



IBM Software Group

# 金融行业软件开发管理平台方案

**Rational** software

姚炳雄 [yaobx@cn.ibm.com](mailto:yaobx@cn.ibm.com)



**ON DEMAND BUSINESS™**

© 2006 IBM Corporation

# 议程

- 软件开发管理的问题及需求
- IBM的解决方案
  - ▶ 软件开发流程
  - ▶ 软件资产库
  - ▶ UCM最佳实践
- 收益

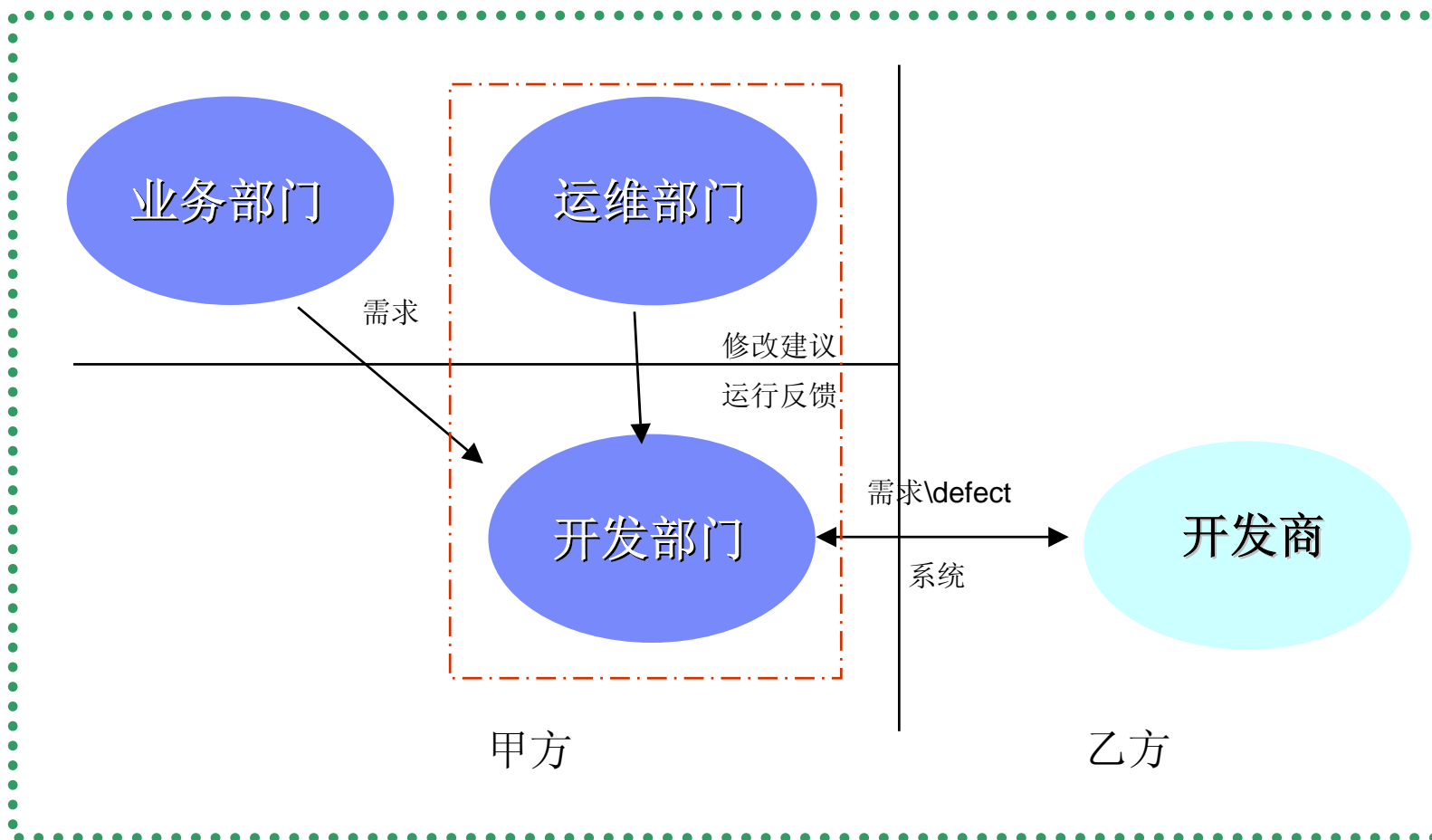


# 软件开发管理中存在的问题

- 需求方面的问题
  - ▶ 业务部门的需求变化快
  - ▶ 需求的来源和形式多种多样
  - ▶ 需求质量无法达到可跟踪标准
- 开发流程方面的问题
  - ▶ 开发队伍没有严格的角色划分
  - ▶ **Bug**反馈不及时,经常将重要的任务遗漏
  - ▶ 开发进度无法控制,多次延期而无能为力
- 软件资产方面的问题
  - ▶ 软件资产没有一个安全的集中的存放和管理,缺乏完善的安全保障和安全机制
  - ▶ 某些软件甚至只有可执行二进制码,软件资产维护受制于人
  - ▶ 难以根据已有资产进行可持续性开发
  - ▶ 资产组织形式和分析设计模型脱节

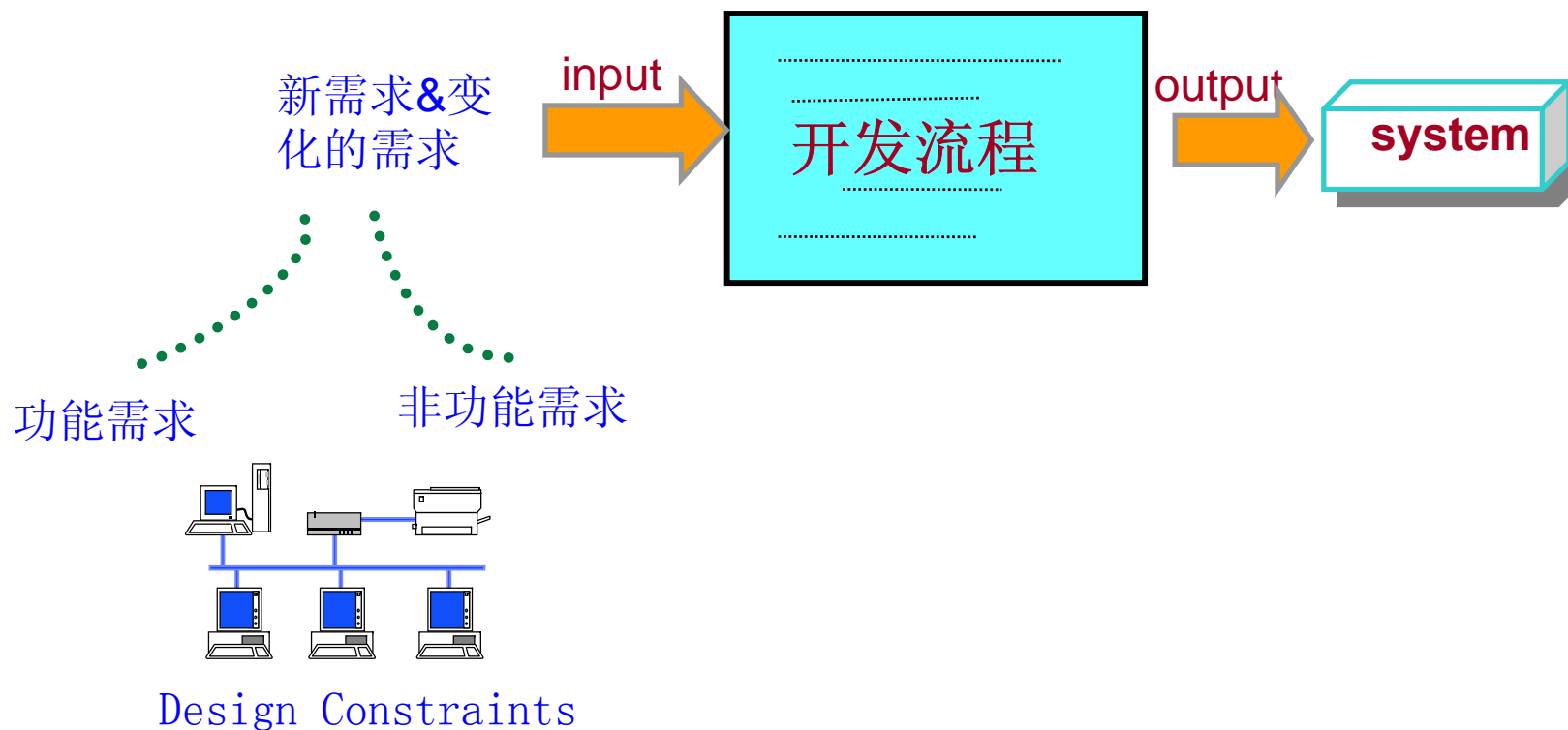


# 开发管理相关组织机构

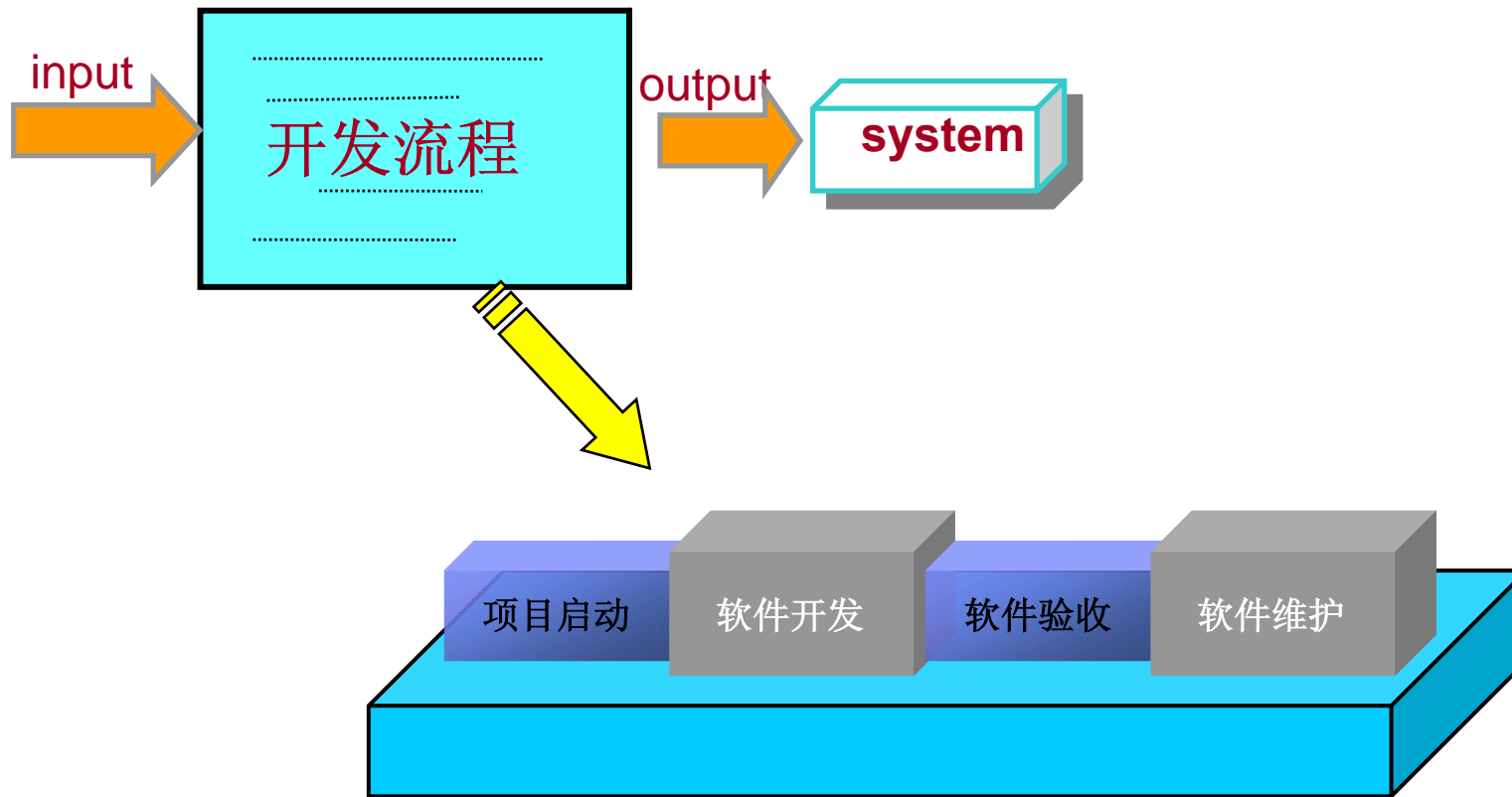


# 软件开发流程

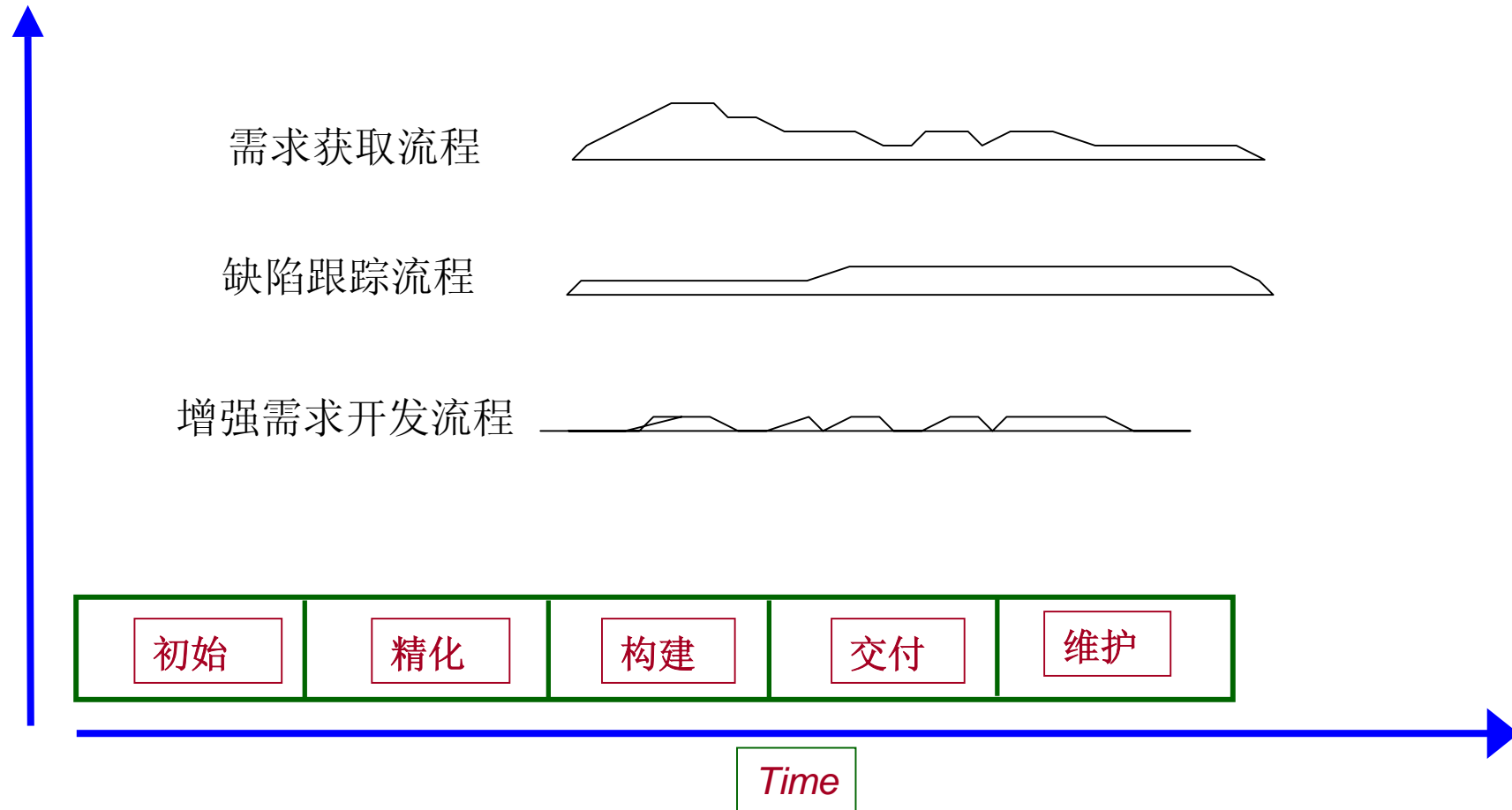
流程:即定义了谁,在什么时间,为达到某一特定目标,去做一件什么事情,并指出怎么去做.



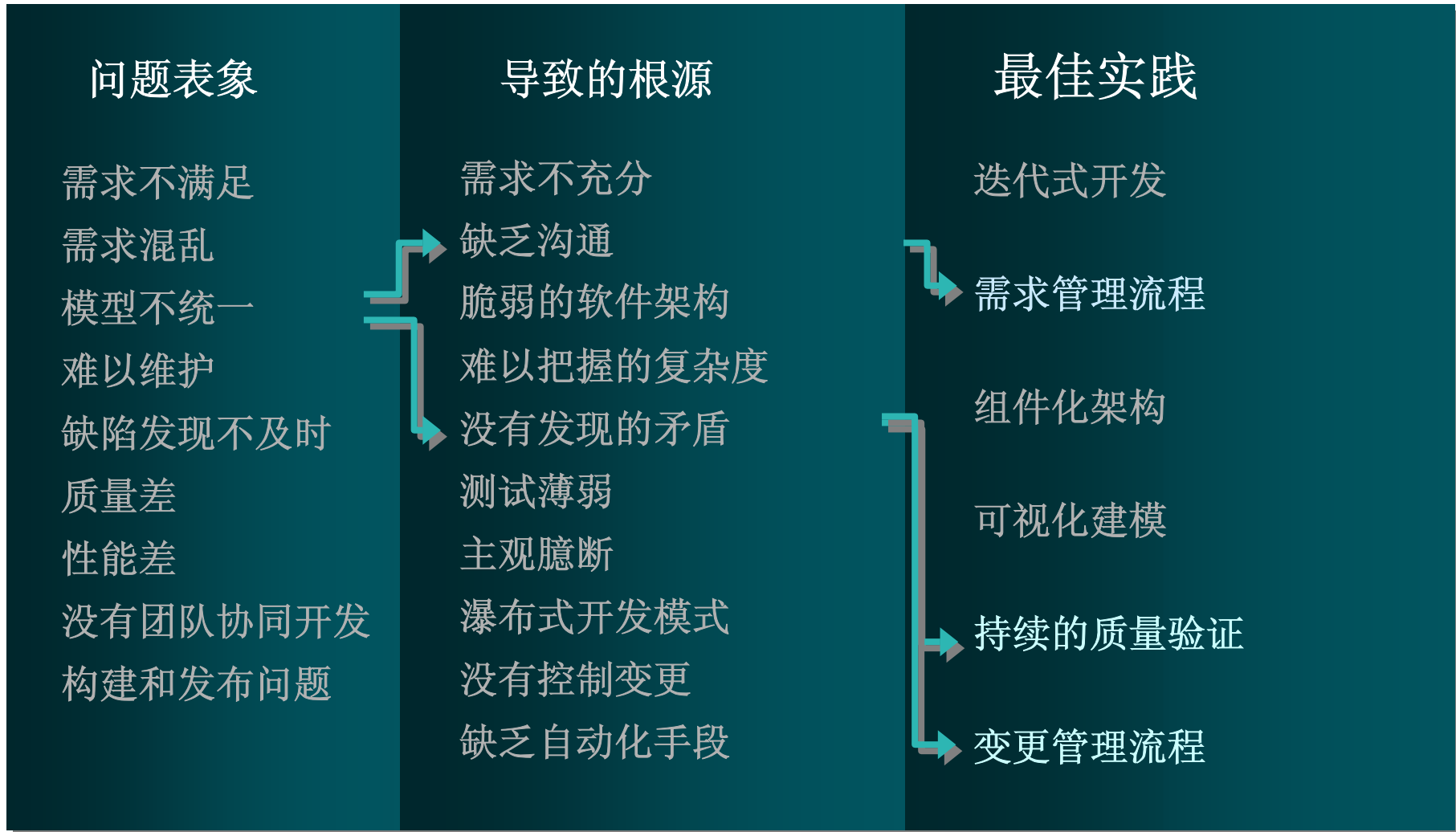
## 软件开发流程(续)



# 软件开发流程（续）



# 由表及里追述原因





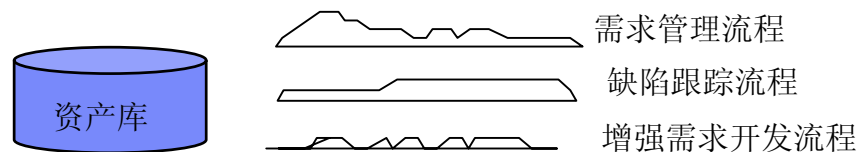
# 议程

- 软件开发的问题及需求
- **IBM的解决方案**
  - ▶ 软件开发流程
  - ▶ 软件资产库
  - ▶ UCM最佳实践
- 方案的预期收益

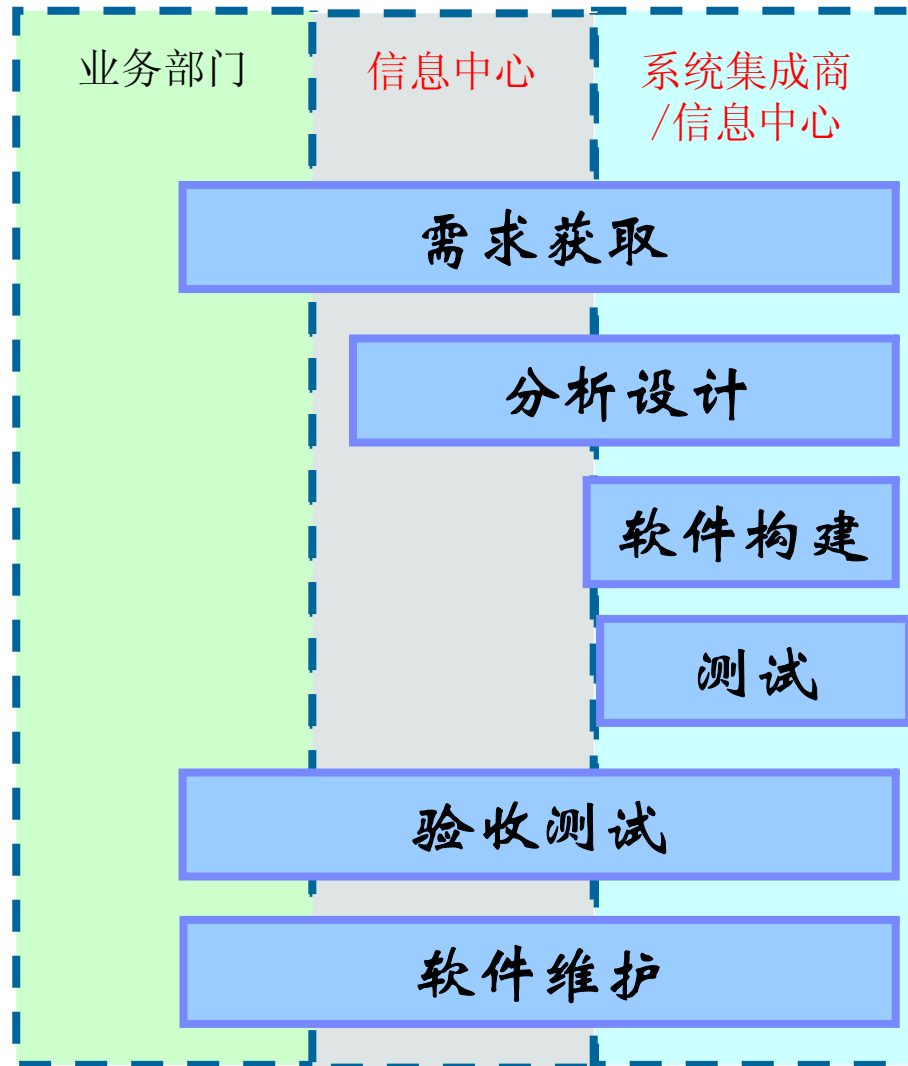


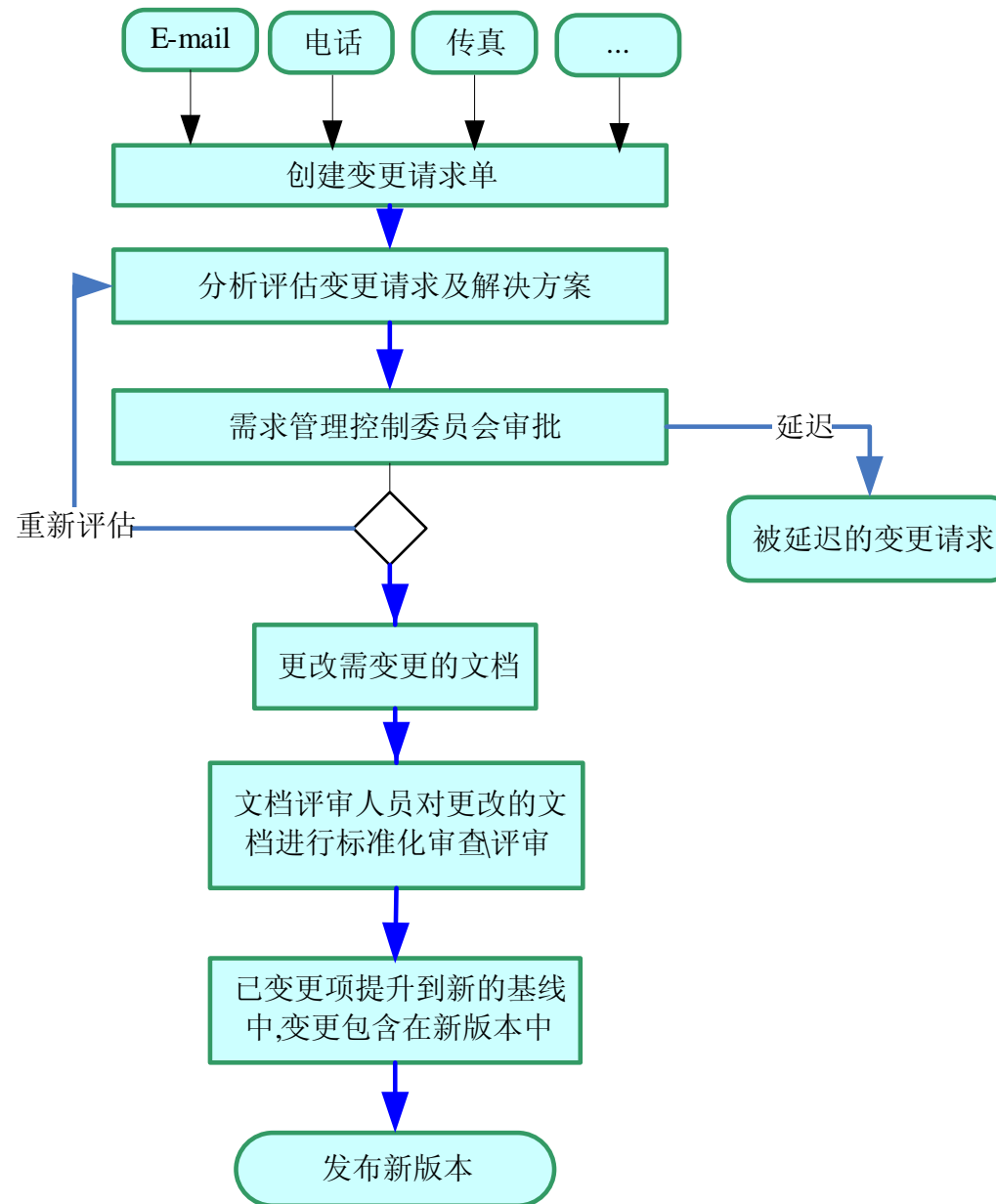
## 解决方案

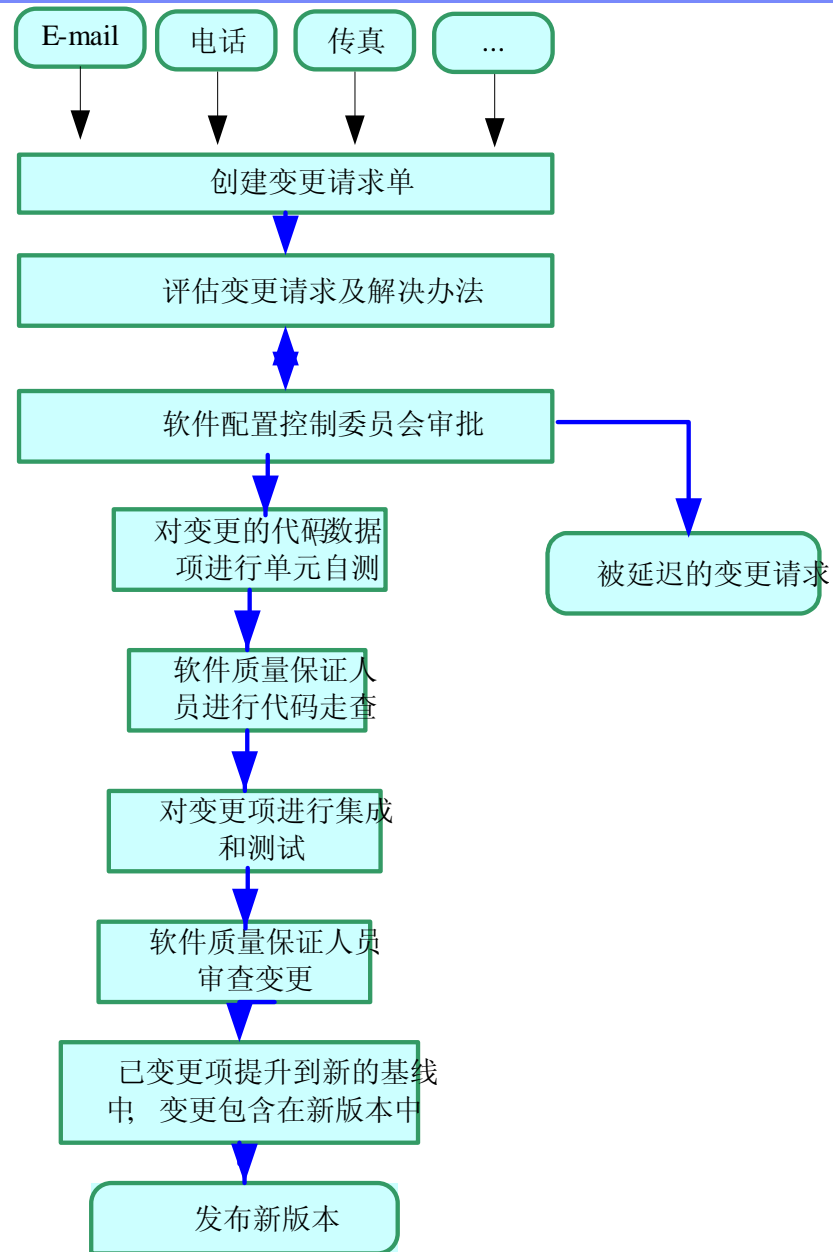
- 一个资产库
  - ▶ 软件资产库
- 三个管理流程
  - ▶ 需求管理流程（含工作处理流程）
  - ▶ 需求变更开发流程
  - ▶ 缺陷跟踪流程

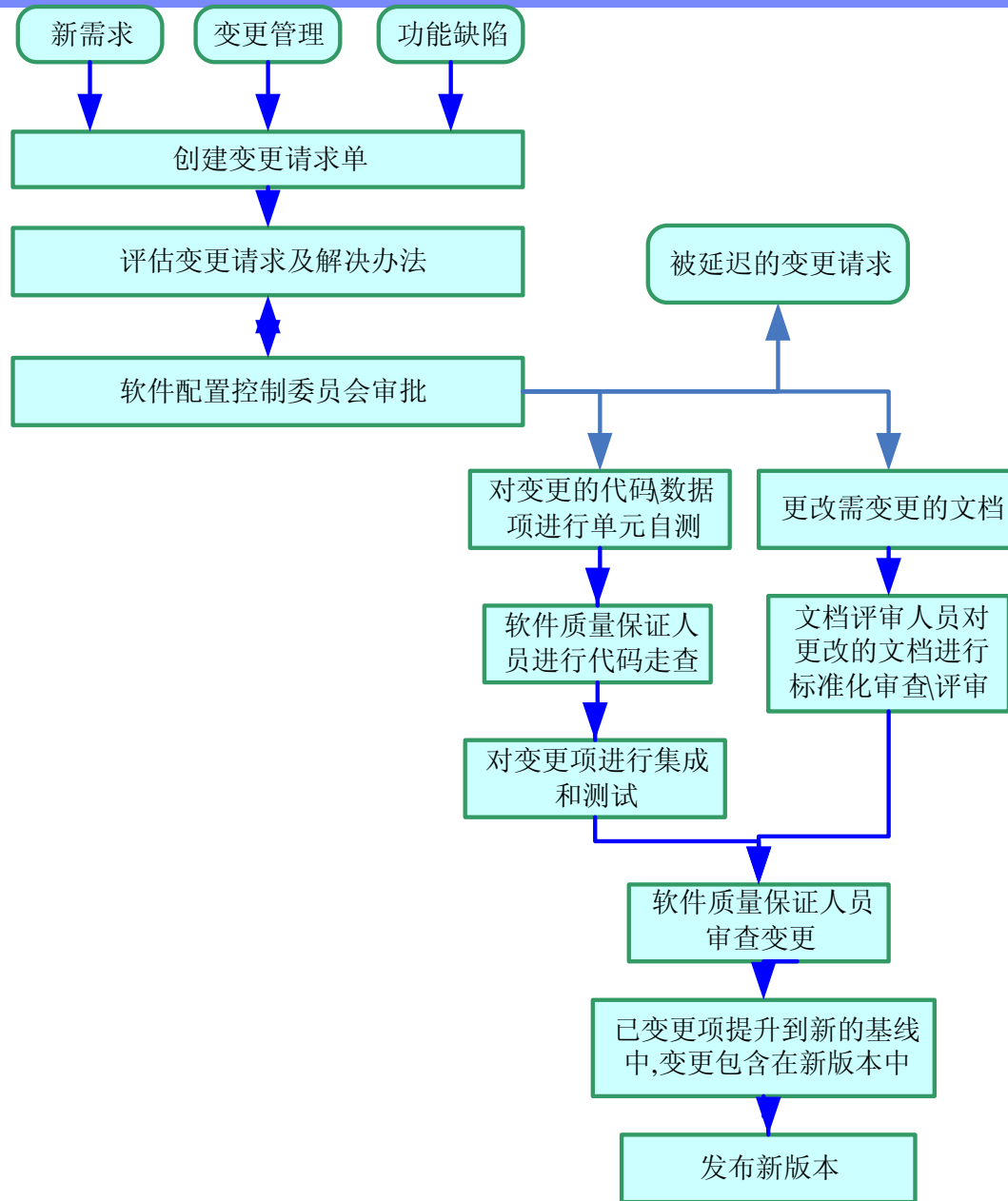


# 组织与流程









# 所有人都能方便的访问需求

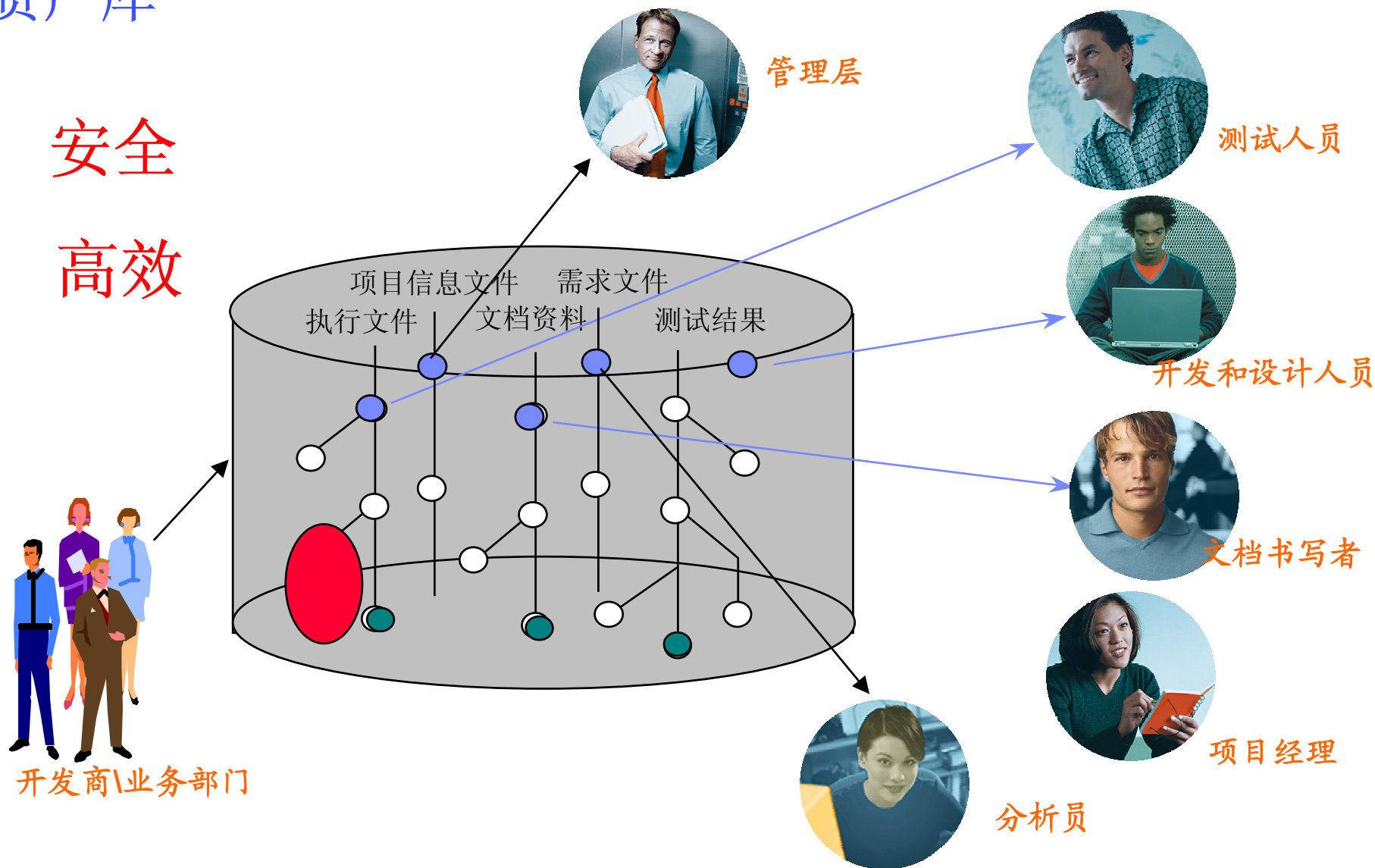


集中共享的需求管理平台



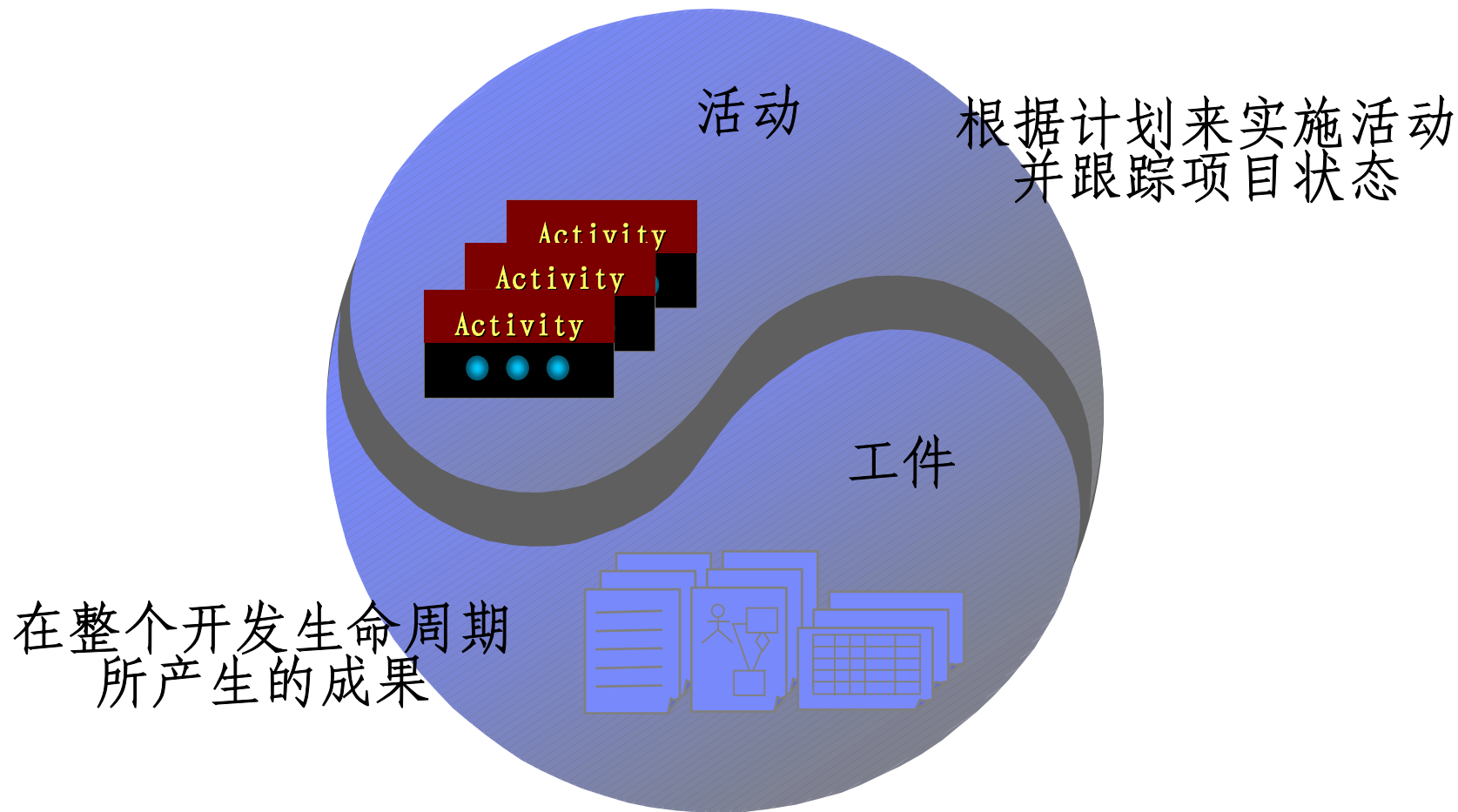
# 资产库

安全  
高效





# 统一变更管理是基于活动的



## 统一变更管理的基石

- ◆ 最强大的软件资产管理解决方案
- ◆ 超过 350,000 用户
- ◆ 并行开发自动化
- ◆ 易于使用, 高度灵活的缺陷及变更跟踪
- ◆ 超过 240,000 用户
- ◆ 强大及可定制的项目统计报告



最近六年最畅销的 **SCM** 解决方案\*

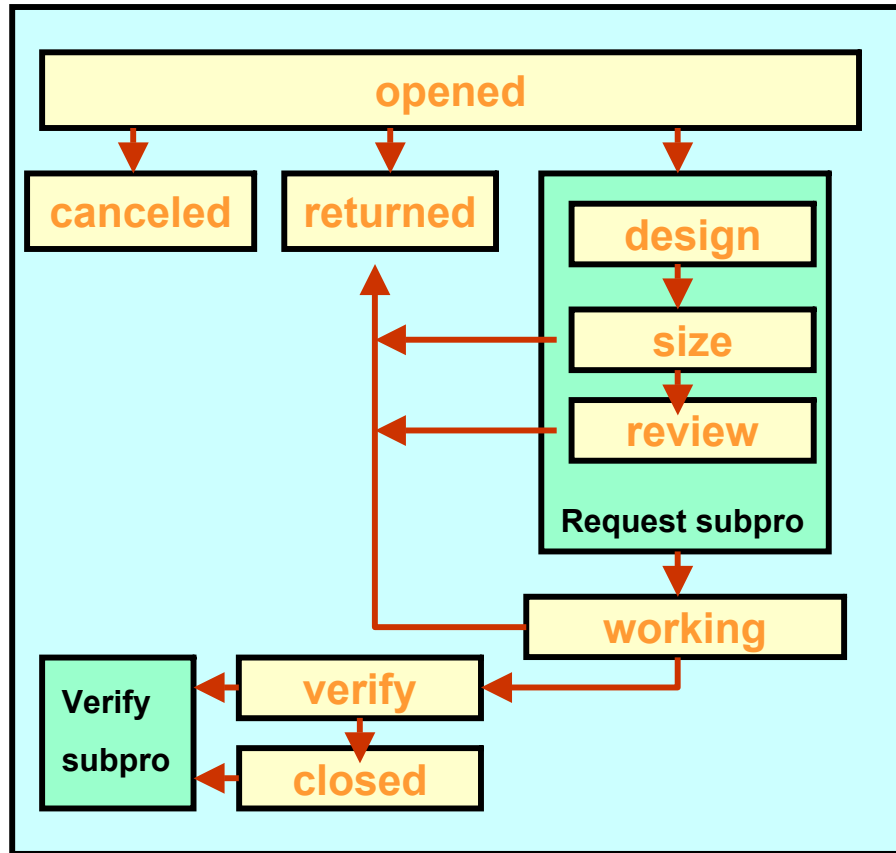
\* “SCM Tools Forecast and Analysis, 2001-2005”, IDC, June 2001

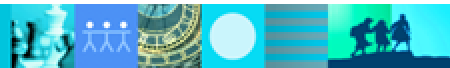
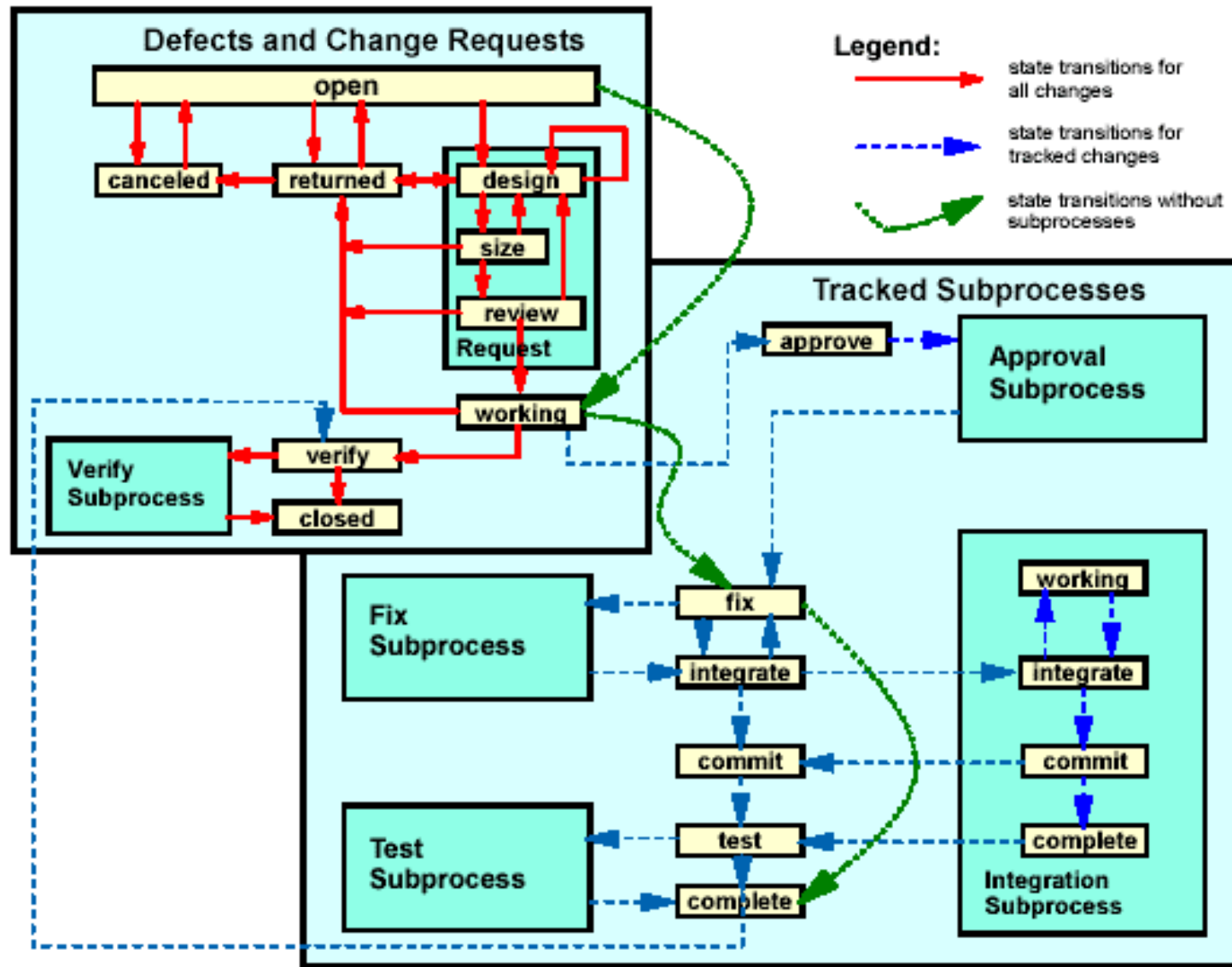
## 议程

- 软件开发的问题及需求
- IBM的解决方案
  - ▶ 软件开发流程
  - ▶ 软件资产库
  - ▶ UCM最佳实践
- 方案的预期收益



# 流程的抽象

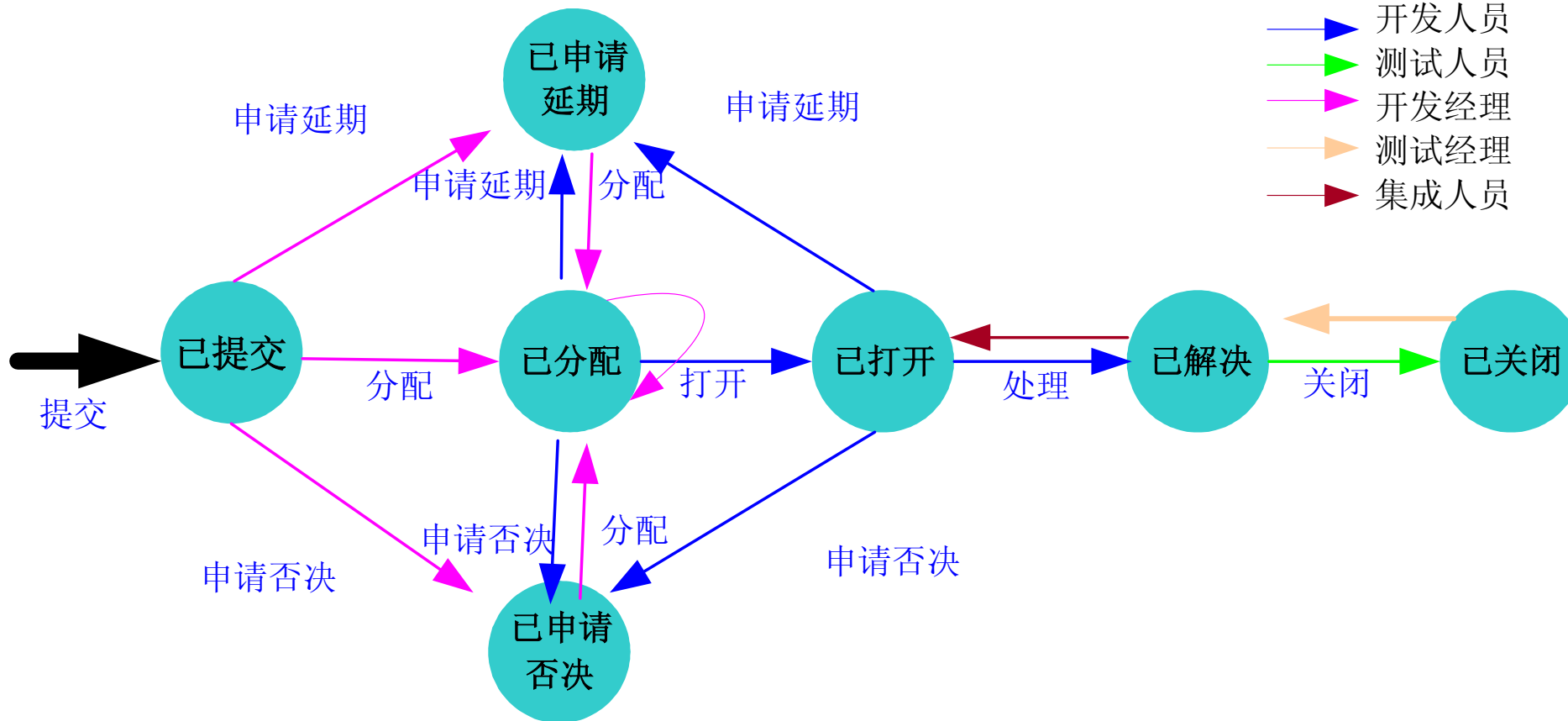


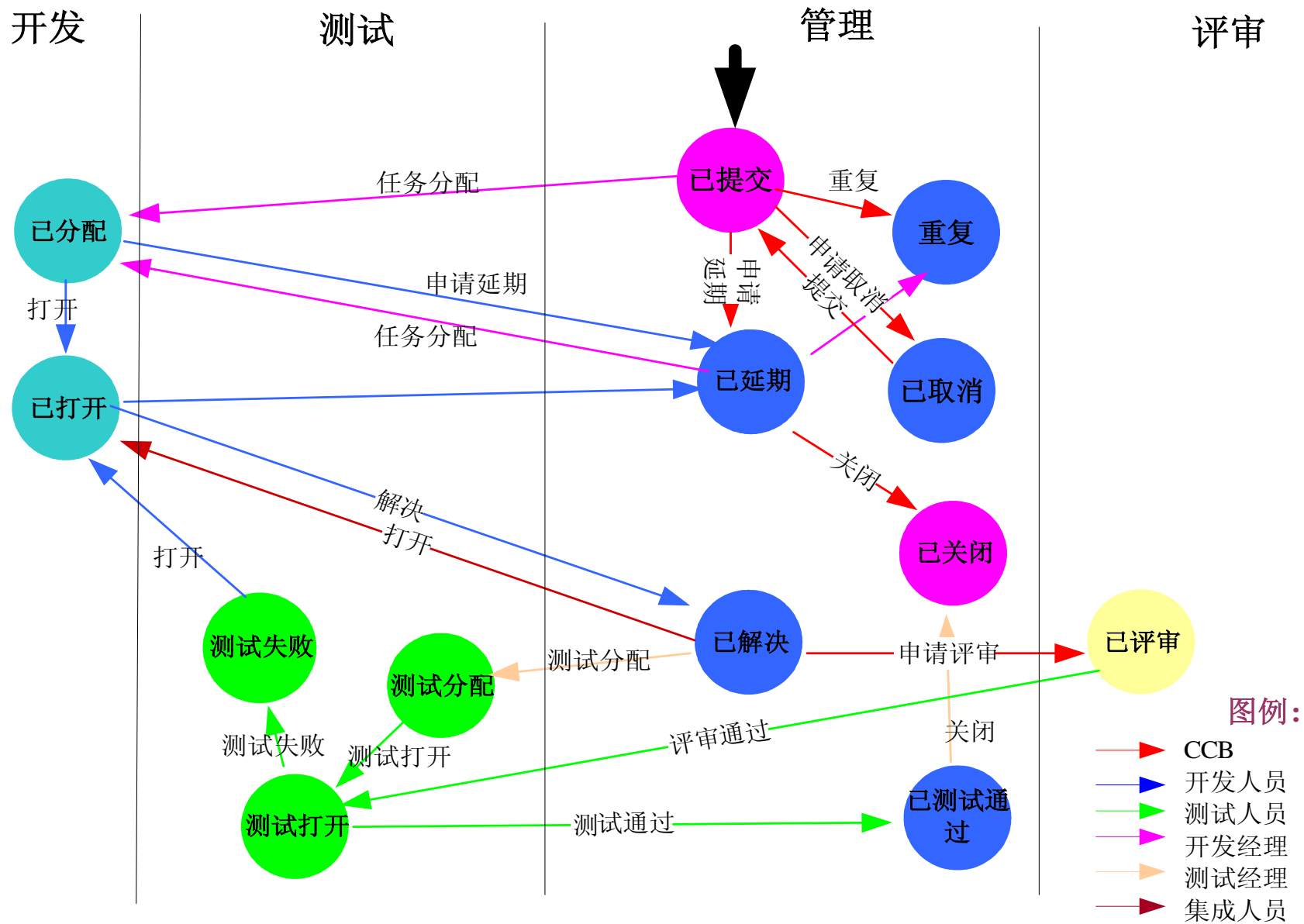


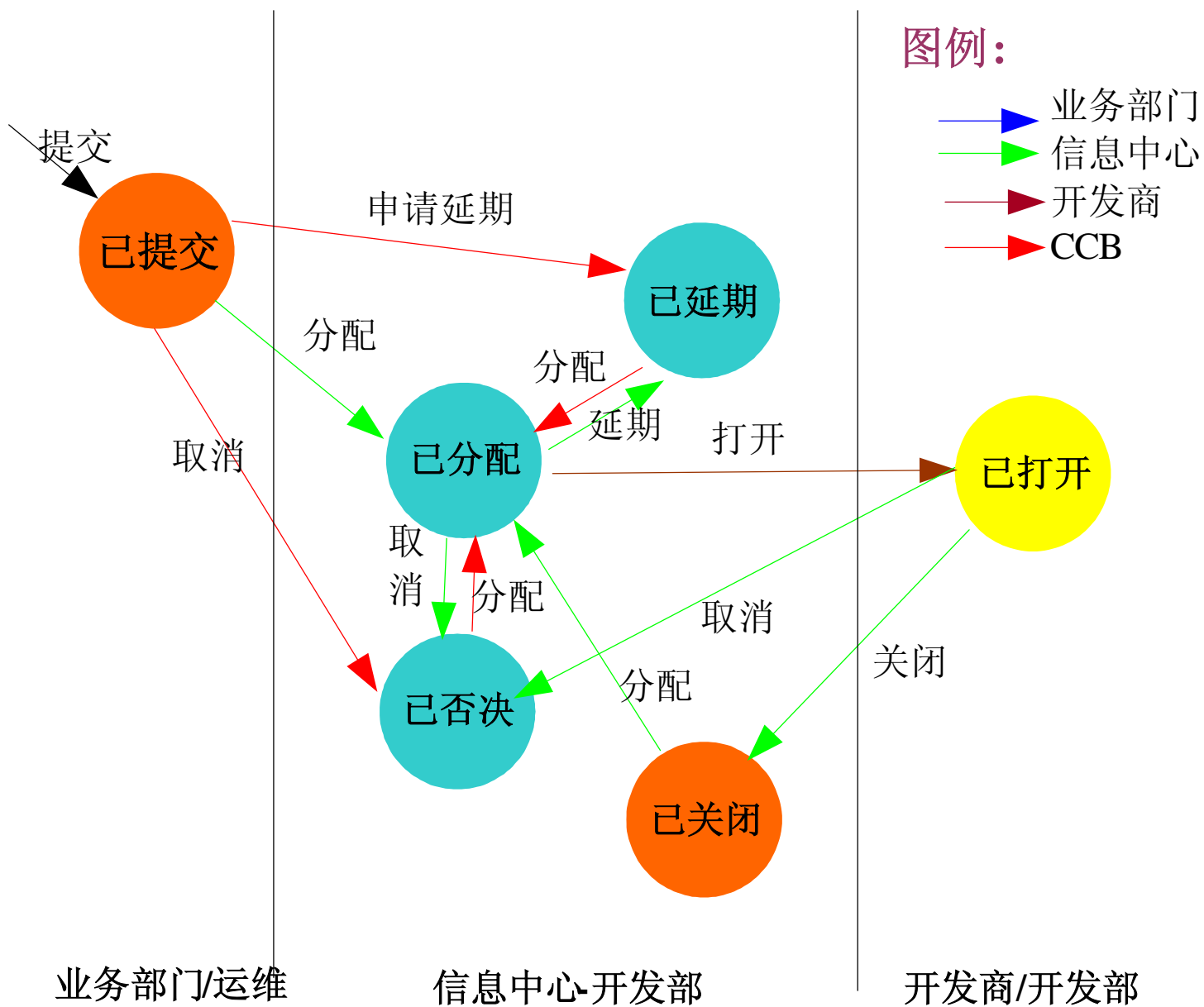
# 流程的实现

图例:

- CCB
- 开发人员
- 测试人员
- 开发经理
- 测试经理
- 集成人员









# 团队合作—统一变更管理的工作流程

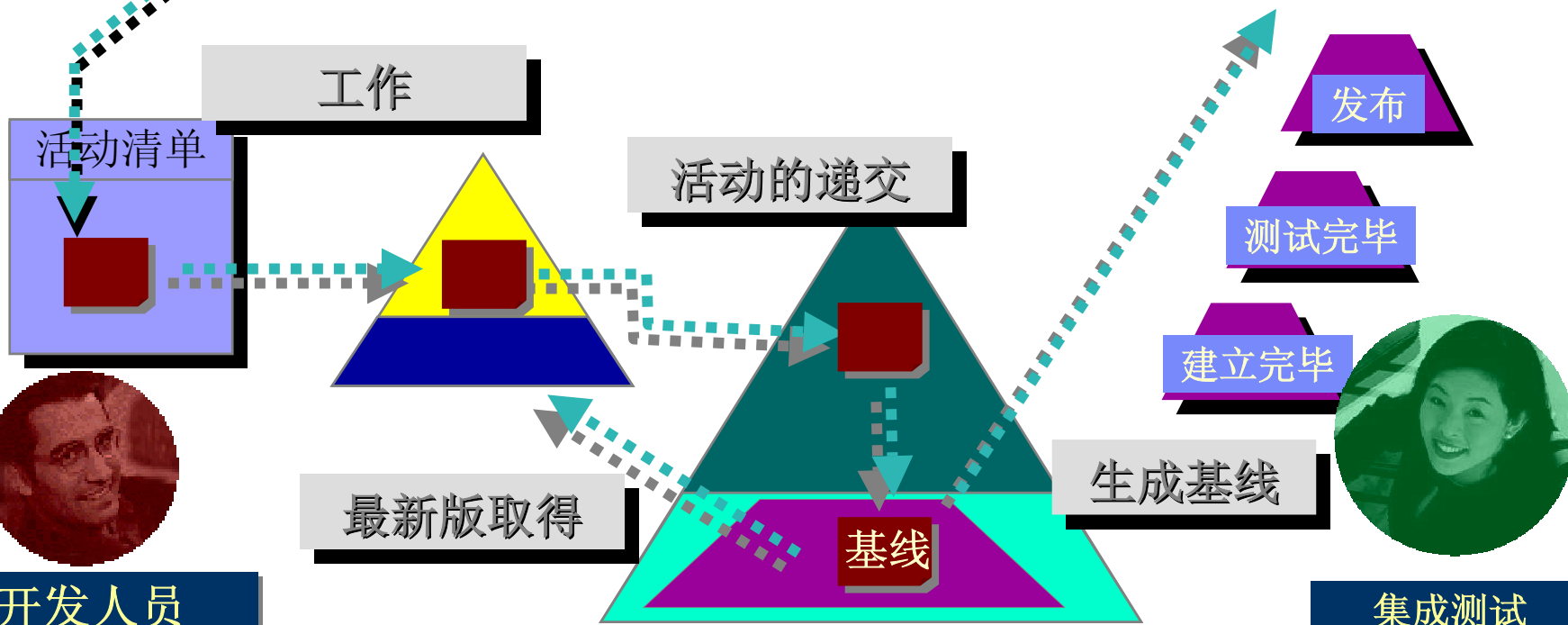


项目经理

## 活动的组合

变更请求	优先级	负责人	项目
缺陷 98	1	lgu	WebRelay
GUI按钮追加	3	Jim	WebRelay
缺陷 527	1	Jacky	WebRelay

基线的升级



开发人员

集成测试负责人



# 需求管理平台



集中共享的需求管理平台



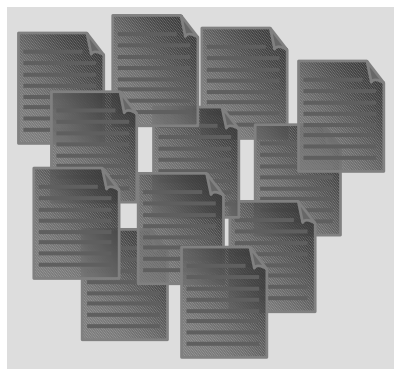
# 议程

- 软件开发的问题及需求
- IBM的解决方案
  - ▶ 软件开发流程
  - ▶ 软件资产库
  - ▶ UCM最佳实践
- 方案的预期收益



# 管理高层对象

- 更好地表示你的软件架构



- X** 文件
- X** 目录
- X** 标签
- X** 构建

## UCM Process

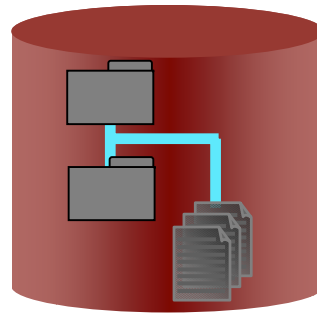
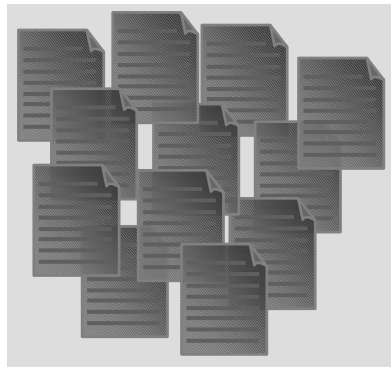
Rational ClearQuest

Rational ClearCase

直接可用的流程

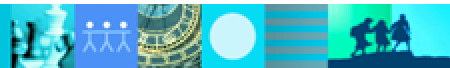
# 管理高层对象

- 更好地表示你的软件架构



**UCM Process**

- Rational ClearQuest**
- Rational ClearCase**



# 议程

- 软件开发的问题及需求
- IBM的解决方案
  - ▶ 软件开发流程
  - ▶ 软件资产库
  - ▶ **UCM最佳实践**
- 方案的预期收益



# 流程与配置库配合的策略

### Rational ClearQuest

管理活动

- ◆ To Do Lists
- ◆ Workflow

### ClearQuest:有组织的活动

Request	Priority	Owner
Special Promo	1	Terry
Bug 527	2	Sandy
Add GUI button	2	Kim

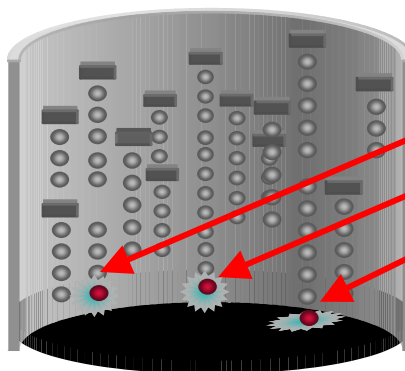
### Rational ClearCase

管理工件

- ◆ Versioning: code, models, XML, HTML
- ◆ Parallel development

### 变更集

Special Promo	
a.html	V5
c.xml	V3
b.jpg	V8



将活动和工件结合在一起



## 在安全的存储库中进行工件的标识和存储

- 标识对哪些工件进行版本管理
  - ▶ 源代码
  - ▶ 对象代码, 数据
  - ▶ Web 内容
  - ▶ 在生命周期其他点上产生的工件
    - 文档和图象
    - 模型和设计规范
    - 需求和测试用例
- 存储工件的存储库必须具备可访问性, 可扩展性, 分布处理能力以及容错性



1



## 控制并审计对工件的变更

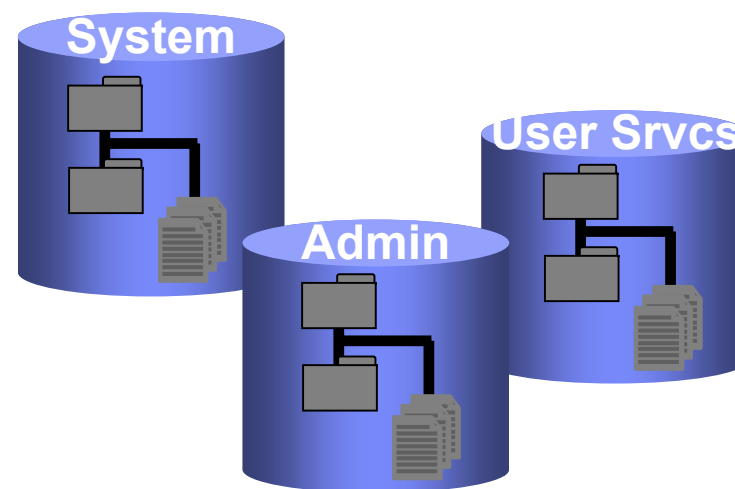
- 确定对不同类型工件的存取访问控制并控制具体的访问
- 对工件发生的变更进行审计
  - ▶ 什么样的变更?
  - ▶ 谁进行的变更?
  - ▶ 什么时候发生的变更?
  - ▶ 为什么要进行变更?



2

## 将工件组织为版本化的构件

- 使用基于构件的体系结构是RUP 中的一个最佳经验
  - ▶ 减少复杂性
  - ▶ 有利于共享和重用
- 在 SCM 中使用构件有助于将逻辑设计和物理实现紧密结合在一起
- 提供了更加智能的创建和使用 SCM 基线的机制



## 在项目里程碑创建基线

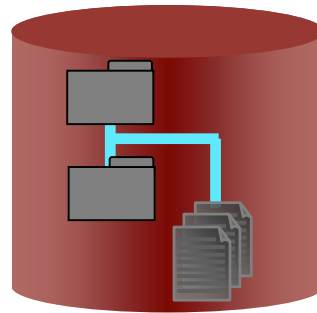
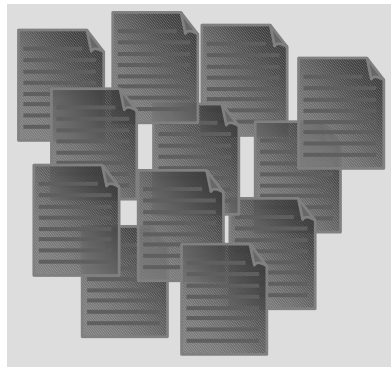
- 基线记录了一个构件中所有工件的一个单一版本
  - ▶ 将不同构件的基线组合起来可以进行更大系统的构造
- 至少应在每次迭代后创建基线，但实际上基线的创建要更频繁
- 基线的好处
  - ▶ 再现性
  - ▶ 可跟踪性
  - ▶ 报告



4

# 管理高层对象

- 更好地表示你的软件架构

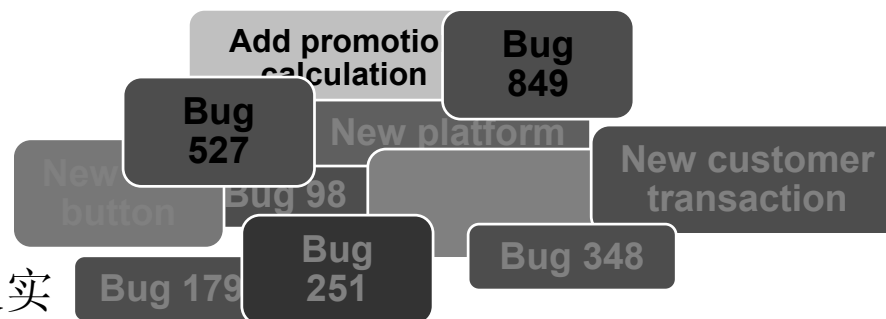


UCM Process

- Rational ClearQuest
- Rational ClearCase

## 记录并跟踪变更请求

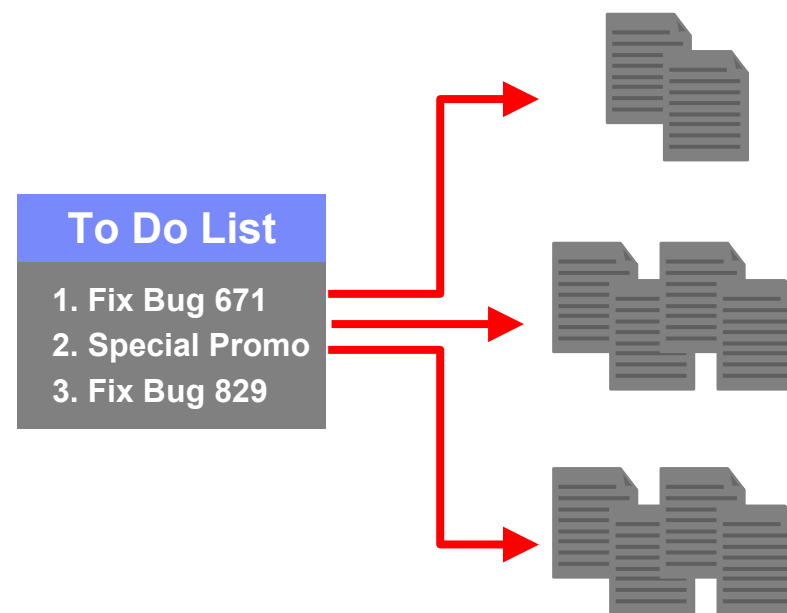
- 变更请求会有多种形式并且来自不同的地方
  - ▶ 内部及外部错误报告
  - ▶ 来自市场及工程部门的功能增强请求
  - ▶ 需求，设计，及文档变更请求
- 必须对变更请求进行管理和跟踪
- 理想地，你应该可以跟踪变更请求以及实现该请求而产生的变更



5

## 使用活动来组织和集成

- 如果变更不从逻辑上进行组织，基于文件的版本控制是相当琐碎并容易产生错误
- 活动包含了实现活动的变更集以及其他活动属性
- 使用活动进行变更组织的好处
  - ▶ 更接近自然的工作方式
  - ▶ 提供了对错误和项目管理的逻辑链接
  - ▶ 对手工工作进行了自动化



6

## 维护稳定和一致的工作空间

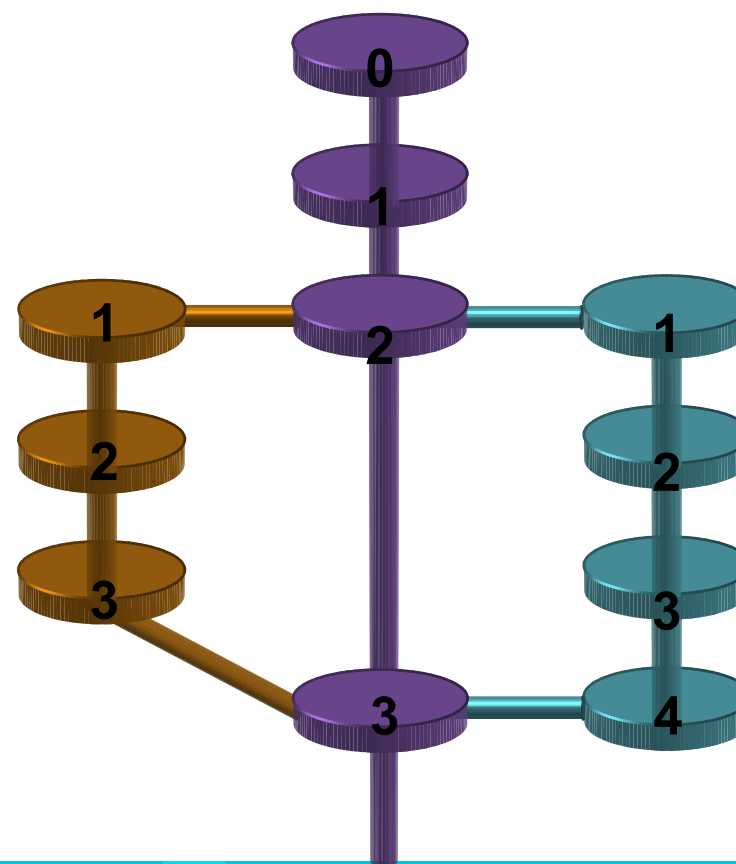
- 工作空间使用户方便地访问与给定任务相关的构件
- 用户可以均衡下面两个方面：
  - ▶ 同其他用户共享工作
  - ▶ 同其他用户的工作隔离
- 生产率同稳定性成正比



7

## 支持对构件的并行开发

- 串行开发可能是理想的，但今天这种方式既不有效也不实用
- 开发人员不应等待其他开发人员的工作完成
- 隔离的配置是进行有效并行开发的关键
- 分支是相当重要的
- 合并甚至会更重要





## 尽早和经常进行集成

有效的并行开发需要隔离



过度隔离可能会造成集成恶梦

- 项目都面临着集成和隔离的矛盾
- 在两者之间求得平衡是成功的关键
  - ▶ 开发人员应该总是可以控制自己的工作空间
  - ▶ 管理人员和集成人员必须在不牺牲个人生产率的前提下用一种可靠、可行的方式使开发人员同项目的进度保持一致
- 较晚或不经常进行集成会不利于迭代，进而引发更多风险

9

## 确保软件构建的再现性

- 构建应当是可进行全面审计的
  - ▶ 内容
  - ▶ 时间
  - ▶ 谁
- 不同构建之间的比较可以在不进行大范围调试的情况下轻易发现问题所在
  - ▶ 从上次构建以来发生了哪些变更?
  - ▶ 我们使用了相同的方式来进行系统构建吗?我们使用了不同工具或不同选项进行系统构建吗?



10

# 议程

- 软件开发的问题及需求
- IBM的解决方案
  - ▶ 软件开发流程
  - ▶ 软件资产库
  - ▶ UCM最佳实践
- 收益



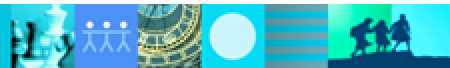
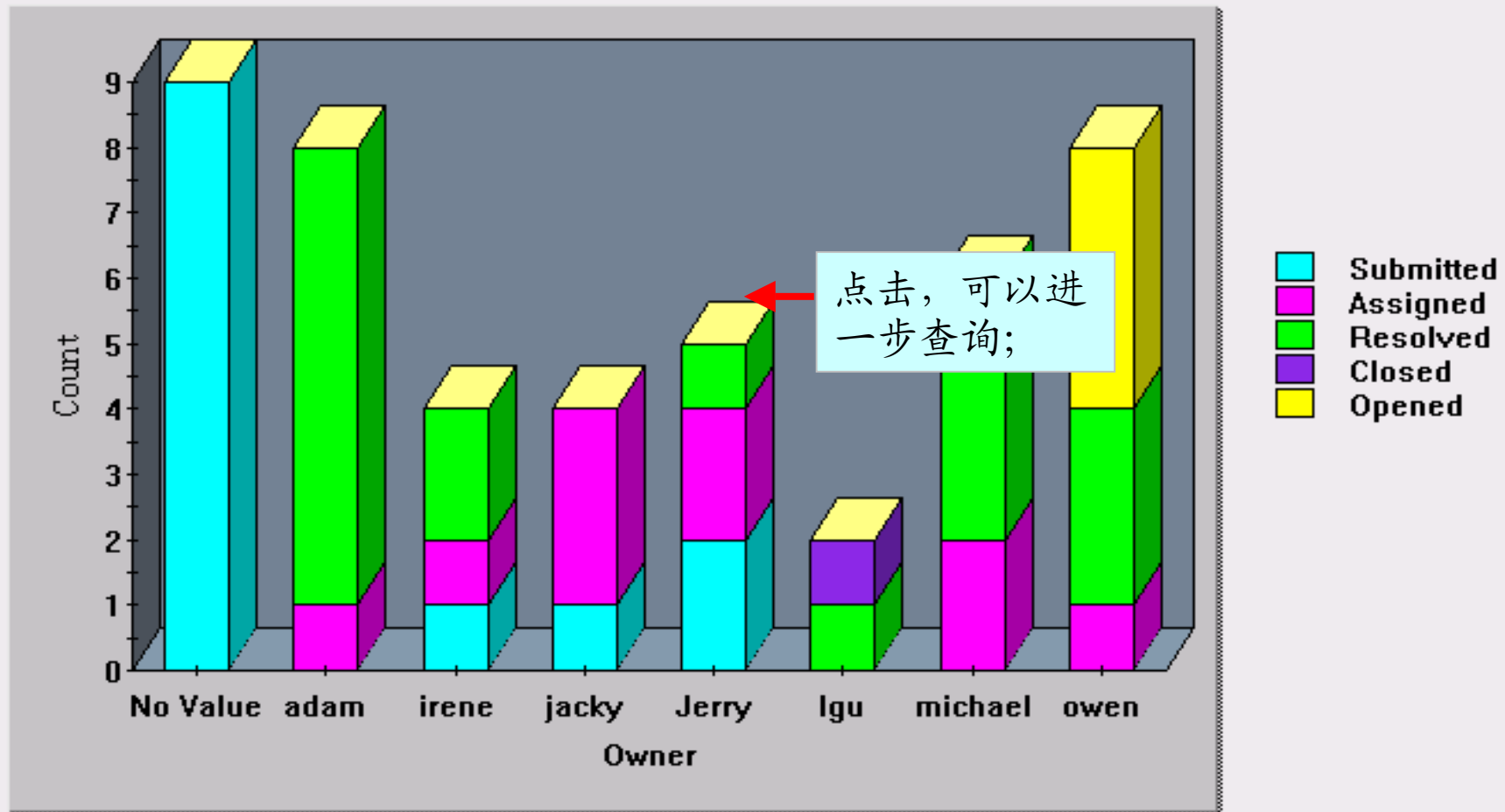
## 收益

- 安全的资产管理、清晰的范围管理
- 有序高效的开发流水线
- 准确及时的项目状态报告
- 清晰的资源分配状况报告
- 科学的项目评估
- 更准确细粒度的发布
- 需求的可追踪性

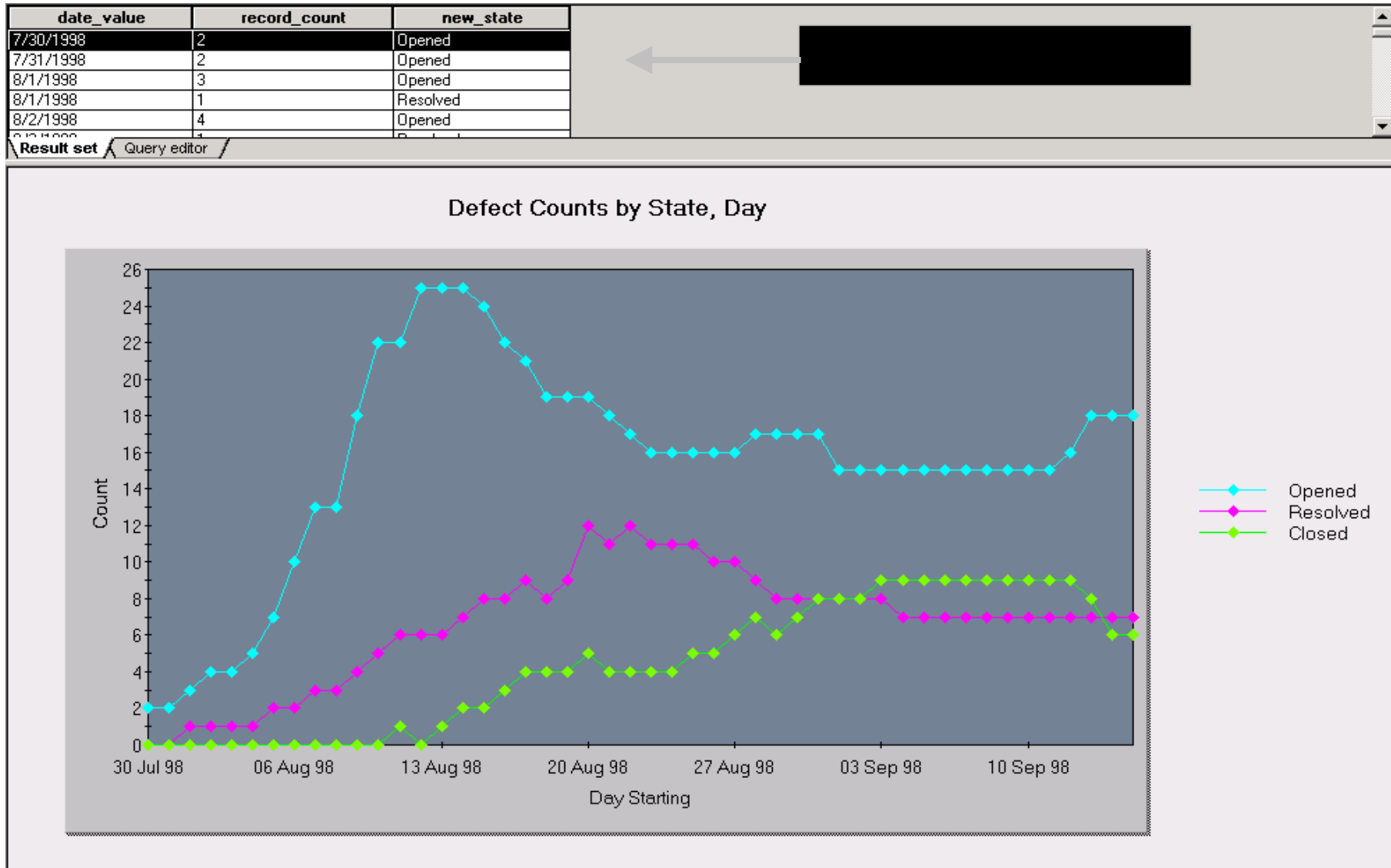


# 分布图一了解项目资源分配情况和项目质量、进度

Defects by Owner and State



# 趋势图—了解项目质量趋势

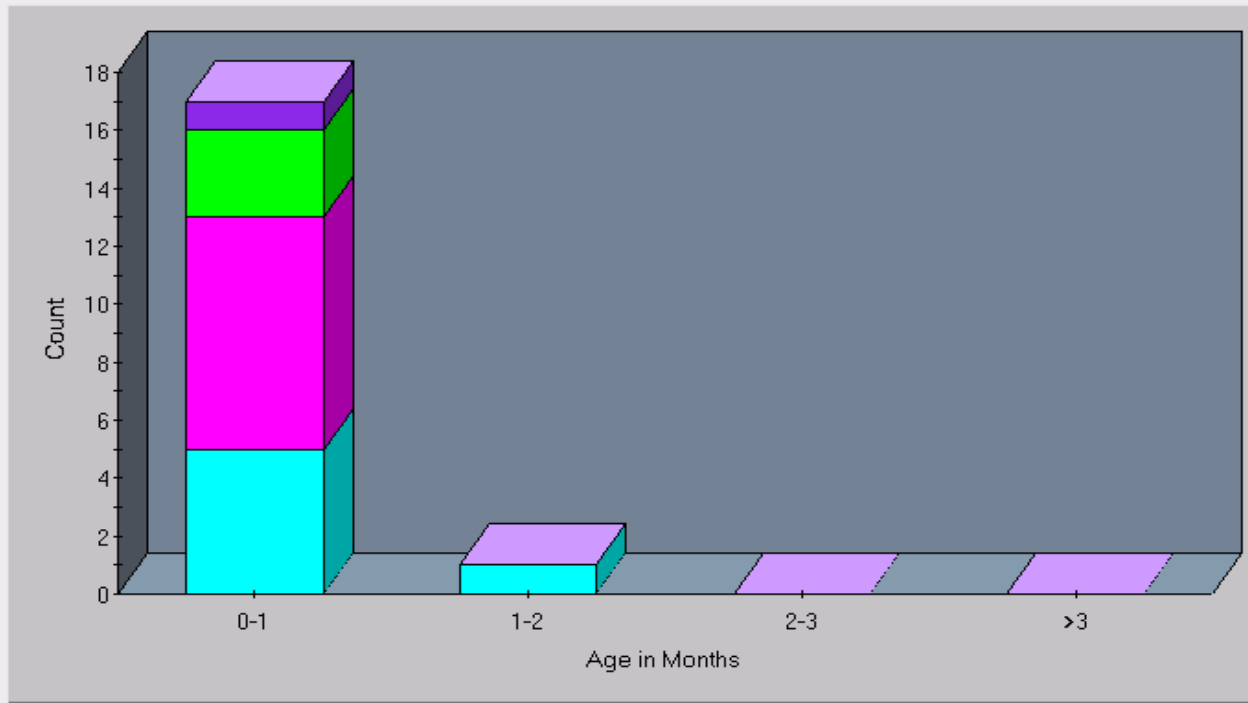


# 时长图—了解项目支持维护能力

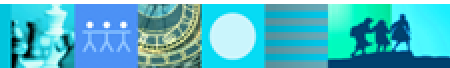
record_count	new_state	priority
8	Opened	
1	Opened	1-Resolve Immediately
5	Opened	2-Give High Attention
3	Opened	3-Normal Queue

Result set / Query editor /

Open/Priorities (4) 30 Day



- Opened;2-Give High Attention
- Opened
- Opened;3-Normal Queue
- Opened;1-Resolve Immediately



# 深入查询 (Drill Down)

Rational ClearQuest - [SAMPL : Training (Chart (Defect))]

File Edit View Chart Actions Query Window Help

Run Chart New Defect

Count	Owner	Priority
24		
6	admin	1-Resolve Immediately
4	engineer	1-Resolve Immediately
3	engineer	2-Give High Attention
9	engineer	3-Normal Queue
1	lead	1-Resolve Immediately
4	lead	2-Give High Attention

Result set Query editor

### Active Priority Defects by Owner

Click "Drill Down", you can further "Drill Down"

- 1-Resolve Immediately
- 2-Give High Attention
- 3-Normal Queue

Ready Record: 1 Count:





## 案例1:某银行协作开发模式-需求

- 案例简介：某金融企业信息中心组织机构下设置开发部、运维部。信息中心要和开发商甲、开发商乙一起协同开发一个内容管理系统。开发阶段主要由开发部协同两个开发商进行开发，开发完成后，由运维负责维护。开发商甲、乙和开发部分别负责其中不同的模块的开发任务，总体协调由开发部，开发商甲、乙相互独立，但系统少量的交互。
- 预期目标：
  - ▶ 建立严格的安全访问控制机制；建立安全的软件资产保护体系。
  - ▶ 建立统一、流畅的交流协作开发平台，减少沟通成本。
  - ▶ 建立三方协同工作流程，使整个开发有序、高效运转
  - ▶ 及时、准确、清晰地勾画出项目进展情况（新需求开发进度、bug修改进度、需求变更情况）。
  - ▶ 为运维部门的发布定义一个清晰的统一的入口，协调和运维的工作流程。

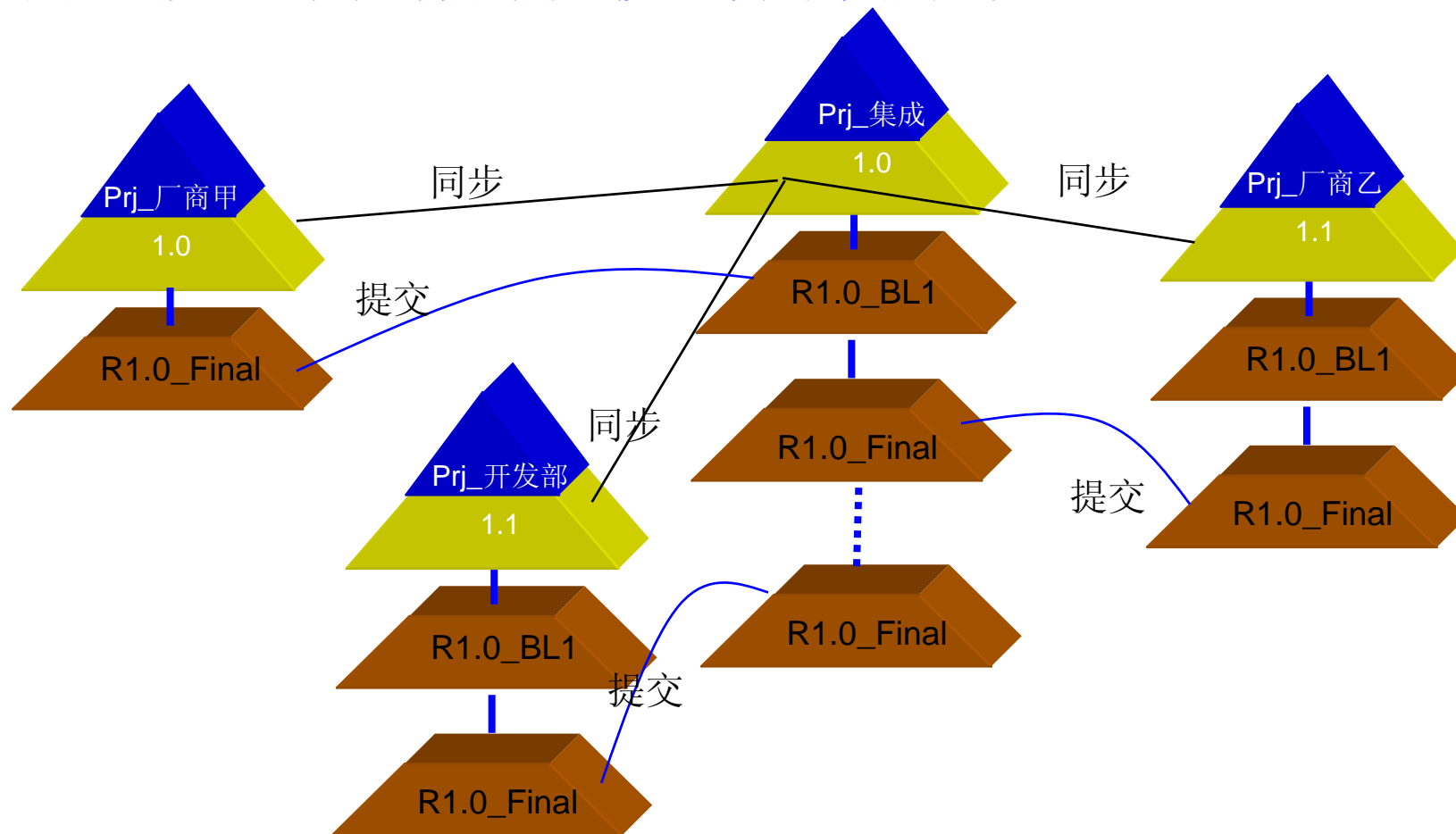


## 案例1:某银行协作开发模式-解决方案

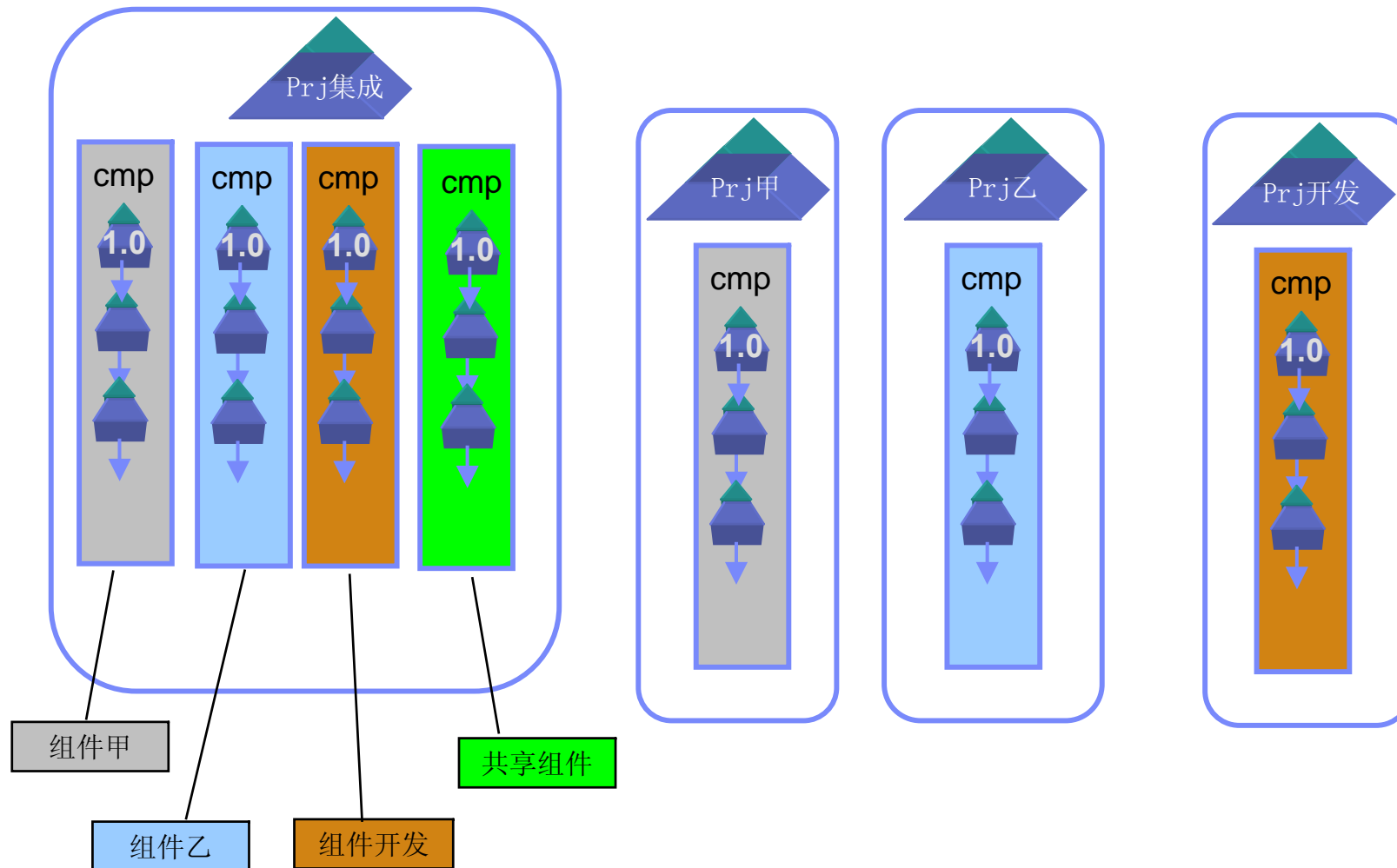
- 步骤:
  - ▶ 1.利用ClearCase和ClearQuest构建UCM环境.
  - ▶ 2.采纳UCM缺省的工作流程
  - ▶ 3.倒入现有成果数据



# 案例1:某银行协作开发模式-解决方案



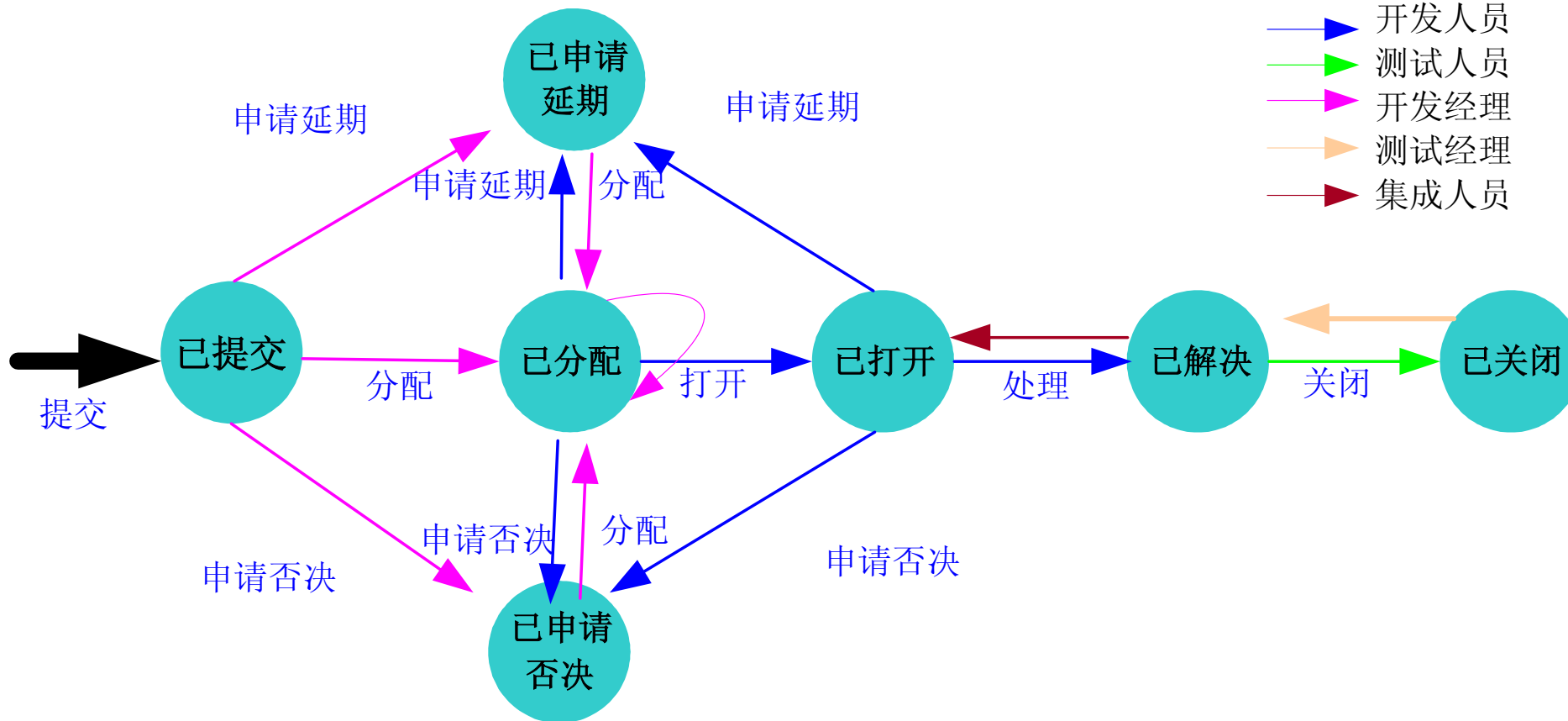
# Clear Case中组件关系



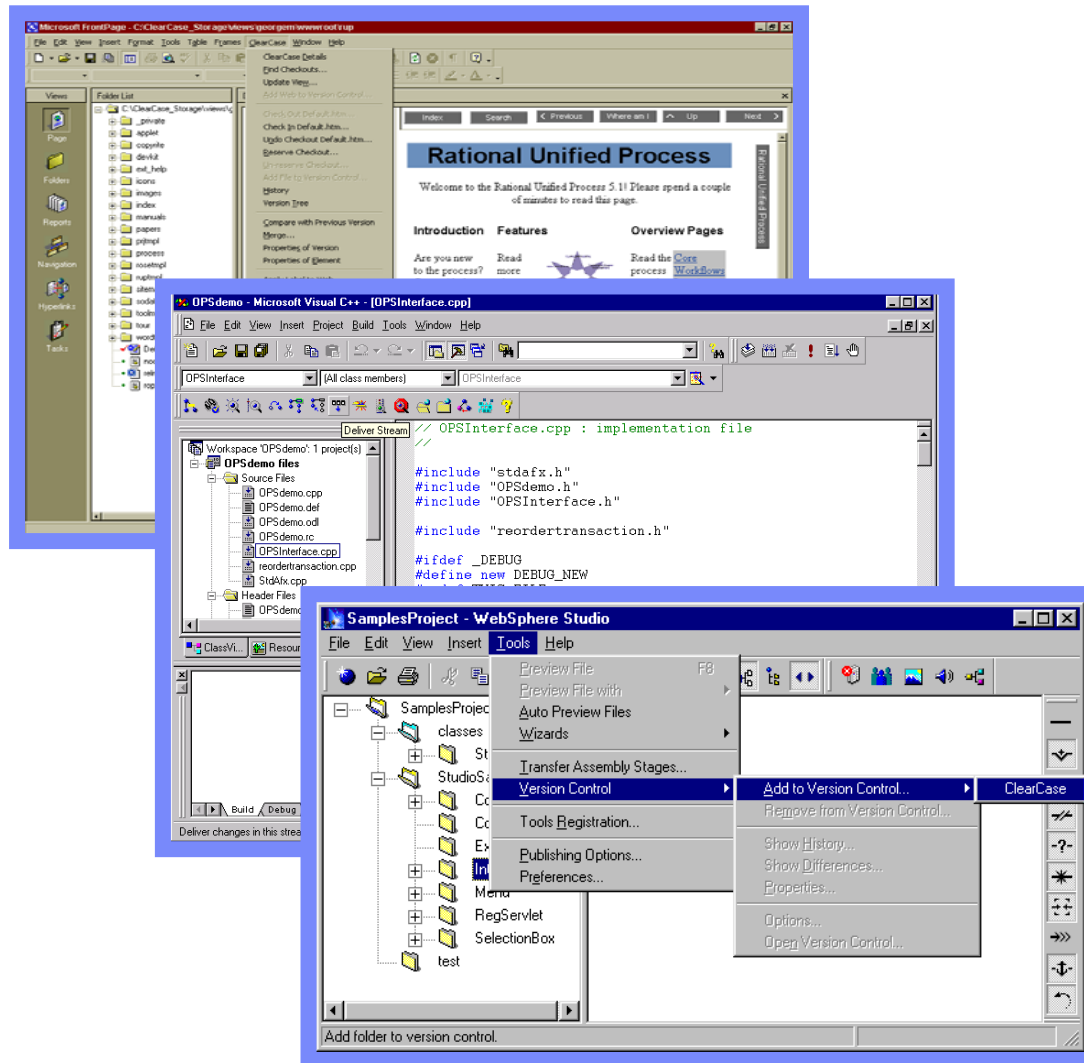
# 流程的实现

图例:

- CCB
- 开发人员
- 测试人员
- 开发经理
- 测试经理
- 集成人员



## 易于采用：同流行的IDE集成

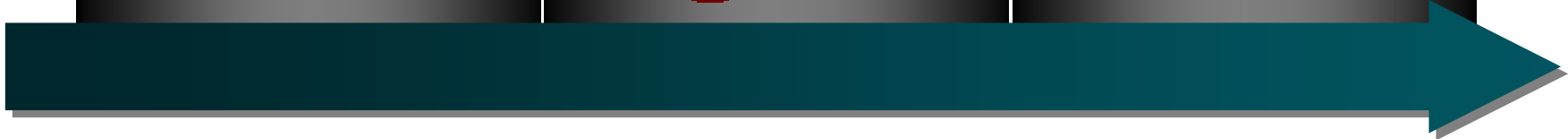
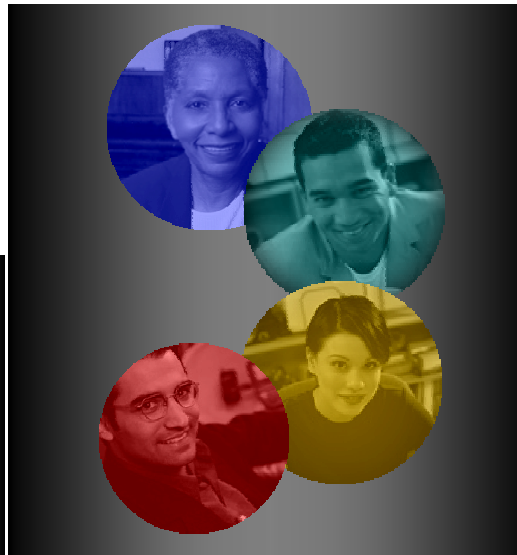


### ClearCase Integrations

- Allaire ColdFusion
- Bluestone SapphireWeb
- HAHT HahtSite
- IBM WebSphere, VisualAge for Java
- Netscape Application Builder
- Microsoft Visual Studio
- Microsoft Visual FrontPage and Visual InterDev
- Sun NetDynamics
- Symantec Visual Café

## 可部署于任何地点、任何规模的团队

- **ClearCase MultiSite**
- **ClearQuest MultiSite**



## 国内外部分配置管理工具用户列表

- 国内金融行业（银行、证券、保险等）
  - ▶ 中国建设银行研发中心（北京、上海、武汉、厦门、成都、广州）
  - ▶ 中国工商银行开发中心（珠海、广州）
  - ▶ 中国银行软件开发中心（北京，深圳）
  - ▶ 中国人民银行清算中心
  - ▶ 中国人民银行金融电子化开发公司
  - ▶ 中国交通银行开发中心（上海）
  - ▶ 中国进出口银行
  - ▶ 上海证券交易所
  - ▶ 平安/合众/PICC保险
  - ▶ 美国友邦保险有限公司上海分公司
  - ▶ ...
- 国外前11大金融服务机构的9个
  - ▶ Citigroup
  - ▶ Fannie Mae
  - ▶ Morgan Stanley Dean Witter
  - ▶ Merrill Lynch
  - ▶ American Express
  - ▶ Household International
  - ▶ Bear Stearns
  - ▶ Capital One Financial
  - ▶ Charles Schwab
- 国外前6大银行的4家
  - ▶ Bank of America
  - ▶ J.P Morgan Chase
  - ▶ Wells Fargo
  - ▶ FleetBoston Financial
- 系统集成/电信制造商/电信运营商/政府等
  - ▶ 华为科技
  - ▶ 中兴通讯
  - ▶ 大唐移动
  - ▶ 普天集团
  - ▶ 辽宁/河北/江苏/浙江移动
  - ▶ 北京网通/联通
  - ▶ 国家质检局
  - ▶ 中国民航信息集团
  - ▶ 神州数码/东软/用友软件/中软/中创
  - ▶ ...
- 外企驻中国研发中心
  - ▶ IBM研发中心
  - ▶ 爱立信Ericsson
  - ▶ Motorola
  - ▶ 西门子Siemens
  - ▶ Agilent
  - ▶ 朗讯Lucent
  - ▶ 北电Nortel
  - ▶ 诺基亚Nokia
  - ▶ 菲利普Philip
  - ▶ 阿尔卡特Alcatel
  - ▶ Cellon
  - ▶ 明基BenQ
  - ▶ 通用GE
  - ▶ 三星SamSung
  - ▶ ...





# IBM Rational的使命 - 保证客户成功!

