



# ArcGIS Server 的性能和可伸缩性

Esri中国（北京）有限公司 马克玲

# 常见性能相关的问题

- 我的web 应用中, 地图出图为什么慢？
- 最短路径分析速度有点慢，该如何调优？
- 我该如何配置服务的实例数，才能达到性能最优？
- 我要配置多少台GIS服务器，才能满足现有用户规模？
- ArcGIS Server 该如何部署，才能达到性能最优？

# 主要内容

- 案例：Esri性能测试
- 案例：性能诊断
- ArcGIS Server 10.1的部署及可伸缩性

## 案例：Esri性能测试



# Esri性能测试概览

- 测试的动机
- 测试环境
- 测试方法
- 最佳实践
  - 服务配置优化
  - 服务器架构优化
  - 虚拟环境部署推荐方案

# 测试的动机

- 在开发期间进行回归测试
  - 性能
    - 针对各种服务类型，数据类型和位置
    - 架构组成部分—— SOM/SOC/WS handlers
  - 负载下的服务质量
    - 内存泄露
    - 响应延迟（例如，绘制错误）
    - 并发问题（例如，进程挂起— 死锁）
- 发现最佳的服务器架构
  - 可伸缩性
  - 冗余/ 容错



# 测试环境

- 硬件

- Dell PowerEdge M1000E 盘柜
  - 16台 Dell PowerEdge M600 刀片服务器
    - 2, 四核, Intel E5420 Xeon, 2.5GHz CPUs
    - 2\* 150GB, 万转, 3GBps串行连接SCSI, RAID 0
    - 8 GB内存
    - 2\* 1GBps 网卡
  - MD 3000i iSCSI SAN 存储阵列
    - 4\* 400G 万转SCSI硬盘, RAID 5
  - 20G 网络交换机

- 软件

- Visual Studio 2008 Team Test
- SQL Server 2008 + Reporting Services

# 测试方法（概览）

- 我们要度量什么？

## 机器性能矩阵

- CPU(% 利用率)
- Memory(可用的，每进程)
- Disk I/O (% 空闲时间， bytes/sec)
- Network(上传/下载， bytes/sec)

## 服务矩阵

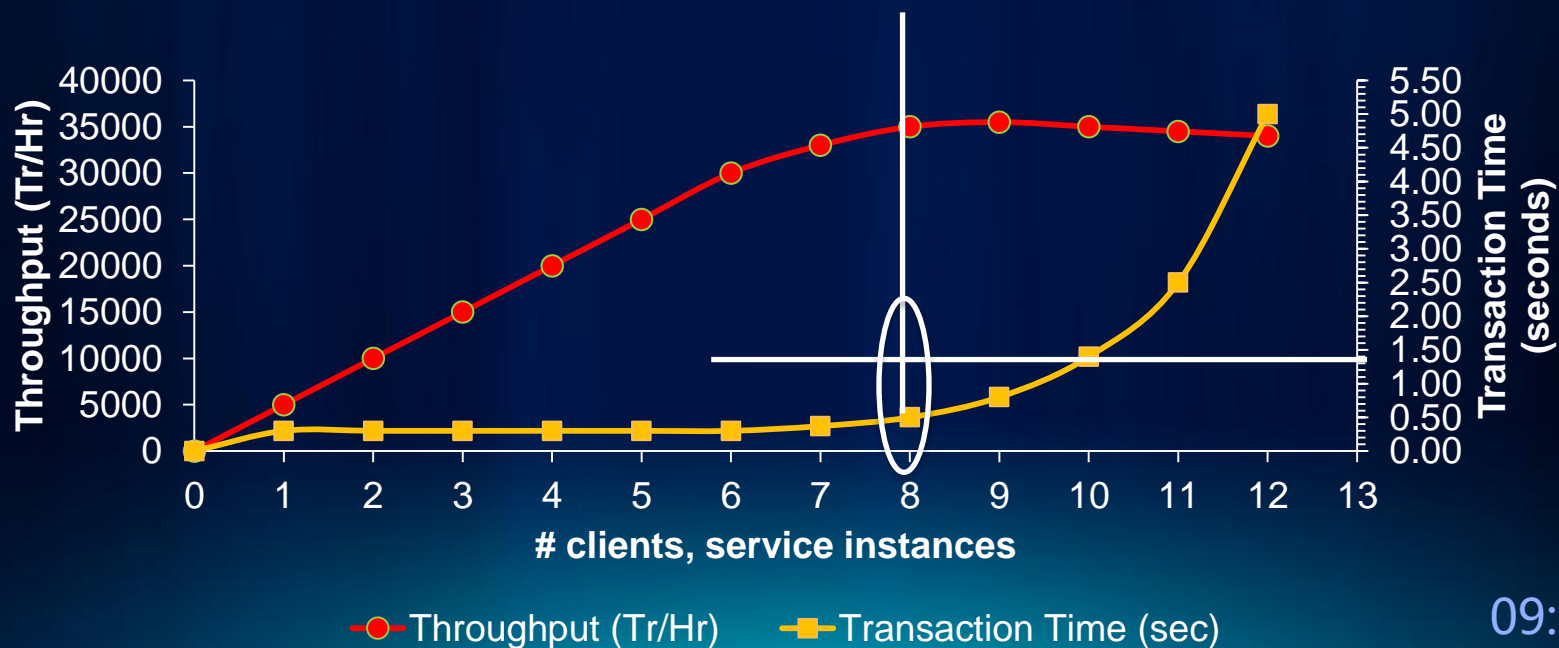
- 每个事务的平均时间
- 吞吐量（事务数/小时）

- 测试类型
  - 压力测试（单步-负载）
  - 疲劳测试（持续-负载）
  - 用户-工作流测试（真实环境）



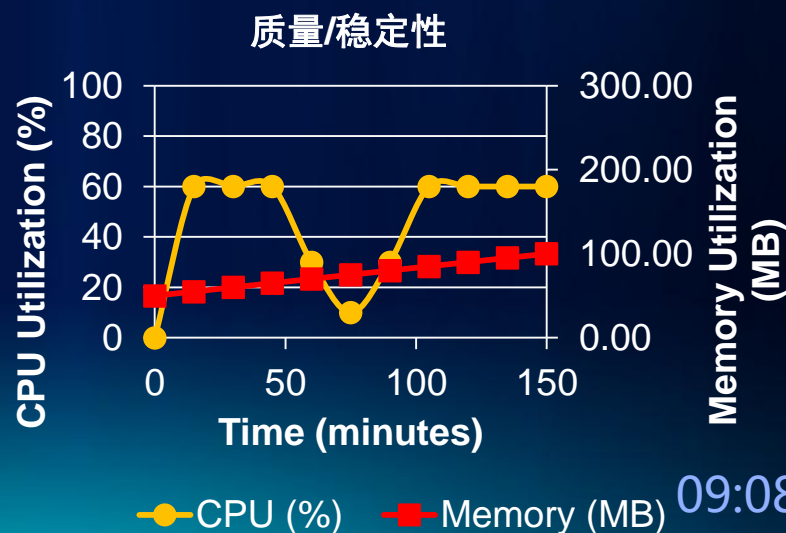
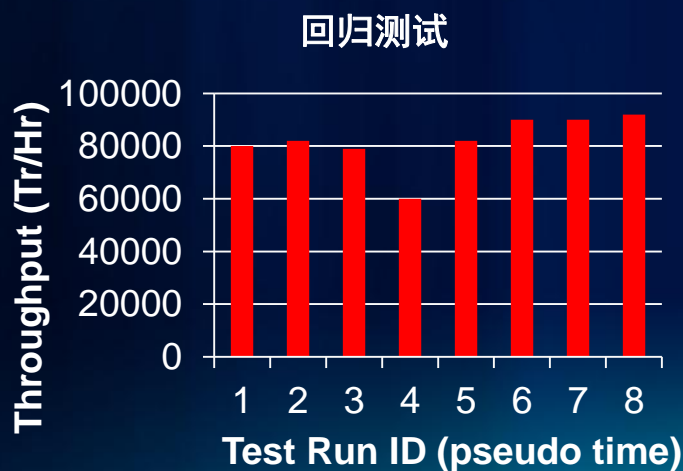
# 测试方法: 压力测试

- 目的：
  - 找到可接受事务处理时间的最大吞吐量
  - 决定每核应该配置的最优服务实例数
- 过程：
  - 单步负载测试（修改客户端增量/服务实例数）
  - 每步运行5分钟，并记录平均吞吐量



# 测试方法：疲劳测试

- 目标：
  - 性能回归测试
  - 服务质量（内存泄露，功能错误）
  - 决定长期的稳定性（死锁）
- 过程：
  - 维持负载在“压力”测试中最大负载量的60%
  - 短时间运行用于回归测试，长时间运行用于质量/稳定性测试

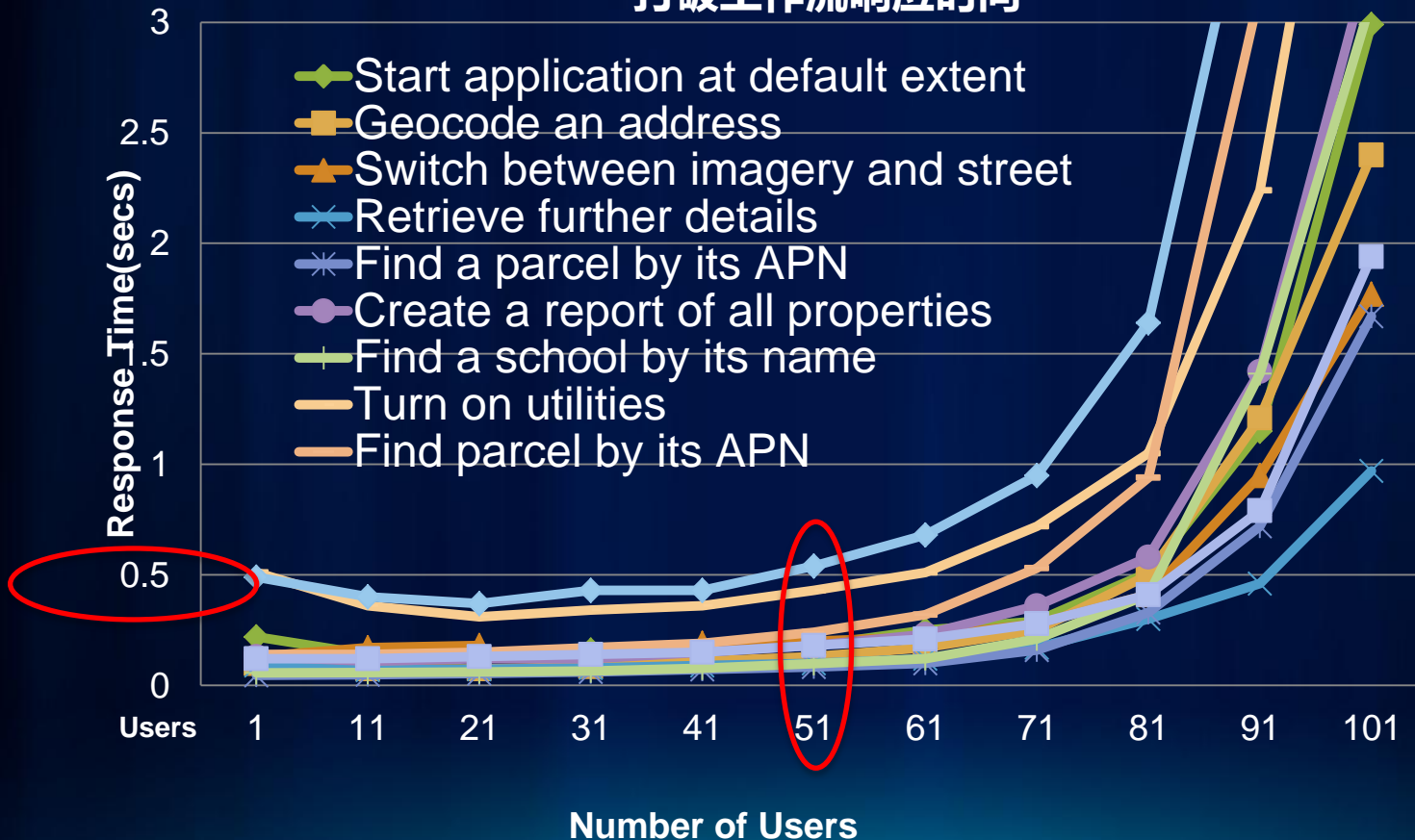


# 测试方法：用户 workflow 测试

- 目标：模拟和负载测试 “真实环境” 用户 workflow
- 过程：
  - 记录用户与多个服务的多种资源和操作交互的 workflow（包含请求间的思考时间）
  - 决定 workflow 中每一步骤以及对于整个 workflow 的可接受的事务时间
  - 在单步负载测试中重复执行 workflow，直到可接收的事务时间被打破

# 测试方法：用户 workflows 测试

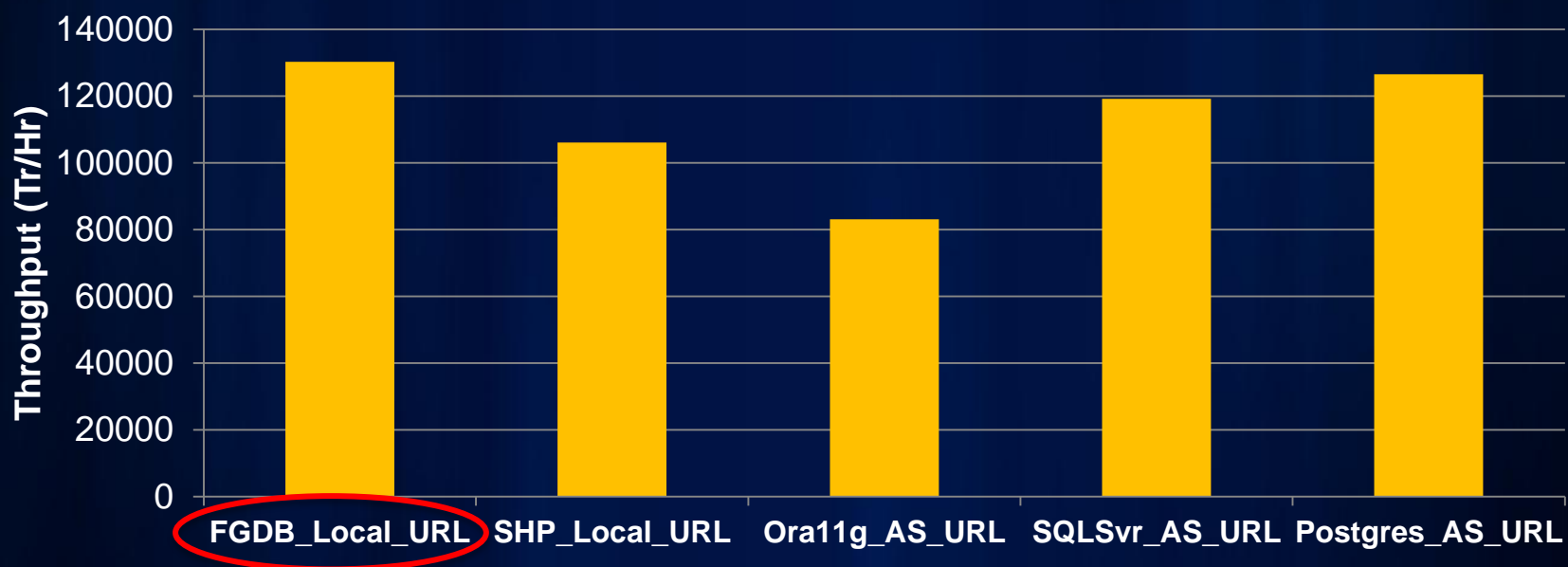
打破 workflow 响应时间



# 最佳实践：服务配置优化

## 综合 - 数据格式

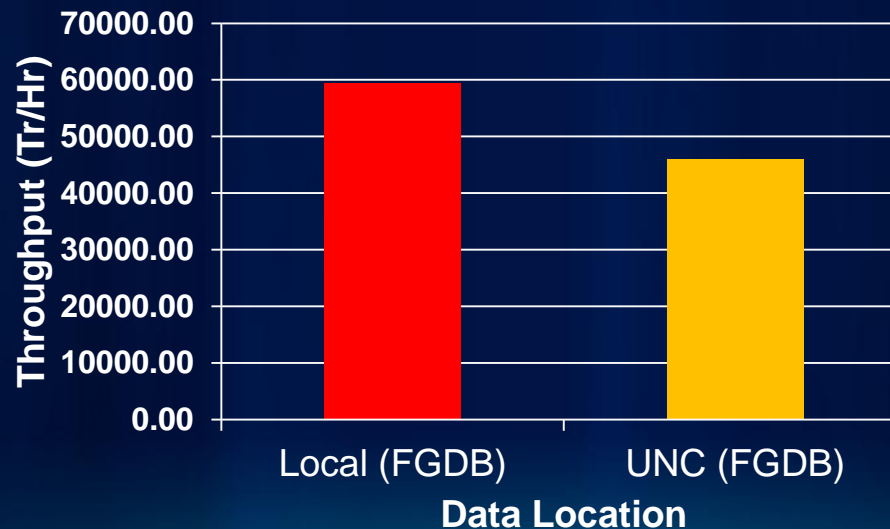
低复杂度地图：Throughput vs. Data Source



# 最佳实践：服务配置优化

## 综合 – 数据存储位置

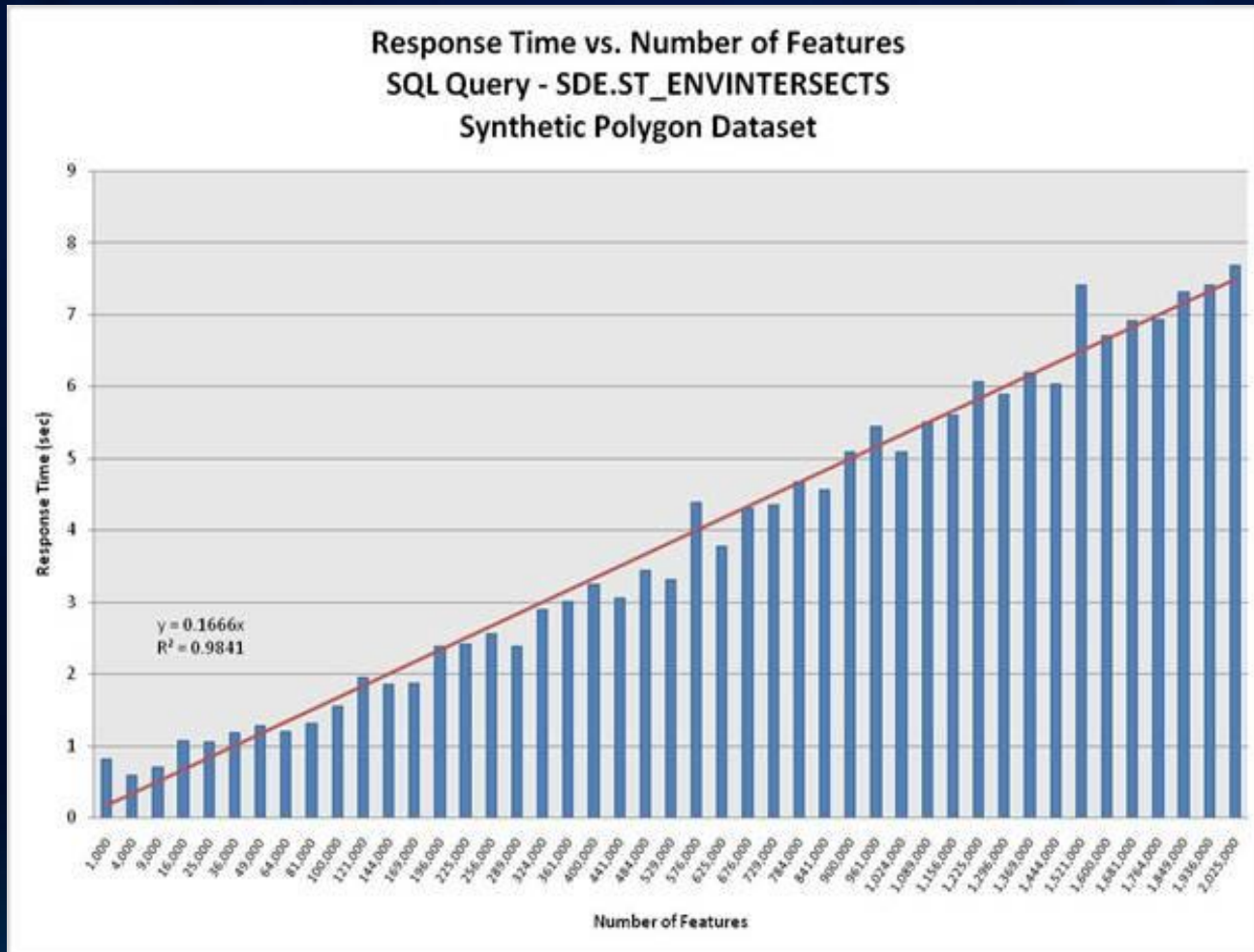
- UNC/CIFS/SMB 协议总有额外消耗（非常多的附加信息）
- 尽可能存储数据到本地
- 大量I/O 请求下差距明显





# 最佳实践：服务配置优化

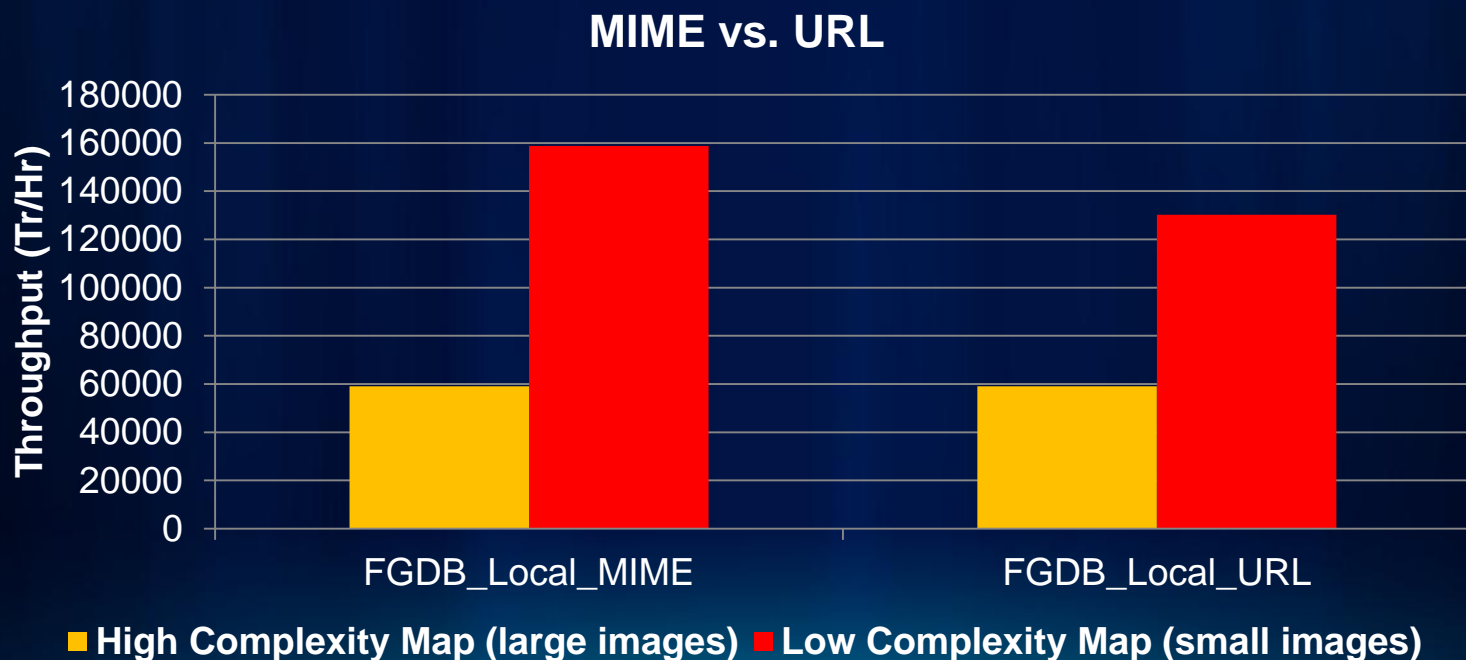
## 地图文档 – 设置比例尺依赖 (减少复杂度)



# 最佳实践：服务配置优化

## 地图服务 – 请求返回类型(MIME vs. URL)

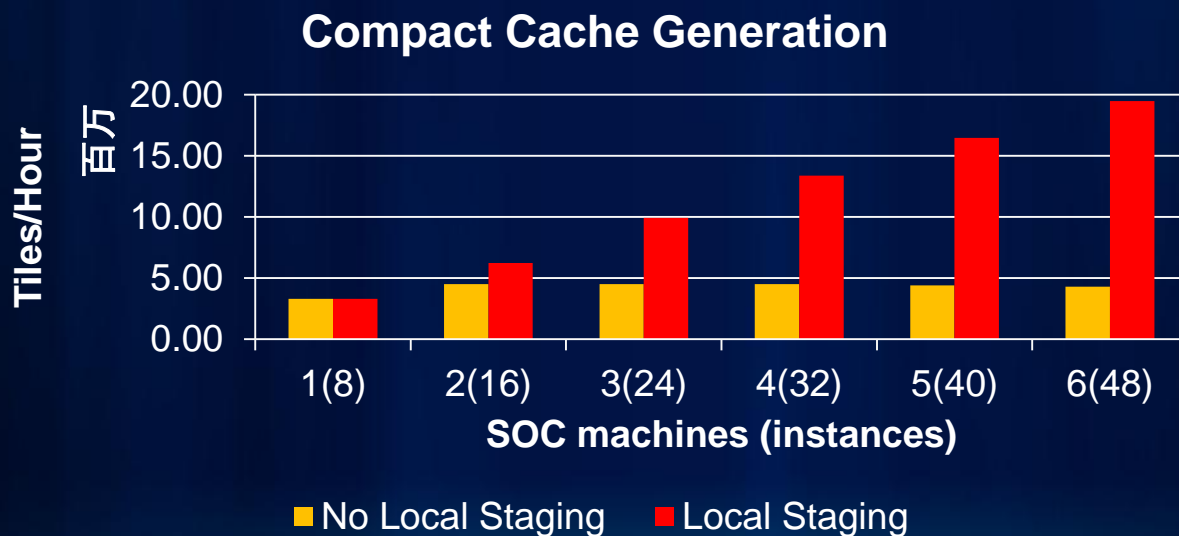
- MIME 的可伸缩性优于 URL
- Disk/UNC 共享瓶颈会先于网络带宽出现



## 最佳实践：服务配置优化

### 地图缓存 – 紧凑缓存产品(本地临时存储)

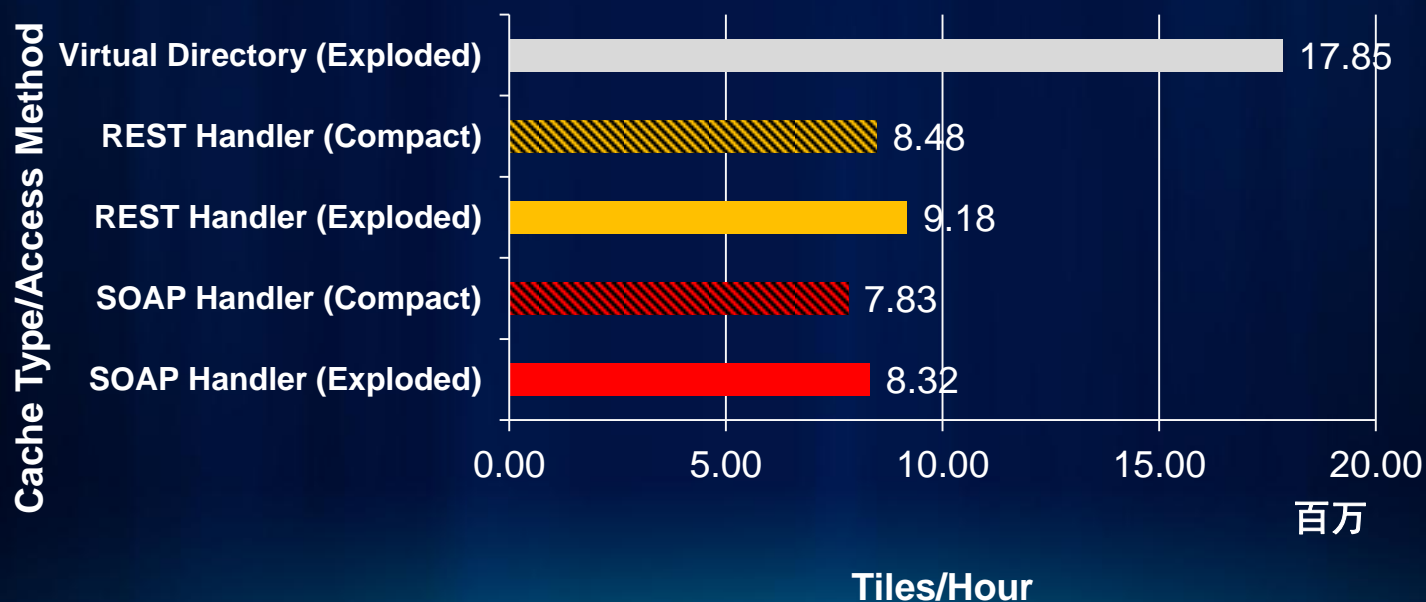
- 开启线性伸缩性
- 性能数倍于非本地临时存储的吞吐量



# 最佳实践：服务配置优化

## 地图缓存 - 消费方式

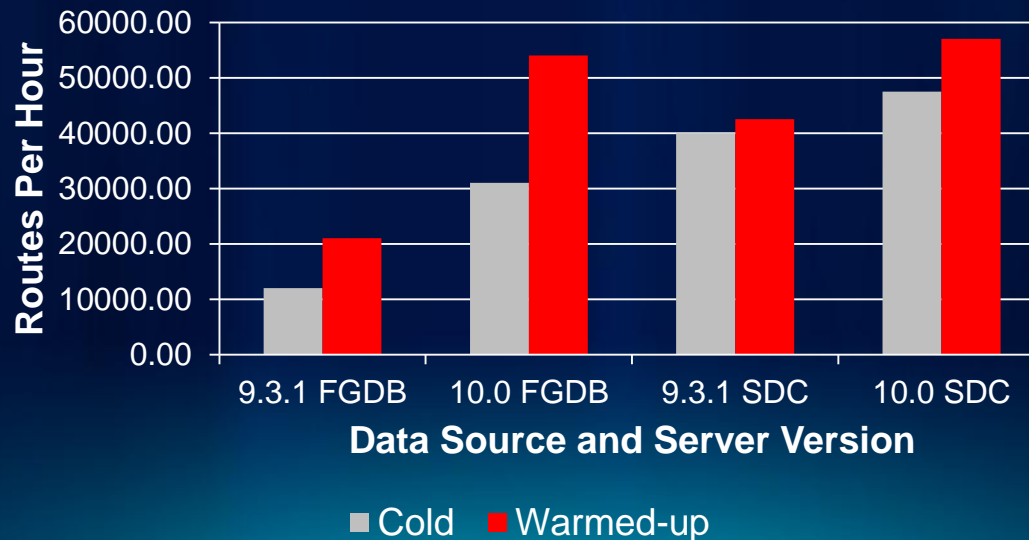
- 访问紧凑缓存要比访问离散缓存略慢
- 访问缓存效率 – SOAP < REST < Virtual Directory



# 最佳实践：服务配置优化

## Geocode, Network Analyst

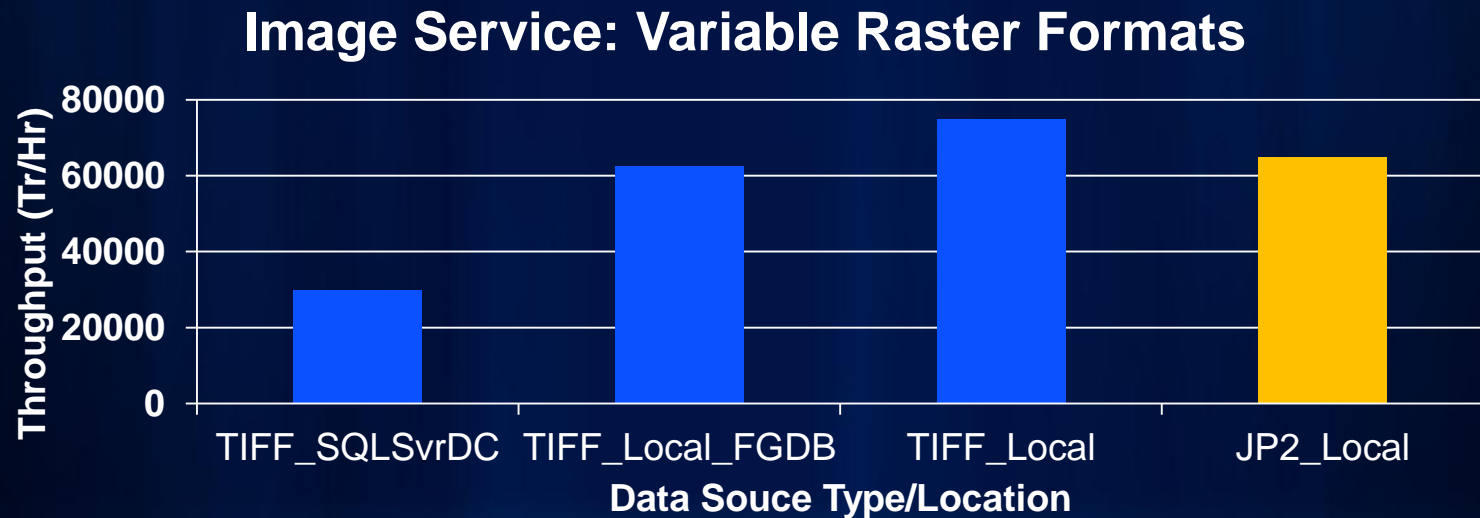
- 为了优化性能，服务热身是有必要的
  - 在上线前使用最常用的路线访问服务
  - 使用ArcScripts Java 工具(scriptID 16873)预打开FGDB中的文件



# 最佳实践：服务配置优化

## 影像服务

- 栅格格式
- 压缩
- Tiled, TIFF 拥有最大的吞吐量

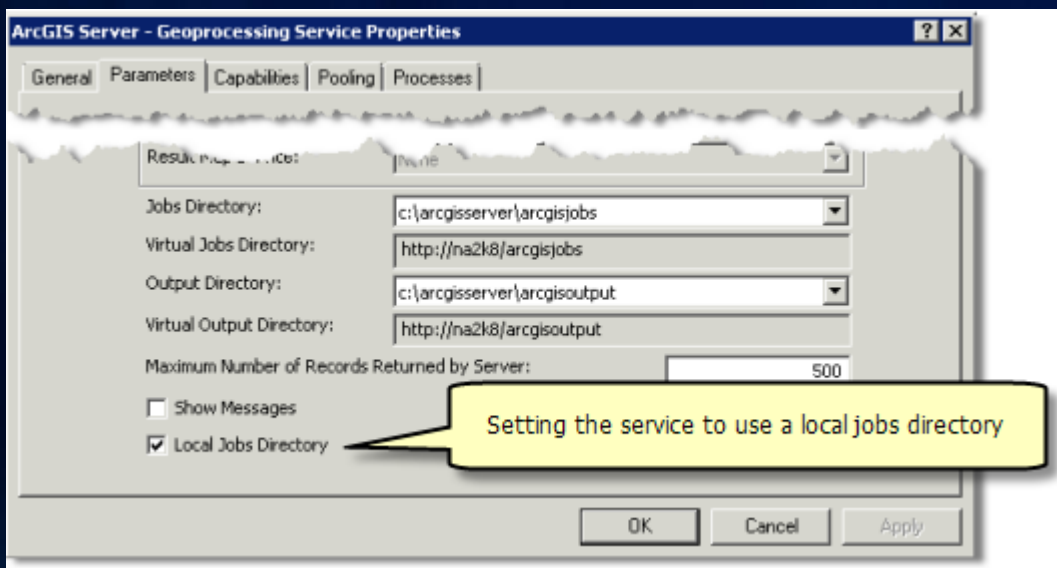




# 最佳实践：服务配置优化

## Geoprocessing服务 – 本地 Jobs 目录

- 最重要的性能影响因子
- 9.3.1/10.0 允许简单部署



# 最佳实践：架构优化

## Web services handlers

- LSASS 优化 (仅限.NET)
  - 默认每个服务请求都会校验
  - 改变IIS应用程序池标识来避免这个问题
    - 具体操作，参考技术文章：  
<http://support.esrichina-bj.cn/2010/0208/569.html>
- 额外的handlers确保SOC性能线性增长

### 11 SOC Machines (88 cores)



# 最佳实践：架构优化

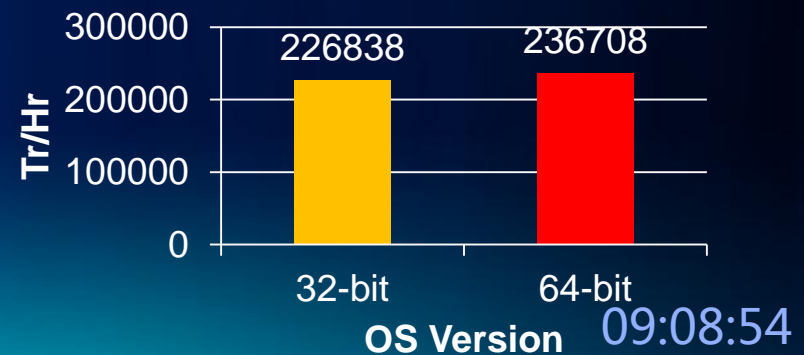
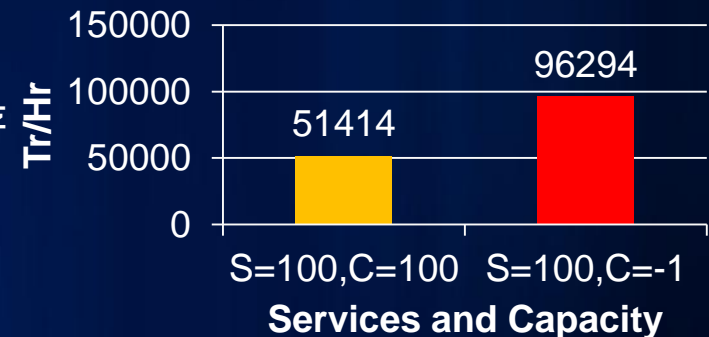
## 软件网络负载均衡

- 伸缩性依赖于适当的Web server线程管理
  - 在web garden 中IIS 工作进程数/CPU 分配的比率
  - Apache 线程配置

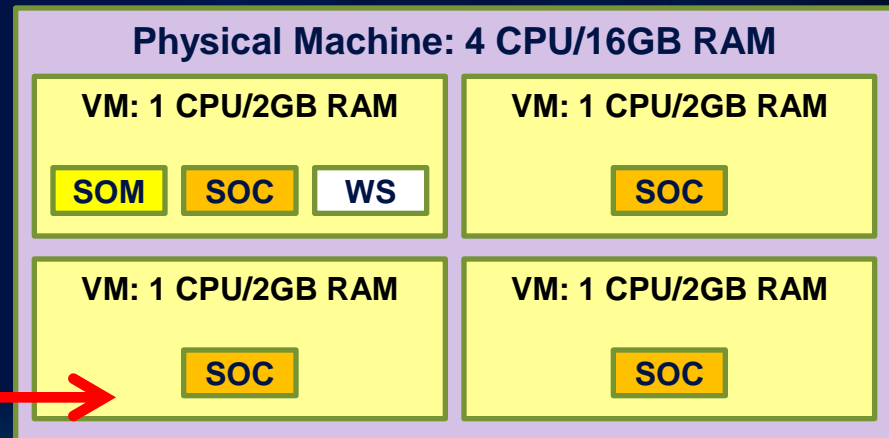
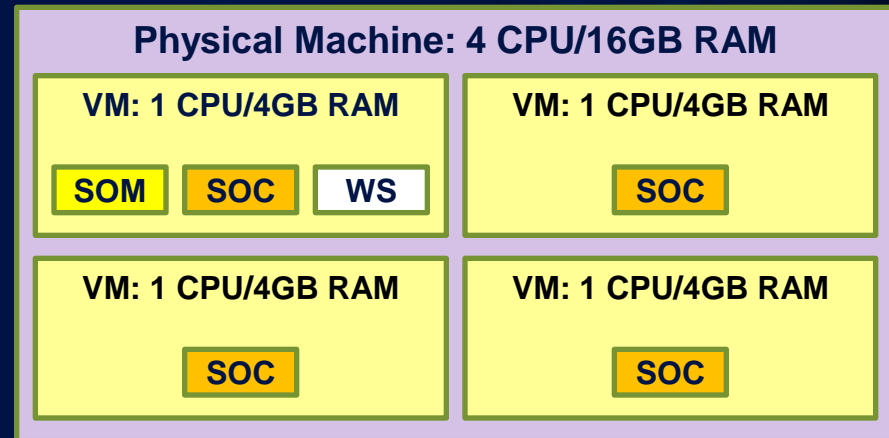
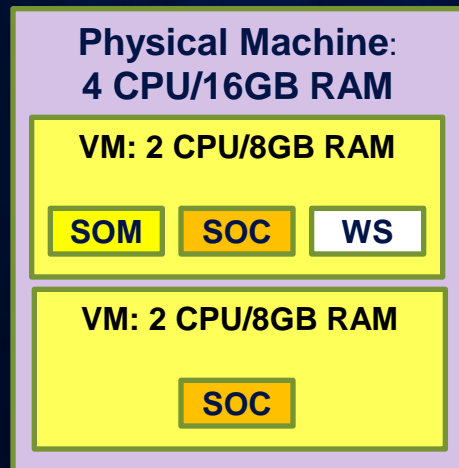
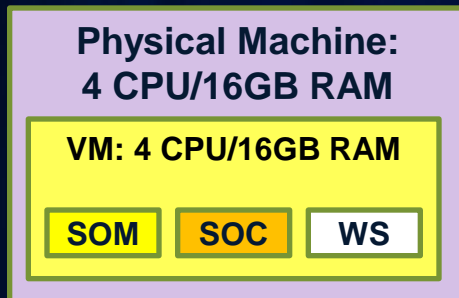
# 最佳实践：架构优化

## SOM/SOC

- SOM 很难出现瓶颈
  - 每核cpu在负载60%时每秒支持165个 地图绘制请求
  - 仅在需要冗余备份的情况添加额外的SOMs
- 尽量不使用 “Capacity”
  - 仅当需要为非-ArcGIS Server进程保留内存时使用
  - 启动/停止 SOC 比内存交换效率更低
- 32 vs. 64 bit(仅限10.1以前版本)
  - ~5% 改善



# 最佳实践：虚拟环境部署方案推荐

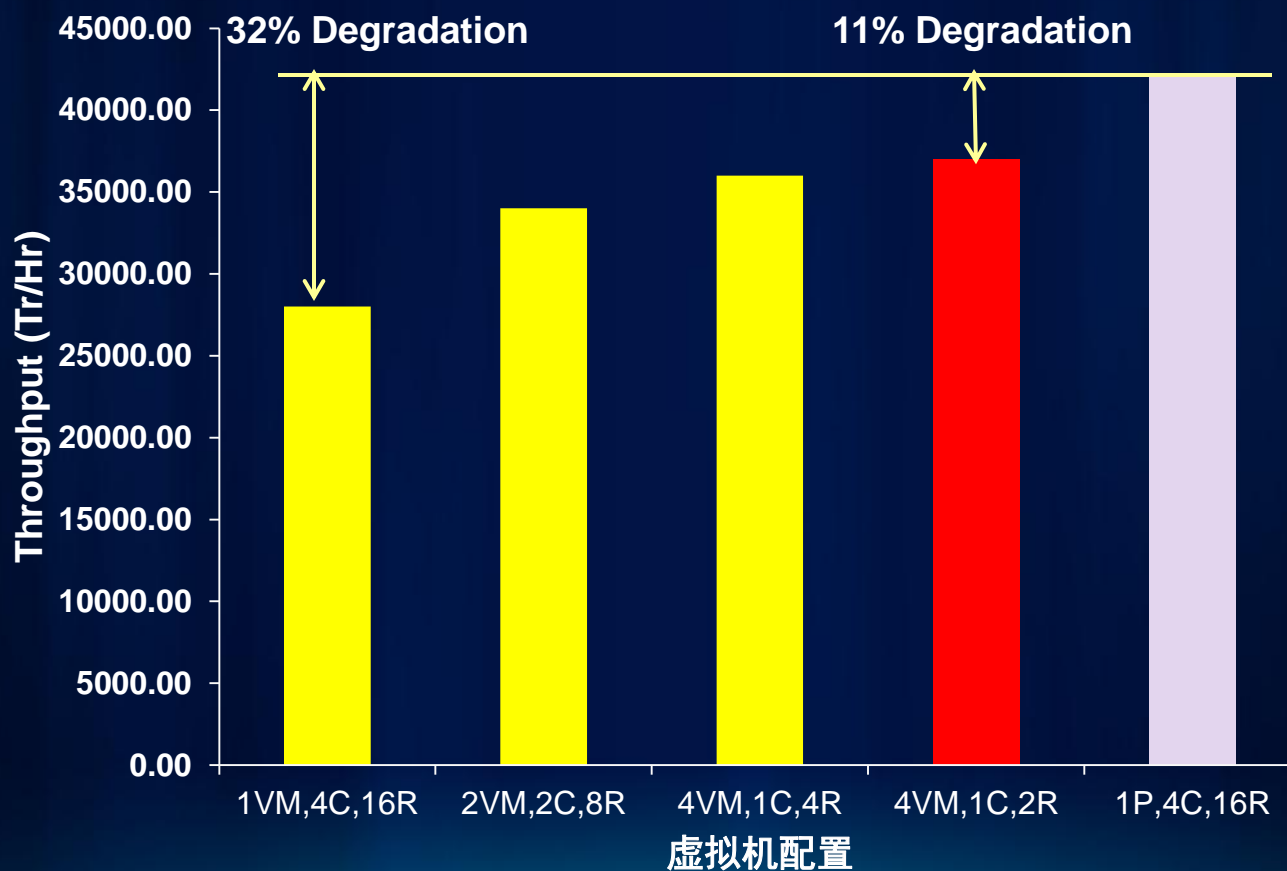


最优配置

4 VMs, 1 CPU/VM, 2GB RAM/VM

# 最佳实践：推荐的虚拟机配置

虚拟化损耗(物理机 vs. 各种虚拟配置)





# 性能诊断



# 案例分析

- 案例1：CPU 为何压力上不去？
- 案例2：开发阶段很好，上线之后出现问题？

# CPU 压力上不去



为什么不是所有的CPUs 都被使用？

# CPU 压力上不去

## 单客户端

- 串行请求

## 思考时间

- 并行请求
- 思考时间 > 事务时间

## 单实例

- Max instances = 1

## WS handler

- Lsass.exe
- 并发量过大，web server出现瓶颈

## Disk I/O

- UNC 造成I/O瓶颈

## 上线之后出现问题



我不理解，开发阶段明明很快？

## 上线之后出现问题

### 业 务

- 给司机分配工作区。
- 司机围绕工作区内的几个中心进行配送。
- 全国范围。

### 应 用

- 需要编辑工作区并分配路线。
- 构建客户化的Web应用。
- 访问位于全国总部的服务器。
- 上线试运行。



# 上线之后出现问题

## 问题

- 即使只是地图浏览，性能也很慢。
- 不一致性 – 一般很快，偶尔慢。
- 只是某些机器会慢。

## SOC?

- 将 ArcGIS Server 的日志调到Verbose 级别。
- 可以通过日志搜索大量的运算信息、渲染时间等等。
- SOC 一直很快。

## 应用?

- 尝试另外的应用, 使用相同的服务。
- 在应用中通过日志记录逝去时间。
- 通过发送和接收时间间隔查找问题。没有应用逻辑。

## 客户端 硬件?

- 没有问题需要监控。

# 上线之后出现问题

## 网络?

- 限制的带宽。
- 通信竞争。
- 为什么其它应用没有问题?

## 再看应用?

- 一个事物- 多次网络交互。
- ADF + DCOM。

## 解决方案

- 短期: 提升网络超时时间。
- 长期: web API + SOE。

# 上线之后出现问题

## 应用设计

- 最小化交互(SOE)
- Basemap + Operational

## 间歇性的问题

- 分析, 预测很棘手
- 通常难以复现
- 监听解决方案

## 日志

- 提升日志级别
- 请求和子请求级别

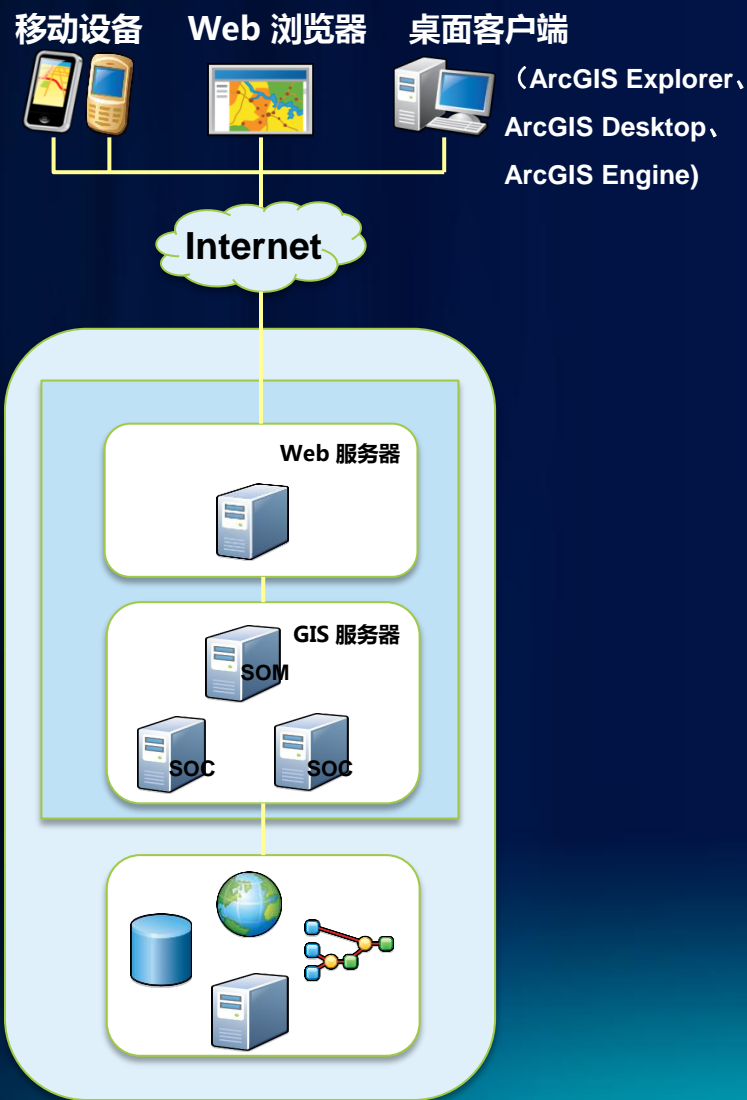


# ArcGIS Server 10.1的部署及可伸缩性

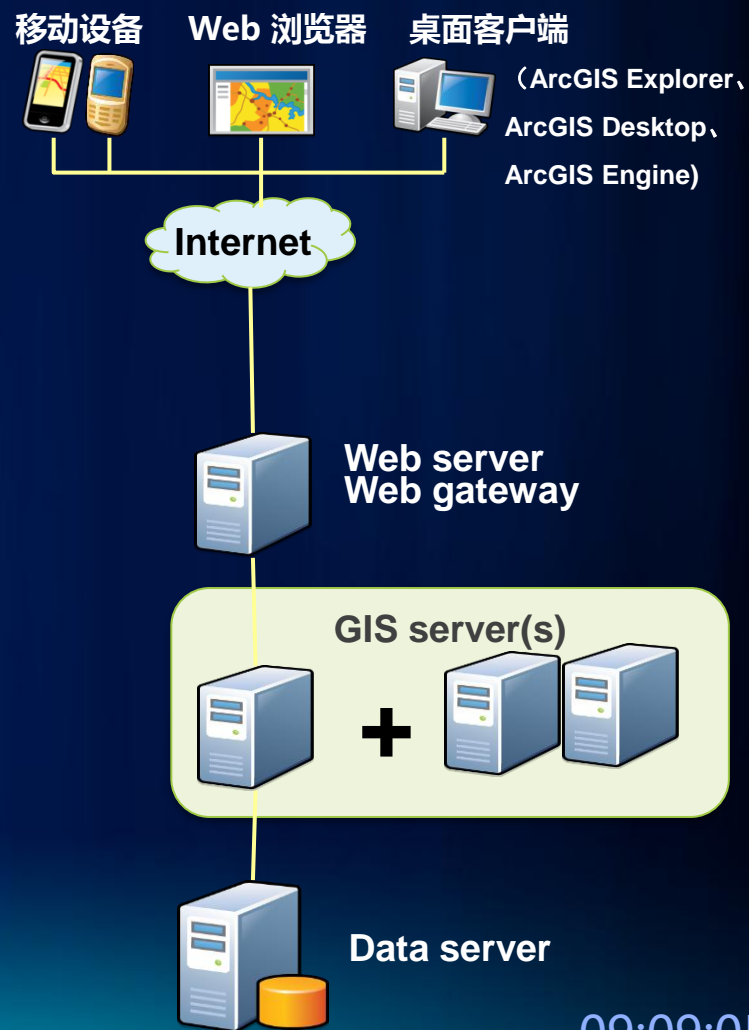


# ArcGIS Server 10.1 系统架构的改变

## 10以前版本的系统架构



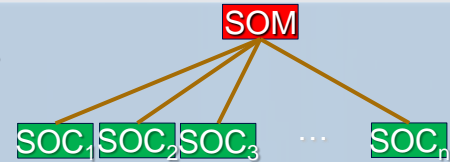
## 10.1的系统架构



# 集群部署架构演进

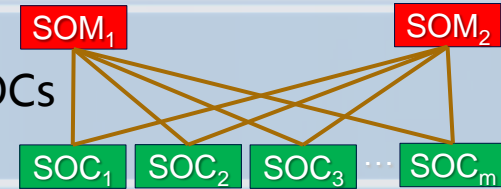
## 标准架构

- 单个SOM,多个SOCs
- 最简单



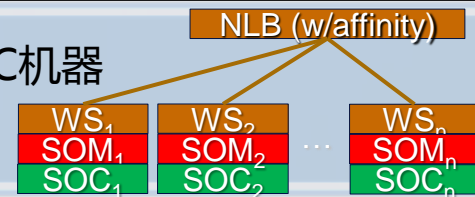
## 容错的标准架构

- 两个SOM, 多个SOCs



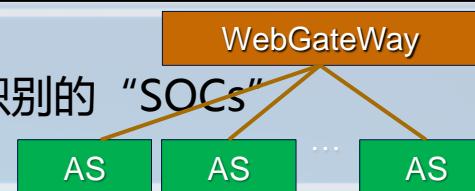
## Silo 架构

- 所有都是SOM+SOC机器
- 需要负载均衡



## Peer to Peer架构

- 没有 "SOM" .
- 多个平等的, 彼此识别的 "SOCs"
- 内部负载均衡



# 带群集的P2P 架构

- 每台机器都一样，彼此认识
- 很容易按比例伸缩
- 很容易安装
- 没有单点故障
- 支持分配机器组给一个服务

WebGateway



GIS Server



GIS Server



GIS Server



GIS Server





## Peer to Peer 架构特性

### 故障转移

- Server 架构自动具备

### 容 错

- 没有单点故障，自动具备
- 备份配置文件

### 监 控

- 很容易整合标准监控工具

### 系统恢复

- 脚本工具便于更简单的系统恢复



# 您的问题解决了么？

- 我的web 应用中，地图出图为什么慢？
- 最短路径分析速度有点慢，该如何调优？
- 我该如何配置服务的实例数，才能达到性能最优？
- 我要配置多少台GIS服务器，才能满足现有用户规模？
- ArcGIS Server 该如何部署，才能达到性能最优？



分 享 地 理 价 值

# ArcGIS在线体验中心(原群英萃)

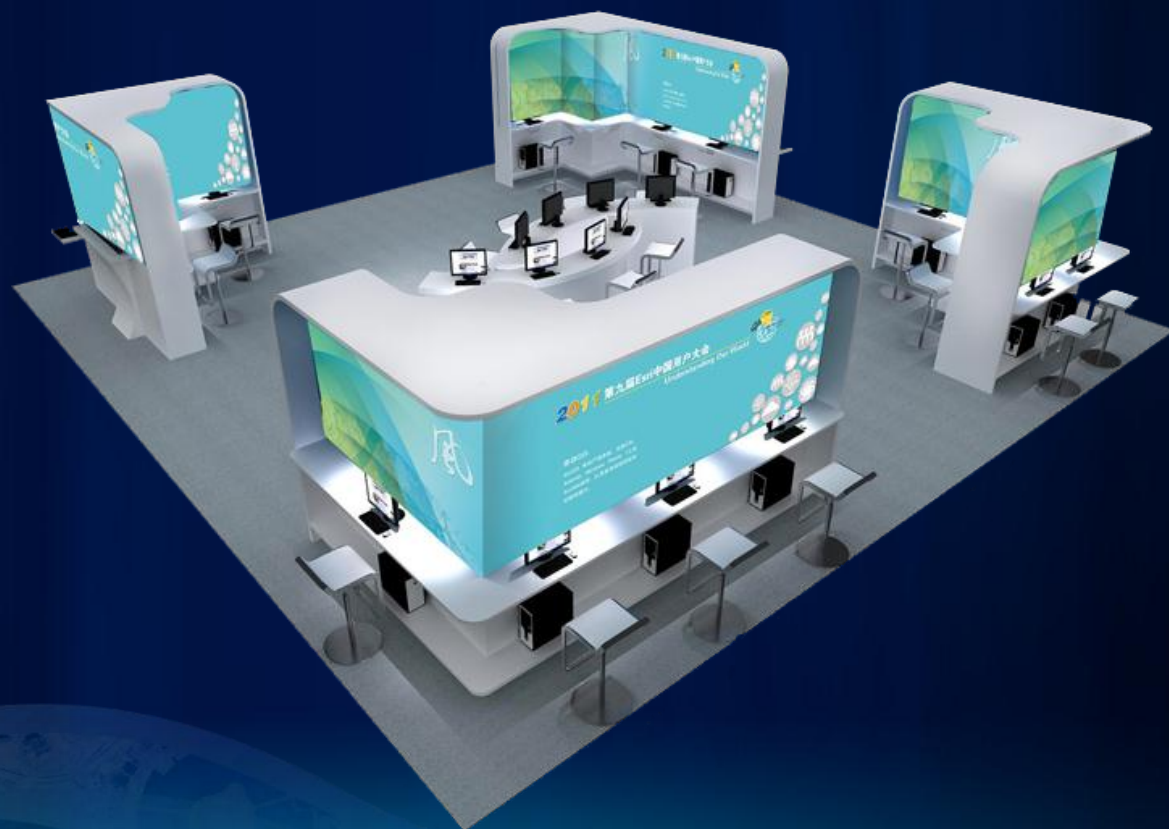


在线体验  
视频中心  
地图长廊  
资源下载  
知识中心  
虚拟课堂



<http://tm.esrichina-bj.cn>

# 欢迎移步到体验区 体验炫彩GIS世界



# 高级地图缓存技术

Esri 中国（北京）有限公司 程轩昂

