

Simple Network Operations Center Project Summary

by Mike Conneen
\$Revision: 1.1 \$

Table of contents

1 Opportunity Statement.....	2
2 Domain Analysis.....	2
3 Stand Alone Monitor.....	3
4 Notification Layer Approach.....	3

1. Opportunity Statement

Information Integrators Inc. builds and deploys various web application and web service frameworks. We have always focused on the business domain. However, there are situations where we participate in the hosting and management of such services. To date, we have no means of monitoring the health of our web applications. To that extent, other than using JUnit and client supplied regression testing tools, we have no means to regression test our applications either.

As a few examples, we are responsible for managing the following...

- SheddableShell
A small e-comm site that we have a services and residual agreement with.
- St. Joseph School
- II Wiki
- II Scarab

From an application management perspective, basic needs are:

- Ping Alive
Is the box up, if not.. e-mail or page someone
- HTTP Alive
Is the HTTP request alive, if not.. e-mail or page someone
- HTTP Responsive
Is the HTTP request responding within a threshold of time..
- Did a schedule process run, such as backups.
-

Another item that should be considered is a recurring down count. For example, if the monitor case runs every 3 minutes, and the web site has been down for 5 occurrences, then it should probably STOP massaging the list as to not flood a wireless phone or e-mail account.

From an application quality assurance perspective, basic needs are:

- Unit test case - doGet, evaluate response
- Unit test case - doPost, evaluate response
- Web Services ?????

2. Domain Analysis

There are several ENOC type open source solutions available. If our core competency is to be operations management, then a robust solution is required. However, if we simply need to offer a base solution to ensure that our services are running and healthy, then we may be able

to wrap one of the unit testing frameworks available to satisfy both needs.

Applications within the domain are..

Domain	Solution Description
NOC	MRTG This is the leading product for traffic analysis. There are perl add-ons that can also measure the health and responsiveness of various HTTP requests.
NOC	MARS - Monitoring Application for Resources and Servers
NOC	http://www.ipmonitor.com
Unit Testing	jakarta-commons/Latka
Unit Testing	Anteater
Other	We could use jakarta-commons/jelly along with jakarta-commons-sandbox/messenger or our own services/notification framework and build our own framework which satisfies our requirements.

Note:

TO DO: document pros cons of leading approaches.

3. Stand Alone Monitor

4. Notification Layer Approach

If we wrap, extend, or build our own set of services, following is a conceptual view

In the above diagram, the MonitorTestCase is responsible for executing a monitor test suite against a URL and publishing the results of the suite to a messaging layer. An EventListener is an object that has subscribed to the messaging layer thus registering an interest in a specific event.

Pros	Cons
<ul style="list-style-type: none"> Clean separation of duties Can schedule and script using existing frameworks, such as JUnit, Latka, etc. 	<ul style="list-style-type: none"> Overly complex for monitoring framework.. what if messenger goes down?

<ul style="list-style-type: none">• Scheduler and Listener can be on different machines, networks, etc.	
---	--