

Enterprise Application Monitoring with

11/10/2007

Presented by James Peel

james.peel@altinity.com / www.altinity.com

opsview

Who am I?

- James Peel - james.peel@altinity.com
- Job: Managing Director of Altinity
 - Pre-sales and implementation
 - Introducing new bugs into Opsview
 - Running Altinity
- Luggage: Munich airport

What is Opsview?

- Altinity's monitoring software
- Abstraction layer above Nagios 'engine'
- Provides a framework for adding new monitoring functionality
- Released under GPL v2

opsview

<http://opsview.org/>

Objectives

- Talk through our approach to application monitoring
- Flag issues and discuss possible solutions
- Talk about some customer examples

“Network Monitoring”

Traditional approach...

- Monitor network infrastructure
- Monitor servers
- Monitor network services - DNS, HTTP, SMTP, etc

Thoughts

- There are a thousand tools that can do “network monitoring”
- Just a pre-requisite for monitoring critical applications
- Network monitoring is about technology. Increasingly organisations want business oriented view of their IT systems

Application Monitoring?

- Enterprise applications: eCommerce sites, claims processing, CRM, document management, etc
- Sit across multiple pieces of infrastructure
- Often developed as web applications

Application Monitoring

- Common elements -

Web Server

Presentation Layer

Monitoring requirements:

- Availability
- Performance
- Resources
- Events

Application Server

Business Logic

Monitoring requirements:

- Availability
- Performance
- Resources
- Events

Database Server

A persistent store of application data

Monitoring requirements:

- Availability
- Performance
- Resources
- Events

OS / Hardware

“Platform” for application

Monitoring requirements:

- Availability
- Performance
- Resources
- Events

Network Infrastructure

- Content Switch (load balancer)
- Firewall
- External communication

Summary

- Standard monitors used in standard ways
- Don't actually need to know what application does

Challenges

- Deciding what *not* to monitor
- Dealing with clusters...

Challenges - Cluster Monitoring

- Active / active clusters
 - Monitor individual elements to ensure operation
 - Represent clustered applications as host in Nagios
 - Monitor cluster via “VIP”

Challenges - Cluster Monitoring

- Active / passive clusters
 - Basic monitoring of individual elements
 - Use negate plugin to confirm active node is offline
 - Monitor application via “VIP” if possible

Challenges - Cluster Monitoring

Longer term approach

- use API to reflect changes to cluster in monitoring

Application Monitoring

- Application specific elements -

Database Instances

- Database schema
- Batch jobs
- Processing queues
- Data import / export

Application Software

- Deployment status
- Error queues
- Threads, memory pools, instances
- Application log files

Summary

- Standard monitors used in very specific ways
- Need detailed understanding of application

Challenges

- Ensuring custom monitors don't place unnecessary load on application
- Getting information from relevant parties
- Building support for monitoring into application design

Application Monitoring

- Synthetic Transactions -

End-to-end monitoring

- Objective is to confirm entire system is working
- Provides a user centric view
- Are we getting expected result?
- Did transaction complete within expected timeframe?

“Web applications”

Designed for use by people!

- Session based authentication
- Layout is app specific and can change
- Objective is to confirm entire system is working

“Web applications”

Example synthetic transaction plugin

1. Authenticate with website
2. Check account balance
3. Get first three market IDs
4. Get information for market IDs

Web services / APIs

Designed for use by machines!

- Objective is to confirm entire system is working
- Are we getting expected result?
- Did transaction complete within expected timeframe?

Much easier :)

Application Monitoring

- System event logs -

System event logs

- Very popular!
- *Lots* of useful data
- Separate logs for:

operating system, database server, database instances, batch processing jobs, data import / export, application server, application instances, application messaging, web server, web sites, content switch, firewall, router.

Log file monitoring

- Common formats (Log4J / syslog)
- Challenges:
 - How do we identify “interesting” entries?
 - Volume of new log entries often high
 - How much information is too much?
 - Correlation between events

Log file monitoring

Altinity Approach - Log Daemon

- Set definition for log file type
- Create 'rules' file to match log entries
- Pass data back to server via NRPE

Log file monitoring

Altinity Approach - Lessons learned

- Start by focussing on high priority events, don't try and alert on everything
- “Correlation” achieved by using specific logs to alert on specific events
- We don't currently try and use rules to process multiple log events

Questions?

Don't worry, there is still more to come...

Application Monitoring

- We're monitoring our application, so what now? -

Displaying status information

- Network oriented view for technical support teams
- Pretty green lights for CEO
- Business oriented view for everyone else

Hierarchical View

- Provides “10,000 meter” view of network
- Allows engineer to drill down exposing more detail
- Differentiates between handled and unhandled events

Hierarchical View

» Altinity

Status Summary For Hostgroup Altinity

Host Group	Host Status Totals		Service Status Totals	
	Handled	Unhandled	Handled	Unhandled
<u>Altinity Development Systems</u>	13 UP 3 DOWN	8 DOWN 2 UNREACHABLE	46 OK 2 WARNING 29 CRITICAL	
<u>Altinity Production Systems</u>	36 UP		164 OK	
<u>Third Party Systems</u>	4 UP		4 OK	
<u>VPN</u>	6 UP		11 OK	

Keyword View

- Displays hosts / service checks based on keywords
- Great for application monitoring

Keyword View

Status for Altinity's Web Sites

Service	Host	Status	Status Information	Acknowledged
HTTP	docs.opsview.org	OK	OK - HTTP/1.1 302 Found - 0.058 second response time	
	downloads.opsview.org	OK	HTTP OK HTTP/1.1 200 OK - 5566 bytes in 0.062 seconds	
	trac.opsview.org	OK	HTTP OK HTTP/1.1 200 OK - 5169 bytes in 0.049 seconds	
	www.altinity.com	OK	HTTP OK HTTP/1.1 200 OK - 16297 bytes in 0.278 seconds	
	www.altinity.org	OK	HTTP OK HTTP/1.0 200 OK - 112479 bytes in 3.624 seconds	
	www.opsview.org	OK	HTTP OK HTTP/1.1 200 OK - 11018 bytes in 0.027 seconds	

Reporting

Opsview uses three databases

- Configuration database
- Runtime database - NDOUtils
- Data Warehouse (ODW)

Reporting

Challenges

- Figuring out correct table structure for ODW
- Aggregating data into suitable form for reporting
- Calculating table sizes and storage requirements

Reporting

- Opsview Reports: Uses BIRT
- Business Objects

Choosing right approach

- Organic vs Process Driven -

Organic approach to monitoring

- Deploy servers and Opsview software
- Add hosts to system with basic monitoring (ping)
- Start figuring out what to monitor for each host
- Refine, review, extend, repeat

Issues with organic approach

- Hard to accurately predict project timescales
- Often hit unexpected issues mid way through
- Easy to end up with inconsistencies in monitoring configuration

Process driven

- Develop monitoring catalogue
- Refine and review catalogue until complete
- Use catalogue to implement monitoring system
- Take advantage of customer's change control processes
- Consider a parallel QA system for testing changes

Monitoring Catalogue?

- Defines monitors - input parameters, expected output
- Defines templates
- Defines notification profiles
- Used for validating system when implemented

Advantages

For application monitoring...

- Allows custom monitoring requirements to be established in-advance
- Flags any gaps in knowledge
- Helps avoid requirements creep
- Saves time during implementation

Customer Insights

- What worked and what didn't -

Irish Revenue

- Security model - node in each zone
- Synthetic transaction may take 10 mins to complete
- Prod and DR environments
- Moving target because not live

“Health Insurance Company”

- Had to reverse Opsview master / slave comms
- Integration with events management system on IBM mainframe
- Very specific application monitors

“Online Betting Exchange”

- Scale of their system
- Moving to < 1 minute monitoring interval
- Opsview API developed for their test and QA environment

Questions?

Don't panic, that really is the end...

Contact Information

James Peel

E: james.peel@altinity.com

T: +44 (0)870 787 9243

- Opsview: <http://opsview.org/>
- Altinity: <http://altinity.com/>