Lessons Learned

Air Force Network Integration Center

Mr. Douglas Gray – AFNIC
Dr. Chester Harris - AFMC

7 Feb 2012
Overview

- Problem Statement
- Background
- ERP Performance Study Lessons Learned
- AFNIC Network Validation Service Offerings
- Software Developers Forum Wiki
Multiple organizations are involved with the delivery of mission critical enterprise capabilities:

1. Application PMO
2. DISA & GCSS Computing Services
3. DISA Network Services
4. AFSPC NOSCs
5. Base-level client operations & AFECMO

These independent organization activities cause disjointed execution of:

1. Capacity planning
2. Escalation procedures
3. Change control
4. Help desk

Resulting in:

1. An incongruent End-to-End transactional path architecture
2. Inability to create a accurate application performance baselines on the Operational Network

Impacting users with:

1. Long response time due to bottlenecks
2. Unclear service level expectations
3. Frequent outages from uncoordinated changes
4. Long restoral times when diagnosis involves multiple organizations
SAF A6/CIO/CTO sponsored an AFNIC led End-to-End Network Performance Test using ECSS

Testing was conducted from May – July 2011
1. Detailed description of End-to-End transactional path is necessary

2. Establish an independent performance baseline for both application and network

3. Application performance should be measured across the entire transactional path

4. Client/Application communications directly impact end-user response time

5. Configuration Management is the key to optimal performance and availability
   (14 anomalies within a 12 month period)
Lesson Learned: Detailed description of End-to-End transactional path is necessary

Recommendations:

Program Office

- Request End-to-End transactional path architecture in the SLA from service providers

AFNet

- Provide program offices with a detailed architecture of all major AF infrastructure components within the application’s transactional path
#2 Establish a Performance Baseline

Lesson Learned: Independent performance baselines for the application and network are critical to resolving operational issues.

Recommendations:
- **Program Office**
  - Create and document baselines of the “no-WAN” User and WAN User response times.

<table>
<thead>
<tr>
<th>Change in Baseline</th>
<th>No-WAN</th>
<th>WAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change Agent</td>
<td>Application</td>
<td>Network</td>
</tr>
</tbody>
</table>

**AFNet**
- Create and document baselines for network latency across the End-to-End transactional path
  - End-to-End = AF user locations to AF hosting locations.
No-WAN and WAN User Response Time Baselines

**No-WAN Transactional Path** – User connected directly to the operational web application server

**WAN User Transactional Path** – Operational End-to-End traffic flow from user to web application server

**Best Case Scenario**

**End User Experience**

Legend:
- TCP/IP Interface (Segment)
- Minimal ERP Traffic Flow
- SSL Session

Cyberspace Without Limits
Each 4 hour summary is the average of 16 user sessions each with 95 browser-based requests.

- User sessions last 10s
  - 8s in host
  - 2s in network
- Browser requests average 105ms (10s ÷ 95 turns)
  - 85ms service (server & DB) time
  - 20ms network latency

These local testing results would *mislead* you to focus on application service time.
Base Proxies Obscure Network vs System Time when they act as the data center boundary to the base clients. WAN time appears to be inside the data center.

With a AF/GCSS/DISA WAN

- User sessions last 30s
  - 9s in host
  - 19s transitioning the network
  - 0.6s transferring 250KB
- Browser requests average 300ms
  - 90ms service (server & DB) time
  - 200ms network latency
  - 400Kbps effective throughput

- For this case, User responses spend over twice as much time in the network as in the system
- Managing page requests is the primary way of reducing response time.
Lesson Learned: Application performance monitoring needs to be incorporated into development (develop SLAs with hosting/service providers)

Recommendation:
Program Office: Monitoring Requirements need to be documented in SLAs with service providers
- Identify metrics that may impact application
  - End-to-End Latency
  - Error Rates
- How will the service/application KPPs be measured?
  - End user response (Goal/Threshold)
  - Availability (Goal/Threshold)

AFNet: Work with the Program Offices to ensure network impacts have been incorporated into KPPs
Lesson Learned: Client/Application communications directly impact end-user response time

Recommendations:

Program Office
- Download .jar files when they change
- Reduce the overall number of transactions
- Compress data

AFNet
- Balance AFNet operational mission requirements against simplifying the End-to-End transactional path

Current Efforts: SAF A6/CTO is leading an effort to simplify the ERP End-to-End transactional path as part of the CEITB Target Baseline
Response Time Lessons from ECSS

Working with a “Simplified” WAN

- User sessions last 18s
  - 9s in host
  - 7.5s transitioning the network
  - 0.6s transferring 250KB

- Browser requests average 190ms
  - 90ms service (server & DB) time
  - 83ms network latency
  - 400Kbps effective throughput

- Even on a streamlined network, user responses spend almost as much time in the network as in the system
- Reducing the number of page requests may be the easiest way to reduce response time
When a user asked for a page, the page usually asked for pictures, etc. separately, generating multiple network turns.

User Response time is the sum of the time to download the separate components for the page.

Fewer and smaller components = shorter user response times.
Performance Golden Rule: Only 10-20% of the end user response time is spent downloading the HTML document. The other 80-90% is spent downloading all the components in the page.
1. Cache static content at the client
   - Speeds access for your frequent users
   - Use Cache-Control:max-age=nnn or Expires headers to minimize reloads & update checks
   - Focus on image and JAR file caching
   - Separate style sheets & scripts to create cache-able “files”

2. Compress dynamic content at the data center
   - Reduces page “weight” (and user wait) up to 70%
   - Use Content-Encoding:gzip to ask the client to accept compressed input
   - Don’t compress images and .pdf’s ... it is already done
   - Do compress HTML, scripts, and style sheets.

3. Now you can work “inside” the application and DB
   ECSS realized a 35% reduction in end user response time through Caching
Lesson Learned: Configuration Management is the key to optimal performance and availability (14 anomalies within a 12 month period)

Recommendations:
Program Office
- Establish internal CM plan
- Establish external CM plans with service providers

AFNet
- Inform program offices of changes
- Work with program offices to determine impact to end users

Current Efforts: SAF Ops Baseline Working Group (AFSPC A6OI) is establishing the “AFNET Approval Process” responsible for overall configuration of the Ops Baseline
ECSS Transactional Path Component Owners

Cyberspace Without Limits
Application Performance Management (APM)

- Provide AF focal point for triage/ troubleshooting efforts supporting the integration of AF systems, applications, and services into the Air Force Network (AFNet), DISA and other AF approved hosting environments

Dynamic Network Analysis Division

- Provide performance analysis on developed software
- Analyze packet and response time delay
- The software must be installed on the operational AFNET or emulated AFNET environment
Future Application Performance Management Lessons Learned will be posted to a link off the SDF wiki

SDF Wiki link:

https://www.milsuite.mil/wiki/AFNIC_Software_Development_Forum