



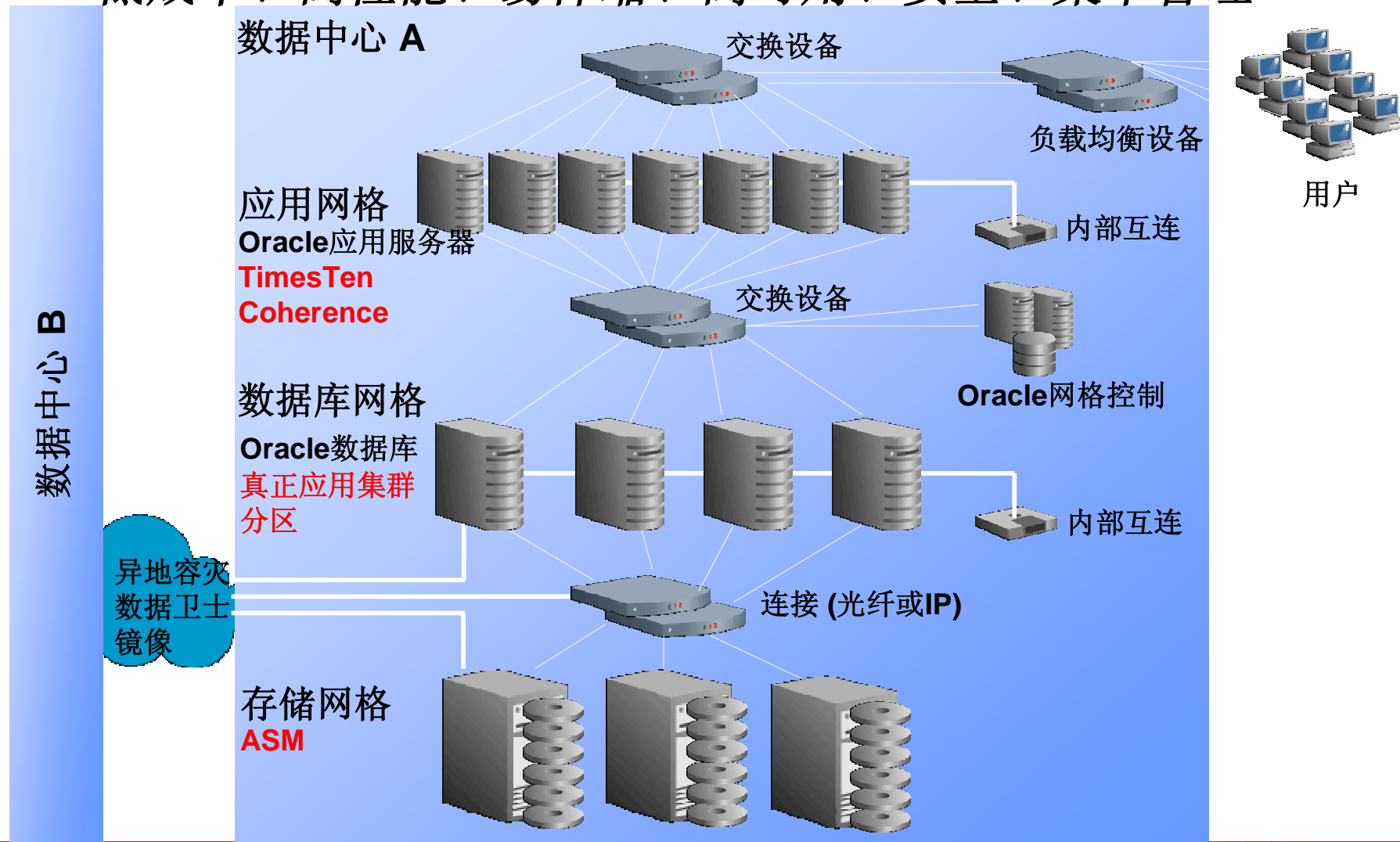
ORACLE®

## 金融行业高性能最佳实践

李宪文  
高级技术顾问

# 最为开放 完整和领先的架构 Grid+SOA

低成本、高性能、易伸缩、高可用、安全、集中管理



ORACLE

# 真实世界可伸缩性的领导者

- Winter TopTen™- 世界上最大的和负载最高的数据库调查
- 世界上最大的商业数据库
  - Yahoo @ 100TB
- 世界上最大的Linux数据仓库
  - Amazon.com @ 24.7TB
- 世界上最大的Linux OLTP数据库
  - Amazon.com @ 4TB
- 世界上最大的科学计算数据库
  - Max Planck @ 222TB
- 世界上最大的Unix OLTP数据库
  - US Patent & Trademark Office @ 16.4TB
- Oracle运行着十大中的九个最大的Unix OLTP数据库
- Oracle运行着十大中的十个最大的Linux DSS和OLTP数据库



ORACLE®

# 即将到来的里程碑

- 出现 Petabyte (1000 TB) 的数据库
  - 现在最大的是 300TB
  - LOB 数据快速的增长
- 出现 1000核的数据库
  - 现在最大的是 100 到 300 核
- 出现 Terabyte 缓存区
  - 在基准测试中已经使用了这么大的缓存区

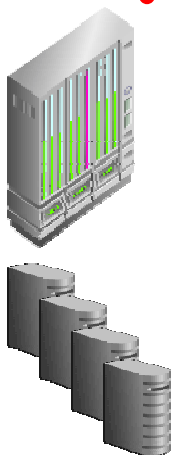


# Oracle已经为伸缩性做好准备

## - 25年的投入

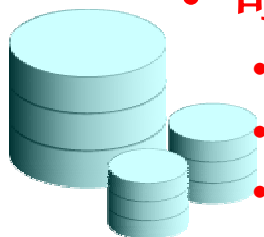
### • 可伸缩的执行

- Scale-up在大型SMPs
- Scale-out在集群服务器
- 并行操作
  - Query, DML, DDL, Loads, Unloads, Recovery
- 行级锁
- 读一致性



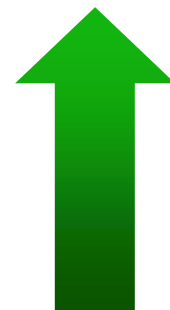
### • 可伸缩的存储

- 表分区
- 自动存储管理
- Bigfile表空间
- 可传输表空间
- 9种索引类型



### • 可伸缩的可用性

- RAC
- Flashback
- Data Guard
- RMAN
- 在线表重定义
- 只读表空间



### • 可伸缩的管理

- 网络控制
- 自动优化
- 数据库资源管理器
- 自动工作负载库



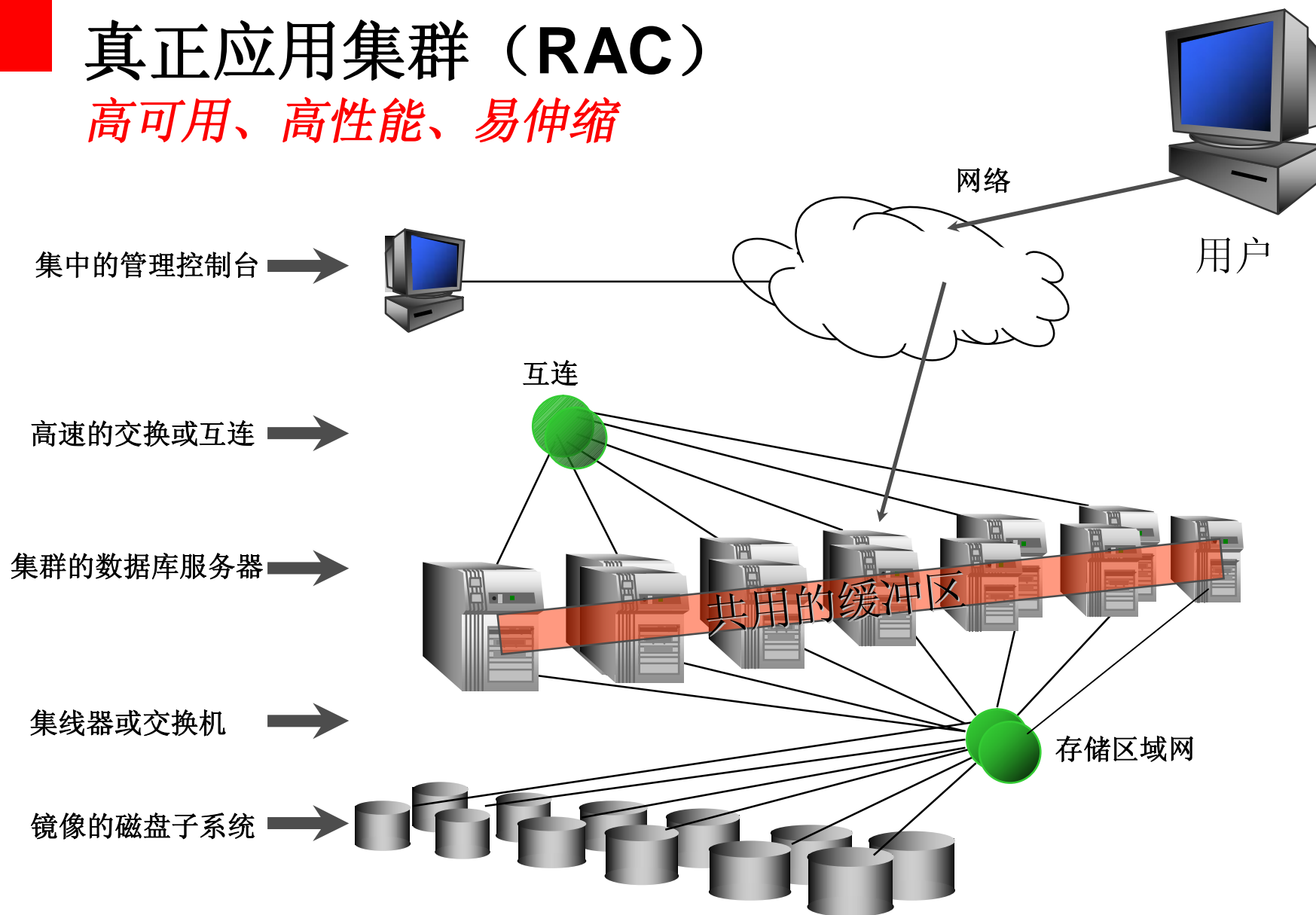
# SMP纵向扩展



- 非常成熟
  - 20年的经验
- 许多用户使用大型的**SMPs**
  - 64 到 128 CPUs
  - Sun E25K, HP Superdome, IBM Regatta
- 单一系统映像
  - 易于管理
  - 易于编写应用
- 工作的很好，但是成本很高
- 处理能力有上限
- 需要一台备机实现高可用性

# 真正应用集群 (RAC)

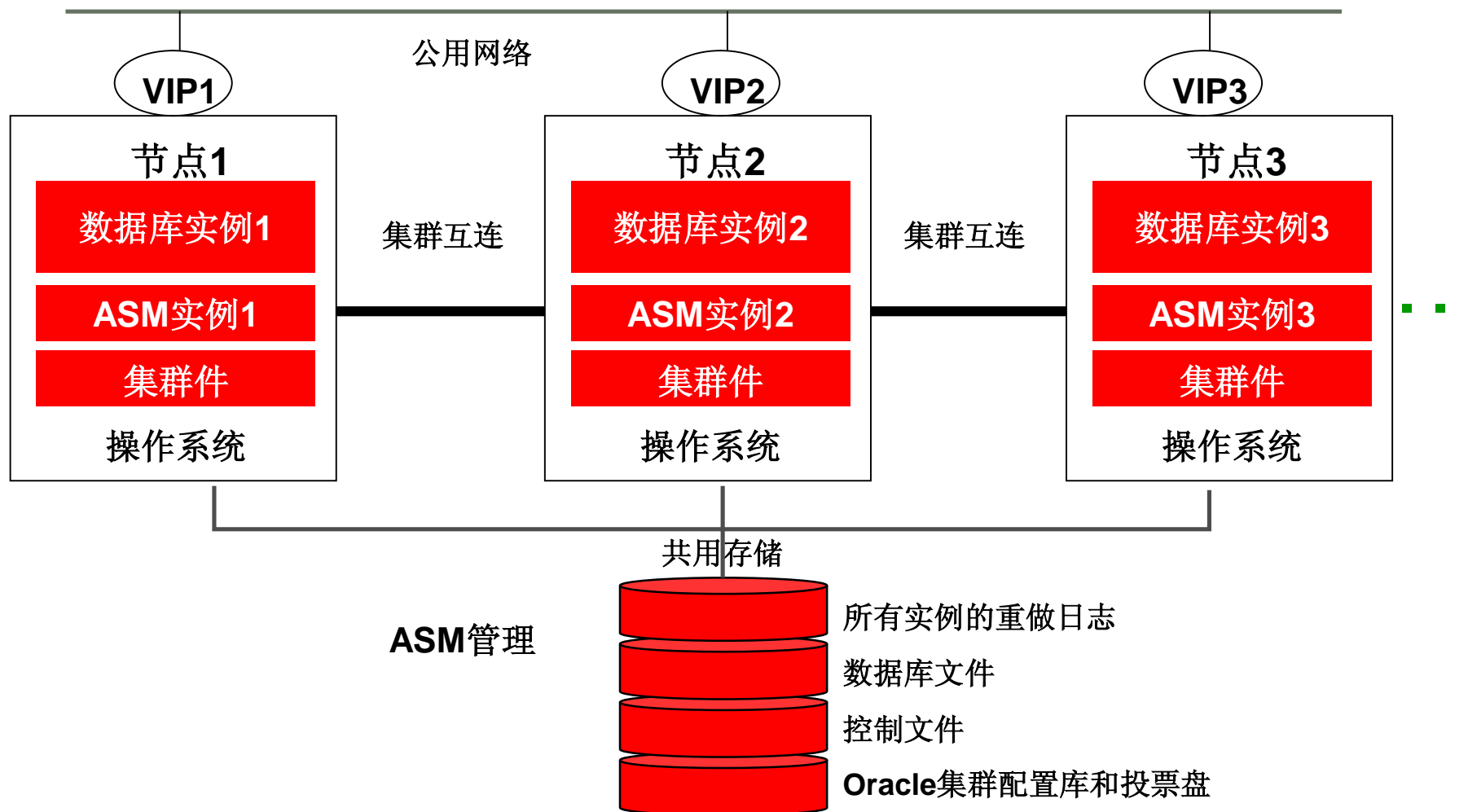
高可用、高性能、易伸缩



ORACLE

# 真正应用集群架构

## 共用缓冲区/磁盘架构

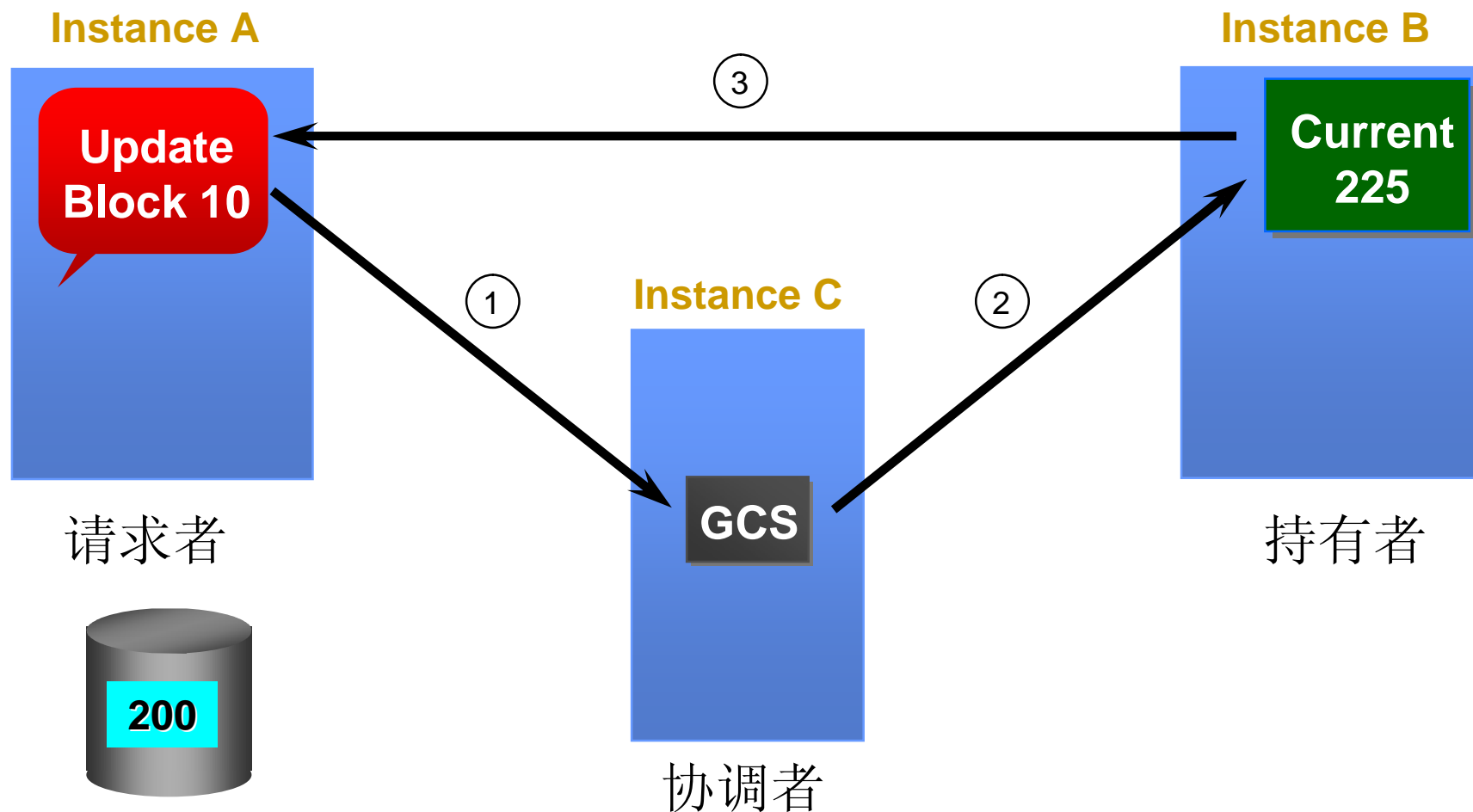


ORACLE



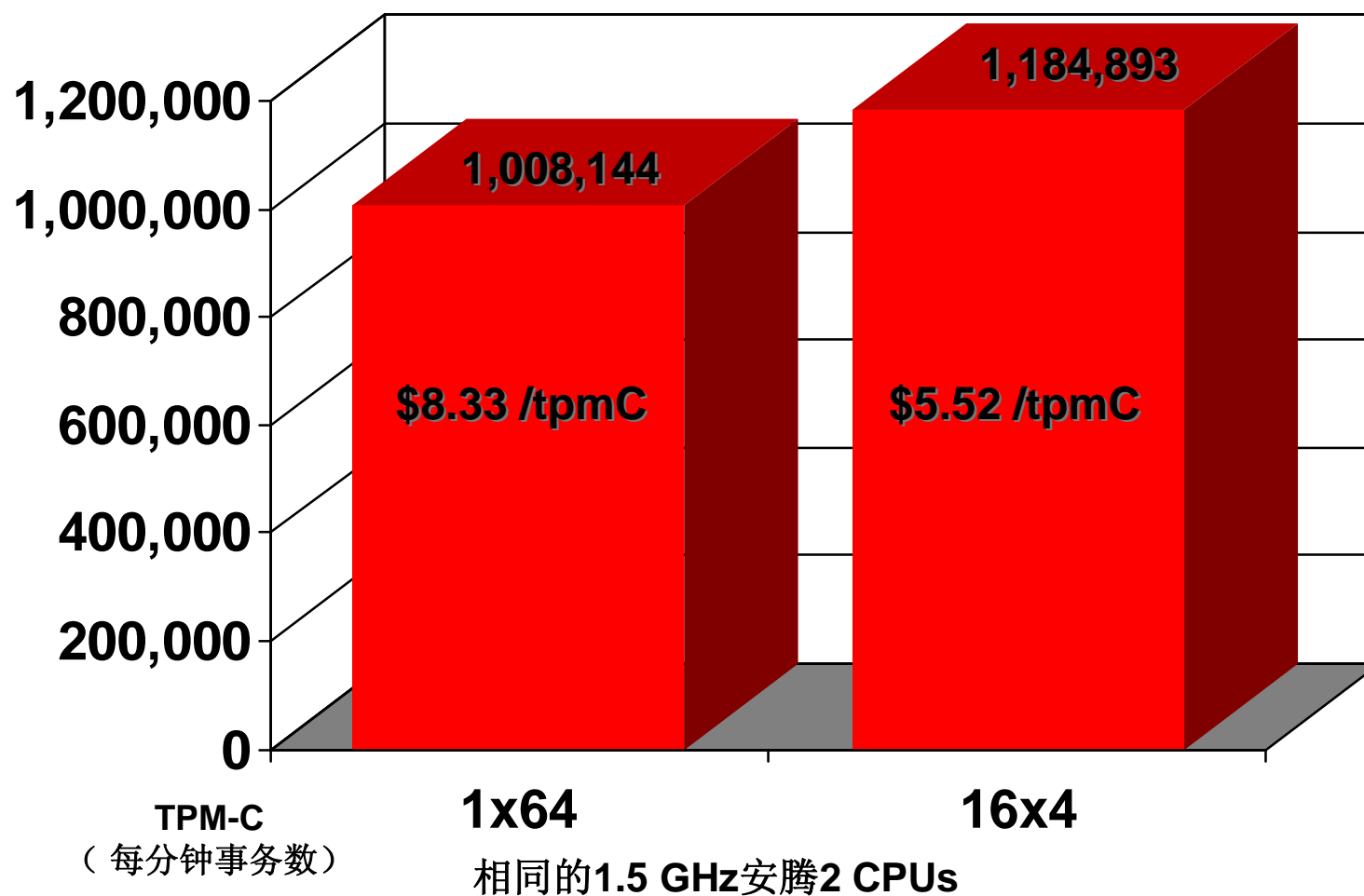
# RAC: 伸缩性

消息传递成本和集群大小无关



# 真正应用集群 (RAC)

## 低成本、高性能



ORACLE

# 案例：澳大利亚银行

## 无与伦比的成本与性能优势

- 最近将其5+ TB 集团数据仓库从SMP 43cpu Sun E10k 迁移到运行Oracle 10g的基于Intel的服务器集群上
- 在他们的生产机和新的Oracle集群上，他们运行了查询测试
- 结果对于他们，证明了Oracle的性能和成本优势
- Sun E10k = \$1M USD+
- 4 x Intel servers = \$150k USD

	Test 1	Test 3	Test 4	Test 5	Test 6
	GDW Production: 43CPU Sun E10K	OC: 16CPUs - 4x4 Intel Xeon Grid	时差	性能提升 Grid / RAC	
成本	US\$1M+	US\$150K			
Test 1	00:55:00	00:05:30	00:49:30	89.6%	
Test 2	06:00:00	00:28:55	05:21:05	89.1%	
Test 3	01:20:00	00:13:55	01:06:32	82.9%	
Test 4	01:30:00	00:58:17	00:31:43	64.3%	
Test 5	04:20:00	00:28:34	03:51:26	88.9%	
Test 6	00:30:00	00:02:30	00:27:30	91.0%	

ORACLE

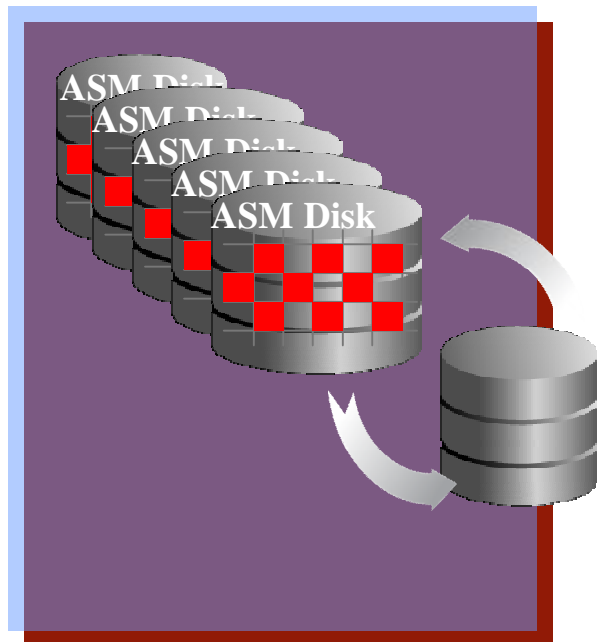


# 充分验证的可伸缩性

## RAC 8+节点的用户

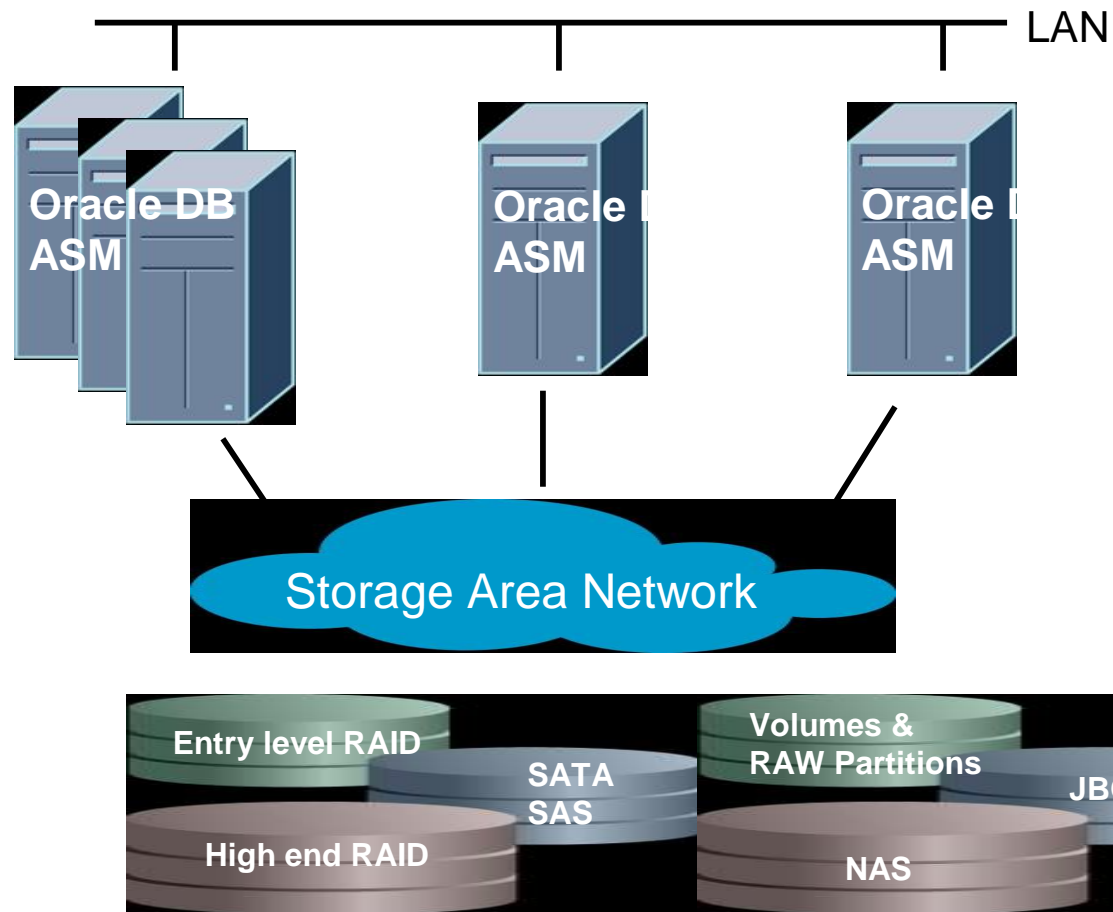
- Citigroup
- Burlington Coat Factory
- J2 Global Communications
- Genworth Financial
- Amazon.com
- MSDS
- Mercado Libre
- Yahoo! Overture
- Babcock Engineering
- Ordnance Survey
- Dell
- Yahoo!
- SAIC
- Fairmont Hotels
- ADESLAS
- Evite.com
- Quelle AG
- Telstra
- Gas Natural
- MyTravel
- Thompson
- AOL
- Vivo
- Sagawa Kyubin

# 自动存储管理ASM

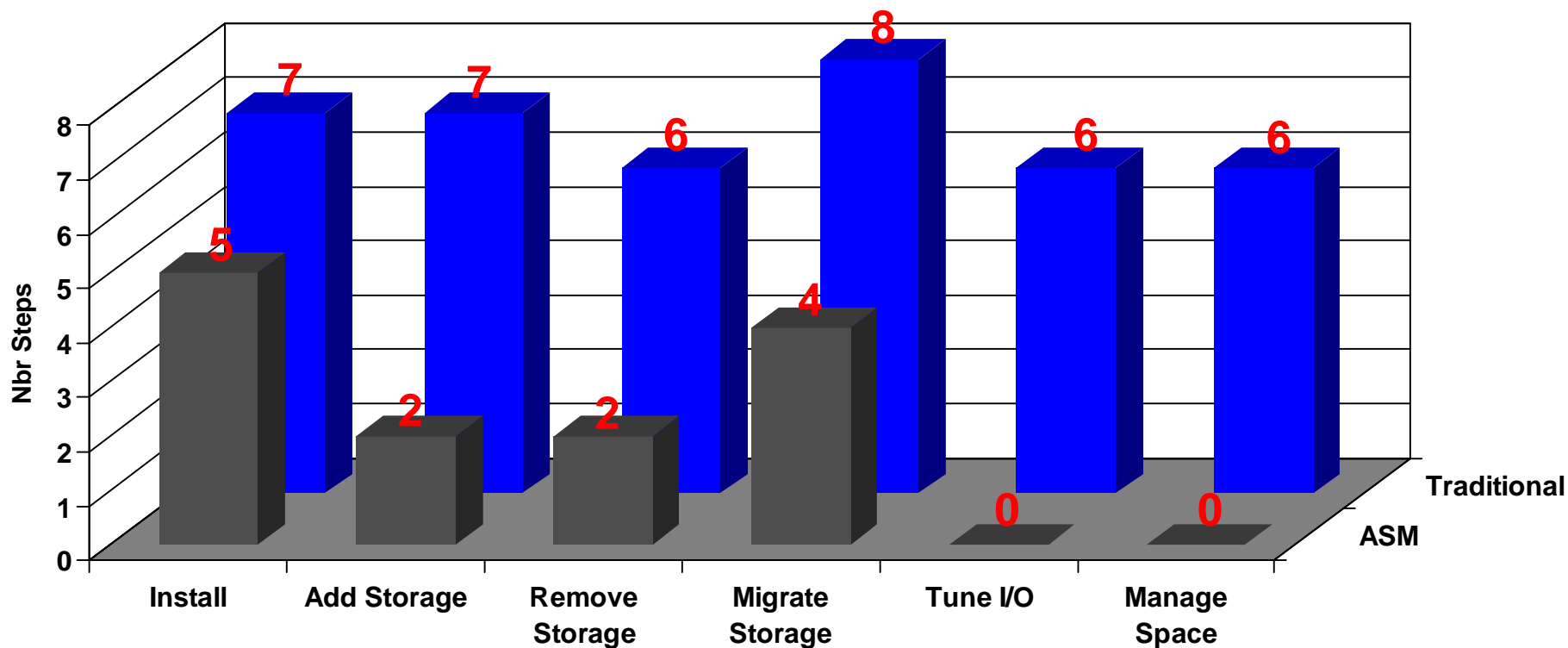


- 是推荐的和最好的**Oracle**数据库存储管理方式
  - 比文件系统易于管理
  - 相当于裸卷的性能
  - 内置在**Oracle**数据库
  - 所有数据库的共享存储池
- **免费**, 广泛使用
  - > 65%的10g RAC部署在ASM
  - > 25%的10g用户在使用ASM
  - 许多VLDB超过10TB

# 自动存储管理



## 使用ASM降低数据管理成本

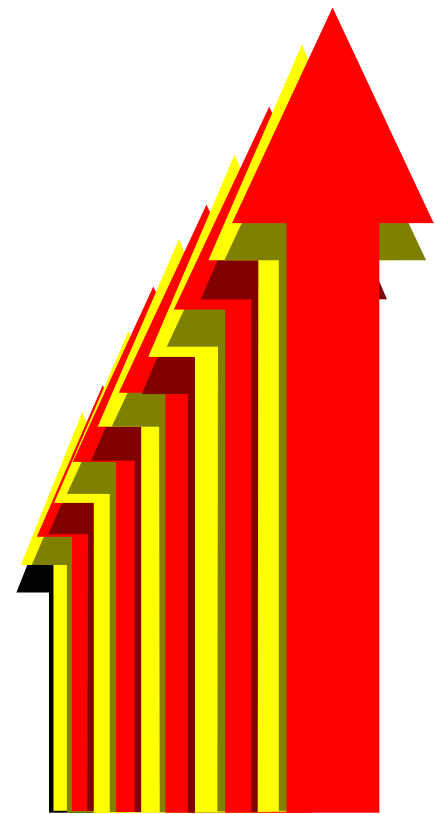


“节省成本的最好方法是减少复杂性”



# 为何要数据分区（Data Partitioning）

- 表与索引数据的爆炸性增长
  - 在大型数据库系统中 表数据量通常  $\gg 10\text{GB}$
  - 现在表的数据量比以前中等规模的数据库还大 !!
- 超大规模表与索引的要求
  - 部分数据区失败不会对其他数据区造成影响
  - 数据管理操作可在固定的维护窗口下运行
  - 数据大量增长的情况下，保持执行效率
- 解决方案：Data Partitioning
  - 分而治之



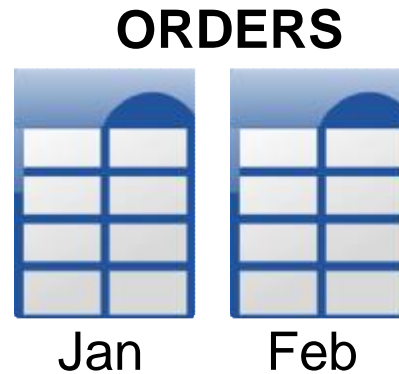


# Oracle 分区 (Partitioning)

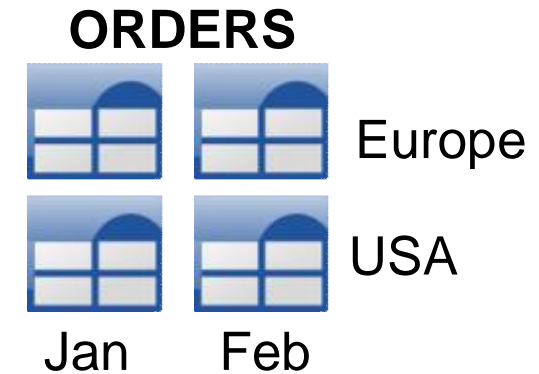
高性能、高可用、易管理



大表  
难于管理



分区 (范围、哈希、列表)  
分而治之  
易于管理  
提高性能



组合分区 (范围-哈希, 范围-列表)  
更高性能  
更灵活的适合业务需求



# 分区裁剪

- 减少读取数据的内部机制
  - 只有相关的分区会被访问
- 对应用透明
  - 自动分析SQL语句
- 两种分区裁剪方式
  - 静态裁剪
  - 动态裁剪

# 静态分区裁剪

```
select sum(amount_sold) from sales
where times_id
between '01-MAR-2004' and '31-MAY-2004';
```

- 相关的分区在解析时知道
  - Look for actual values in PSTART/PSTOP columns in the plan
- 优化器对SQL语句有准确的信息



# 动态分区裁剪

04-Jan

04-Feb

04-Mar

04-Apr

04-May

04-Jun

- 复杂查询的高级裁剪机制
  - E.g. beneficial for Star schemas
- 循环语句在运行时评估相关的分区
  - Look for the word 'KEY' in PSTART/PSTOP columns in the Plan

```
select sum(amount_sold)
  from sales s, times t
 where
    t.time_id = s.time_id
 and t.month in
    ( '05-Mar' , '05-Apr' , '05-May' );
```

ORACLE

# 完全分区智能的连接

## ✗ 相同分区的表会智能的处理

Partitions containing the same Subset of data are joined

Joins returning no data are suppressed right away

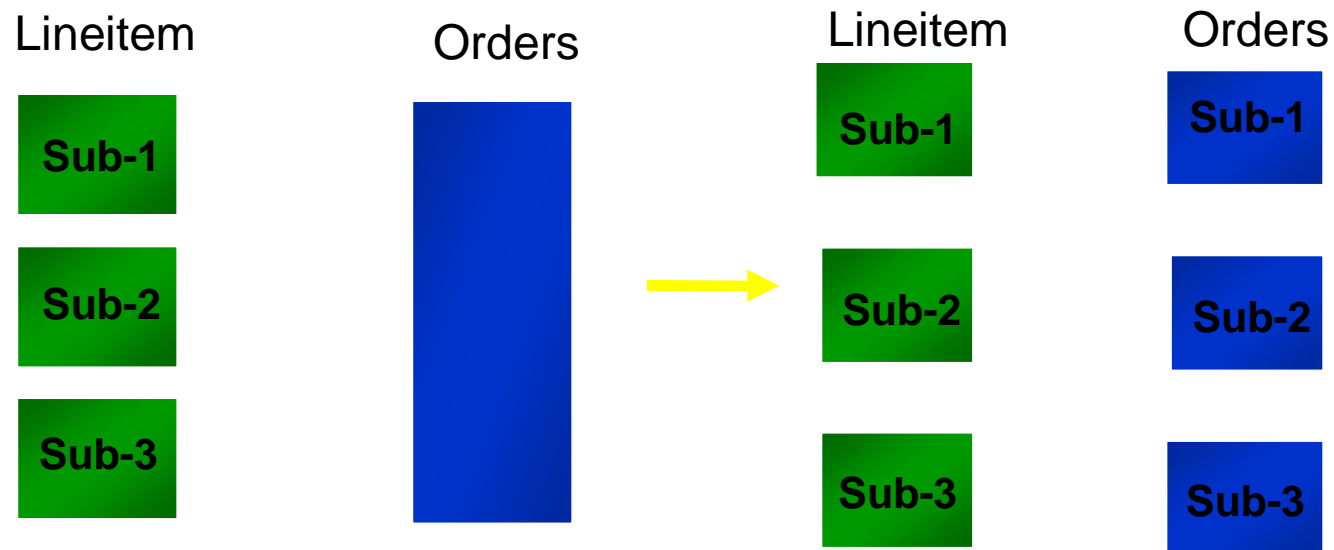
## ✗ 透明的性能提高

Common improvement of 15 – 30%

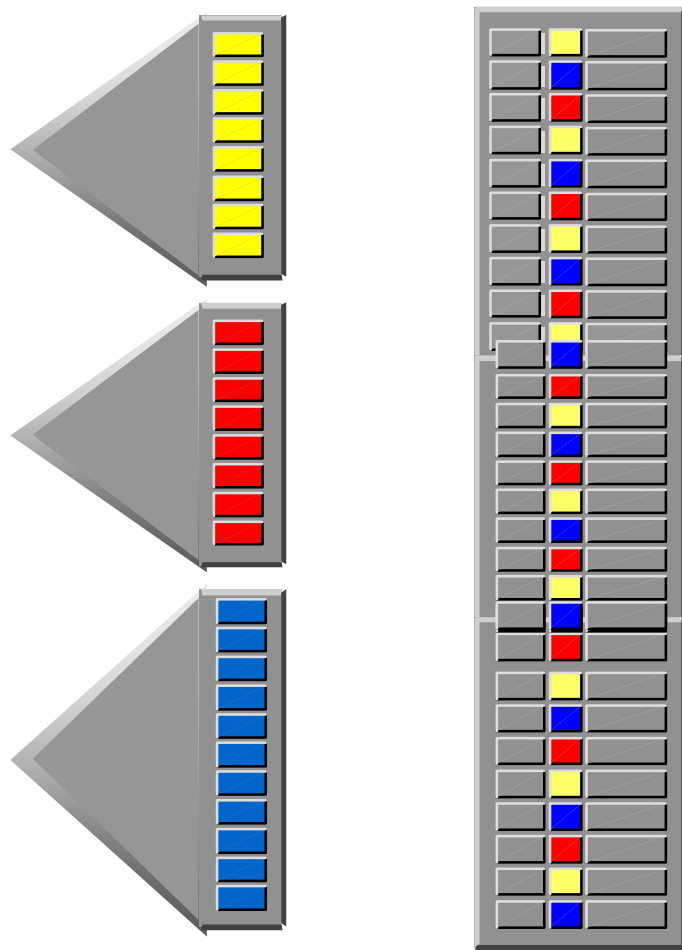


## 部分分区智能的连接

- If Lineitem table is partitioned by the join key, then Orders table can be re-distributed to enable partition-wise join



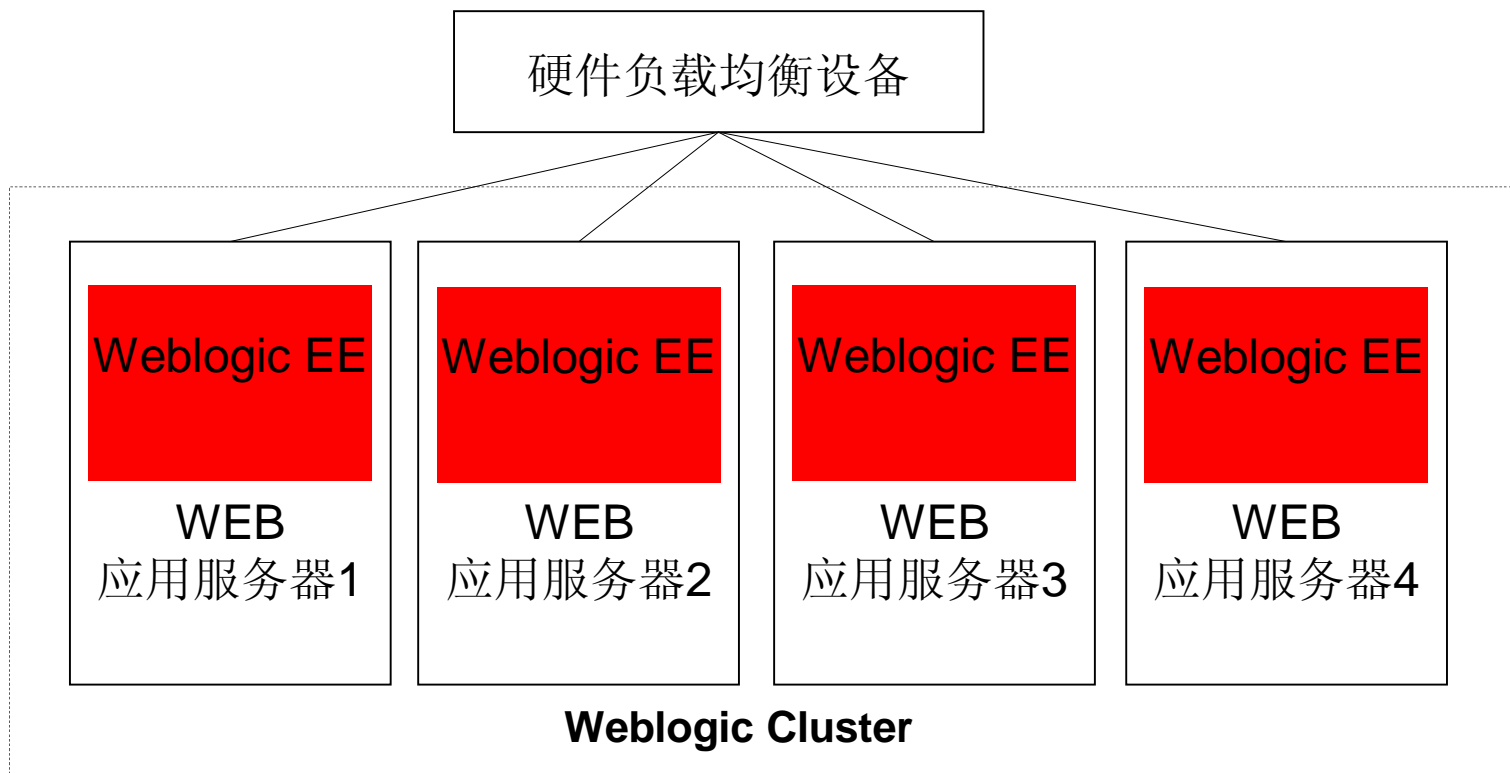
# OLTP索引热点消除



- 数据以单向增长的方式插入
  - 唯一键值索引
  - 在打包应用中常见
- 全局哈希分区索引
  - 数据插入到多个分区中
  - 消除热点

# Weblogic Enterprise Edition

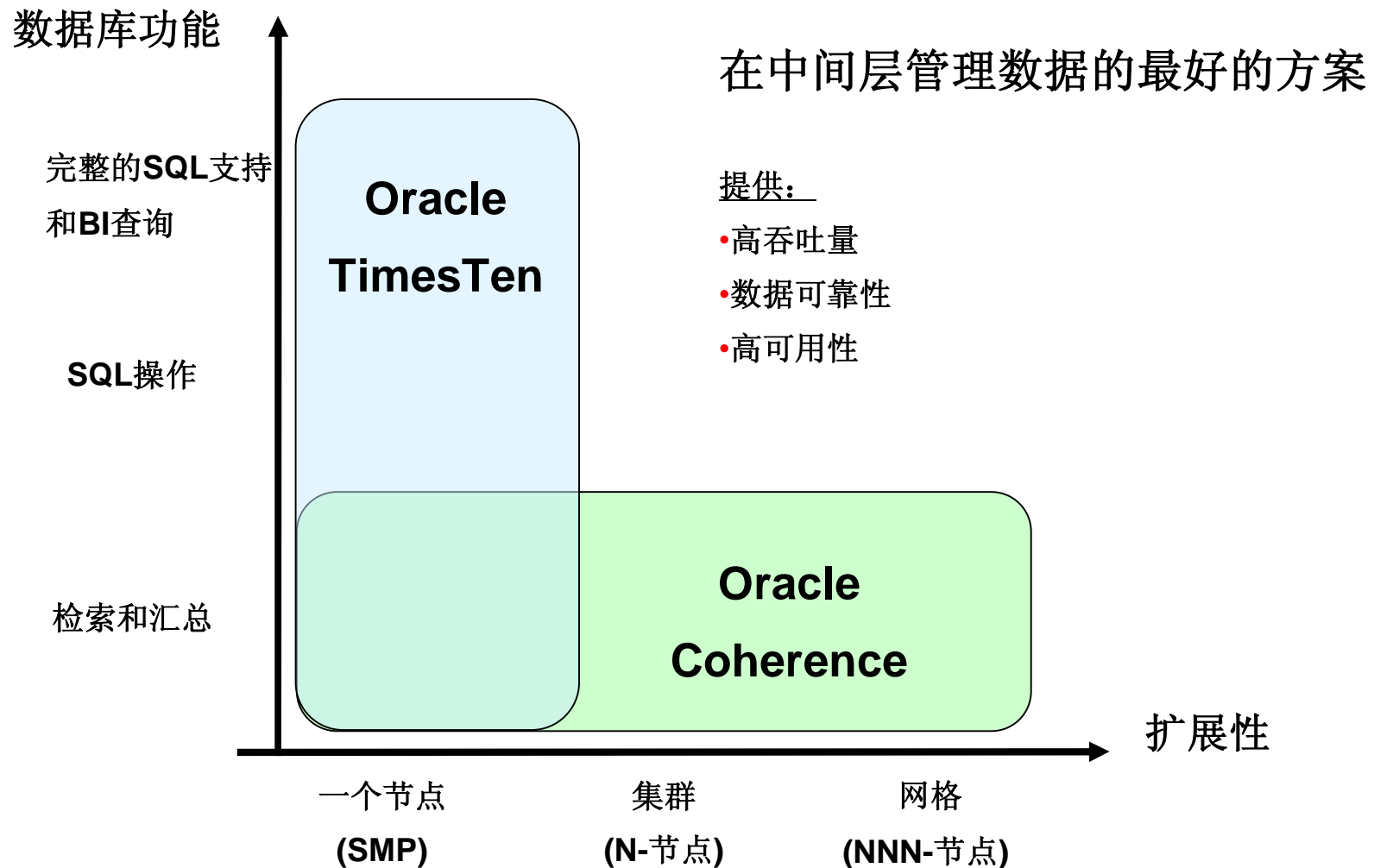
## 构建可靠的Web应用集群



- 通过硬件设备提供可靠的负载均衡
- 通过Weblogic集群提供应用的高可用性



# TimesTen和Coherence

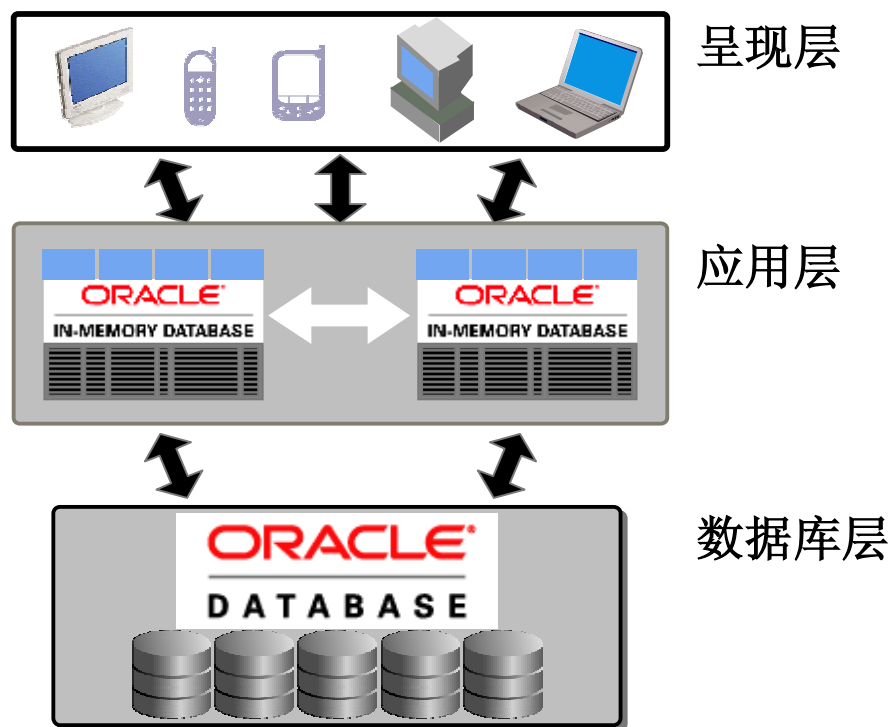


ORACLE

# Oracle TimesTen内存数据库

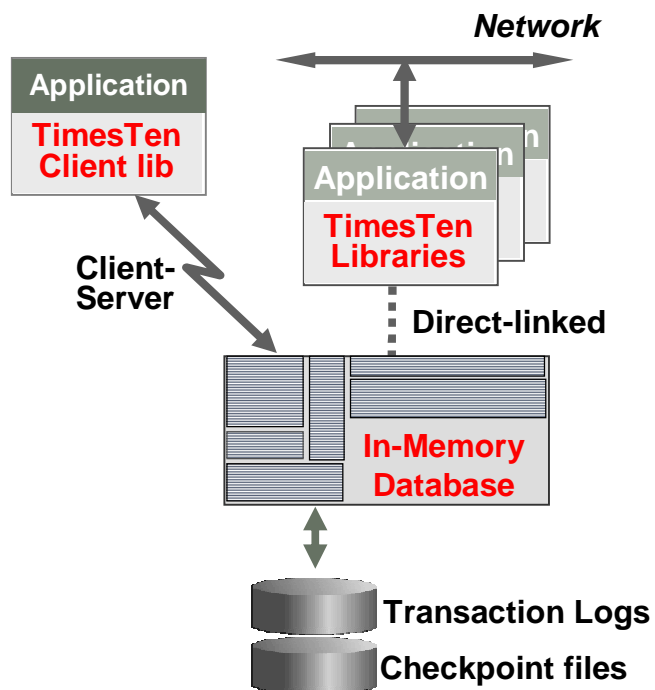
针对物理内存而优化的应用层数据库，可以获得极高的响应速度并实现实时数据缓存

“针对性能要求高的系统的应用层数据管理解决方案”



ORACLE

# TimesTen内存数据库

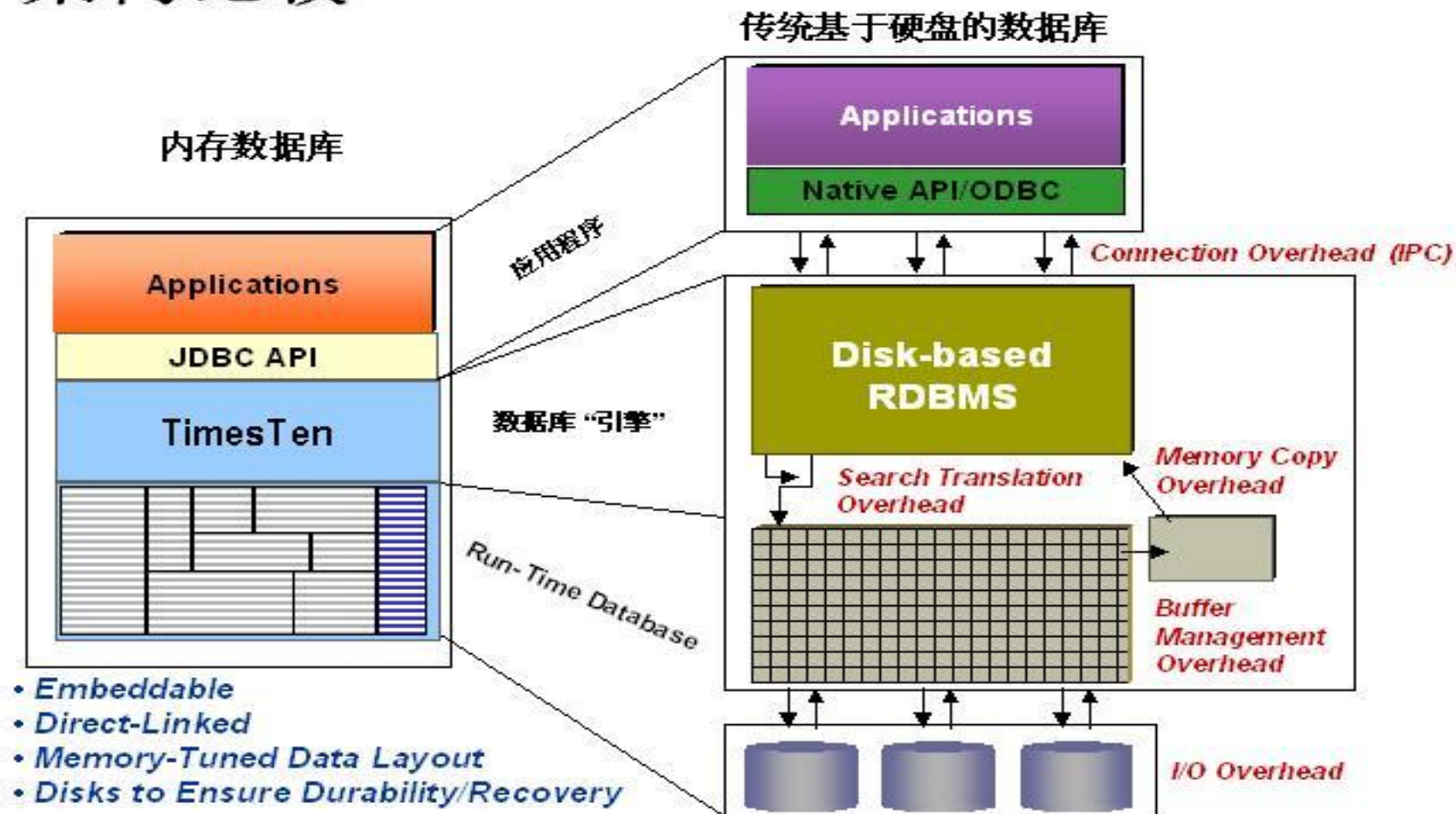


- 内存中的RDBMS
  - 全部数据库在内存中
  - 标准的访问接口ODBC/JDBC, SQL 92
  - 与Oracle数据库兼容
- 极高的性能
  - 极快的响应时间
  - 高吞吐量
  - 支持嵌入式
- 持续性和持久性
  - 数据库存放在磁盘上
  - 支持事务的ACID属性
- 实时的服务
  - 在线的非阻塞的操作
  - 实时的数据库变更通知
- 几乎不需要管理

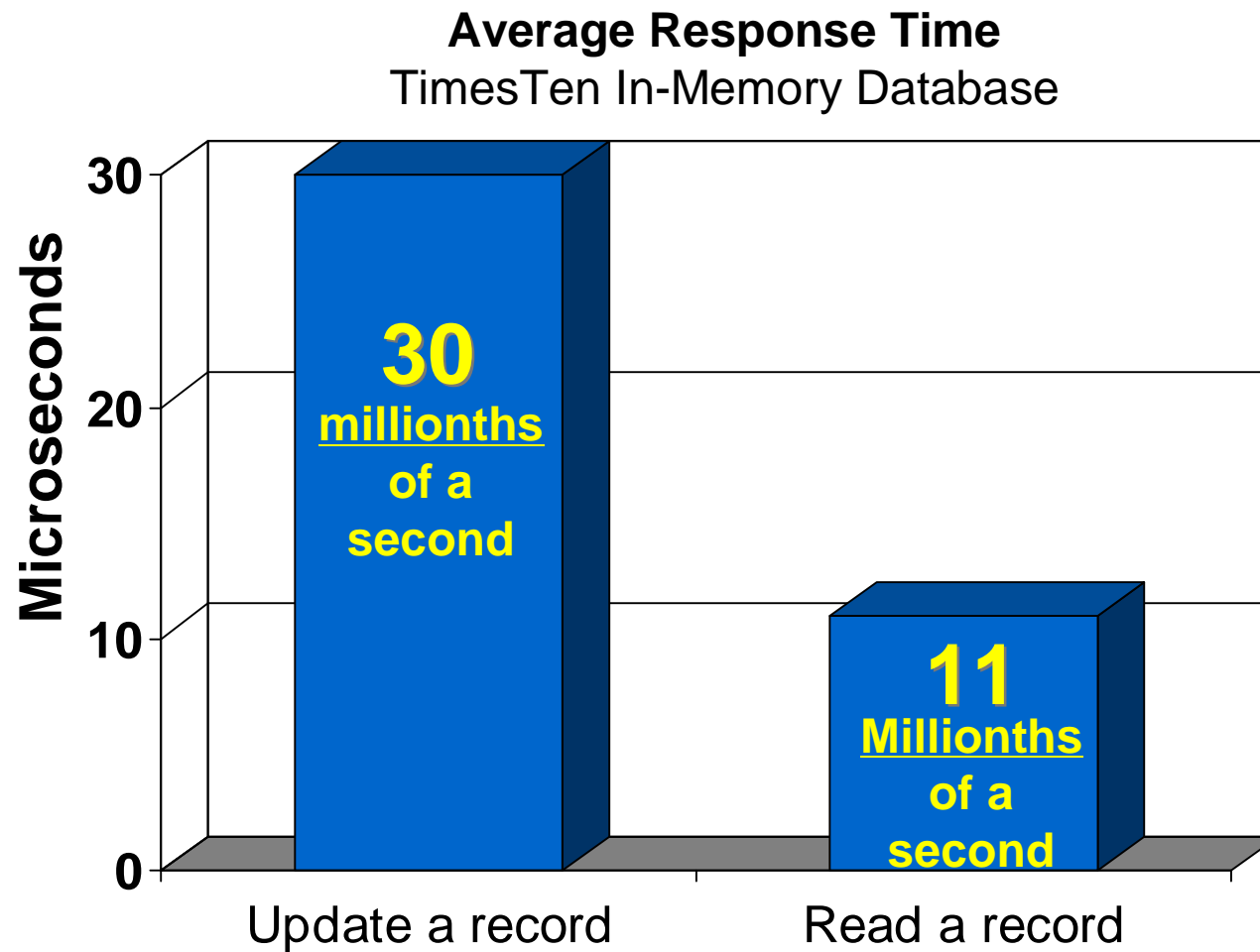
# 内存数据库与磁盘数据库的比较

数据库特性	Oracle Database 10g	Oracle TimesTen In-Memory Database
目标应用	关键业务应用	关键业务应用
数据模型	关系型 - SQL	关系型 - SQL
优化	磁盘为中心	内存为中心
典型部署方式	数据库层	应用层
架构	客户机/服务器	嵌入式（直接数据访问）
响应时间	毫秒到秒级	微秒到毫秒级
数据容量	TB到PB级	GB级
可扩展性	Cluster (RAC) & SMP	SMP

# 架构比较

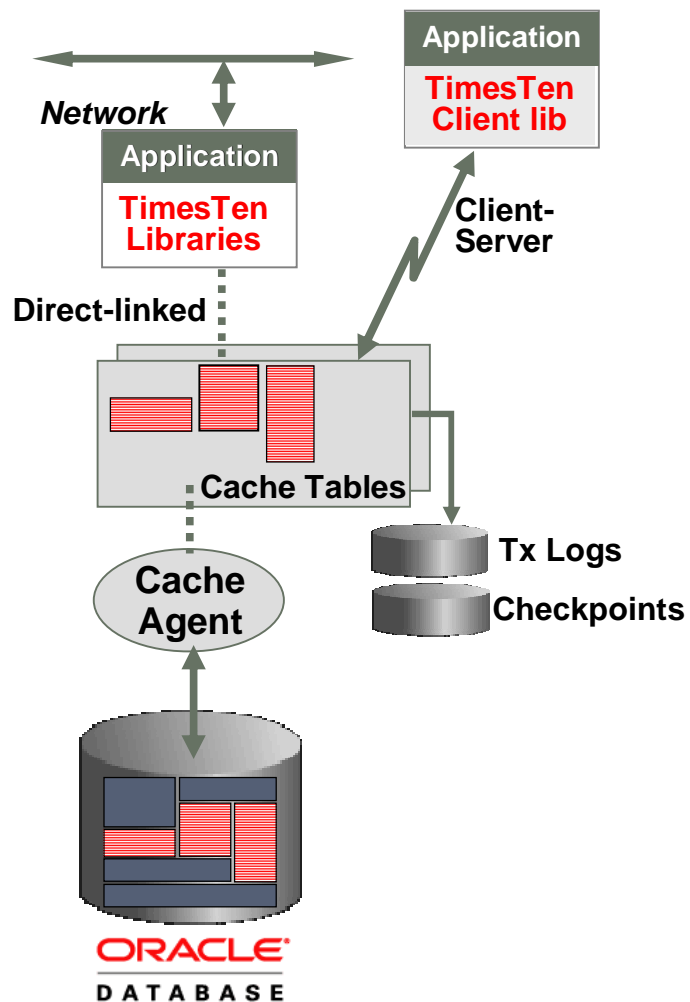


# 极快的响应速度



ORACLE

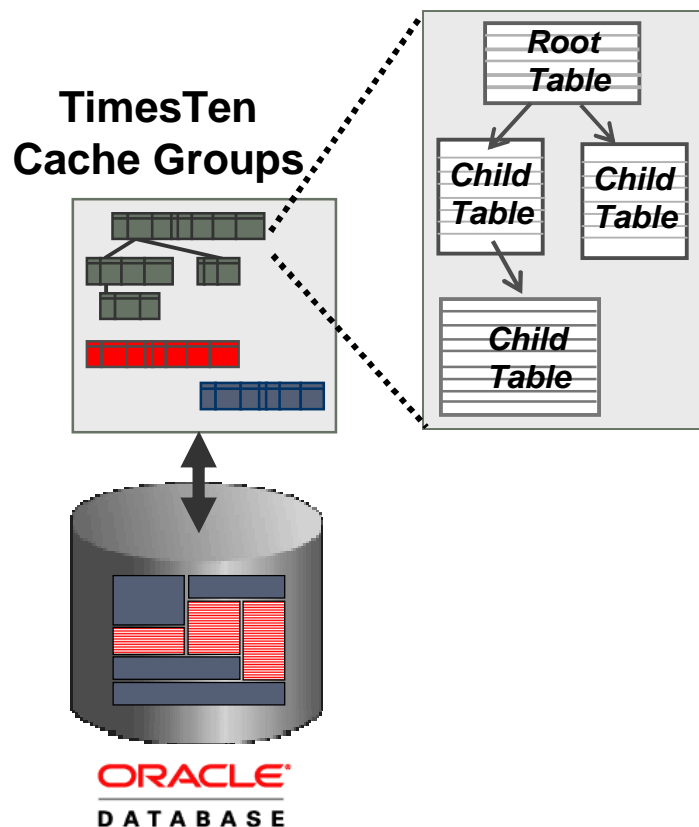
# Cache Connect to Oracle



- 从Oracle数据库缓存表
- 用户配置的缓存组
  - 缓存独立的表或有关联的表
  - 缓存行和列的全集或子集
- 只读的或可更新的
  - 想常规的数据库表一样访问缓存表
- 自动的数据同步
  - 从TimesTen到Oracle
  - 从Oracle到TimesTen

# 从Oracle数据库缓存数据

**Cache Group**—描述缓存在TimesTen内存数据库中的Oracle数据，支持SQL WHERE语句



例子：缓存多于500个购买订单的重要客户的名字和地址

```
CREATE CACHE GROUP PremierUsers  
FROM CUSTOMER (  
    NAME VARCHAR2(100) NOT NULL,  
    ADDR VARCHAR2(100) )  
WHERE CUSTOMER.ORDER > 500;
```



# 案例：ODS 项目

## TimesTen 用于

- § 解决数据库服务器性能处理瓶颈
- § 提升BO报表展现的性能

## 性能指标

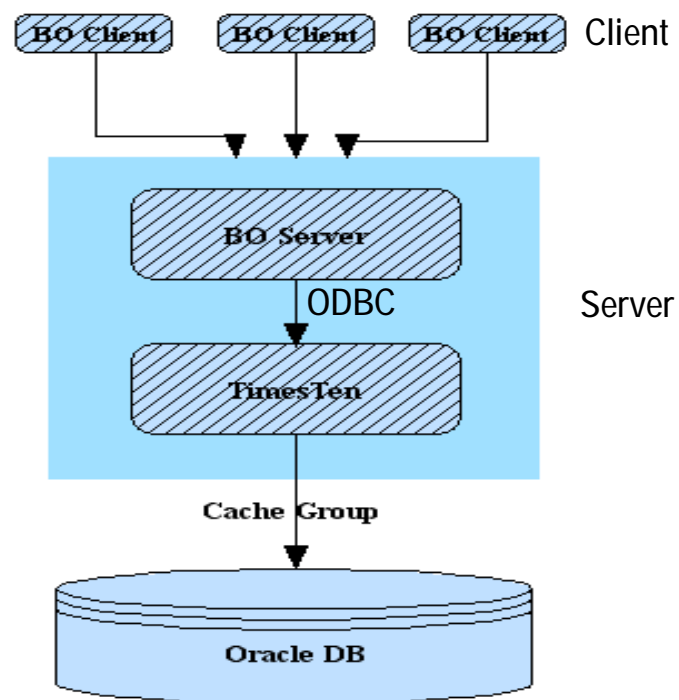
- § 展现速度提高在3倍以上

## 配置

- § 4-CPU server
- § AIX 平台 v5.0, 7856MB
- § TimesTen6.0.2
- § Oracle DB 9i
- § BO Server 为 BusinessObject XI

## TimesTen的价值

- § 提高展现速度
- § 移植工作量较少





# 什么是Coherence

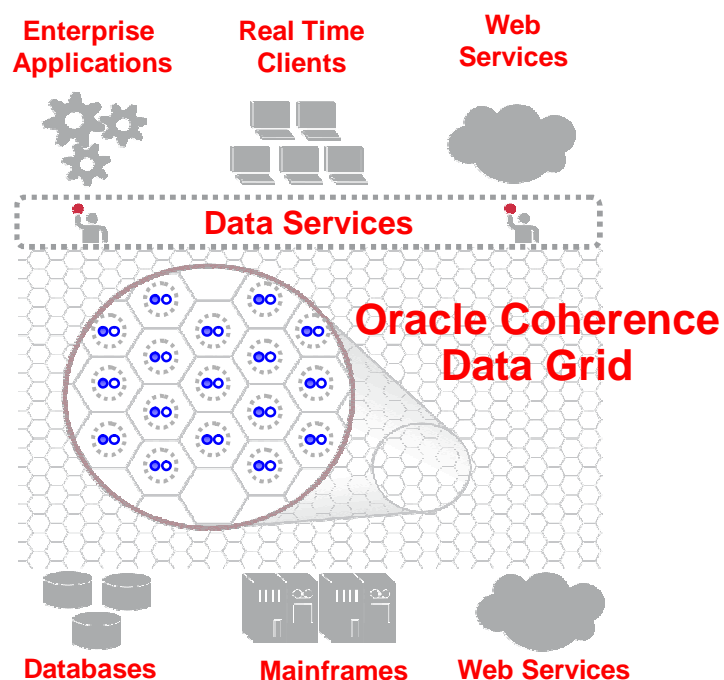
- Tangosol, 领先的可靠内存数据网格软件供应商
- 支持极速的事务处理 (XTP)
- 标准的Java Archive “JAR”, 标准的动态链接库“DLL” for .NET
- 使用于应用层
- 比应用服务器更强的数据管理
- 比应用服务器更好的伸缩性能

“最复杂`java.util.Map`实现”

# Oracle Coherence数据网格

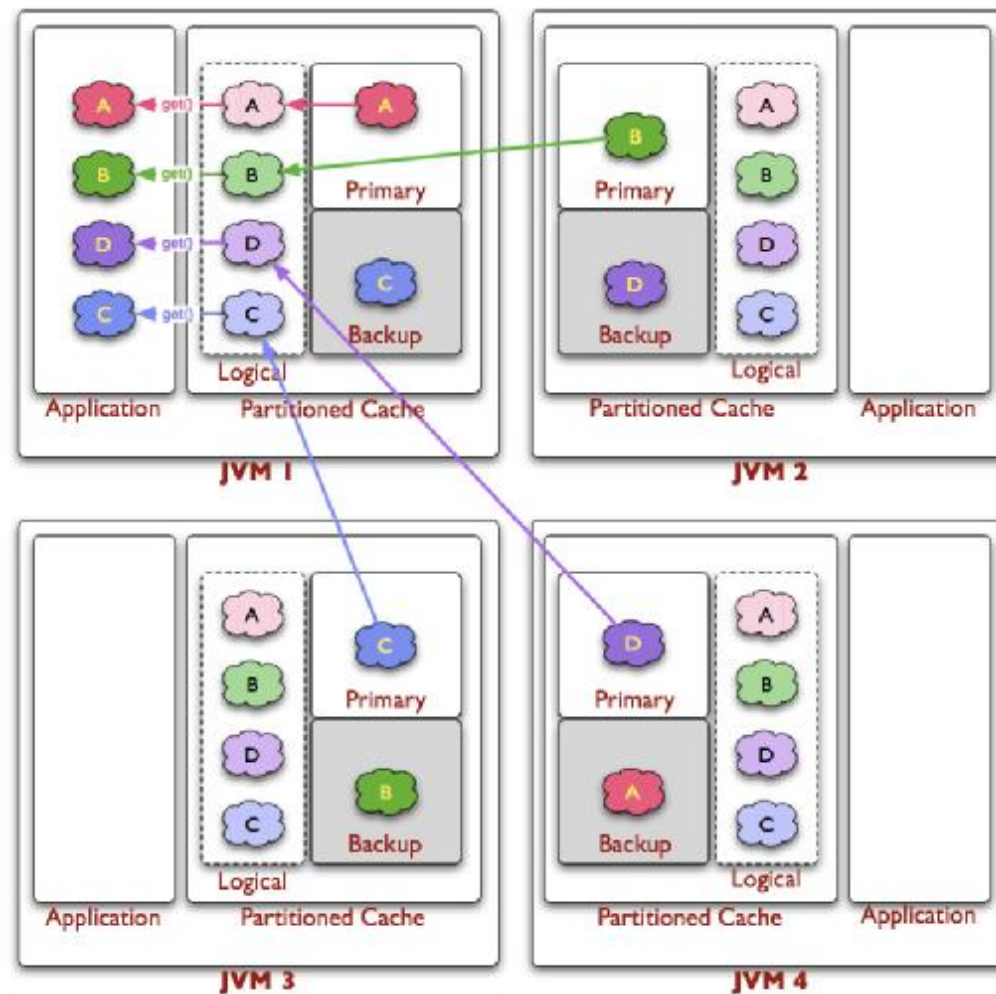
分布式内存数据管理

## 数据虚拟化

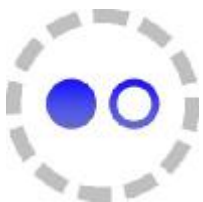


- 通过一个单一的一致数据视图提供一个可靠的数据层
- 提供动态数据能力包括容错和负载均衡
- 保证数据能力和处理能力同步扩展

# 分布式数据管理



# 数据网格使用场景



## 缓存

应用从数据网格请求数据，而不是后台数据源



## 分析

应用从数据网格运行查询和场景建模



## 事务

数据网格作为一个记录事物系统，存放数据和业务逻辑



## 事件

基于事件自动的处理

# 案例：FedEx

## 提高在线跟踪性能

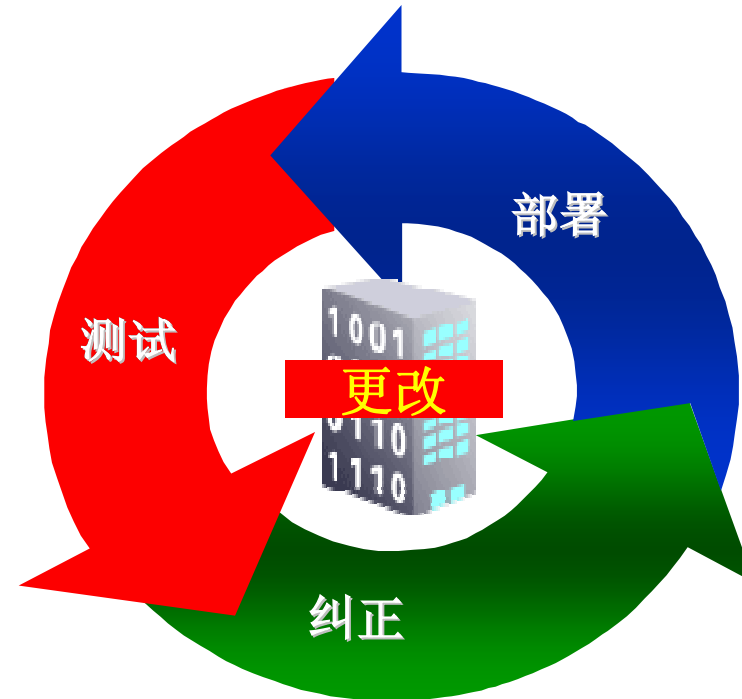


- 公司介绍
- 领先的运输、电子商务和商业服务供应商
- 运营4个业务：FedEx Express, FedEx Ground, FedEx Freight, 和 FedEx Kinko's
- 行业：货物空运服务
- 挑战
- 扩展在线包裹跟踪系统，处理增长的运送量和每个包裹的跟踪数据
- 由于系统性能，客户满意度下降了24%
- 管理和季节性相关的运输成本
- 解决方案
- Oracle Coherence
- Oracle数据库
- 结果
- 部署了**Oracle Coherence**在中间层的内存中虚拟化数据，使包裹的状态信息可以在线立即得到
- 平均减少了**68%**的页面刷新时间
- 在**6个月**提高了**38%**的客户满意度
- 通过扩展在线服务选项减少了客户服务成本和提高了客户满意度
- 通过为数据网格动态增加低成本服务器应对季节性业务高峰

ORACLE

# 真正应用测试

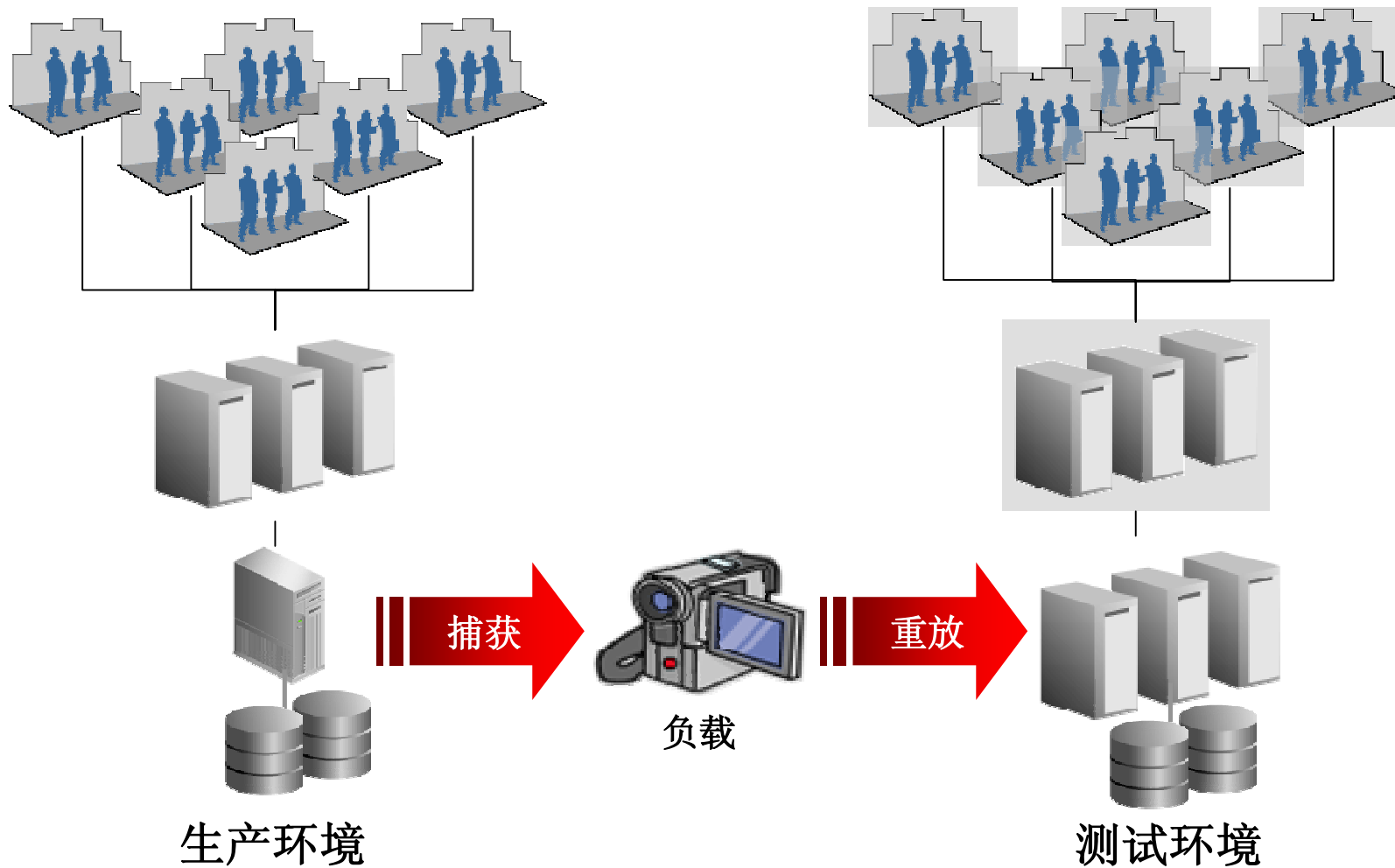
- 价值
  - 技术的快速采用
  - 更高的测试质量
- 业务受益
  - 成本更低
  - 风险更低
- 特性
  - 数据库重放
  - SQL Performance Analyzer (SPA)



通过高级测试实现业务灵活性

# 数据库重放

## 测试生产级的系统更改



ORACLE®



# 使用数据库重放的真正应用测试

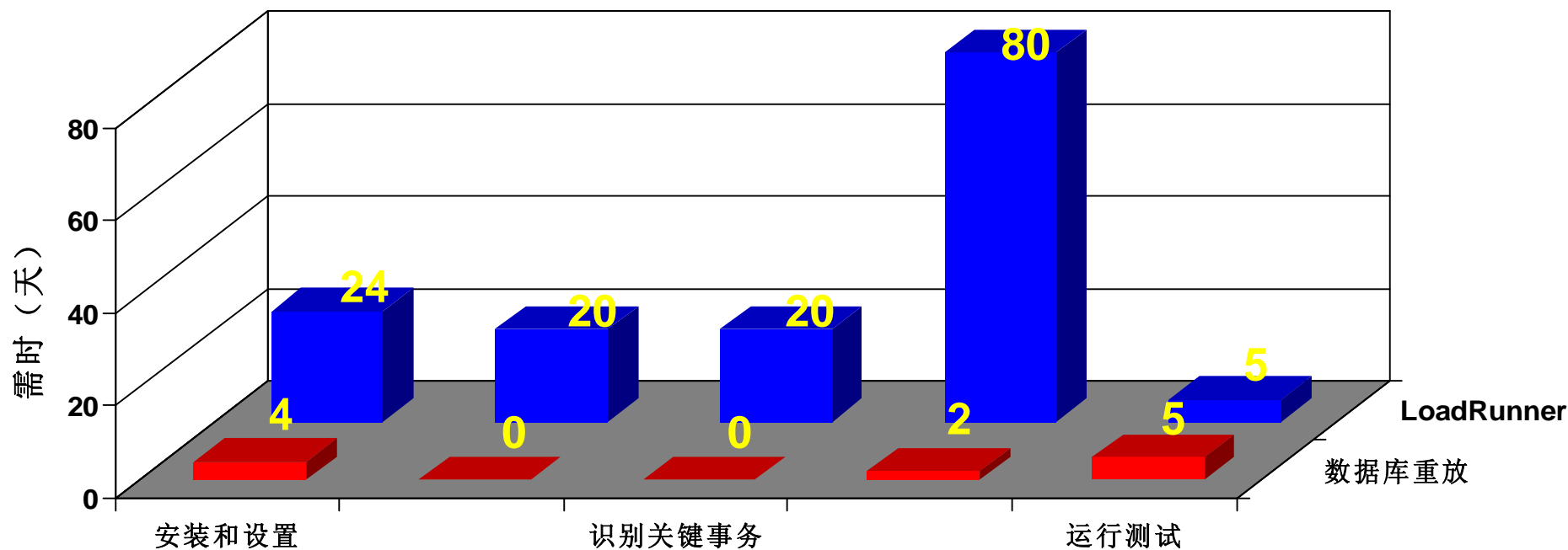
- 在测试环境中重放生产负载
- 在对生产环境进行更改前识别、分析和修复潜在的不稳定因素
- 在生产环境中捕获负载
  - 使用实际负载、时限和并行特性捕获完整生产负载
  - 将捕获的负载移到测试系统
- 在测试环境中重放负载
  - 在测试系统中做出需要的更改
  - 使用完整生产特性重放负载
  - 执行提交排序
- 分析和报告
  - 错误
  - 数据差异
  - 性能差异



分析和报告

# LoadRunner 与数据库重放对比

## 电子商务套件测试



总测试时间

数据库重放: 1/2 个月

LoadRunner: 7 1/2 个月

ORACLE

# 数据库重放对比

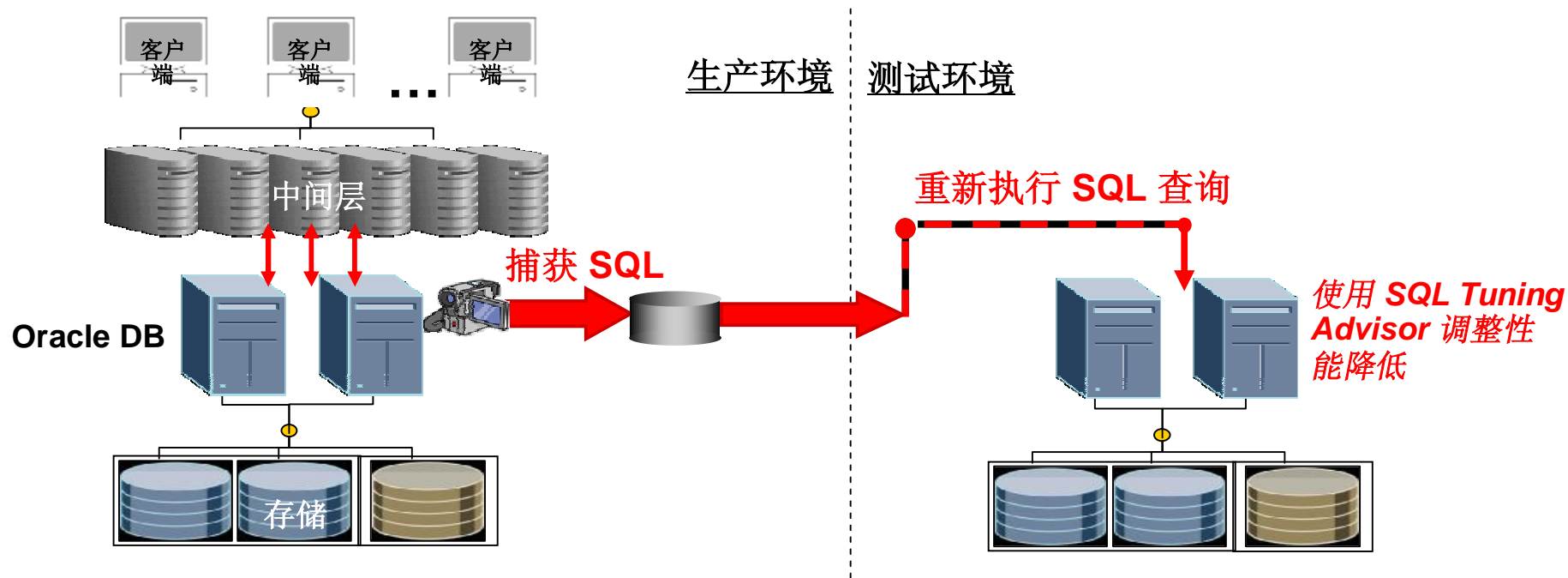
 只有 **Oracle** 才能构建的技术

	第三方 负载测试工具	Oracle 数据库重放
负载	人工模拟负载	实际应用程序负载
测试范围	1-5% 的工作流	100% 的工作流
测试工作量	<ul style="list-style-type: none"><li>• 简单应用程序：数周</li><li>• 复杂应用程序：数月</li></ul>	<ul style="list-style-type: none"><li>• 简单应用程序：数天</li><li>• 复杂应用程序：数天</li></ul>

ORACLE®

# SQL Performance Analyzer

- 测试更改对 SQL 查询性能的影响
- 在生产环境中捕获 SQL 负载（包括统计信息和绑定变量）
- 在测试环境中重新执行 SQL 查询
- 分析性能更改 — 提升和降低



ORACLE

# SPA 报告

## SQL Performance Analyzer Task Result: SYS.SYSTEMCHANGES1

Task Name **SYSTEMCHANGES1**

Task Owner **SYS**

Task Description

SQL Tuning Set Name [HR\\_WORKLOAD](#)

STS Owner **APPS**

Total SQL Statements **50**

SQL Statements With Errors [0](#)

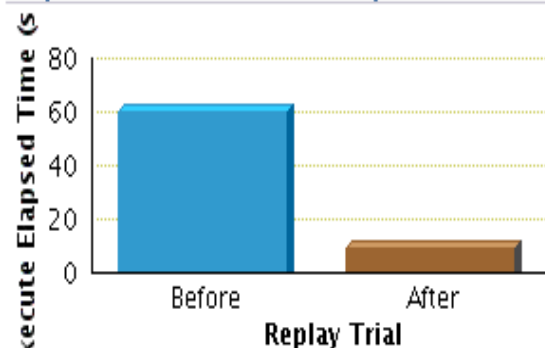
Replay Trial 1 **Before**

Replay Trial 2 **After**

Comparison Metric **Execute Elapsed Time**

### Global Statistics

#### Projected Workload Execute Elapsed Time

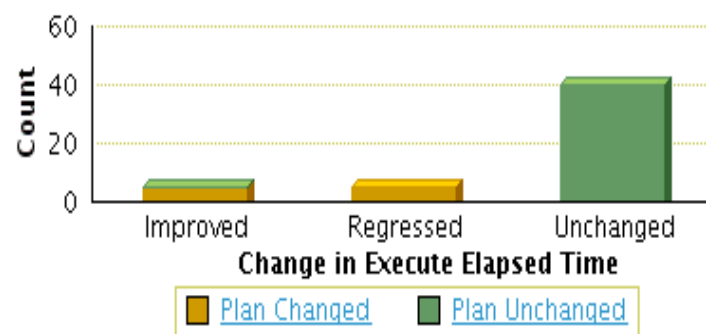


Improvement Impact [89%](#) ↑

Regression Impact [-5%](#) ↓

Overall Impact [85%](#) ↑

#### SQL Statement Count



#### Recommendations

Run SQL Tuning Advisor to tune regressed SQL statements.

[Schedule SQL Tuning Advisor](#)

# SQL Performance Analyzer 优点

从：

手动创建负载

综合负载

耗时数月的手动分析

部分负载

高风险

到：

自动的负载捕获

生产负载

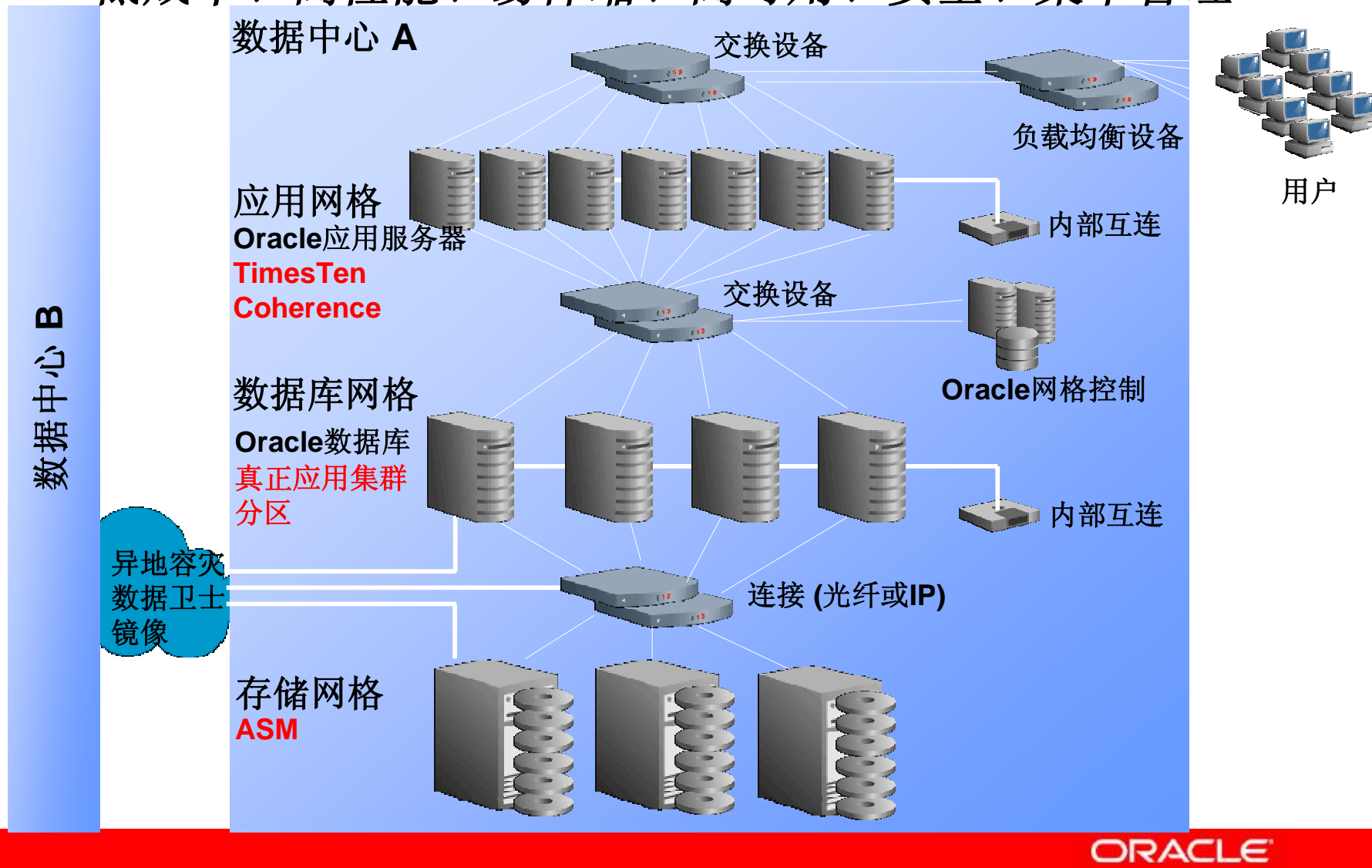
耗时数分钟的自动分析

完整负载

低风险

# 最为开放 完整和领先的架构 Grid+SOA

低成本、高性能、易伸缩、高可用、安全、集中管理





**ORACLE IS THE INFORMATION COMPANY**