

第9章 按需应变运行环境下的整合

第 9 章 — 按需应变运行环境下的整合

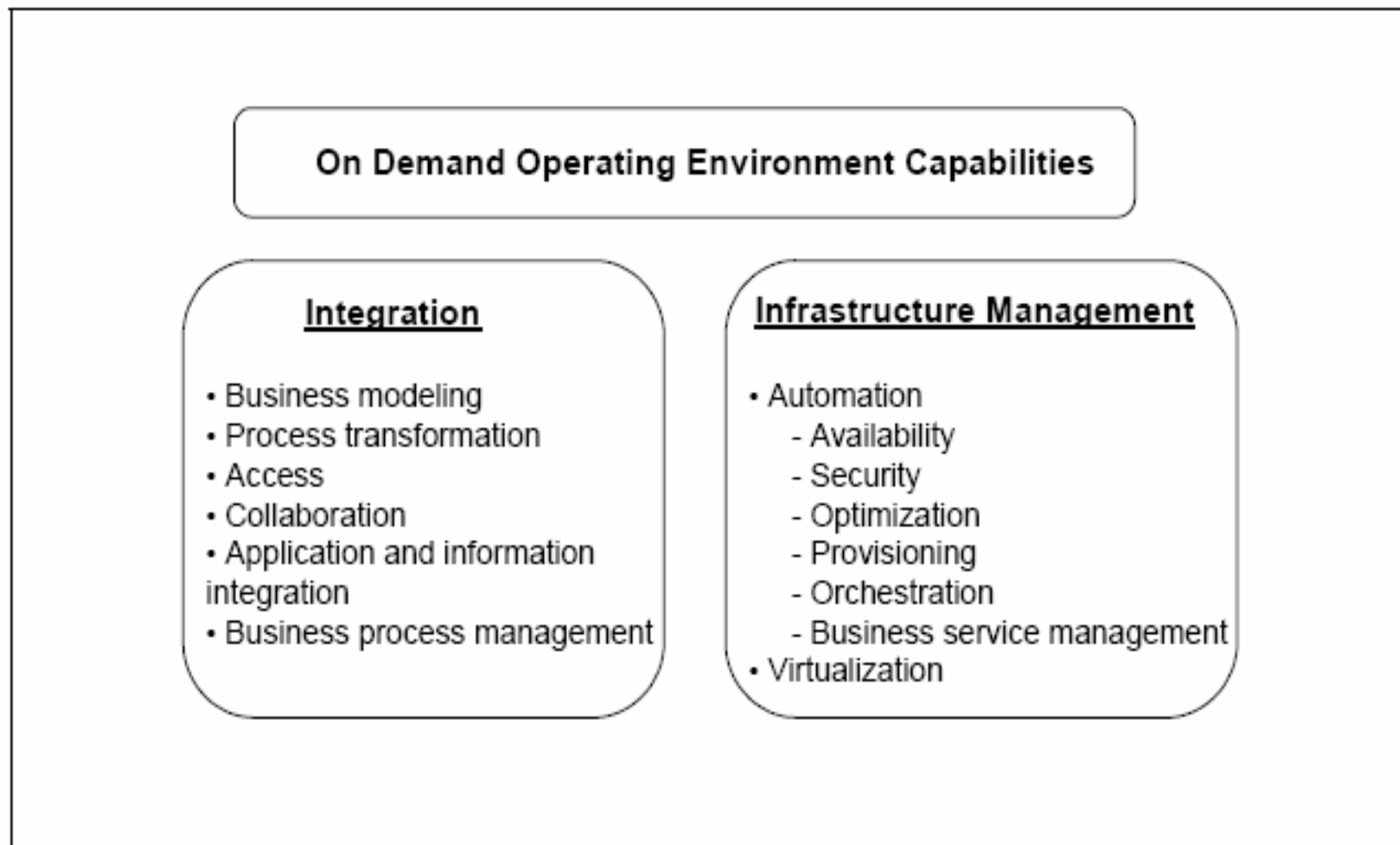
■ 章节目标

- 1. 了解整合的含义和内容；
- 2. 了解整合的方法；
- 3. 理解WEB服务、SOA以及门户技术在整合中的作用；
- 4. 掌握实现整合的相关产品；
- 5. 掌握不同平台系统进行整合的方法，了解一个完整的解决方案设计过程；
- 6. 了解在规划整合的运行环境时对非功能因素的考虑；
- 7. 掌握高可用拓扑结构的实现方法。

本章目录

- 9.1 整合企业新的基础架构
- 9.2 整合的方法
- 9.3 整合涉及的技术
- 9.4 整合与产品
- 9.5 案例
- 9.6 规划整合的运行环境

9.1 整合企业新的基础架构（一）



9.1 整合企业新的基础架构（二）

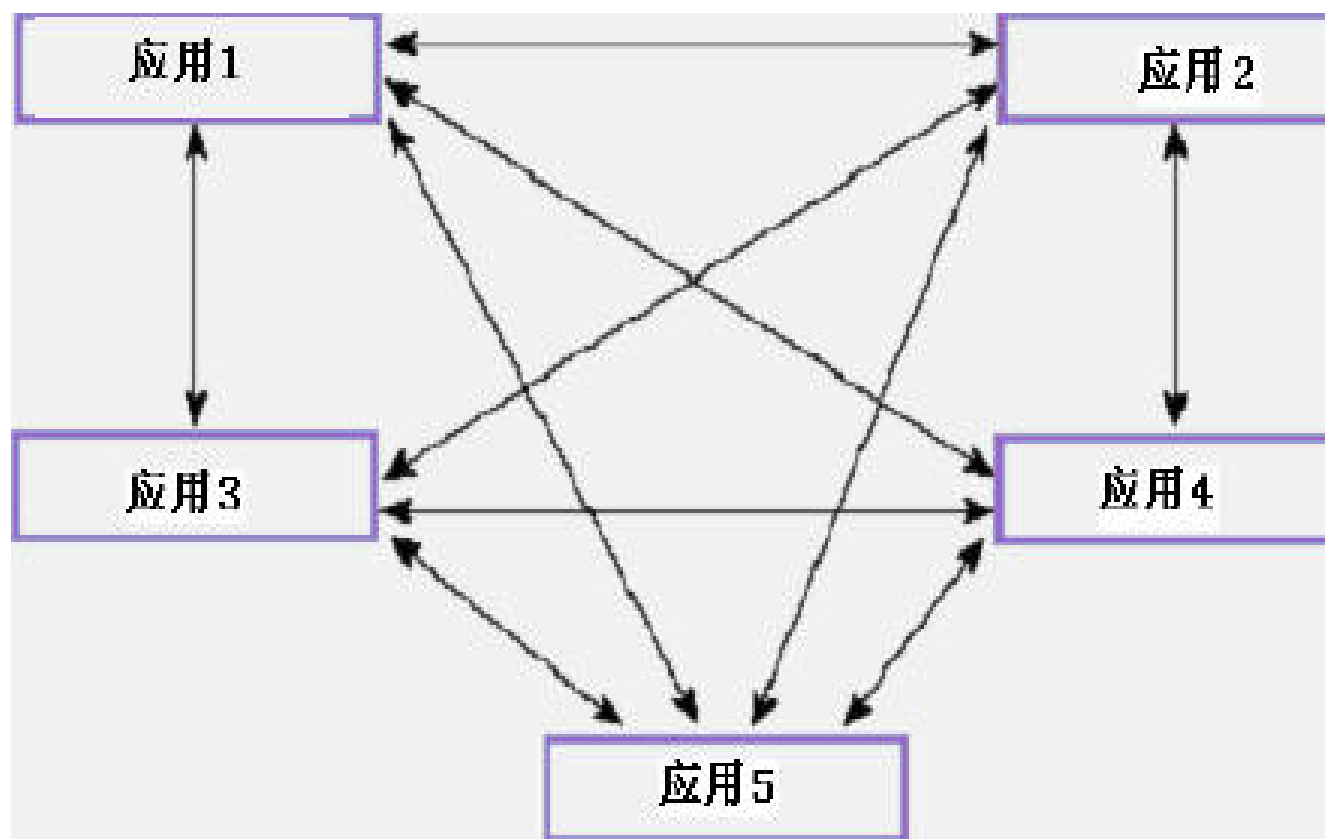
- 企业内部的整合
- 跨越企业的整合

操作环境的整合：

- 人的整合
- 流程的整合
- 信息的整合
- 业务流程重组**BPR**

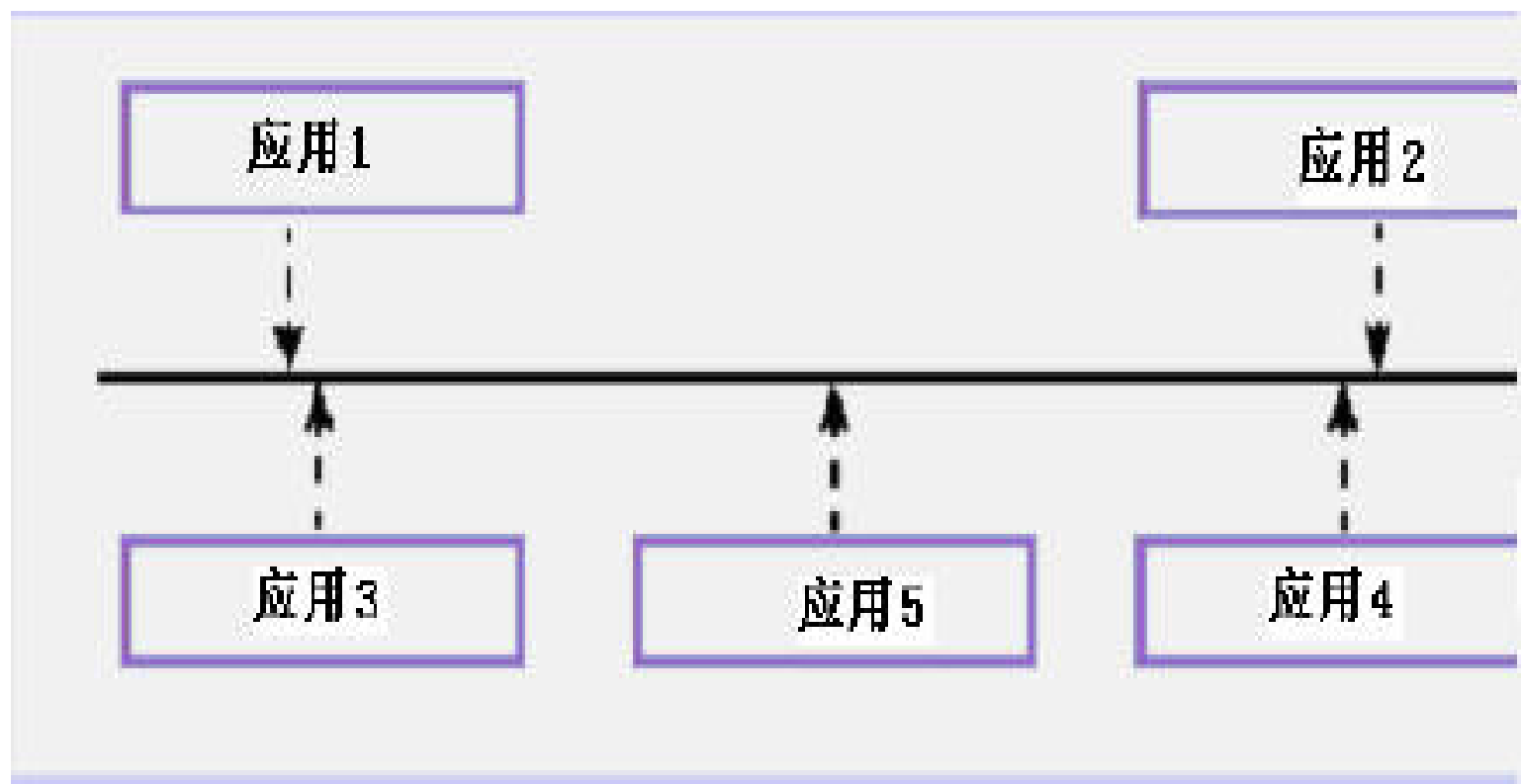
9.2 整合的方法—发展过程

■ 第一阶段：孤立系统间的互联



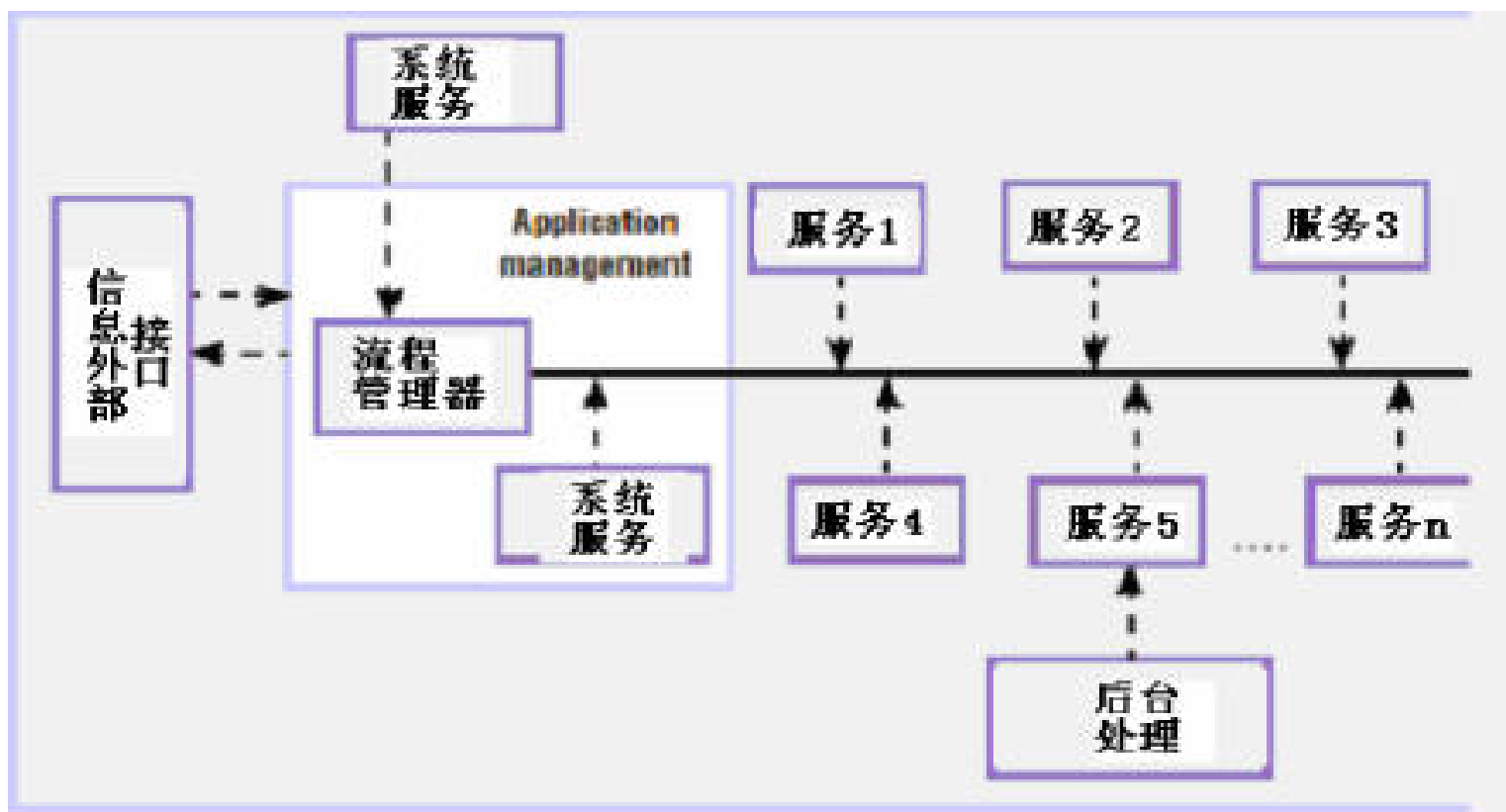
9.2 整合的方法 — 发展过程

- 第二阶段：企业应用整合



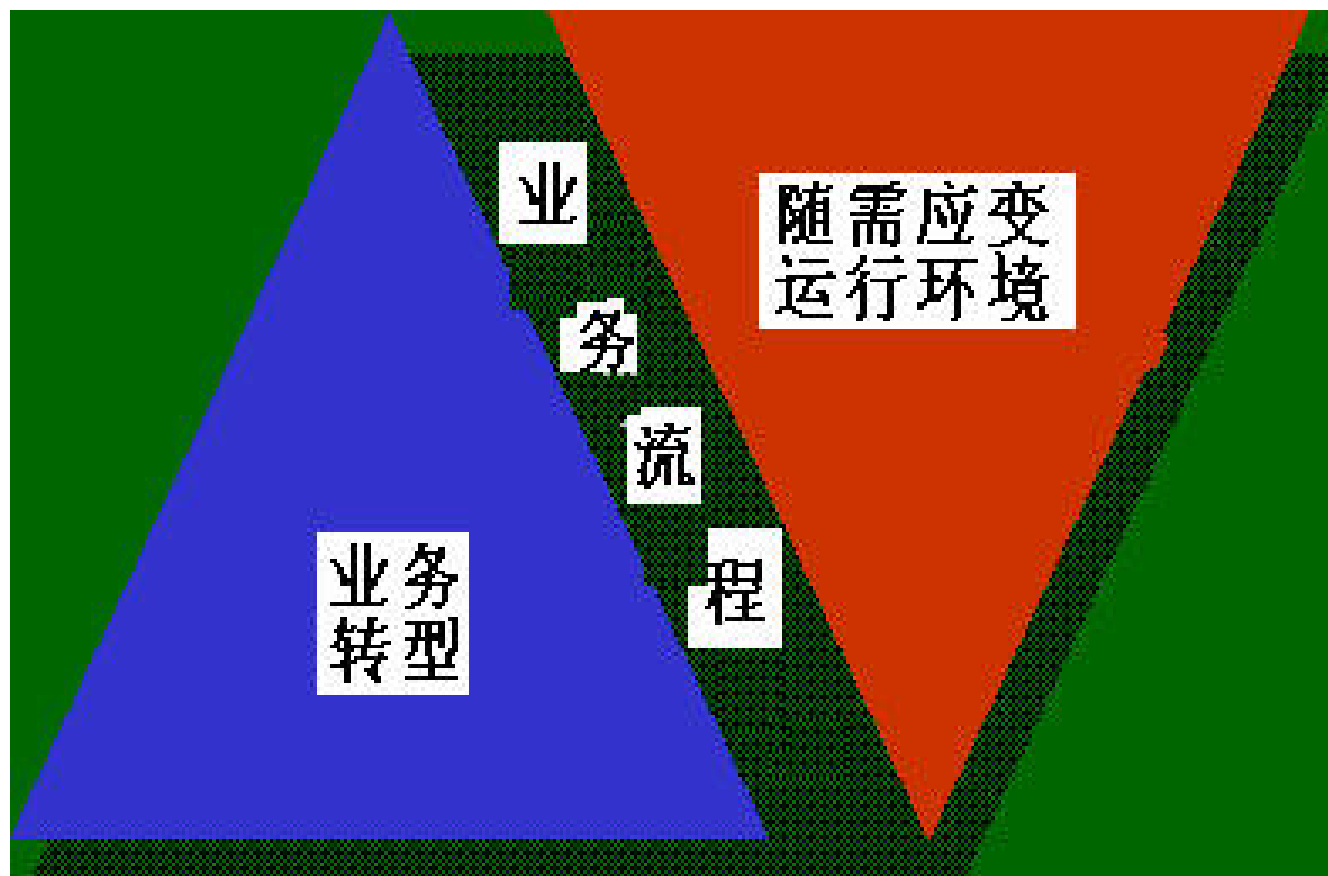
9.2 整合的方法 — 发展过程

■ 第三阶段：面向服务的企业应用整合

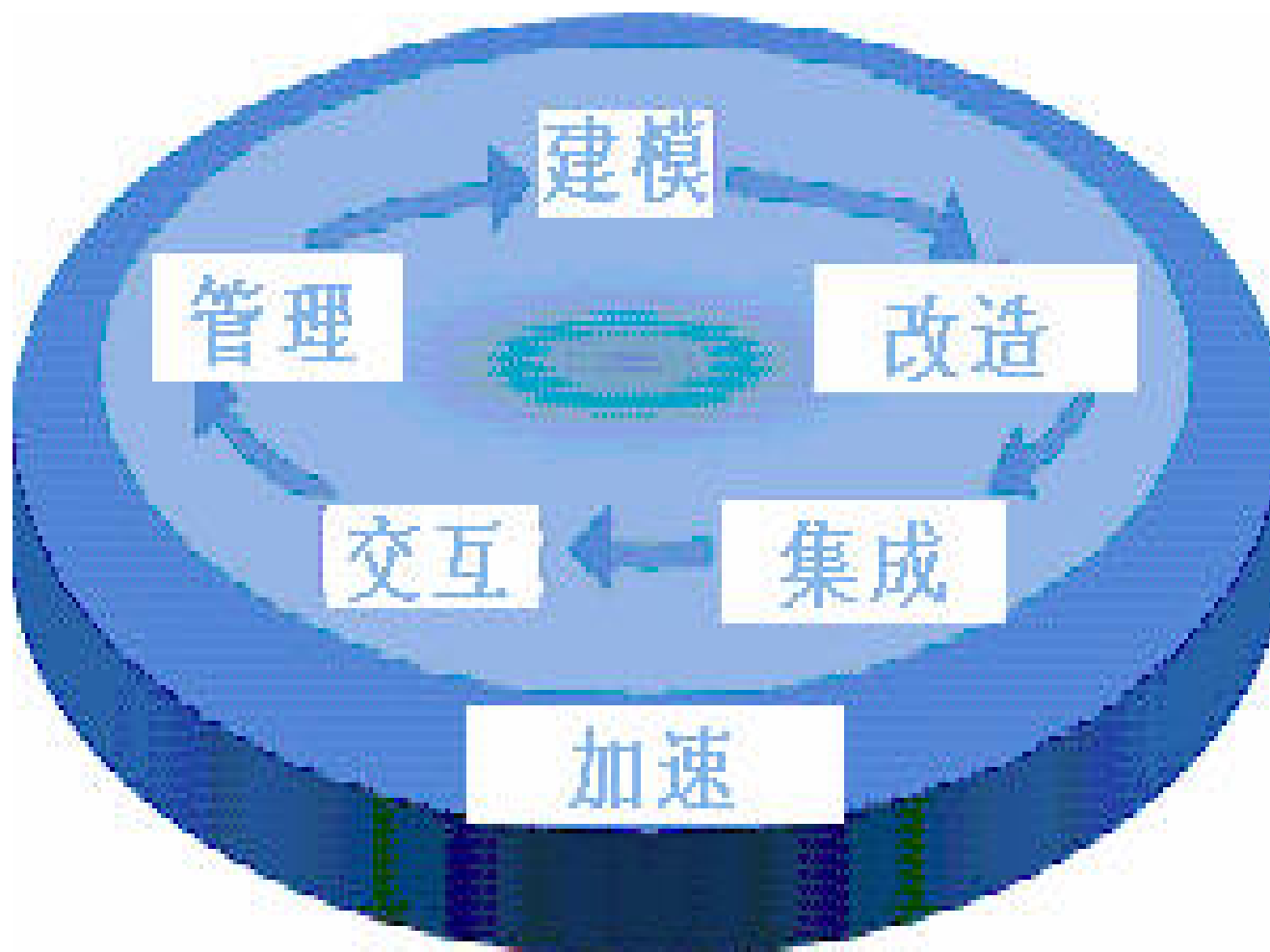


9.2 整合的方法一着手

- 从分析企业的实际需求开始



9.2 整合的方法——六步法



9.3 整合涉及的技术

- WEB 服务
- 面向服务体系结构
- 整合 IT 使用环境的基础 – 门户技术
 - 远程门户网站 Web 服务 (Web Service for Remote Portals, WSRP)

9.4 整合与产品

- 9.4.1 IBM三大整合新利器
- 9.4.2 WebSphere MQ与整合
- 9.4.3 Websphere Portal与整合
- 9.4.4 大型机, CICS TS, WAS, Rational与整合
- 9.4.5 Integrated solution console与整合
- 9.4.6 Lotus Workplace与整合
- 9.4.7 IBM Workplace与整合

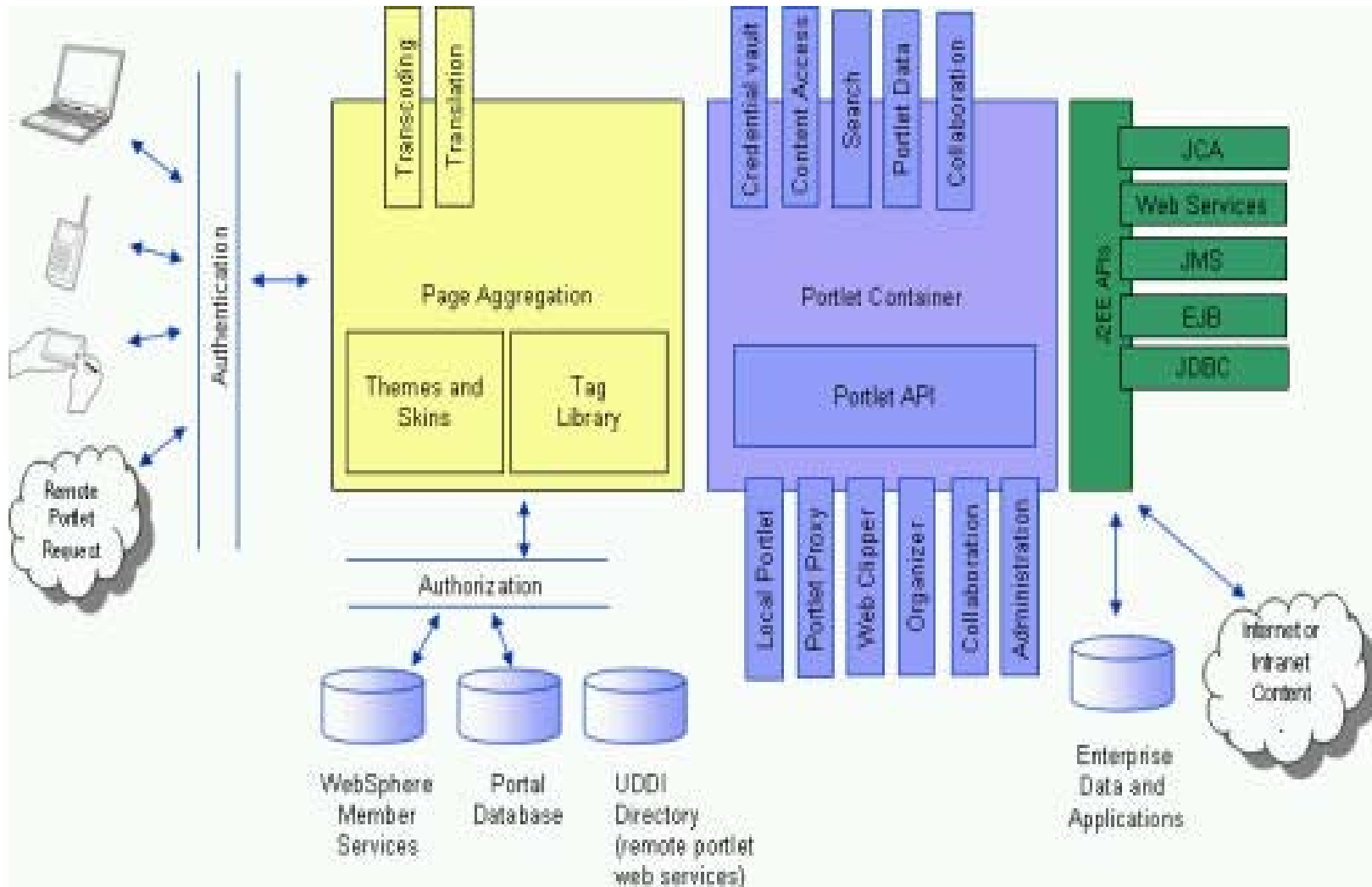
9.4.1 IBM三大整合新利器

- IBM WebSphere Process Server——基于SOA整合的基石
- IBM Tivoli Monitoring 6.1——切入IT资源提升IT服务管理
- IBM Lotus Notes/Domino 7.0 ——革新的客户机模式

9.4.2 WebSphere MQ与整合

- WebSphere MQ
- WebSphere MQ Integrator
- MQ Workflow
- CrossWorlds
- Holosofx

9.4.3 Websphere Portal与整合



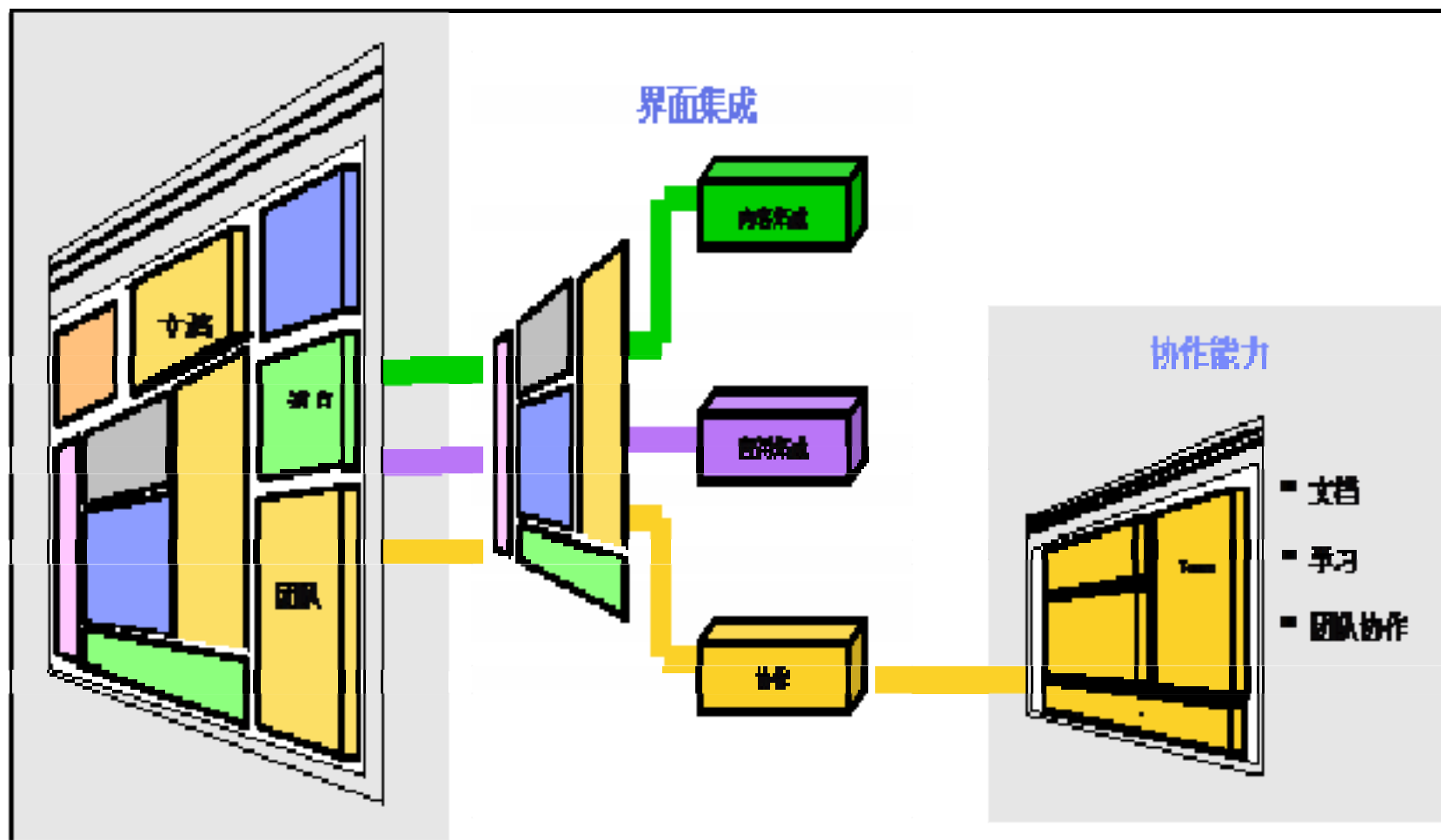
9.4.4大型机，CICS TS，WAS，Rational与整合

- WebSphere Application Server for z/OS
- CICS Transaction Server for z/OS
- Rational

9.4.5 Integrated solution console与整合

- 它基于Web界面，提供包括设置，配置和运行监控在内的对IT环境的基本系统管理功能。
- 方便IT管理者们经由一个基于Web界面的控制平台管理所有的IBM软件和非IBM软件产品。
- 主要功能是监控资源的健康状态、问题诊断和系统管理

9.4.6 Lotus Workplace与整合



9.4.7 IBM Workplace与整合

- IBM Workplace包括了Lotus Workplace、WebSphere Portal、Lotus Notes/Domino和WebSphere Everyplace，以及对这些产品提供的新的客户端技术。

9.5 案例

- 9.5.1 基于.net和J2EE的系统的整合
- 9.5.2 在主机平台上开发一个基于XML的解决方案

9.5.1 基于.net和J2EE的系统的整合

- 整合使用不同技术（例如J2EE和.Net）的两个系统
- 需要快速实现整合
- 希望实现松散耦合

可选技术：

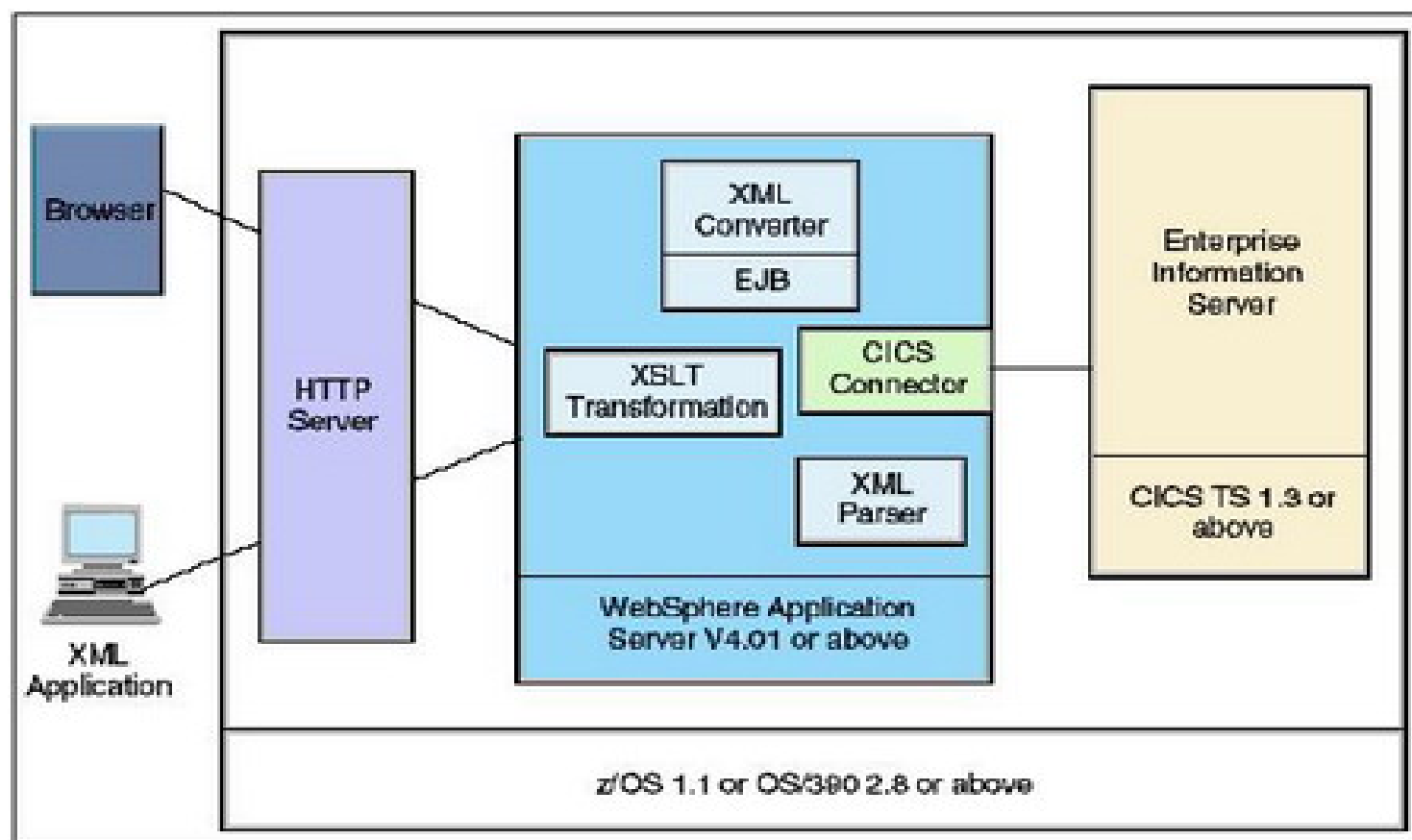
- Web 服务
- 消息技术
- 适配器或连接器

9.5.1 基于.net和J2EE的系统的整合

- 直接连接模式（Direct Connection）
- ESB 模式
- 扩展企业SOA模式（Extended Enterprise SOA patterns）

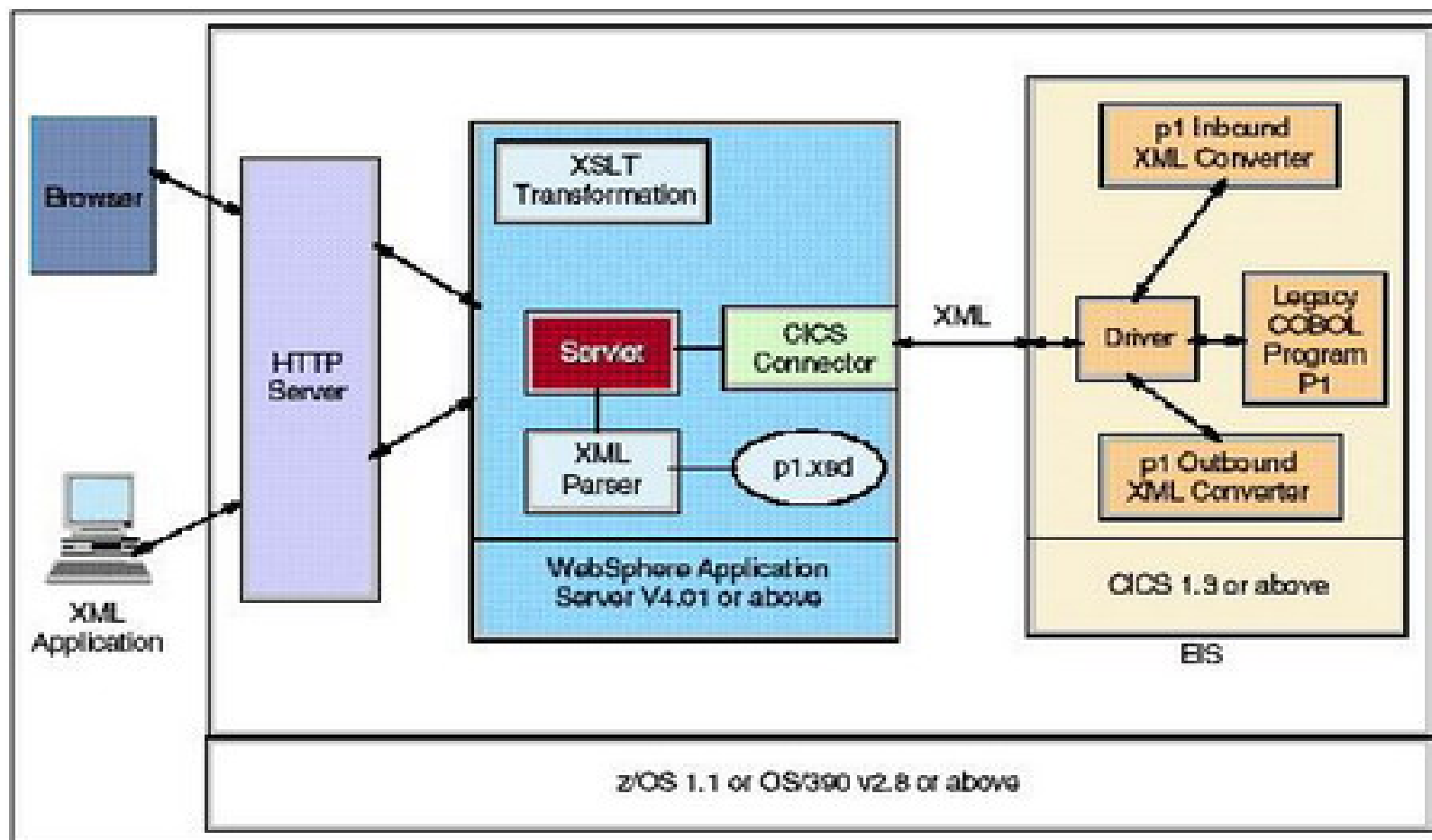
9.5.2 在主机平台上开发一个基于XML的解决方案

- 开发一个到现有的主机应用的**XML**接口



9.5.2 在主机平台上开发一个基于XML的解决方案

■ 建立新的应用



9.6 规划整合的运行环境

- 9.6.1 运行环境规划中的非功能性因素考虑
- 9.6.2 运行拓扑结构实例

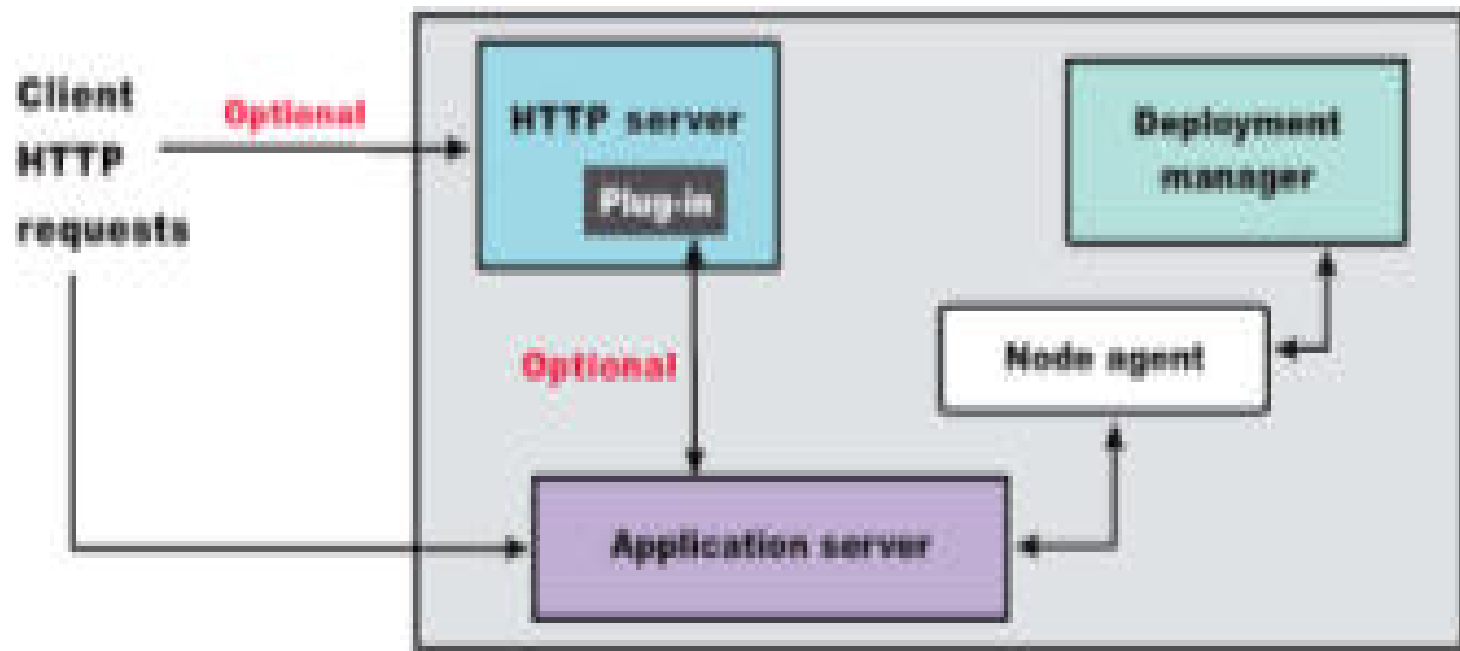
9.6.1 运行环境规划中的非功能性因素考虑

- 可扩展性
- 可用性
- 可维护性
- 可管理性
- 安全性

9.6.2 运行拓扑结构实例

- 9.6.2.1 单个服务器拓扑结构
- 9.6.2.2 高可用性系统拓扑设计

9.6.2.1 单个服务器拓扑结构



9.6.2.2 高可用性系统拓扑设计

- 系统所有的实体都必须有冗余设计；
- 必须监控每一个实体的故障情况；
- 出现故障的实体不能够接受任何工作。

9.6.2.2 高可用性系统拓扑设计

