Grouper dashboard and report

Wiki Home	**Grouper Release Announcements**	**Grouper Guides**	**Grouper Deployment Guide**	**Community Contributions**	**Internal Developer Resources**

This is a discussion about an item for a future Grouper release.

Due to the increased capabilities of Grouper (v2.5+) to support Access governance processes like attestation, in addition to lifecycle dimensions like Object Types, Services in the UI, enabled/disabled dates and visualization UI there are gaps in the functionality of the system to express the "general state" of a service to a "service owner" in a way to help them manage these characteristics of their service. Give that all of these individual characteristics need to be managed, and that they often also have interactions between them, as well as have impact across the distributed access control it becomes a bit complex for most service owners and general users to be able to understand the current state of the Service. The following additions to Grouper should help improve the management of these characteristics and hopefully start to form a set of governance model tools in general.

- Attestation interacting with Object Types
  The way an Object is manage in Grouper depends on several characteristics of the object. However the most fundamental of those is the Object type. All Object type values do require the same maintenance and management processes. Rather, the interval, formality of the process, and likely the individual(s)/group(s) that are involved in those processes vary by type and usage. So a "default" makes sense, but flexibility (down to the object in question) is also required.
    - General examples:
        - A "System of Record" group (or folder) in Grouper likely requires only automated synchronization on a frequent and regular basis. However it may also be desired to have an "independent" (possibly human based) review of the process during critical points in the calendar year, or when substantial changes occur in the data set.
        - A "Reference" group, of "Basis" group may only be maintained by humans, and could be institutionally important enough to warrant that any changes requires multiple people/groups to agree before an effective change in the groups definition is permitted.
        - While other "Reference" group, of "Basis" group may only be maintained by humans, might not be as institutionally important and thus not require as formal, frequent or multi-approver models for governance.
- Services interacting with Object types.
  The current model:
  To identify and manage services in Grouper is a Grouper attribute (type = service) and then a set of folders are marked as being part of that service by attaching the attribute to them.
  While that is somewhat helpful to define the "things in Grouper that are part of a service" it is currently not possible to intersect that concept with the additional governance characteristics like: Object type, attestation, and enabled/disabled dates.
- Enable/Disable interacting with Object Types/Services
    - The UI currently does not:
        - surface Enable/Disable information to the surface for the users
        - provide any way to view "en masse" this characteristic much less a way to find "upcoming(future)", "recent(near past)", nor specific windows(arbitrary time slices) views across Objects in the system
    - Grouper does not currently provide:
        - Global controls for "auto disable" (by Object Type/Service) on a schedule specific to the Object Type(general)/Service(more specific)/Object(most specific) configuration
        - "re-enable" processes from recent (AKA: "undo events"),
        - ad hoc (AKA: "user controlled time slices") with any context (folder, group, Service, etc...)
- Visualization interacting with "all the above"
    - The ability to capture more meta data and to be able to "attest" a Visualization is also needed to verify the meta state of "group math" over time as well.
    - Being able to auto trigger, or at least know when to re-verify the "group math" due to changes in the graph is also desired.

In an attempt to address some of these issues this set of features are envisioned:

1. By Service context (AKA: attribute markers that select objects into an arbitrary set to be processed as a logical unit)
    a. Note: It is expected that some objects would be in more than one context/set. And how they would need to be evaluated may be different for each set.
    Example: Service "A" may require all manually maintained reference groups to be reviewed every 10 days. And might become dependent upon a Reference group that has Attestation set to be reviewed every 50 days. That is the kind of discrepancies that should be surfaced with this kind of a governance review process.
2. A way to see (in the UI) and document the current state (a serialized report) a "summary of the context" for the following:
    a. Identify groups by raw (single value) type(s) : <type> Count=N
    b. Including identify groups without any type. : "NON_TYPED" Count=N
    c. Identify groups by type value combinations : "ref,readonly,service" Count=N
    d. Identify sets of groups/Memberships by attestation lifecycle (Due in the next N days, Was attested in the previous NN days, Past due by NNN days, etc...)
    e. Identify sets of groups/Memberships by enabled/disabled lifecycle (will be enabled/disabled in the next N days, Was enabled/disabled in the previous NN days)
    optionally include other meta data about the objects found (custom attributes)
    f. Most recent attestation date
    g. Next attestation due date
    h. Most recent date when each identified group's membership list was updated
    i. Most recent date when privileges were updated (by identified object)
    j. Most recent date when privileged Memberships were last updated
    k. Ability to "ignore"(do not use/show) meta data about objects by type in a report selection/document.
    Example#1: For a given Service report it would likely not be useful to include/"count"/"consider" when a System of Record group change

updated a manually maintained Reference group.
Example#2: For a System of Record report it might be required to include/"count"/"consider" the details for type="readonly,basis,ref" groups.

3. The configuration of a "UI/report" state should be stored in a reusable way to that the user does not need to "configure before using" the feature.
   Suggest a "set of sets" attributes that are assigned to the Service attribute marker. To allow more than one report option per context to be selected/scheduled.
4. Ability to schedule a "time based interval" for auto producing such reports with email notifications to a configured set of people for review.
5. Ability to manually request/trigger producing such reports with email notifications to a configured set of people for review.

NOTE: Some of these could also be applied directly at a group (type=policy, etc...) context too. Specifically with respect to Memberships Attestation, "Enable/Disable dates", custom attribute viewing/reporting.