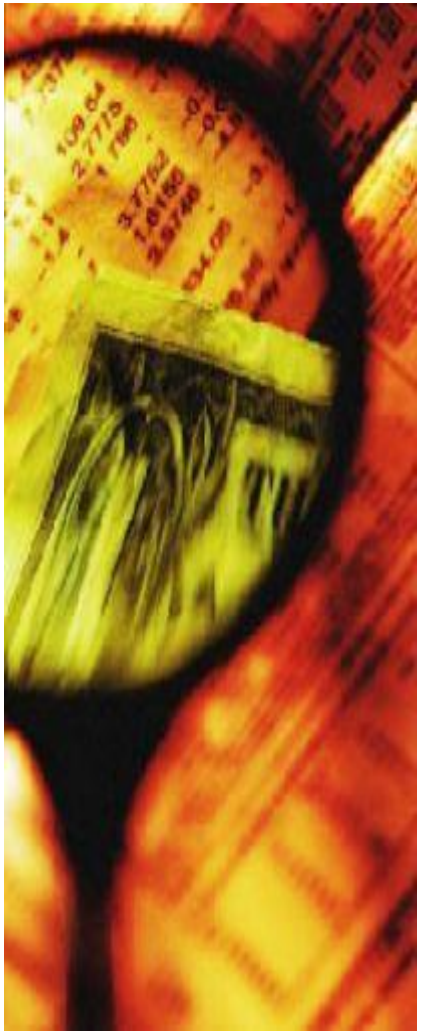
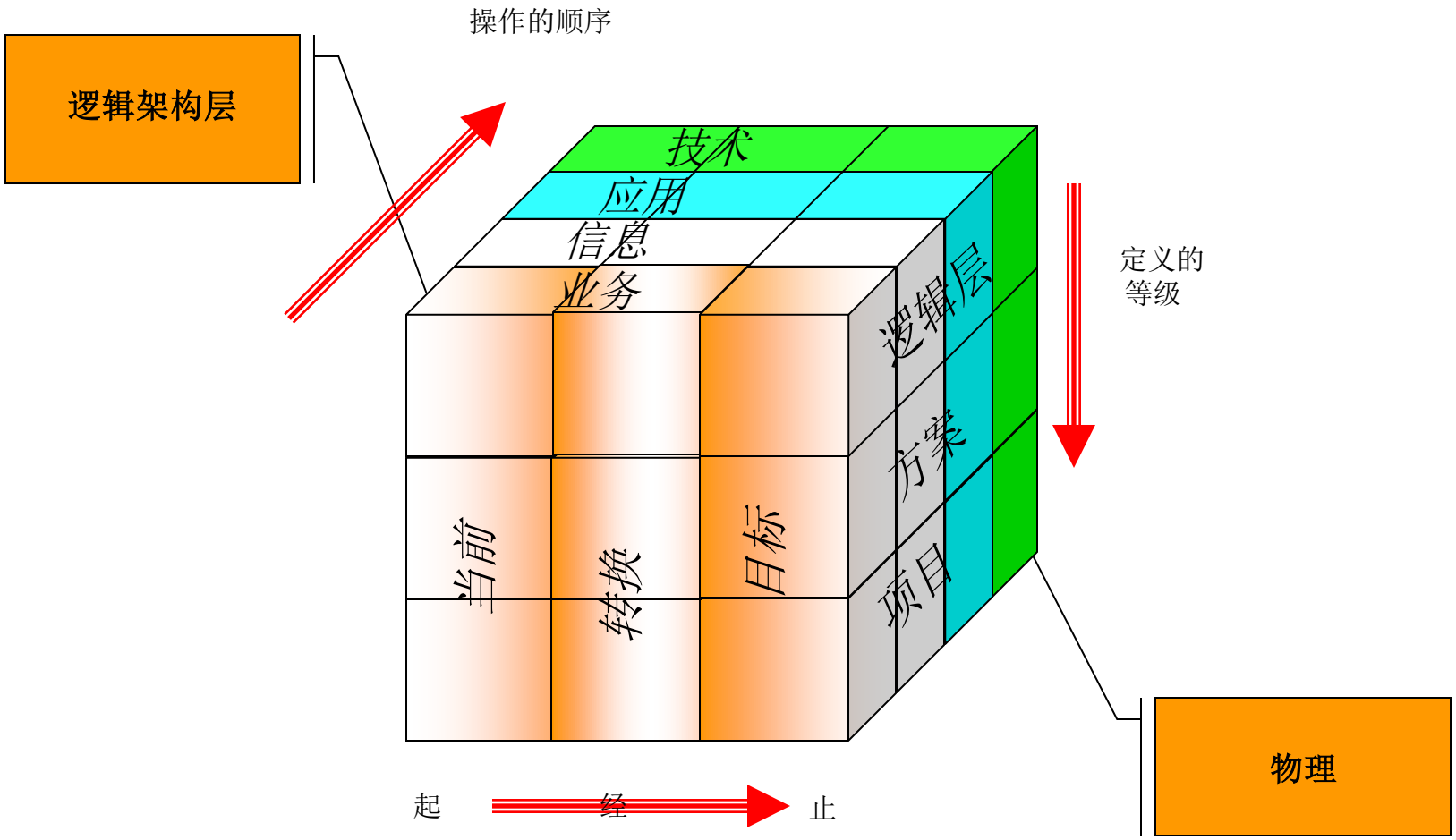
数据仓库技术架构

Agenda

- 架构设计原理
- 整体架构说明
- 方案说明
- 物理体系架构
- 架构前瞻



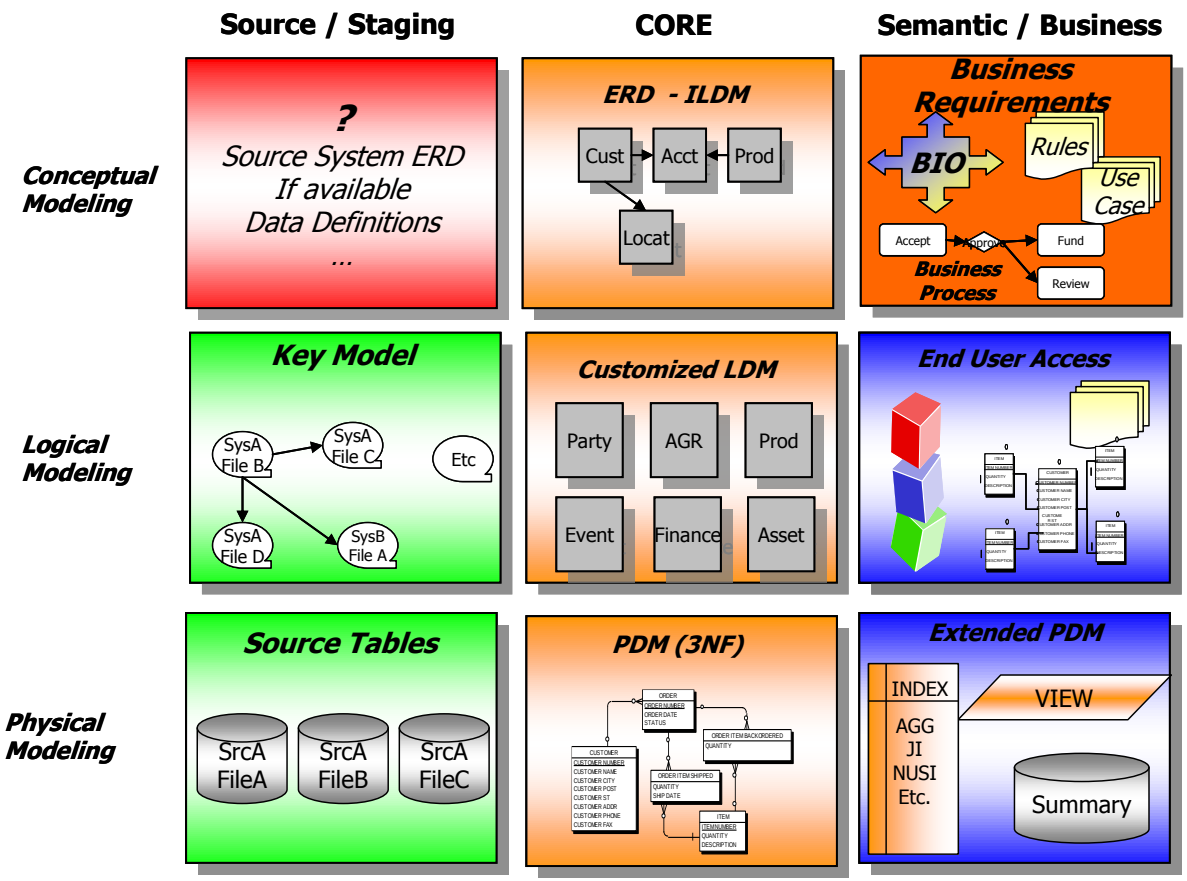
架构立方



Recommend
火龙果软件

架构设计原则

- 简单化
- 纯粹化
- 标准化
- 扩展性
- 保障性
- 一致性



Recommend
火龙果软件

体系架构目的

- 提供一组集中的数据仓库架构集合
  - 一般从概念参考架构和逻辑层次入手
    - > 增强对一个经典数据仓库环境所提供的能力的理解和认识
    - > 建立理想的逻辑设计，以体现架构中各种服务环节的能力
    - > 为将来数据仓库架构的实现、物理化和部署奠定基础

架构定义

- 信息技术 系统架构 (通用定义) – 一个系统各组件的结构，包括它们之间的内部关系，以及一些原则和指导方针，用来管理系统架构内的组件及内部关系的质量、设计和不断变化的过程，使其最终达到一个理想的状态。

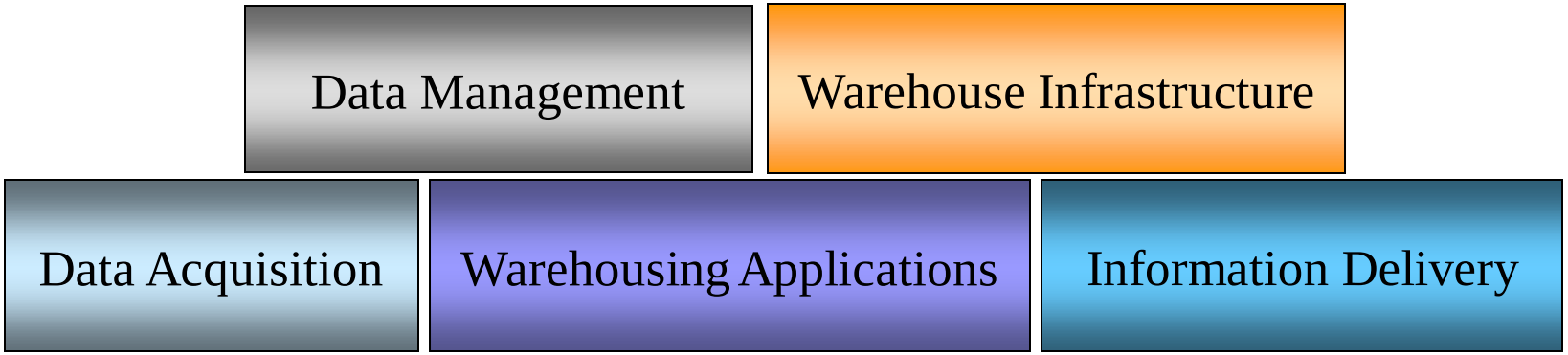
- 数据仓库架构 – 专为满足企业信息需要的一种系统架构，包括定义数据仓库能力，组件、流程以及相应的原则，并为数据仓库设计的实施提供参考和依据。

目标

- 建立一个数据仓库的体系架构:
  - 遵循企业的IT目标，并成为一个可集成多种产品和服务的提供者
  - 可以按照业务运营的水平和方式来支持数据仓库环境的运行和管理
  - 帮助所有用户能够更好地使用数据仓库所提供信息用于提高业务成效，并提供访问的便利性
  - 推动并保证数据仓库体系内的一致性和可重用性
  - 为企业级数据仓库的能力提供文档说明
  - 为适用将来架构的扩展提供一个基准

参考架构描述

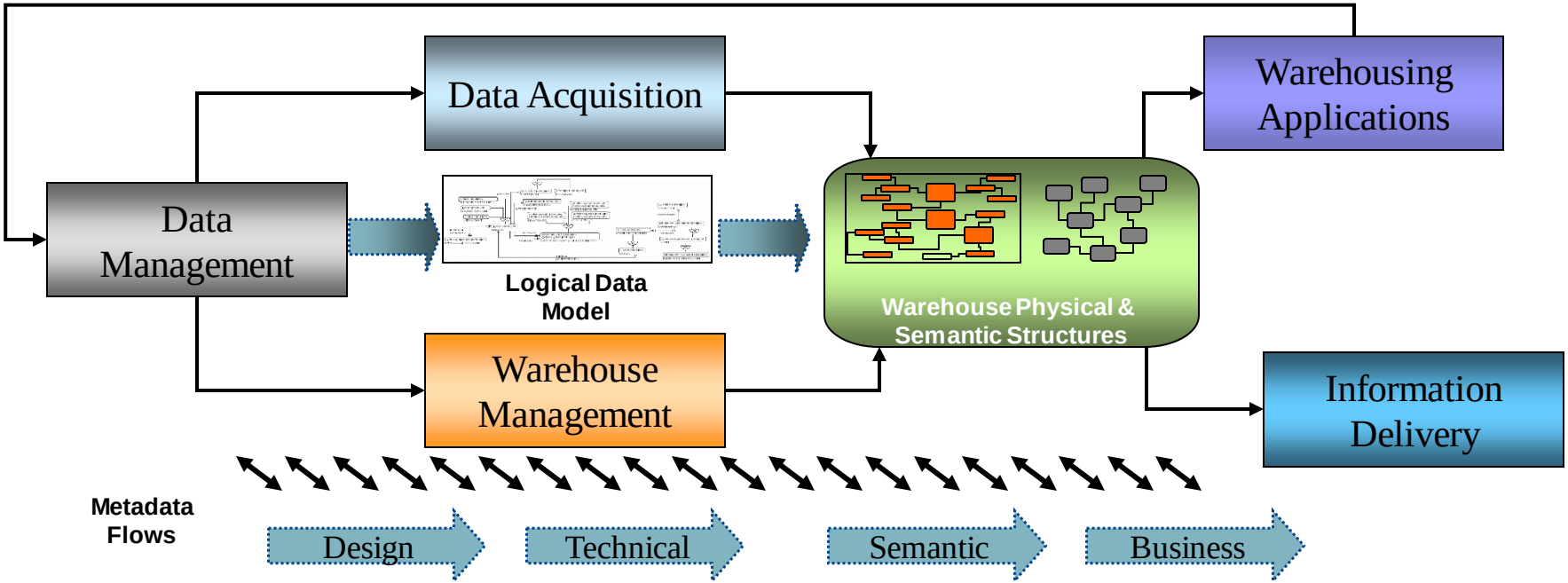
- 参考架构将涵盖一个经典数据仓库环境所提供的主要能力 – 例如，它定义了哪些要素组成了现代数据仓库的环境
- 它建立一套完整的术语和数据仓库面向服务的结构，将为企业内的业务和IT部门所普遍使用
- 在参考架构中建立了五大主要服务类别，如下所示：



Recommend
火龙果软件

逻辑架构描述

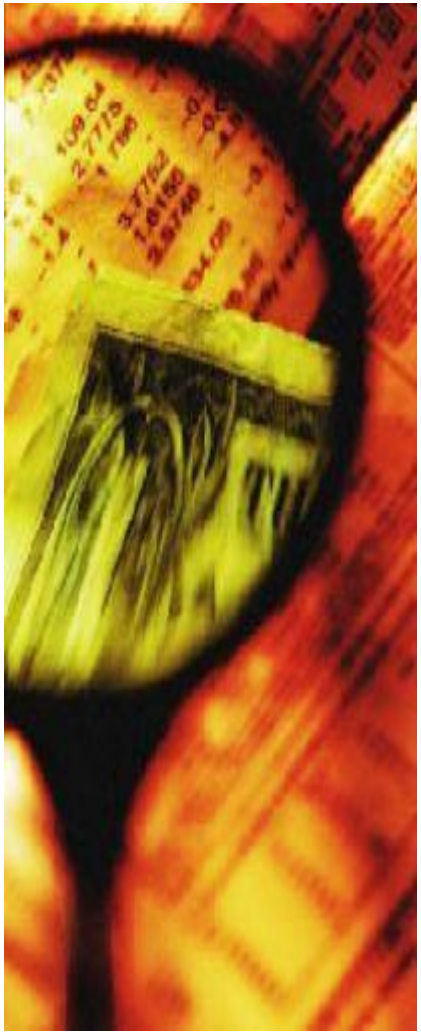
- 逻辑架构的设计用于支持参考架构所定义的服务能力
- 更加关注这些功能组件如何进行整合，以及如何支持环境内的数据流和元数据流
- 逻辑架构为正在进行的数据仓库实现和部署提供一个重要的基础



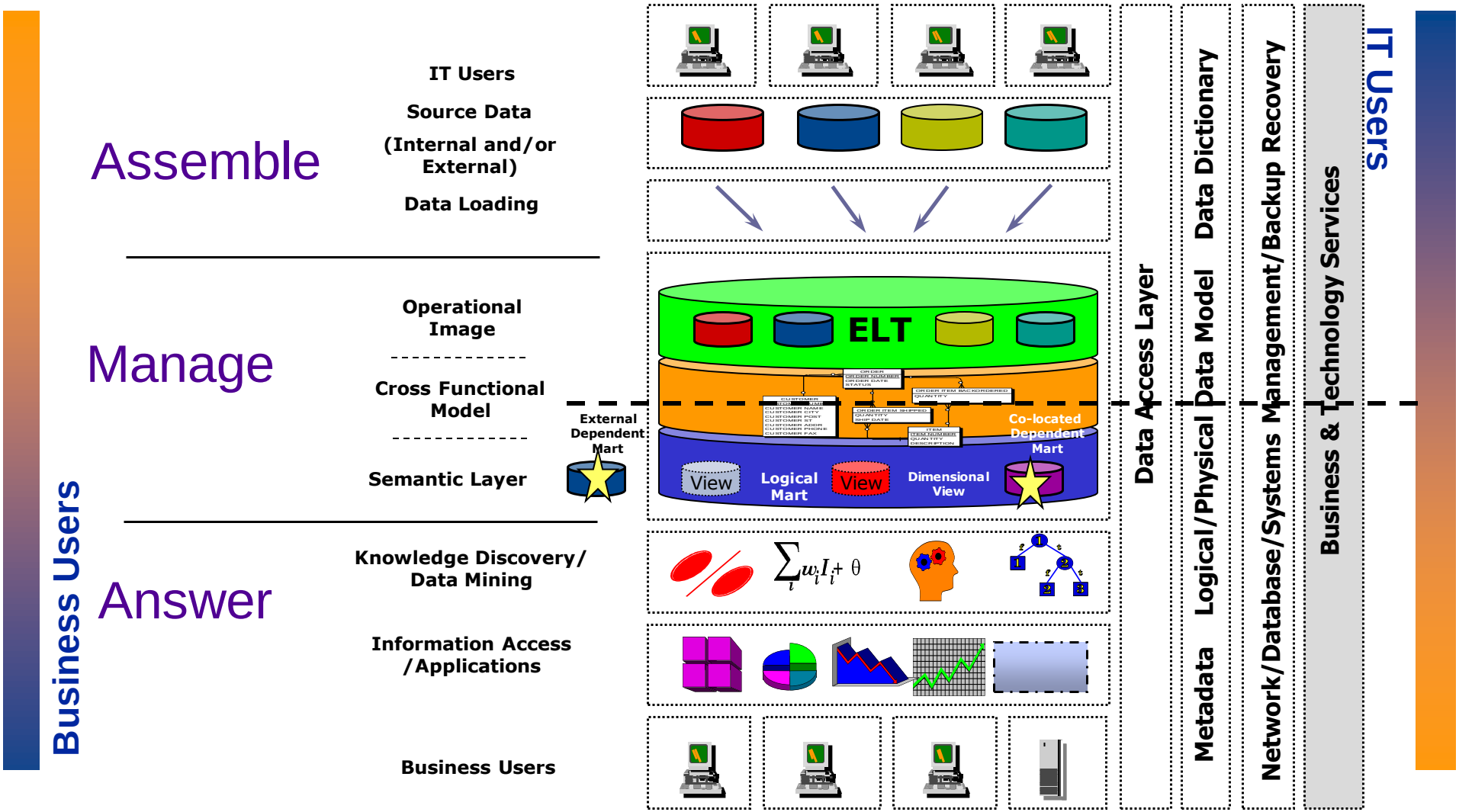
Recommend
火龙果软件

Agenda

- 架构设计原理
- 整体架构说明
- 方案说明
- 物理体系架构
- 架构前瞻

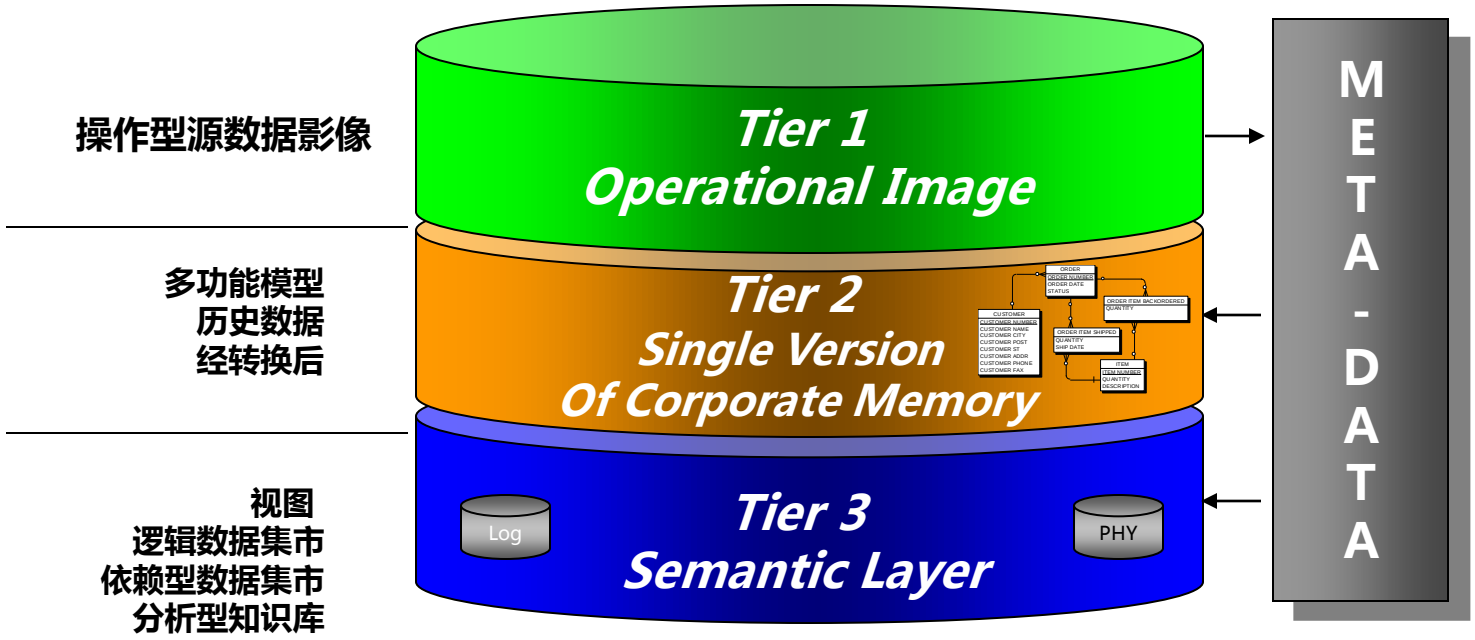


Teradata Warehouse Framework – Since forever…



Recommend
火龙果软件

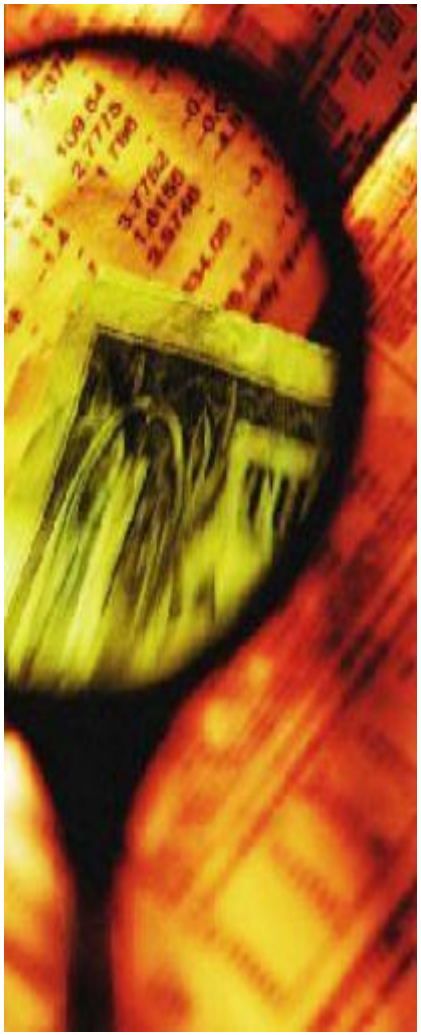
EDW 应用逻辑架构



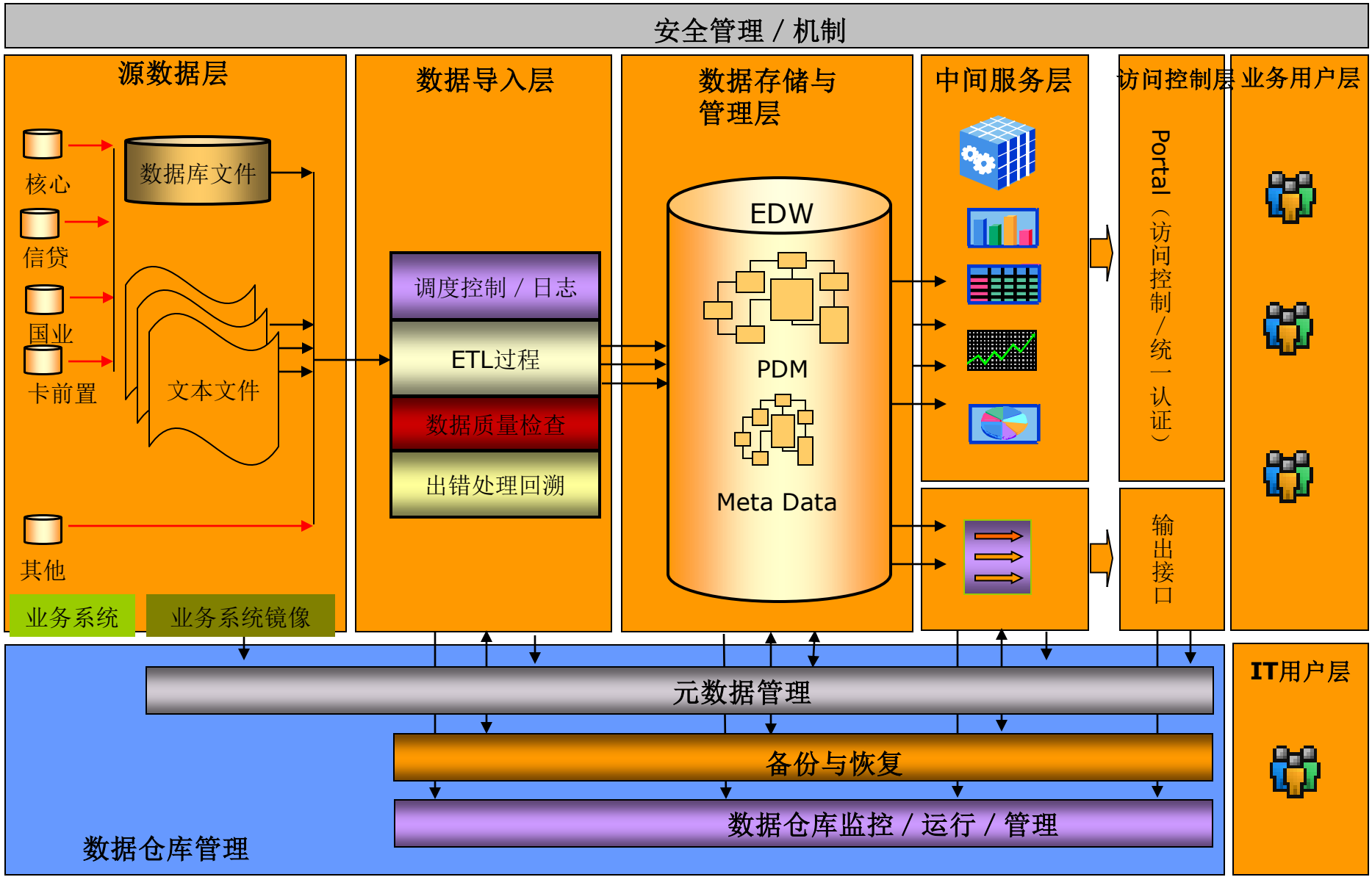
Recommend
火龙果软件

Agenda

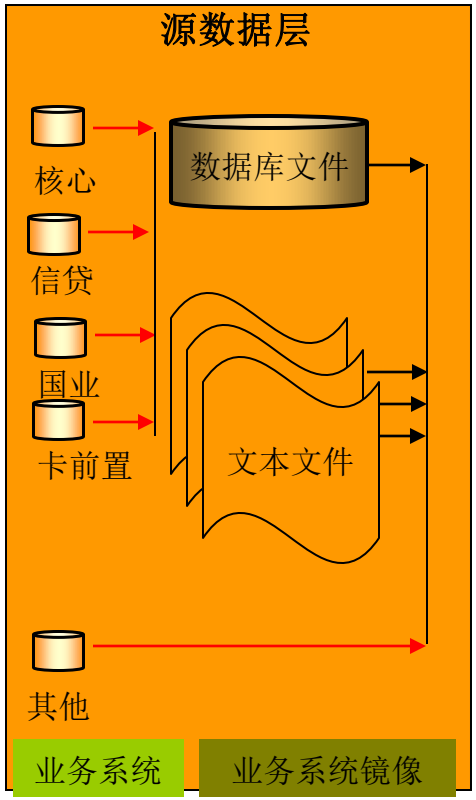
- 架构设计原理
- 整体架构说明
- 方案说明
- 物理体系架构
- 架构前瞻



数据仓库整体技术体系架构设计



1、源数据层



- 【组成部分】
  - > 业务系统
  - > 不同平台
  - > 不同编码
  - > 抽取成文件形式或数据库镜像
- 【存在形式】
  - > 文本文件
  - > 数据库文件
  - > 关系数据库**RDBMS**及其对象如**View**（在线数据）

关键问题：增量数据抽取方式

- 工作范围
  - > 定义源数据范围
  - > 确定源数据接口
- 挑战
  - > 数据抽取或影响业务系统性能
  - > 业务系统无法抽取出正确的增量数据
  - > 业务系统抽取全量数据，时间窗口无法保证
- 解决方案
  - > 业务系统提供增量数据
  - > 或者，业务系统提供全量数据
    - 在Teradata数据库内比对找出增量
  - > 或者，业务系统直接提供全量数据库文件（中国证券登记结算公司）
    - 由ETL系统解码后加载到Teradata数据库内
    - 利用Teradata强大的并行性能进行比对，从而找出正确的增量数据

2、数据导入层 (ETL)

- 【功能与作用】
  - > 从源数据层的数据抽取，采用文本文件或数据库文件
  - > 数据加载采用Teradata的数据加载工具Fastload
  - > 整个ETL流程管理采用Teradata的DW Automation工具。
  - > ETL服务器配置两台，互为备份，同时也承担数据输出/分发的工作
- 【组成部分】
  - > ETL调度与日志跟踪管理模块
  - > ETL运行脚本模块
  - > 数据质量检查模块
  - > 出错处理与回溯模块

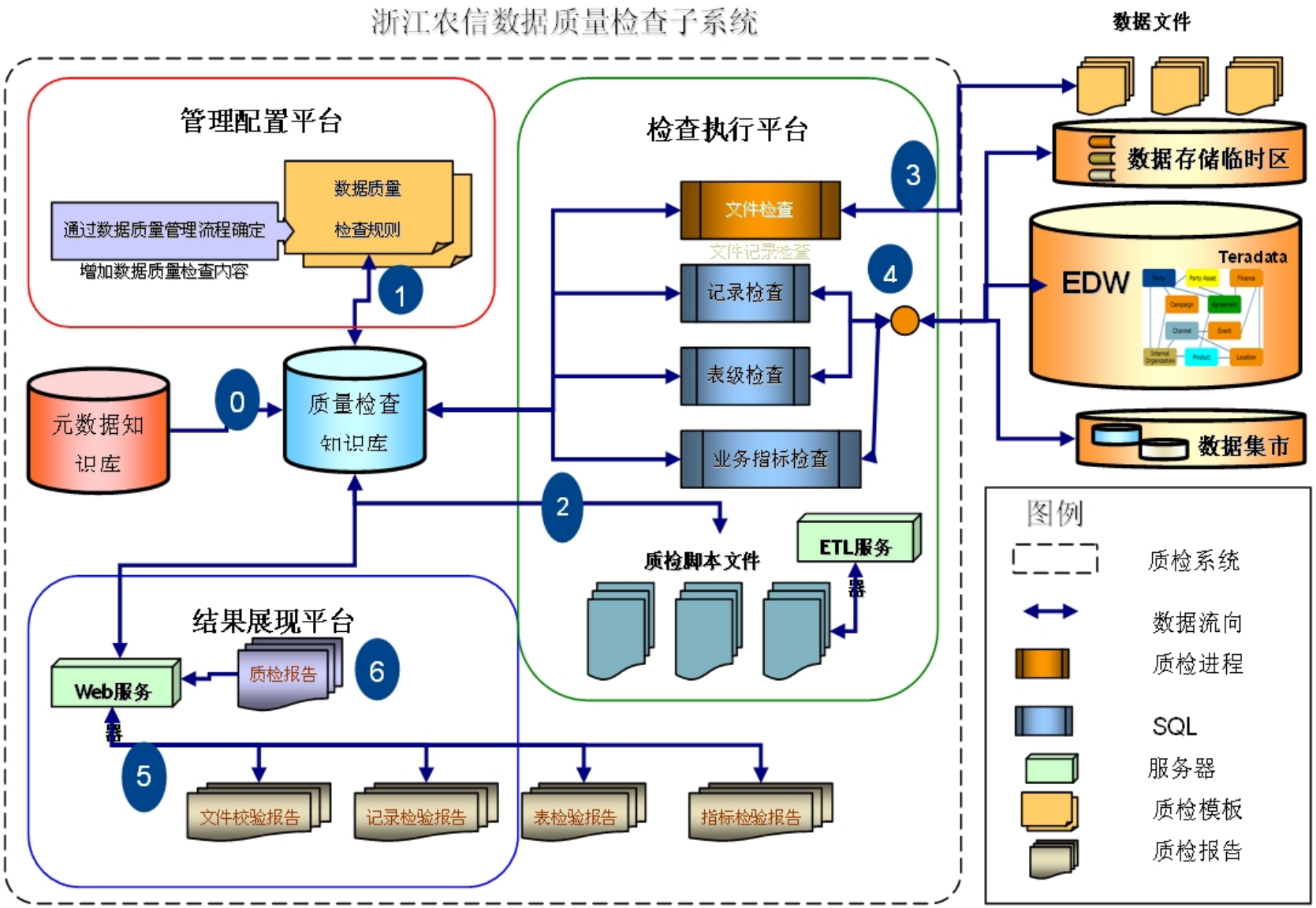
关键问题：ETL开发工作量与数据质量控制

- 挑战
  - > ETL开发工作量巨大，如何有效降低工作量，提高生产效率
  - > 如何有效管理控制ETL流程
  - > 如何实施数据质量管理
- 解决方案
  - > Teradata Minerva
  - > DW Automation
  - > DQC

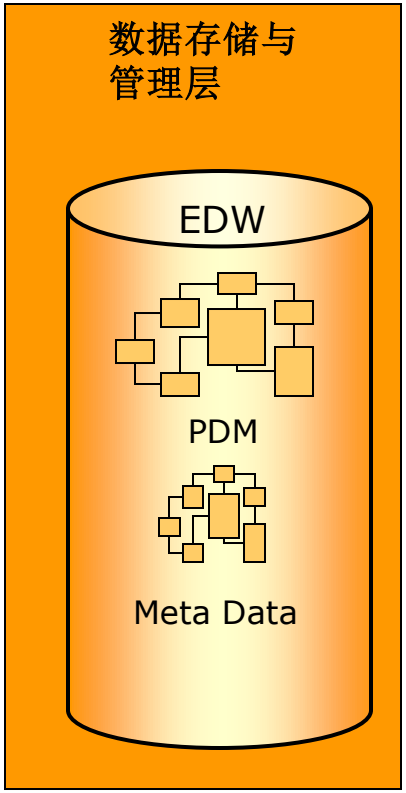
ETL脚本自动生成工具Minerva与流程管理工具DW Automation

- **Teradata Minerva**
  - > 全自动的**ETL**脚本生成工具
  - > 只需在完成源系统调研后，基于调研成果的**ID**报告**(Excel)**，可直接生成后续**ETL**脚本
  - > 包括**ETL**转换脚本、数据质量检查脚本等
  - > 可极大提高**ETL**开发效率和质量
  - > 便于后续运行维护，源系统发生变更时，只需修改**ID**报告，就可由**Minerva**完成后续所有相关**ETL**脚本的更改工作
- **DW Automation**
  - > **Teradata**的**ETL**运行调度与监控工具
  - > 不仅涵盖了**ETL**过程，还涵盖了数据库备份等后续运行调度工作
  - > 提供可视化管理和监控界面，并可方便地完成出错处理和回溯处理

数据质量管理总体架构 - DQC



3、数据存储与管理层 (中央数据库)

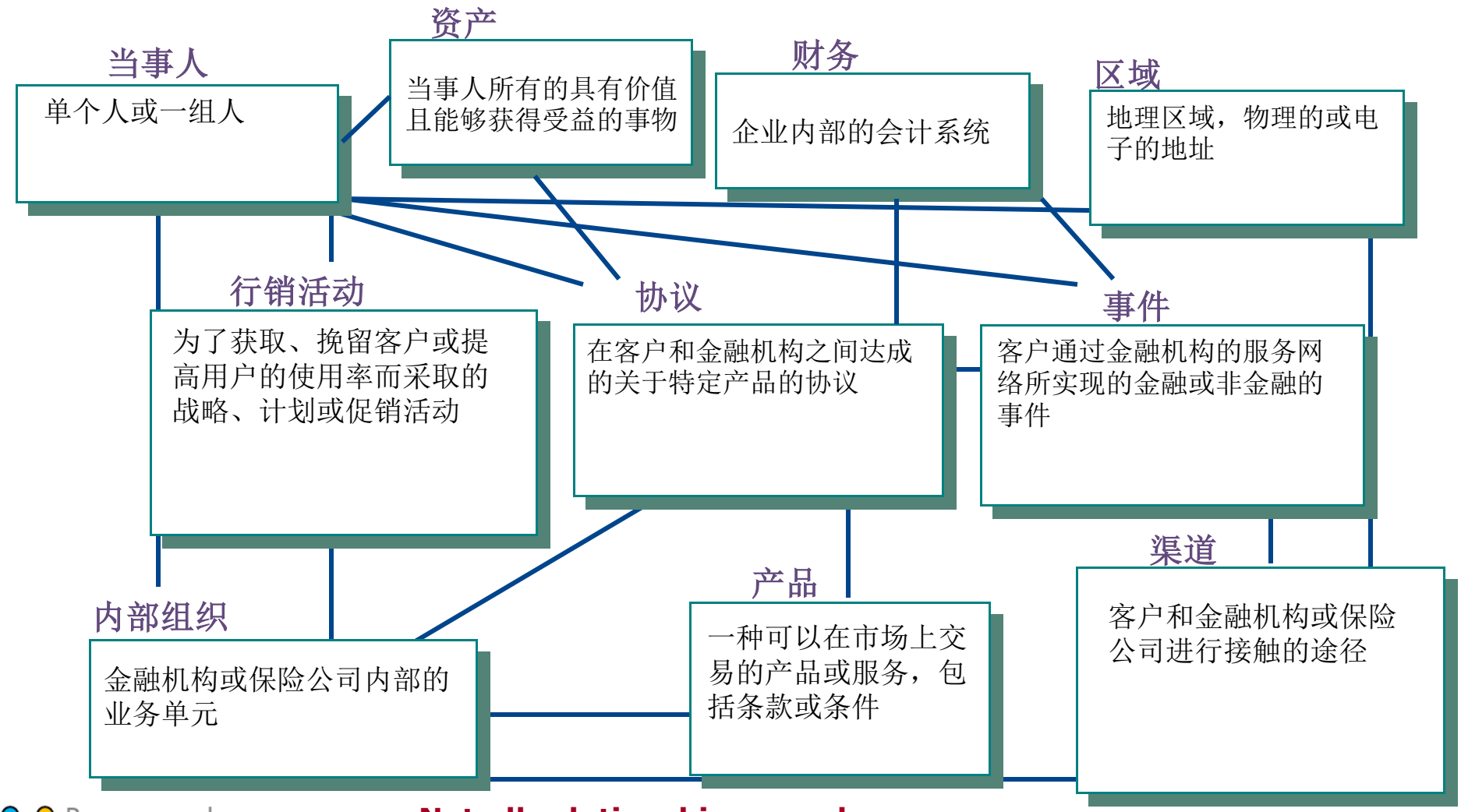


- 【功能与作用】
  - > 负责存储和管理来自各种源数据系统的数据，并为访问用户提供数据服务
  - > 数据按照企业级逻辑数据模型分主题存放
  - > 采用**Teradata**关系型数据库
- 【组成部分】
  - > 采用**Teradata**的**FS-LDM**作为模型设计的基础
  - > 工具采用**ERWin**，客户化**LDM**和**PDM**。同时也包括维护和管理数据模型各个版本的过程与方法
  - > 采用**Teradata**数据库以及**5500**系列硬件平台和磁盘阵列

关键问题：数据模型设计

- 挑战
  - > 如何确保数据模型的稳定性和灵活性
  - > 数据粒度如何划分
- 解决方案
  - > 采用FS-LDM，为国内同业广泛采用
    - 最符合国内本地化实践的数据模型
    - 业内最富实践经验的模型设计师
  - > 合理规划数据存储区域与存储粒度

Teradata金融业逻辑数据模型FS-LDM 主题区域



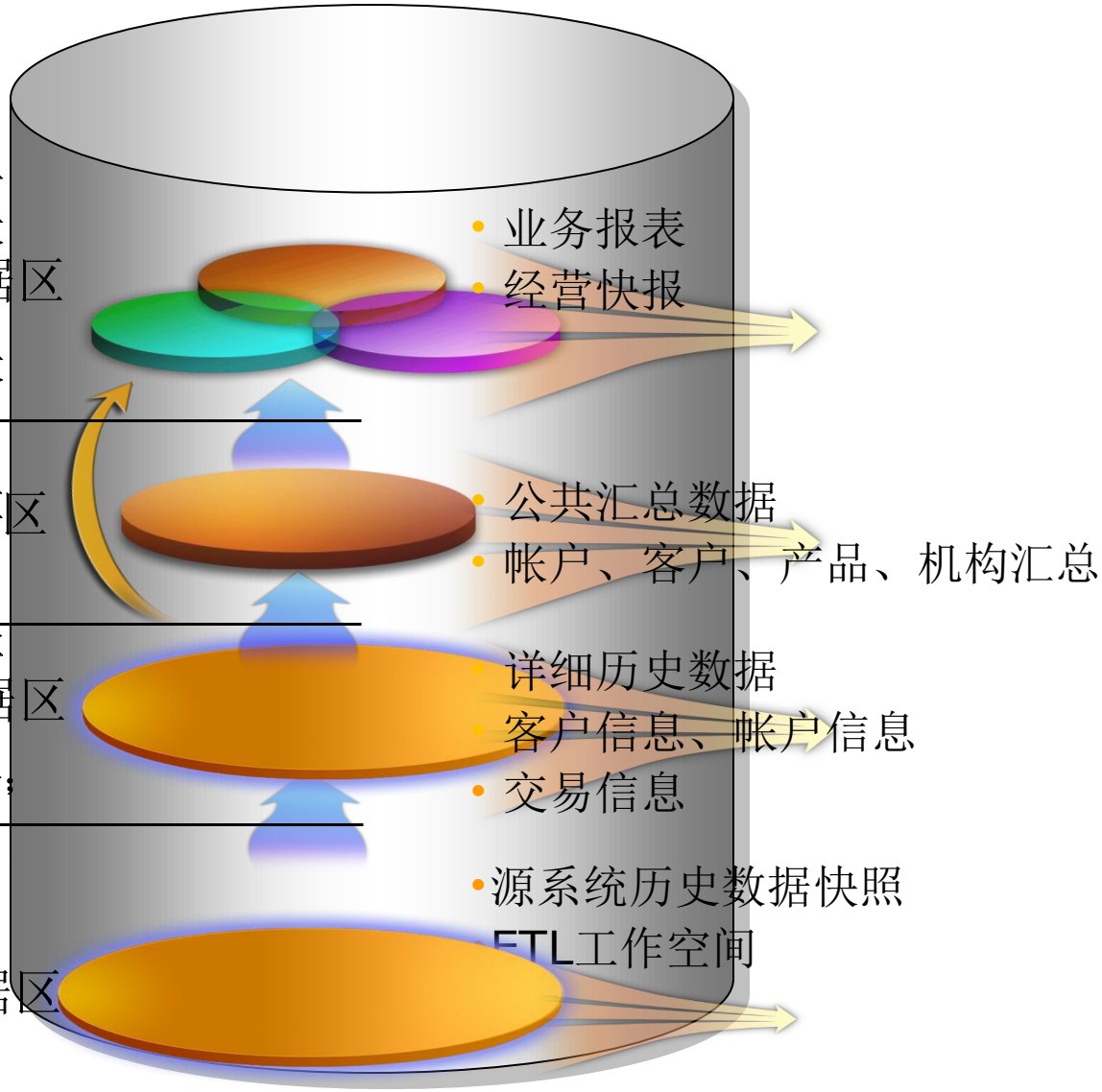
Not all relationships are shown

Recommend
火龙果软件

数据存储区域与粒度规划

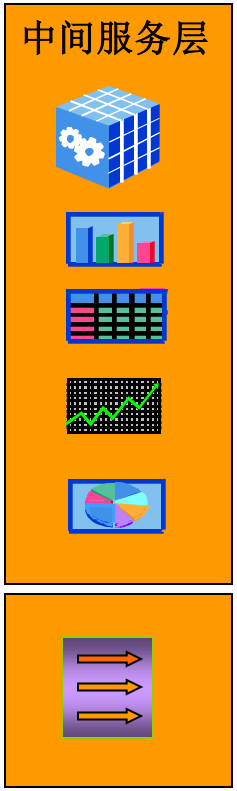
数据存储区域与粒度规划

- 临时数据的物理数据模型：用于在**ETL**处理过程中临时保存各类业务的原始明细数据；
- 详细数据的物理数据模型：用于保存各类业务的明细数据；
- 中间数据的物理数据模型：用于保存各类业务的粒度汇总数据，业务指标等；
- 面向应用的物理数据模型：用于保存各类面向分析型业务应用的汇总数据；
- **Meta Data**元模型：用于存储元数据管理中的元数据；



Recommend
火龙果软件

4、中间服务层 (业务应用)



- 【功能与作用】
  - > 向用户提供包括**OLAP**服务、报表服务、查询服务、数据挖掘服务和中间件服务等多种服务应用
  - > 为用户对中央数据的访问提供各种方式的服务，从而实现访问方式的多样化和信息存取的透明化
- 【组成部分】
  - > 报表与查询工具（如：**BIEE Plus**）
  - > **OLAP**工具（如：**Essbase**）
  - > 数据挖掘采用**TeraMiner**
    - 直接在数据库中挖掘
    - 直接在数据库中应用模型
    - 消除数据转移、简化数据管理、开发更为精确模型、缩短模型开发和部署的周期、充分利用**Teradata**的平行处理能力和扩展能力

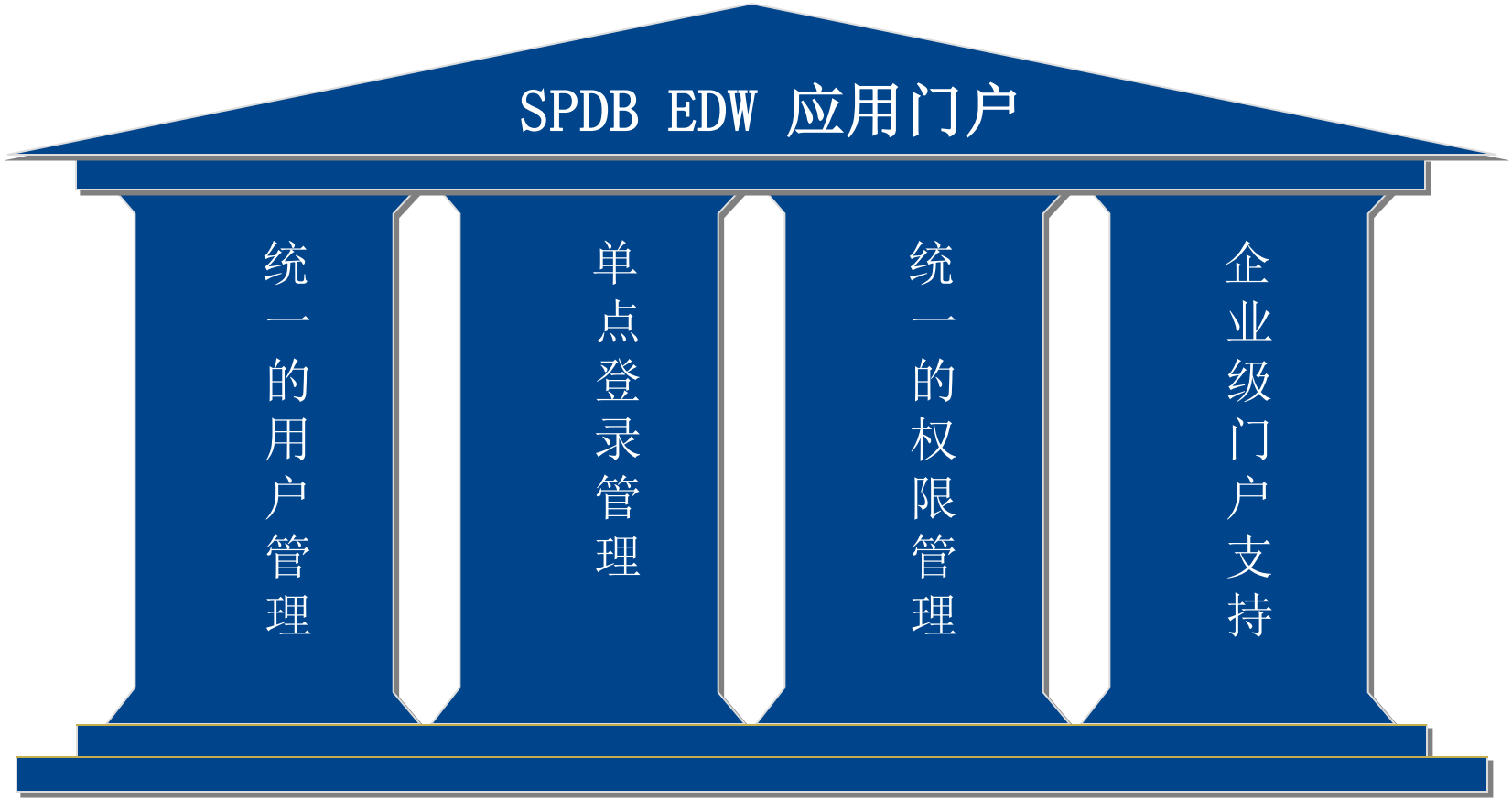
5、访问控制层

访问控制层

Portal（访问控制／统一认证）

- 【功能与作用】
  - > 主要包括**WEB**、认证、安全、门户四方面的服务。提供**HTTP Web**服务、门户的用户登录、用户认证、门户**Web**应用、并提交用户层请求到中间服务层，对用户实施安全策略，为用户管理报表、查询文档，提供个性化定制等。
- 【组成部分】
  - > **Web Server(**如**Apache)**
  - > 应用服务器**(**如：**Weblogic**）
  - > 实现单点登录和用户安全认证

5.1 EDW 应用门户



5.2 EDW应用门户与企业门户的集成

界面集成

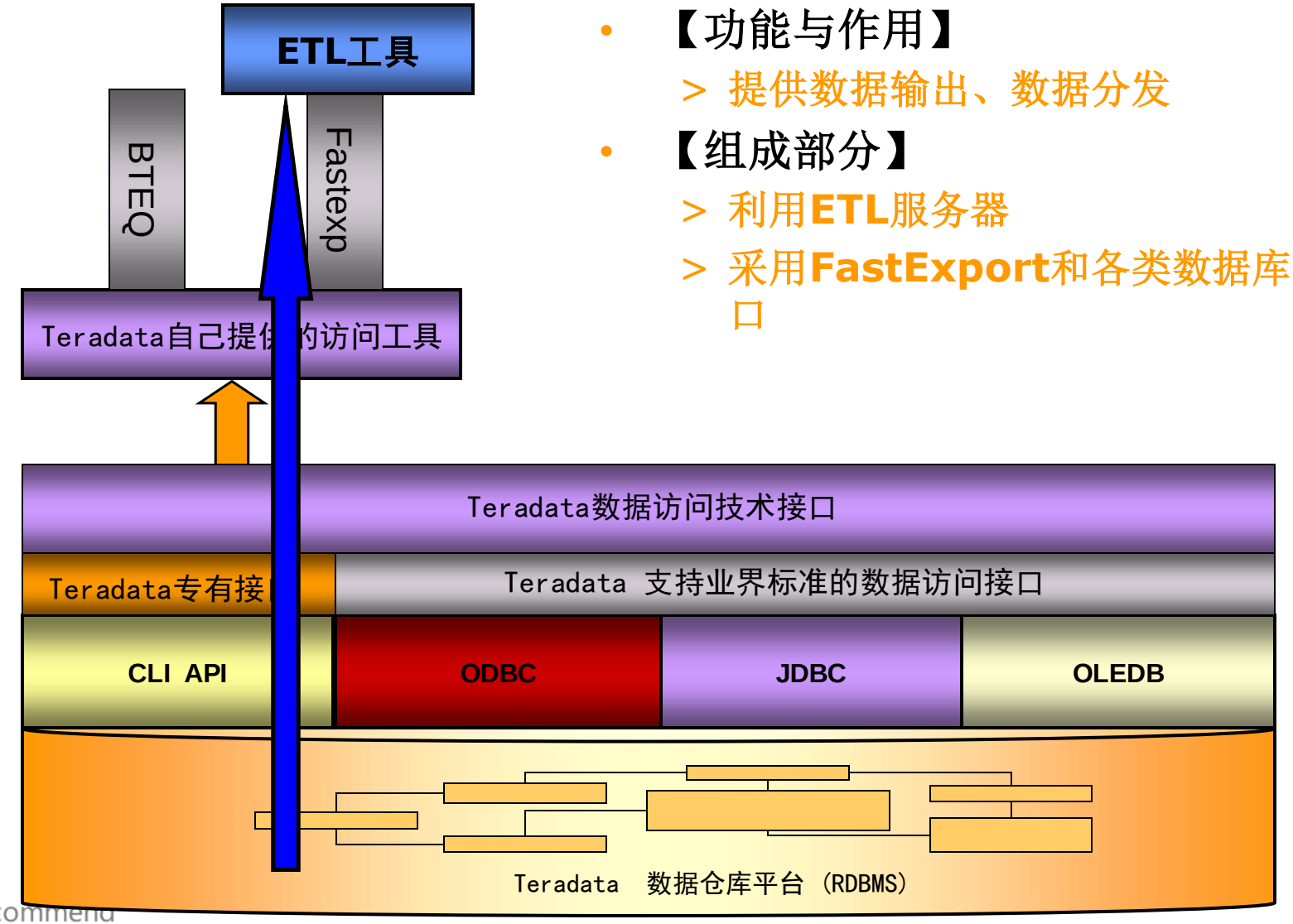
用户集成

权限集成

单点登录

安全审计

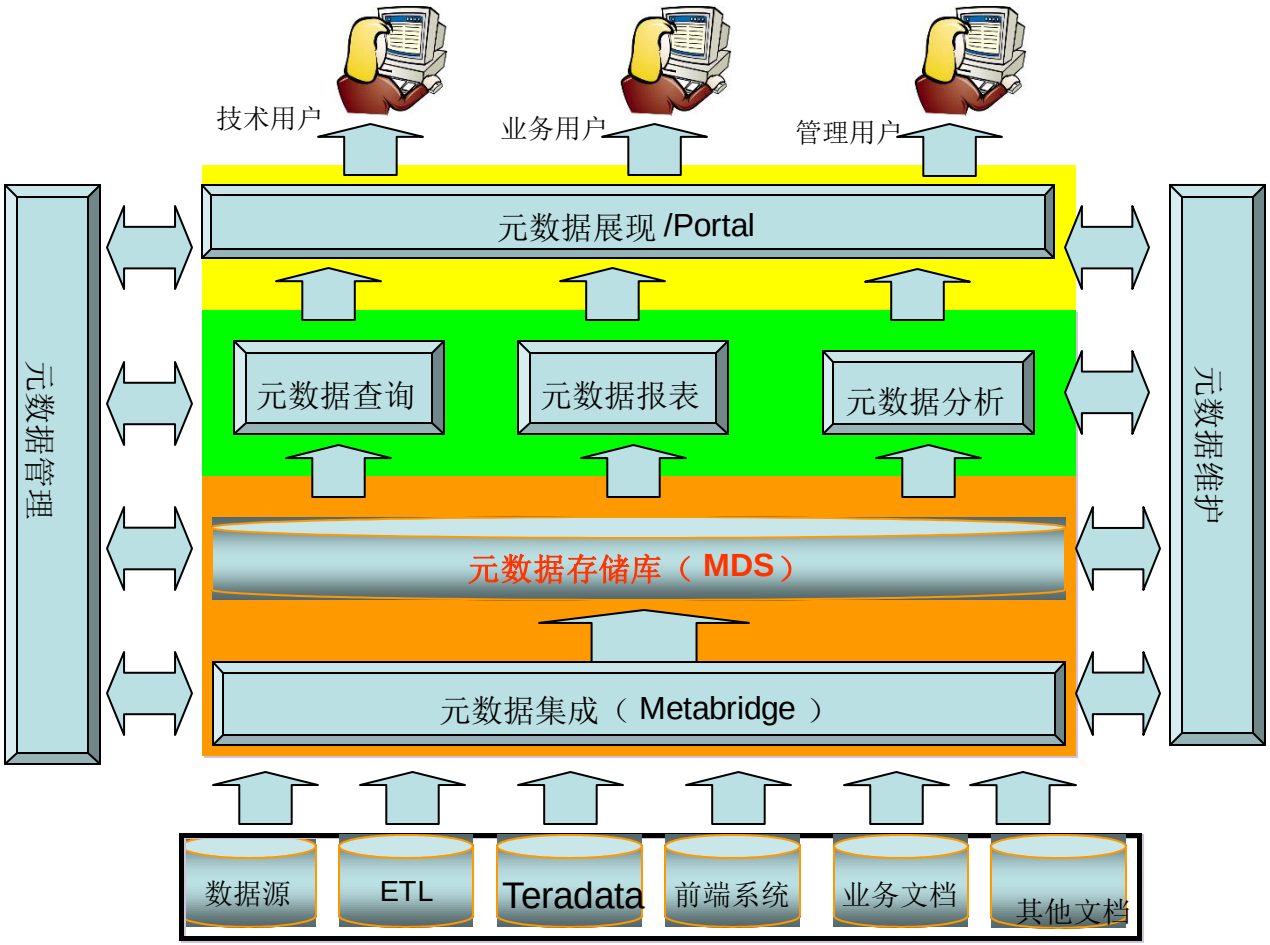
6、数据输出/分发接口



- 【功能与作用】
  - > 提供数据输出、数据分发
- 【组成部分】
  - > 利用**ETL**服务器
  - > 采用**FastExport**和各类数据库接口

Recommend
火龙果软件

7、元数据管理



- 【功能与作用】
  - > 提供元数据抽取、管理、应用的体系架构
- 【组成部分】
  - > 元数据管理采用**Meta Data Service**

8、用户安全管理及系统安全管理

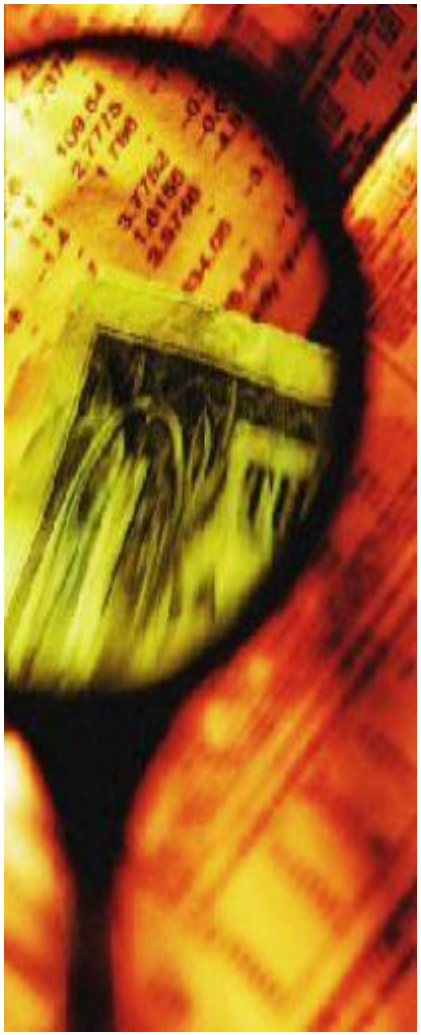
- 用户分类
  - > 业务用户
    - 决策管理层用户（决策用户）
    - 部门专业用户（一般用户）
    - 信息分析型用户（超级用户）
  - > 技术用户
  - > 操作用户
- 用户管理
  - > 用户分组
  - > 用户级别
- 用户安全管理
  - > 操作系统的安全控制
  - > 数据权限控制
  - > 数据库权限控制
  - > 前端功能的用户管理和访问权限控制
- 用户日志及审计
- 数据转移安全
  - > 网络安全与协议
  - > 操作系统安全
  - > 软件工具安全
- 数据存储安全
  - > **MPP**架构
  - > 物理安全、用户登录与数据存取
  - > 数据保护 – 磁盘阵列**RAID**技术、**Teradata**数据库级数据保护
  - > 备份与恢复
- 数据访问安全
  - > 数据库级访问 – 登录管理、存取权限、视图对象、存取日志

9、系统管理与维护

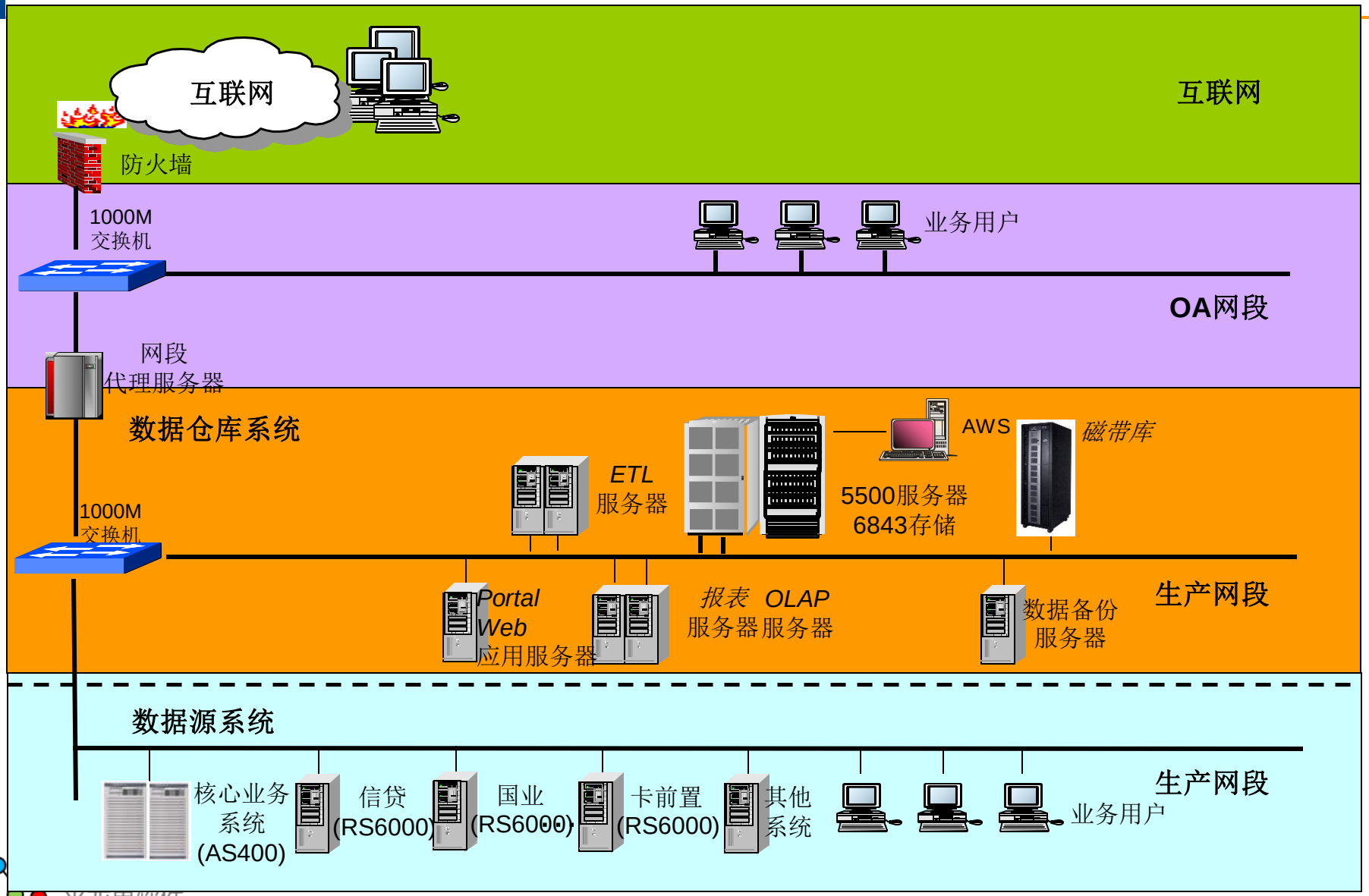
- 系统管理与维护
  - > 元数据(Meta Data)管理
  - > 操作管理与维护(OA&M)
  - > 数据库管理
  - > 数据字典
  - > 逻辑数据模型
  - > 物理数据模型
  - > 网络管理
  - > 系统管理
  - > 数据管理
  - > 工作负载管理
  - > 用户与安全管理
  - > 专业技术服务与咨询

Agenda

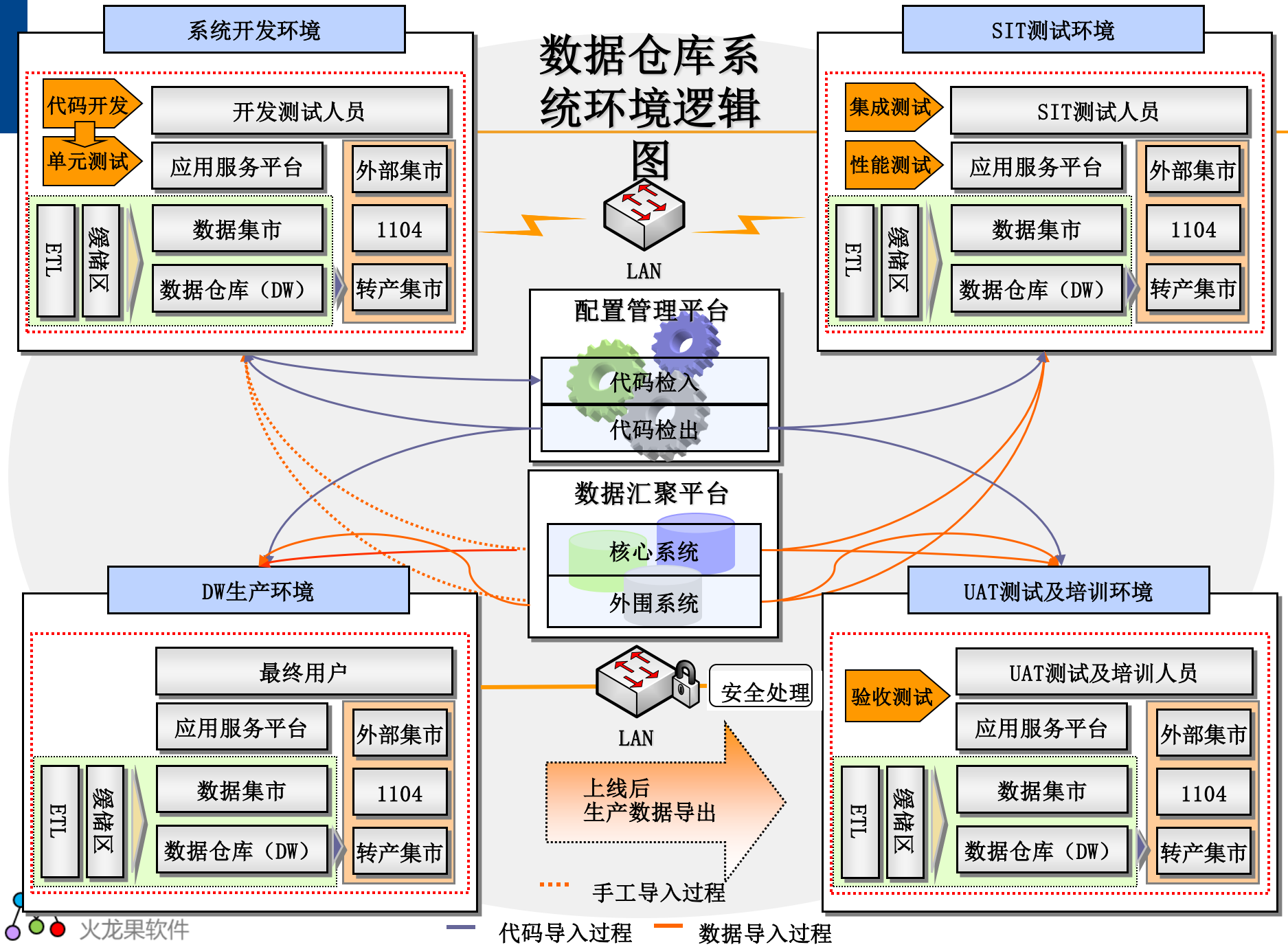
- 架构设计原理
- 整体架构说明
- 方案说明
- 物理体系架构
- 架构前瞻



生产系统总体物理架构

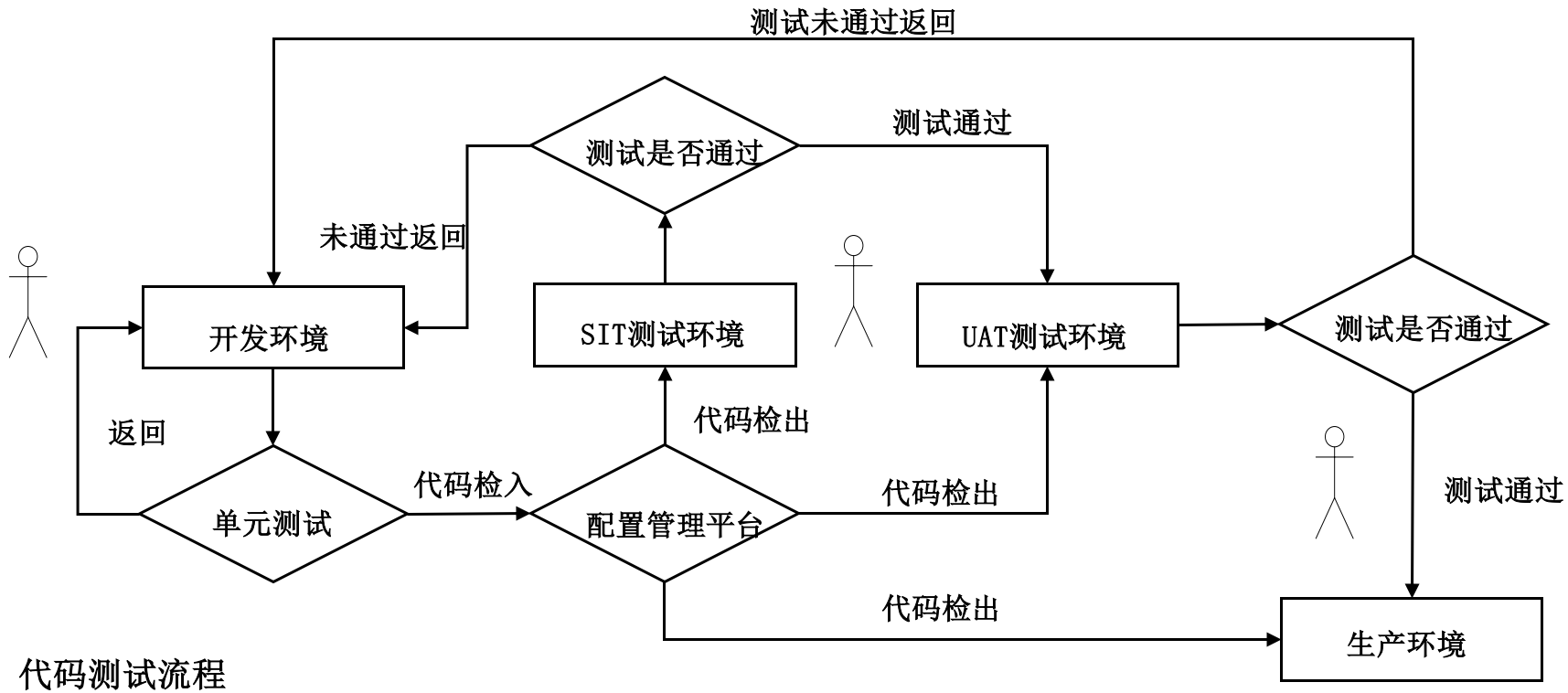

互联网
互联网
防火墙
1000M
交换机
业务用户
OA网段
网段
代理服务器
数据仓库系统
AWS
磁带库
ETL
服务器
5500服务器
6843存储
1000M
交换机
Portal
Web
应用服务器
报表
服务器
OLAP
服务器
数据备份
服务器
生产网段
数据源系统
核心业务
系统
(AS400)
信贷
(RS6000)
国业
(RS6000)
卡前置
(RS6000)
其他
系统
业务用户
生产网段

数据仓库系统环境逻辑图



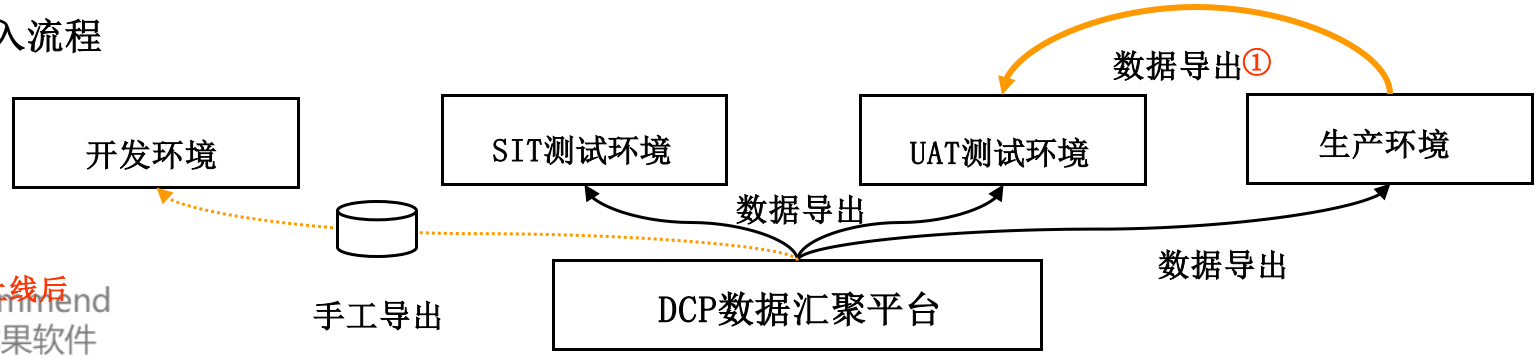
火龙果软件

开发测试代码及数据迁移流程



代码测试流程

数据导入流程



①一期上线后
Recommend
火龙果软件

数据仓库系统环境说明

- 数据仓库系统实施过程中包含四个环境：开发环境、集成测试环境、UAT测试环境、生产环境，这四个环境构成数据仓库的总体技术架构中的技术环境配置，将会在未来数据仓库整个生命周期中一直存在；
- 在四种环境之间的代码的检入和检出均通过项目的配置管理平台，无法使用项目配置管理平台的开发过程，则通过手工将代码在四个平台之间迁移；
- 在系统进行测试过程中，如果发现代码中存在缺陷，则开发人员需要将这些代码从配置管理平台中检出到开发环境中进行修改，并在测试完成之后再次检入到配置管理平台中供测试系统进行继续测试；
- 对于开发环境来说，不仅仅是完成编码开发的环境，同时开发人员需要在开发环境中完成单元测试的过程；
- 对于系统集成测试环境来说，主要为了完成系统整体功能的测试，同时在这个环境中也可以完成系统性能和压力测试的过程；
- 对于UAT环境来说还将作为整个系统的培训环境和系统最终验收需要的环境；

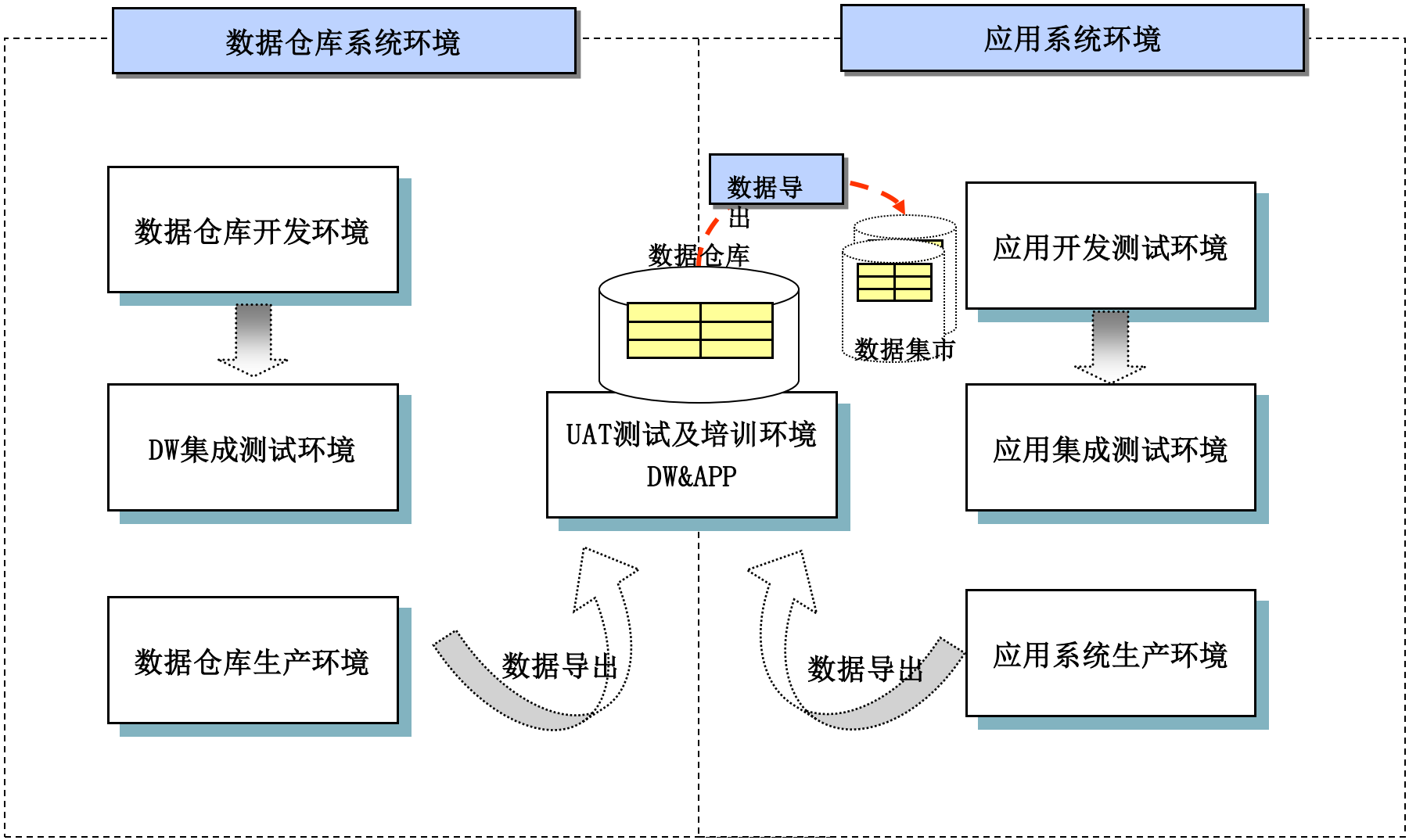
数据仓库系统环境说明（续）

- 开发环境中的资源的需求要满足各个数据仓库和应用开发的需要，如：ETL开发环境、报表开发环境；
- 集成测试环境对资源的需求与开发环境相同，并且两个环境的网络需要相互通讯；
- UAT测试环境的建立参照最终的生产环境，UAT环境的容量需求依赖于被测试的应用的要求。 UAT测试应利用生产环境中的实际数据进行相关的UAT测试工作；
- 在开发环境，开发人员之间可以通过网络进行通讯，通过开发环境需要与项目具体的配置管理平台进行通讯；

数据仓库与应用系统环境之间的关系

- 在DW 系统中，可以划分为两个环境，即：数据仓库环境、应用系统环境；在每个环境中都包含：开发环境、集成测试环境、UAT环境和生产环境；
- 数据仓库环境是作为应用开发环境的数据基础平台，应用开发要利用数据仓库生产环境中提出的数据进行开发；
- 应用系统和数据仓库系统需要共享UAT测试环境，UAT的数据来自数据仓库生产环境；
- UAT环境需要和数据仓库生产环境以及应用开发生产环境进行网络通讯；
- 数据的导入、导出通过网络传输的方式来进行；

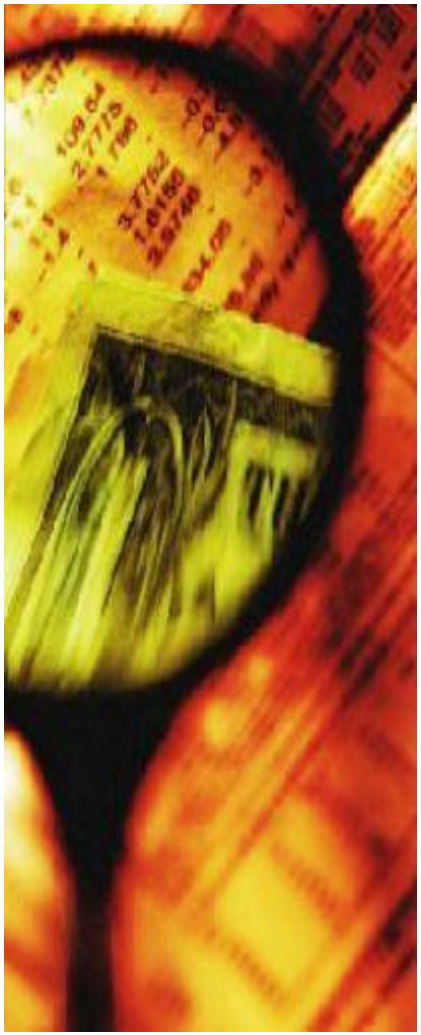
数据仓库与其所支持的应用系统开发及测试环境之间数据依赖关系



Recommend
火龙果软件

Agenda

- 架构设计原理
- 整体架构说明
- 方案说明
- 物理体系架构
- 架构前瞻



多温度的数据仓库体系

- 频繁访问
- 更多面向操作型智能
- 涵盖较近的历史

- 可临时调解
- 趋势分析
- 深度历史

决策型智能

决定温度的因素:

- 数据访问的频度
- 数据更新的频度
- 数据的维护
- 历史的深度

多温度的技术优势

- 通过多温度技术
  - > 能够充分利用最新的磁盘技术而提高的容量
  - > 为不同的用户保持不同的**SLAs**
  - > 根据业务规则和对系统工作负载的分析对系统的资源的使用优先级顺序进行区分
  - > 关注管理工作负载，而不是数据

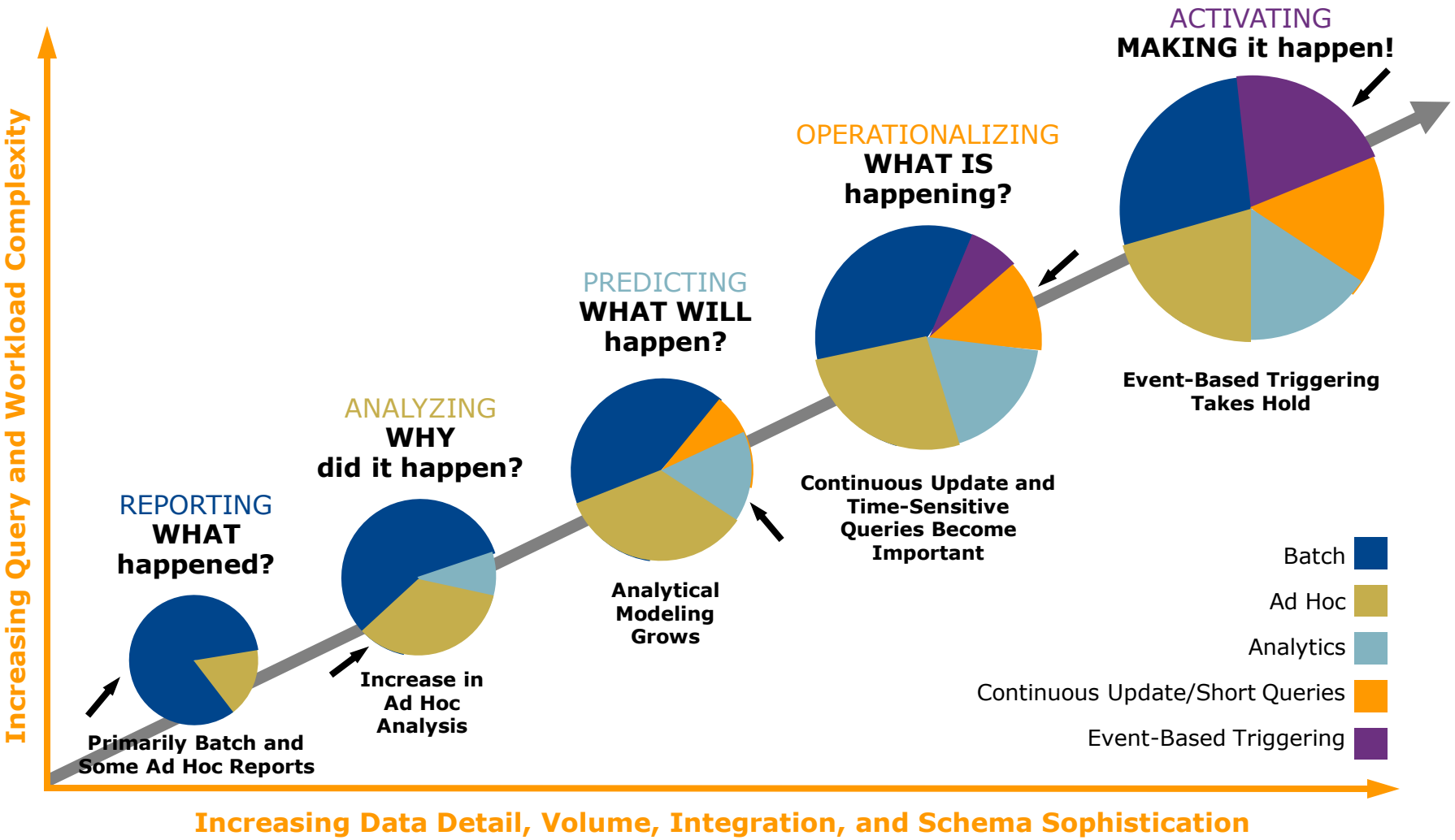
Recommend
火龙果软件

数据仓库多温度技术

- 多温度数据仓库：能够基于业务规则对系统资源的使用设定优先级，同时最大化存储的使用
- **Teradata通过以下技术实现多温度：**
  - > Partitioned primary index (PPI)
  - > Join Index (including pre-joined, aggregate and sparse indexes)
  - > Multi-value compression
  - > Priority Scheduler
  - > Teradata Active System Management (TASM)
    - **Teradata Workload Analyzer**
    - **Teradata Dynamic Workload Manager**
    - **Teradata Manager Dashboard and Trend Analysis**
    - **The Regulator**
    - **Priority scheduler**
    - **Workload classification and management by individual query attributes**
  - > **Large Disk support**

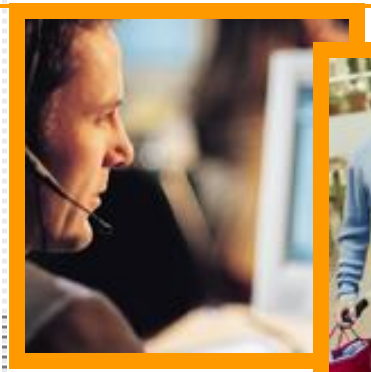
It's all about workload management!

Comparing DW Evolution to DW Maturity



Recommend
火龙果软件

Active Enterprise Intelligence



Strategic Intelligence

BI Tools

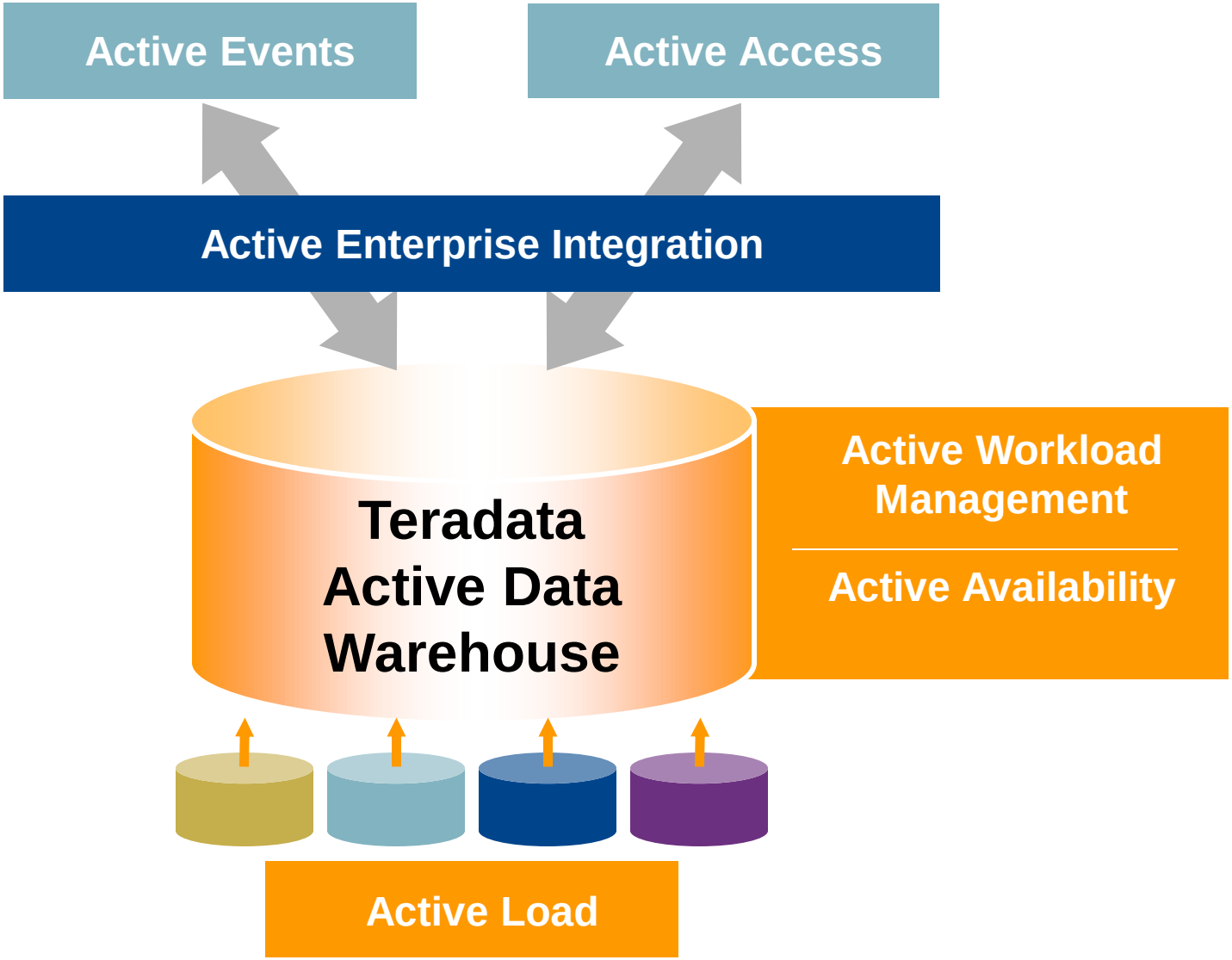
Operational Intelligence

Active Infrastructure

Teradata Active Data Warehouse

Recommend
火龙果软件

Active Data Warehouse Elements



Recommend
火龙果软件

Database Workload Continuum

Transactional (OLTP)

User profiles
- Customers
- Clerks

Services
- Transactions
- Bookkeeping

Access profile
- Frequent updates
- Occasional lookup

Data
- Current "state" data
- Limited history
- Narrow scope

Tactical (ODS)

User profiles
- Front line services
- Customers – indirectly

Services
- Lookups
- Tactical decisions
- Analytics (e.g. scoring)

Access profile
- Continuous updates
- Frequent lookups

Data model
- Current "state" data
- Recent history
- Integrated business areas

Strategic (EDW)

User profiles
- Back office services
- Management
- Trading partners

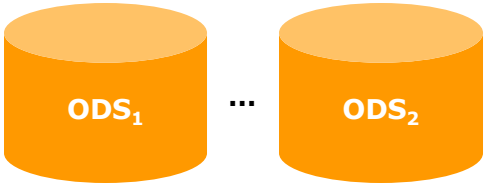
Services
- Strategic decisions
- Analytics (e.g. scoring)

Access profile
- Bulk inserts – some updates
- Frequent complex analytics

Data model
- Periodic "state" data
- Deep history
- Enterprise integrated view

Workload Continuum

Transactional Repositories



Tactical Decision Repositories

Strategic Decision Repositories

Recommend
火龙果软件

Database Workload Continuum

Transactional (OLTP)

User profiles
- Customers
- Clerks

Services
- Transactions
- Bookkeeping

Access profile
- Frequent updates
- Occasional lookup

Data
- Current "state" data
- Limited history
- Narrow scope

Tactical (ODS)

User profiles
- Front line services
- Customers – indirectly

Services
- Lookups
- Tactical decisions
- Analytics (e.g. scoring)

Access profile
- Continuous updates
- Frequent lookups

Data model
- Current "state" data
- Recent history
- Integrated business areas

Strategic (EDW)

User profiles
- Back office services
- Management
- Trading partners

Services
- Strategic decisions
- Analytics (e.g. scoring)

Access profile
- Bulk inserts – some updates
- Frequent complex analytics

Data model
- Periodic "state" data
- Deep history
- Enterprise integrated view

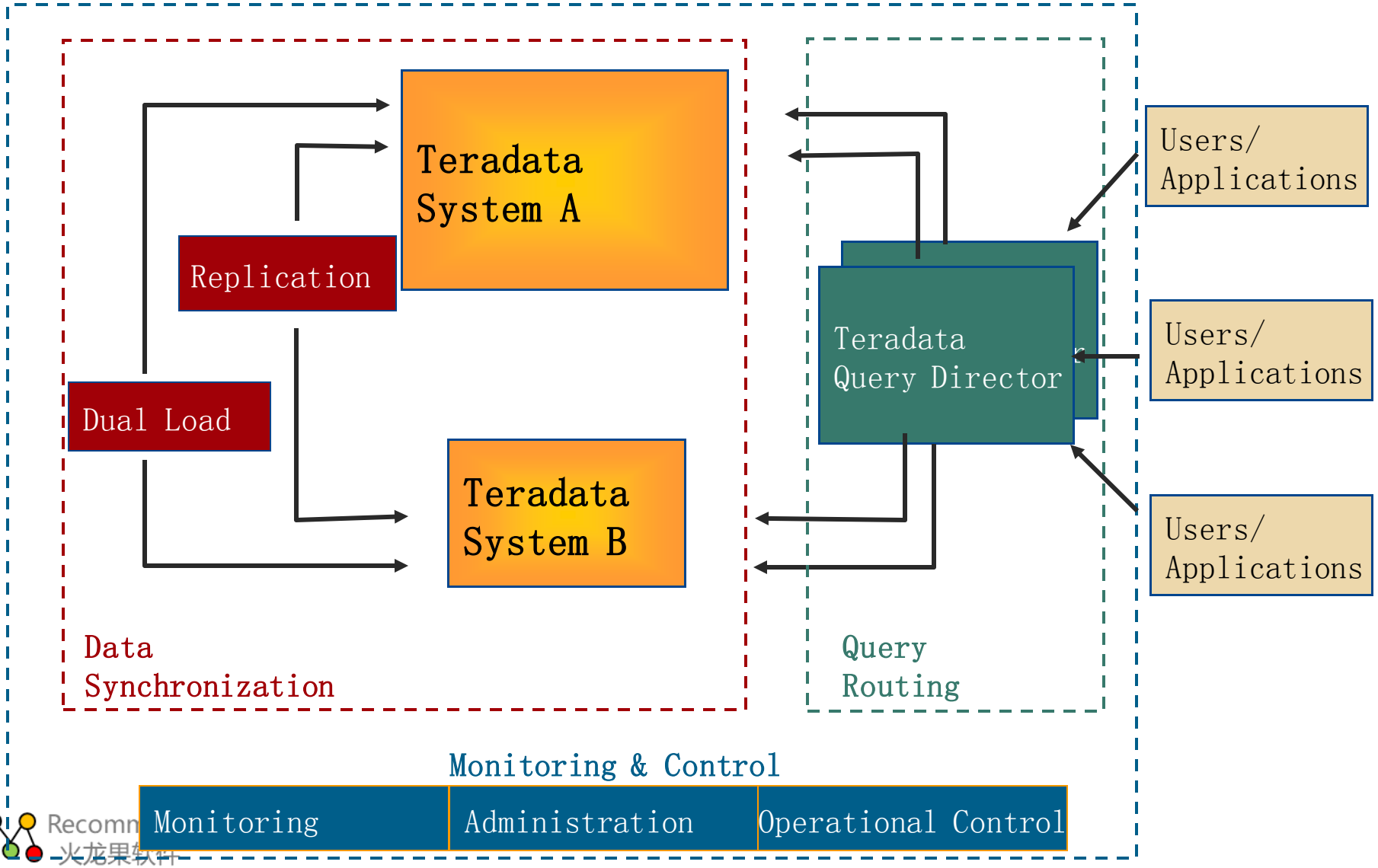
Workload Continuum

Transactional Repositories

Tactical and Strategic Decision Repositories

Recommend
火龙果软件

Dual Active解决方案架构



Recomm
火龙果软件

Dual Active的优点

- 在数据库级别不存在单点故障和不可用状态
- 有效地消除了系统计划性与非计划性的宕机时间
- 后端架构对于前端用户而言是透明的
- 充分利用了企业资产并能够进行工作负载的平衡
- 大大增强了数据仓库的对外持续服务能力

DW Roadmap

阶段 1

- 焦点于**EDW** 基础建设
- 数据汇聚平台**DCP**
- 逻辑数据模型
- 物理数据模型
- 数据质量
- 数据仓库管理与运行
- 报表移植
- 少量分析型应用部署

阶段 2

- 更多的应用部署
- 元数据管理
- 工作负载管理
- 容量规划
- 动态数据仓库概念验证

阶段 3

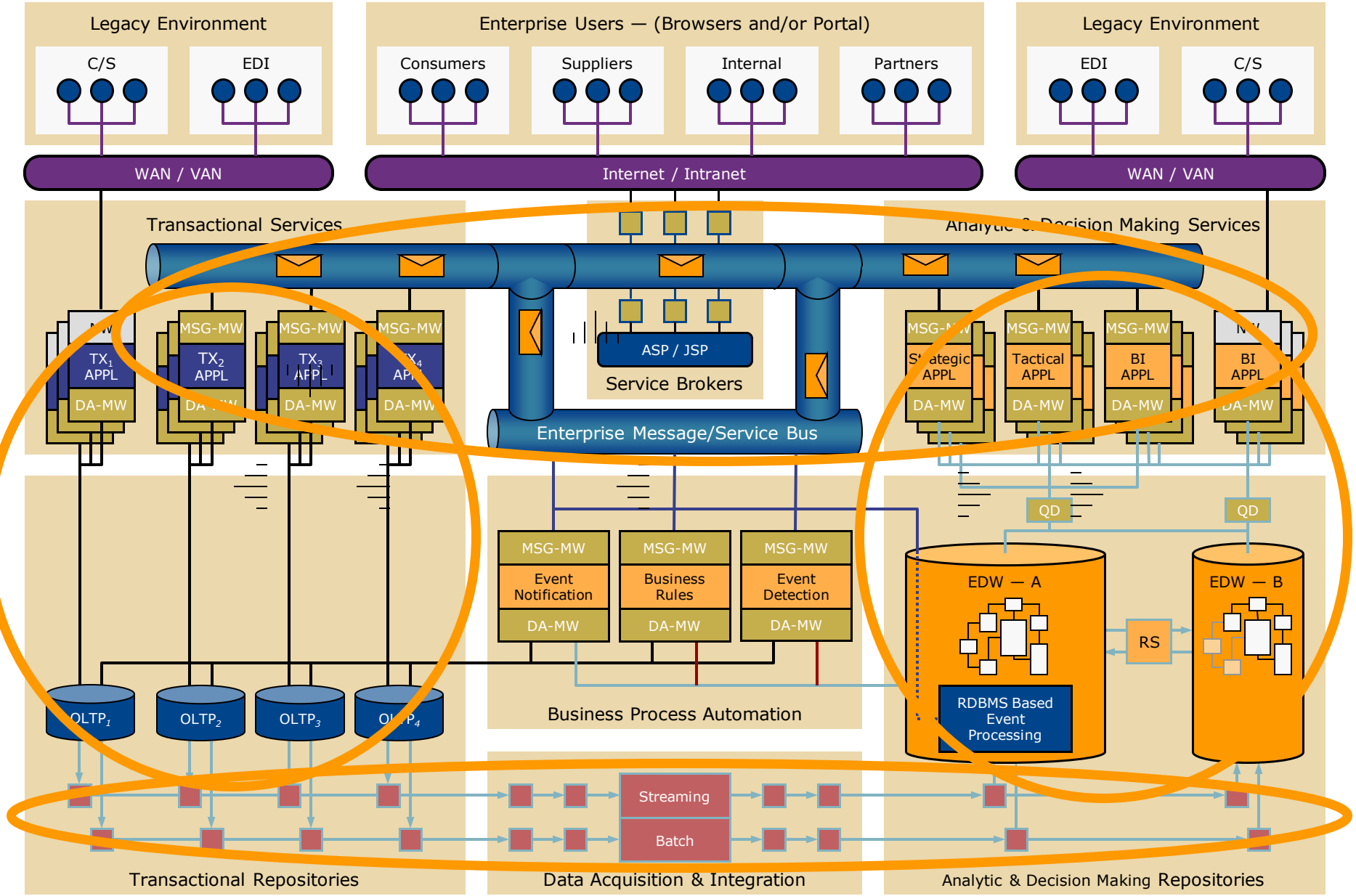
- 动态数据仓库
- 近实时数据采集与数据加载
- 决策智能与操作智能
- 工作负载管理

阶段 4

- 企业信息总线**EAI**
- 动态数据仓库
- **Event-Based Processing**
- 工作负载管理
- **SOA**应用架构

Recommend
火龙果软件

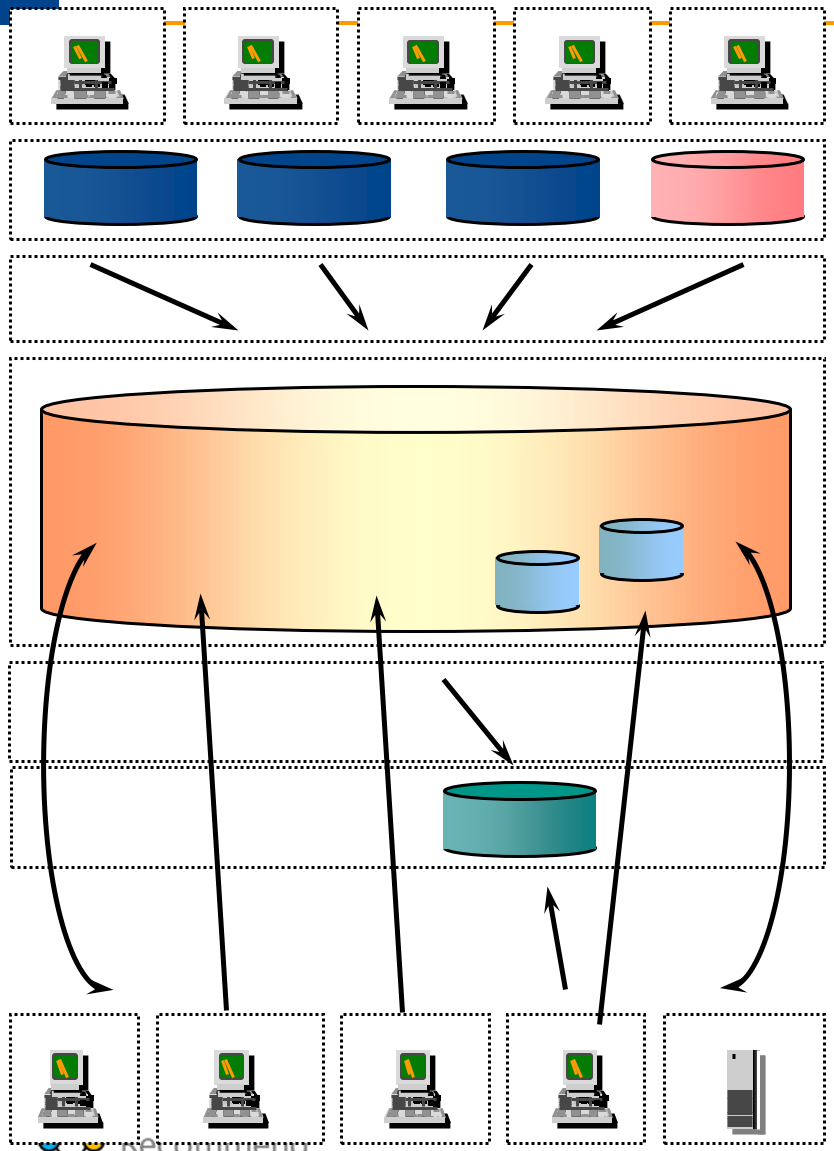
Teradata's Real-Time Enterprise Reference Architecture



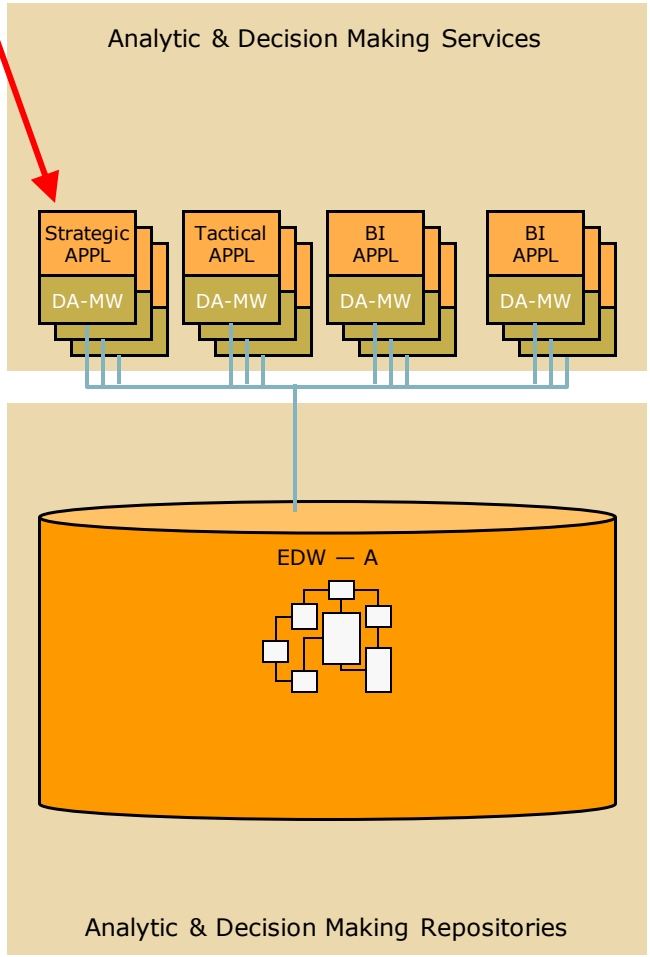
Recommend
火龙果软件

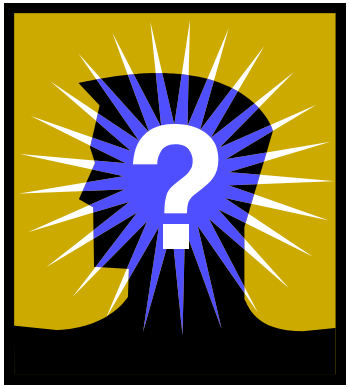
Evolution to "Service Oriented Architecture"

"Applications" are replaced by "services"



Evolution to Service Oriented Architecture



recommend
火龙果软件

Questions.....