

Site Info Participant List Improvements DRAF

SAK-11368

When you add a roster, Site Info displays a list of all rosters you have access to. If the Registrar or someone else with high level access tries to add a roster, they get a list that's too long to work. I'm afraid the list needs to be paged.

SAK-13292

For sites with large number of participants, the Site Info tool takes too long to load. Need a search mechanism and perhaps paged queries. A couple comments from others -

- you get a long load time as it loads the users, but there is a very good chance that you don't want to manage users, rather you might want to manage tools
- there is no way to page or search through a long list of users - makes managing a large set of users very painful

SAK-11179

We have a site with 23000 students, and need to change the status of one to inactive. And since only 200 users are displayed at a time, to go through the list is taking a little long :)

Could we have a search button, or is there a reason why Viewing is limited to 200 whereas in other cases you can view all?

Some possible solutions (see <http://www-personal.umich.edu/~gsilver/sak/sak13292>)

- 1) Make loading participant list a conscious action. Link to Site Info displays Site Info (all the metadata associated with a site) - but not the participant list
- 2) Within that view have a link to the participant list.
- 3) Build some intelligence into the list. If below a certain number, display all list items, if above that number, chunk to the threshold, present chunking controls.
- 4) Chunking controls are:
 - a) pager (shown if context chunk > threshold)
 - b) alpha index (always shown)
 - c) search (always shown)
 - d) section toggle (shown if more than one section)
- 5) The following workflows need to be supported:
 - a) page through list, in variable chunks, set by user for session (as is now)
 - b) get an alpha chunk (ie. "a through c")
 - c) page through this alpha chunk if alpha chunk > threshold
 - d) search for string (q - what would the string search into?)
 - e) page through this search chunk if search chunk > threshold
 - f) view by sections
 - g) return to default view (first view of list) from either an alpha chunking or a search chunking or a section chunking
- 6) Challenges/questions
 - a) Chunking mechanism dependencies: there are 4 chunking mechanisms, some of them might operate within a chunk created by others, others do not. How to distinguish when this is the case, when not. The UI will need to make this clear.

For simplicity, numeric pager is the only one that operates within chunks created by other means (search, alpha paging, section selection), which will always be a chunking of all participants.

- b) alpha pager choices to be made
 - (1) prebaked alpha list (from language bundle) or
 - (2) list that reflects actual list membership (if there are no users with last name starting with R, do not show R in the alpha list, or show it inactive)

(1) seems the simpler choice, even though it may lead to "There are no participants in the x - z range" situations. (2) is the preferred solution but will involve an expensive collation between a list from the language bundle and the actual contents of the membership list.