

# Priorities for Functional Enhancements

Unable to render {include} The included page could not be found.

## Combined Grouper-Signet Roadmap

This update to the roadmap reflects the discussion that occurred at the combined Signet-Grouper working group session at the Internet2 meeting in San Diego on Monday, October 8, 2007.

There are two lists, "A" and "B". Very roughly, these reflect larger (A) and smaller (B) tasks, but they might also be thought of as tasks on which work can proceed with relative independence from each other. Unless otherwise indicated, each task impacts work of both the Signet and Grouper projects.

The ordering of tasks below does not reflect priority or the sequence in which we'll approach the work. At the combined working group session a quick and dirty straw poll was taken of ordering of A list items. For interest's sake, the results, in order, were A1, A2, A3, A6, and a tie among the others.

### "A" List

A 1	<b>Extension hooks</b>	Implement infrastructure within Grouper & Signet to enable independent extension of key internal events. Pre- and post-processing hooks will be provided for each "primitive API operation".  This would make many other tasks more feasible.
A 2	<b>Notification of changes</b>	Propagate group, membership, or privilege changes to the infrastructure and consuming applications. Depends on A1.  Includes support for email notification of changes.  Addresses (near) realtime & incremental provisioning needs.
A 3	<b>Web service interface</b>	Build on XML import/export architecture to define functional services that could be wrapped as a Web service interaction (e.g., SOAP). Provide for wider integration of group & privilege services into applications and enterprise integration infrastructure. In particular, simpler or embedded UIs are made more feasible by reliance on this interface.
A 4	<b>Web service interface facades</b>	Determine which subsets of native API capabilities should be exposed through more focused end points to facilitate access by applications to Grouper- and Signet-provided access management capabilities. Also investigate how facades may be used to manage access to underlying group and privilege management and query capabilities.
A 5	<b>History &amp; audit</b>	Gain clarity on types of audit requirements to be supported by Grouper, and whether their support should be enabled per group, per naming stem, or categorically. Build on the hooks or change notification infrastructure of A1 and A2.  Develop an EDDY agent to send events into an external infrastructure that supplies persistence and analysis services.
A 6	<b>Rule-based actions</b>	For Signet, support automatic activation of privileges when prerequisites are met, and automatic revocation of privileges when conditions are not met, typically based on changes in a person's status and affiliation.  For Grouper, automatically deactivate or reactivate groups and memberships.
A 7	<b>I2MI integration</b>	1. Identify/implement a framework in which combinations of I2MI components (currently Grouper API, Grouper UI, Signet API, Signet UI, Ldappc, and Subject source adapters) can be easily integrated (not just in a single JVM). This is largely an issue of managing configuration and 3rd party libraries.  2. To the greatest extent possible, ensure that I2MI components use same means for achieving same ends, using the same configuration files for common purposes.  3. Provide a demo Subject database common to all quickstart or demo packages.
A 8	<b>XML import/export for metadata management and integration - Signet</b>	Complete the work that's underway. Includes utility wrappings to support command driven and scripted interactions.  Provide support for handling assignment & privileges changes that result from metadata changes.

### "B" List

B 1	<b>Namespace transition support - Grouper</b>	The hierarchy of naming stems in a deployment will change over time. Although the XML Import/Export tool supports prune & graft, for large changes that could take a while and be disruptive. Ability to logically "move" or "copy" a group or a selection of groups from one naming stem to another would be superior.
B 2	<b>Unresolvable Subject deletion utility</b>	Enable deletion of memberships or privileges held by subjects no longer resolvable in any subject source.
B 3	<b>Subject API v1.0</b>	Finalize the definition of v1.0 of the Subject API. Enhance the JNDI and JDBC reference implementations accordingly. Restructure how source adapters are configured and initialized to completely decouple the reference implementation from 3rd party implementations (especially, change the current role played by the SourceManager in this regard).
B 4	<b>Improve Ldappc</b>	Wait until the "Notification of changes" task is complete and then adapt Ldappc to propagate changes to groups and memberships. Provide configurable capability for assembling values of a privilege attribute.

[?](#) Questions or comments? [i](#) Contact us.