基于 DO-178B 的民用飞机机载软件工具的鉴定和应用

孙全艳[1,2]　赵京洲[2]　章晓春[2]
(1. 上海交通大学自动化系,上海 200030;
2. 上海飞机设计研究院飞控系统设计研究部,上海 200436)

Research and Application on Civil Airborne Software Tool Qualification based on DO-178B

Sun Quanyan[1,2]　Zhao Jingzhou[2]　Zhang Xiaochun[2]
(1. Department of Automation, Shanghai Jiao Tong University, Shanghai 200030;
2. Flight Control Department, Shanghai Aircraft Design and Research Institute, Shanghai 200436)

摘要:随着民用机载系统复杂性的不断提高,软件工具在机载软件系统开发中被越来越广泛地使用。从适航合格审定角度探讨了软件工具在民用飞机领域的鉴定工作,包括工具鉴定准则、工具鉴定数据等。

关键词:机载软件;工具鉴定;DO-178B;民机合格审定

【Abstract】 With the increasing complexity of the airborne software systems on the civil aircraft, more software tools have been widely used in airborne software development process. Based on the certification requirements, this paper discusses the software tool qualification, including tool qualification criteria, tool qualification data, etc.

【Keywords】 Airborne Software; Tool Qualification; DO-178B; Civil Aircraft Certification

0　引言

民用航空电子系统的机载软件在提高开发效率和保证安全性这两方面有着非常严格的要求。为此,国际航空无线电委员会(RTCA)针对民用航空电子系统的软件开发制定了 RTCA/DO-178B 标准(以下简称 DO-178B)。这一标准基于在系统安全性评估过程中确定的软件对潜在失效条件的影响将软件划分为 A、B、C、D、E 五个等级,软件等级越高所付出的成本越高。如:发现一个较小程序缺陷的代价为 10 万美元至 50 万美元;发现一个较大程序缺陷的代价为 100 万美元至 5 000 万美元[1]。随着机载软件代码总量和复杂性的逐年增加,软件开发商面临着巨大的挑战。为了能在开发成本和项目进度的限制下完成开发项目,软件开发商不断改进其开发流程,寻求有效的方法使用软件工具。

一套成熟的软件工具可以减少工作量、降低开发成本和减少人为引入的错误,替软件开发商节省足够的资源。软件工具与机载软件本身相同,都是由软件表达的复杂逻辑组成,开发过程和环境都非常类似。现行的 DO-178B 标准提出使用类似于软件本体的基于流程的研制保证方法,对软件工具进行鉴定。FAA Order8110.49 对工具鉴定提出了目标和相应的操作方法,但在实际的软件工程中,工具类型多种多样,涉及到机载软件开发全生命周期的各个阶段,不同类型的软件工具其安全性影响的重点也不同。

1　工具鉴定概述

1.1　工具的定义和分类

DO-178B 标准中将机载软件工具分为软件开发工具和软件验证工具[2]。

软件工具(Software Tool):一个计算机程序,用来协助开发、测试、分析、生成或修改另一个计算机程序或文档等。软件工具引入的最直接的目的是减少工作量、降低开发成本和减少人为引入的错误。同时,软件工具的引入也会带来最直接的不利影响,首先,工具的研制有时对于机载软件项目来说是独立的;其次,工具自身的错误可能通过工具的重复使用而被放大。

软件开发工具(Software Development Tool):此类工具的输出是机载软件的一部分,因此开发工具可能会引入错误。软件开发工具通常被用来创建或更改生命周期资料。一个开发工具能创建新的需求、设计、代码或其它数据,或者修改其含义。如果这个工具能在最终交付的产品中引入错误,那么该工具就被归类为开发工具。例如,某个工具可以从规则化的软件低级别需求中直接生成源代码,如果该源代码是机载软件产品的一部分,则该工具可能会引入与安全性相关的错误,那么这个工具就属于软件开发工具。

软件验证工具(Software Verification Tool):此类工具并不能引入错误,但也可能检测不到存在的

错误。软件验证工具通常被用来协助评估一个软件生命周期数据项对于一些标准,如一致性、兼容性、结构覆盖等的正确满足。验证工具无法创建新的需求、设计、代码或其它数据,也不能修改其基本含义。如果一个工具无法检测出存在的错误,同时也不会引入错误,那么该工具就被归类为验证工具。例如,一个静态语法分析器,用来分析源代码是否满足软件编码规范的要求,此时,该工具并不会引入错误,但是也可能检测不到源码中的错误,那么这个静态语法分析器就是软件验证工具。

1.2 工具鉴定的目的

DO-178B 第 12.2 节中提到:“当本文件中描述的软件研制流程通过软件工具的使用而省略、减少或自动进行,且工具输出未能按第 6 章的要求(或等效方式)进行完整验证时,要求对相应的软件工具进行鉴定。”且“工具鉴定的目标工具应提供至少与其省略、减少或自动进行的流程相当的置信度。”

对于机载软件开发商来说,通过工具鉴定可以省略对工具输出的验证。例如,如果一个代码生成器从规则化的低级别需求直接生成源代码,而这个代码生成器通过了工具鉴定,那么使用该工具生成的源代码与对应的低级别需求之间的相关验证工作都可以省略,从而大大减少了工作量。

2 工具鉴定方法

2.1 工具鉴定需求

并不是所有的软件工具都需要进行工具鉴定,对软件工具的分类并不能决定该工具是否需要进行鉴定。如果某个工具的输出是向适航审查方提供证据来表明 DO-178B 目标或该目标一部分的符合性的唯一方法,那么这个工具就需要被鉴定。换言之,如果工具的输出经过人工评审或通过一些已鉴定工具进行验证,那么就需要这些额外的证据(人工评审的结果或已鉴定工具的验证结果)以及工具的输出一起表明对目标的符合性。

可以通过评估以下三个问题来决定工具是否需要进行鉴定:

(1)工具是否在机载软件中引入错误或无法检测出存在的错误。

(2)工具的输出是否未通过其它验证活动的验证而直接使用。

(3)工具的输出是否被用来表明目标的符合性或等效代替该目标。

如果上述三个问题的回答均为“是”,那么该工具必须进行鉴定(如图 1 所示)。

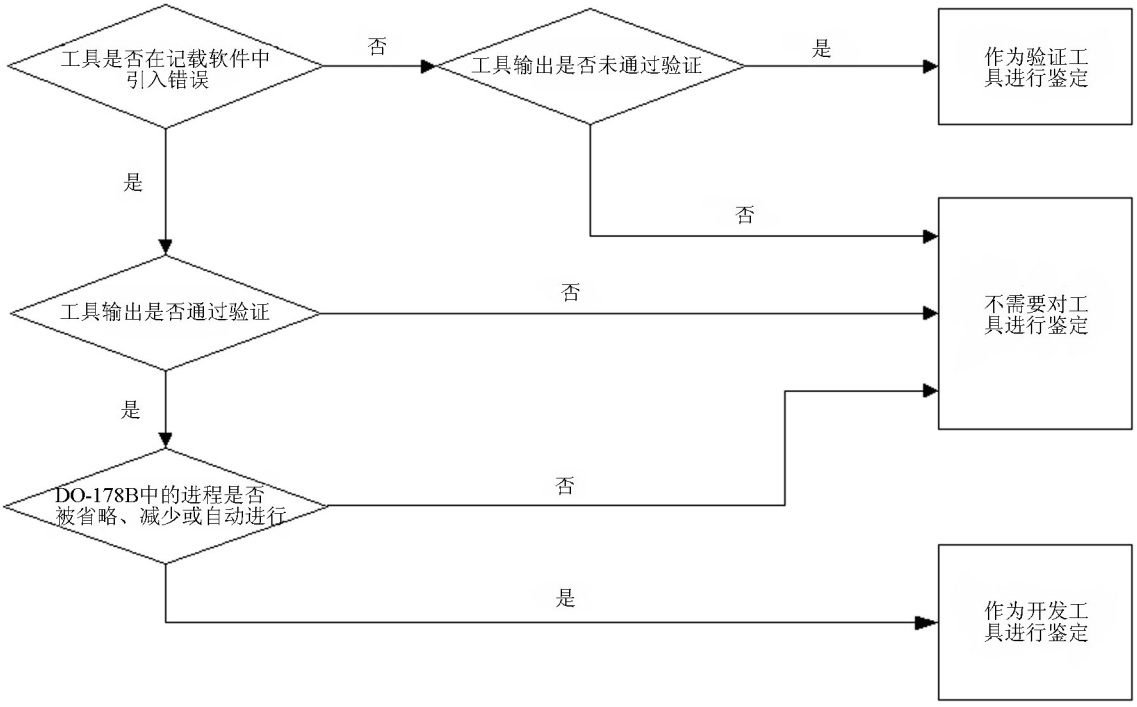

图 1 工具鉴定需求

机载软件所使用的可能需要进行鉴定的开发工具包括:自动生成源代码的设计工具(代码生成器)、生成在模拟器上运行的代码的工具、二进制译码器(如交叉编译器和格式转换器)等。机载软件使用的可能需要进行鉴定的验证工具包括:自动进行代码评审或设计评审的工具、从需求生成测试用例/程序的工具、判断通过/未通过状态的工具、追踪和报告结构覆盖验证结果的工具、确定需求覆盖验证结果的工具等。文档编辑、需求管理、构型管理、质量管理中使用到的工具通常不需要工具鉴定。

代码库和实时操作系统不归为工具的范畴,因为这二者是实际执行机载软件功能的一部分。通常由设计保证过程来验证,不需要进行工具鉴定。

2.2 工具鉴定准则

开发工具的鉴定准则比验证工具的鉴定准则严格。通常而言,开发工具鉴定的工作量与 A 级别机载软件的开发工作量相当,验证工具鉴定的工作量与 D 级别机载软件的开发工作量相当。在进行开发工具的鉴定时,必需考虑 14 条工具鉴定准则,而对验证工具的鉴定仅需考虑其中的 7 条准则[3]。

开发工具和验证工具都需要满足的鉴定准则主要有以下几条:

(1)仅当工具输出具备确定性时需要被鉴定;

(2)工具鉴定结果只对使用该工具的系统有效;

(3)功能上未进行分区的组合工具应按照一个工具来进行鉴定;

(4)工具鉴定活动也应遵循构型管理和质量保证相关目标;

(5)确立工具操作需求的符合性方法;

(6)工具操作需求(TOR)应被评审;

(7)展示符合工具操作需求的正常工作状态。

开发工具需要额外满足的鉴定准则主要有以下几条:

(1)工具的鉴定与使用工具的机载软件相一致,目标均需被满足;

(2)展示符合工具操作需求(TOR)的非正常工作状态;

(3)执行基于需求的覆盖分析;

(4)执行对应工具软件等级的结构覆盖分析;

(5)完成对应工具软件等级的鲁棒性测试;

(6)分析潜在的错误;

(7)工具的等级可能会降低。

在对软件工具进行鉴定时,需要采用富有条理的方法来进行相关工作。最佳的方法是使用表格(见表 1),该表格通常包括以下内容[4]:

(1)工具标识(Tool ID):工具的完整标识。这一栏内应包括工具的版本,这样可以对项目中的任何更改进行评估分析。在提交适航合格审定计划时,这一栏可以留成“待定”,在了解工具的细节后再加以判断和分析。

(2)工具名称(Tool Name):表明工具作用的名称。

(3)使用(How Used):描述在生命周期内如何使用工具。开发人员应能通过这一栏的描述判断工具的类型以及该工具是否需要进行工具鉴定。

(4)分类评估(Categorization Assessment):工具的类型。

(5)分类判断(Categorization Justification):应该包含以下内容:使用工具可省略哪些目标、工具结果可用来表明哪些目标的符合性、需要哪些额外的证据或数据与工具输出结合使用、对工具输出进行的验证活动等。这一栏的内容应和“使用”一栏的内容相结合,为工具的分类提供充分的依据。

(6)鉴定评估(Qualification Assessment):工具是否需要进行鉴定(是或否)。

(7)鉴定理由(Qualification Justification):按照本文 2.1 节的评估方法表述工具的鉴定理由。这一栏应列出使用工具可省略的目标。在很多情况下,目标之间是关联的;因此,满足一个目标可能就不必再提供其相关联目标的符合性证据。

在软件开发商的公司内部,大部分软件工具往往是在很多项目上沿用的。因此,工具鉴定数据可以重复使用并能够减少将来取证的工作量。通常的做法是将这类分析作为一份单独的文档,同时将某个软件开发项目中使用的所有工具按表格形式进行信息收集并在同一份文档中进行汇总。

表 1　工具标识和鉴定分析文档(样例)

工具名称	使用	工具标识/版本	分类评估		鉴定评估	
			验证工具/开发工具	分类判断	是/否	鉴定理由
低级别鲁棒性测试用例生成器	检查低级别需求,生成鲁棒性测试用例及其相关的测试程序,这些测试程序将在目标环境中执行	TBD-1	验证工具	工具的输出不会修改生命周期数据。工具仅有可能无法生成所需的鲁棒性测试用例和程序或生成一个错误的测试用例或程序	是	当生成的测试用例成功运行时,可满足 DO-178B 的附录 A6-3 的目标。工具的输出和对运行结果正确性的评估可为 A6-3 目标提供符合性证据。工具得到鉴定后,可保证工具生成的测试程序的正确性,则无需再对 A7-1 目标提供符合性证据
注:TBD-1,本工具仍在购买过程中。购得工具后,将更新"工具标识"一栏。同时,将重新对工具进行评估以确保之前的分析和结论仍然适用。						

3　工具鉴定数据

需要提交和仅供评审的软件工具生命周期数据的格式和封装因工具类型而不同。相比较来说,验证工具进行工具鉴定时需要的生命周期数据较少,而开发工具则相应较多。开发工具鉴定数据的设计保障等级应与项目机载软件的等级一致,应按照 DO-178B 标准中构型控制类型(CC1 或 CC2)的要求进行构型管理。对验证工具来说,鉴定数据可以都是 CC2 类的。开发工具和验证工具需要的生命周期数据见表 2。

表 2　开发工具和验证工具需要的生命周期数据

生命周期数据	适用的工具类型	是否提交
软件适航合格审定计划(项目级)PSAC	开发工具 & 验证工具	提交
工具操作需求 TQR	开发工具 & 验证工具	供评审
软件完结综述(项目级)SAS	开发工具 & 验证工具	提交
工具验证记录(测试用例、程序和结果)TVR	开发工具 & 验证工具	供评审
工具鉴定计划 TQP	开发工具	提交
工具完结综述 TAS	开发工具	提交
工具鉴定开发阶段数据(需求、设计和代码)	开发工具	供评审

3.1　工具鉴定计划(Tool Operational Plan-TOP)

工具鉴定计划可以视为工具的适航合格审定计划。工具鉴定计划是申请人和适航审查方进行充分沟通后制定的。开发工具必须有工具鉴定计划;对于验证工具,申请人需在使用该工具的软件适航合格审定计划中有所描述,似乎工作量较小,但是对于多次复用的验证工具来说,这并不是最有效的方法。因此,推荐为验证工具仍需编制一份单独的工具鉴定计划,这样可以使项目取证更为顺利,也方便工具在其它项目上的重复使用,从而克服了"工具鉴定结果仅适用于指定系统"的限制。DO-178B 标准的 12.2.3.1 节描述了该文档中应当包含的内容。

工具鉴定计划是针对软件工具如何开发和验证制定的计划文件,与按 DO-178B 要求开发的机载软件的适航合格审定计划类似。

3.2　工具操作需求(Tool Operational Requirements-TOR)

工具操作需求是工具鉴定生命周期数据中最重要的文档,无论是开发工具还是验证工具都必须具备工具操作需求。该文档中需要明确描述工具将完全代替或部分代替人工劳动完成机载软件研制过程中的哪些活动。DO-178B 标准的 12.2.3.2 节描述了该文档中应当包含的内容。

工具操作需求编制的困难之一是区分用户手册和功能。在大多数情况下,用户手册和功能是相同的。在某些情况下,用户手册包括的信息是使用该工具功能时的具体程序。无论如何,对于工具可提供的功能都应有一份清晰的描述以达到可验证的目的。

在通常情况下,软件工具仅可在某一特定操作系统下的指定平台上使用。这类信息应包括在工具的操作环境中。

虽然 DO-178B 第 12.2.3.2 节中未提出要求,但一种常见的做法是将工具的使用限制包括在这份文档内。例如,某工具仅能够处理不超过 250 个变量定义。这类限制在开发工具的验证中是至关重

要的。

开发工具的工具操作需求文档还有额外的要求。与过程相关的活动应包括在内,异常操作状态下的修正措施也应当被验证。

3.3 工具完结综述(Tool Accomplishment Summary-TAS)

工具完结综述可以视为工具相关合格审定活动的总结,该文件与工具鉴定计划相对应。开发工具必需有工具完结综述;对于验证工具,申请人需在使用工具的软件完结综述中有所描述。考虑到工具在多个项目上的重复使用,同样推荐为验证工具需编制一份单独的工具完结综述。DO-178B 标准中虽未明确描述该文档应包含的内容,但通常的做法是以第 11.20 节中"软件完结综述"的要求作为指南。

为了更好地衡量符合性,工具完结综述应有一句总结性的语句,即:该工具满足 DO-178B 的 12.2 节中描述的开发/验证工具的所有要求以及已批准的工具鉴定计划中所述的需求。

3.4 工具鉴定数据复用

工具的鉴定必须作为项目机载软件审定的一部分,不能脱离项目单独对工具进行鉴定。但在相似的项目环境中,以往鉴定过的工具根据具体情况,其鉴定数据可提供一定程度的复用。通常有三种情况[5]:

(1)工具本身及其使用环境和方法都未发生改变;

(2)仅工具的使用环境或方法发生改变;

(3)工具本身发生了改变。

对于第一种情况,工具鉴定数据的复用需要注意以下方面:工具是否在申请人的项目中使用并获得批准;工具鉴定的等级是否等于或者低于以往项目中鉴定的等级;工具生命周期数据自上次鉴定后是否发生改变;工具的使用环境是否与上次鉴定时等价;工具的操作需求是否与上次工具鉴定时一致;申请人能否获得上次鉴定的数据;申请人能否证明当前使用的工具与以往鉴定使用的工具是相同版本。申请人需要在项目的软件适航合格审定计划或工具鉴定计划中说明;同时,申请人还应提交自从上次鉴定至今在工具的使用过程中发现的所有问题报告(包括已经解决的问题)。

对于第二种情况,申请人需要在项目的软件适航合格审定计划或工具的鉴定计划中说明变化情况,并且分析变化的影响范围,其主要包括:确定工具验证环境需要进行的相应调整,和重新进行验证的范围;工具使用环境的详细情况在工具操作需求中说明;工具按照工具操作环境要求正确安装。

对于第三种情况,申请人需要进行更大范围的影响分析,至少包括:工具操作需求、工具软件需求、工具的设计、工具代码、工具开发环境和流程、需重新进行验证的环节。

4 商用成品软件工具

机载软件开发生命周期流程的各个阶段(需求、规范、设计、实现)和"正确性"流程的各个阶段(质量保证、构型管理、验证、适航联络)都会使用到商用成品软件工具[6],如:与需求管理相关的 Reqtify、DOORS、RequsitPro;与分析相关的 RT-Builder、PolySpace、RapidRMA;与设计相关的 Rhapsody、SCADE、Matlab;与测试相关的 VectorCast、Code Test、LDRA、RTRT;与实现相关的编译器/连接器、Tornado;与构型管理相关的 PVCS、Dimensions、ClearCase 等等。

以 SCADE 为例:ESTEREL 公司的 Safety-Critical Application Development Environment(简称 SCADE)是一个高安全性应用开发环境,覆盖了嵌入式开发的整个流程,能够节约 50% 以上的开发成本和时间,提高开发进程的自动化程度。DO-178B 标准规定了 40 个对软件开发过程的验证进程目标,使用 SCADE 可以完全省略其中 21 个,13 个目标的工作量也可因 SCADE 的使用而减少。

SCADE 是由一系列具有特定功能但相互之间又相对独立的模块构成的,目前通过 DO-178B 工具鉴定的模块有:代码生成器 KCG(开发工具)和覆盖率分析器 MTC(验证工具)。作为开发工具,KCG 通过工具鉴定需要有大量的文档作技术支持[7],见表 3。KCG 6.0.1 版在两个操作平台上完成了工具鉴定,即 Windows XP SP2 和 Solaris10,这意味着在进行工具鉴定时,这两个操作平台在 KCG 正常工作时被调用到的库函数已得到了充分的分析。开发 KCG 6.0.1 版时使用的编程语言是 C 和 ML,在工具鉴定计划中应充分表明使用这两种语言是"合适"的,语言的特性不会影响验证活动。

KCG 从最初的 3.1 版开始,各版本都在多个机型的机载软件开发上得到了应用,因此也有一套成熟的工具鉴定包供申请人使用。适航当局对机载软件进行适航合格审定时,不仅要审核工具本身,还会关注工具的使用环境和方法。因此,KCG 的工具鉴定包只能作为工具鉴定数据的主体,不能将两者等同。

表 3　KCG 工具鉴定数据包(摘录)

文件名称	文件标识
计划和标准	
KCG6.0 工具鉴定计划	KCG-PL-006
KCG6.0 软件开发计划	KCG-PL-007
KCG6.0 软件验证计划	KCG-PL-008
KCG6.0 软件构型管理计划	KCG-PL-010
KCG6.0 软件质量保证计划	KCG-PL-011
软件需求标准	KCG-ST-004
软件设计标准	KCG-ST-005
软件编码标准	KCG-ST-006
软件测试标准	KCG-ST-007
需求	
SCADE 语言参考手册	KCG-SRS-007
KCG 工具需求数据	KCG-SRS-008
设计	
KCG 架构设计文档	KCG-DD-004
软件库和运行时间设计文档	KCG-DD-008
低级别软件需求到高级别软件需求的架构追溯性	KCG-MTX-006
高级别软件需求分配到软件组件的矩阵	KCG-MTX-007
低级别软件需求到高级别软件需求的详细设计追溯性	KCG-MTX-009

5　国内经验和国外发展趋势

在民用飞机机载软件的适航合格审定方面,国内尚无机型经历过一次完整的 DO-178B 标准规定的软件生命周期流程审查,在工具鉴定方面就更无经验可循。

DO-178B 标准对所有的开发工具的鉴定仅笼统地要求满足软件开发流程的所有目标;而验证工具的鉴定仅说明要求达到工具操作需求。目标和要求都不尽明确,导致申请人和适航审查方执行工具鉴定的具体审定工作难以进行。因此,DO-178C 标准在修订时专门设立了工具鉴定工作组来研究相关对策。修订的 DO-178C 标准按照以下三条准则将工具鉴定的软件等级划分为 5 级(见表 4):

(1)准则 1,工具的输出是软件产品的一部分,并且有可能会向软件产品中引入错误;

(2)准则 2,工具使验证流程自动化,并有可能漏检错误;其输出用以证明验证流程中没有被自动化的部分的省略以及可能影响最终软件产品的开发流程步骤的省略;

(3)准则 3,在其预计的使用范围内可能会漏检错误。

同时,修订的 DO-178C 标准有一份关于工具鉴定的补充材料,其中详细规定了工具的研制流程和工具鉴定各等级需满足的目标。

表 4　工具鉴定的软件等级

软件等级	准则		
	1	2	3
A	TQL-1	TQL-4	TQL-5
B	TQL-2	TQL-4	TQL-5
C	TQL-3	TQL-5	TQL-5
D	TQL-4	TQL-5	TQL-5

6　结论

工具鉴定是一个不可忽视的过程,必须遵循现行 DO-178B 标准的 12.2 节及 ORDER 8110.49 第 9 章的要求。从 DO-178C 标准的修订来看,适航审查方对工具鉴定的要求正趋于规范并将更加严格。对于审查方来说,很多工具按可执行项目要求的功能,仅仅需要进行最少的鉴定工作。但对申请人来说,工具的成本、鉴定工作量以及将来对该工具的复用都必须被考虑到。因此,确定工具的鉴定状态并按型号合格证申请时现行有效的标准完成对工具的鉴定,对作为申请人的民用飞机制造商来说将会是一个巨大的挑战。

参考文献:

[1]Jean-Louis Camus, Bernard Dion. Efficient Development of Airborne Software with SCADE Suit [J]. Esterel Technologies, 2003.

[2]DO-178B, Software Considerations in Airborne Systems and Equipment Certification[S]. RTCA/EUROCAE, 1992.

[3]Order 8110.49, Software Approval Guidelines[S]. U. S. Deparment of Transprotation Federal Aviation Administration. June 3, 2003.

[4]Mike Dewalt. DO-178B Seminar[EB/OL]. Copyright CSI Inc. 2009.

[5]蔡喁. 机载软件审定中工具的鉴定问题[EB/OL], 上海适航合格审定中心. 2009, 7.

[6]Charles Soderstrom. HighRely Whitepaper-Tool Qualification Overview[EB/OL]. Copyright HighRely Inc. 2005-2007.

[7]王云明. KCG 6.0.1 Qualification[EB/OL]. Copyright Esterel Technologies, 2009, 12.