

Provisioning MACE Grouper 1.1 vs. 1.2

This summary was posted to the grouper-user and grouper-dev email lists.

At Brown this week, we've successfully provisioned our 11,000 groups into MACE Grouper 1.1 and 1.2 in our QA environment. I've been promising some numbers on performance for some time, and we have some interesting preliminary figures to share. We have not drilled down yet to determine exactly where major bottlenecks are, but at this early stage, we're pleased with the performance for what we've seen. Additionally, our figures give some high level performance metrics differences between Grouper 1.1 and 1.2 rc1.

1. Environment

Our environment in QA is identical to our production environment, with 1 server dedicated to the Oracle DB, and 1 server dedicated to the application & provisioning server. The only difference is that the production DB server has an HBA card for SAN connectivity, which will help with DB I/O. Development is significantly less robust--in development, Oracle and the app server run on the same 3 year old single CPU server with 1GB of RAM. Production and QA specs look like this:

Hardware:

- Sun SunFire X4100 M2 x64 Servers
- Four 2.4Ghz/1Mb processors
- 4Gb RAM (DDR2-667)
- Two 73Gb 10K RPM SAS drives (hardware mirrored)
- 1Gb/full duplex network

Software:

DB server:

- RedHat Linux Enterprise Server 3
- Oracle 10g pretty much straight out of the box

App server:

- RedHat Linux Enterprise Server 3
- Apache 2.2.4
- apr and apr-util with ldaps support
- mod_ssl for https support in apache
- mod_authnz_ldap for apache authentication via Brown's ldap(s) server
- Tomcat 5.5.20
- Sun JDK 1.5.0_10
- mod_jk 1.2.21
- ant 1.7.0
- MACE Grouper 1.1 or 1.2, at various times. Running 1.2 rc1 now.

2. The task

Brown's existing group management tool (aka Brown Grouper) currently manages 11,000 groups, with 8,500 groups as members of other groups, and a total of 370,000 person memberships in all groups. Brown has 18,000 users in our SunOne LDAP registry. The current task is to script the one-time provisioning effort of moving all existing Brown Grouper memberships to the MACE Grouper database. On an ongoing basis, there will be a differential provisioning run every night to update provisioned group changes, but this will be a small fraction of the total, except during registration periods 2 or 3 times per year. Our script currently only handles stem creation and group memberships, but we are incorporating ACLs next, and this is expected to increase run time to some degree. We also have some fine tuning of the provisioning script to better handle edge cases of complicated group math we currently use in Brown Grouper. We will be working to improve performance of the script and the overall system at the same time.

3. Results

Grouper 1.1:

We first ran the provisioning script in QA on Grouper 1.1, and the results were better than our days-long tests in development, but not yet acceptable. We successfully provisioned the 11,000 groups in MACE Grouper in about 6 hours.

Grouper 1.2:

Last night, we ran the provisioning script in QA on a clean installation of Grouper 1.2 rc1, with a dramatic improvement just in the new release. We successfully provisioned the 11,000 groups in MACE Grouper 1.2 in about 3.5 hours.

From briefly inspecting the systems during the provisioning run, there was a mild to moderate iowait bottleneck on the Oracle server, although we have more work to do to identify the direct cause of the bottleneck. We've done essentially no tuning of the Oracle system (yet), and no analysis of what queries are taking the longest (yet). We expect some potentially significant performance increase to come from database hardware and software configuration changes, along with some query optimization.

The web interface performance [speed] has been entirely acceptable on any of the systems we've used, even with a large data set. Faster systems did provide quicker web interface response, and of course pages displaying large membership lists were slower than others, but the slowest pages were acceptable. We have not load tested with multiple simultaneous users in the web interface.

One interesting lesson we've learned in our first cursory analysis of our data in MACE Grouper is that the web interface provides much greater visibility into the nature of our data. Having the ability to list an individual subject's memberships was not supported by Brown Grouper, and it is seen as a big win for the group that will manage the data. On the down side, MACE Grouper has revealed a number of undesirable issues in our data. There will be a significant data management project of unknown scope to come from the increased visibility MACE Grouper provides.

4. Next steps

These are the figures we have at this point. If you would find other information useful, let me know, and I can supplement. Our next tasks proceed on 2 fronts...technical and administrative. Technically, we are adding ACLs to our provisioning process to maintain existing rules in Brown Grouper that define who can view groups or list memberships, etc. Once that is done, we will get our daily differential provisioning script in place and start monitoring speed and accuracy over several weeks. We also have odds and ends in the MACE Grouper software to address, such as building a custom UI for at least 2 audiences. Administratively, we are interviewing our current group management staff with our full data set in MACE Grouper 1.2. We'll meet to go through common management tasks to identify any functional changes we may need to make in order to accomplish current needs. During this process, we will also be identifying any data issues we may want to clean up, and we'll be identifying the skill set we think will be needed to support this application going forward. Out of those administrative conversations we'll establish policy to govern the use of MACE Grouper in production.

I hope this helps!

James Cramton
Lead Programmer/Analyst
Brown University
jcramton@brown.edu
401 863-7324