



08

## 云计算的架构

---

## 08 架构变革-云计算的架构

- 前言
- 现有IT系统的主要问题
- 采用云计算技术后新系统的架构初探

## 8.1 前言

### ● 云计算起源和发展



1961年斯坦福教授John McCarthy 提出计算资源可以成为一种重要的新型工业基础。类似水、电、气和通信。

1999年Salesforce成立，2001年发布在线CRM系统

2001年Google CEO Eric Schmidt 在搜索引擎大会上首次提出“Cloud Computing”概念。

2003年Google逐步开始在内部使用云计算，2008年推出Google AppEngine云计算平台

2006年Amazon正式对外推出弹性计算服务（EC2）

.....各大全球知名厂商跟进(IBM, MicroSoft....)

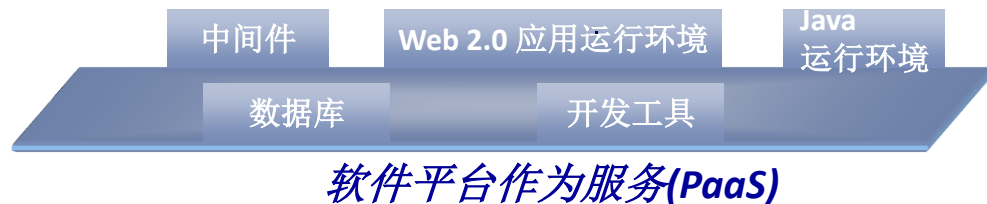
## 8.1 前言

### CLOUD explanation for Cloud Computing

- **C**ost efficient (高效低成本)
- **L**arge-scale (海量大规模)
- **O**ptimized (应用类型优化)
- **U**sage-centric (使用似水电)
- **D**elivery model (服务交付新模式)

## 8.1 前言

### ● 云计算分类



### 举例



Blue Cloud,  
PureScale  
Application  
System

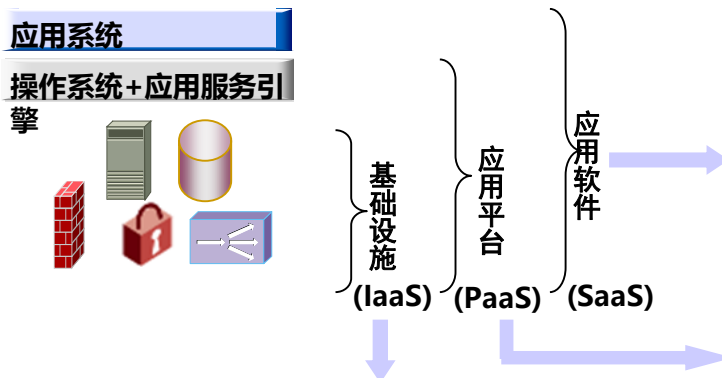


Computing  
on Demand

## 8.1 前言

### ● 云计算应用(服务)的分类

根据提供的服务类型，将云计算应用(服务)分为三类



#### Software as a Service

用户通过标准的Web浏览器来使用Internet上的软件。

用户不必购买软件，只需按需租用软件  
典型应用：Lotus Live, Salesforce.com

#### Platform as a Service

提供应用服务引擎，如互联网应用程序接口/运行平台等。

用户基于该应用服务引擎，可以构建该类应用。

典型应用：Google AppEngine, IBM PureScale Application System, SAE

#### Infrastructure as a Service

以服务的形式提供虚拟硬件资源，如虚拟主机/存储/网络/安全等资源。

用于无需购买服务器、网络设备、存储设备，只需通过互联网租赁即可搭建自己的应用系统

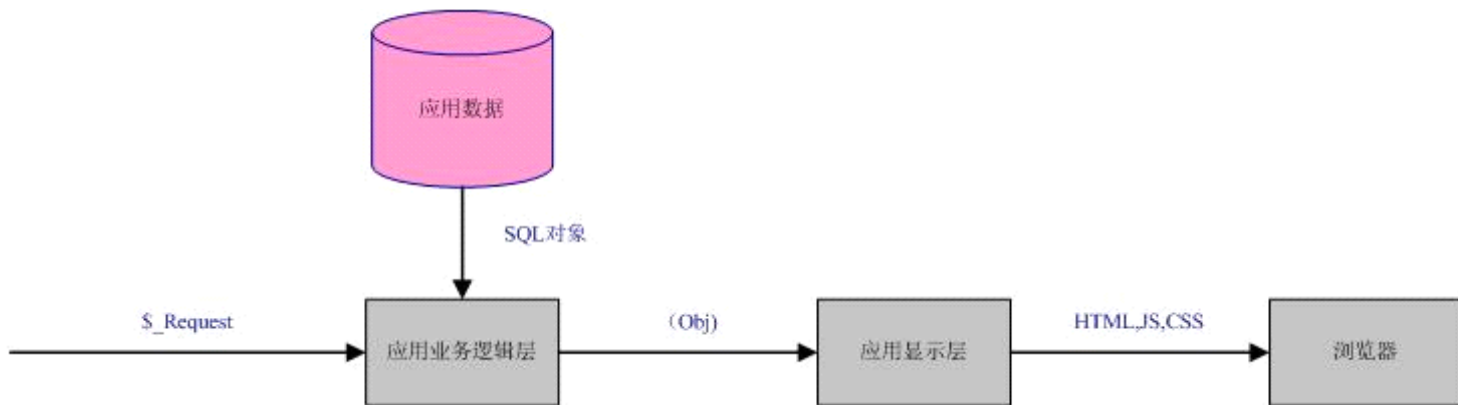
典型应用：Amazon Web服务 IDC

## 8.1 前言

云计算即是商业模式的变革，也是技术的变革。

## 8.2 现有IT系统的主要问题

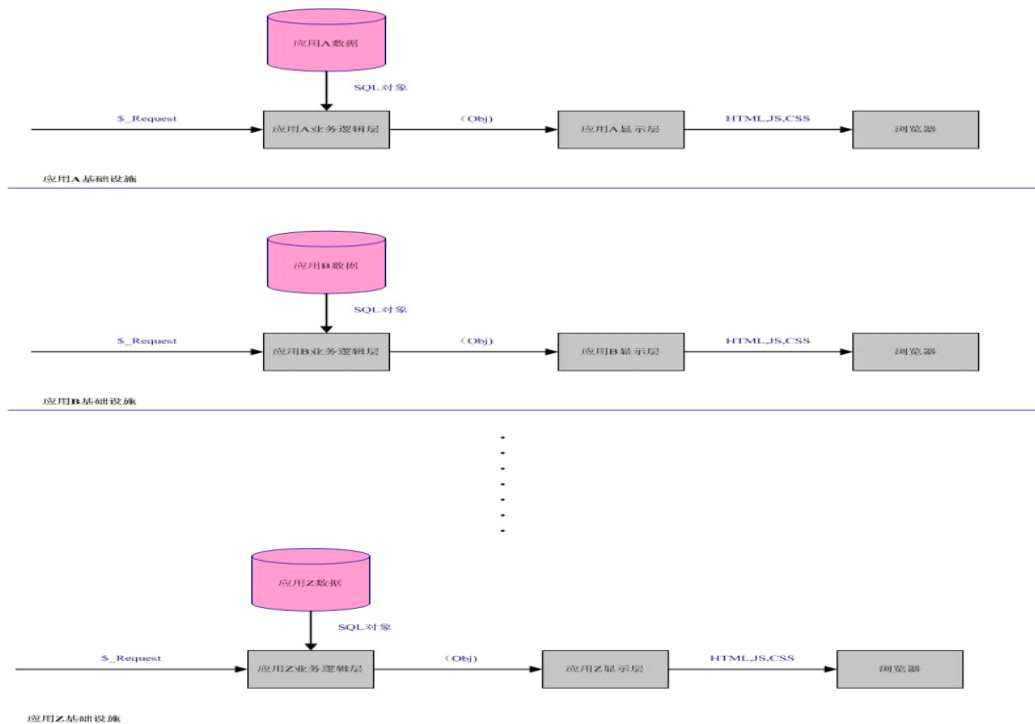
对于我们的Web 业务应用（架构师、开发人员将会围绕数据创建了传统的“n”层软件栈（数据存储层、业务逻辑层与显示层））来说相应的应用全景图





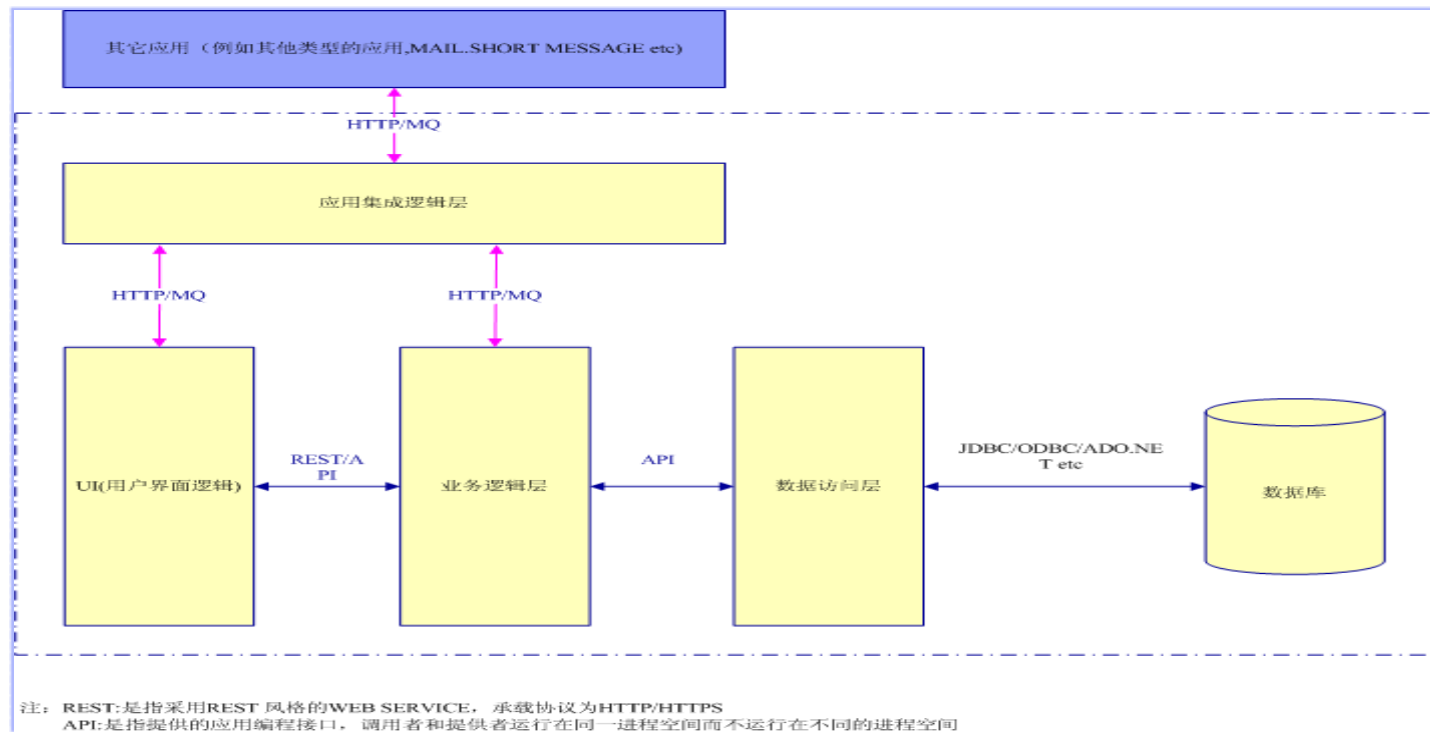
## 8.2 现有IT系统的主要问题

随着按照这种方式建设的应用系统越来越多,就出现了如下图所示的情况



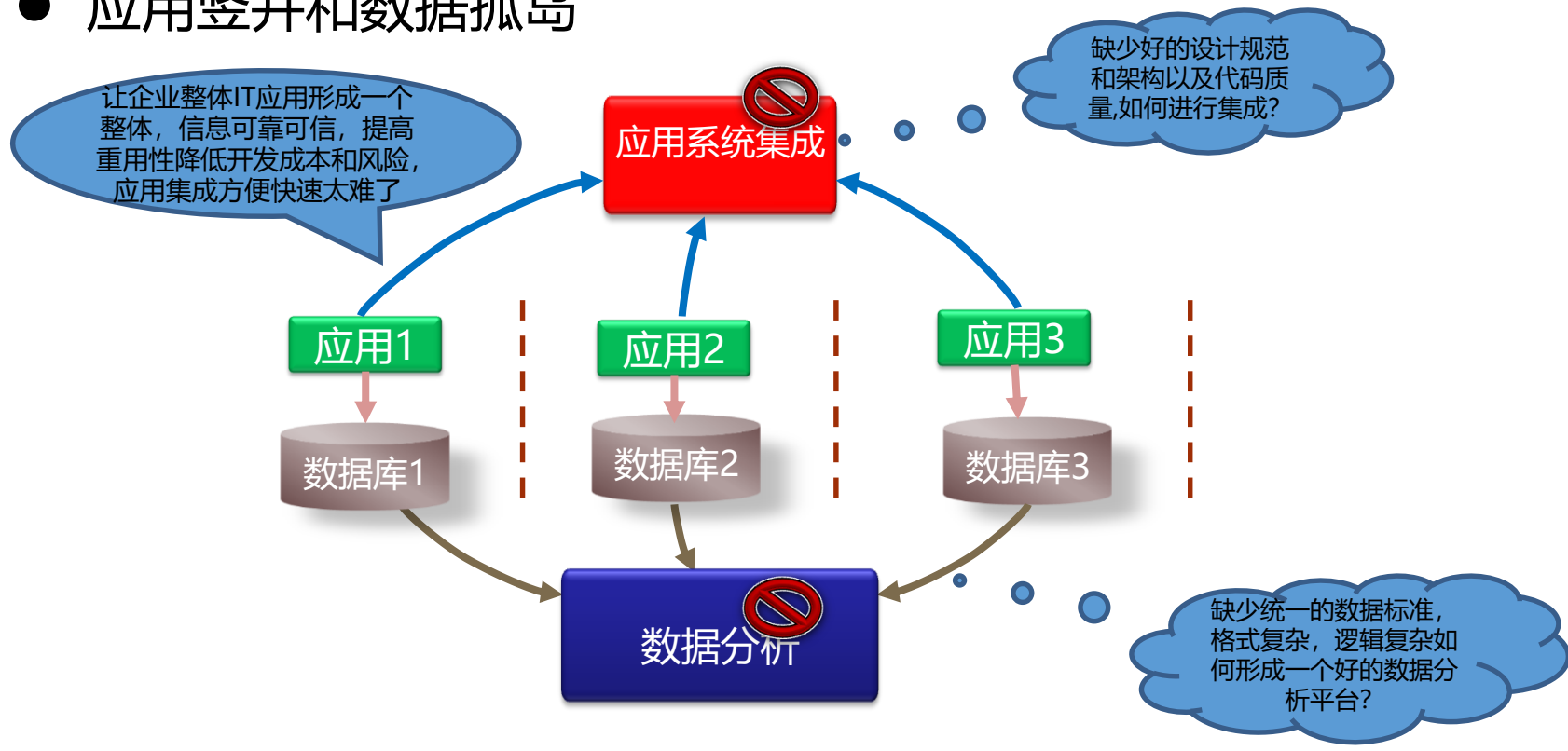
## 8.2 现有IT系统的主要问题

下图所示的是一个我们最喜欢用的经典的应用分层架构设计图



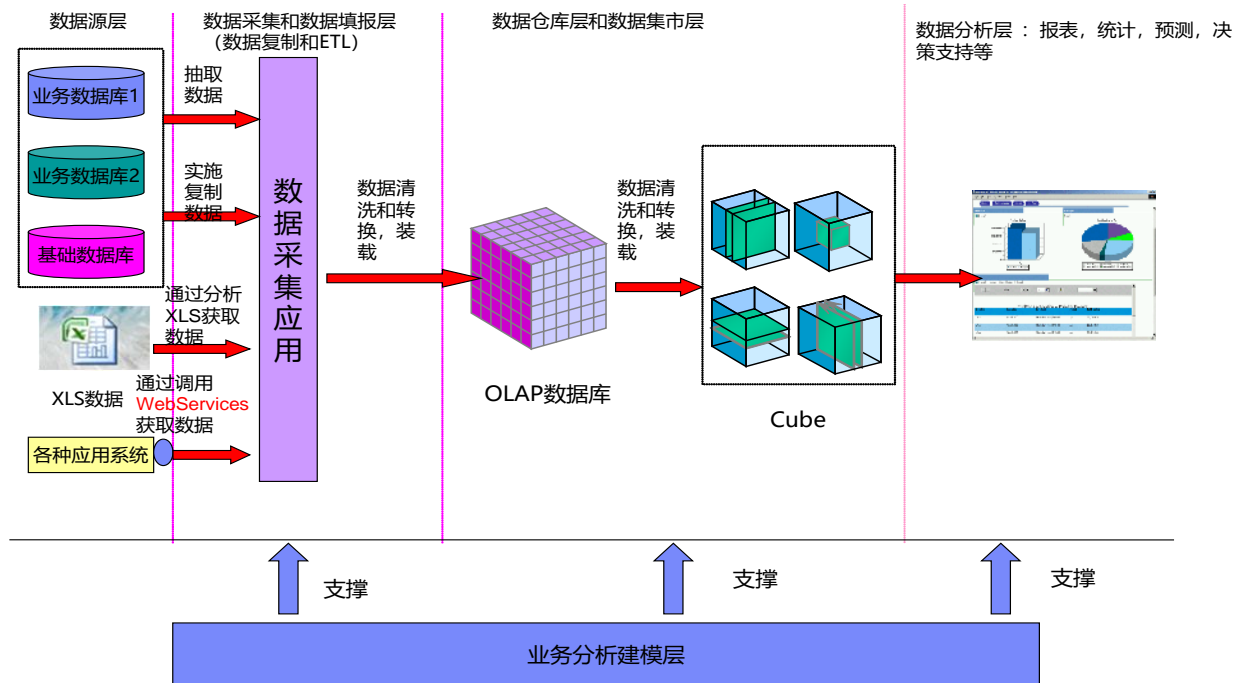
## 8.2 现有IT系统的主要问题

### ● 应用竖井和数据孤岛



## 8.2 现有IT系统的主要问题

很多企业当中都在建设数据中心以形成一个好的商业智能(BI)平台；以达到辅助业务决策管理的功能。相应的数据流架构图例子



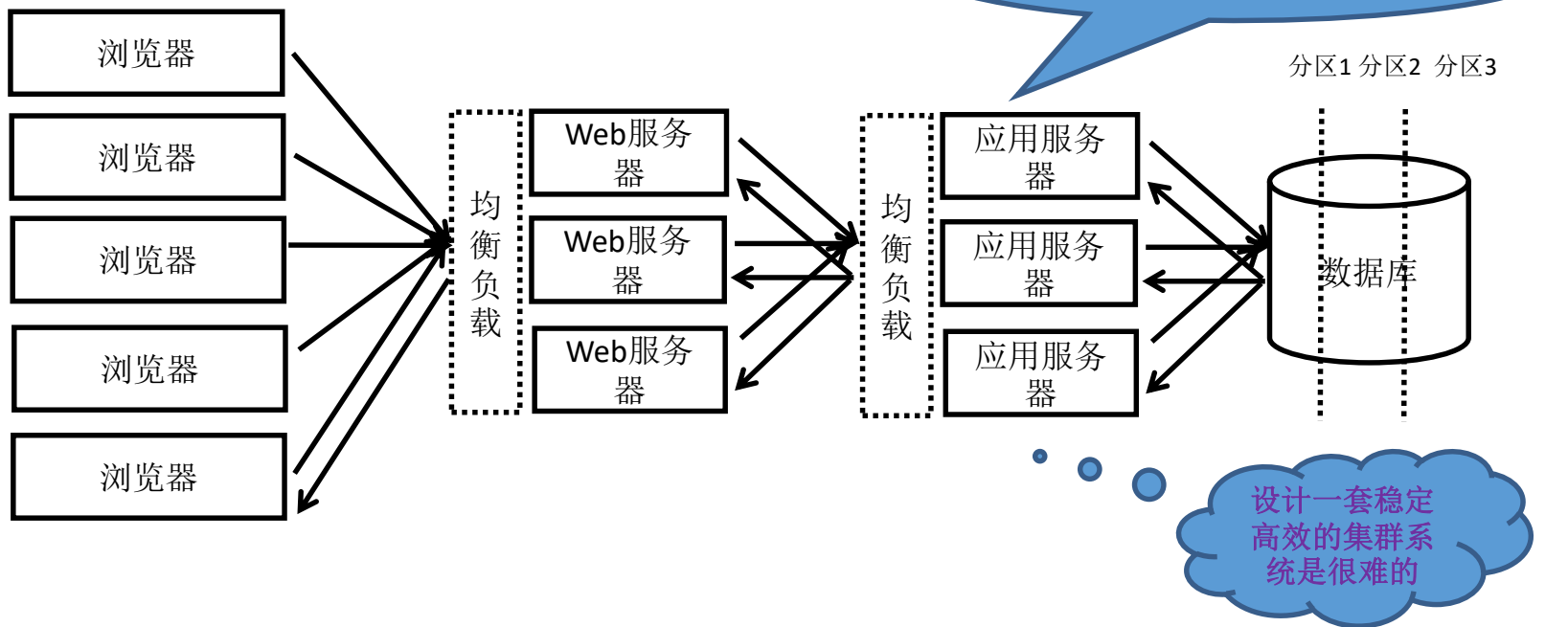
1.无法获取准确、完整的数据；甚至不知道从哪获取

2.数据量太大，处理速度太慢了；疯掉

3.用户会认为我们这个系统就是一个报表系统罢了

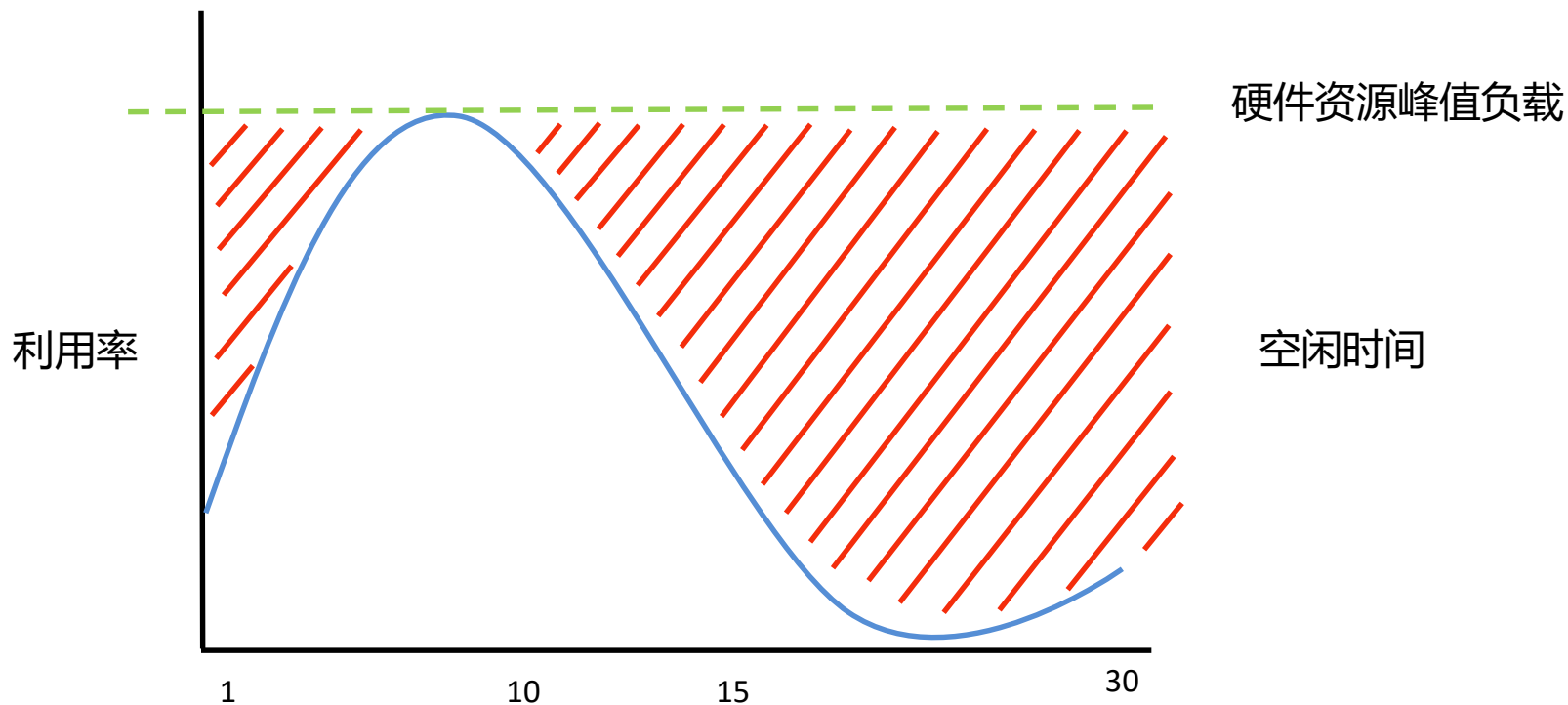
## 8.2 现有IT系统的主要问题

### ● 集群、均衡负载、数据分区架构设计



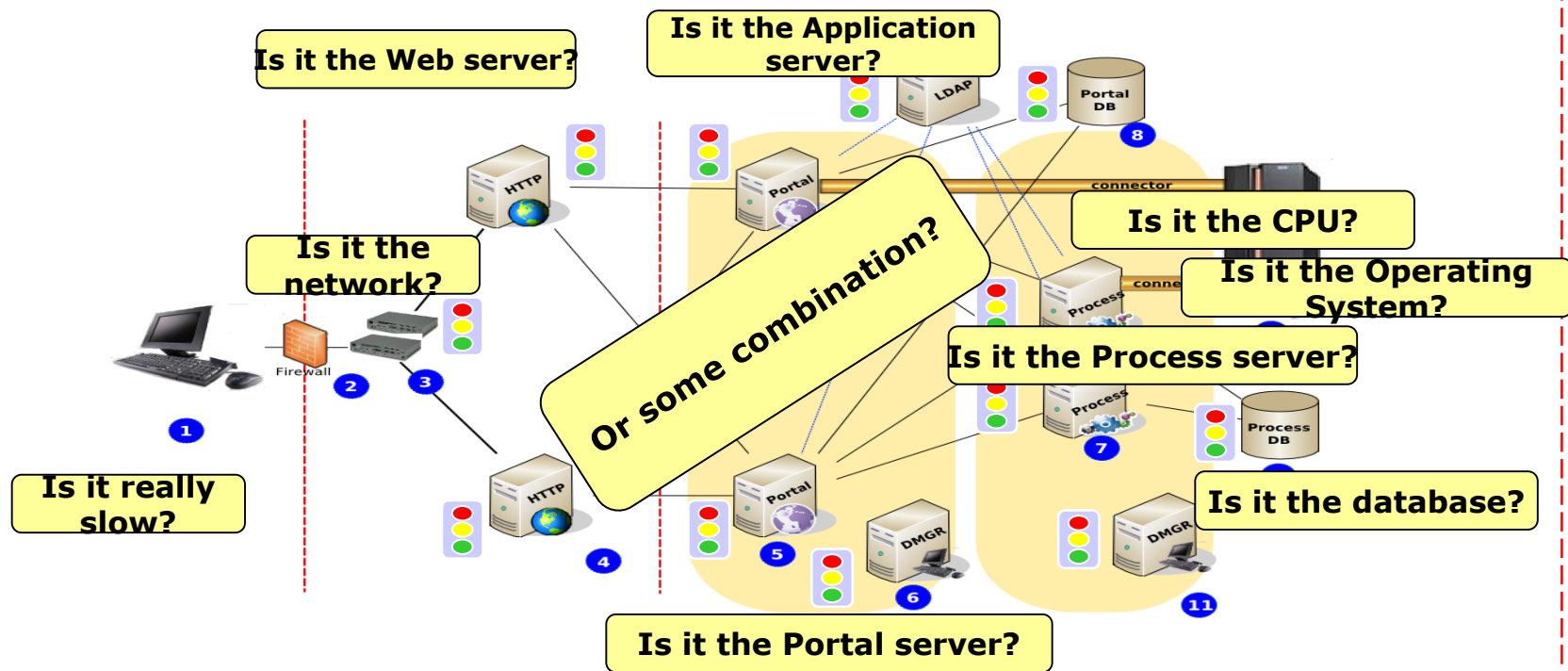
## 8.2 现有IT系统的主要问题

硬件利用率太低，费电、占空间、运维成本高；简而言之太不低碳了



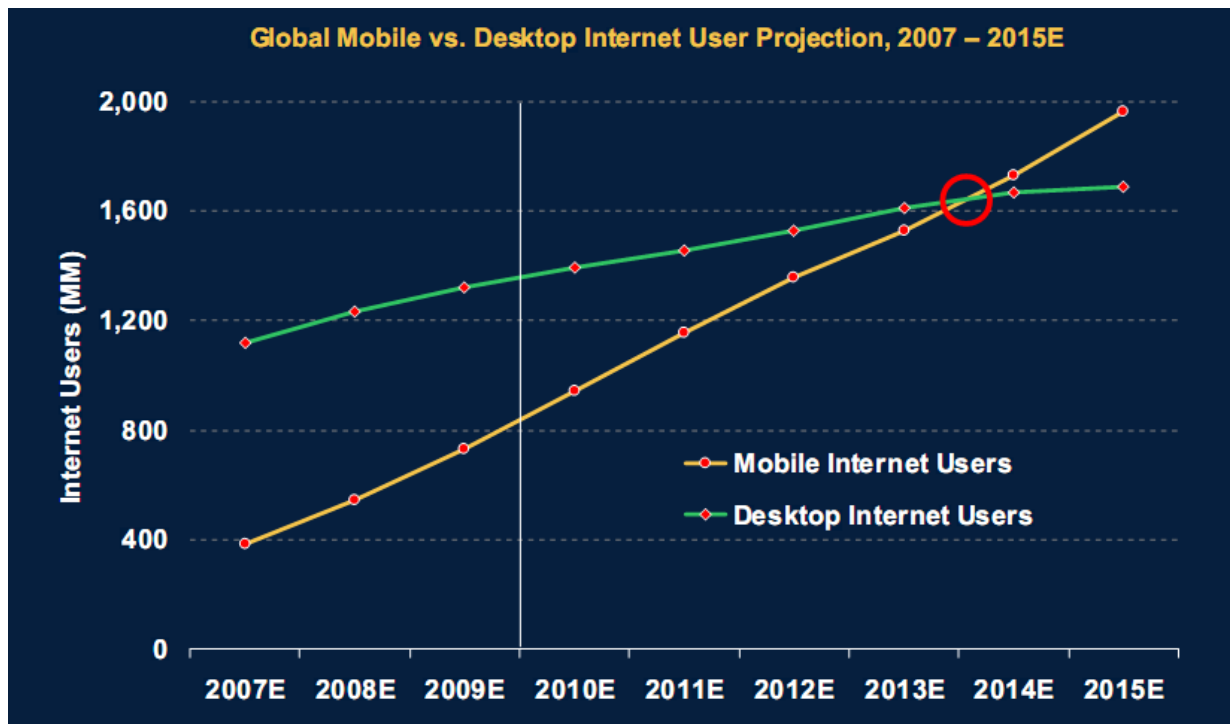
## 8.2 现有IT系统的主要问题

监控对我们的业务至关重要,并且监控不是买一个东西实施下就好了的



## 8.2 现有IT系统的主要问题

移动互联网、物联网的发展；特别是移动互联网的发展所带来的变革





## 8.3 采用云计算技术后新系统的架构初探

### High Level View of Paas Cloud

接入端-PC、智能手机、智能平板 等

应用-核心支撑应用（统一用户管理中心、数据开发平台、监控）,各种业务应用 etc

应用运行支撑-J2EE应用服务器、MQ、ESB、WorkFlow、Hadoop、Web服务器 etc

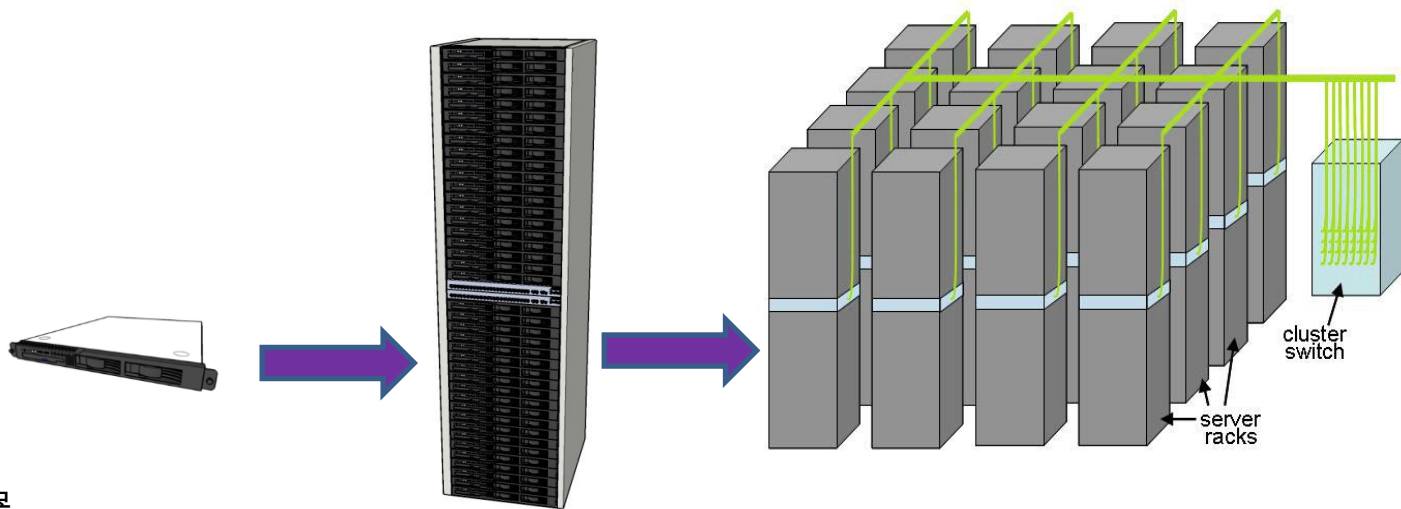
数据-关系型数据库、NoSQL etc

IaaS- IaaS云计算平台管理：服务器虚拟化、存储虚拟化、网络虚拟化、自动化

硬件-服务器/存储/网络

## 8.3 采用云计算技术后新系统的架构初探

高密度堆叠服务器，服务器低功耗、高性能。



服务器

CPU: x86/ Power

DRAM

DISK:HDD/SSD

Architect: 机架式/刀片

Racks (机柜)

10台以上的服务器

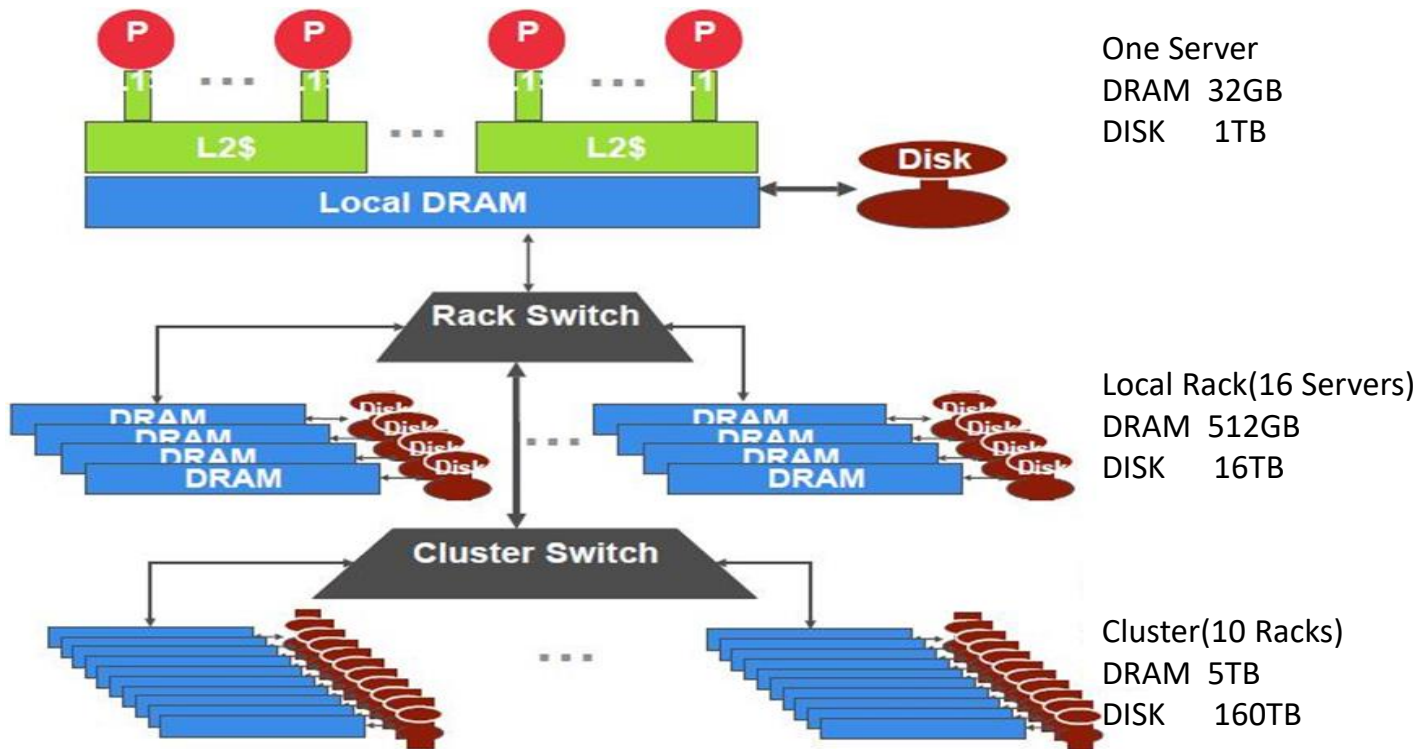
以太网交换 (光纤)/或者

Infiniband网络

Cluster(服务器集群)

## 8.3 采用云计算技术后新系统的架构初探

### 硬件体系层次提供的存储能力：一个例子



## 8.3 采用云计算技术后新系统的架构初探

### 使用SSD固态硬盘提升性能

L1 cache reference	0.5 ns
Branch mispredict	5 ns
L2 cache reference	7 ns
Mutex lock/unlock	25 ns
Main memory reference	100 ns
Compress 1K bytes with Zippy	3,000 ns
Send 2K bytes over 1 Gbps network	20,000 ns
Read 1 MB sequentially from memory	250,000 ns
Round trip within same datacenter	500,000 ns
Disk seek	10,000,000 ns
Read 1 MB sequentially from disk	20,000,000 ns
Send packet CA->Netherlands->CA	150,000,000 ns

### 8.3 采用云计算技术后新系统的架构初探

很多云计算项目都是在做硬件虚拟化等一些事情

## 8.3 采用云计算技术后新系统的架构初探

### laas&部分Pass

在云计算实施的前期或者在很多场景的时候；主要关注点在于应用的可靠运行、快速开发和部署、机器资源的充分利用、以及方便的运维等问题；对于这个时候我们应该定位于主要采用laas云计算架构（即很依赖于硬件虚拟化技术）和部分采用Pass云计算架构来解决。

- 1.重点采用laas 云计算架构中的硬件虚拟化技术等技术（服务器虚拟化、网络虚拟化、存储虚拟化）以提高硬件的利用率、降低机房占用空间和功耗。
- 2.快速和方便地给应用提供应用所需要的服务器资源（VM）、网络资源、存储资源。
- 3.快速和方便地给应用提供应用所需要依赖的平台软件资源，例如数据库系统（DB2）、J2EE应用服务器（WAS）、WEB 服务器（IHS）等。
- 4.快速和方便地自动地把应用部署到相应的硬件环境中。
5. 硬件虚拟化技术（例如VMWare, PowerVM,xen等）的能力需要在这体现。

注意：

- 1.对于在这种情况下，每台服务器所具有的CPU core数目和内存数量越大越好。不要弄一些性能较差的机器干这种事情。
2. SSD和大内存对数据库性能的提升是很明显的。
3. 这个就是我们通常所说的以大变小

## 8.3 采用云计算技术后新系统的架构初探

### IaaS&部分PaaS



### 8.3 采用云计算技术后新系统的架构初探

很多云计算项目都是在做硬件虚拟化等一些事情，但是实施好的话也大大地提高了应用的可靠性、性能；和降低了维护成本（机器少了）和降低机房占用空间和降低了能耗（新机器的能源利用率更高、单位TPS值的功耗更低）。

**但是数据孤岛、应用竖井这些最关键的问题并没有解决。**



## 8.3 采用云计算技术后新系统的架构初探

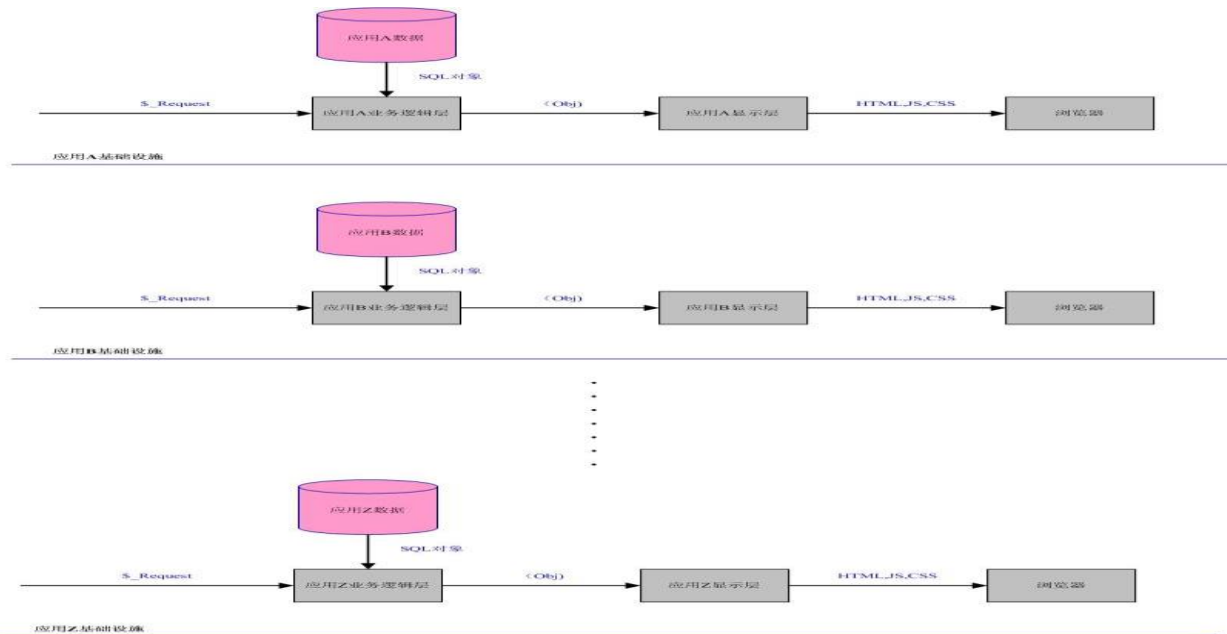
### 数据开放平台，数据既是服务

我们一定要坚信特别是信息架构师更要坚信，处于大多数系统核心的是**数据**，而不是**算法**（或者称之为代码）。随着互联网（固定互联网和移动互联网）技术和物联网技术的发展，最终用户产生和消费的数据将比以往更加推动信息技术的使用，我们业务流程的运转需要各个环节的人员产生和消费相应的数据，数据需要更加地及时、有效、精确；我们的业务的运营越来越离不开相应的数据。

在任何情况下我们需要通过Web来呈现给用户使用的所有功能归根结底都是一个界面一个具有较好用户体验的界面来更好地产生和消费**数据**，以促进人与人之间的协同、人与机器之间的协同以及业务流程更加高效、精准的运转从而提高企业的经营效率和效益。这些**数据**就构成了我们企业应用信息系统的核心价值，不论这些数据是合作伙伴创建的还是我们的一线员工和管理层所创建的。市场需求的变化促进了业务和业务流程的变化和促进了人的变化，促进了**数据**的变化即需要创建和消费、利用更多类型、更多种类、更大量的**数据**，**数据**推动了我们更多的产品/应用，所以架构师、开发人员将会围绕数据创建了传统的“n”层软件栈（数据存储层、业务逻辑层与显示层）即我们的应用都是由数据来驱动的。

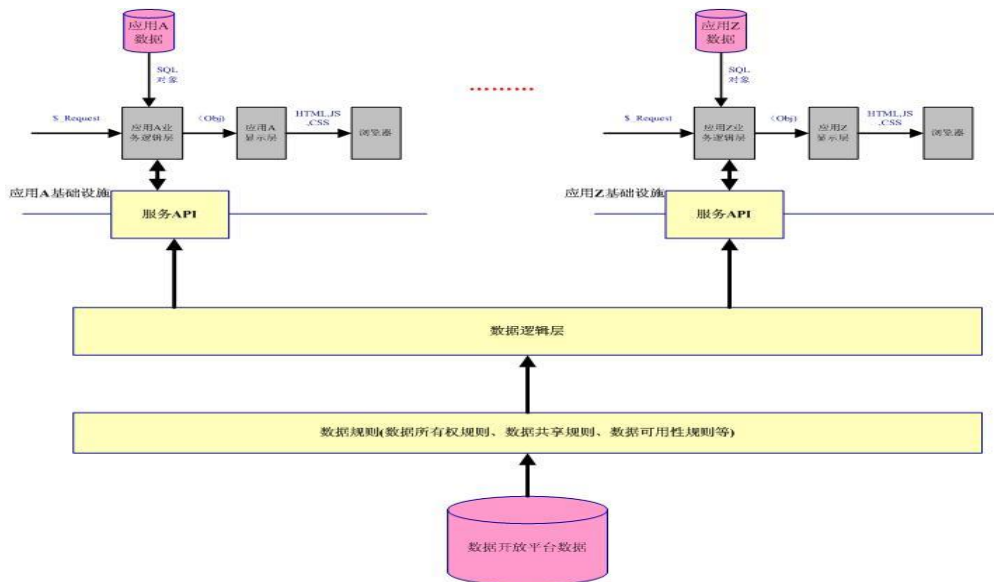
## 8.3 采用云计算技术后新系统的架构初探

**问题：** 如下图按照传统的方式建设的方式必然产生数据孤岛，能够被共享的业务核心数据被分散到各个应用，并且各个应用的开发商很多不一样的甚至是同一个开发商由于各种问题导致数据编码标准不一致，数据不一致和不可信等问题；这些问题导致不能够形成完整和精准的数据视图，导致很难进行数据分析和支撑业务流程的运营。



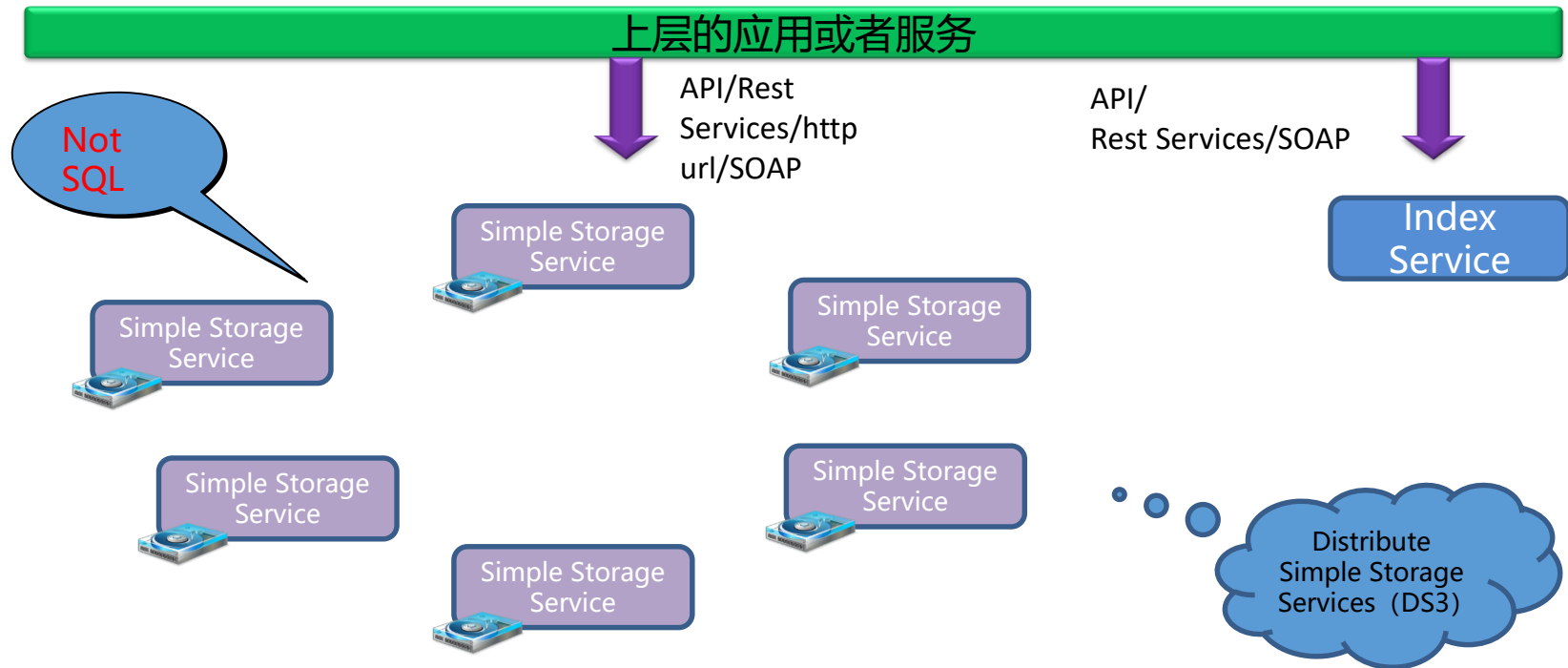
## 8.3 采用云计算技术后新系统的架构初探

**目标：** 我们需要采用新的思路，即把在企业中能够被支撑各个业务场景的业务应用系统所共享的基础数据全部放入到统一的数据存储池中，并且让这个统一的数据存储池提供相应的服务API让各个业务应用使用（查询、增加、修改等），各个业务应用系统不再保存和维护这些数据，与各个业务应用私有相关的数据有相应的各个业务应用进行维护和控制。这个数据存储池我们称之为数据开发平台，统一相应的编码规则、数据元定义等等。

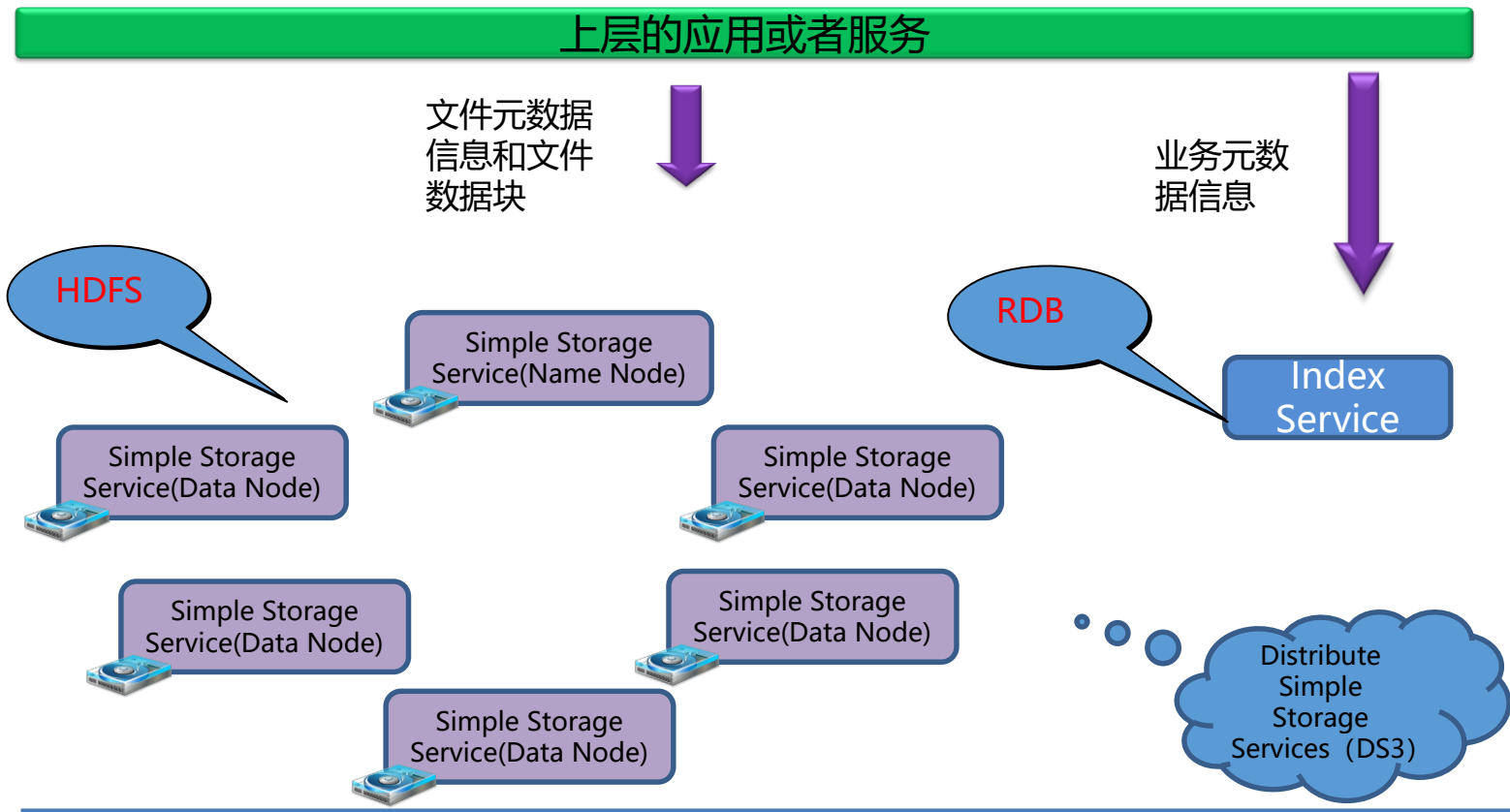


## 8.3 采用云计算技术后新系统的架构初探

在我们建设支撑企业业务运营的业务应用当中，我们不仅要管理关系型的业务数据，还有许多非关系型的数据需要进行管理（如文档、图片、音频/视频等），我们不能把这些数据信息由不同的单独业务应用进行单独管理，我们应该统一管理起来提供服务器让不同的业务应用进行利用(存储、提取等)。



## 8.3 采用云计算技术后新系统的架构初探



## 8.3 采用云计算技术后新系统的架构初探

### 统一用户管理中心 (UMC)

**问题：** 建设了大量的业务应用系统来支撑我们的业务运营，但是这些业务应用系统在用户认证、用户授权等权限控制方面都自己建设自己的相应权限模块来完成用户认证、用户授权等功能即每个业务应用都完全拥有和控制自己的用户认证和用户授权信息；但是这种建设方式导致了以下的主要问题：

1. 同一个用户在不同的业务应用中都有相应的用户信息，例如用户名、密码等信息；这就会导致以下几个问题：

1.1 用户信息不一致，例如某个用户密码不一致。

1.2 因为修改了相应的信息需要到不同的业务应用进行修改（修改了密码、地址等基础信息）导致修改的工作量大增，所以绝大部分用户不愿意修改相应的信息（因为改了不是给自己找麻烦嘛☺）。并且这种现象也导致了一些安全隐患，例如用户的密码很多都是初始密码例如88888，因为按照安全策略一般来说是需要定期修改密码和密码需要符合一定的安全强度的，如果我都改我哪记得住，这不是自己给自己找麻烦啊☺。

1.3 间接地导致数据孤岛的形成。

2. 在各个业务应用中都按照自己的编码规则建设一套组织机构，导致从业务角度来看是同一套组织机构但是在各个业务应用中是不一样的，这不是间接地导致数据孤岛的形成。

3. 实现SSO(单点登录)困难，一个完整地单点登录SSO是一个完善的用户认证、用户授权的机制而不仅仅是一个用户的登陆过程。从而导致界面集成的困难，即我们建设门户实现后的效果是很差的，实现的效果距离与我们所熟知的一些互联网应用有很大的差距。

4. 很多情况下每上一个应用系统，这个应用系统都要重新实现用户认证、用户授权等权限控制模块，这不是做重复地事情增加工作量也做不好嘛☺

。。。等等其它问题。

## 8.3 采用云计算技术后新系统的架构初探

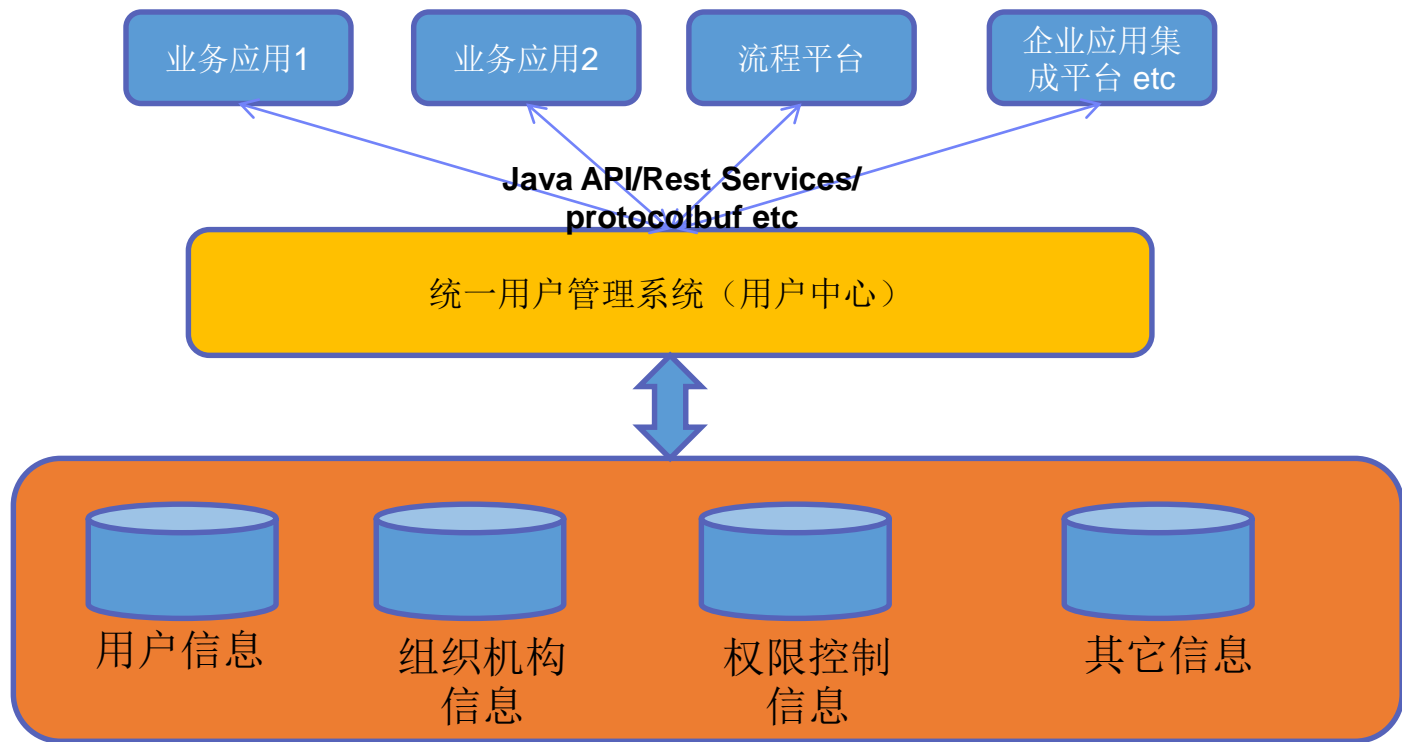
### 统一用户管理中心 (UMC)

**目标：** 建设一个统一用户管理中心系统 (UMC) 以避免上述所述问题的时候了，它主要要达成以下的目标：

1. 统一用户管理中心系统 ( UMC ) 统一完全管理和控制相应的用户、组织机构、基础权限控制信息，以形成一个完善地用户认证、用户授权机制。业务应用（不管新应用还是老应用）不再保留自己的用户认证、用户授权信息，不再需要建设单独的用户、组织机构、基础权限控制等模块，除非是和应用特定需求密切相关的详细的数据访问权限信息。
2. 业务应用通过统一用户管理中心系统 ( UMC ) 提供地API来完成相应的用户认证、用户授权。
3. 让整个企业的IT应用系统的单点登录 (SSO)更加地简单、快捷、有效。
4. 统一用户管理中心系统(UMC) 的数据信息其实是主数据的一部分。

## 8.3 采用云计算技术后新系统的架构初探

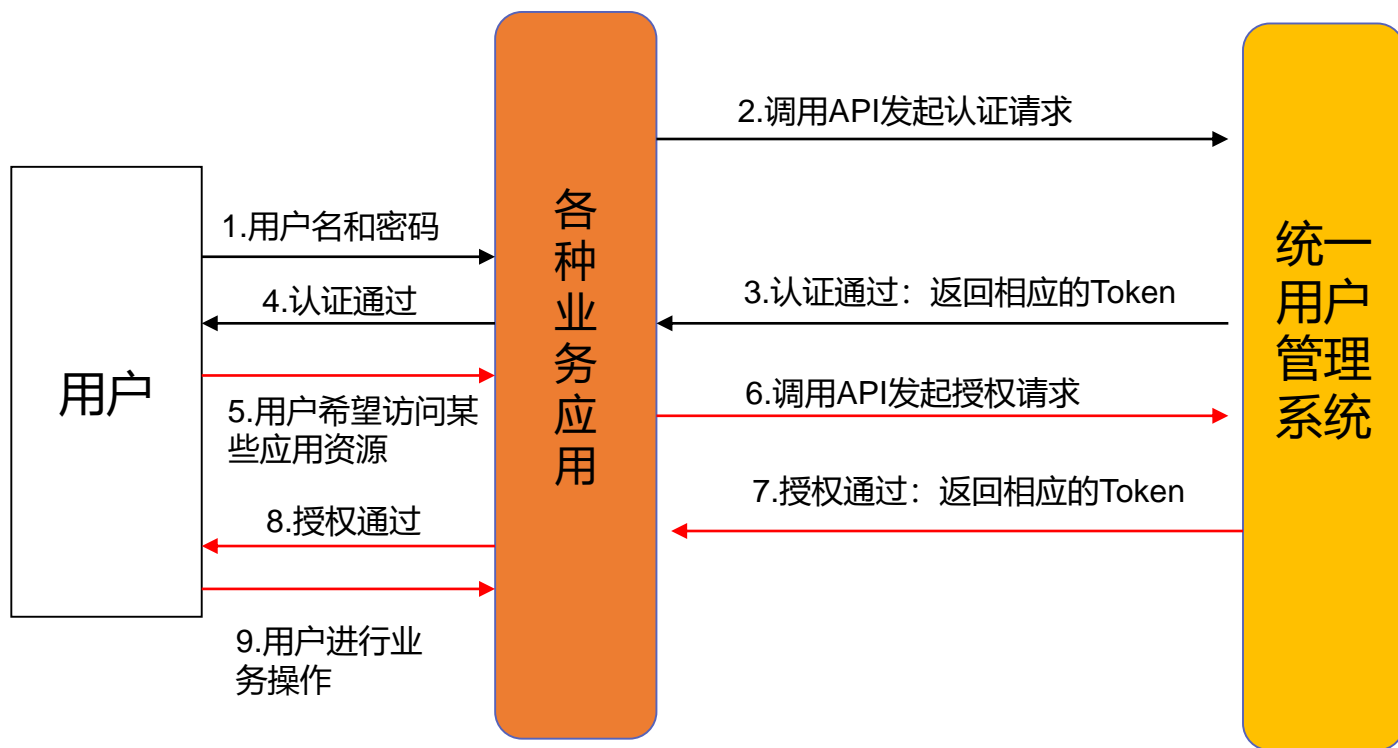
一个非常简单的整体逻辑架构如下图所示：





## 8.3 采用云计算技术后新系统的架构初探

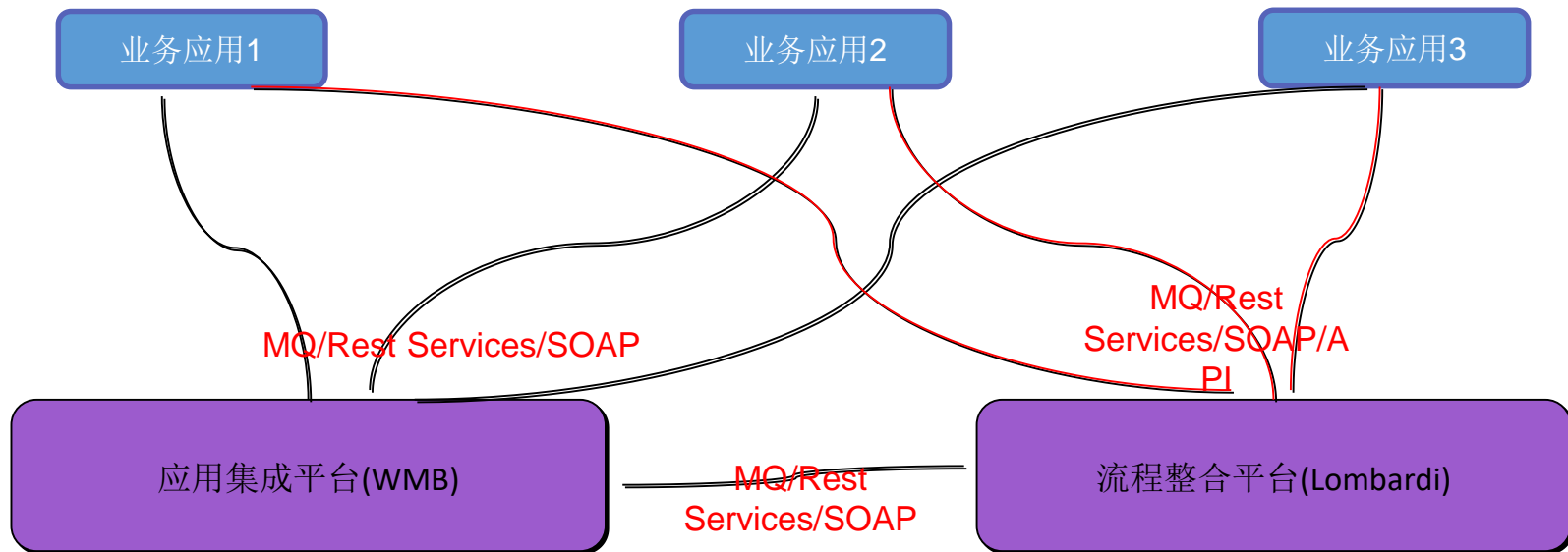
### 用户认证和用户授权：例子流程



## 8.3 采用云计算技术后新系统的架构初探

### 应用集成和流程整合

我们不能为了应用集成而集成，为了流程整合而整合；而把相应应用集成和流程整合系统包含在某个应用当中；我们应该在企业架构层面建设应用集成平台和流程整合平台，所有应用之间的集成和跨应用业务流程的整合通过应用集成平台和流程整合平台来完成。



### 8.3 采用云计算技术后新系统的架构初探

我们需要面对大并发和大数据量的挑战

## 8.3 采用云计算技术后新系统的架构初探

**挑战：** 可伸缩性是我们每天奋力抵抗的一大架构压力。我们所做的每一项架构及设计决策，身前身后都能看到它的踪影。对于大并发量的用户核心业务应用系统，可伸缩性是生死交关的问题。在一个可伸缩的架构中，资源的消耗应该随负载线性（或更佳）上升，负载可由用户流量、数据量等测量。如果说性能衡量的是每一工作单元所需的资源消耗，可伸缩性则是衡量当工作单元的数量或尺寸增加时，资源消耗的变化情况。换句话说，可伸缩性是整个价格-性能曲线的形状，而不是曲线上某一点的取值。并且我们需要达到以下几点：

- 1.资源利用率能够随着负载的增长能够线性增长。形象点就是说，如果负载不断地增加，我们能够不断的添加机器（通过负载均衡机制）来处理；并且系统的响应时间不会产生剧烈的波动
- 2.系统的架构设计应该能够面对系统数据、用户数增长10倍以上的情况。形象点说：如果现在的系统能够承受10000个用户的使用，那系统现在的这个设计能够承受10万个用户的使用。
- 3.由于整个系统将是由多台机器之间协同工作，单台机器的失效、以及性能严重退化不会影响到整个系统的对外提供的较好地服务质量。
- 4.系统能够提供一个稳定的响应时间，不能出现剧烈的波动。
- 5.系统监控、管理起来方面简单，并且通过相应的诊断日志和工具能够很方便的定位出错误的原因和性能的瓶颈所在。

## 8.3 采用云计算技术后新系统的架构初探

### OLTP关系数据库层面

我们大部分应用都是建立在关系型数据库上（例如DB2),让关系型数据库体现出很好的性能有很多方式：

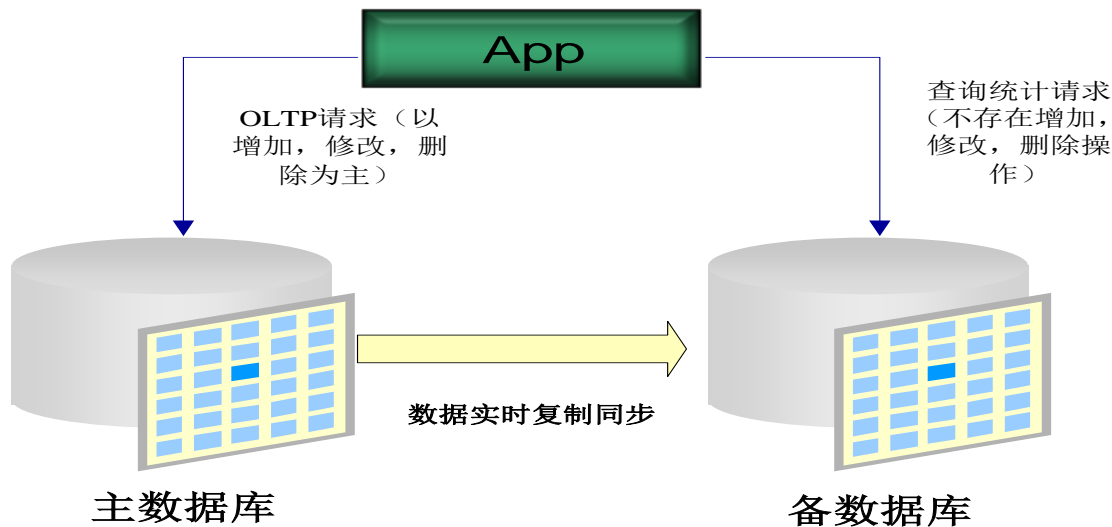
- 1、大机器、大内存、SSD、高端存储。（IBM 高端小机、存储）
- 2、优良的数据库物理设计
- 3、编写优良、执行效率高的SQL语句。
- 4、编写优良、执行效率高的业务逻辑代码。
- 5、关系数据库集群（DB2 Purescale）

并且但是上面做好后，在数据量急速增长和并发用户数急速增长的情况下，还是可能会遇到较为严重的性能问题。

## 8.3 采用云计算技术后新系统的架构初探

### OLTP关系数据库层面

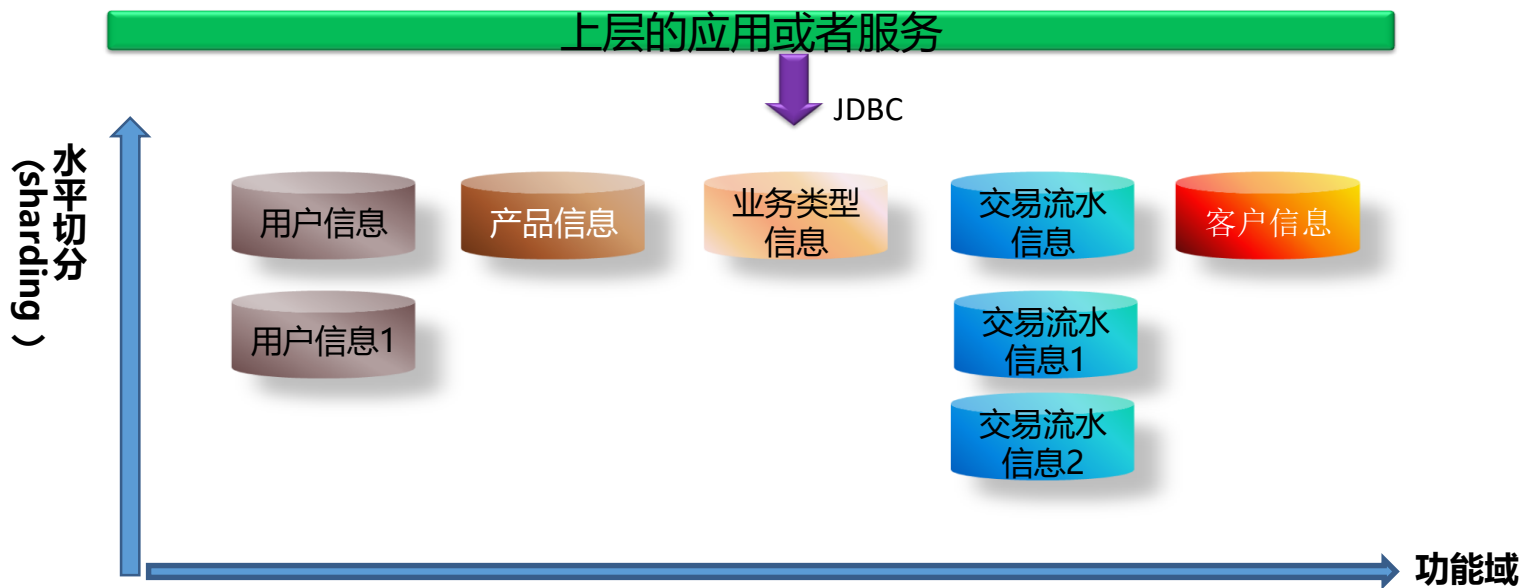
**解决方式一：**在很多场合下我们可能会形成一个主-备方式的数据库架构方案来分解压力如下图所示，这种方式能够缓解相应的性能压力，但是我们都知存在相应的数据热点问题，即某些数据存在高强度的删除，修改，查询等操作；一台服务器还是不能够满足相应需求。如果性能要求并不是特别高的话实施对某个数据库实施数据库集群也可解决



## 8.3 采用云计算技术后新系统的架构初探

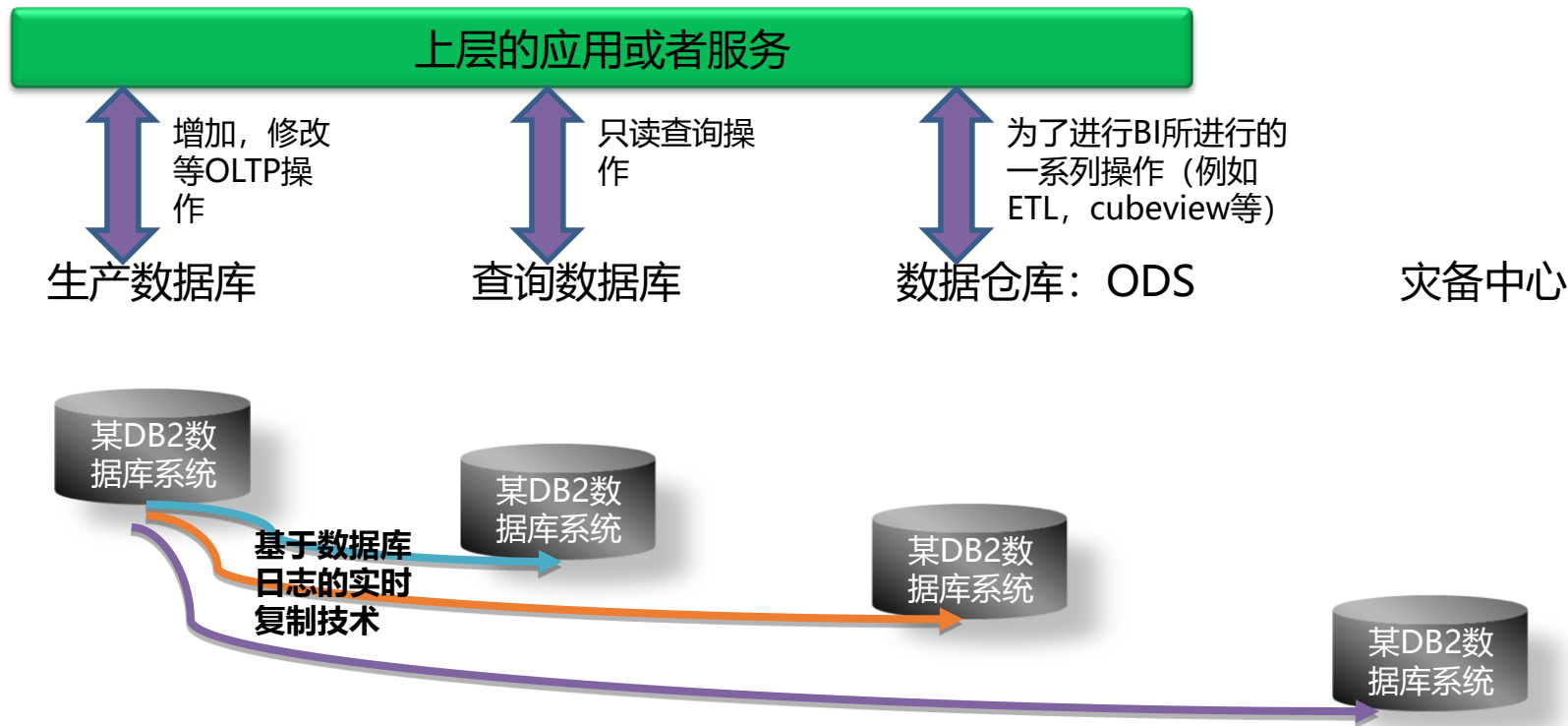
### OLTP关系数据库层面

**解决方式二：**按照功能域进行分解和数据的水平切分（即Sharding）。如果还遇到相应的性能问题，可能需要对某个数据库实施数据库集群甚至需要在业务逻辑代码层引入memcache



## 8.3 采用云计算技术后新系统的架构初探

### OLTP关系数据库层面—多重复制





## 8.3 采用云计算技术后新系统的架构初探

# NOSQL

is simply

# Not Only SQL!

## 8.3 采用云计算技术后新系统的架构初探

BASE理论简介:ACID 理论的另外选择

# NOSQL特点

- 它们不是关系型数据库，它也不是什么类型应用都能用有自己的使用场景
- 它们可以处理超大量的数据
- 它们运行在较为便宜的服务器集群上
- 它们打破了性能瓶颈
- 没有过多的操作。
- **缺点：现在基本上都是开源产品还不是特别成熟。**

# 几个主流开源的NoSQL产品

- Apache CouchDB: 源自IBM, (IBM 支持)
- Apache cassandra: 源自Facebook
- Apache MongoDB

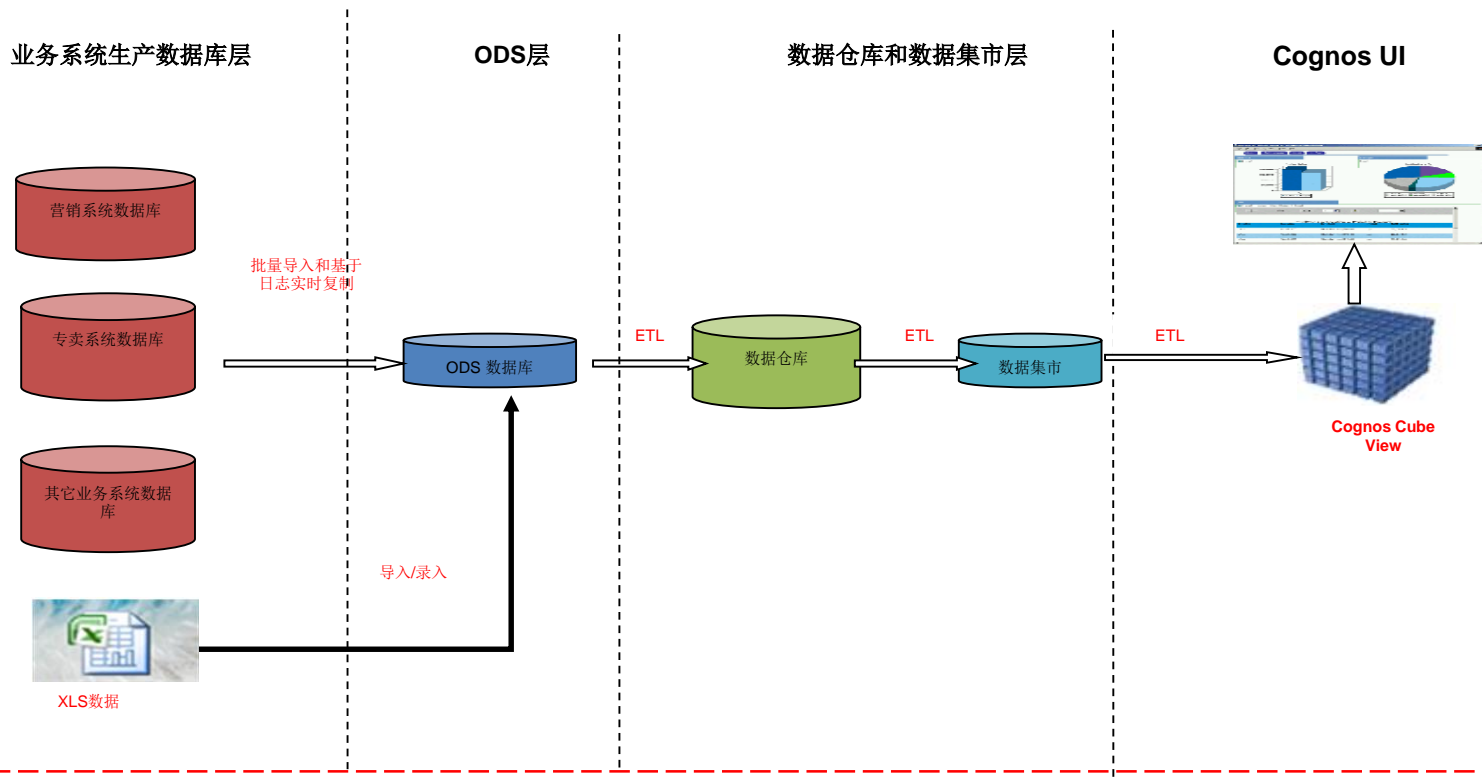
相应的介绍文档可以从<http://www.slideshare.net/> 网站上搜索获取

### 8.3 采用云计算技术后新系统的架构初探

在BA（数据挖掘分析等）领域我们遇到的大数据挑战更加严峻

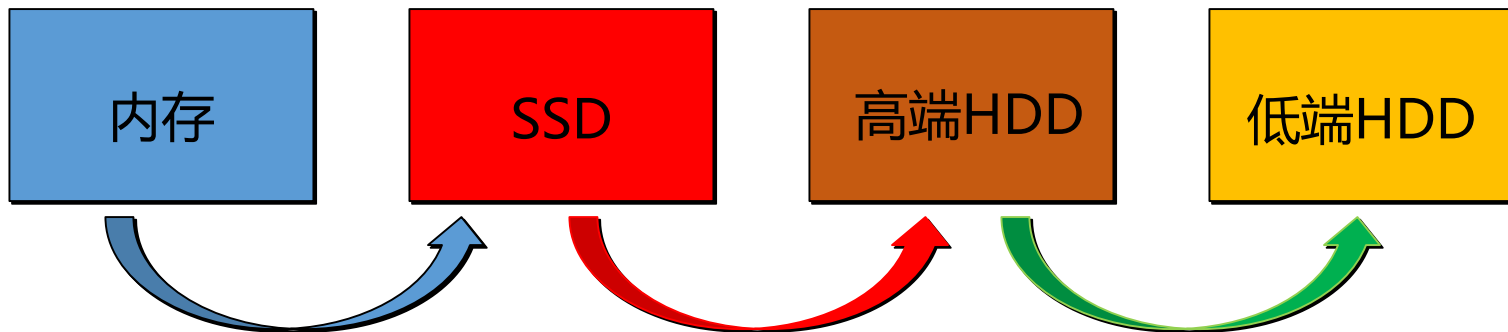
## 8.3 采用云计算技术后新系统的架构初探

### 很经典的一个BA系统的技术架构



## 8.3 采用云计算技术后新系统的架构初探

很经典的一个BA系统的技术架构



## 8.3 采用云计算技术后新系统的架构初探

### 很经典的一个BA系统的技术架构

#### 特点:

- 1.如果每天产生的增量数据不大（例如几十个G以下的话），在一定机器配置的情况下；进行相应地装载、清洗、数据挖掘等工作没有多大的问题。
- 2.如果数据量非常之大的话，一般会采用大机器、大内存、SSD、高端存储、高速网络（万兆网/Infiniband网络）来进行；但是成本可能非常巨大（例如硬件成本、正版的软件成本）；并且还会出现数据量上到一个量级别后甚至会出现硬件再好也无法应对。
- 3.实施的架构其实是非常复杂的
- 4.在大部分客户场景（很多客户没有那么多的数据）的情况下用此架构是合理有效的

## 8.3 采用云计算技术后新系统的架构初探

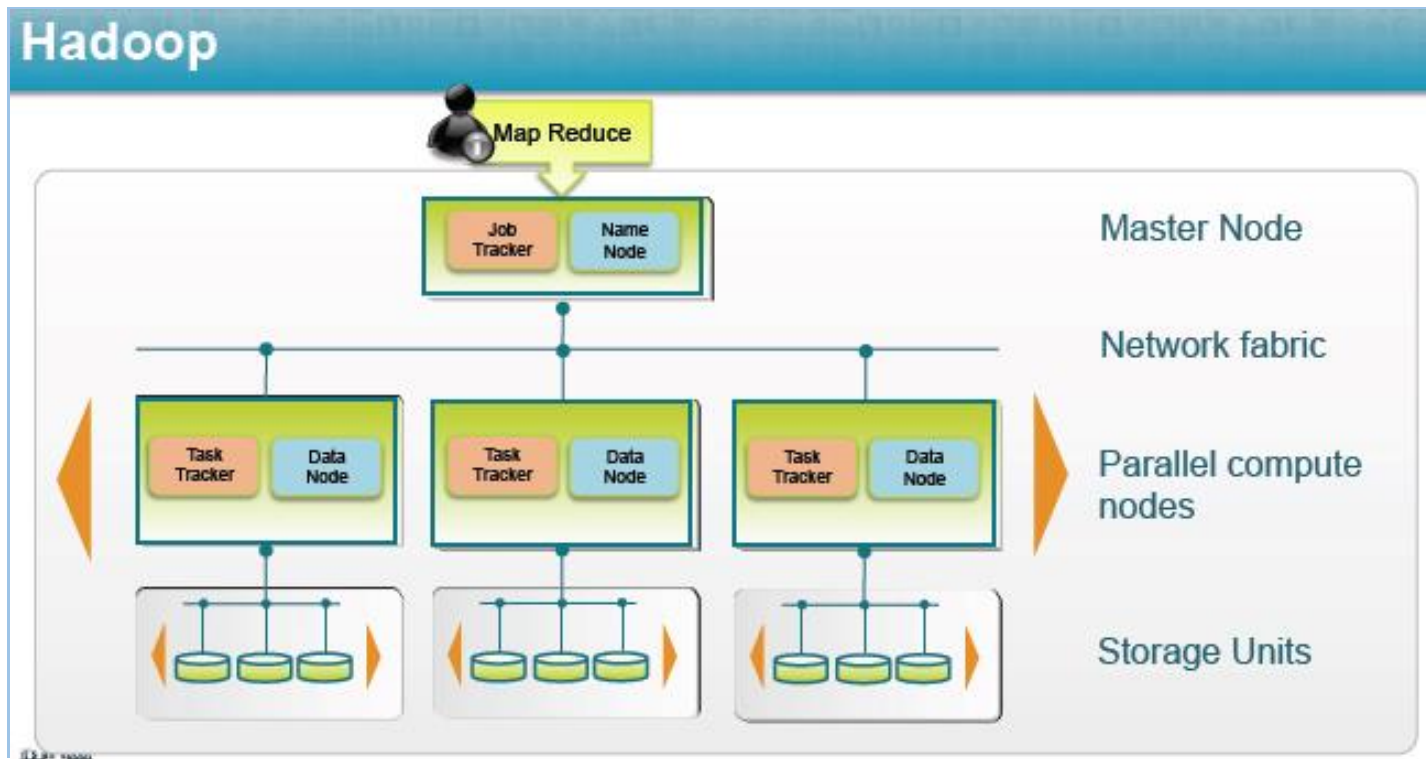
### 引入Hadoop框架：IBM InforSphere BigInsight

在我们的企业应用系统中，不仅仅是关系型数据还存在大量的非关系型数据都需要进行分析，例如日志、Office文档等等；并且在很多数据量（关系型和非关系型）增长到一个让人非常恐怖的时候（例如一些互联网应用），在做数据分析的时就必须采用Hadoop框架进行大规模的机器集群来进行处理。



## 8.3 采用云计算技术后新系统的架构初探

### 引入Hadoop框架：IBM InforSphere BigInsight



## 8.3 采用云计算技术后新系统的架构初探

### 引入Hadoop框架：IBM InforSphere BigInsight

业务系统数据源



关系型数据库数



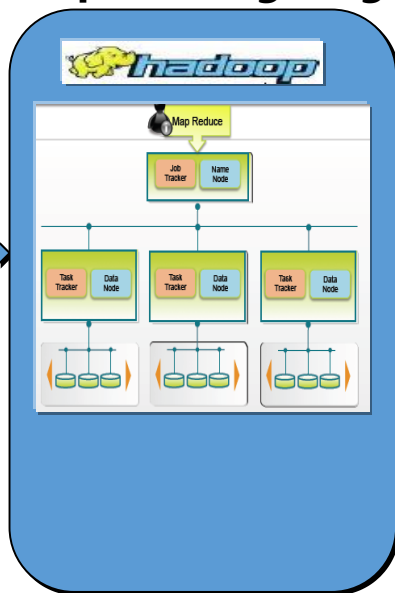
非结构化数据



半结构化数据

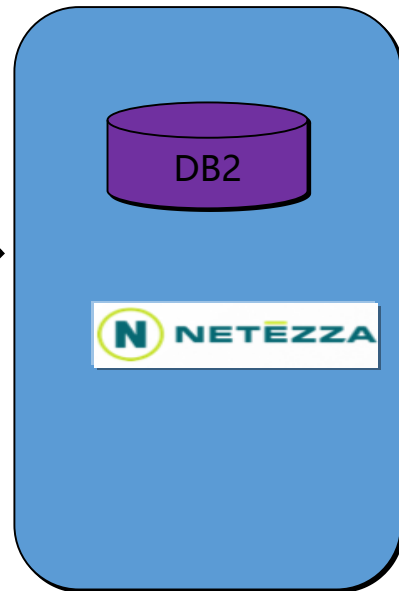
增量实时  
装载 / 全  
量 / 非实  
时装载

Hadoop集群: IBM  
InforSphere BigInsight



结果装载

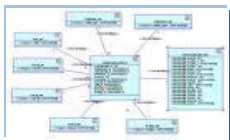
计算结果



DB2 10将能与Hadoop ( IBM InforSphere BigInsight) 无缝紧密的进行集成

## 8.3 采用云计算技术后新系统的架构初探

### 业务系统数据源



#### 关系型数据库数



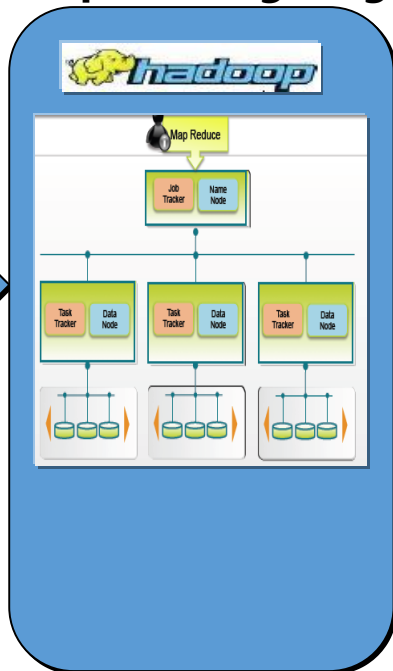
#### 非结构化数据



#### 半结构化数据

增量实时  
装载 / 全  
量 / 非实  
时装载

### Hadoop集群: IBM InforSphere BigInsight



结果装载

### 计算结果



### 数据可视化

SPSS

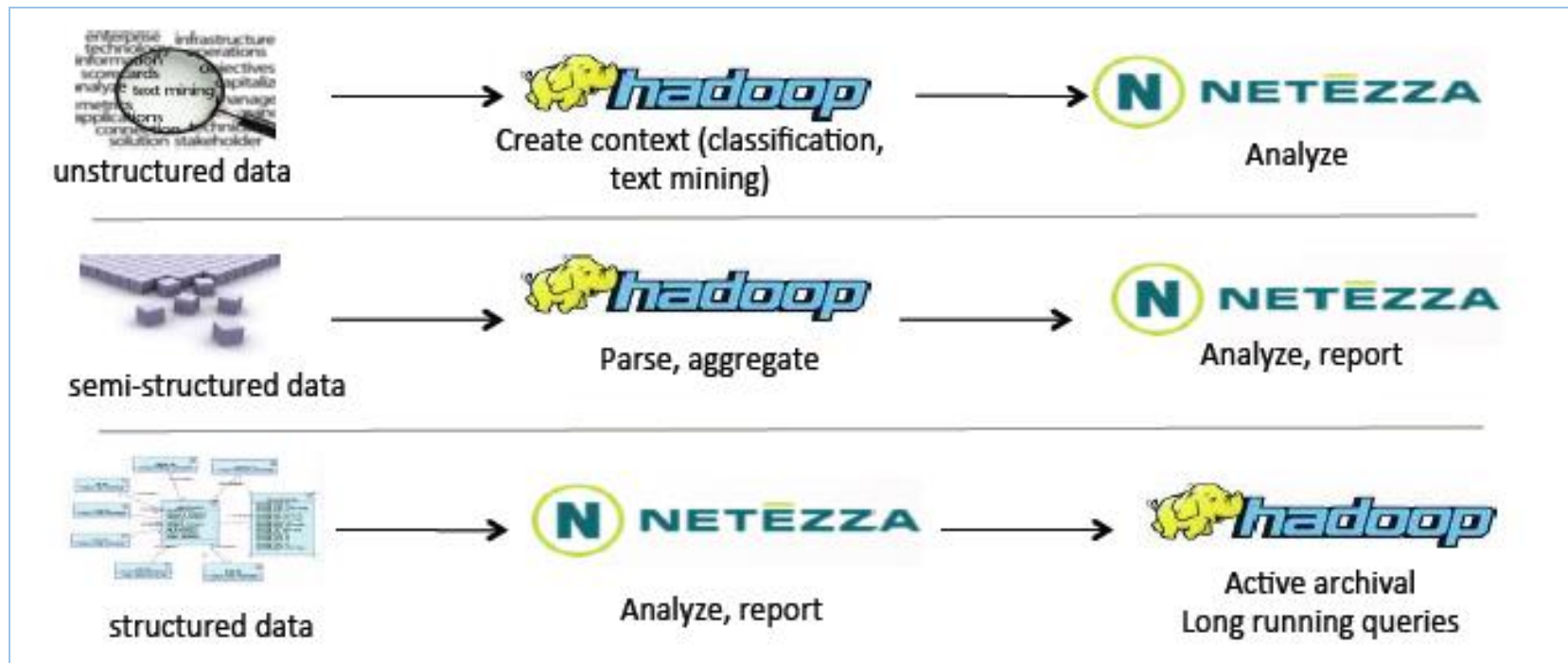
Cognos

其它

DB2 10将能与Hadoop ( IBM InforSphere BigInsight) 无缝紧密的进行集成

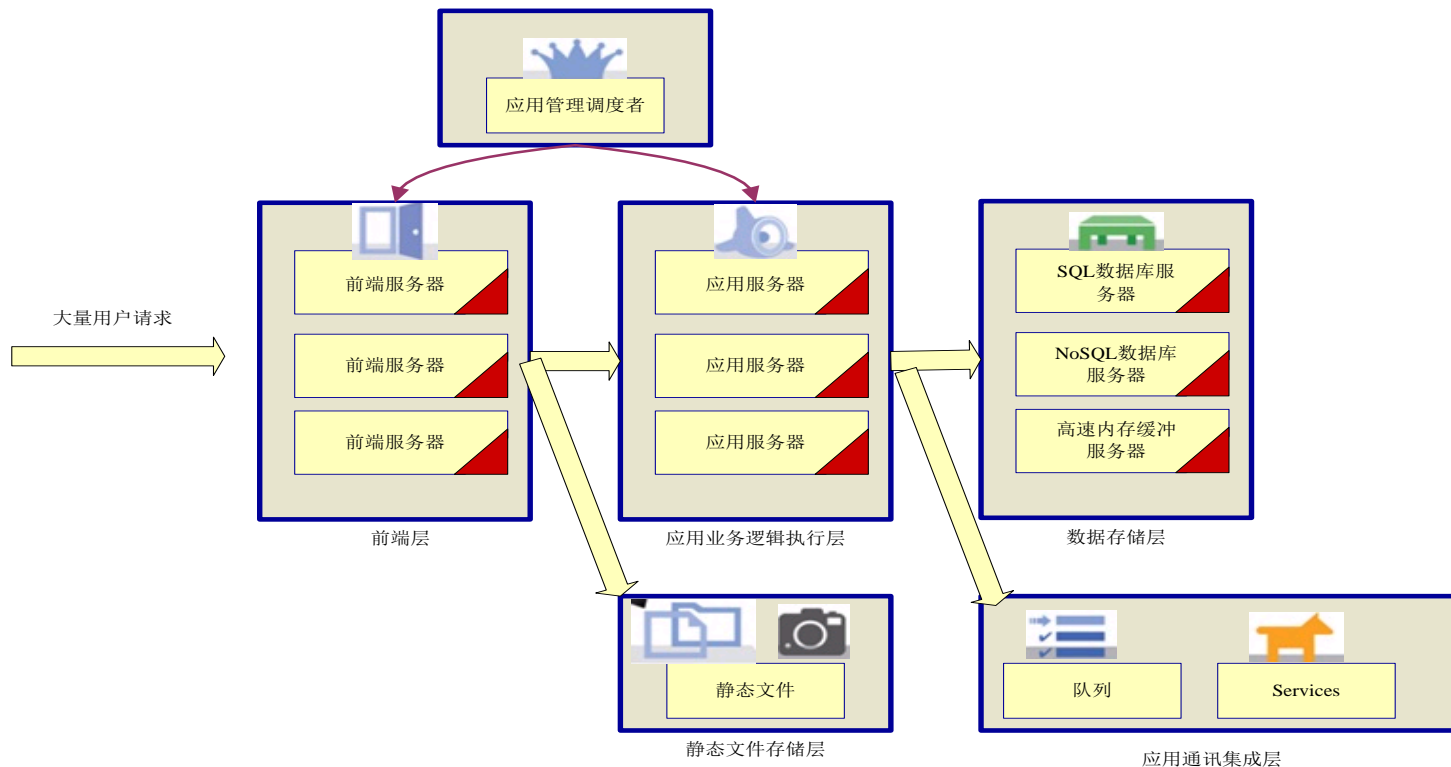
## 8.3 采用云计算技术后新系统的架构初探

### Hadoop框架与Netzza集成的混合场景



## 8.3 采用云计算技术后新系统的架构初探

### 架构的小结





End of Session  
Thank you!