7th Conference - Amsterdam - Performance BoF

Attendees:

Jason Shao - Rutgers
Max Whitney - NYU
Linda M Place - UMich
David Horwitz - Cape Town
Stephen Marquard - Cape Town
Leif Johansson - Stockholm
Klas Lindfors - Stockholm
Joakim Lundin - Stockholm
Tim Archer - CSU
Megan May - Sakai Central
Raad - Caret

Desires:
Functional testing – need to get test data that validly sets the systems up for functional tests
Performance testing – enough data to detect performance problems on QA servers
Load Testing – make it easier to set up load testing configurations that are similar to production systems in key metrics (e.g. number of users, number of sites, users/site)

messages
discussion forums
assignments

Testing profiles:

- small: couple 1K users
- medium: 20K -60K total users
- large: couple 100K users

"common environment" in this context refers to the data population of sakai install – the default starting point of users, sites and content.

Make the data population tool shrinkwrapped – so that various institutions can deploy on their own QA servers in order to provide testing results for as many configurations as possible.

Alan Berg (U of A) and Stephen Marquard have agreed to set up the skeletons in a very general way, then they will fill in details on things which are known problems. e.g many discussion board entries x many site members.

Stephen Marquard says Alan has offered to work on the scripting side of it – sash is mentioned as a scripting tool.

Max would like to see the common environment scripts be compartmentalized so that only those tools being used in a local installation can be populated with data. Maybe just a yes|no list of toolnames that says whether a specific tool is deployed in the given installation.

Jason:

Generating synthetic test data is very expensive

Q: is this load testing data, or is this better default sample data

Comment: Jason sees the value in load testing generic tools, but it doesn't make sense to set up a stock sakai build given own resources. Would like to be able to deploy common environment on local build.

Jason advocates setting tool standards within Sakai group, so that test scripts themselves can be shared – e.g. Grinder, JMeter, something open source.

Stephen Marquard:
JMeter, Selenium tend to be very specific in their script language, very hard to generalize. Idea of the setup scripts is to set up predictable data, so that these specific values are predictable, and the scripts can be re-used.

Tim Archer: If I use JMeter, I'll share those JMeter scripts. There will be *some* value there.

Linda Place gives some history. Load Testing is very very hard.

Stephen: Local issues are of course local – some unis have a problem with gross number of people, others have problems with number of members in a course site.

Feed the specific issues back to the QA group itself. Once the local uni's issues are well-articulated those specific issues can be fed back into the formal QA process. Permitting the problem to be discovered earlier in the process.

Linda makes a good point that issues identified post-QA certification are harder to get developer cycles on, whereas if identified before QA cert, more likely to get developer cycles.

Having common environment data will advance the goal of identifying performance bugs during QA, because the data will have more trust as common environment, rather than being (potentially) dismissed as only site specific.

Tim: Own systems group is refusing to provide more RAM (512K current) because they don't see why they should. Good Performance metrics will allow the argument to be taken to own systems group.

Linda: small group of dedicated volenteers - use collab for discussion.

Jason:

Linda: spreadsheet for matching observed load to scenarios.

Jason : sharing production data

Leif: need more concrete examples before committing resources

Stephen: need to flesh out profiles. Alan keen to do scripting. Need to capture representative production profiles.

Megan: use svn contrib for scripts

Leif: Identify areas of framework to identify fixes needed.