



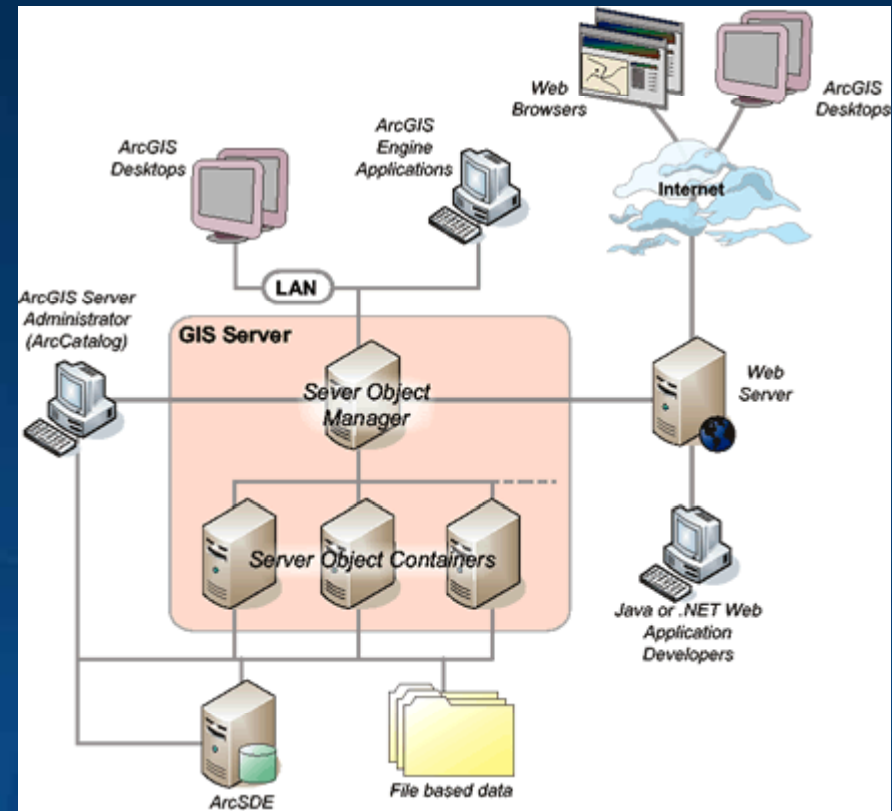
Deploying and Tuning ArcGIS Server

Presented by Jim Mason and Ian Sims
ESRI Server Development

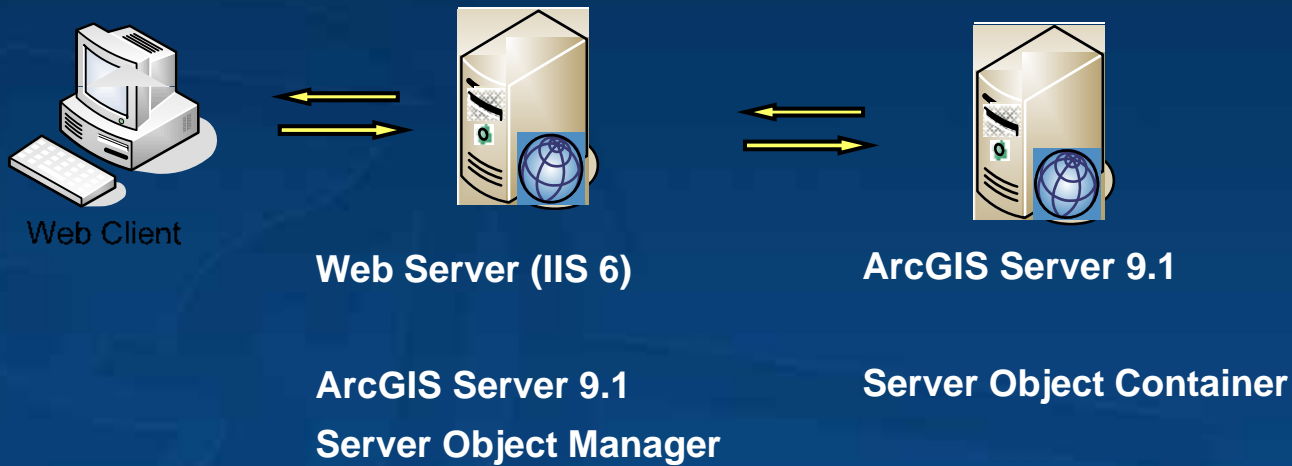


Built using ArcGIS Server

- Software components for building server-side GIS applications
- Designed to
 - Build centrally-managed applications
 - Expose advanced GIS functionality
 - Support multiple users



ArcGIS Server Case Study Hardware Environment





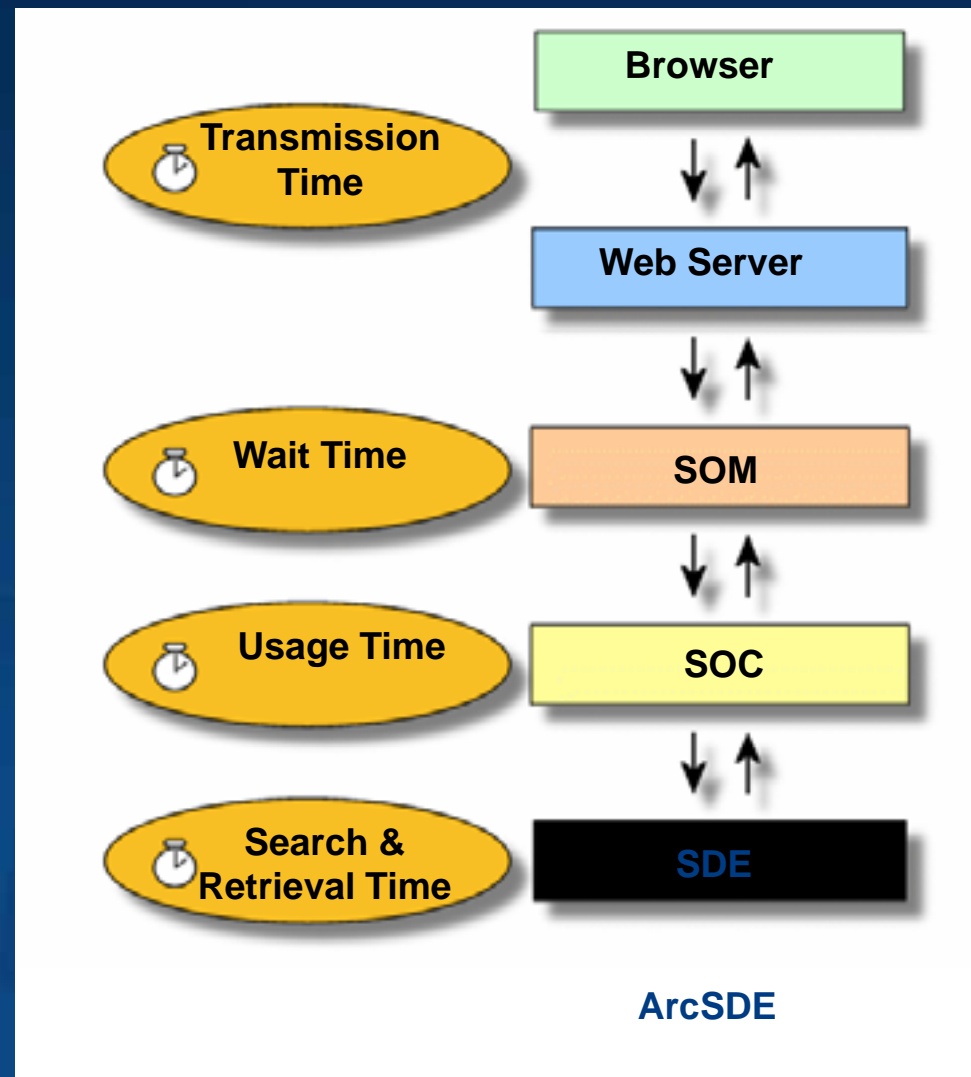
Network Analyst Samples on EDN

- Point to Point Driving Directions
- Closest Facility
- Routing with Time Windows
- Samples in C#, VB.Net, and Java
 - Includes a NetworkAnalyst utility class



ArcGIS Server: Response time factors

- Four main factors of response time
- Multiple tiers
- Performance bottleneck can occur in each tier





Bottlenecks

- CPU
- Network
- Memory
- Disk
- Threads
 - Server Object availability

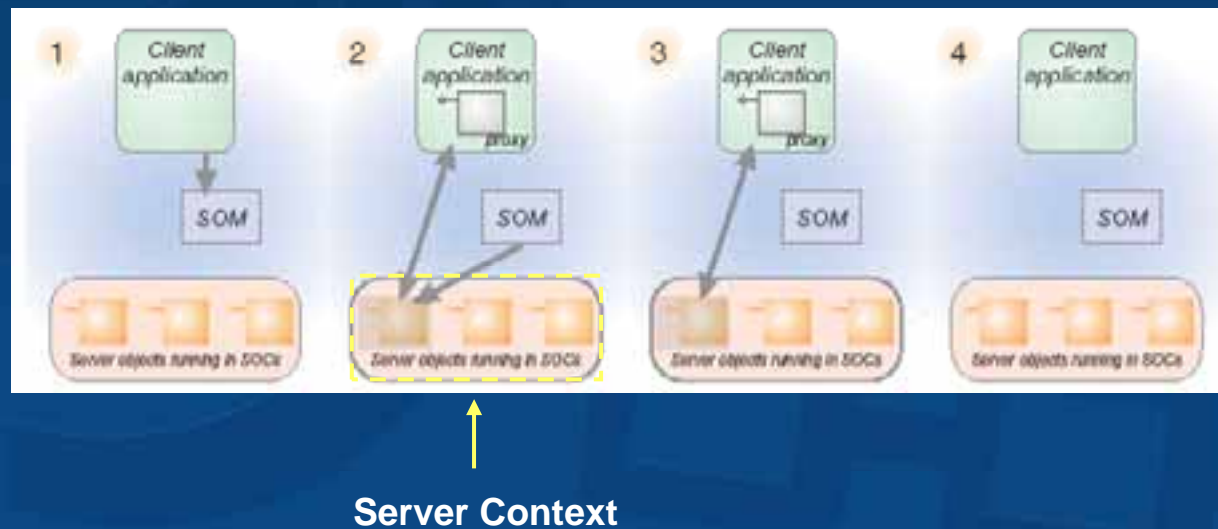


Bottlenecks

- CPU
- Network
- Memory
- Disk
- Threads
 - Server Object availability

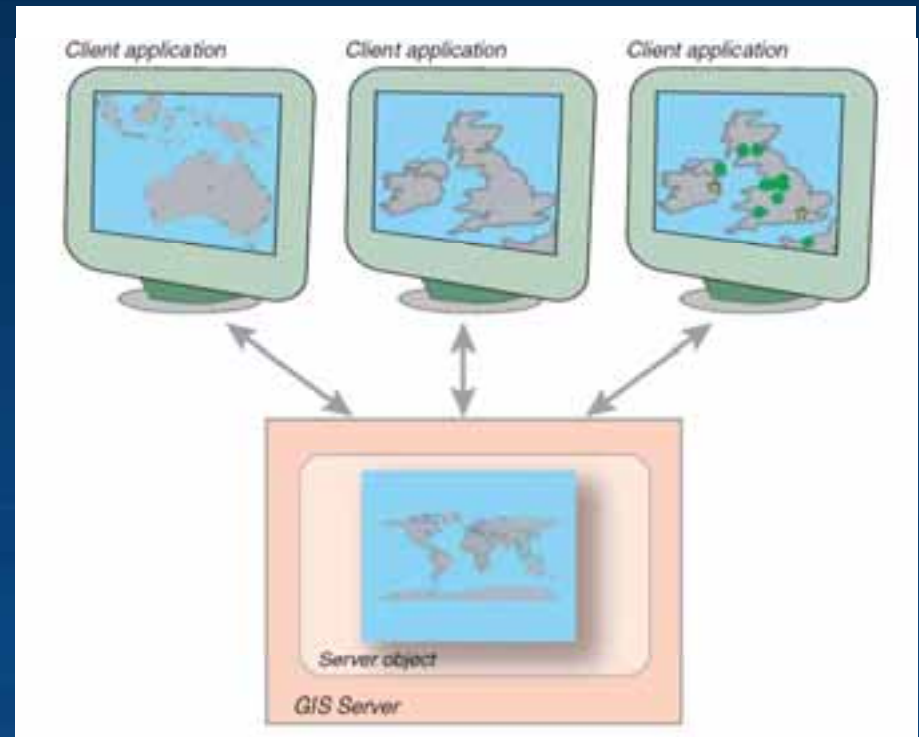
Pooled Server Objects

1. Connect to SOM and request server object
2. SOM accesses **context** and returns a proxy
3. Client accesses server object through proxy
4. Client releases server object *back to the pool*



Shallowly Stateful Applications

- Must maintain some state information (Extent of the map, layer visibility, etc.)
- Use the web server's session state management
- Cannot change the properties of a pooled server object
- Scale better due to shared object pool





Network Analyst and Shallowly Stateful Applications

- Serializable context
- Allows use of pooled server objects
- Example: Step by step directions

The screenshot shows a web-based driving directions application. At the top, there is a map with a yellow highlighted route. Below the map are navigation controls (zoom in, zoom out, pan, home) and three links: [New Directions](#), [Reverse Directions](#), and [Printable Version](#). The main content area is titled "Driving Directions" and contains a table with four steps. The third step, "Turn right on Bay St", is circled in red. The table columns are Step, Directions, Length, and Time.

Step	Directions	Length	Time
1	Depart 580 Beach St, San Francisco, Ca. Go West on Beach St	< 0.1 mile(s)	< 1 minute
2	Turn left on Leavenworth St	0.1 mile(s)	< 1 minute
3	Turn right on Bay St	0.3 mile(s)	< 1 minute
4	Turn left on Polk St. Arrive at 2401 Polk St, San Francisco, CA	0.4 mile(s)	< 1 minute

Local intranet

ArcGIS Server – Configuring Pooled Instances



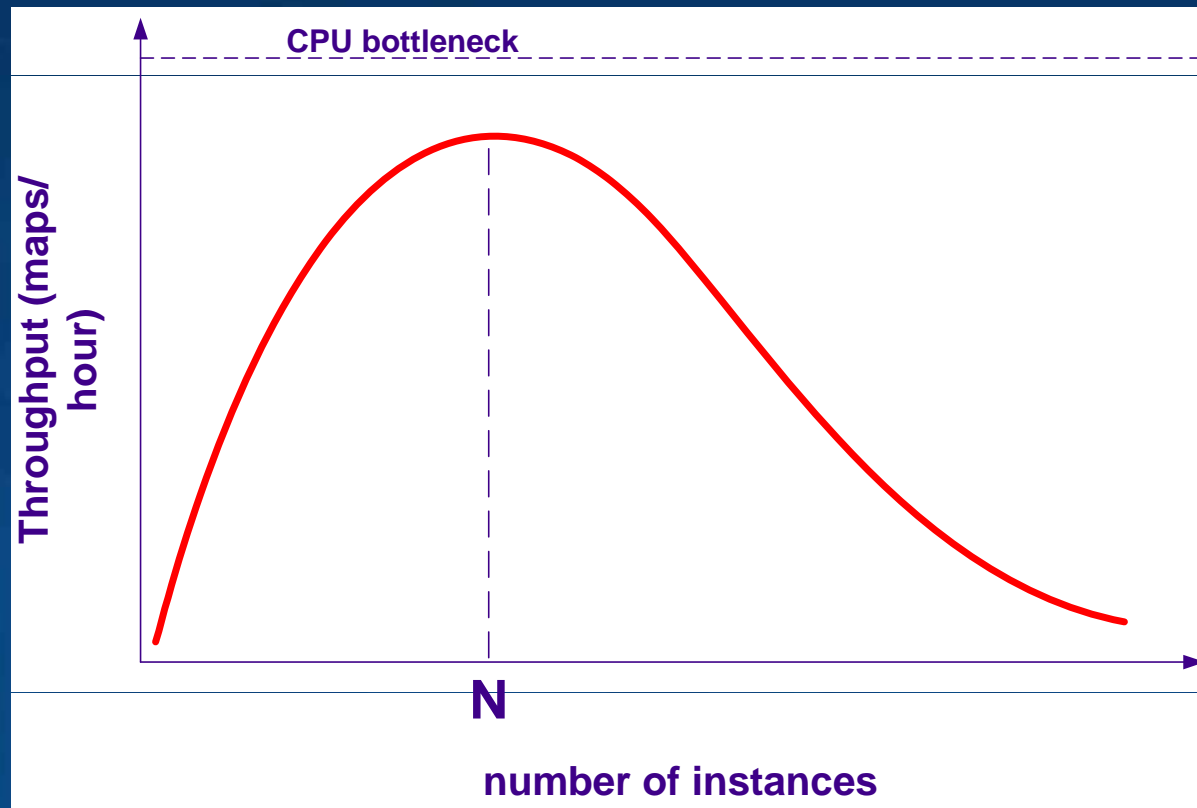
- Define Min-Max instances
- Instances are distributed across all host servers

A screenshot of the 'ArcGIS Server - Map Server Object Properties' dialog box, specifically the 'Pooling' tab. The dialog has four tabs: 'General', 'Parameters', 'Pooling', and 'Processes'. The 'Pooling' tab is active and contains the following text: 'Pooling enables instances of the configuration to be pre-created and available for immediate use by clients.' Below this, there are two radio button options. The first is 'Pooled - A request to the server for this object returns a pre-created object from the pool of available objects.' This option is selected. Below it are two input fields: 'Minimum number of instances available:' with the value '2' and 'Maximum number of instances available:' with the value '8'. The second radio button option is 'Not Pooled - A request to the server for this object returns a newly created instance of the object.' Below it is one input field: 'Maximum number of instances available:' with the value '4'. At the bottom of the dialog, there are three input fields: 'Server Object Maximum Usage Time:' with the value '600' and the unit 'seconds', and 'Server Object Maximum Wait Time:' with the value '60' and the unit 'seconds'. At the very bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Apply'.



ArcGIS Server optimum number of instances

- Set instances to level where maximum throughput occurs (usually between 2 to 4 instances per CPU)





Bottlenecks

- CPU
- Network
- Memory
- Disk
- Threads
 - Server Object availability

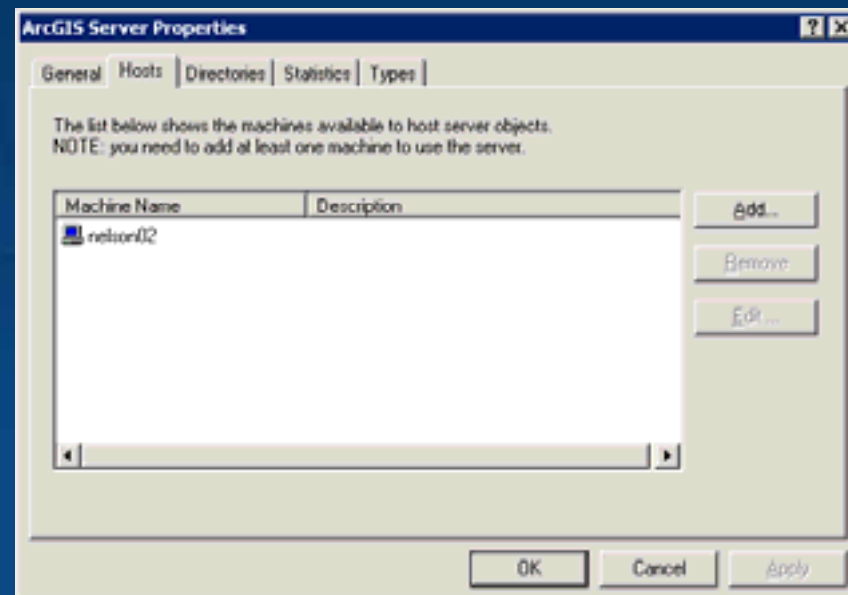
ArcGIS Server Case Study Hardware Environment





ArcGIS Server – Adding more CPUs

- Add container machines (SOC) to the SOM
- Requirements
 - Any machine on your local network
 - Must have GIS Server ArcObjects and SOC components





Detecting Bottlenecks

- Single user testing is inadequate
- Simulating multiple users
 - Low Tech: Recruit others in the office and synchronize using the PA system
 - High Tech: Load simulation tool



Load Simulation Tool

- Build
 - Code your own test harness
- Buy
 - Price: (free/bundled to \$100K or more)
 - Technical vs. Functional
 - Microsoft
 - Web Application Stress Tool (WAST)
 - Application Center Test (ACT)
 - Visual Studio 2005 Team Edition for Testers

Measuring Web Performance

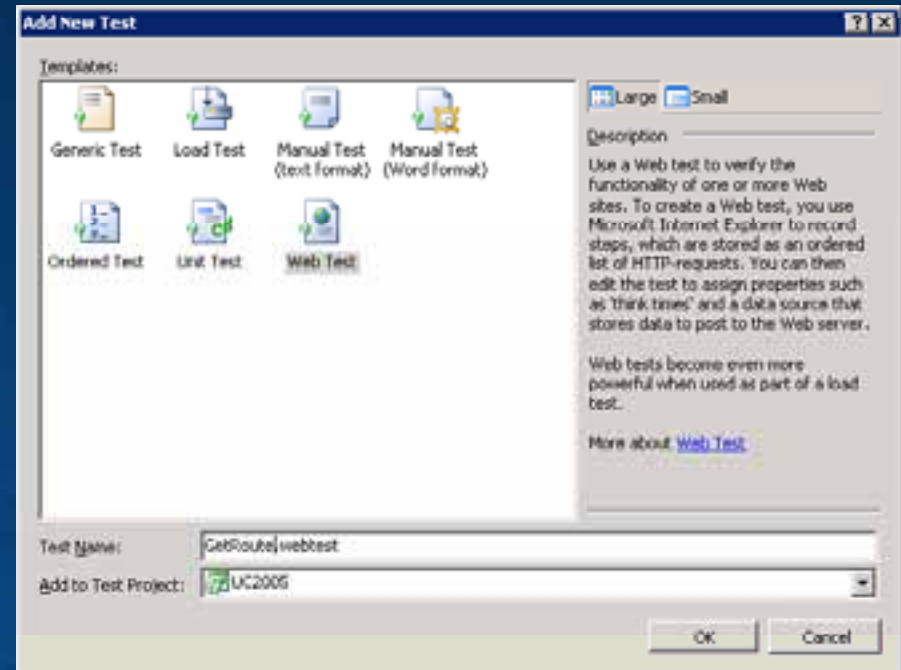


- Client Response Time
 - How “fast” does web application appear to remote browser hitting the site
 - Response Time (average, 75th percentile, max)
- Machine Throughput
 - Measured in requests/sec
 - Determines number of servers you need to buy

Visual Studio 2005 Team System



- Tester tools are now on par with Developer Tools
- Unit, Web, Load, and Manual tests
- Measures code coverage





VS 2005 – Recording a Web Test


- A series of HTTP URLs that can be created or recorded from a browser session.

A screenshot of a Microsoft Internet Explorer browser window. The address bar shows the URL `http://nelson02/Routing/CSHarp/Directions.aspx`. A 'Web Test Recorder' window is open, displaying a list of recorded HTTP requests and their parameters. The first request is to `http://nelson02/Routing/CSHarp/Default.aspx` with form post parameters including `__VIEWSTATE=dDw5Mj4MDkxOTA7Qz4z/r`, `FromStreet=580 Beach St`, `FromCity=San Francisco`, `FromState=CA`, `FromZip=`, `ToStreet=2401 Polk St`, `ToCity=San Francisco`, `ToState=CA`, and `ToZip=`. The second request is to `http://nelson02/Routing/CSHarp/Directions.aspx` with parameters like `__EVENTTARGET=ToolBar1`, `__EVENTARGUMENT=ReverseDirections`, and `__VIEWSTATE=dDwMTg0DM30TY203Q6`. The main content area of the browser shows a 'Directions' page with a map. The map displays a route from 2401 Polk St, San Francisco, CA to 580 Beach St, San Francisco, CA. The route is highlighted in yellow, with a red segment indicating the starting point. The map includes street names like Lombard, Greenough, and Polk. Text on the page indicates a distance of 0.8 mile(s) and a time of 1 minute(s).

VS 2005 – Creating a Load Test



New Load Test Wizard [?] [X]

 **Edit load profile settings for test scenario**

Scenario

- Load Profile**
- Test Mix
- User Profile
- Browser Mix
- Network Mix
- Counter Sets
- Run Settings

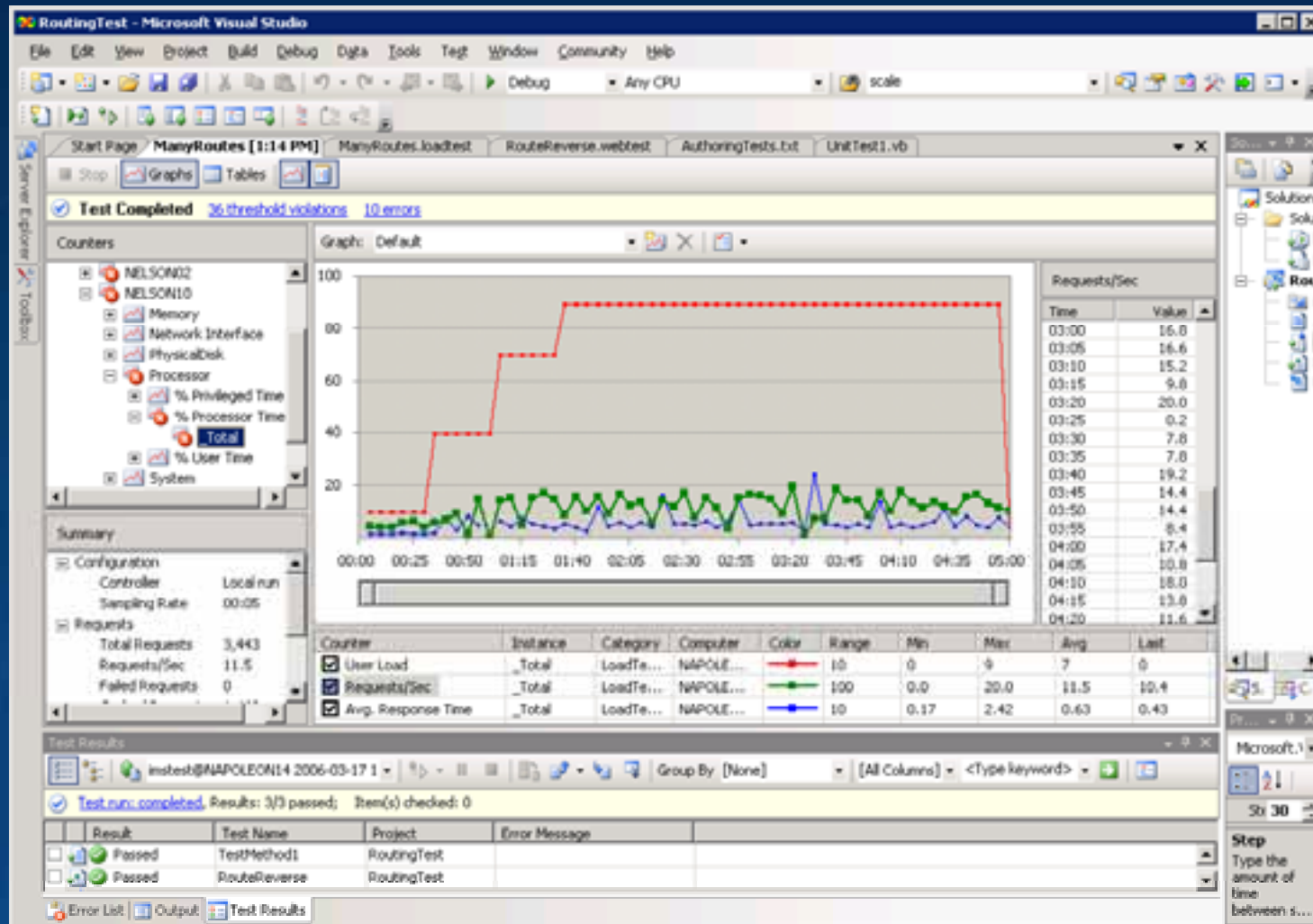
The load profile is used to emulate a user load during a load test.
Select a load profile:

Constant Load:
User Count: users

Step load:
Start user count: users
Step duration: seconds
Step user count: users/step
Maximum user count: users

< Previous Next > Finish Cancel

VS 2005 – Running a Load Test





Monitor Statistics and Log Files

- Creation Time
- Wait Time
- Usage Time

ArcGIS Server Properties

General | Hosts | Directories | **Statistics** | Types

Server Obj: SanDiego.MapServ
Host(s): <All>
Stat. Type: SO. Usage Time
Interval: Last 60 minutes
Show Statistics

Statistics Time Range
Start Time:
2005-07-19T16:55:00

Server Object Usage Time:
Total number of requests: 2
Number of requests succeeded: 2
Number of requests timed out: 0
Avg usage time: 1.921500 Seconds
Min usage time: 1.734000 Seconds
Max usage time: 2.109000 Seconds
Sum usage time: 3.843000 Seconds

C:\Program Files\ArcGIS\log\Server-20050719-175021.xml - Microsoft Internet Explorer

Address: C:\Program Files\ArcGIS\log\Server-20050719-175021.xml

2005-07-19T17:54:09	Detailed	SanDiego.MapServer	Server Context created.	nelson02	3144	1.59999999999911E-02	4006
2005-07-19T17:54:11	Detailed	SanDiego.MapServer	Server Context released.	nelson02	3916	1.73400000000001	4007
2005-07-19T17:54:11	Detailed	SanDiego.MapServer	Server Context created.	nelson02	3256	0	4006
2005-07-19T17:54:13	Detailed	SanDiego.MapServer	Server Context released.	nelson02	3868	2.10899999999998	4007

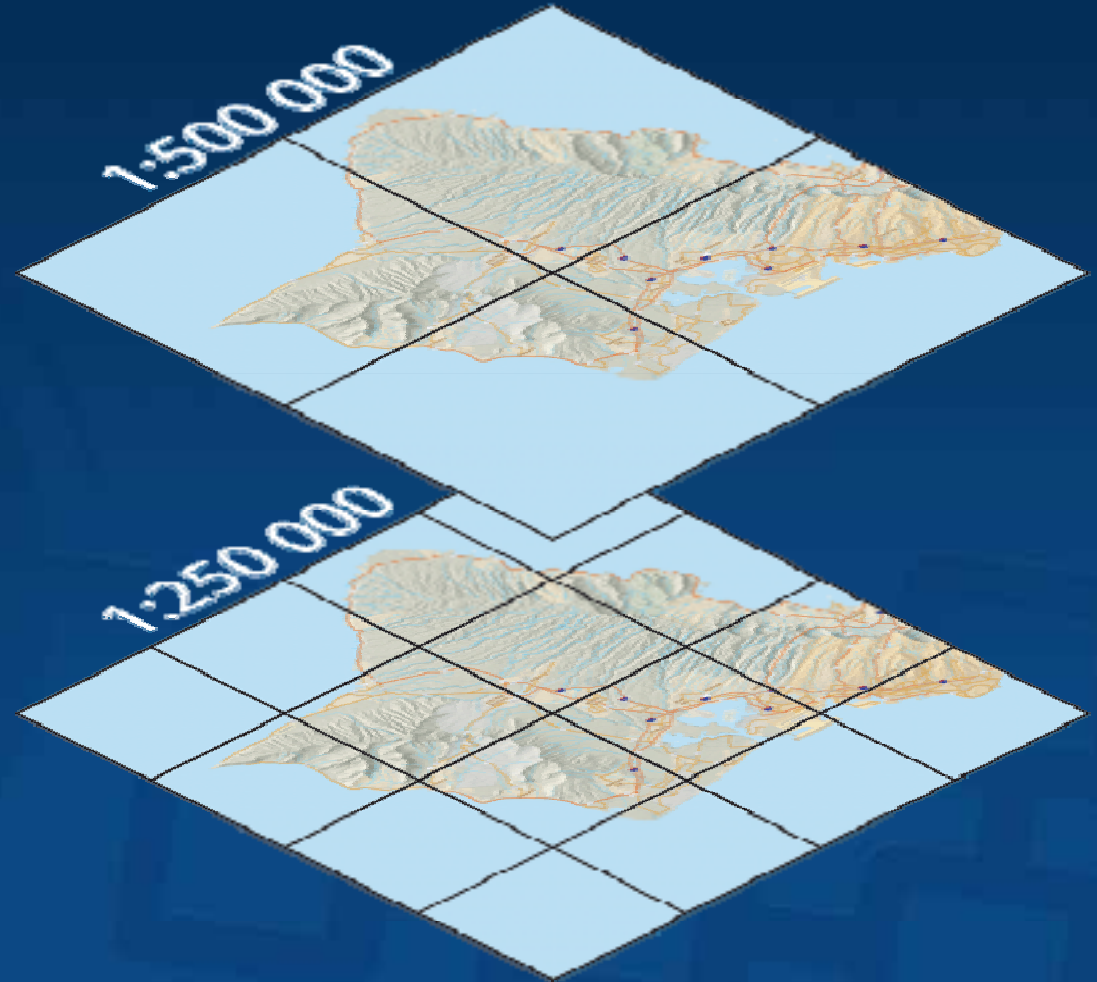


Log File Behavior

- **Modify Log Level in Server.cfg**
 - 0 (None): No logging
 - 1 (Error): Serious problems
 - 2 (Warning): Additional errors.
 - 3 (Normal): Common administrative messages of the server, warnings, and errors are logged.
 - 4 (Detailed): Common messages from user use of the server, including server objects, normal messages, warnings, and errors are logged.
 - 5 (Debug): Verbose messages to aid in troubleshooting

Map Server Cache

- Rapid display of complex base maps
- Tiles pre-rendered at specified scales
- Retrieved from web server (<http>)





Generate Map Server Cache tool

- New Geoprocessing Tool
- Generate cache on one or more servers
- Specify
 - Scales
 - Tile size including DPI
 - Antialiasing – smooth labels and lines
 - Processing threads



Cache Generation

- Data: StreetMap + hillshade
- Extent: Island of Oahu
- Scales: 1:500K, 1:250K, ..., 1:4K
- Tile size: 128 x 128
- Files: 311,468
- Size: 0.25 GB
- Size on disk: 1.25 GB
- Servers: Three 2-CPU servers, 1GB RAM

- Creation Time: 1 hour, 40 minutes



Bottlenecks

- CPU
- Network
- Memory
- **Disk**
- Threads
 - Server Object availability



Tile Size

- Default size: 512x512
- Larger size produces fewer tiles
 - Less disk space (block size)
 - Faster creation
 - Easier to manage
- Smaller size (128x128)
 - Allows incremental update of the display
 - Improves panning performance



Bottlenecks

- CPU
- Network
- **Memory**
- Disk
- Threads
 - Server Object availability

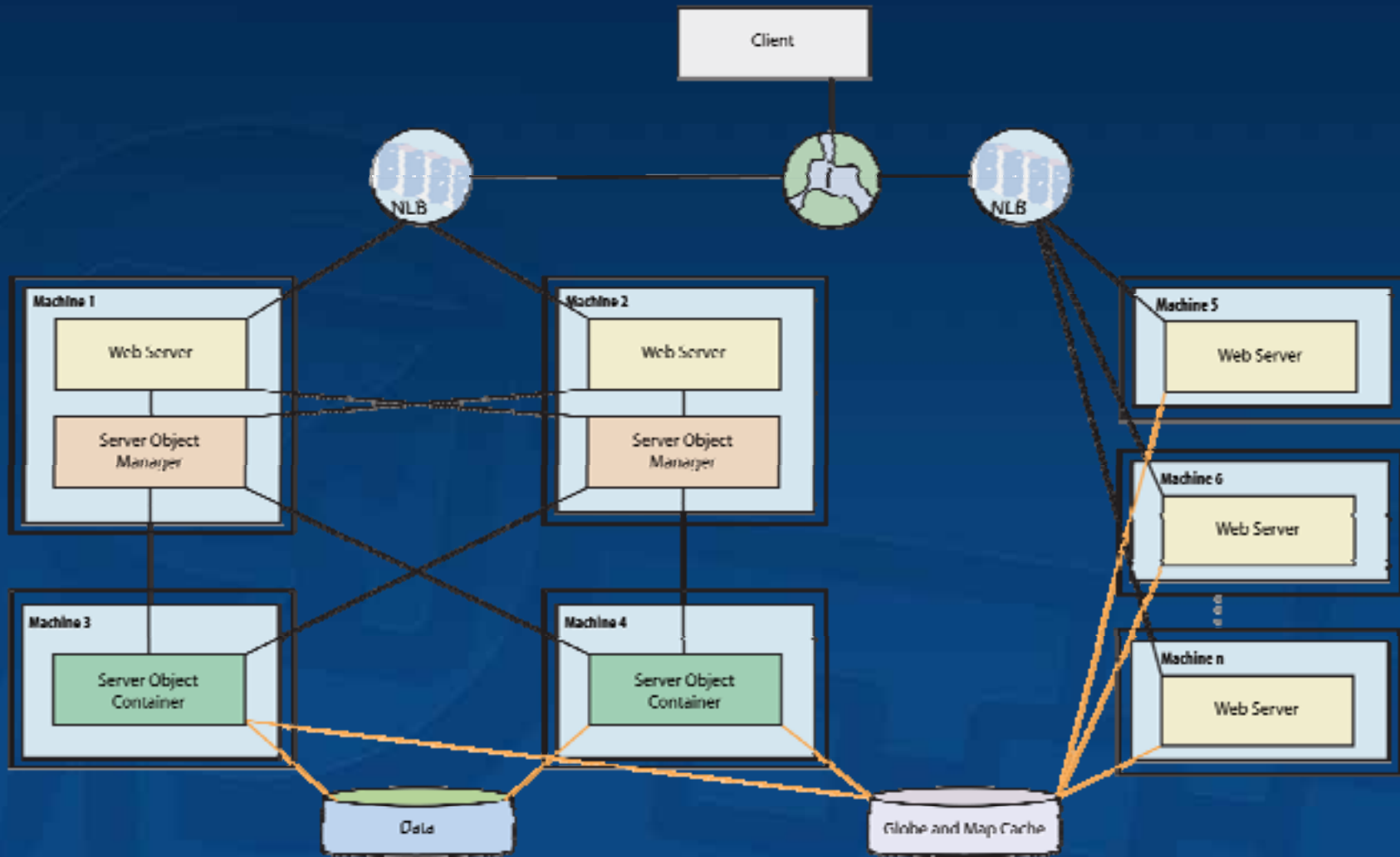


Generate Map Server Cache

- Renders 4K x 4K images
- 1 GB consumed by just 2 processes
- Options to resolve
 - Increase memory to 2 or 3 GB
 - Reduce number of instances per server



High Availability Configuration





Session Evaluations Reminder

Session Attendees:
Please turn in your session evaluations.
... Thank you