要的情况下，复制，再生，重新测试或修改软件产品检索。发布活动的目的是确保只有授权的软件使用，特别是软件制造，除了归档和检索。

h．软件负荷控制确保加载的可执行目标代码和参数数据项文件，如果有的话，采取适当的保障措施进入系统或设备。

i．软件生命周期环境控制确保用于生产的软件工具是经过标识的，可控制和可检索的。

### 2.3.2适航软件的配置管理过程

Do178c列举了22种数据项：

* 软件方面的验证计划
* 软件开发计划
* 软件验证计划
* 软件配置管理计划
* 软件质量保证计划
* 软件需求标准
* 软件设计标准
* 软件代码标准
* 软件需求数据
* 设计说明源代码
* 可执行目标代码
* 软件验证例子和程序
* 软件验证结果
* 软件生命周期环境配置索引
* 软件配置索引
* 问题报告
* 软件配置管理记录
* 软件质量保证（SQA）记录
* 软件完成综述
* 数据追踪
* 参数数据项

### 2.3.3数据控制类别

软件生命周期的数据可以被分配到两个配置管理控制类别之一：控制类别1（CC1）和控制类别2（CC2）。不同的软件级别中，相同的软件生命周期的数据所规定的控制类别是不同的。下表为CC1和CC2与配置管理进程有关活动的对应关系。

|  |  |  |
| --- | --- | --- |
| SCM进程活动 | CC1 | CC2 |
| 配置鉴定 | ● | ● |
| 基线 | ● |  |
| 可追溯性 | ● | ● |
| 问题报告 | ● |  |
| 变更控制-完整性和鉴定 | ● | ● |
| 变更控制-跟踪 | ● |  |
| 变更审查 | ● |  |
| 配置状态统计 | ● |  |
| 恢复 | ● | ● |
| 已批准变更的保护 | ● | ● |
| 媒体选择，刷新，复制 | ● |  |
| 释放 | ● |  |
| 数据保留 | ● | ● |

# 三、关键技术和方法

## 3.1软件过程建模

软件过程建模技术是实施软件过程管理和改进的一个重要环节,因此软件过程建模技术的研究工作在软件过程技术研究领域中处在至关重要的位置。软件过程建模的概念最早是由Leon Osterweil提出的。1987年,在第9届国际软件工程会议上,Leon Osterweil发表了著名的论文”Software processes are sofetware too”,从此软件过程建模领域的研究开始兴起,一系列软件过程建模的相关概念被确定下来,同时研究人员也开发了大量的软件过程建模方法。软件过程模型可以被用来精确地描述软件过程的组成和结构从而消除对于过程定义的不一致理解。很多实践表明,软件过程建模技术能够被有效地应用于描述和分析软件组织的软件过程,并有效地为发现过程改进的机会提供支持。

## 3.2软件过程建模的方法

### 主要考虑过程所涉及的各种成分的分类

这种分类是以过程所涉及的各种成分(如活动、角色、产品、资源和约束等)为出发点来考虑过程建模。主要有四种不同的建模方法。

#### 1.以活动为中心的建模方法

这种方法以过程中一类主要成分,即以活动为中心构造过程模型。具体方法是先确定这些活动以及它们之间的执行顺序,再收集与各个活动相关的其它数据,如活动涉及的产品、资源和约束等。

#### 2以角色为中心的建模方法

这种方法以过程中各活动所涉及的角色为中心建模。

#### 3.以产品为中心的建模方法

这种方法以过程中另一类主要成分,即以产品为中心建模。用这种方法建立的过程模型非常类似于E-R模型,主要考虑的是产品及产品之间的关系,如一个模块对另一个模块的调用关系,一个设计说明对一个需求分析说明的依赖关系等。

#### 4.基于过程模板的建模方法

先确定过程的各类成分,如活动、角色、产品等,然后把它们定义成模板。再针对每一个模板在不同的抽象层次上构造该模板的具体对象,最后定义同一抽象层次中所有对象之间的各种关系,从而构成一个完整的过程模型。

### 主要看软件过程模型侧重于描述功能目标还是侧重于描述活动行为。

分类时主要考虑过程模型是否支持软件过程的动态性,

#### 1.面向活动的过程建模方法

这种活动的特点是在建模时主要描述过程活动以及这些活动可能存在的、应该怎样被执行的顺序,并把这种排序作为这些活动的一个函热即在描述这些活动的同时还要描述它们之间的先后执行顺序。

#### 2.面向目标的建模方法

这种方法把过程活动以及活动的顺序看成是所述功能目标的一个函数。建模时主要描述过程活动必须满足什么样的功能目标。

#### 3.面向活动和面向目标的建模方法的结合。

面向目标的建模方法适合于处理过程活动的动态排序,而面向活动的方法更适合于按可能的顺序执行过程活动。事实上这两种方法是互补的,它们的长处将引导我们在过程定义时,把静态的确定过程活动顺序与动态地产生过程活动顺序结合起来。

### 所采用的不同形式化方法的分类。

#### 1.过程程序设计方法

这种建模方法的出发点是“软件过程也是软件”的论断。因为软件过程与软件产品具有广泛的类同性,对软件过程的描述亦是一种程序设计形式。这种方法把过程所涉及的软件对象用其所需工具与开发方法编程。它通过关系、触发器和谓词等机制对过程的功能、行为和对象进行详细、确定的算法描述。

#### 2.多功能分解方法

这种建模方法把一个软件过程用带有输入属性和输出属性的一个过程元素集来表示。即把一个过程定义为反映输入与输出关系的数学函数集。这些函数可以按照语法进一步进行层次分解,形成一个过程的多个子过程步。这种分解可以一直进行下去,直至产生的子过程步映射到一个外部工具或由人员操作实现。

#### 3.基于计划的建模方法

针对过程程序设计方法和功能分解方法中不能处理实际过程中的偶发事件的弱点，提出了基于计划的过程建模方法。这种方法基于人工智能技术,把软件过程用一组过程步集和一组约束集来表示。约束集用于指出怎样选择过程步以及按开么顺序执行过程步。

#### 4.基于Petri网的建模方法

当前的过程建模方法中,基于Petri网或其变种,进行过程建模的方法占有相当数量,因为Petri网具有很强的表达能力,能够有效地形式化描述软件过程的并发性和活动与产品之间的关系。而且这种图形表示易于理解和管理软件过程

#### 5.基于规则的建模方法

基于规则的建模方法和语言是人们普遍看好的一种过程建模技术。这种方法提供了活动的动态链接机制,从而很自然地描述了过程的不可预见性,也为人们控制过程提供了最为灵活的手段。

#### 6.基于逻辑的建模方法

该方法基子某种逻辑语言来描述过程模型。

#### 7.基于知识的建模方法

基于知识的建模方法提供了对过程模型的增量式形式说明能力和可重用能力。这种方法把过程知识(例如过程活动、过程实施者,产品对象、工具以及它们之间关系等)用面向对象方法抽象成各个不同的类,存子知识库中。过程建模时,根据需要查询知识库,`从中获取有关过程活动及其它对象的抽象描述,选取或构造所需的过程模型。并对过程模型进行分析和推理,最后生成过程实例及相应的活动计划。

#### 8.混合风格的建模方法

所谓混合风格是指建模方法中融合了多种形式化技术和手段。

**主要参考文献**

1. 民航局令第207号, 关于修订《航空发动机适航规定》的说明
2. 民航局令第207号, 航空发动机适航规定 尉询楷，杨立，刘芳，战立光，冯悦.
3. AC 20-115C. AirBorne Software Assurance,dated July 19,2013.
4. AB 20-115B. RTCA,Inc,Document RTCA/DO-178B, dated January 11, 1993.
5. 沈小明等. 机载软件研制流程最佳实践. 上海交通大学出版社，2013
6. SAE ARP4754A, Guidelines for Development of Civil Aircraft and Systems, 2010.12.
7. SAE ARP4761, Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment, 1996.12.
8. RTCA DO-178C, Software Considerations in Airborne Systems and Equipment Certification, dated December 13, 2011.
9. RTCA DO-330, Software Tool Qualification Considerations, dated December 13,2011.
10. RTCA DO-331, Model-Based Development and Verification Supplement to DO-178C and DO-278A, dated December 13, 2011.
11. RTCA DO-332, Object-Oriented Technology and Related Techniques Supplement to DO-178C and DO-278A, dated December 13, 2011.
12. RTCA DO-333, Formal Methods Supplement to DO-178C and DO-278A, dated December 13, 2011. Tong M,Thurman D R,Guynn M D. Conceptual design study of an advanced technology open-rotor propulsion system[R]. ISABE-2011-1311,2011.
13. SEI. CMMI for Development, Version 1.2-Improving Processes for Better Products. SEI, CMU, 2006.
14. Kitchenham BA, Mendes E, Travassos GH. Cross versus within-company cost estimation studies: A systematic review. IEEE Trans. on Software Engineering, 2007,33(5):316−329.
15. Jorgensen M, Shepperd M. A systematic review of software development cost estimation studies. IEEE Trans. on Software Engineering, 2007,33(1):33−53.
16. 王长元，赵莉，王淑蓉.软件工程与建模[M].西安交通大学出版署，2010.
17. 李明树，杨秋松，翟健.软件过程建模方法研究[J].软件学报，2009
18. 范玉顺.面向对象的Petri网方法及其在软件工程中的应用[J].计算机应用，1998
19. 乐晓波，汪琳.面向对象的Petri网建模技术的研究[J].计算机工程，2002
20. Lee S, Shim J, Wu C. A meta model approach using UML for task assignment policy in software process. In: Proc. of the 9th Asia-Pacific Software Engineering Conf. on APSEC 2002. IEEE Computer Society, 2002. 376−382.、
21. Bendraou R, Combemale B, Cregut X, Gervais MP. Definition of an Executable SPEM 2.0. In: Proc. of the 14th Asia-Pacific Software Engineering Conf. on APSEC 2007. IEEE Computer Society, 2007. 390−397.
22. Park S, Choi K, Yoon K, Bae DH. Deriving software process simulation model from SPEM-based software process model. In: Choi K, ed. Proc. of the Asia-Pacific Software Engineering Conf. on APSEC 2007. 2007. 382−389.
23. 陈迎欣.小组软件过程的仿真建模方法的研究[D],哈尔滨工程大学博士论文，2006
24. J.Peterson.Petri net theory and the modeling of systems[M].中国矿业大学出版社
25. 张绍阳.基于性能分析的一种OOPN关联矩阵建立方法[J].西北大学学报.
26. Atkinson DC, Weeks DC, Noll J. The design of evolutionary process modeling languages. In: Weeks DC, ed. Proc. of the Software Engineering Conf. on 2004 11th Asia-Pacific. Washington: IEEE Computer Society, 2004. 73−82.
27. Franch X, Ribó JM. A structured approach to software process modelling. In: Proc. of the 24th Conf. on EUROMICRO. IEEE Computer Society, 1998. 753−762.
28. Bhuta J, Boehm BW, Meyers S. Process elements: Components of software process architectures. In: Proc. of the Int’l Software Process Workshop (SPW 2005). LNCS 3840, Springer-Verlag, 2005. 332−346.
29. Min SY, Lee HD, Bae DH. Soft PM: A software process management system reconciling formalism with easiness. Information and Software Technology, 2000,42(1):1−16.
30. Henderson-Sellers B, Gonzalez-Perez C. A comparison of four process metamodels and the creation of a new generic standard. Information and Software Technology, 2005,47(1):49−65.
31. 李思广，林子禹，胡 峰，潘小劲.基于UML的软件过程建模方法研究，计算机工程与应用，2003
32. [张璇](http://epub.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CJFD&sfield=%e4%bd%9c%e8%80%85&skey=%e5%bc%a0%e7%92%87&scode=21832615%3b09324537%3b09337460%3b17304744%3b24411866%3b27362069%3b); [李彤](http://epub.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CJFD&sfield=%e4%bd%9c%e8%80%85&skey=%e6%9d%8e%e5%bd%a4&scode=21832615%3b09324537%3b09337460%3b17304744%3b24411866%3b27362069%3b); [王旭](http://epub.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CJFD&sfield=%e4%bd%9c%e8%80%85&skey=%e7%8e%8b%e6%97%ad&scode=21832615%3b09324537%3b09337460%3b17304744%3b24411866%3b27362069%3b); [代飞](http://epub.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CJFD&sfield=%e4%bd%9c%e8%80%85&skey=%e4%bb%a3%e9%a3%9e&scode=21832615%3b09324537%3b09337460%3b17304744%3b24411866%3b27362069%3b); [谢仲文](http://epub.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CJFD&sfield=%e4%bd%9c%e8%80%85&skey=%e8%b0%a2%e4%bb%b2%e6%96%87&scode=21832615%3b09324537%3b09337460%3b17304744%3b24411866%3b27362069%3b); [于倩](http://epub.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CJFD&sfield=%e4%bd%9c%e8%80%85&skey=%e4%ba%8e%e5%80%a9&scode=21832615%3b09324537%3b09337460%3b17304744%3b24411866%3b27362069%3b)，[面向软件非功能需求的软件过程建模方法](http://epub.cnki.net/kns/detail/detail.aspx?QueryID=4&CurRec=12&FileName=JFYZ201607018&DbName=CJFDLAST2016&DbCode=CJFQ&pr=)[M] 计算机研究与发展