

SDCC 2017 | 上海

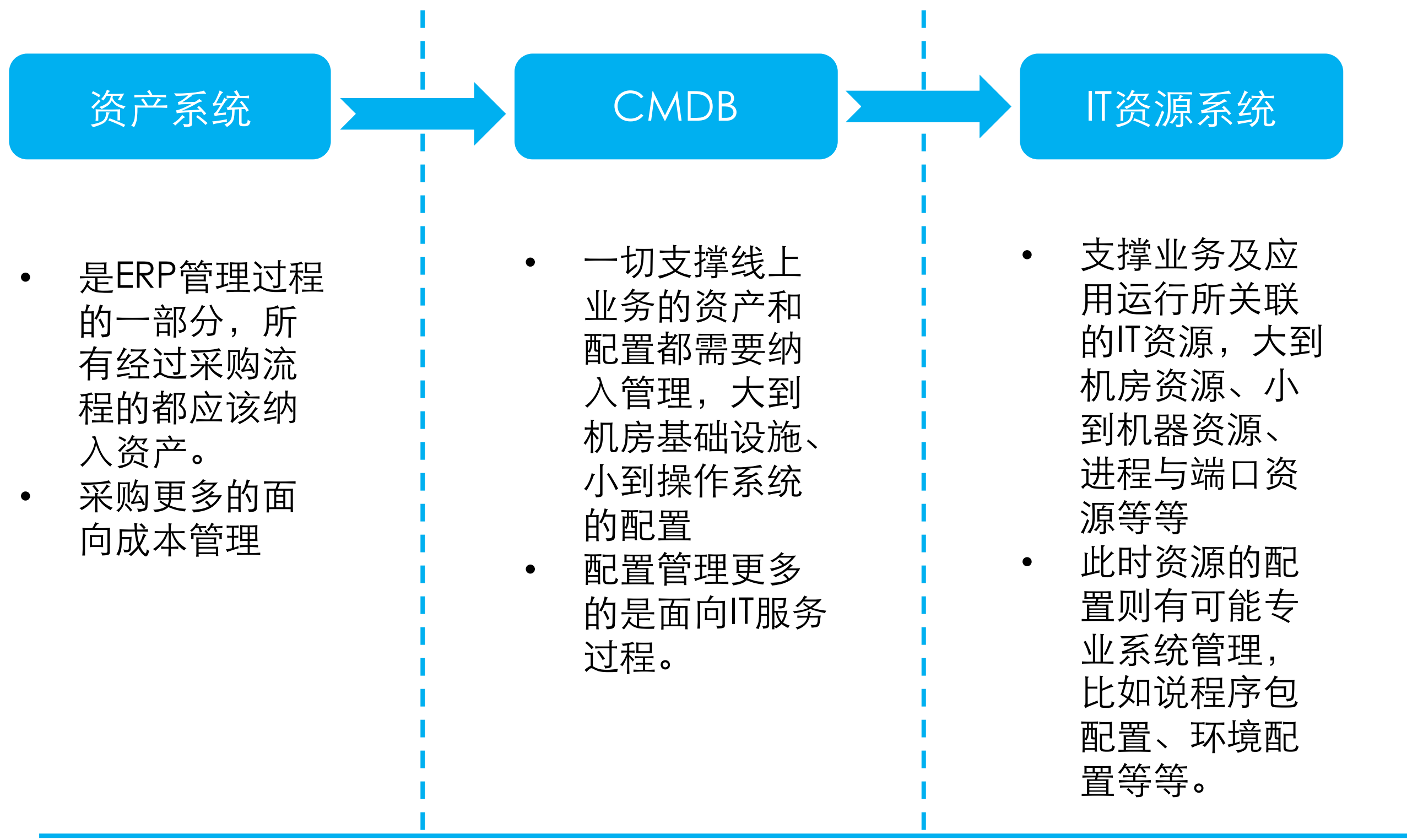
互联网运维开发实战峰会

CSDN

CMDB，从资产到资源的转变

——来自国信证券的实际案例总结

传统CMDB的问题与制约



范围逐渐缩小，从离线到在线、从资产到业务的过程



- 代码的配置管理SCM。聚焦在源代码的整个管理过程。
- 应用的配置管理ACM。聚焦在应用相关的配置，一般由应用包产生。从管理模式上来说，该配置可以升级成环境的配置管理。
- 环境的配置管理ECM。系统级的环境配置管理，比如说系统的内核调优参数

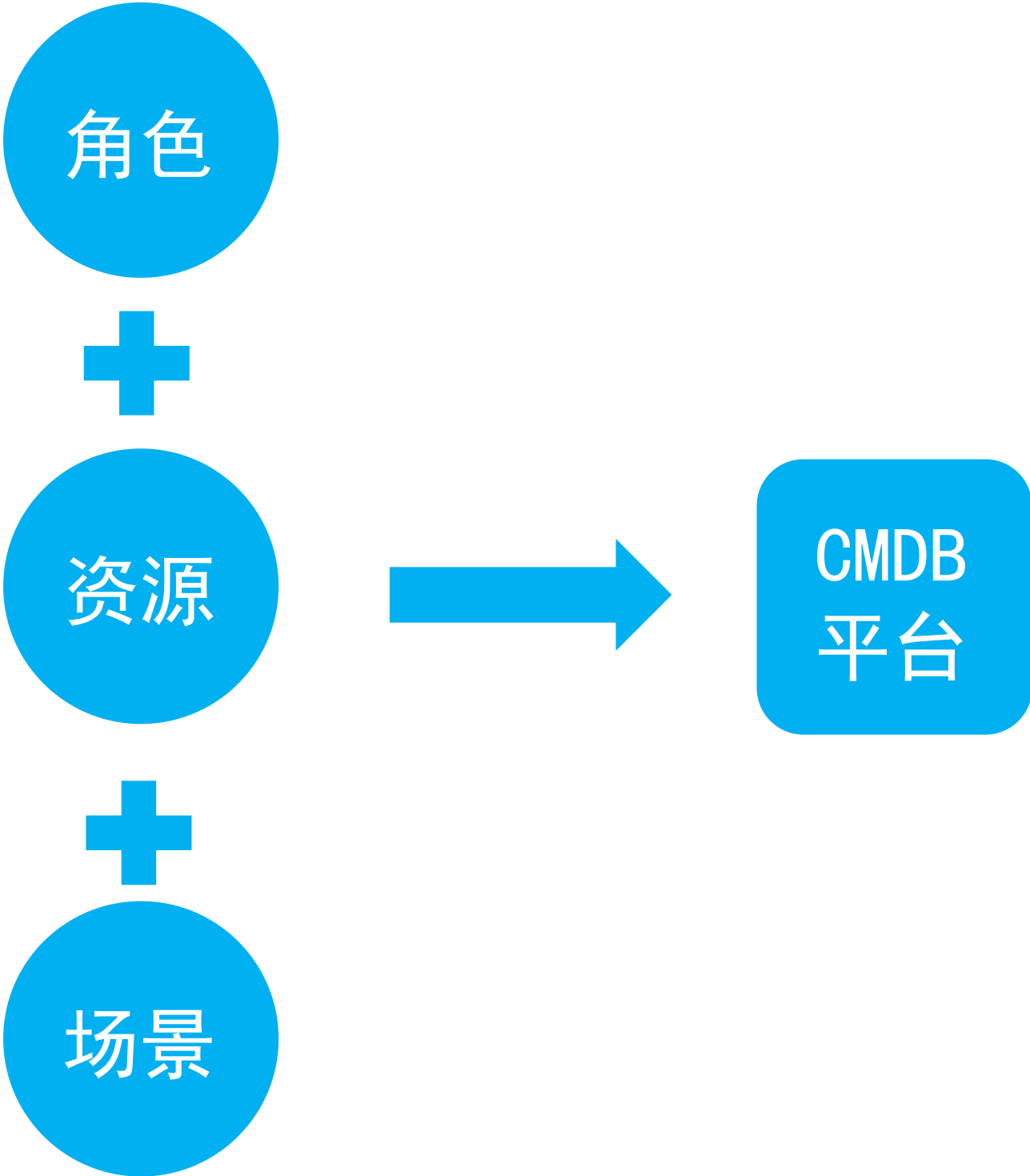
资产管理	配置管理
目的:管理资产成本、合同和使用情况	目的:作为基础为 ITIL 提供基础的模型
价值:更低的资产总拥有成本, 减少购买	价值:面向业务管理, 保证业务服务的可靠 性、质量, 通过关联的其他 ITIL 流程来保证
资产:可以被基于合同跟踪的 IT 组件	配置项:为了其运维角度而被管理的逻辑和 物理资源
区别:不可能作为配置项来管理的资产包括在仓库中的显示器, 打印机、服务器等等	区别:不可能作为资产来管理的配置项包括 客户化的 java 组件、流程、服务等等.
关系:资产之间的关系被维护以用于回收流程 。	关系:配置项之间的关系被维护以评估变更 风险、分析原因、评估服务影响 。

表现	描述
把CMDB当成一个数据库来看待	认为CMDB就是一个数据库，里面存放的就是一些配置信息，不考虑其上的管理流程
CMDB的管理方法论没有统一	CMDB的管理方法论，自动化是一方面，但也要关注流程的作用
认为CMDB的管理粒度越细越好	越细在人工、无场景的情况下，失败的可能性就越大
CMDB是一次建设就可以完善的	CMDB是数据，是一个持续完善、动态变化的过程
缺少领导的支持	因为CMDB的建设一个全员建设和参与的过程，必须要领导深度支持
你的配置项缺少Owner	缺少Owner的配置项维护是非常失败的，每个配置项必须落实到具体的责任人。

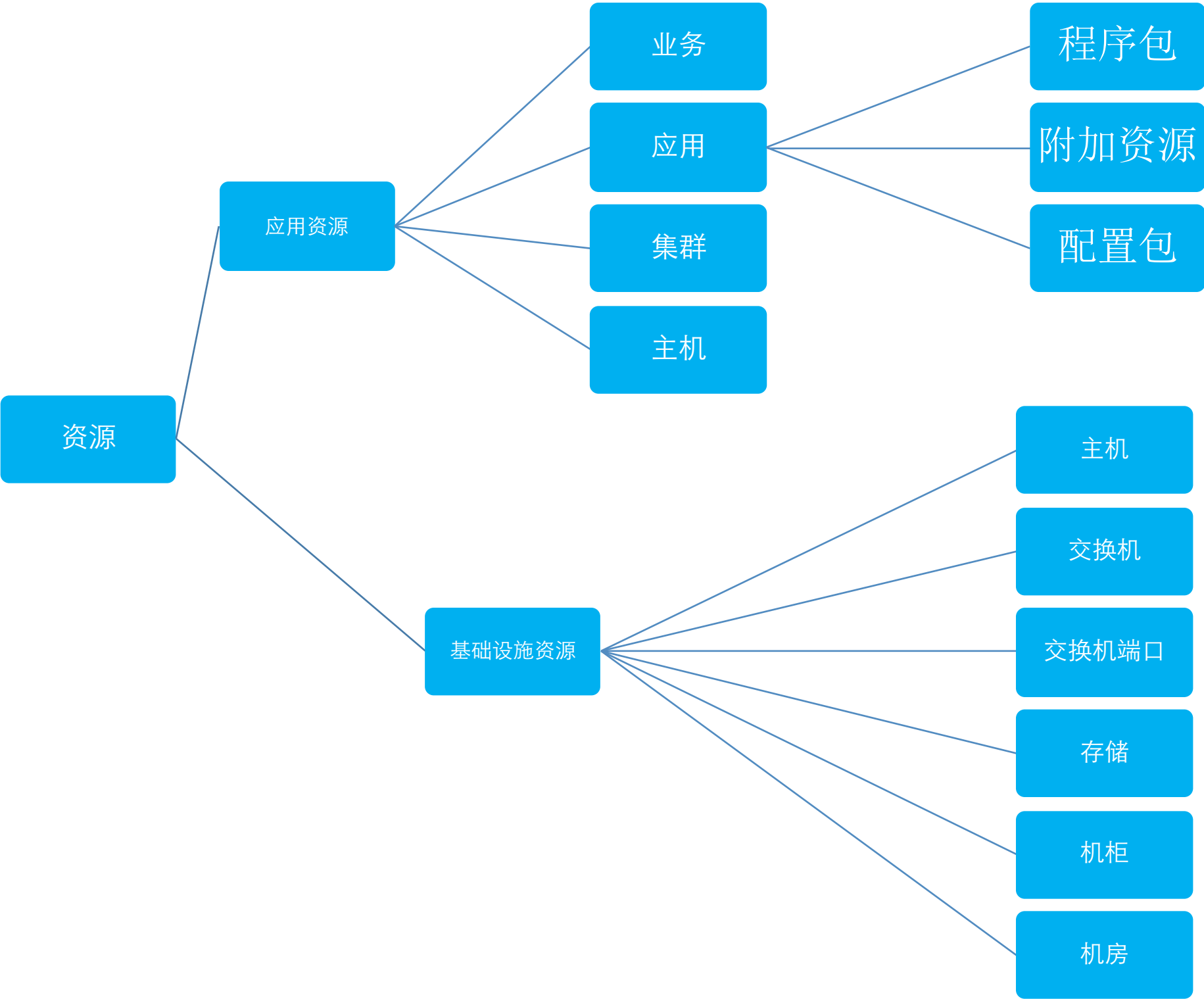
表现	描述
数据中心基础资源组负责CMDB建设	这是CMDB失败的大部分原因所在，纯粹是为了上CMDB而上CMDB。
DBA负责CMDB建设	CMDB变成一个真正的数据库
一次建设，运动式维护	这是一种结果驱动式的CMDB建设，而非过程式的CMDB建设，让CMDB的作用逐渐流于形式。
CMDB没有Excel维护方便	从表格的能力管理来说，excel的维护绝对是远超所有系统，但管理能力的嵌入，才是CMDB的核心能力。
不重视配置标准化能力	配置月标准化管理成本就越低

表现	描述
CMDB是元数据中心	CMDB是元数据中心，但不知道如何理解这个元数据？
缺少应用的维度关联CMDB	仅仅是为了管理配置项，管理资源，而不知道这些资源的应用者是谁。
缺少更多场景的支撑和驱动	CMDB数据的维护绝不是孤立维护的过程，而是要加入场景化的运维需要，驱动其不断维护的过程。这种维护分直接和间接的两种。
严格控制外围系统的数据自己建设	监控系统、事件系统、分析系统严格禁止其独立维护数据的过程，从而导致CMDB逐渐边缘化。
人工维护意味着规则隐藏	人工引入维护的数据越多，就以为这个规则越隐藏，未来被修改的可能性就越低。

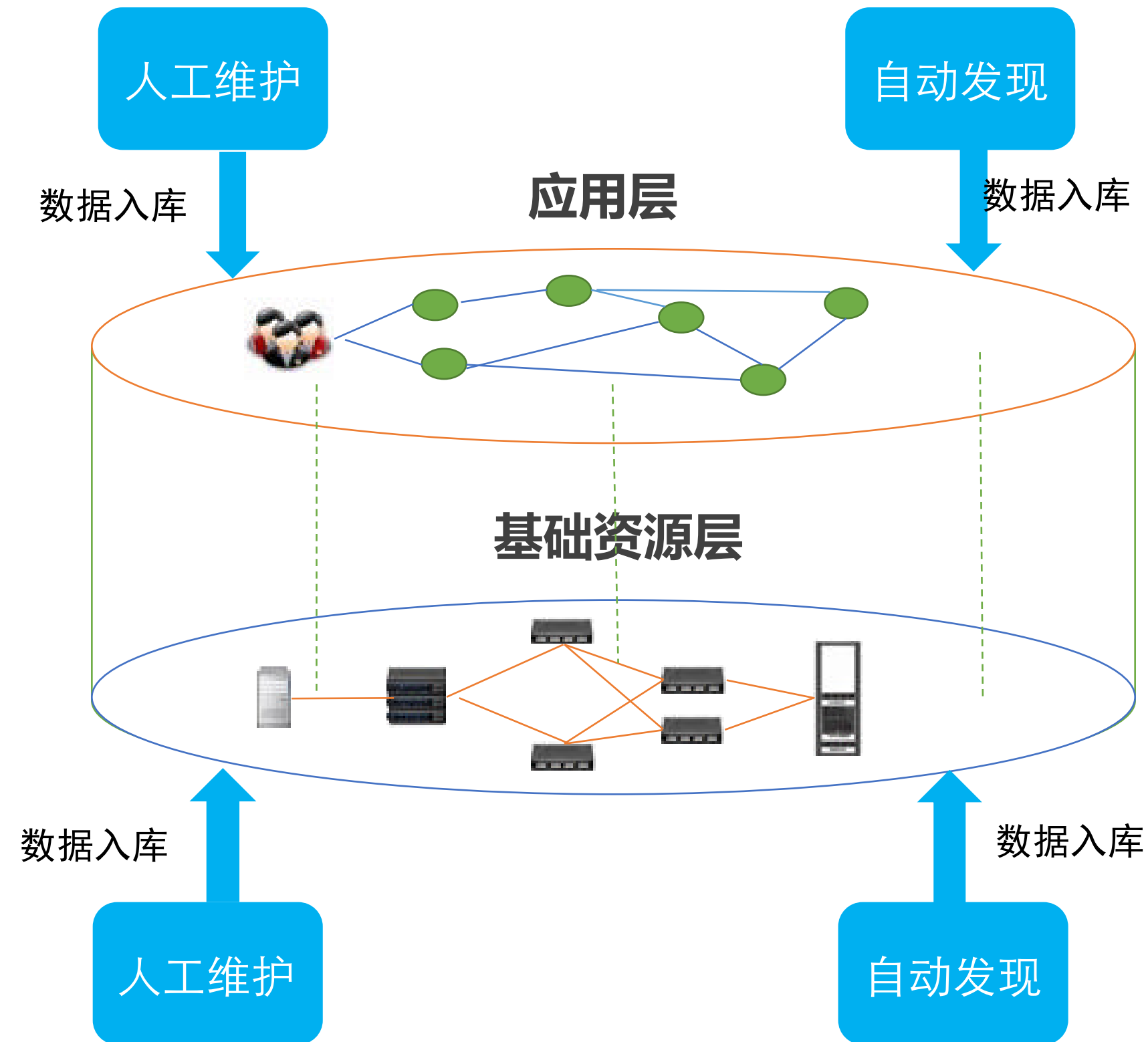
CMDDB实现与实施



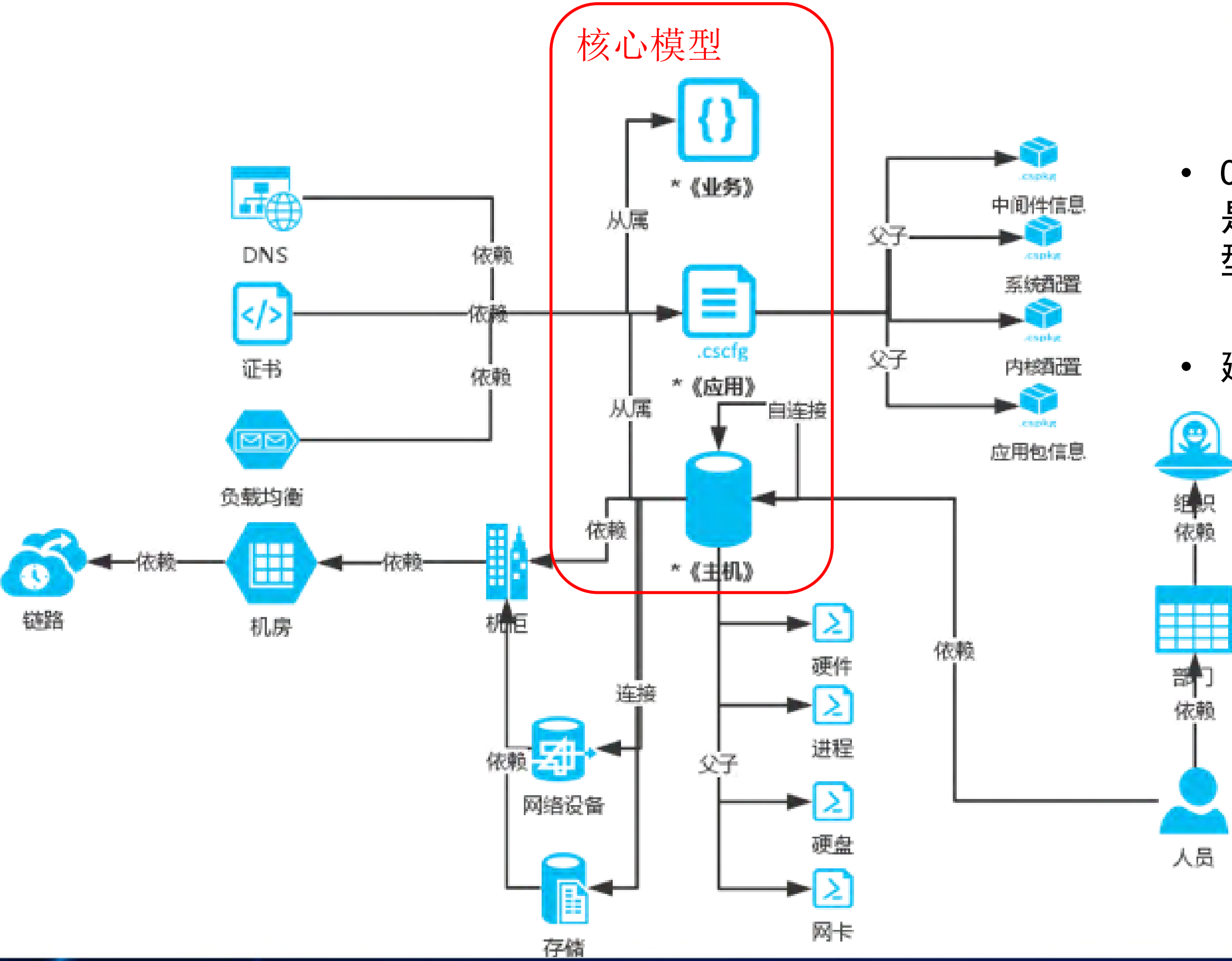
构成组件	描述
资源	在你的组织中到底存在多少的资源对象需要管理。就是Ci对象。
角色	这些资源对象到底是被谁管理的。对象的Owner是谁
场景	这些管理员平时的日常活动是什么。比如说流程，关联平台的场景



- 资源一分为二：基础设施资源和应用资源。
- 根据各自的资源域不断的向下划分，建立完整的资源管理谱系，这是系统分析的方法。



- CMDB架构分基础资源层架构和应用资源层架构
- 应用层资源架构把相关的资源以应用为中心实现资源整合。
- 资源及其资源的关系称之为拓扑（应用拓扑、物理拓扑）
- 资源管理方式有人工维护和自动发现两种方式。流程是人工维护的一种复杂场景和手段。



- CMDB分核心模型和扩展模型。核心模型是业务、应用、主机和程序包；扩展模型是基于这个实例的关联对象。
- 建立以应用为中心的资源管理模型



Jenkins

请记住创建的应用，基于它才能完成生命周期的管理

People

Build History

Project Relationship

Check File Fingerprint

We Need Beer

Build Queue

plugin_selector

plugin-compat-tester

Build Executor Status

Master

1 Idle

2 Idle

remote-slave-3 (offline)

remote-slave-6

1 Building plugin-compat-tester #5348

2 Building plugin_selector #2

remote-slave-7 (offline)

remote-slave-8 (offline)

AllAll DisabledAll FailedAll QAAll SetAll UnstableInfrastructureJenkins coreLibrariesOther ProjectsPluginsRuby

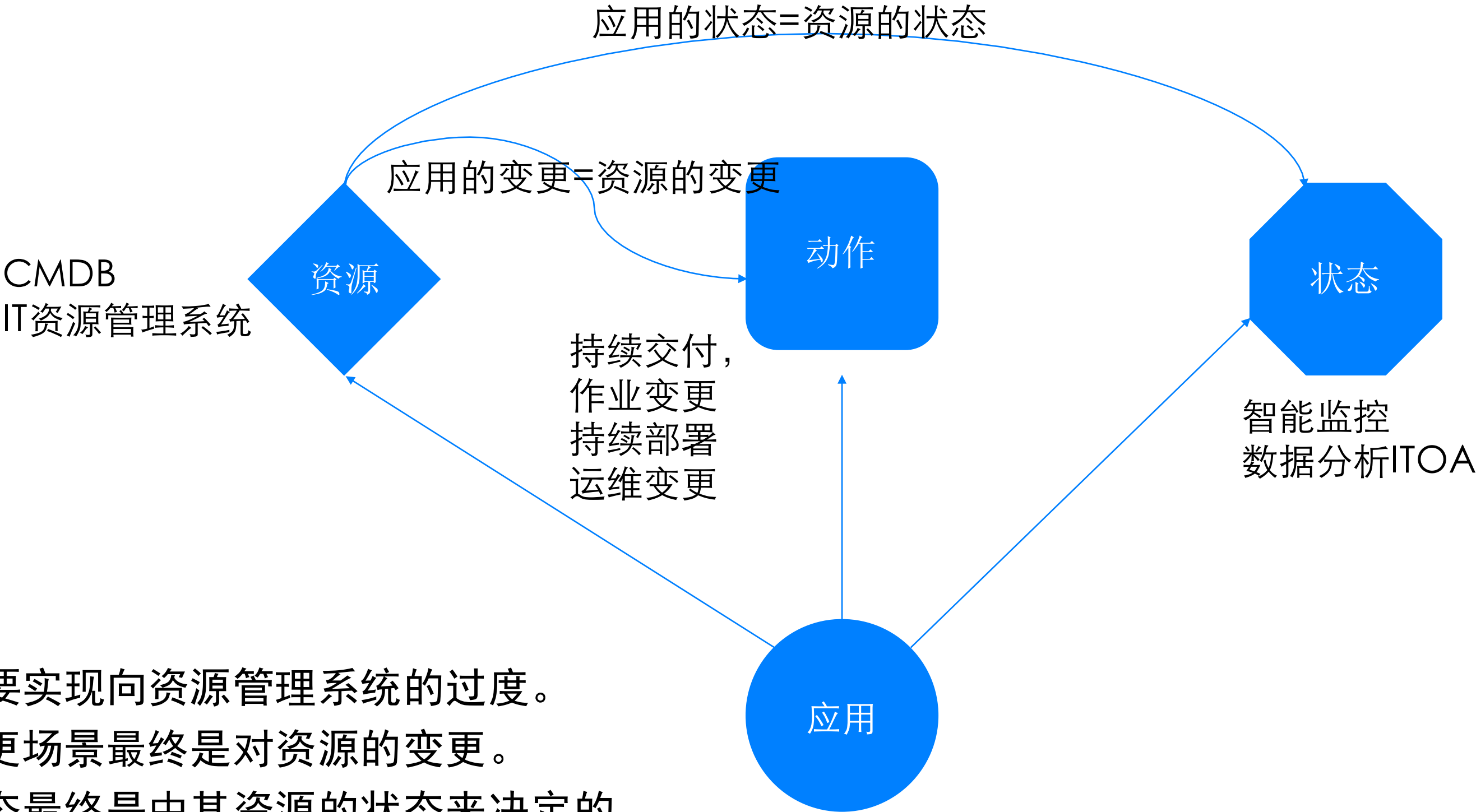
S	W	Name	Last Success	Last Failure	Last Duration	LC
		infra_plugins_svn_to_git	1 yr 5 mo (#593)	1 yr 4 mo (#768)	4 min 54 sec	
		infra_svnsync	1 yr 2 mo (#21199)	1 yr 2 mo (#21243)	1.5 sec	
		infra_update_center	9 hr 22 min (#4602)	22 min (#4611)	13 min	
		infra_update_center_stable	12 hr (#2062)	22 min (#2065)	10 min	
		jenkins.cb.javacustom	N/A	26 days (#1)	9 sec	
		jenkins.its_branch	1 mo 20 days (#46)	20 days (#51)	54 min	
		unit.runtime-audit	1 yr 6 mo (#28)	21 days (#90)	43 sec	
		its_svnfs	N/A	1 mo 27 days (#11)	19 sec	
		plugin-compat-tester	15 days (#5131)	8 hr 7 min (#5388)	27 min	
		plugins_backup	1 yr 1 mo (#15)	7 days 17 hr (#20)	2 min 37 sec	
		plugins_github-api	12 days (#4)	5 days 21 hr (#7)	55 sec	
		plugins_icloud	6 days 16 hr (#13)	1 day 20 hr (#16)	59 sec	
		tools_maven-hpi-plugin-maven-2.x	25 days (#41)	11 days (#42)	14 hr	

Icons: S M L

Legend RSS for all RSS for failures RSS for last latest builds

Help us localize this page

Page generated: Apr 18, 2012 12:24:07 PM Jenkins ver. 1.460-SNAPSHOT (no-04/11/2012 14:23 GMT-kohsuke)



- CMDB系统要实现向资源管理系统的过度。
- 应用的变更场景最终是对资源的变更。
- 应用的状态最终是由其资源的状态来决定的。

关系类型		类型描述	表达模式	类型例子
CMDB关系类型	父子关系	父不在，子则销毁了	父子表或者叫明细表。	服务器与网卡关系、服务器与进程关系。
	物理关系	是一种物理关系，物理存在的。	外键，关联关系表达	服务器与机柜的关系，机柜与机房的关系，大部分实体关系都是如此；服务器和交换机之间的关系等等。
	业务关系	是一种因业务产生的关联关系。	数据库关联表来表达	比如说应用之间的访问关系。

业务流视图

- 基于业务访问流的可视化呈现
- 业务访问流的呈现是基于服务和接口服务的呈现

架构视图

- 架构视图是一个完整业务和应用的全景视图
- 架构视图是基于应用最小粒度和基于业务最大的粒度呈现

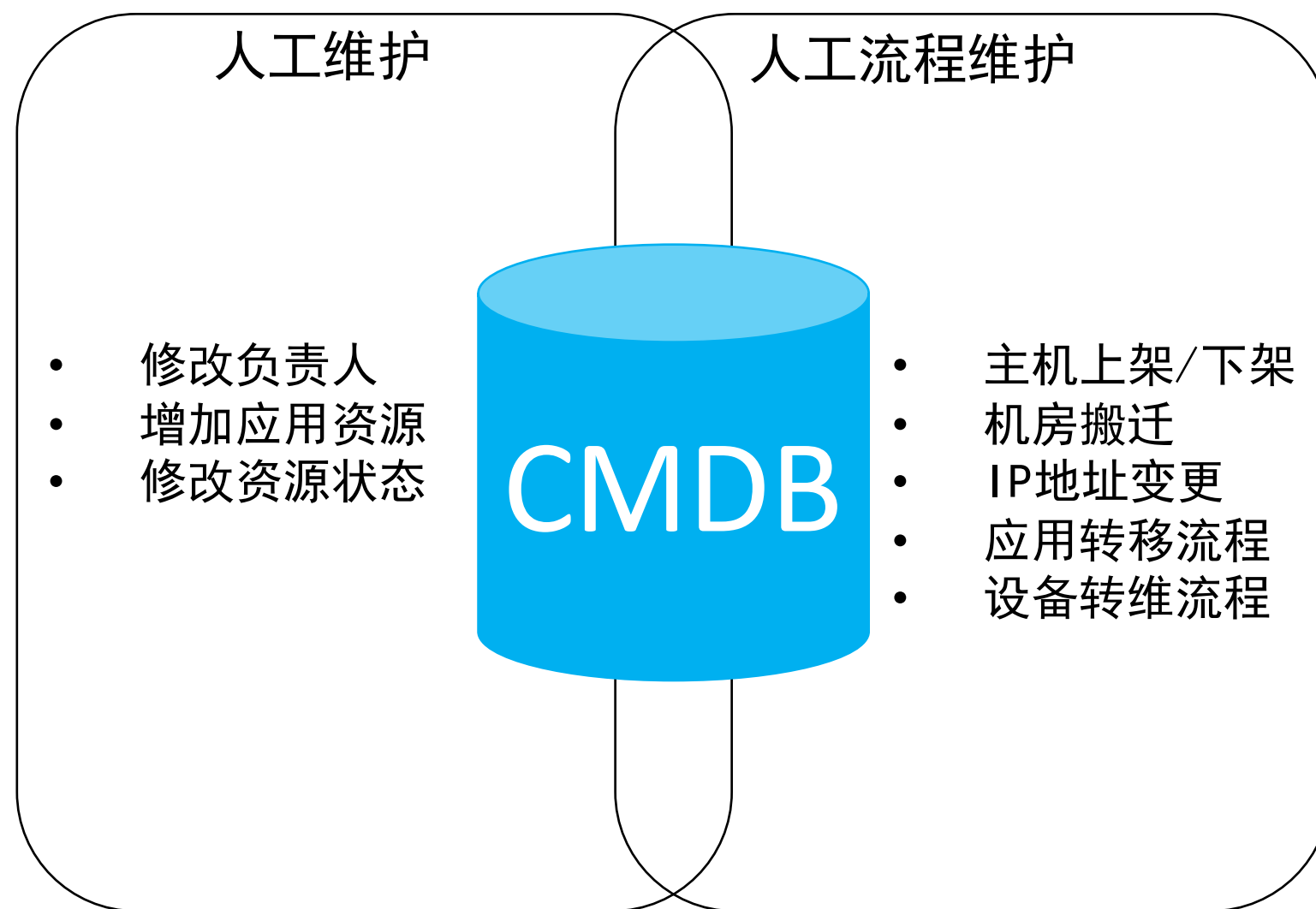
部署视图

- 部署视图是一个应用和一个业务的部署视图
- 部署视图包含了节点、组件、应用等内容

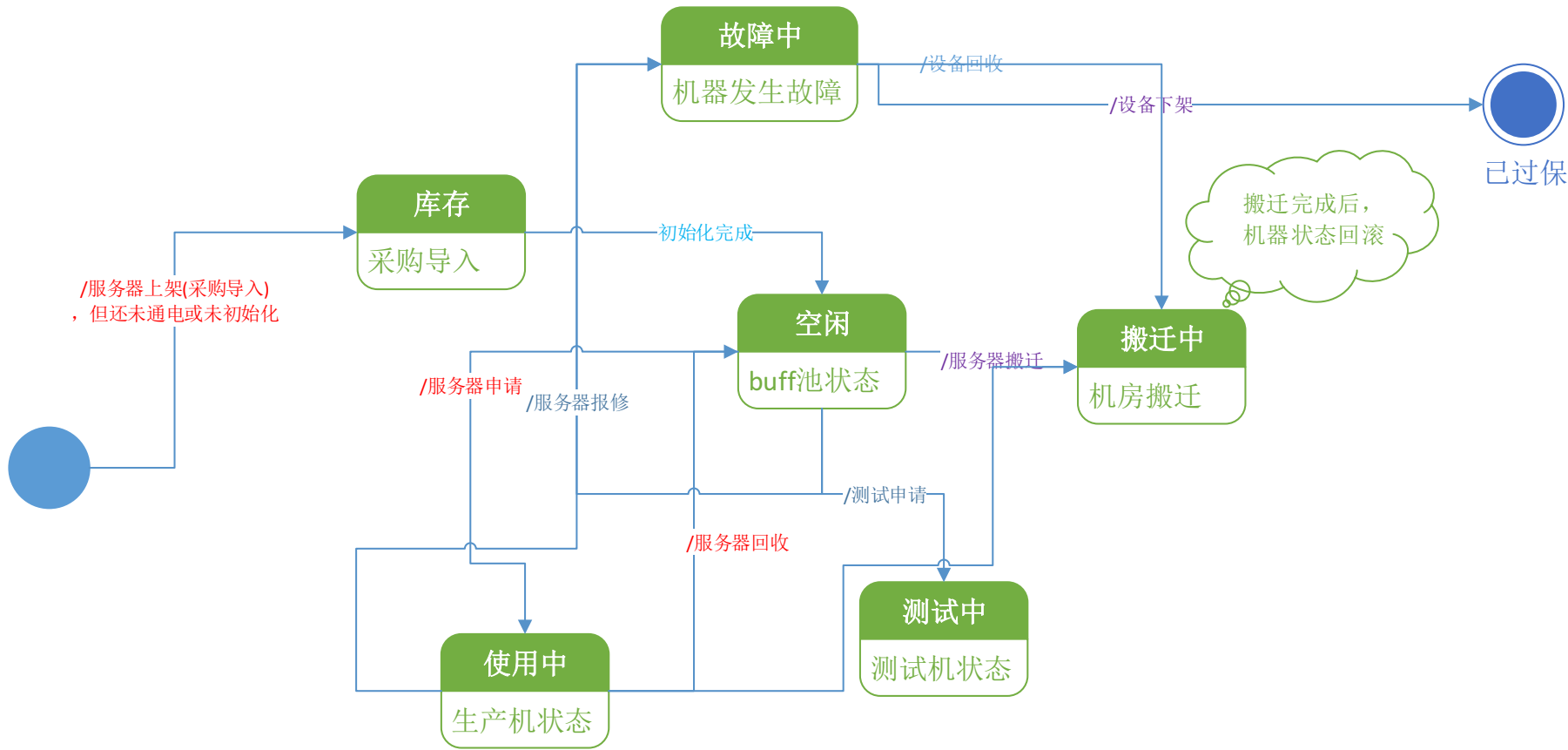
物理视图

- 物理视图就是底层基础设施的完整概貌
- 物理视图包含了机房、机柜、网络、服务器、虚拟化等信息

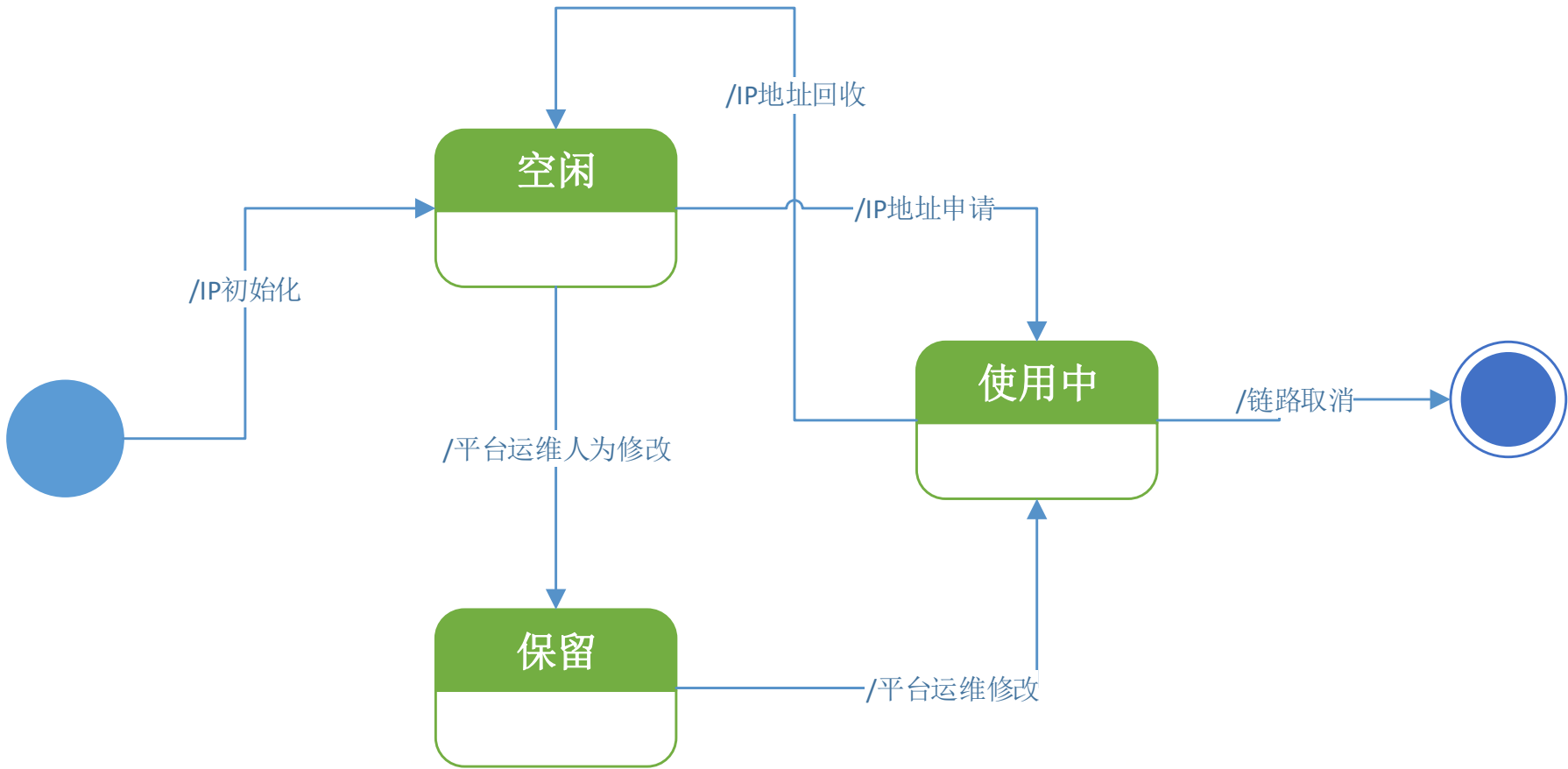
关于CMDB的流程设计



- CMDB流程是多角色协同控制的有效手段，流程一定是梳理某个场景下的多角色的职责关系。
- 资源的状态变化是需要人工触发或者流程触发
- 资源的状态的原子性保证需要通过流程来解决并发冲突问题
- 资源的状态变化是需要流程来控制过程（空闲-》申请中-》使用中）



服务器状态转换图



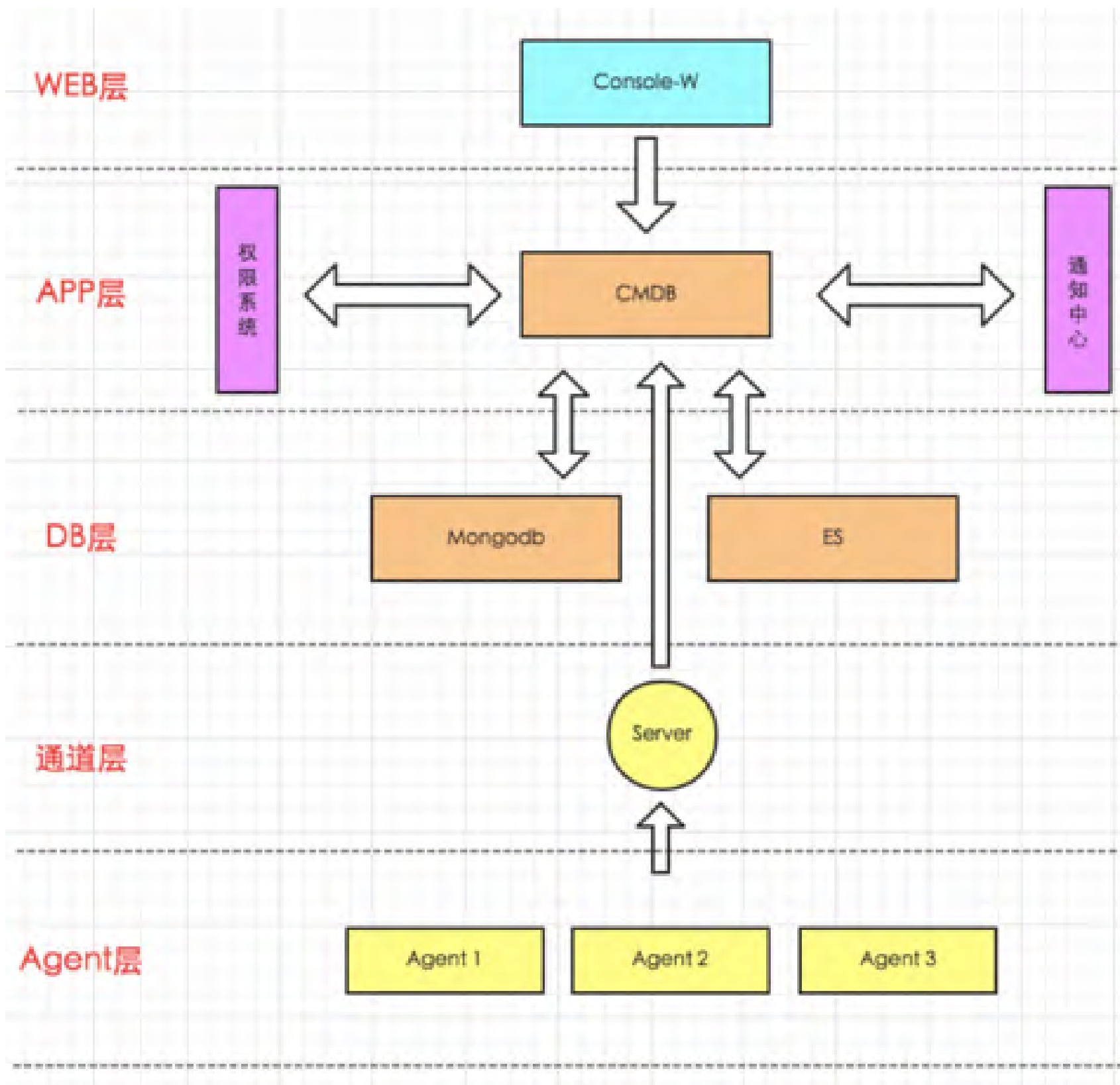
IP地址转换图

该图实现了资源/状态/动作三合一

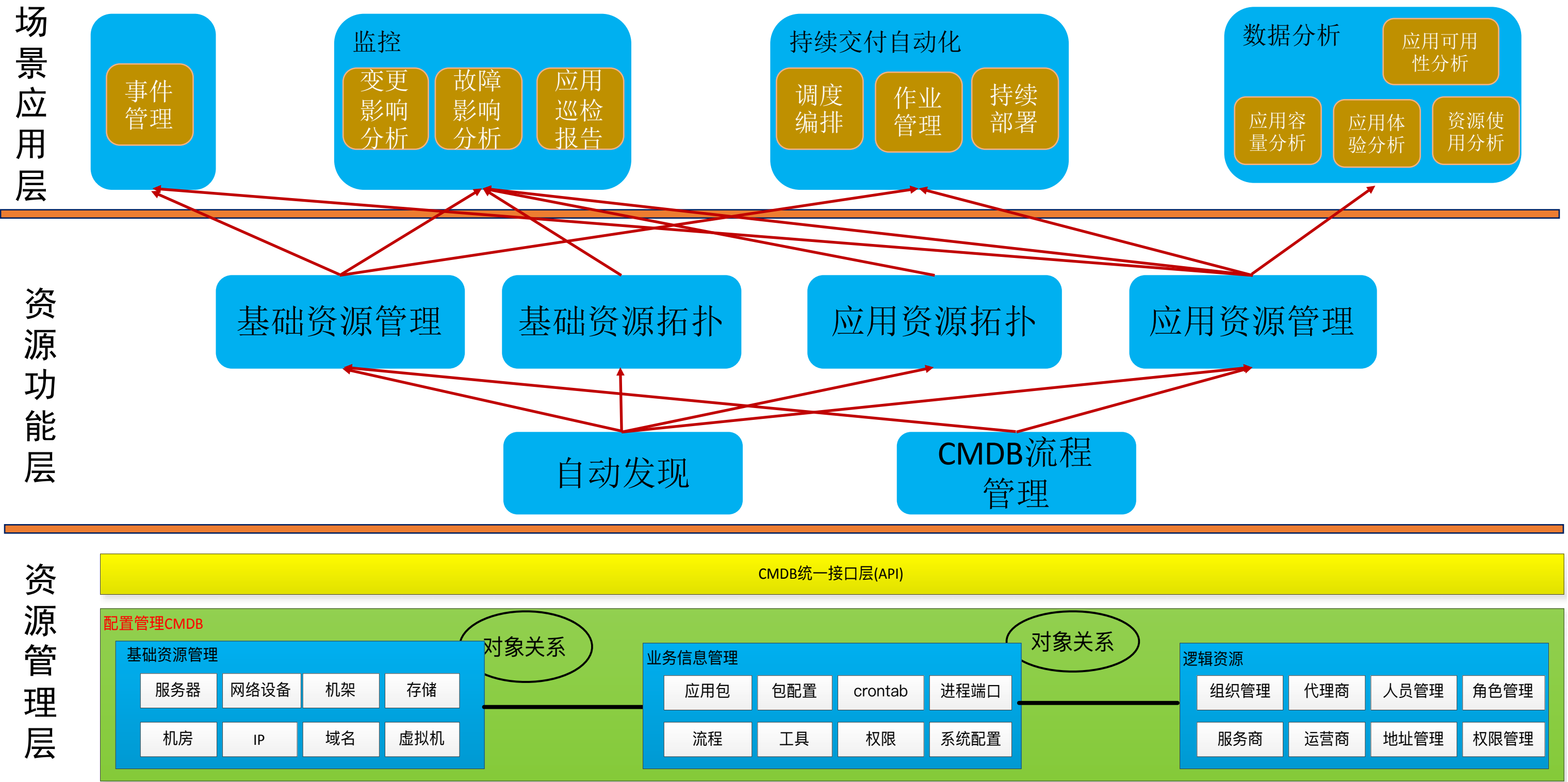
CMDDB的平台架构设计



- 建立完善的CMDB平台管理能力
- CMDB平台要覆盖基础资源管理、业务资源管理、逻辑资源管理等等

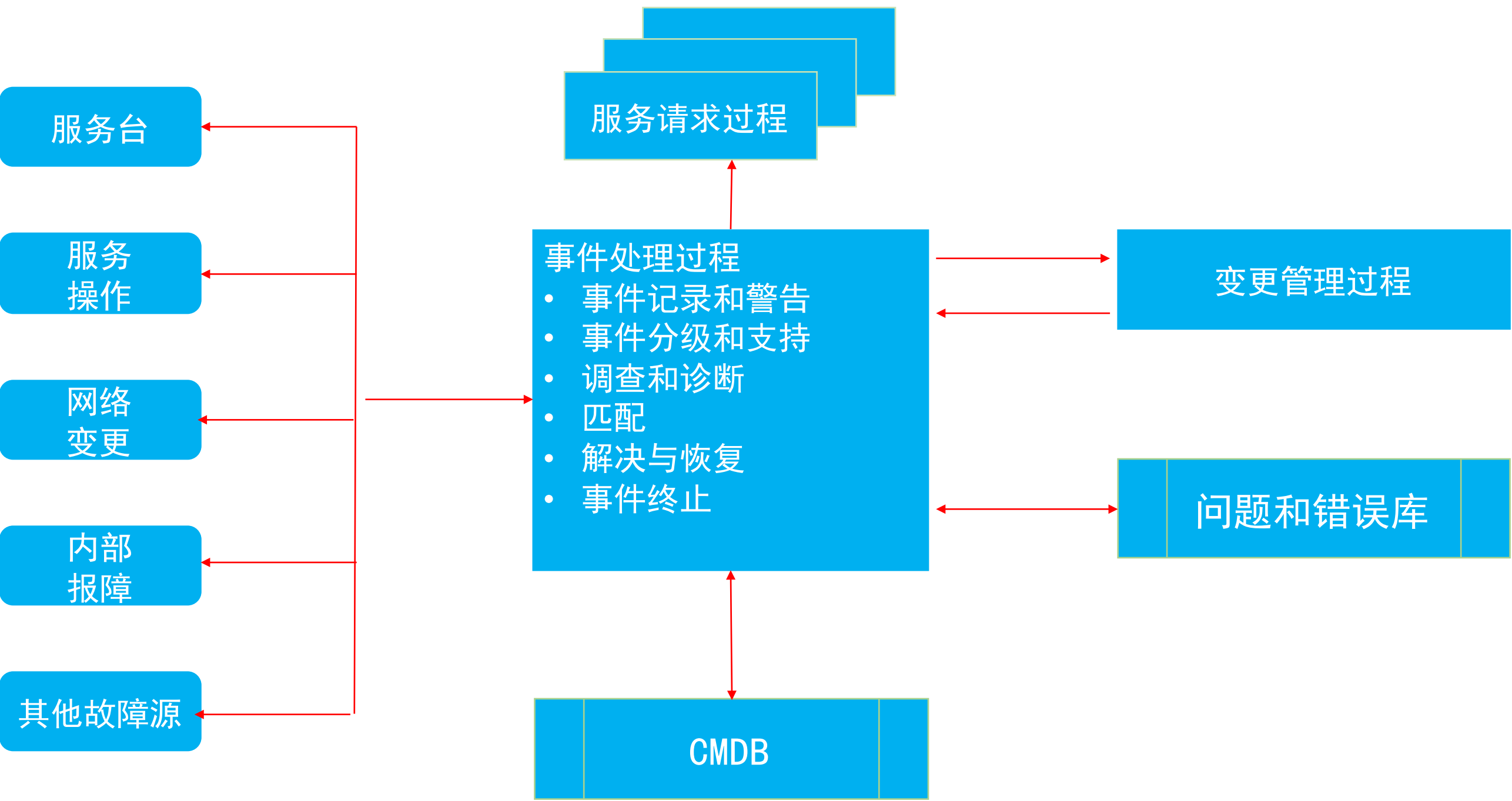


- CMDB实现通用资源CI管理逻辑
- 选用Mongodb数据库来存储数据，因为其介于关系数据库和非关系数据库之间，是非关系数据库当中功能最丰富，最像关系数据库的。可以任意定义对象，非常适合于用来存储灵活可定制的CMDB数据
- CMDB存储了一份冗余数据在ES，便于用户全文搜索
- CMDB有两个存储仓库，分别为mongodb和es。mongodb存储关系型数据，es只是用来做全文搜索

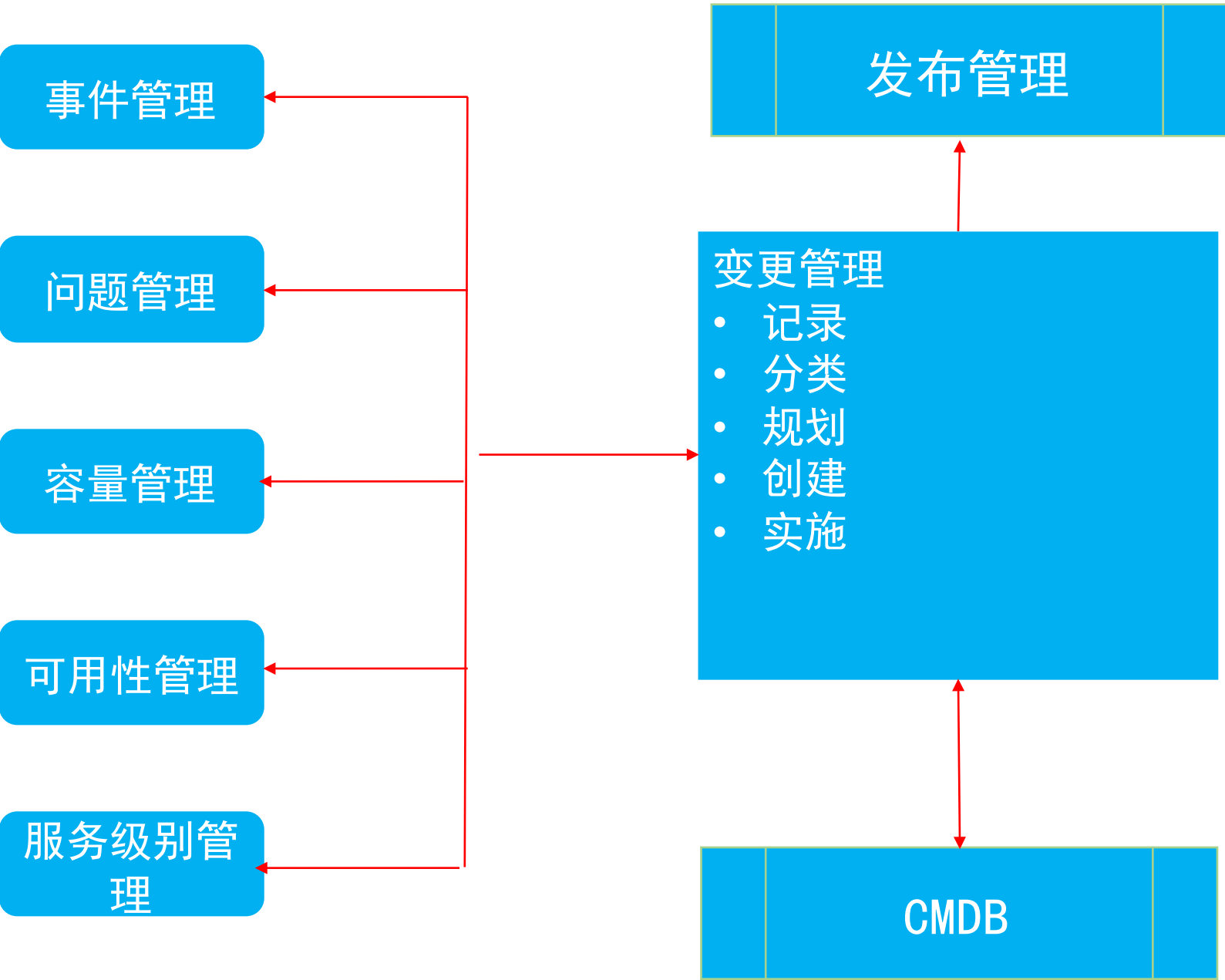


- 确定CMDB的管理层次，分期建设；领导参与；核心能力覆盖。注：不包含拓扑管理的功能

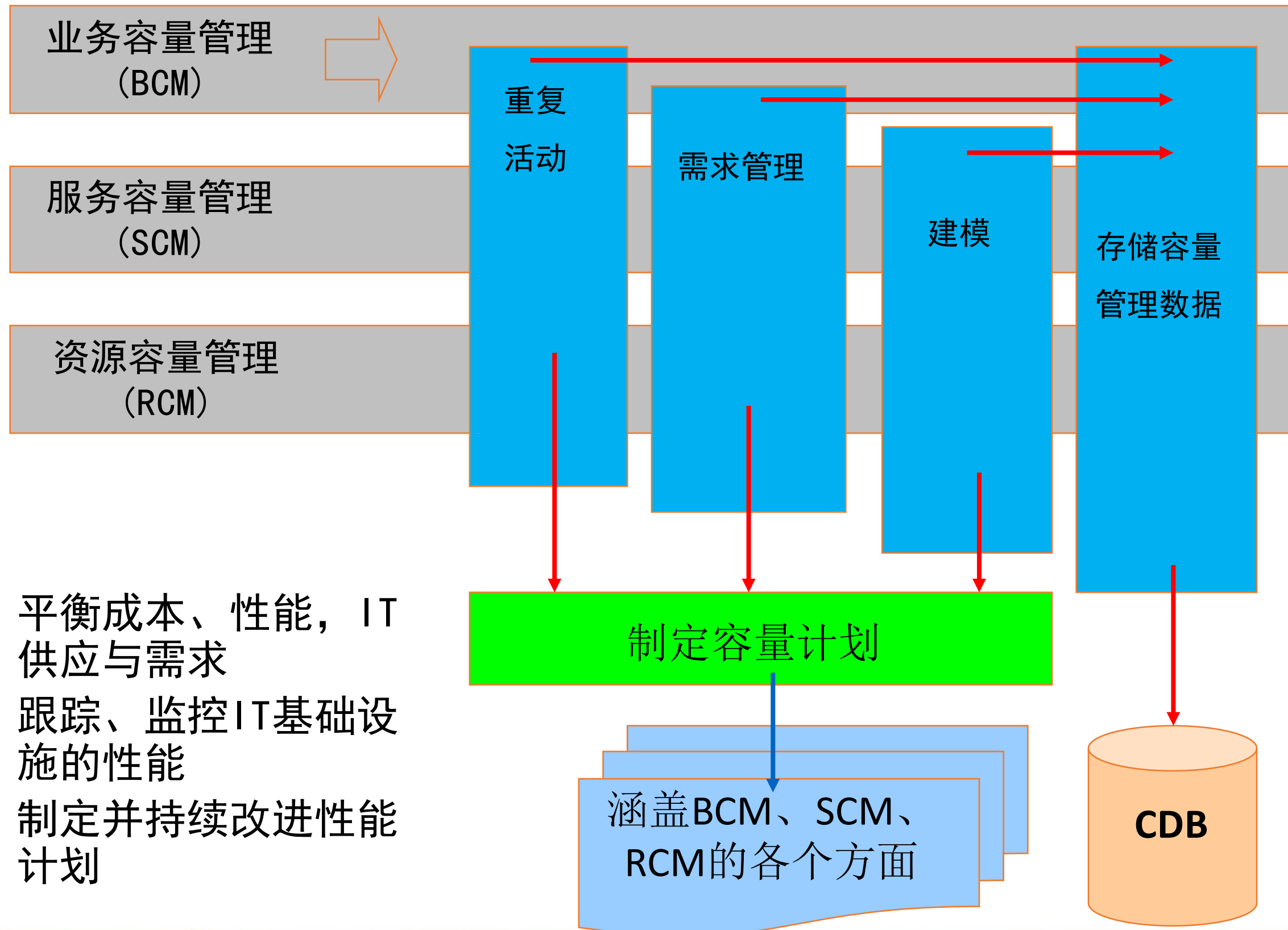
CMDB的场景化支撑



- 事件收敛
- 事件和资源对象状态关联
- 事件推送
- 事件升级



- 变更对象的资源管理
- 变更对象的状态管理
- 变更对象的权限管理
- 变更的影响评估



← 开发测试中间件

✈ 部署

🔧 维护

🕒 查看历史任务

📦 应用管理

程序包

配置包

交付流水线

场景管理

保存

取消

输入参数设置

重新初始化

构建

测试

预发布

发布

构建

Jenkins构建项目

打包

上传程序包

+ 添加步骤

+

tool-2-for-flow测试

测试集群查询

测试环境检查

测试环境发布

测试环境部署后检查

通知负责人

+ 添加步骤

tool-2-for-flow预发布

预发布环境查询

预发布环境检查

预发布环境发布

预发布环境部署后检查

通知负责人

+ 添加步骤

默认集群2发布

生产环境查询

生产环境检查

生产环境发布

生产环境部署后校验

通知负责人

+ 添加步骤

如意门 文档

仪表盘

域名管理

我的域名

修改历史

回收站

禁用快照

高级配置

NS Pool

NS Member

ISP Type

IDC Link

View

Zone

DNS工具

快搜

王金银 注销

查询条件

域名: 域名 IP|CNAME: IP|CNAME ZONE: 全部 负责人: 王金银 链路: 全部 业务分类: 全部

查询 重置

查询结果

注意: 红色背景表示该域名在流程中未生效, 请等待管理员审批。自己域名没权限操作? 请点击 认领

启用域名链路

禁用域名链路

转让

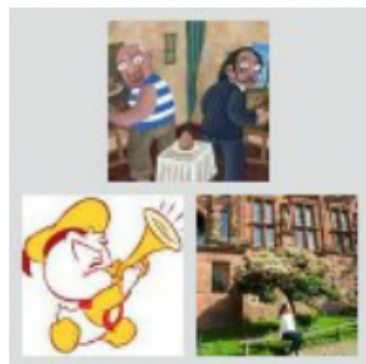
新增

域名	所属zone	TTL	调度策略	负责人	业务分类	修改日期	历史	操作
查询无结果, 请尝试修改查询条件								

第1页 / 共1页 / 共0条 / 每页 10 行

1

跳转



DevOps运维行业交流群



SDCC 2017 | 上海

互联网运维开发实战峰会

CSDN

谢谢