
如何实施基于CMMI的过程管理

冯云显

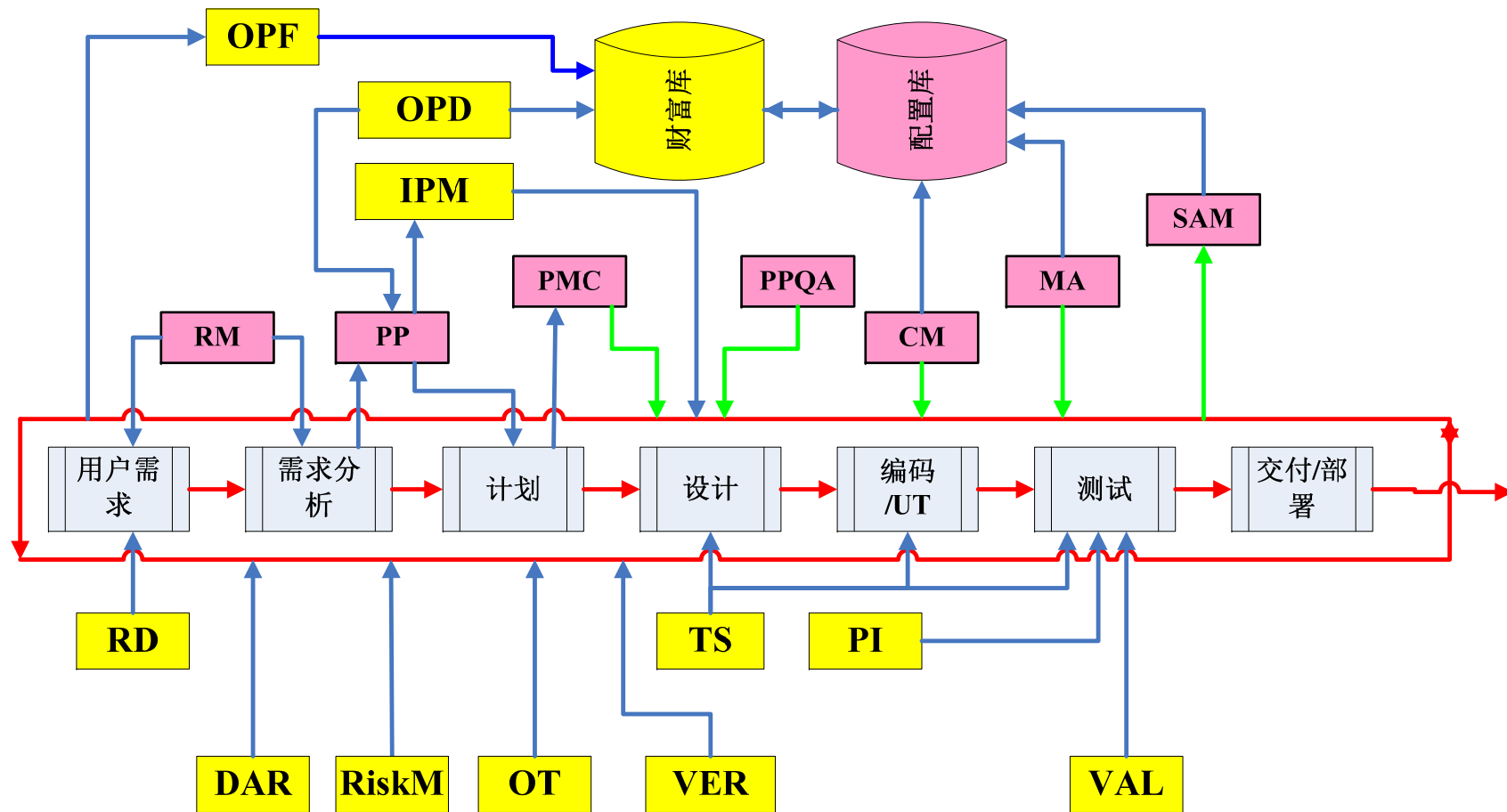
Topic

- 过程域之间的关系图
 - 看看“变”化
 - 过程改进的一些反对借口
 - CMMI执行中的一些常见问题
 - 项目组如何实施CMMI?
 - “小”变化“大”改进
 - Q&A
-

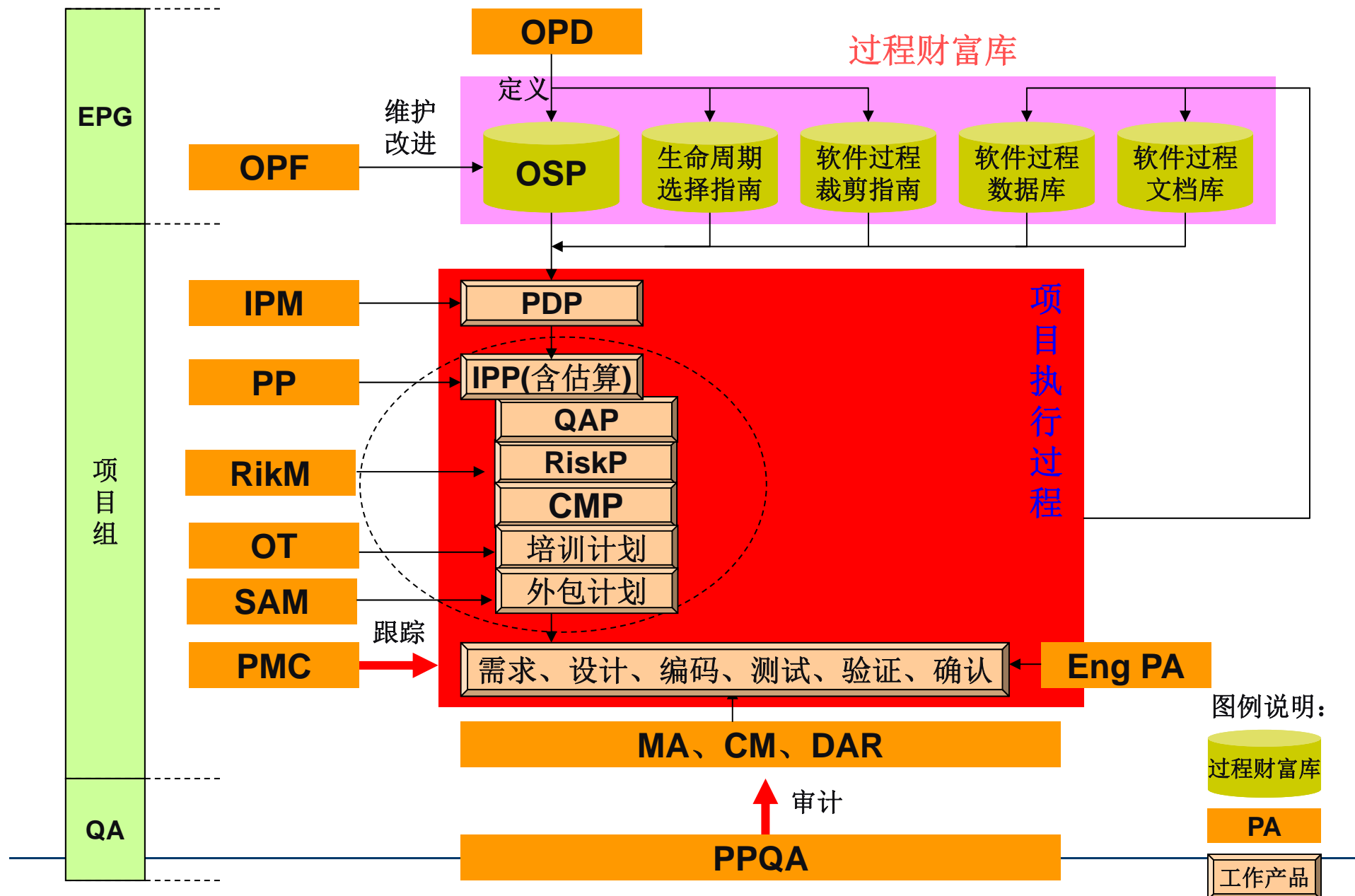
CMMI2级软件开发实施与PA关系

CMMI L2-3各PA地位

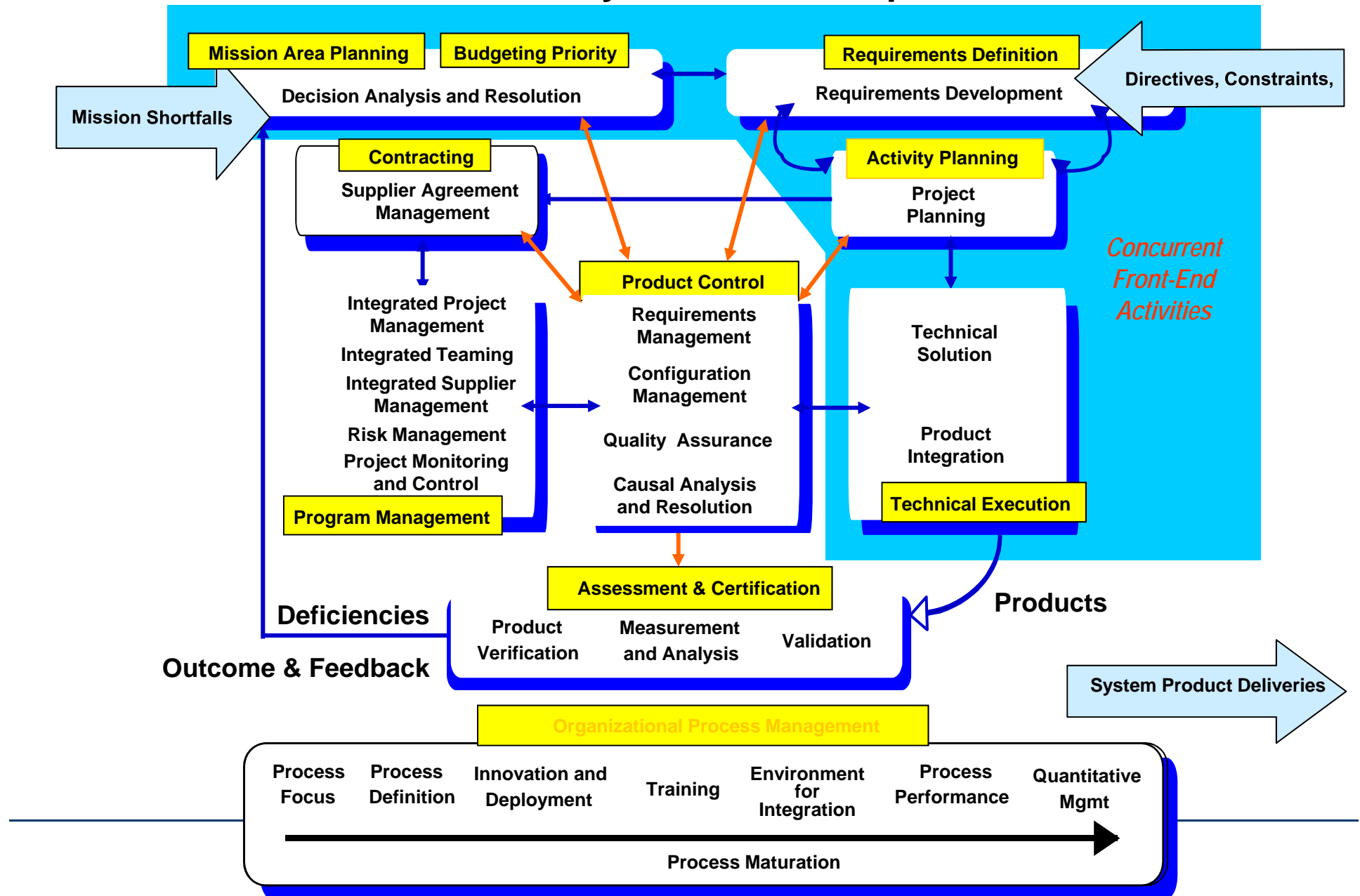
(图中粉色为L2、黄色为L3)



CMMI-ML 3级 软件过程体系结构图：



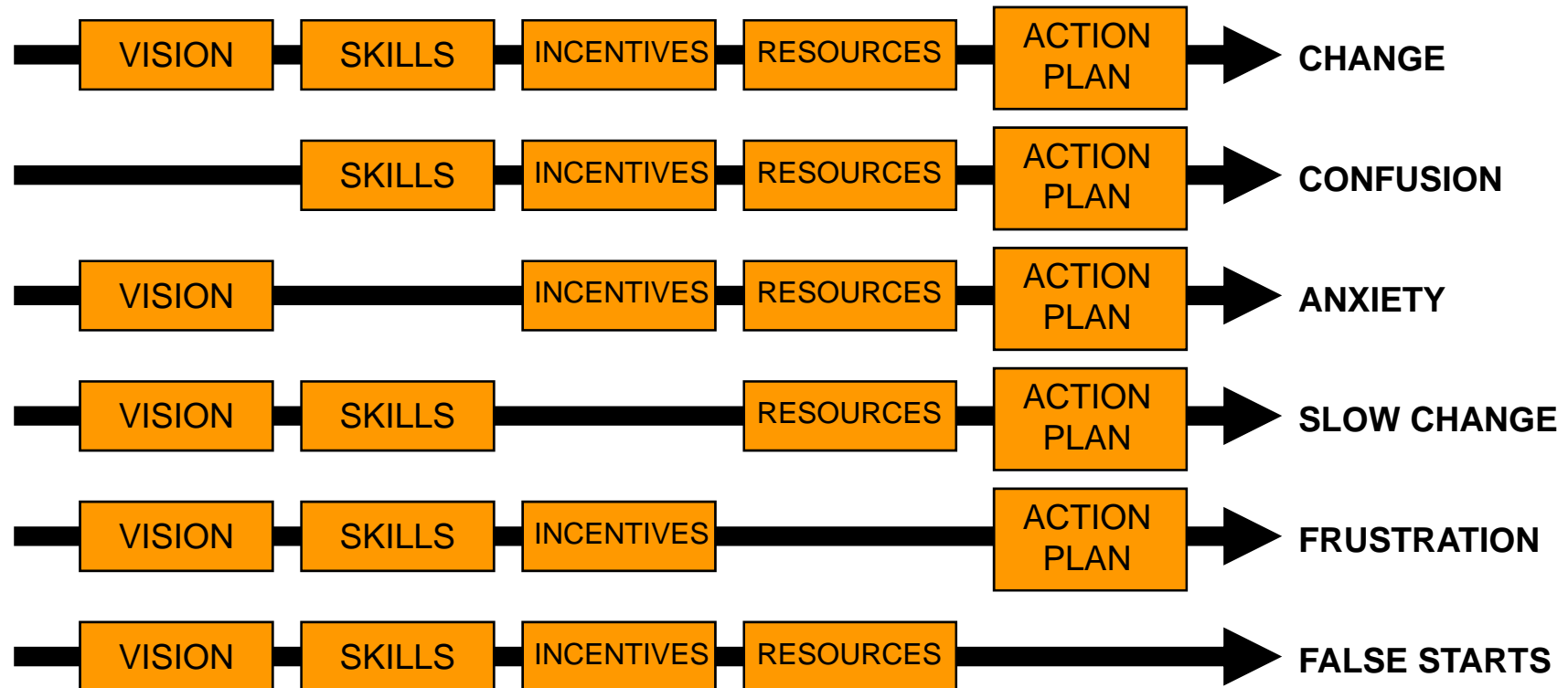
Life Cycle Relationships



Topic

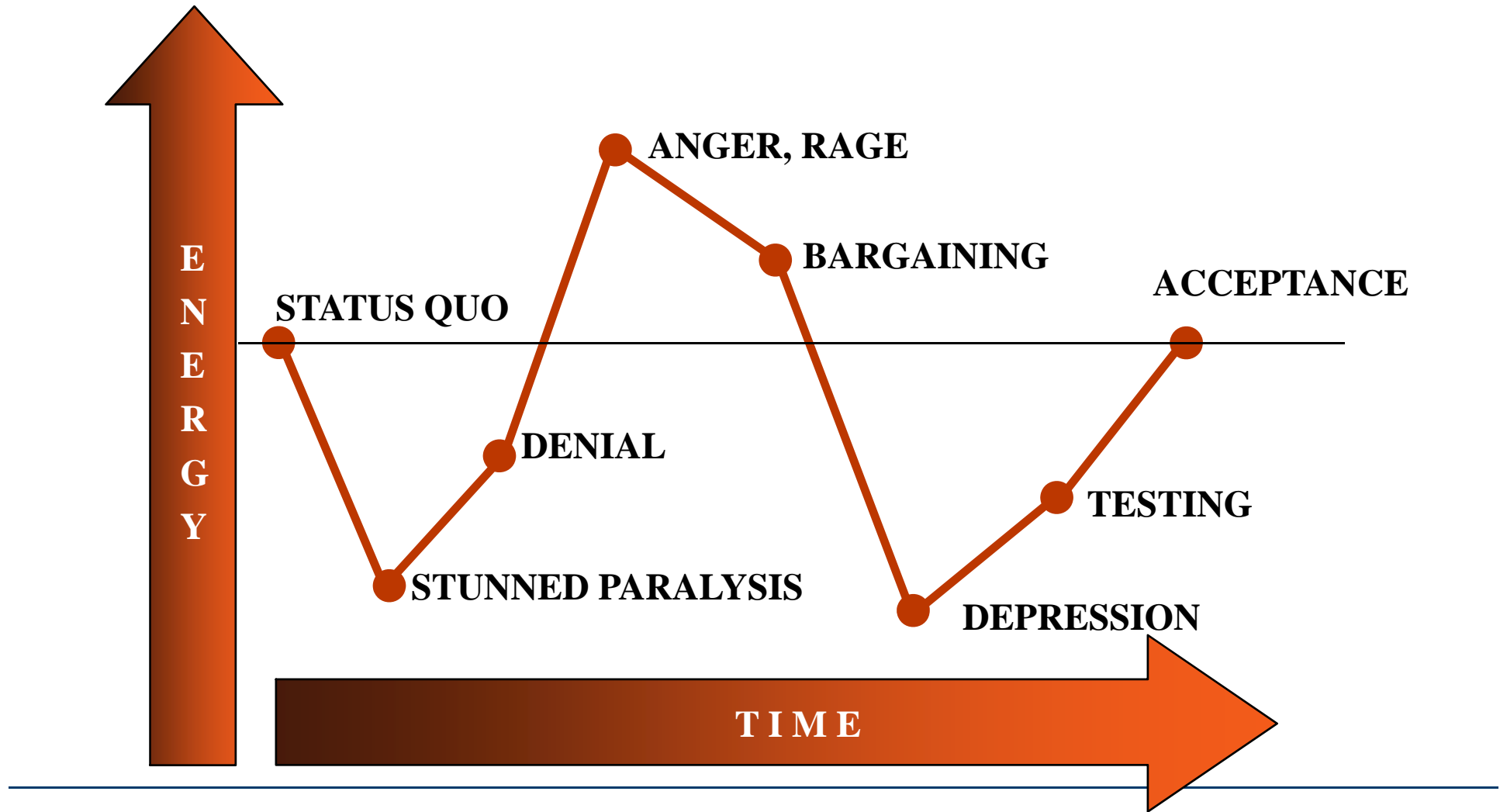
- 过程域之间的关系图
 - 看看“变”化
 - 过程改进的一些反对借口
 - CMMI执行中的一些常见问题
 - 项目组如何实施CMMI?
 - “小”变化“大”改进
 - Q&A
-

Effects of Missing Change Elements



Source: Ambrose, 1987

Resistance To Change



Topic

- 过程域之间的关系图
 - 看看“变”化
 - 过程改进的一些反对借口
 - CMMI执行中的一些常见问题
 - 项目组如何实施CMMI?
 - “小”变化“大”改进
 - Q&A
-

- **CMMI** 可能是一个比较新的改进模型，但不愿意进行过程改进的借口却不是新的。
 - 有过程改进实施经验的人士则对这些借口不很陌生，我们将在以下讨论一下反对过程改进的借口。
-

反对借口

1. 我没有时间做这个，以后再说吧！
 2. 我不相信它（或：我看不出它有什么好处）
-

反对借口（续）

3. 我的客户很开心，我不需要改变任何事情！

- ◆ 你难道不想让你的客户更加开心吗？
 - ◆ （“只有平庸的人才总是认为他是最棒的”
—— **William Somerset Maugham**）
-

反对借口（续）

- 4. 这不过是一种时髦，象**TQM**一样。
 - ◆ **CMM**从**1990**年早期就已经有了，这是时髦吗？
 - 5. 它的代价太大了！
 - ◆ 质量代价多少？
 - ◆ 不做这个才会代价太太！
-

反对借口（续）

- 6. 它不适用于我的项目！（或：我们的项目不同！）
 - 7. 我的客户不会付这笔钱的！
 - ◆ 客户本就不应该付这笔钱。
 - ◆ 最佳实践本就是你应该做的事情。
-

反对借口（续）

8. 那根本就是多余的！（或：我们有真正的工作要做）



呢？

风险增大！

	<u>Project A</u>	<u>Project B</u>	<u>Project C</u>	<u>Project D</u>	<u>Project E</u>
做这些	Code Fix Deliver	Code Fix Deliver	Code Fix Deliver	Code Fix Deliver	编码 修正 交付
存在的风险	没有需求控制 设计缺陷 交付产品有Bug	Requirements Design	Requirements Design Test	Requirements Design Test	需求 设计 测试
	活动无计划 对状态不清楚 需求没有实现 没有版本控制 没有文档	活动无计划 对状态不清楚 需求没有实现 没有版本控制 没有文档	Plan Track QA CM Document	Plan Track QA CM Document	计划 跟踪 质量保证 配置管理 文档化
	没有资源 没有标准 没有历史数据 人员缺乏培训 管理层不知道	没有资源 没有标准 没有历史数据 人员缺乏培训 管理层不知道	没有资源 没有标准 没有历史数据 人员缺乏培训 管理层不知道	PAL OSSP OMDB Training Reviews	资产库 OSSP OMDB 培训 审核
	没有部门级承诺 没有组织级协调 不知道成熟度级别	没有部门级承诺 没有组织级协调 不知道成熟度级别	没有部门级承诺 没有组织级协调 不知道成熟度级别	没有部门级承诺 没有组织级协调 不知道成熟度级别	EPG MSG SCAMPI

Topic

- 过程域之间的关系图
 - 看看“变”化
 - 过程改进的一些反对借口
 - CMMI执行中的一些常见问题
 - 项目组如何实施CMMI?
 - “小”变化“大”改进
 - Q&A
-

项目管理域常见问题

- ❑ 粗糙的项目定义过程（PDP），没有好的裁剪方法（指南）是一个重要原因。
 - ❑ 不做估计，认为这没有意义，或者只做工作量估计，或者估计由PM说了算。
 - ❑ 不做WBS，不明白WBS是一种普遍适用的方法，而养成好习惯。
 - ❑ 不善于充分合理安排并行作业。
 - ❑ 不善于掌握人对事物认识是逐步逼近和迭代改进的方式，认为什么事都必须一步到位。
-

项目管理域常见问题

- ❑ 迫于交付有期限，只能倒排时间，加班加点。想不到用过程定义、并行作业、充分的重用、步骤的合并、抓大(重)放小(轻)、简化文档、快速的PDCA、缩短子过程时间等办法。
 - ❑ 计划的制定与实施，仰仗团队的技能、合作能力，更需要良好的人人积极向上的热情，否则计划不能完成，产品质量也无保证。
 - ❑ 风险和问题混淆。
 - ❑ 风险管理形式大于内容，风险冻结，缺少跟踪记录。
-

项目管理域常见问题

- ❑ 局限于对进度跟踪，不关注对工作量、成本、规模、风险等的跟踪。
 - ❑ 不重视日常跟踪，到阶段或项目结束才做总结。
 - ❑ 不重视问题的处理机制。
 - ❑ 不设定项目的警戒值。
 - ❑ 集成项目管理不集成，过程不集成，计划不集成，人员不集成。
 - ❑ 不能有效识别关键干系人，并进行管理。
-

支持过程域常见问题

- ❑ QA审核流于形式，未能有效发现问题。
 - ❑ 过程审计不及时，发现问题纠正代价高于不纠正。
 - ❑ 项目组不重视过程的质量，对QA审计不配合。
 - ❑ 高层不理解QA的重要作用，不能给予QA工作有效支持。
 - ❑ QA和QC概念不清，产品质量问题责任推给QA。
 - ❑ 配置管理的变更分析不到位，影响配置项不明确。
 - ❑ 配置审计流于形式，不重视功能审计。
-

支持过程域常见问题

- 不明确度量分析的目的，为度量而度量。
 - 度量分析与项目监控脱离。
 - 度量缺乏统一的定义。
 - 项目只对重大问题使用决策分析方法，不了解对于小问题也适用于DAR。
-

工程过程域常见问题

- ❑ 需求主动挖掘不够，需求不明确。
 - ❑ 需求很少进行优先级分析。
 - ❑ 需求不进行基线管理，需求变更频繁，不按变更流程执行变更，导致项目没完没了，产品间不一致很多。
 - ❑ 需求跟踪流于形式。
 - ❑ 不明确总体技术方案就匆忙设计和实现。
 - ❑ 设计中描述不清楚，对复杂的结构没有用图形描述。
 - ❑ 复杂的业务处理没有用流程图体现业务流程，而是用大段的文字描述。
-

工程过程域常见问题

- ❑ 需求发生了变化或需求理解不够，设计工作反复，返工工作量大。
 - ❑ 详细设计流于形式，或编码完成于详细设计之前，对后续的指导意义不大。
 - ❑ 详细设计和实际的编码已经脱节，不具有溯源性，对后期的维护，指导意义也不大。
 - ❑ 编码风格不统一或不遵循编程规范。
 - ❑ 对程序文件、类、方法不做注释，或者极少做注释。
-

工程过程域常见问题

- 单元测试不充分，缺少测试方面的经验。
 - 代码审核的力度不够，审查覆盖率低，投入审核的工作量很少。
 - 代码评审经验：
 - 由于代码量巨大，不可能评审所有代码 。
 - 从需求角度，评审需求不是很明确的模块 。
 - 从编码工程师角度，新员工、规范性差的、代码质量差的员工成为代码审核的重点。
 - 从代码重要性角度，设计复杂的、核心的、基础性组件也是检查的重点。
 - 代码评审不需要项目组的全体成员参加 ，只需要一个架构师和两个有经验的编码工程师就可以评审了。
-

工程过程域常见问题

- ❑ 不设计测试用例就执行测试。
 - ❑ 由于前期需求和设计文档不够明确，不能有效设计用例，测试陷入黑盒中的黑盒。
 - ❑ 测试用例往往浮于表面，无法深入挖掘系统的隐性需求，测试用例不能体现设计中的问题，不能体现一些业务关键点处理。
 - ❑ 测试覆盖面不够。
 - ❑ 回归测试时，无法很好地把握与修改点相关性的测试点。
 - ❑ 无法准确把握测试的入口与出口准则，特别是测试出口准则的把握。
-

测试出口准则控制要严

- 代码应该覆盖**100%**的设计和需求
 - 应该合理使用转移语句，以保持程序的结构化性质
 - 在代码和单元测试用例以及在代码和设计之间，已经建立了映射关系（跟踪矩阵）
 - 对前面所开发的设计和规格说明以及在规格说明和客户需求之间的映射关系（跟踪矩阵），已经进行了细化/修改
 - 应该成功地执行所有的测试用例；对没有通过的测试用例要进行分析
 - 语句覆盖应该达到**100%**
 - 应该对系统的新编代码和核心代码进行同行评审
 - 进行同行评审的代码行数应该不小于系统总的代码行数的**25%-35%**
 - 同行评审所花费的工作量，应该不小于编码和单元测试阶段所花费的总工作量的**20%-30%**
 - 所发现的不同严重等级的缺陷应该小于给定值（例如**1.0-2.0/KLOC**）
 - 所发现的严重程度等级 **A** 缺陷已经全部修复，没有修复的严重程度等级 **B** 缺陷应该小于给定值（例如**20%**）
-

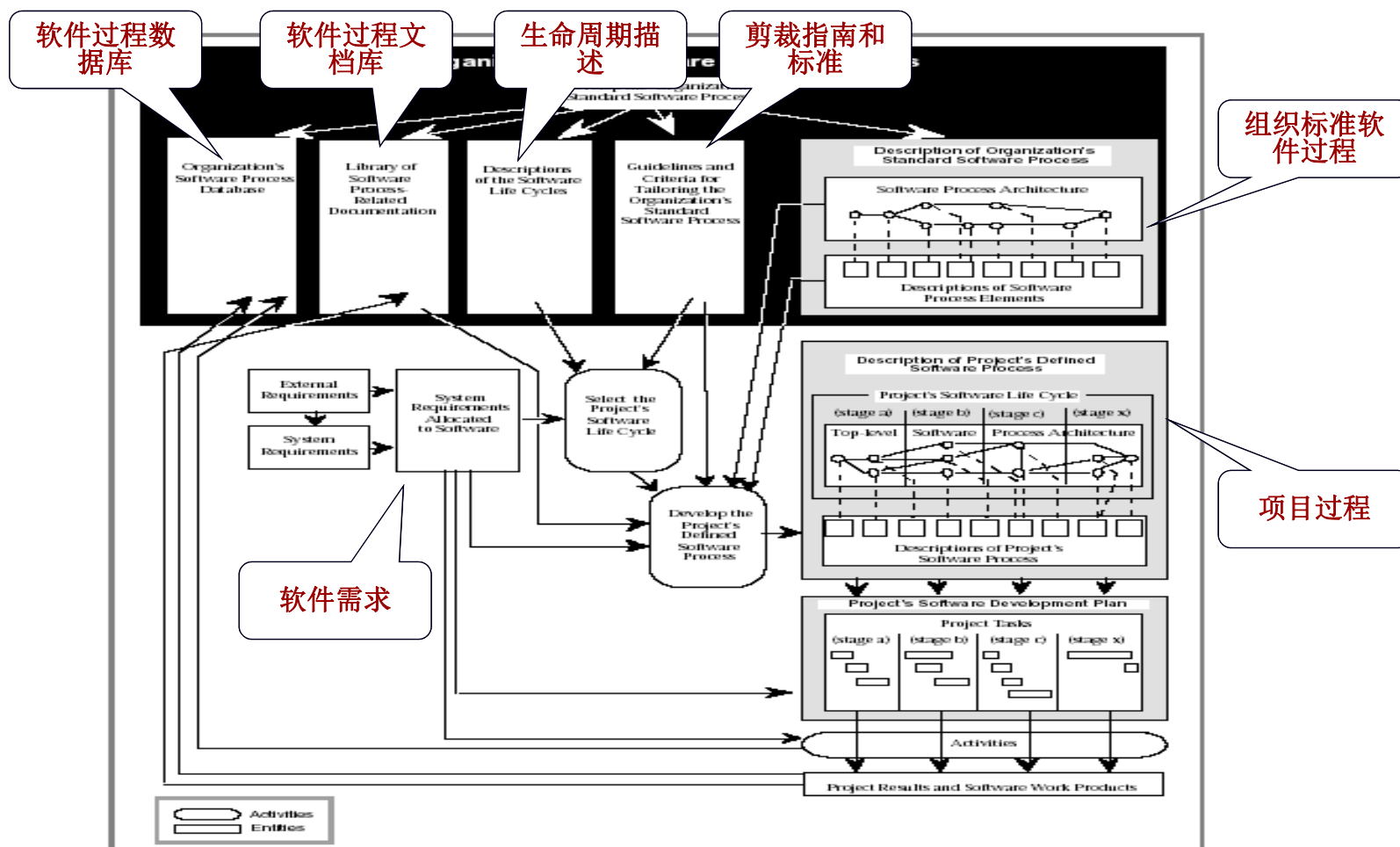
组织过程域常见问题

- 没有真正明确组织培训需求。
 - 对培训资源投入不足，培训执行力不够。
 - 缺乏有效的培训考核手段。
 - 过程改进计划和活动流于形式。
 - 过程改进只是少数人（EPG）的职责，员工游离于过程改进之外。
 - 急功近利思想导致过程改进失败。
 - 过程改进执行力不足，这是过程改进失败最重要的原因之一。
 - 过程改进缺乏配套机制。
-

Topic

- 过程域之间的关系图
 - 看看“变”化
 - 过程改进的一些反对借口
 - CMMI执行中的一些常见问题
 - 项目组如何实施CMMI？
 - “小”变化“大”改进
 - Q&A
-

定义项目过程&制作项目计划



如何进行项目的跟踪与监控

— 项目跟踪与监控方法

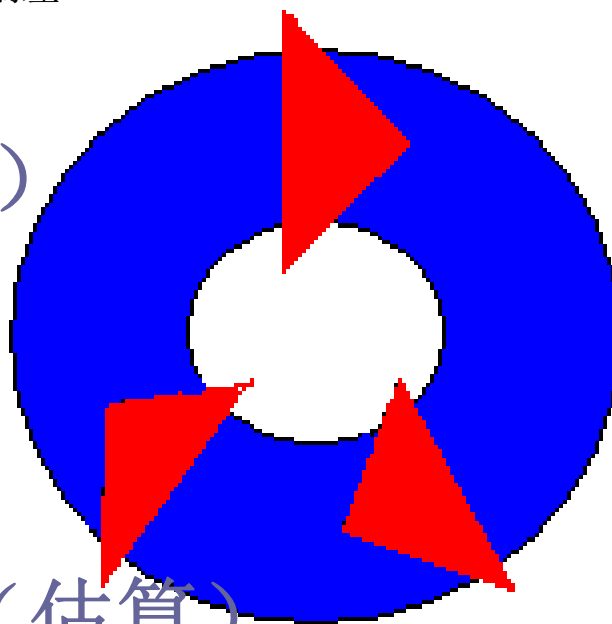
项目跟踪与监控的本质

- 计划（估算）
- 收集实际数据
- 比较实际值与计划（估算）值的差异
- 重新计划（估算），修正偏差

计划（估算）

比较

重新计划（估算）



第一步 制订项目计划

- 估算项目规模
 - 计划项目进度
 - 确定里程碑
 - 确定阶段
 - 计划项目任务
 - 估算项目成本
 - 评估项目风险
 - 评估与确定项目的关键计算机资源
 - 估算与分配项目的工作量
-

第二步 确定项目跟踪与监控策略

- 确定跟踪与监控的范围
 - 规模
 - 进度（关键路径，所有路径）
 - 费用（所有费用，设计费、折旧费等）
 - 工作量（按核心工作流，按明细任务等）
 - 风险（前五大风险，前十大风险，所有风险）
 - 关键计算机资源
 - 其他（干系人，承诺，技术进展，需求状态，基线状态等）
-

确定项目跟踪与监控策略（Cont.）

□ 确定每个跟踪项的跟踪频度

- 规模（每周、每月、每阶段、每里程碑等）
 - 进度（每天、每周、每月、每阶段、每里程碑等）
 - 费用（每周、每月、每阶段、每里程碑等）
 - 工作量（每天、每周、每月、每阶段、每里程碑等）
 - 风险（每周、每月、每阶段、每里程碑等）
 - 关键计算机资源（每周、每月、每阶段、每里程碑等）
 -
-

确定项目跟踪与监控策略（Cont.）

□ 确定数据的收集方式

- 明确每项跟踪数据的提供者
 - 明确每项跟踪数据的接收者
 - 明确使用的工具（数据库、电子表格等）
 - 确定收集途径（自动工具、邮件、在例会上陈述等）
-

确定项目跟踪与监控策略（Cont.）

□ 确定每个跟踪项的偏差计算公式及域值

- 规模（（当前实际规模-上次估算规模）/上次估算规模； $\pm 10\%$ ， $\pm 20\%$ 等）
 - 进度（（当前实际进度-计划进度）/计划进度）； $\pm 5\%$ ， $\pm 10\%$ ， $\pm 20\%$ 等）
 - 费用（（实际费用-预算费用）/预算费用）；预算 $\pm 10\%$ ， $\pm 20\%$ 等）
 - 工作量（（实际工作量-计划工作量）/计划工作量； $\pm 10\%$ ， $\pm 20\%$ 等）
 - 风险（可在风险管理计划中明确）
 - 关键计算机资源（有/无，充足/不充足，最低数量等）
-

确定项目跟踪与监控策略（Cont.）

- 确定偏差的处理措施
 - 每个跟踪项的偏差超出域值下限时应采取的措施，超出域值上限时应采取的措施
 - 规模
 - 进度
 - 费用
 - 工作量
 - 风险
 - 关键计算机资源（资源不足或不可用时应采取的措施）
 -
-

第三步 在项目过程中进行跟踪

- 在项目进行的过程中，根据项目计划中定义的频率和收集方式收集项目的跟踪数据



第四步 分析跟踪数据

- 利用实际跟踪数据和计划（估算）数据、偏差计算公式计算偏差率：

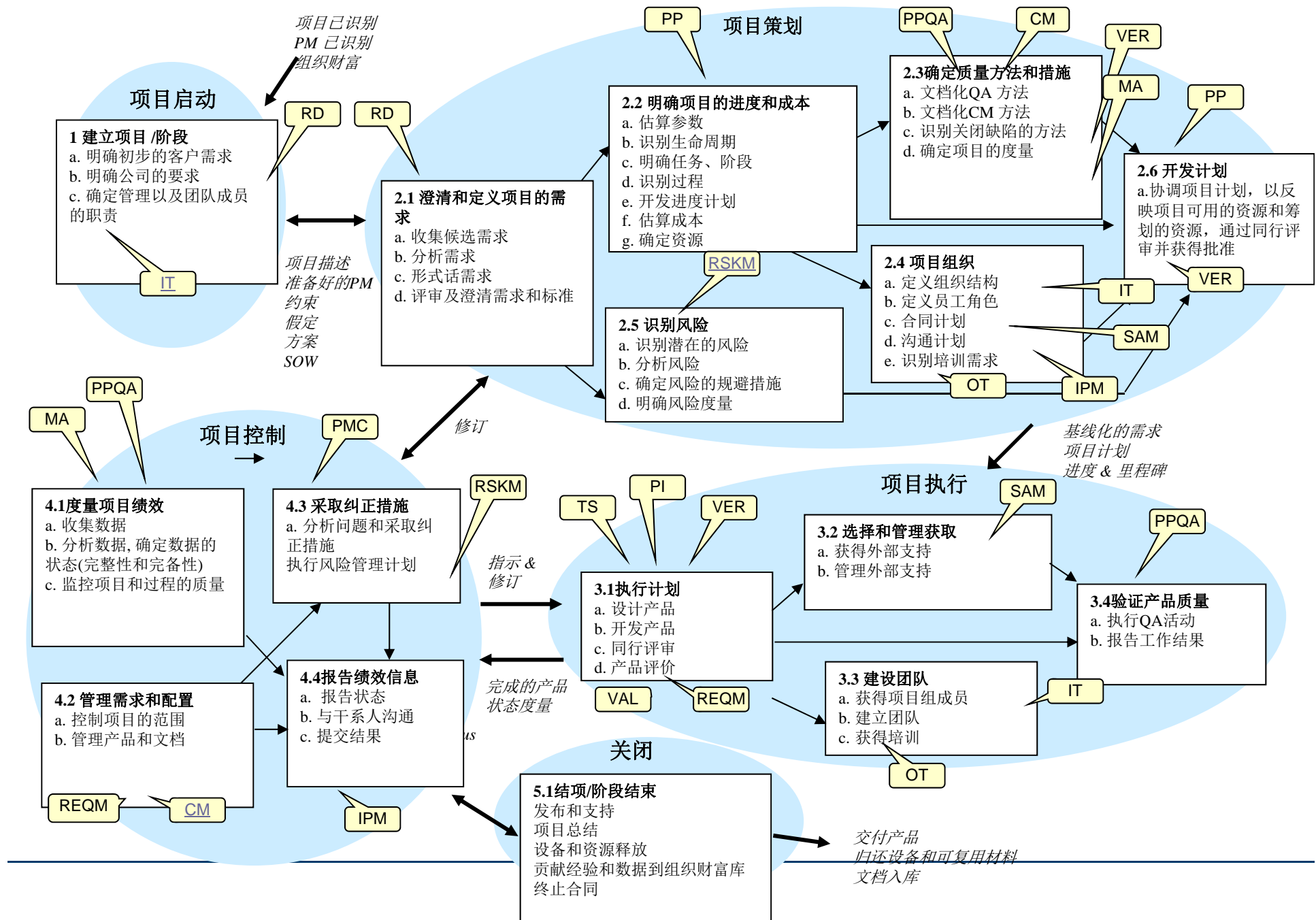
$$\blacksquare \text{ 偏差率} = \frac{\text{实际值} - \text{计划值}}{\text{计划值}} \times 100\%$$

第五步 采取行动

- 根据跟踪与监控策略，对发生的偏差采取相应的行动：
 - 一般情况下，若偏差超出域值范围，应对规模、工作量、费用进行重新估算，应重新调整项目进度、重新计划项目任务、重新分配工作量。
-

案例

	周	月	里程碑
进度	√	√	
风险	√	√	
规模			√
费用			√
工作量		√	
关键计算机资源		√	



实施五步法

- 执行
 - 度量
 - 反馈
 - 改进
 - 总结（最佳实践）
 - 再循环（执行、度量、反馈、改进、总结）
-

Topic

过程域之间的关系图

看看“变”化

过程改进的一些反对借口

CMMI执行中的一些常见问题

项目组如何实施CMMI?

“小”变化 “大”改进

Q&A

Definitions 定义

- Small change means the change can be implemented within 4-6 weeks.
 - 小改变意指这个改变能在4~6周内得到实施。
 - Big improvement means the improvement influents the developing sub-system, people/culture sub-system and environment sub-system.
 - 大改进意指影响开发子系统、人/文化子系统以及环境子系统的改进。
-

Good Practices #1 优秀实践#1

Happy&unHappy 高兴和不高兴

- It is a team building tool.
 - 它是团队建设工具
 - Daily, after work, identify one person facilitates everyone of the team to say What is happy and What is un-happy personally today?
 - 每天工作后，让团队的每个人说说今天发生的愉快和不愉快事件
-

Good Practices #2 Lunch&Learn

优秀实践#2 午餐与经验交流

- It is a team building tool.
 - 它是团队建设工具
 - Daily, lunch with different person. To talk Hobbies, Personal Good Practices for better life.
 - 每天与不同的人午餐，聊聊业余爱好以及分享一些个人美好人生的经验。
-

Work

优秀实践#3不要加班工作

- It is a risk management tool.
 - 它是风险管理工具
 - Define Overtime Work as one of the top risks should be managed by the project team.
 - 把加班工作定义为一个高风险，并被管理。
 - Analysis historical data to know the root cause of overtime work.
 - 分析历史数据以明确加班工作的根本原因。
 - Improve the estimates.
 - 改进估算
-

Good Practices #4 Risk Reward

优秀实践#4 风险奖赏

- It is a risk management tool.
 - 它是风险管理工具
 - Daily, high management find engineers to tell him what are the top risks of their projects and the company.
 - 管理者每天告诉工程师公司和项目存在哪些高级别的风险。
 - Monthly/Annually reward the reporters of the best risk.
 - 每月/每年奖赏那些最佳风险报告者
-

Good Practices #5 Candidate PM

优秀实践#5 候选项目经理

- It is a motivation tool.
 - 它是激励工具。
 - At the end of each project, PM should recommend 2 persons as the candidate PMs.
 - 每个项目结束时，项目经理推荐2名候选项目经理。
 - Candidate PM will practise knowledges and skills in project as an assistant PM.
 - 候选项目经理将作为项目经理的助理在项目中得到项目管理知识和技能的培训和实践。
 - After 2 projects of practices, assistant PM can be selected to be PM.
 - 完成两个项目后，助理项目经理可以被选择为项目经理。
-

Information

优秀实践#6 不要有重复信息

- It is a PM tool.
 - 它是项目管理工具
 - No repeated information within a document or across different documents.
 - 在一个文档或交叉的不同文档里不要有重复的信息
 - Project data is centrally managed.
 - 以项目数据为管理中心。
 - Information can be cross referenced.
 - 信息能被相互引用或参考
-

Good Practices # / My Next Project

优秀实践#7 我的下一个项目

- It is a PM tool.
 - 它是项目管理工具
 - Let the people working for current project know when he/she will be assigned to the next project and what he/she will be responsible for.
 - 让当前项目的人员知道他/她将被分配的下一个项目以及相应职责
 - Tip: People will work faster and better when they are eager to do something next.
 - 小技巧：当人们渴望去做接下来的某些事情的时候，他们会把当前的工作做得更快和更好。
-

Timesheet

优秀实践#8个人时间卡

- It is a Personal Proecess Improvement tool.
 - 它是个人过程改进工具
 - Register the informaiton of tasks and time people personally does and analysis it.
 - 记录任务和工时
 - People will use it to approve estimates.
 - 可以用它来审批估算
 - People will use it to improve their time management skill.
 - 可以用它来改进个人的时间管理技能。
-

There must be some more others?
一定有其它更多的优秀实践？

- Please tell me some since I have told you eight, okay?
 - 请告诉我还有那些优秀实践？
-

What can be done next?

接下来要做的是？

- Step-1: Select the practices you want to implement in the organization
 - 第一步：选择你要在组织实施的实践。
 - Step-2: Set up responsibilities for these small changes
 - 第二步：建立这些小改变的相关职责
 - Step-3: Prepare action plan with time constraint of 4-6 weeks
 - 第三步：准备在4~6周完成的行动计划
 - Step-4: Do it
 - 第四步：执行行动计划
 - Step-5: Improve it
 - 第五步：改进
 - Tip: Get professional supports from EPG.
 - 小技巧：从组织EPG获取专业支持。
-

Topic

- 过程域之间的关系图
 - 看看“变”化
 - 过程改进的一些反对借口
 - CMMI执行中的一些常见问题
 - 项目组如何实施CMMI?
 - “小”变化“大”改进
 - Q&A
-

问题与回答

