

# 23-Oct-2012 Meeting Agenda and Notes

**Meeting Details:** Tuesdays, 4:00 - 5:00 pm Eastern Time, 1:00 - 2:00 Pacific Time

**Dial-in numbers:**

+1-734-615-7474 (Please use if you do not pay for Long Distance),

+1-866-411-0013 (toll free US/Canada Only)

**Access code:** 0150432#

**Participants**

| Who              | With                    | Attending |
|------------------|-------------------------|-----------|
| Rob Carter       | Duke University         |           |
| Celeste Copeland | Univ. of North Carolina |           |
| Warren Curry     | U Florida               |           |
| Michele Decker   | U of Notre Dame         |           |
| Tom Dopirak      | CMU                     |           |
| Keith Hazelton   | UW-Madison / Internet2  |           |
| Steve Olshansky  | Internet2               |           |
| Derek Owen       | U of Notre Dame         |           |
| Andrew Petro     | Unicon                  |           |
| Chris Phillips   | Canarie, CA             |           |
| Gary Sharpe      | UC Davis                |           |
| Bill Thompson    | Unicon / Jasig          |           |

**AGENDA**

1. UC Davis update from Gary Sharpe
2. Event-driven message-based P&I solutions with ActiveMQ
  - See <https://spaces.at.internet2.edu/display/Grouper/Grouper+ActiveMQ+integration>
  - and Chris Hyzer's email on this included below
  - and Jim Fox's email (also below) on using [Amazon SNS/SQS](#) in preference to ActiveMQ
  - This email thread continues to grow with interesting posts: Subject: "Re: [grouper-dev] grouper ActiveMQ integration"
  - Keith found this blog post enlightening on SNS/SQS: <http://bit.ly/pl55Yr>
  - TomD: Oracle ESB with in-memory DB at Cornell
  - KeithH: Some see CIPHER as a mix of
    - Installable packages (Identity Registry, either Open Registry or Central Person Registry)
    - Collection of Recipes, Toolkits and Demo Solutions
    - Which will CIPHER P&I be an installable package
3. CIPHER API work and implications for P&I
  - See <https://spaces.at.internet2.edu/display/cifer/API>
  - In particular <https://spaces.at.internet2.edu/display/cifer/API>
  - and, from Jim Fox at U Dub: <https://spaces.at.internet2.edu/display/cifer/Authorization+Standard+API+Use+Cases>
  -
4. P&I Three and six month milestones and goals
  - Discussion starter on CIPHER P&I three and six months goals and milestones
  - By December 31, 2012
    - Use cases for demo solutions fully documented; Toolkits identified
      - BillT: Rutgers uses Camel for provisioning from OR...
      - Clemson
      - Univ. of Florida
      - UNC
      - Notre Dame
    - Invite some of the above to present at a CIPHER P&I venue in screen capture mode
      - BillT, Unicon, has Adobe Connect capability
      - RobC, Duke, has WebEx capability
    - P&I models documented and illustrated by reference to above
      - See RobCs slides from August Jasig CIPHER talk
  - By March 31, 2013
    - Demo solutions implemented and documented including recipes, tool choices and code snippets
    - Detailed roadmap for P&I deliverables over following 12-18 months

Action items:

(KeithH) Invite Clemson to a call to dive into their P&I solution

(KeithH) Contact BennO re call-in information for CIPHER-API conference calls

(RobC) Contact Gary Sharpe for more details on P&I work at UC Davis

### Grouper ActiveMQ integration

See email thread "grouper ActiveMQ integration"

Here is something I have been working on, let me know any thoughts, perhaps we could discuss at the dev call on wed. Anyone else interested in using this?

<https://spaces.at.internet2.edu/display/Grouper/Grouper+ActiveMQ+integration>

BTW, I hope Penn and Penn State (and Brown?) can document some lessons learned on general ActiveMQ configuration / loadTesting / configuration / etc.

ActiveMQ is an open source application messaging middleware service where applications can send messages to other applications (or groups of consumers) in real-time without polling.

Grouper can be used to manage ActiveMQ authorizations. University of Pennsylvania and Penn State are interested in this component.

ActiveMQ users can send/receive messages to topics/queues, and also can dynamically create topics/queues. Grouper permissions can be exported and kept in sync (cron'ed and real-time) with a local permissions store. An ActiveMQ plugin reads the local permissions and enforces the rules.

The Grouper-ActiveMQ component is a jar that you can add to the ActiveMQ system (along with a few other jars), and some configuration that allows authorization managed from Grouper.

Thanks,  
Chris

## Amazon SNS/SQS vis-a-vis ActiveMQ

At UW we've used ActiveMQ for group change notices for several years. It has generally worked well enough, but for many reasons all of our new group messaging activities are using Amazon's SNS/SQS messaging.

ActiveMQ is expensive. You have to get and maintain hardware, install and document the software, monitor it, be on-call during the night, administer it. It might be open source, but it isn't free.

If you are lucky enough to have an existing ActiveMQ service operated by another department, and they are willing to let use use it mostly for free (our situation), hope they will continue to give you wonderful service after they have migrated their own stuff to RabbitMQ, or OpenMQ, or some other MQ and you're the only one left on the ActiveMQ system.

Connections can hang. We use the CMS client, for c++ with an interface library for plain c. (You know there's trouble with an interface when the first thing you absolutely need is to find an interface package for your language.) This interface uses persistent connections and certain conditions on the server will cause these connections to hang. This is a difficult situation to monitor. Generally an external process has to send test messages and report when none has been processed for some timeout period.

Amazon's messaging service, by comparison, is nearly free, is exceptionally easy to use (there are client libraries for many languages, including both our present favorites: java and python, but I had no trouble writing a c client from scratch), easy to configure (a web client make most administration self-service), and is reliable. It is also RESTful.

We have, as do most universities nowadays, initiatives to use cloud services in lieu of a 'running everything ourselves' mentality. Our people are not so fearful of losing control of even core services. It is the future.

There are a couple of advantages ActiveMQ has over Amazon SNS.

ActiveMQ is an order of magnitude faster than SNS. Is that important? For our network team, monitoring thousands of routers and whatnot, and sending millions of messages per day, it is important. For our group service, not so much. Messaging for course enrollment, operating since summer, now totals about 300,000 adds or drops. All the messages get to our consumers in a few seconds. And a few seconds is a short time for this application.

ActiveMQ always delivers messages in order (I think). With SNS/SQS they can get out of order. Something to deal with.

Jim

---

Some questions regarding SQS...

1. Is there a way to do a topic as opposed to a queue? i.e. I want to clear an authz cache which is on 10 app servers. I want to send the message to one topic, and whoever is subscribed will get the message.
2. I don't really like the polling. I want to send a message and the receiver will get it in real time. With SQS I have to poll every minute, and even then, it might not be ready and it will be there (hopefully) the next time after that...
3. The polling doesn't seem like it scales... if every need to get a message is a poll (every minute? every 15 seconds?) then that are a lot of SSL web service requests as opposed to a socket which is just waiting for traffic. If you have a lot of apps on a machine or a cluster it seems like this polling can add up.
4. Whoops, I think the last time I estimated the cost of SQS I was calculating polling every second, which is expensive. But I think for one endpoint to poll SQS every minute, it costs 50 cents per year 😊 ok, that is pretty cheap... 😊

Thanks,  
Chris