

# GJB5000A

## GJB5000A 简介

关键词：GJB5000A 资料 军用软件标准 GJB5000A 软件

### 一、软件成熟度模型是什么

软件成熟度模型的核心思想是，把软件开发视为一个过程，并根据这一原则对软件开发和维护进行过程监控和研究，以使其更加科学化、标准化、使企业能够更好地实现商业目标。

软件过程成熟度概念的引入，是为了解决路径的问题，是指一个特定软件过程得到清晰的定义、管理、测量、控制和有效的程度。

成熟度概念蕴含的意义是组织能力提高是需要一个演化的进程，有一个从不成熟到相对成熟的过程。通过软件过程评估，可以帮助企业认识所处的位置，通过软件过程模型，可以帮助企业找到前进的目标。

### 二、GJB5000A 是什么

GJB5000A 是一个产品开发模型（Product Development Model, PDM），关注整个体系的问题，是一个过程改进参考模型，描述的是一组有效过程的特征，提供了一套最佳实践，它关注的是：生产率（Productivity）、性能（Performance）、成本（Costs）、相关方满意（Stakeholder satisfaction）。

GJB5000A 是一个产品集，它包括：



- 军用软件能力成熟度模型框架
- 集成模型
- 评估方法和材料
- 各种培训
- 术语

### 三、GJB5000A 军用软件能力成熟度模型框架

军用软件能力成熟度模型框架：

- 由 5 个成熟度等级来表达：每个成熟度等级由若干过程域组成；
- 每个过程域由目标、执行方法组成。

即，成熟度等级中包含关键的过程域，每个过程域中具有一定的目标，以及为了达到这些目标必须要做到的行动步骤，即最佳实践。

### 四、GJB5000A 告诉我们什么

GJB5000A 告诉我们，过程管理方面优秀的软件组织是什么样的，优秀的软件组织也要分等级（1-5 级），每个级别的软件组织都具备有一定的特征，即都执行了某些特殊活动。

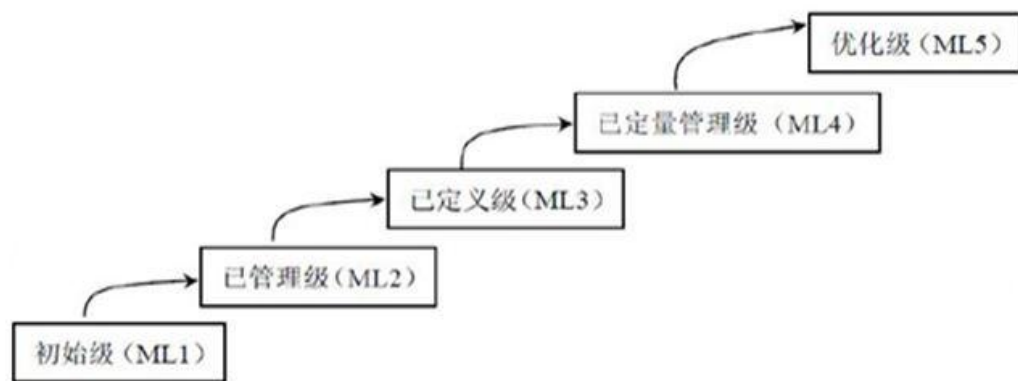
GJB5000A 是一个最佳实践的集合，不一定全部适用自己的组织，但总有一些好的做法可以借鉴。

### 五、GJB5000A 的表示方法



GJB5000A 的表示为五个成熟度级别：

- 基于组织的成熟度
- 每一级别是后续级别的基础
- 22 个过程域（PA）分成 5 个级别
- 过程改进逐级进行
- 表明一个组织的成熟度级别
- 反应了过程改进的顺序



## 六、GJB5000A 各成熟度等级所含过程域

过程域类别	ML2	ML3	ML4	ML5
过程管理类		组织过程定义( OPD )	组织过程绩效( OPP )	组织创新和部署( OID )
		组织过程焦点( OPF )		
		组织培训( OT )		
项目管理类	项目监控( PMC )	集成项目管理( IPM )	定量项目管理( QPM )	
	项目策划( PP )	风险管理( RskM )		
	供方协议管理( SAM )			
工程类	需求管理( ReqM )	产品集成( PI )		
		需求开发( RD )		
		技术解决方案( TS )		
		确认( Val )		
		验证( Ver )		
支持类	配置管理( CM )	决策分析和决定( DAR )		原因分析和决定( CAR )
	过程和产品质量保证( PPQA )			
	测量与分析( MA )			

## 体系框架

GJB5000A 是军用软件研制范围内用于衡量软件过程能力的标准，它通过“过程域和实践”对软件过程能力提出了要求，从研制过程的战略层面到战术层面都明确了标准，二级、三级的要求分布在 18 个过程域，有 76 个目标（特性目标：40，共性目标 36），352 个实践（特性实践：136，共性实践：216），它把项目管理过程、工程过程、支持过程和组织过程全部分解成一个一个动作要求；但是，我们的科研生产过程实际上是一系列连贯动作的组合，有并行有交叉，所以不能把 GJB5000A 直接作为企业的软件过程规范，这是制定体系文件时要考虑的难点问题。

因此，在进行体系架构设计时，必须结合 GJB5000A 标准的要求，对各过程域进行剖析，进行必要的合并与整合，并结合企业的实际进行本地化和简化，形成符合实际的体系架构。

参考体系结构如下：

序号	过程	合并的过程和实践
1	项目管理过程	PP 项目策划
		PMC 项目监控
		IPM 集成项目管理
		RskM 风险管理
2	外包外协过程	SAM 供方协议管理
3	需求管理过程	ReqM 需求管理
4	分析、设计过程	RD 需求开发
		TS 技术解决方案
5	实现过程	PI 产品集成
6	验证与确认过程	VER 验证过程
		VAL 确认过程
7	决策分析过程	DAR 决策分析
		TS SG1 选择产品构件的解决方案
8	培训过程	各过程域的 GP2.5
		OT 组织培训
9	组织过程改进过程	OPD 组织过程定义
		OPF 组织过程焦点
10	配置管理过程	CM 配置管理
11	质量保证过程	PPQA 过程与产品质量保证
12	测量分析过程	MA 测量与分析



基于 GJB5000A 软件体系的文件结构包括四个层次，如下图所示：



## 成熟度等级的理解

《GJB5000A 军用软件研制能力成熟度模型》规定了五个成熟度等级，标准本身给出了这些等级之间的区别和联系的详细说明，许多读者也都有自己独特的理解和解释。下面仅仅以“评审过程”为例，说明五个等级的关注点。





等级	关注（以评审为例）	说明
1、初始	---	---
2、已管理	执行与否	按照组织要求和项目策划的安排开展了相应评审，并留下证据
3、已定义	执行细节	在如何执行的问题上，组织给出了严格的准则（裁剪准则），以及更加细致的要求，包括目的、输入、入口准则、活动、角色、测量、验证步骤、输出和出口准则
4、已定量管理	执行过程的质量	评审对象为“100 页”的需求规格说明，按照组织要求一般应该发现 10 至 20 个问题。如果某次评审只发现了 5 个问题，则可考虑是本次评审的质量存在问题，需要给与关注；如果发现 30 个问题，则说明本次文档编写存在大的质量问题——“用数据来说话”
5、优化	根据执行情况，优化改进组织过程	评审对象为“100 页”的需求规格说明，按照组织要求一般应该发现 10 至 20 个问题。如果大部分项目的需求规格说明评审都只发现了 9 个问题，则说明组织的要求需要调整和改进

## 验证与确认的区别

许多读者对 GJB5000A 三级要求的“验证”和“确认”两个过程域究竟有什么区别，不是十分清楚，下面是所搜集的一些对二者的解释和说明：

说法一：（1）“验证(Verification)的涵义

通过提供客观证据对规定要求已得到满足的认定。

（2）“确认（Validation）”的涵义

通过提供客观证据对特定的预期用途或应用要求已得到满足的认定。

（3）“验证”和“确认”之区别

“验证”和“确认”都是认定。但是，“验证”表明的是满足规定要求，而“确认”表明的是满足预期用途或应用要求，说简单点，“确认”就是检查最终产品是否达到顾客使用要求。

(4) “设计和开发”中“设计验证”和“设计确认”之区别在于：

设计验证的目的是检查设计输出是否满足设计输入的规定要求。

设计确认的目的是检查设计形成的最终产品是否达到顾客的使用要求。

说法二：

1. “确认”是要证明所提供的（或将要提供的）产品适合其预计的用途，而“验证”则是要查明工作产品是否恰当地反映了规定的要求。换句话说，验证要保证“做得正确”，而确认则要保证“做的东西正确”。

2. 验证注重“过程”，确认注重“结果”

3. (Verification) - - - Are we producing the product right?

(Validation) - - - Are we producing the right product? ( ----Boehm )

说法三：

1. 什么是验证？

验证 就是要用数据证明我们是不是在正确的制造产品。注意这里强调的是过程的正确性。

2. 什么是确认？



确认 就是要用数据证明我们是不是制造了正确的产品。注意这里强调的是结果的正确性。

3.验证和确认是一个广泛的概念，感兴趣的读者可以参考 IEEE Std 1012-1998 。

说法四：验证：验证检查某样东西是否符合之前已定好的标准，如：文档评审，要检查的东西是文档，检查标准就是文档的评审标准，又如：测试软件，要检查的东西就是软件，检查的标准就是软件的规格说明，包括功能说明，性能要求等。

确认：检查软件在最终的运行环境上是否达到预期的目标。一般来说，就是调试、验收测试等，这些工作都是在真正的软件需要运行的环境上进行的，在最终环境上运行软件，确保软件符合使用要求。

注：

Validation 更多是从用户的角度或者可以是模拟用户角度来验证产品是否和自己想要的一致。从 SP 来看可能更容易理解这个；

Verification 更多的是从开发方的角度来做评审、测试来验证产品的需求、架构设计等方面是否和用户要求的一致；

说法五：

（1）“验证(Verification)”的涵义

用试验的方法，来检验某个结论是否正确。

实践是检验真理的唯一标准，验证==检验+证明 。

也就是用实践来检验理论是否成立。验证之前，答案可能是对或错的。

## （2）“确认（Validation）”的涵义

已经知道某个结论，看看是否被有效执行，不需要验证理论的正确性，

是对执行力的检验。日本企业喜欢用“确认”一词，可以理解为：结合实物，到现场认真地看着。

# 如何对风险进行排序

对风险发生的可能性(P)分为 5 个等级，每个等级给出一个量化值，对风险可能造成后果的严重性(S)分为 5 个等级，每一个等级给出一个量化值， $\text{风险系数} = P \times S$ 。风险系数越大，表示风险发生的可能性和后果越严重，应该重点关注。

可能性 严重性	1	2	3	4	5
1	1	2	3	4	5
2	2	4	6	8	10
3	3	6	9	12	15
4	4	8	12	16	20
5	5	10	15	20	25

(2)、根据  $P*S*I$  值的大小进行排序

在考虑  $P*S$  的基础上 ,增加一个风险可能发生的时段因素 ,时段可分为 3 个等级 本阶段 , 下一阶段 , 下下阶段 ,给每个时段一个量化值 ,越靠前的时段 ,量化值越大。再根据  $P*S*I$  计算每个风险的系数来决定风险的排序。

# 项目阈值设置

阈值为一个领域或者系统的临界值 ,通常设置进度、工作量、成本、风险等多种状态的阈值 , 根据项目实际进度、工作量、成本与所设置的阈值进行比较 ,以便了解项目当前所处的状态。

阈值的设置方法简要介绍如下：

## (一) 进度阈值

进度阈值可根据进度绩效指数（SPI）或者项目进度偏差进行设置。

### 1 进度绩效指数（SPI）

SPI 来源于“挣值法”，挣（Earned Value，EV），表示实际完成的工作所对应的预算成本，在计划和实际之间建立了一个桥梁。其核心思想有这样几点：

- 1) 用成本指标来表示每个项目任务的价值，集中反映项目任务的时间、资源、成本、复杂度等多方面因素的影响。
- 2) 在实际完成同样工作的前提下，比较预算成本和实际成本之差，得到成本差异。换个角度来说，不管项目组实际花费了多大代价，也只能挣到预算中为这些任务安排的预算价值，超出的部分被看作是项目中浪费的，不应该由项目出资方承担。实际中这种算法也可以平衡由于项目范围差异所带来的成本差异。
- 3) 在花费同样成本的前提下，计划中应完成的任务与实际完成任务进行比较，得到进度差异。这里的进度是完成项目任务（工作量）的进度，不是单纯的时间进度。

对应上述思想，用指标、公式表示就是：

- a) 指标：BCWS 表示计划的任务对应的预算成本；ACWP 表示实际完成的任务对应的实际成本；BCWP 表示实际完成的任务对应的预算成本，就是挣值（EV）。
- b) 成本差异： $CV = BCWP - ACWP$ ，就是实际完成的任务，比较预算成本和实际成本之差。

c) 进度差异：SV=BCWP-BCWS，就是预算成本，比较实际完成的任务和计划中应完成的任务。

当到达某个时间点时，根据 SPI 计算项目实际进度判断是否达到所设置的阈值计算公式则为：

进度绩效指数 (SPI) = BCWP/BCWS

## 2 项目进度偏差 (%)

$$\text{项目进度偏差 (\%)} = \left| \frac{\text{项目实际进度}}{\text{项目计划进度}} - 1 \right| * 100\%$$

## (二) 成本阈值

对于成本阈值，可采用挣值法中的成本绩效指数 (CPI) 或者项目成本偏差进行设置。

### 1 成本绩效指数 (CPI)

成本绩效指数 CPI=BCWP/ACWP，就是已完成工作预算成本/已完成工作实际成本。

## 2 项目成本偏差 (%)

$$\text{项目成本偏差 (\%)} = \left| \frac{\text{项目实际成本}}{\text{项目计划成本}} - 1 \right| * 100\%$$

## (三) 工作量阈值

按项目工作量偏差(%)设置工作量阈值,当判断项目工作量是否达到所设置的阈值时,根据下列公式进行计算。

$$\text{项目工作量偏差}(\%) = \left| \frac{\text{项目实际工作量}}{\text{项目计划工作量}} - 1 \right| * 100\%$$

#### (四) 规模阈值

按项目规模偏差(%)设置规模阈值,当判断项目规模是否达到所设置的阈值时,根据下列公式进行计算。规模可按功能点数/需求数/代码行数等单位进行计算。

$$\text{项目规模偏差}(\%) = \left| \frac{\text{项目实际规模}}{\text{项目计划规模}} - 1 \right| * 100\%$$

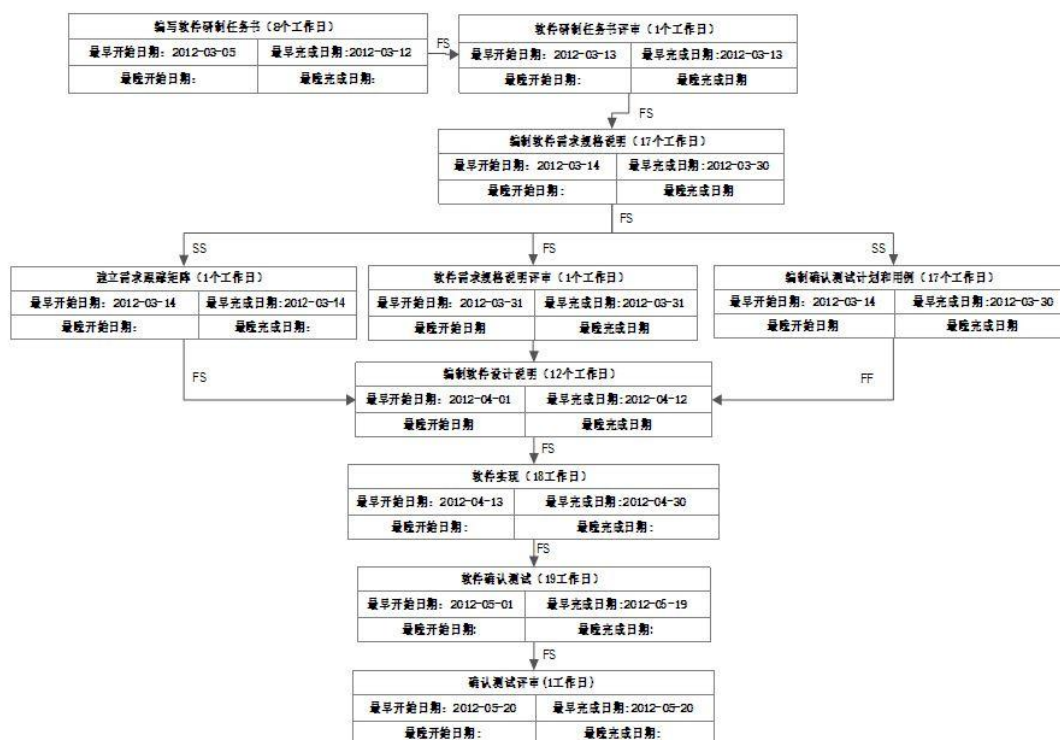
## 如何计算关键路径

在项目计划中,活动与活动之间有4种关系类型,FS(完成-开始),FF(开始-开始),SS(开始-开始),SF(开始-完成),根据活动的关系、工期,计算关键路径,步骤如下。

(一)采用正推法,计算出每个活动的最早开始日期(ES)、最早完成日期(EF)。最早开始日期(ES)指某项活动最早能够开始的时间,最早完成日期(EF)指某项活动最早能够完成的时间,EF根据ES和工期估计,正向推算得到,EF=ES+工期估计-1。

例:假设编制软件研制任务书最早开始日期为:2012-03-05,工期为8个工作日,根据正推法,得到该活动的最早完成日期为2012-03-12,计算以该活动为前导活动的那些活动的

最早开始日期 ( ES ) 和最早完成日期(EF),  $ES(\text{最早开始日期})=EF(\text{前导活动的最早完成日期})+1$ , 以此类推, 求出其它活动的最早开始日期和最早完成日期, 页面示例下图所示。

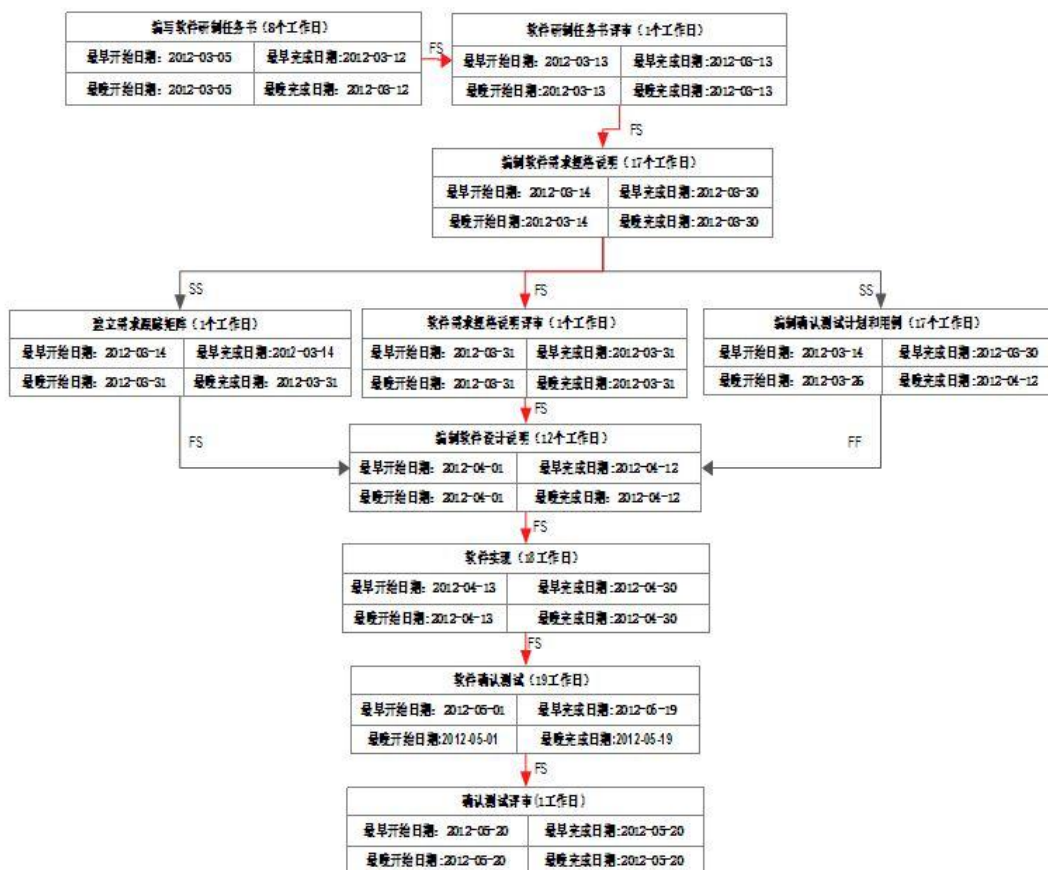


计算活动最早开始日期 (ES) 和最早完成日期(EF)

(二)采用倒推法:计算任务的最晚开始日期( LS )和最晚完成日期(LF)。最晚开始日期( LS )指为了某项活动能够在要求的时间节点完成, 最迟必须开始的时间;最晚完成日期(LF)指为了某项活动能够在要求的时间节点完成, 最迟必须完成的的时间。

设最后一个活动的最晚完成日期 ( LF ) =最早完成日期 ( EF ),  $LS=LF-\text{工期}+1$ 。计算以该活动为后继活动的活动的 LF 和 LS ,  $LF=LS(\text{后继活动的 LS})-1$  ,  $LS=LF-\text{工期}+1$  , 以此类推, 得到其它活动的 LS 和 LF。





### （三）求出关键路径

每个活动的最晚完成日期与最早完成日期 ,或者最晚开始日期与最早开始日期的差额就是该活动的时差，如果某活动的时差为零，那么该活动就在关键路线上，如上图的红线。

## 体系一致性检查

体系文件编写完成后，如何验证各过程、各表单之间的一致性？可通过“表单数据流向分析法”来检查，表单数据流向分析法的步骤是：

（1）按照软件项目阶段顺序和过程域分类整理体系中使用或产生的各种表单、报表、检查单等，并为其逐一编号；

- (2) 按照数据产生或使用的先后顺序，逐一确定其来源和获取方式，并对其进行标识；
- 在进行此项标识时，将会检查出各表之间数据的不一致性，包括量纲的不一致性，数据缺项等。
- (3) 为了检查体系对每个活动是否规定了执行角色和实际，可在此表中同时设置“使用时机”和“执行角色”两项。

## 体系完备性检查

体系文件编写完成后，许多编写者总是担心所编写的体系是否覆盖了 GJB5000A 标准的要求。可通过编制体系文件完整性检查单，然后逐项检查的方式来验证体系的完整性。

编写体系文件完整性检查单时，可从 GJB5000A 的各过程域、目标和实践出发，梳理体系中对应文件、章节，建立对应关系，并给出各软件项目实施体系时的落脚点——即“证据”。

下图是体系文件检查单的一个片段。