

# GBase8t 高可用解决方案

南大通用 GBase  
孙国荣

[sunguorong@gbase.cn](mailto:sunguorong@gbase.cn)

OLTP→OLAP

SMP→MPP Scale Up → Scale Out

SQL→NoSQL→NewSQL

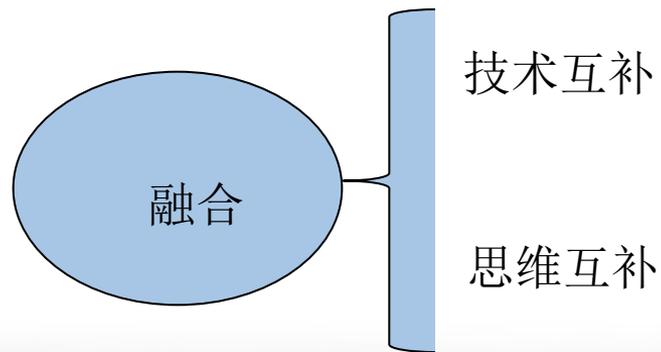
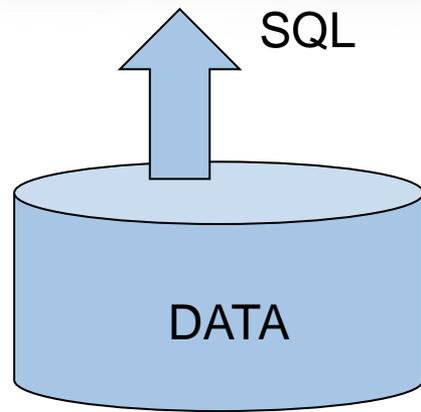
行业客户→互联网+

--历史

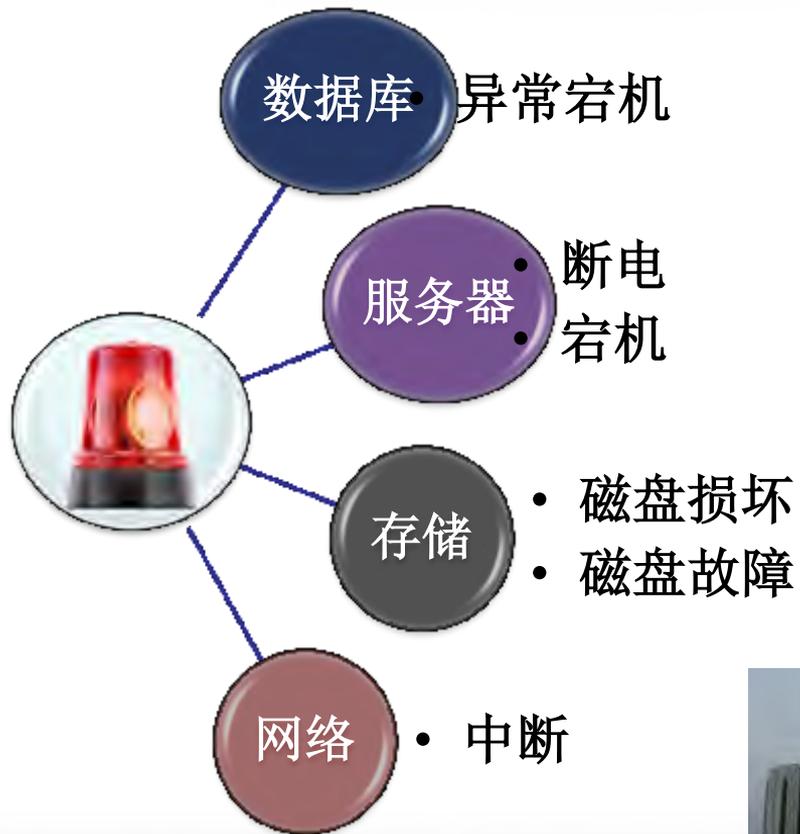
--技术投入

--责任

--不仅是技术



当前信息化系统的高可用性已经成为系统建设的重要环节。目前核心业务系统主要是基于存储建立容灾方案。GBase 8t提供完整的数据库级高可用解决方案，提供秒级故障自动切换功能，使核心系统的可靠性达到了99.9999%。



7 \* 24小时不间断服务



火灾



地震



海啸



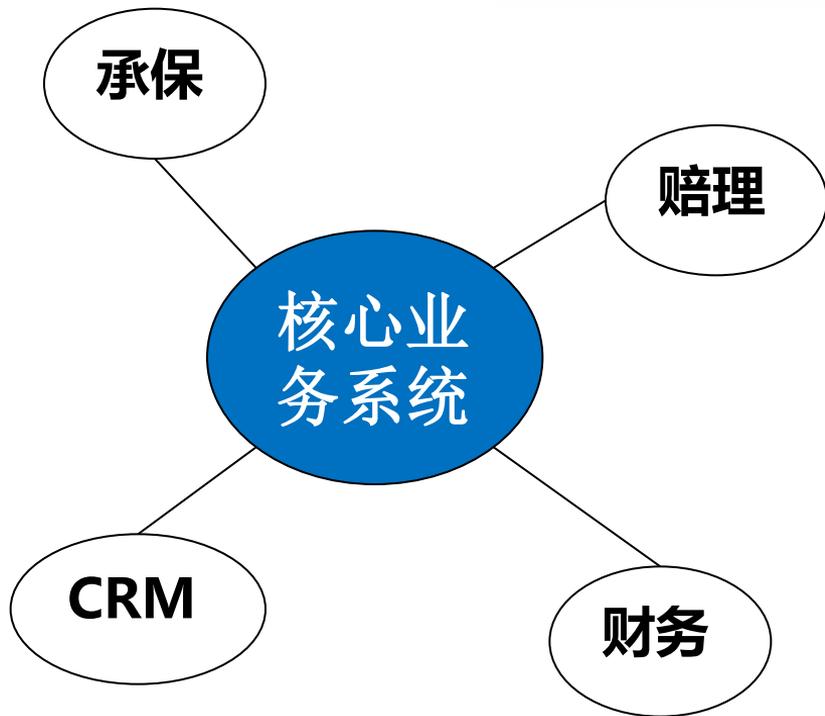
# ➤ 大型企业数据库高可用案例介绍

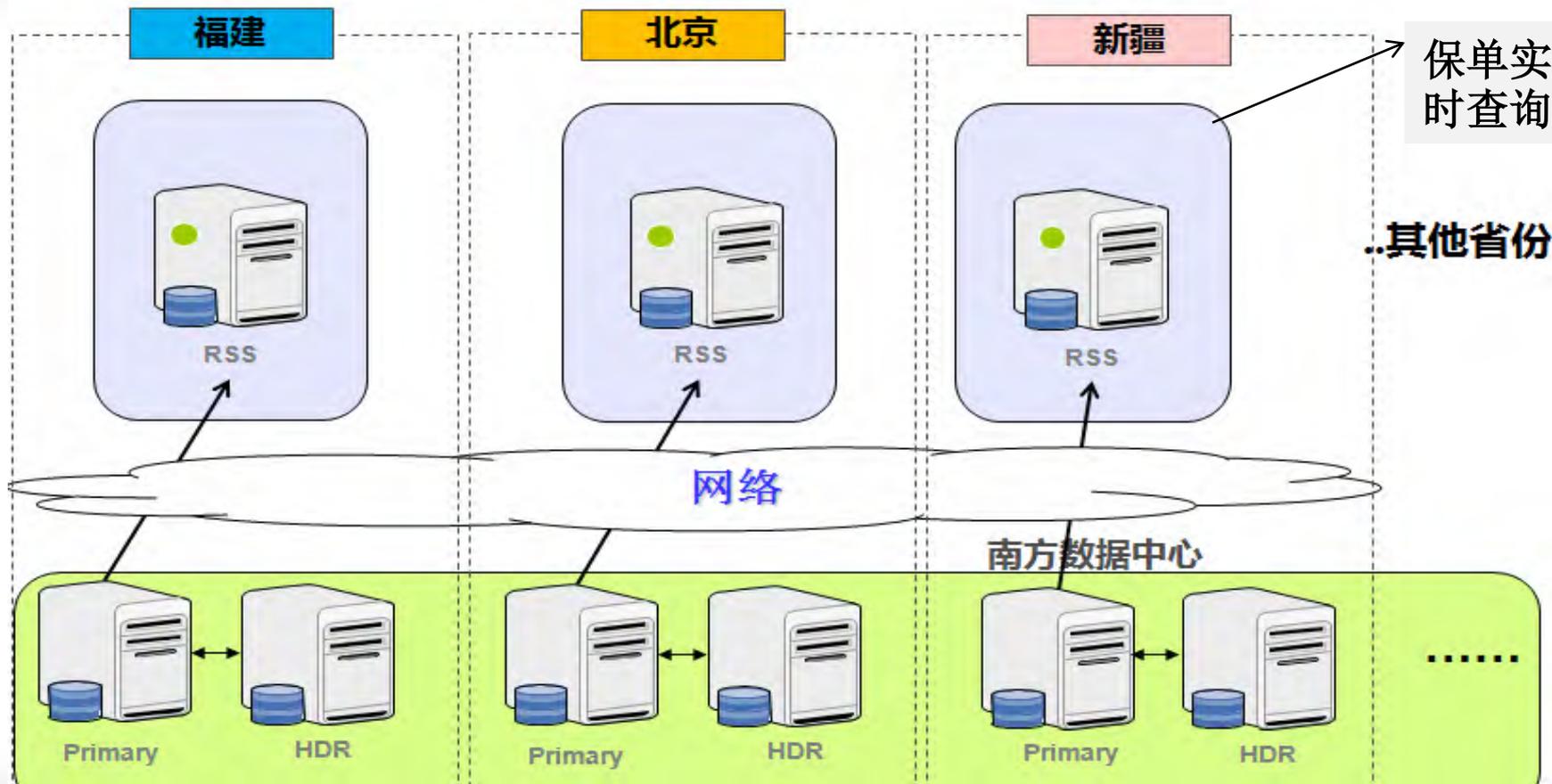
2012年10月份 某国有保险企业完成了全国数据库大集中工作，把36个省级分公司的业务数据库物理集中到南方数据中心。

### 建设目标：

1. 高可用方案
2. 异地容灾
3. 利旧机器
4. 负载分担

- 涉及系统总数  
40个系统
- 数据库总数  
约100个数据库
- 数据总量  
数据总量约100TB ( 全国 )





**超远距离** — 3000公里异地容灾

**低成本** — 无需采购任何第三方产品、低带宽、易维护

**负载均衡** — HDR承担报表分析业务、RSS承担保单查询服务

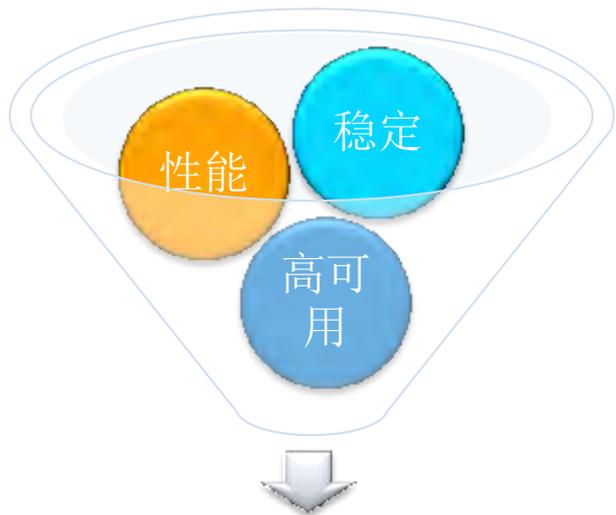
# ➤ GBase8t 高可用方案介绍

2014年9月份南大通用购买了IBM Informix12.1源代码授权

南大通用基于Informix12.1 源代码打造自主商标GBase 8t (Transaction)



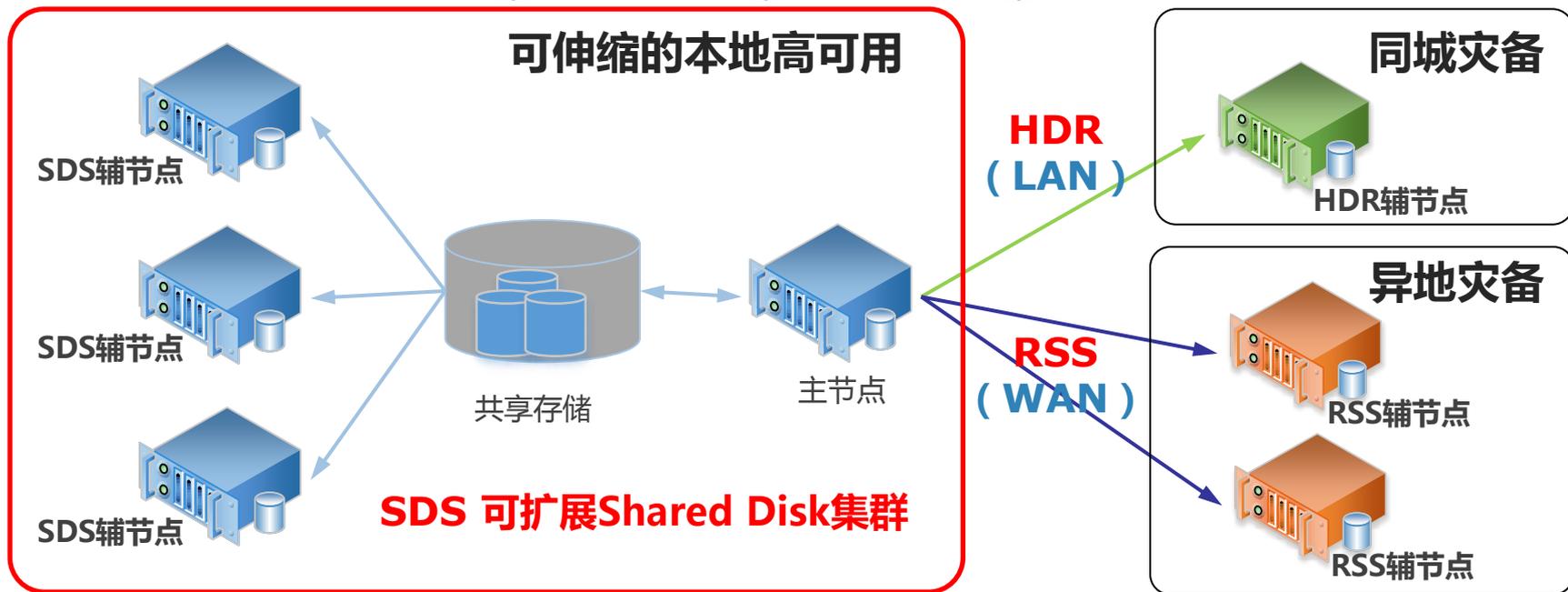
## 联机事务处理系统(OLTP)



OLTP系统基本需求

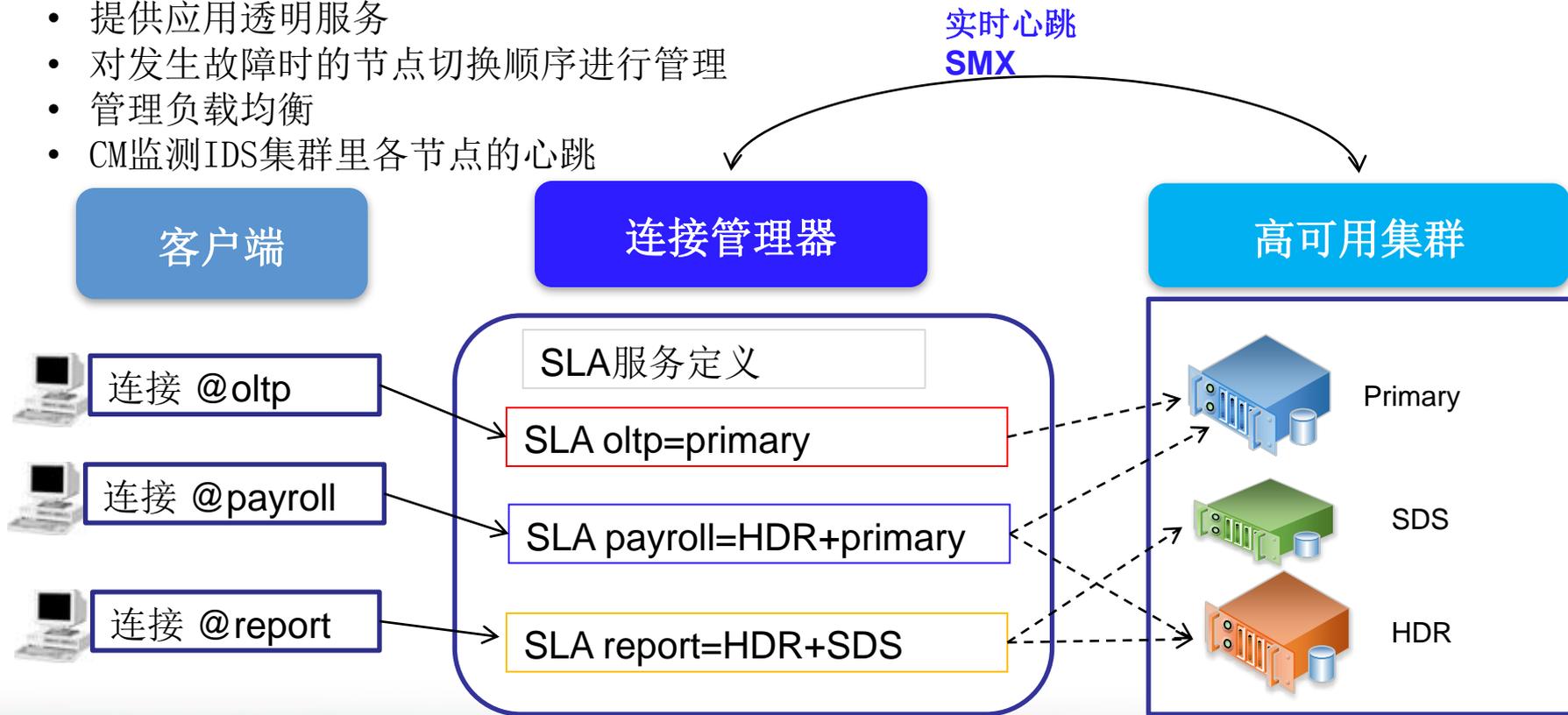


## 连接管理器



	1-HDR	2-RSS	3-SDS
网络连接	光纤连接	互联网或专线	光纤连接
节点间的距离	百公里内， 同一个机房	数千公里	数米内， 同一个机房
对带宽的要求	高	低	高
辅节点的最大数目	1个	多个	多个
存储设备	独立存储设备	独立存储设备	共享存储设备
承受服务器的软硬件故障	是	是	是
承受天灾	否	是	否

- 基于服务层协议（SLA）的连接管理
- 提供应用透明服务
- 对发生故障时的节点切换顺序进行管理
- 管理负载均衡
- CM监测IDS集群里各节点的心跳



## ▪ SQLHOSTS.app

```
OLTP group - - i=12,c=1
oltp_all1 onsoctcp hp-mach3-2 servbcm1 g=OLTP
oltp_all2 onsoctcp hp-mach6-2 servbcm3 g=OLTP

REPORT group - - i=13,c=1
oltp_read1 onsoctcp hp-mach3-2 servbcm2 g=REPORT
oltp_read2 onsoctcp hp-mach6-2 servbcm4 g=REPORT
```

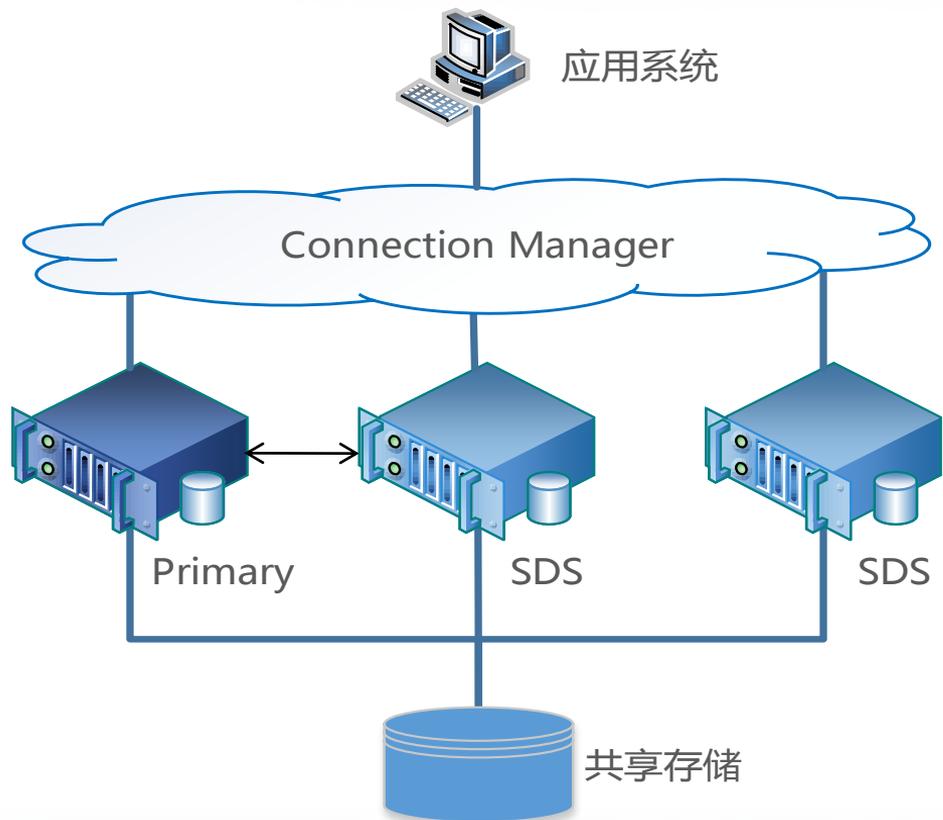
## ▪ Java 程序

**java dmls** 可更新

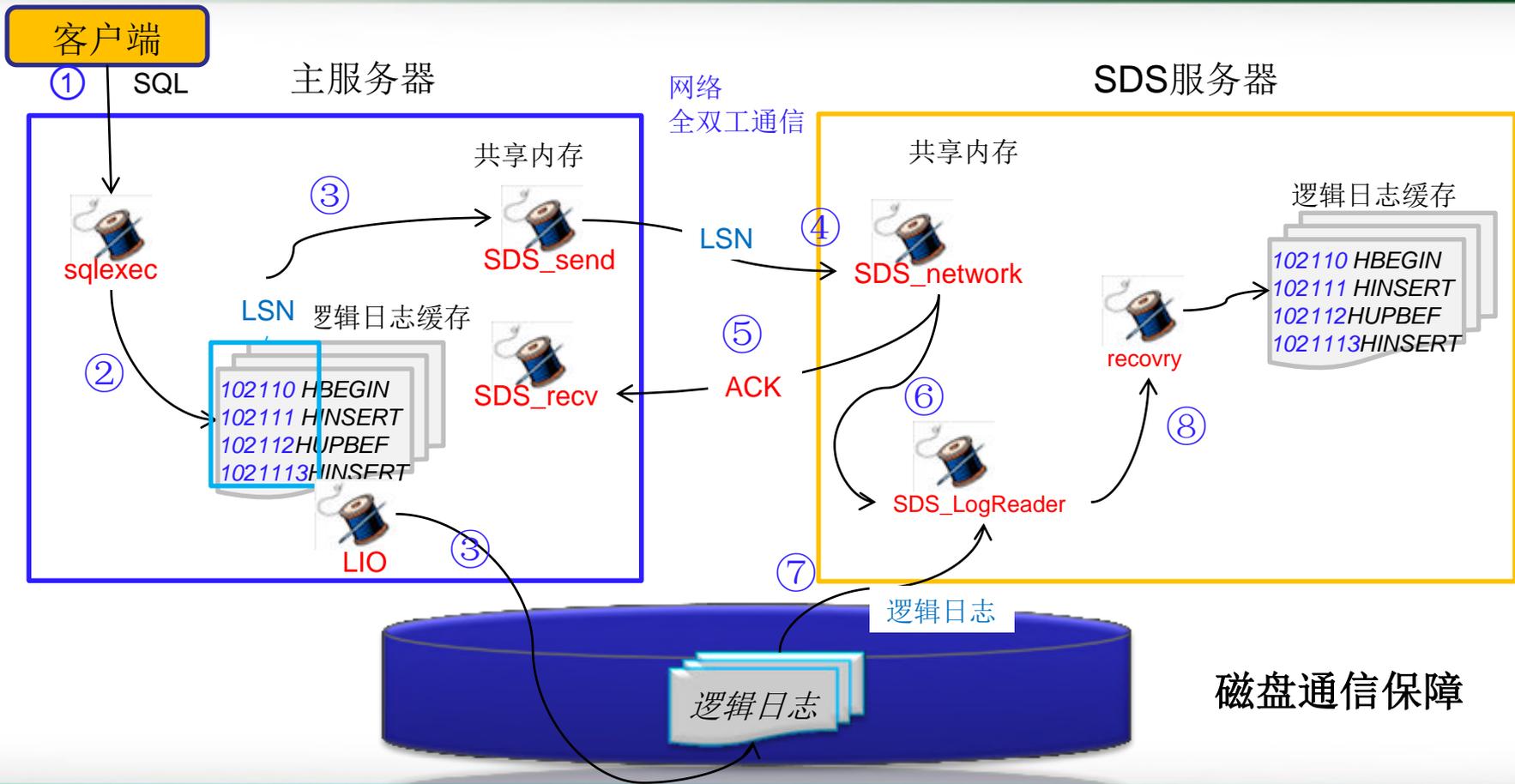
```
URL_STRING ="jdbc:informix-sqli:/demodb:" +
"INFORMIXSERVER=OLTP;SQLH_TYPE=FILE;SQLH_FILE=/work/etc/SQLHOSTS.app;";
```

**Java read-only**应用

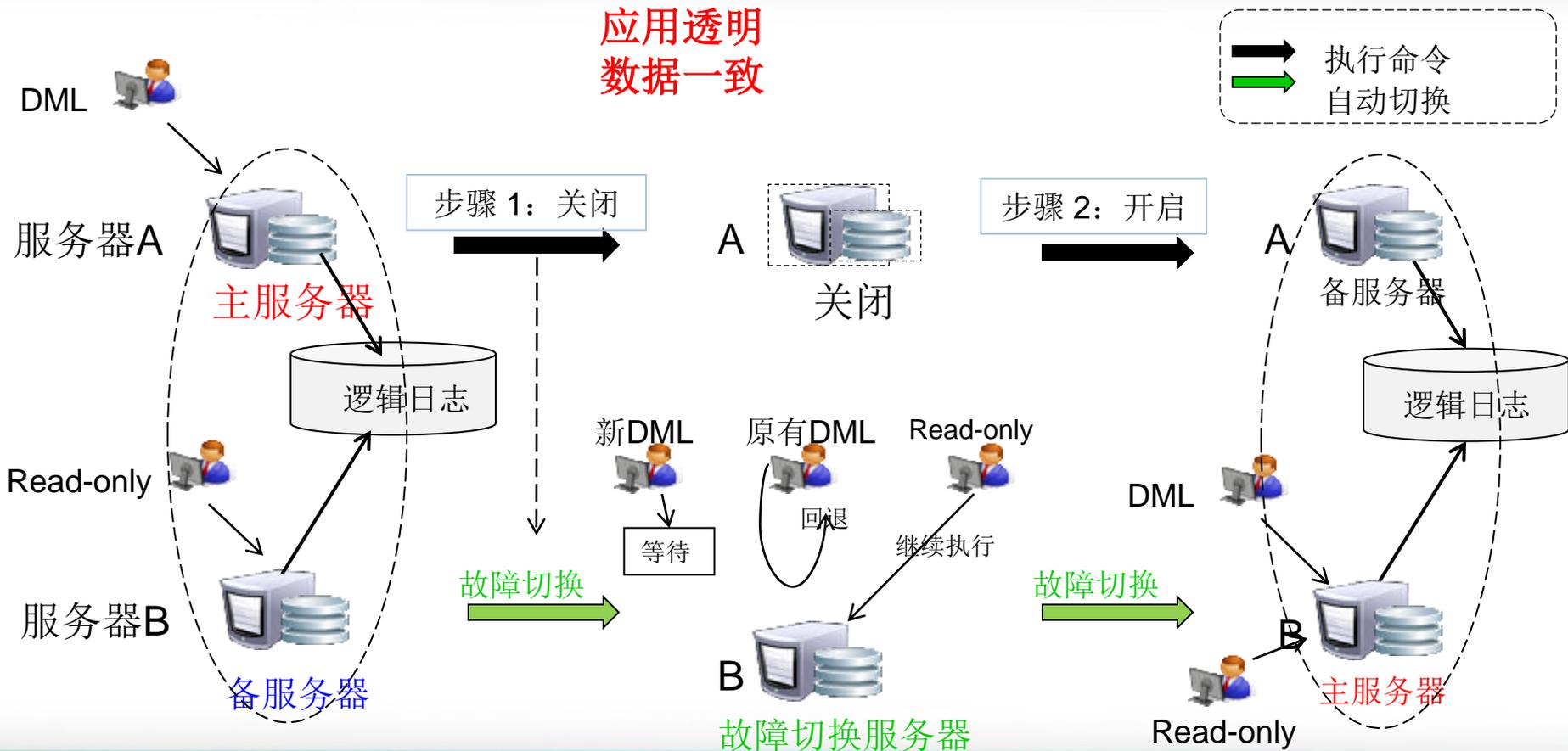
```
URL_STRING ="jdbc:informix-sqli:/demodb:" +
"INFORMIXSERVER=REPORT;SQLH_TYPE=FILE;SQLH_FILE=/work/etc/SQLHOSTS.app;";
```



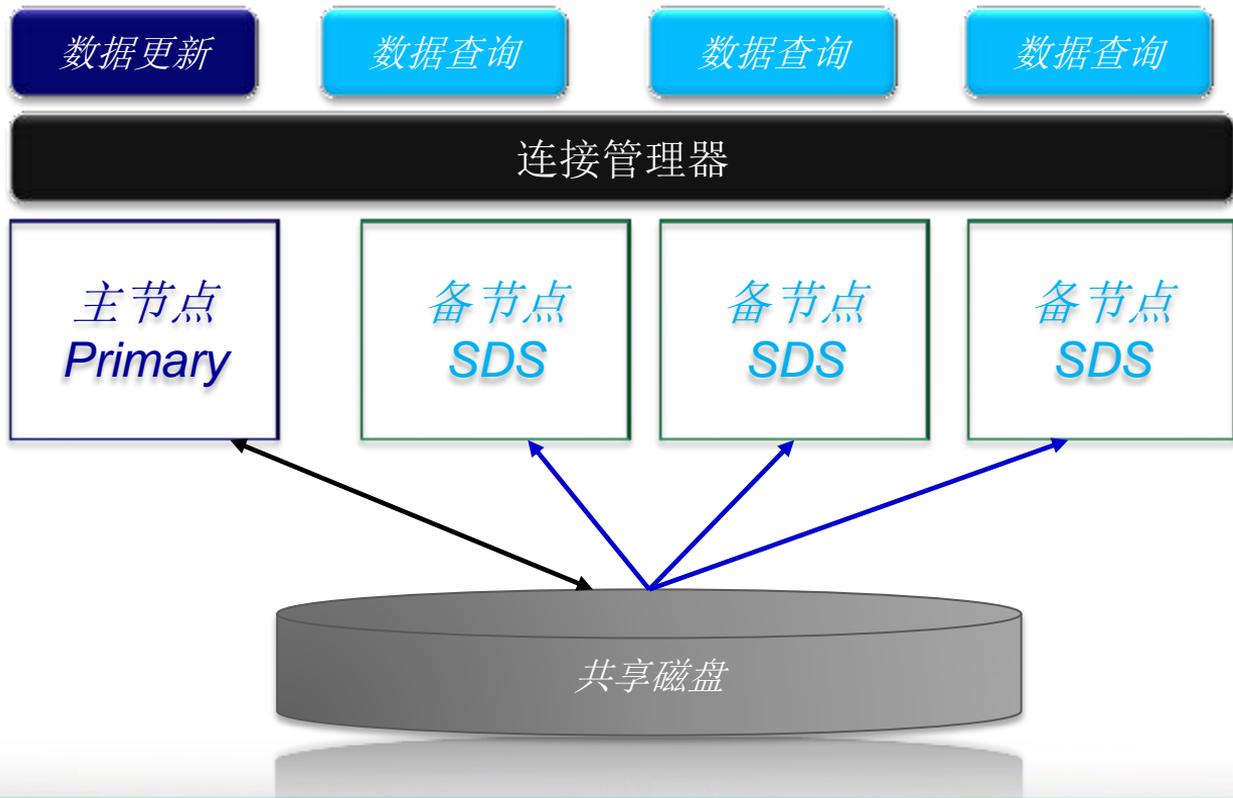
- 基于共享存储的高可用架构，类似于 Oracle RAC技术
- 由CM进行管理并自动维护
- 主节点（Primary）和辅节点（SDS）均可读写
- 主节点故障时，辅节点自动接管服务（升级为主节点）
- 故障切换时间在 30秒以内（故障判断+故障切换）
- 对应用透明



# 故障切换转换过程(Failover Transition Process, FTP)



## 数万并发访问



### 业务需求

- 读写分离
- 大量并发 读业务

### 典型业务场景

- 保险—保单查询
- 政务—征信查询
- 银行—个人信用

### 方案特点

- 高可用
- 多节点负载分担
- 读写分离

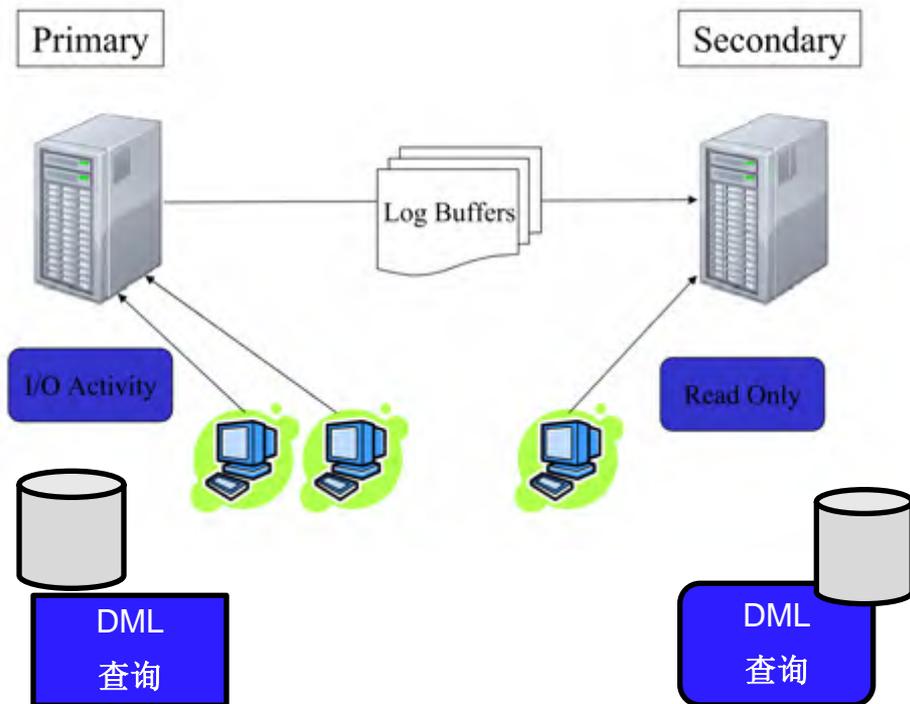


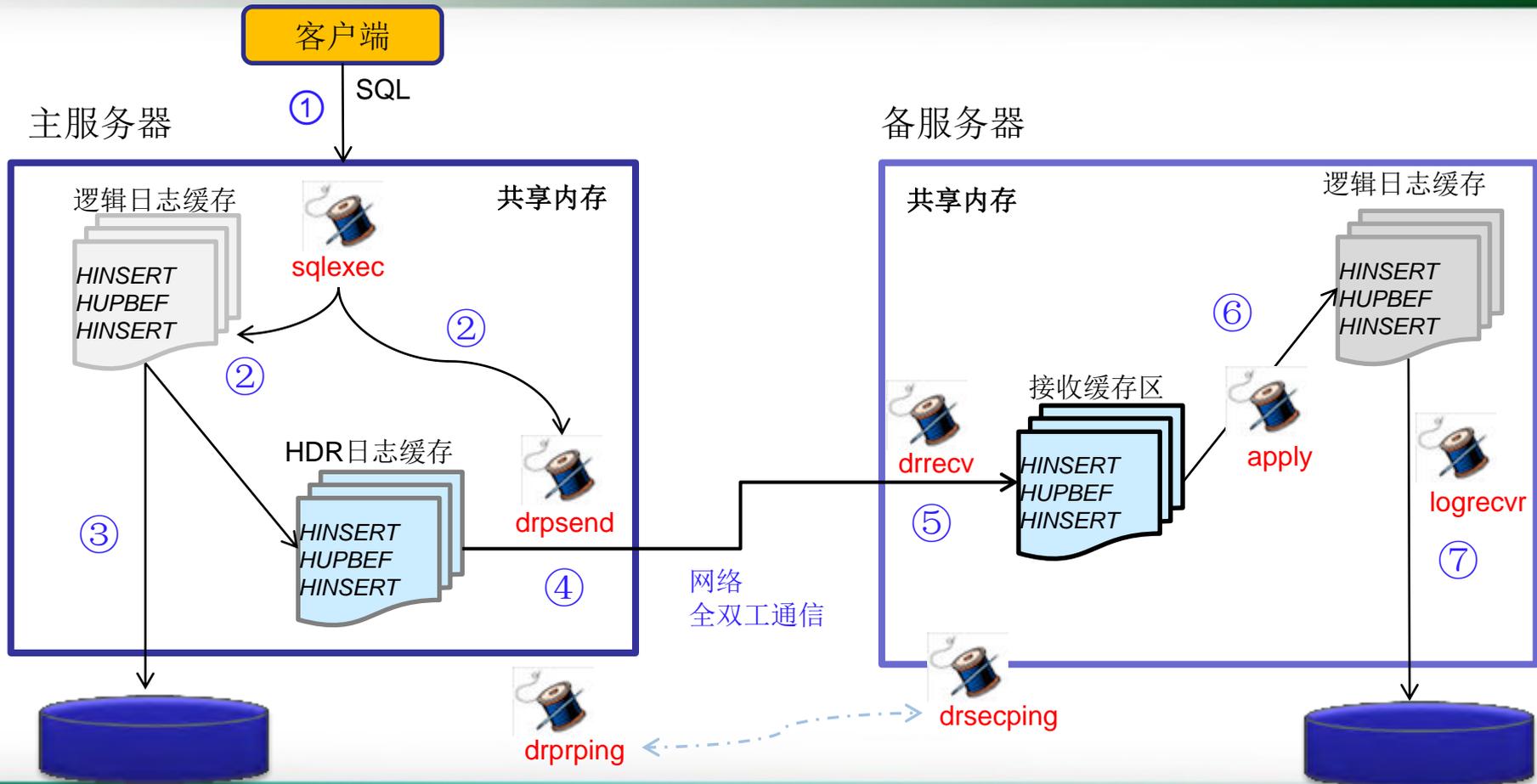
## • HDR 双机实时数据同步热备份方案

- 实例级别的复制
- 基于逻辑日志的复制技术
- 通过网络持续将数据实时复制到备节
- 备机可读写，可实现读写分离

## • 相对HA、存储复制方案优点

- 热备份、可读写、负载分离
- 故障更快切换
- 无数据丢失风险







- 需求：
- 同城2机房
- 数据实时同步，热备份
- 已有方案：存储、第三方数据复制

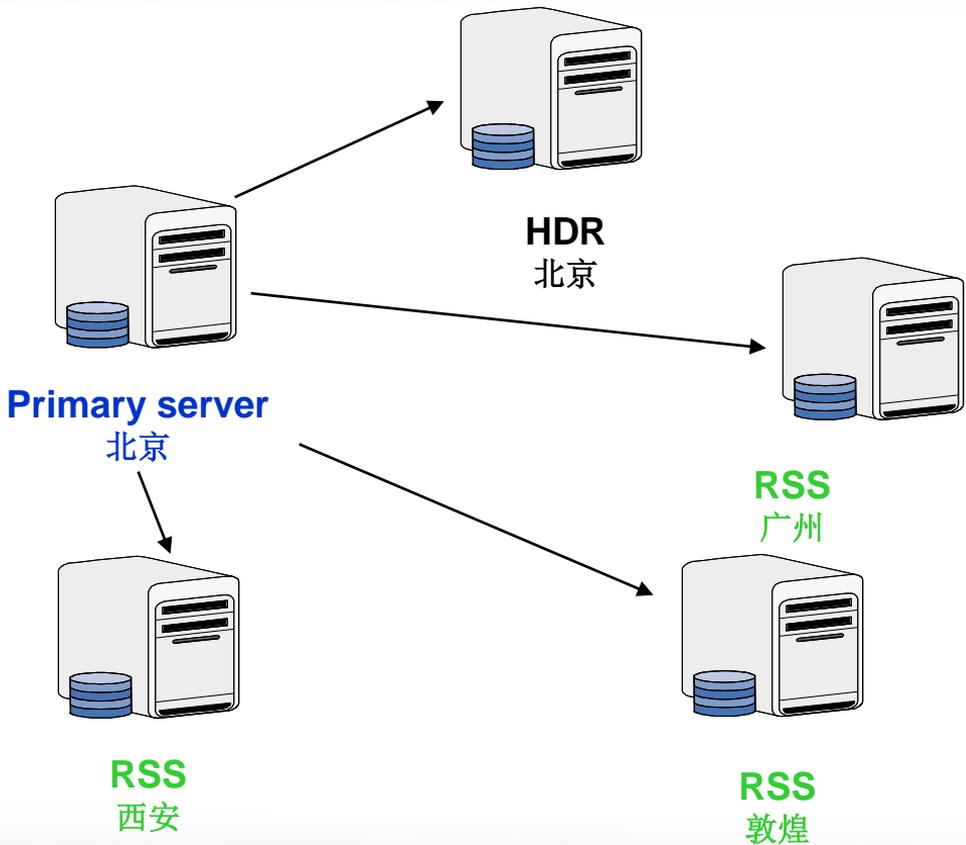
数据实时同步

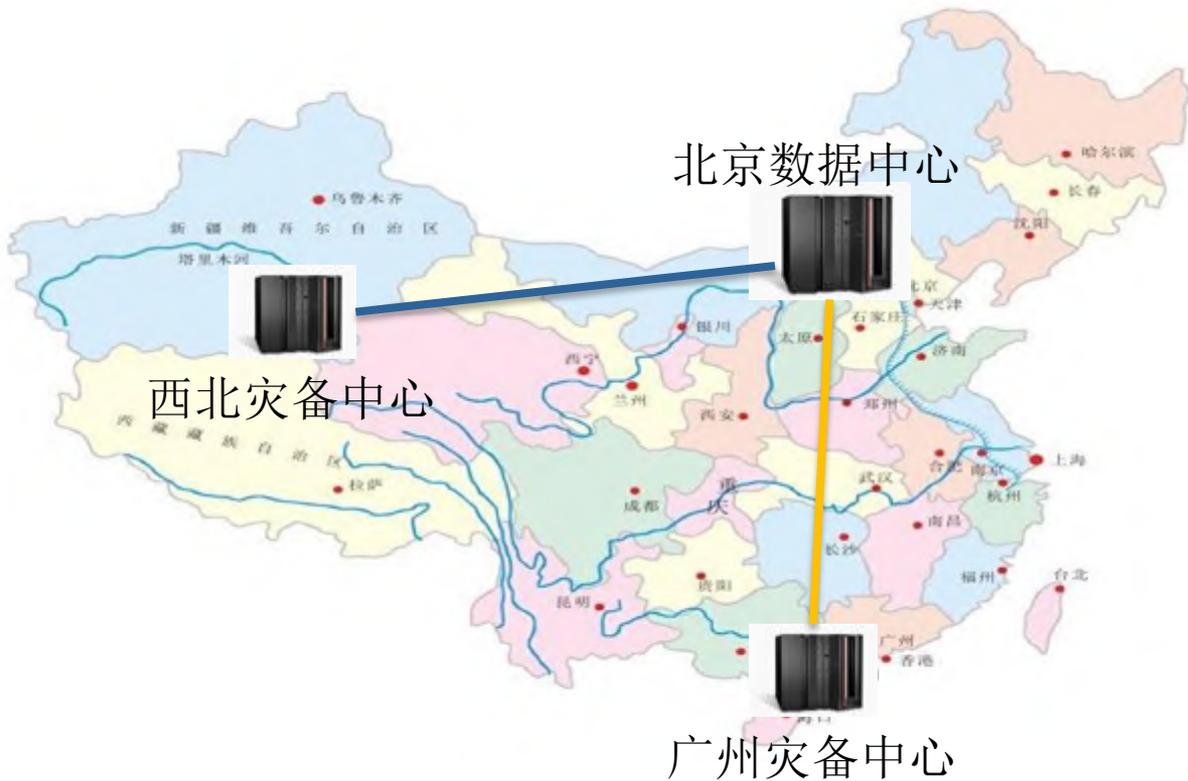
## GBase 8t HDR:

- 实时同步—支持同步、异步模式
- 可实现零数据丢失风险
- 双活备机可提供查询服务
- 百公里距离
- 故障自动快速切换



- 多个热备份
  - 灵活扩展多个RSS节点
  - HDR 的延伸和增强
- 低速网络环境下的可用性保障
- 维护一份完整的数据拷贝
- RSS建立方式
  - 备份/恢复
  - 存储拷贝
- 可用于
  - 建立远程灾备
  - 报表处理



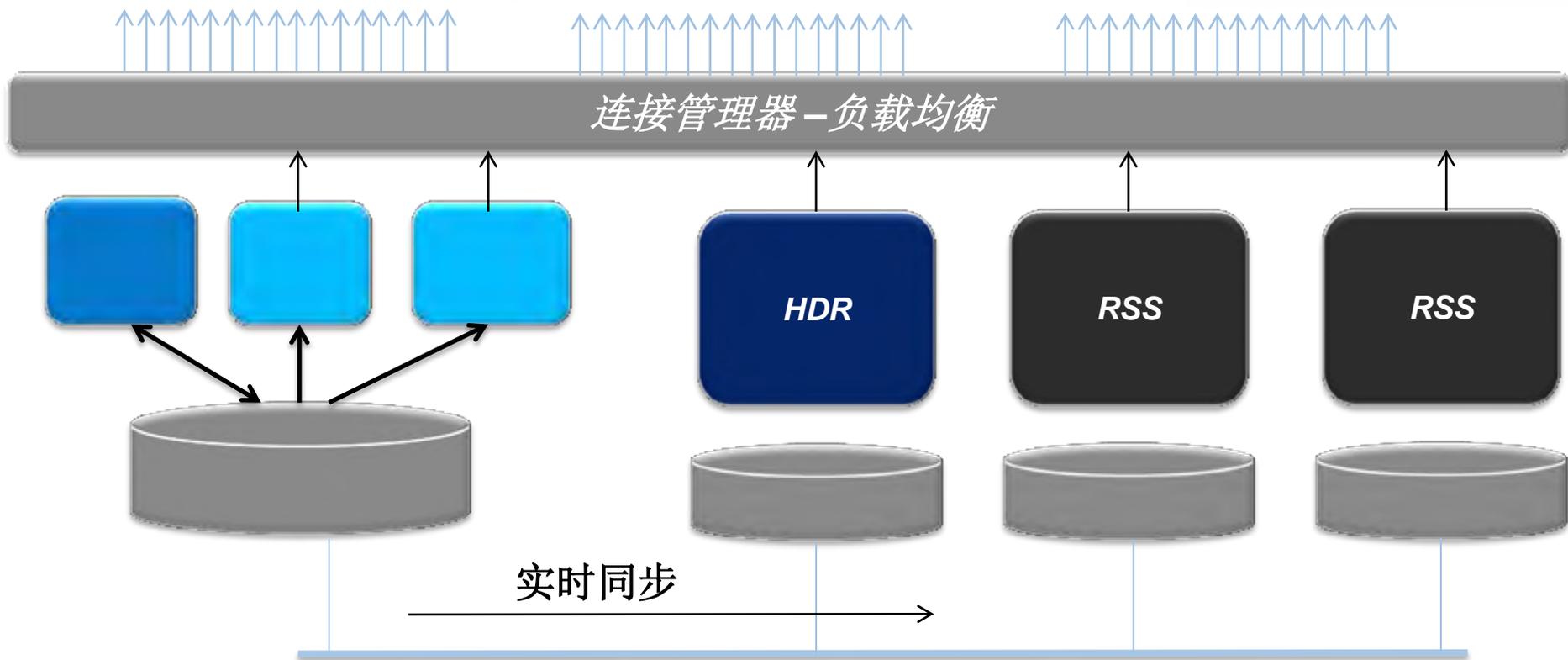


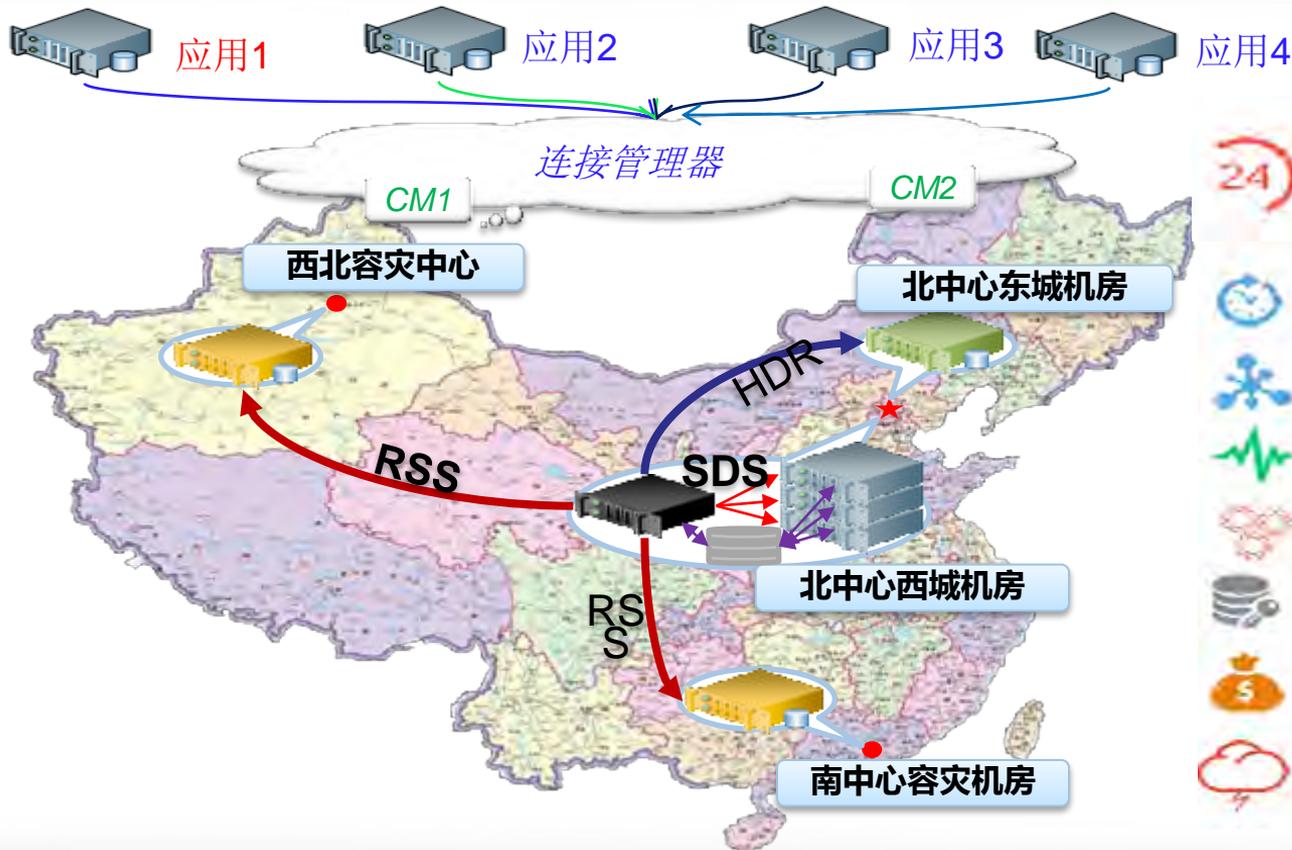
数千公里距离  
网络带宽低  
实时数据同步



**GBase 8t RSS**

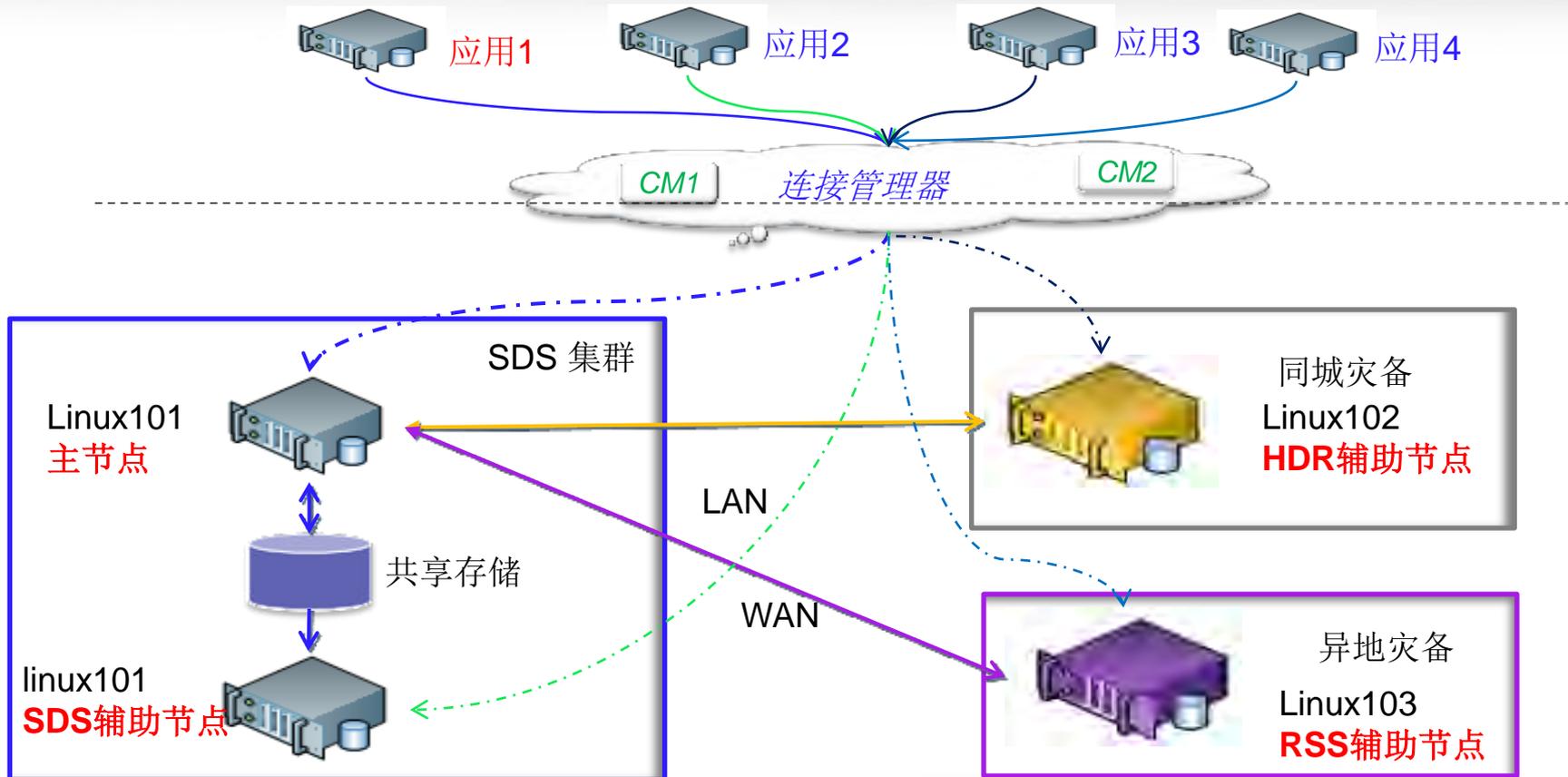
- 超3000多公里案例
- 低带宽
- 实时低延迟
- 备机可读服务



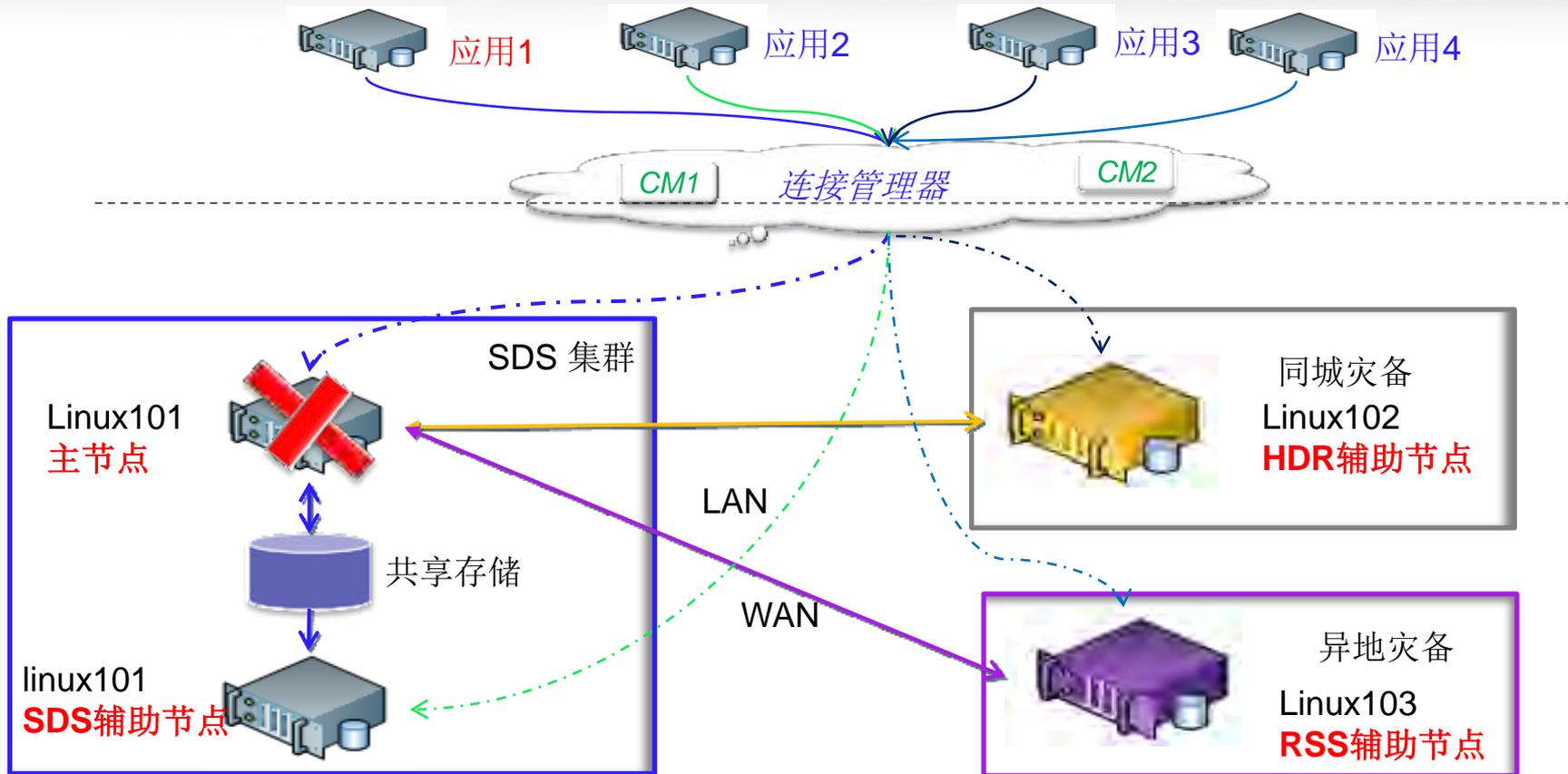


-  99.999%高可用  
提供 7\*24 小时业务保障
-  故障秒级快速自动切换
-  应用透明连接
-  读写分离、负载均衡
-  搭建快速、管理简单
-  数据实时热备份
-  低成本、无需特殊硬件
-  应对不可抗逆灾害

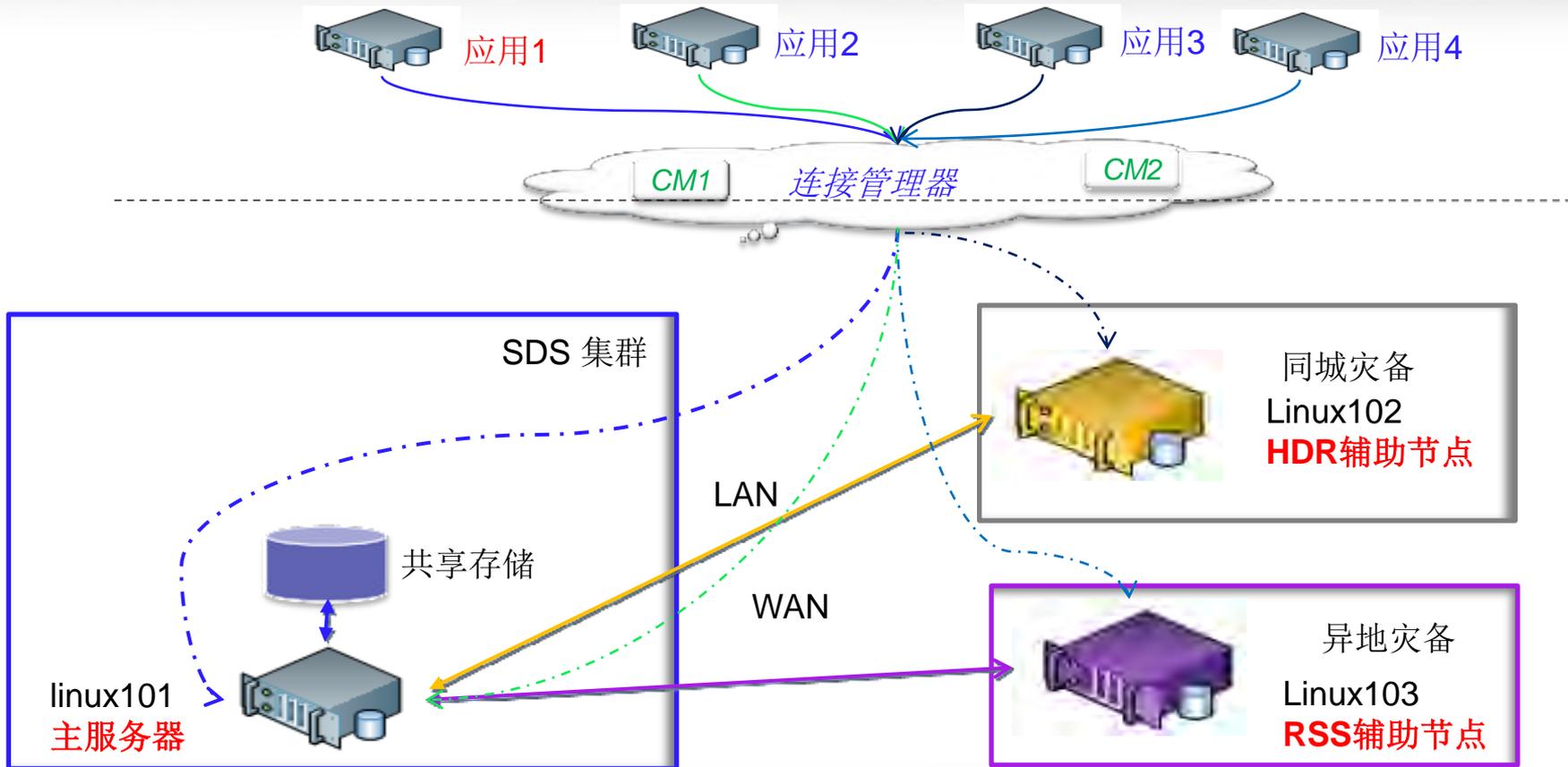
# 高可用组件切换步骤 – 依靠CM自动切换



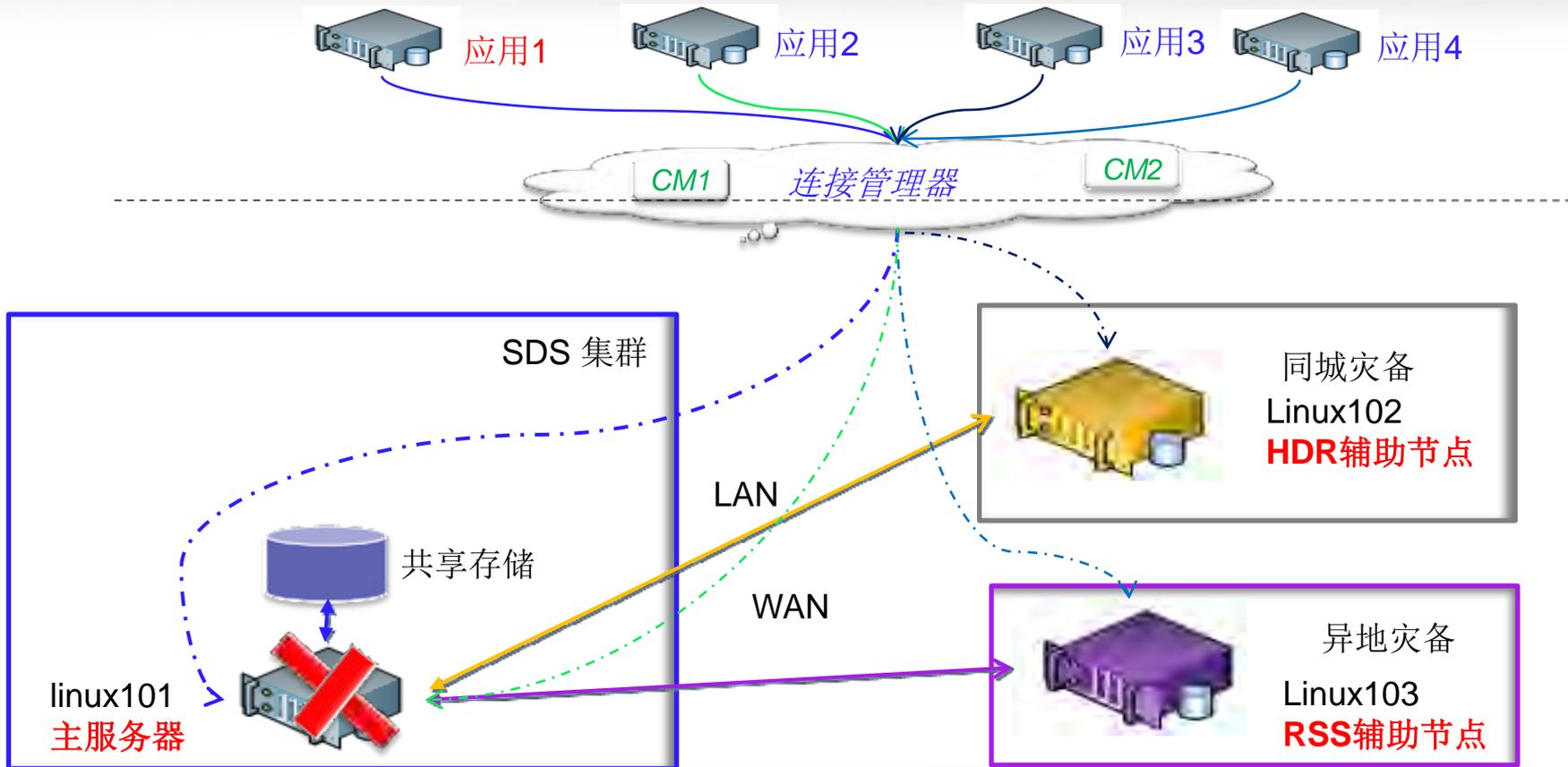
# 高可用组件切换步骤 – 依靠CM自动切换

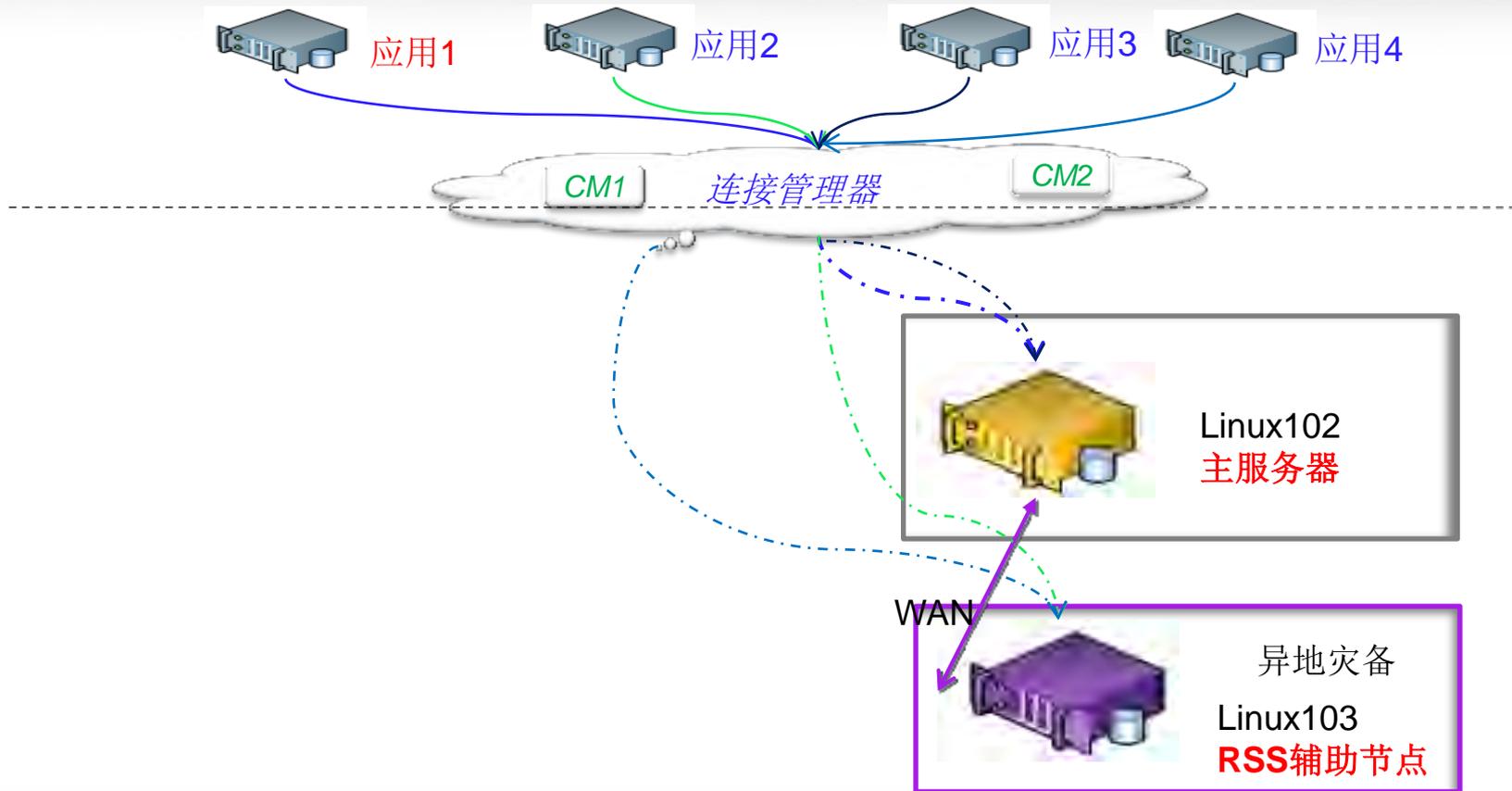


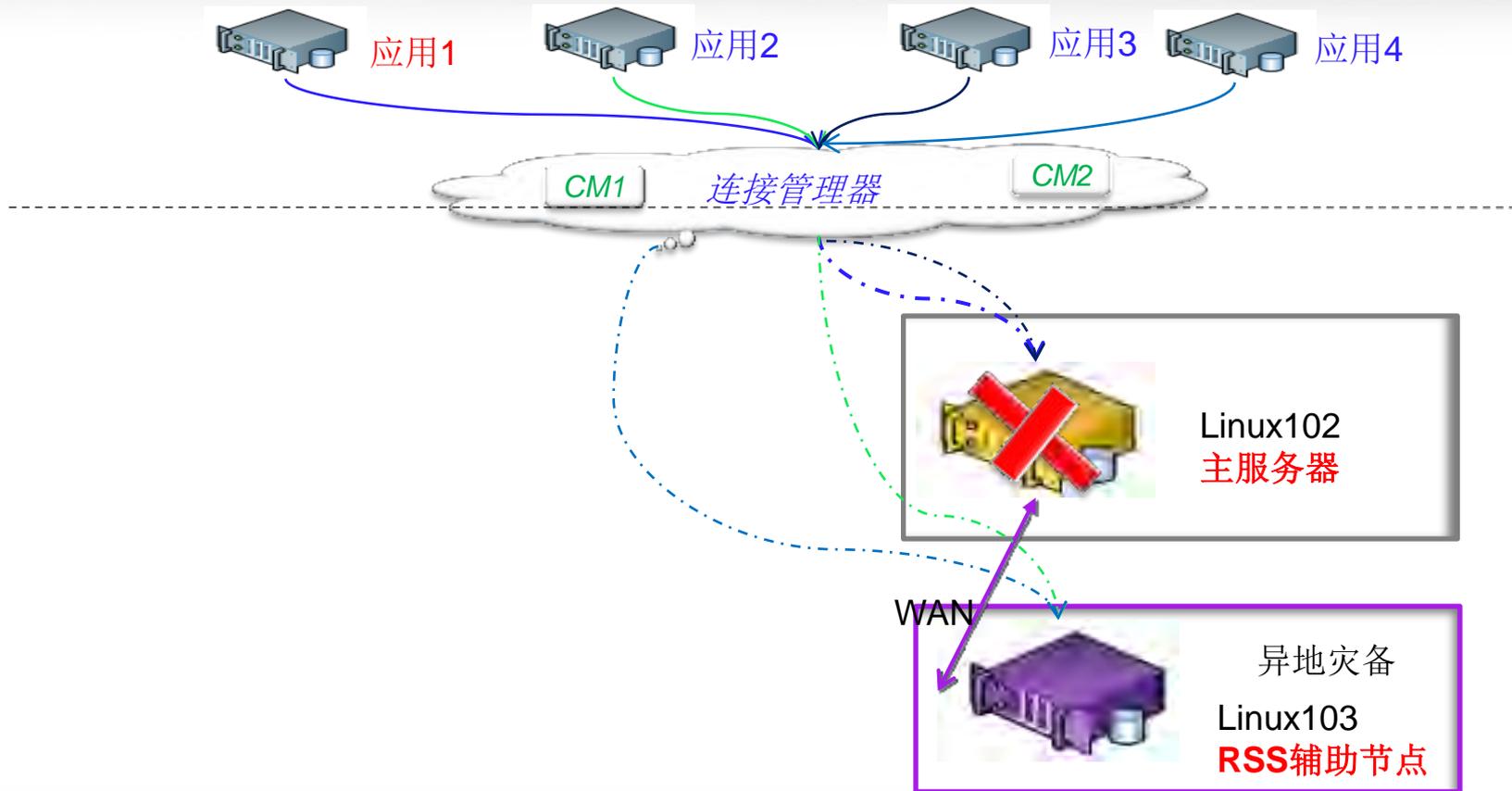
# 高可用组件切换步骤 – 依靠CM自动切换

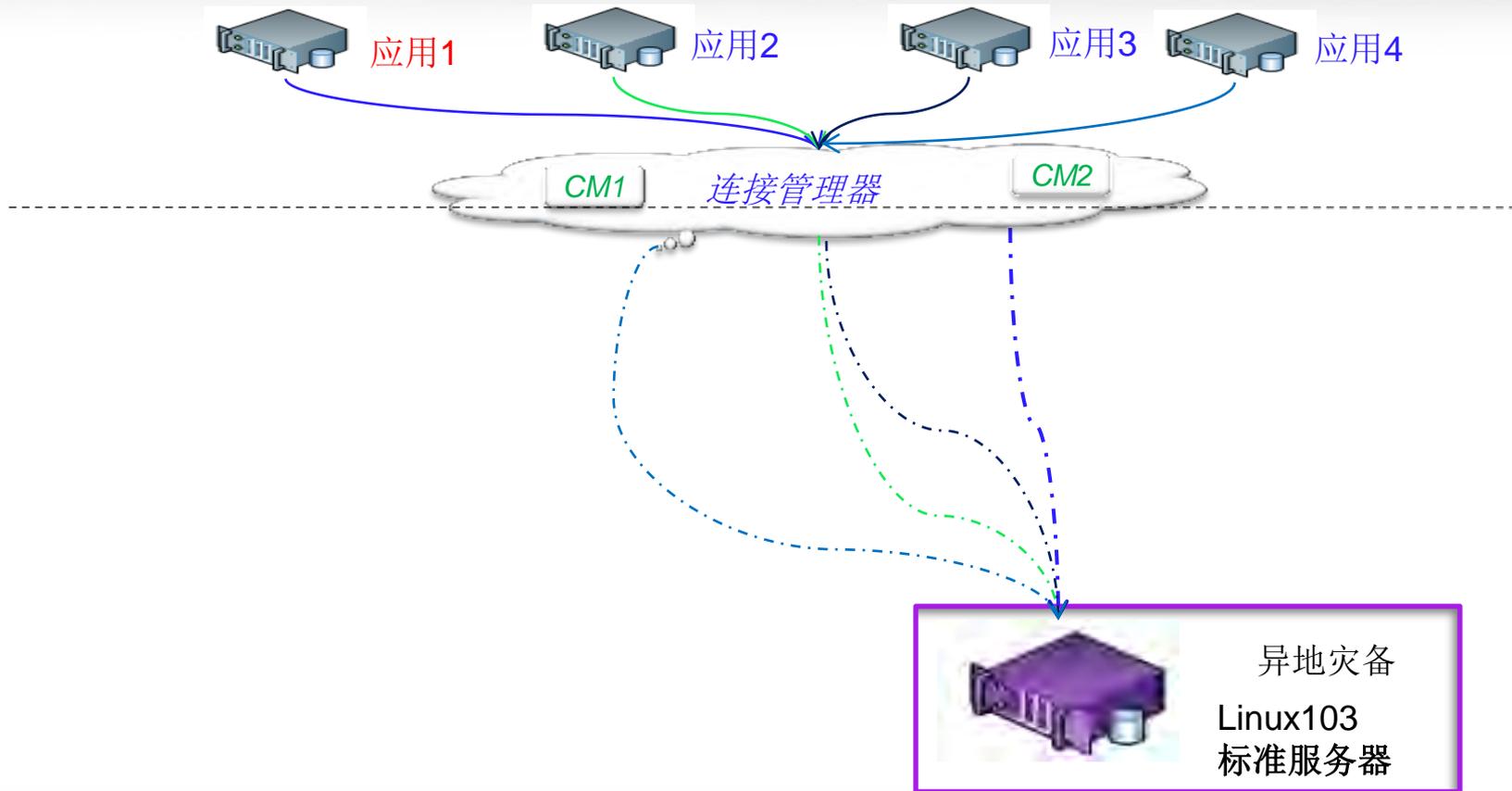


# 高可用组件切换步骤 – 依靠CM自动切换









## ➤ GBase8t 高可用方案优势

➤数据库级别容灾方案比存储级别容灾在RTO和RPO上面有绝对优势。

	同城容灾		异地容灾	
	RTO	RPO	RTO	RPO
GBase8t 数据库容灾	60秒以内	0	120秒以内	接近0
存储容灾	1200秒以上	接近0	1200秒以上	非0

**RTO:** Recovery Time Objective

**RPO:** Recovery Point Objective

## ➤ 可用性

GBase8t 的高可用方案（HDR+RSS）可以保证软件，硬件，环境的灾难，使得系统的可用性有了质的提升，可达到**99.9999%**。

## ➤ 多活—负载分担

GBase8t 的高可用方案多节点均online—“**多活**”节点，可以同时支撑业务，可实现**多副本读写分离**，实现业务**负载均衡**。

## ➤ 稳定性

GBase8t 的高可用方案（HDR+RSS）是**成熟的、可以落地的**数据库级容灾解决方案。在复杂外部环境中，依然可以稳定高效的工作，是核心系统稳定运行的有力保障。

## ➤ 高性能

GBase8t 的高可用方案在海量数据情况下，依然可以达到和单机类似的性能。**读写分离**场景下，**线程扩展**节点。

## ➤ 低成本

SDS, HDR, RSS功能组件完全集成在企业版本中，**不需要额外付费**和**其它硬件资源**。

## ➤ 易维护

故障**秒级切换**，应用**自动切换**。通过**CM**组件实现发生，把数据库切换到相应容灾节点，节约了大量人工操作。



## 物联网时代 感应器 时序数据

智慧城市、物联网、能源和公用事业、工业4.0

医疗、物流、生产安全、交通、智慧楼宇、智慧家庭、环境监控





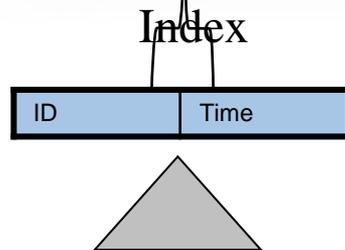
- 典型业务场景
  - 数据质量检测
  - 异常数据检测
  - 连续多个时间点前后时间点数据计算

## 关系型数据库模型

- 记录按插入顺序排放
- 通过索引进行查询加速

Table  
Grows

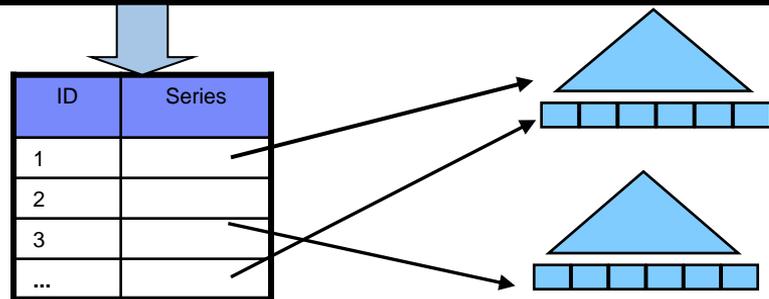
ID	时间点	读值	电压
1	2015-1-11 12:00	Value 1	Value 2
2	2015-1-11 12:00	Value 1	Value 2
...	...	...	...
1	2015-1-11 12:15	Value 1	Value 2
2	2015-1-11 12:15	Value 1	Value 2
...	...	...	...



## TimeSeries数据模型

ID	Time Series
1	[(2015-1-11 12:00, value 1, value 2, ...), (2015-1-11 12:15, value 1, value 2, ...), ...]
2	[(2015-1-11 12:00, value 1, value 2, ...), (2015-1-11 12:15, value 1, value 2, ...), ...]
...	...

Table  
grows



- 采用列式存储，具有相同属相的记录存储在一行
- 同一行中的记录按时间顺序先后存储
- 容器存储结构，内部自动进行索引

## 双数据库管理引擎

时间序列引擎与关系型数据引擎完美结合

内置大量时间分析函数

提供 SQL, JDBC, C-API 编程接口



### 高性能

性能提高 5-20 倍

提供大数据计算分析能力



### 节省空间

时间序列存储管理机制

空间节省 50%

**SDCC**

中国软件开发者大会

SOFTWARE DEVELOPER CONFERENCE CHINA

谢谢

[sdcc.csdn.net](http://sdcc.csdn.net)