TIER Entity Registry Update

The stakeholder community has expressed desire for the Entity Registry clarification and for the project communicate insight into the "Identity Registry" plans for TIER.

In reviewing the Identity Access Management (IAM) the Entity Registry Working Group and the data Structures and APIs Working Group have proposed a data Ecosystem required to support the TIER functions. A modernized Identity Access Management strategy that can create transformative change for Trust and Identity in Education and Research requires a complete, flexible and institutionally extensible set of data classes to support the function and processes of TIER components.

A data repository for Identity Registry functions and other Tier Architectural components requires a unified repository strategy to achieve optimal componentization and institutional flexibility. The repositories intent is to create a unified data concept that spans the institutional and inter-institutional functional silos. For the past couple of decades, many Higher Ed Institutions have implemented a "Thick" registry concept and data structure for their IAM.

In May 2016 the aforementioned TIER workgroups proposed and IAM Ecosystem defining three major data classes in a repository and a fourth supportive data class. Figure 1 shows a Repository cloud in the center of the diagram revealing this IAM data concept.
The three data classes defined include:

- **Entity Registry -** person and non-person objects that require access management functionality and interface to that functionality.
- **Entity Groups and Privileges –** Tier proposes through the Grouper Implementation Guide a useful strategy for defining groups of securable entities from the systems of record "Basis Groups", transformational abstractions into Reference Group forms then relating these to form Application related permission groups. See the TIER Grouper Implementation Guide for more details.
- **ODS, master data, data hub - unified** and normalized person or entity data concepts an institution elects to manage independent of IAM and useful to many line of business applications**.**

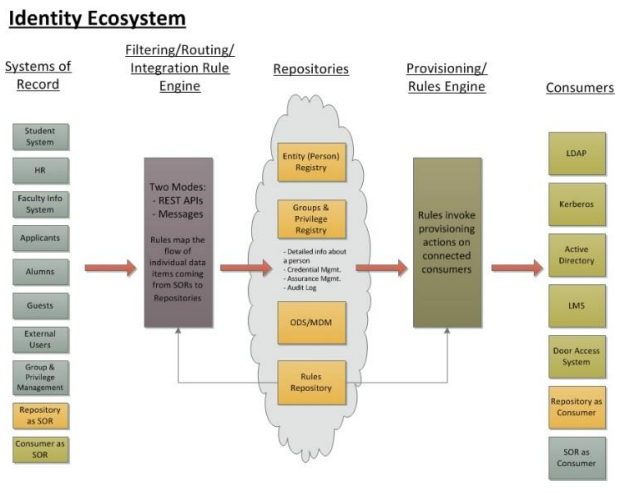

Figure 1 – Identity EcoSystem – TIER Entity Registry WorkGroup May 2016

A fourth class of repository info are rules for mappings of SOR (line of business systems) and transforms from the independent views into an institutional (normalized) view of the data across all SORs. This coupling allows the remainder of IAM and downstream provisioning components to view the data from an institutional point of view. This could be further abstracted to include the federated IAM capability as an institution desires.

The registry, group and person data classes can be as thick or thin as an institutions practice allows. The classes do allow for an evolution over a period of time. *TIER workgroup is recommending that the registry remain thin if possible and be limited to only those information that affect access management related service*. The grouping and provisioning structure will in turn keep track of Basis groups, Reference groups and Application groups as described in the TIER Grouper Deployment Guide. This style of thought could be employed even if Grouper is not used in lieu of another group management product. Over the past couple decades, many institutions have used their entity registry to store ODS / MDM / Data-Hub centric information. At the same time, they may have duplicated other IT teams working on similar data projects and duplicating effort. Think a minute about the notion of "IAM is simply another vertical application" with respect to those data hub efforts. IAM should source data to the hub and consume data from it just like other line of business applications.

Why propose a minimal/thin Registry?

- Avoid rebuilding an ODS or MDM structure that may already be in place or on the institutions strategic path. Use it as a component of the repository.
- Use of a common person (subject) data HUB that is available to application is becoming more prevalent. This can be leveraged by the IAM application as well.
- Agility and flexibility downstream.
- Reduce PII and other privacy implications in the registry.
- Isolate the access management info and the growing aspects federation from the more data rich environment of the data hub.
- IAM is a SOR for Access management info. So for same reason we have other vertical applications that use a common reference data the IAM Registry can be thought of in the same manner.
- Security - Access data only needs to be shared with those with a need to know. Exposure of data is less.
- Groups and Provisioning in an RBAC or ABAC model is better if driven from a Grouping /Provisioning tool (Grouper, Midpoint, etc)

Why not use a thick registry?

- Generally does not scale as well as thin designs
- Produces more data duplication and with other services (like ODS or data hub).
- Duplication of efforts and talent in you IT organization
- Efforts to build thick registries can create complexities and related operational problems.
- More risk based on projects undertaken in the past

Functionality is independent of these data structure specifics, thus the degree of thick or thin in data registry structure does not indicate the richness of the IAM instance. A better indicator revealed in the TIER architecture is the combined richness of the data repository available to IAM.

Entity Registry + Groups & Privilege + ODS/MDM HUB ➔
Achievable Functional Richness of an IAM implementation.

Function:

- Search/Match of individual entity
- Unique Institutional Identifier
- Multiple Affiliation and some richness in there implementation
- Robust Grouping and set manipulation
- Robust creation of Application entitlements/permission
- Provisioning / De-provisioning based on changing group memberships
- Account and credential management
- Loose coupling of functions allow a component based approach TIER or non-Tier components
- Restful API and/or Messaging standards for connecting components and for external sources and consumers of the IAM.
- Federated services and users

TIER Entity Registry Update

- Virtual Organization Support

Function Requirement documents registry and other components for a registry for person, institutional contact, client-service (thing calling an API), service/privileged account. Other parts of the Ecosystem are covered in this document as well.

The proposed structure does not require thin or thick registry. It does provide a path to evolve from the current institution operations to a method suitable to the institution maturity and other factors that affect the IAM footprint. A service oriented architecture can create the same effect so long as all three classes of data are in the repository.

See **Figure 2 – The TIER Architecture on page 4**

The TIER Architecture above reflect the classes of data and the loose coupling. The integration services pipe in the center of the diagram is the "glue or wiring" to bring all the component together in standardized methods. Restful API, Messaging Pub/Sub, and an administrative user interface based on API and messaging methods are used to bridge the gaps in the components.

It should be noted that many sources are consumers and consumers are also sources for information flowing in to the ecosystem and out of the ecosystem.

A look at the data needs of a minimal thin entity registry has concluded as shown below:

Entity Registry attributes: used for all type of entities

- Entity object ID
- Entity Type Code
- Date created
- Date Inactivated
- Entry Description / Name
- Status (suspect, merged, active, inactive)
- Institutional Entity Identifier
- ***Object Maintenance Fields*** *(can be used for any object of field)*
  - *Begin Time Stamp*
  - *End Time Stamp*
  - *Updating entity ID Identifies last updating Entity*
  - *Updating SOR Identifies last updating SOR*

Entity Type = person

- **Person object:**
  - Protect/Secured
  - Person Status (Active, Inactive and Pending)
  - Identifier (1-n)
    - Identifier Type Code requirement 11
    - Identifier ID
      - Institutional Id –required for all person objects
      - Net ID (almost always present)
      - Other possible extended types per institution needs- "ANY SOR ID" – such as for HR, SIS, BANNER, or ORCID, NSF, etc.
      - Access management identifiers- Federated login identifier(s), eppn or similar federated id of choice, Door Card ID, etc
  - Name object (Occurs 1-M)
    - Name Type
      - Legal (used as example other possible types Display, Preferred Former, Alias, etc)
    - First Name (Given1-4)
    - Middle Name
    - Last Name (Surname)
    - Prefix
    - Suffix
  - Contact Method Email Object
    - Email Type
    - Email Address
  - Contact Method Telephone Object
    - Telephone Number – Full number is stored
    - Country Code
    - Area Code
    - Telephone Number
    - Device Type
    - SMS Capable

Entity Type = client/service

- Client object: (a service or code that call and API/message)
  - Identifier ID
    - Institutional Id – required for all client objects
  - Contacts 1-m (who can be notified for any actions about this client)
    - Identifier - EPPN
    - Name - friendly name
    - Email -
  - Sponsor
    - identifier: a permanent (friendly) unique identifier
    - name: friendly name
    - sponsor: Id of the sponsor of this sponsor. (Null iff the root sponsor)
  - Service
    - identifier: a permanent (friendly) unique identifier
    - name: friendly name
    - description: Human readable description of this service.
    - sponsor: Id of the Sponsor of this service.
    - admins: list of administrators (eppns usually)
    - contacts: list of Contacts
    - base URL: host, port, base path
    - authns: list of authentication methods supported
    - authorization service: Service of OAuth authorization service (if OAuth supported)
  - Client
    - identifier: a permanent (friendly) unique identifier
    - name: friendly name
    - description: Human readable description of this service.
    - sponsor: Id of the Sponsor of this service.
    - admins: list of administrators (ePPNs usually)
    - contacts: list of Contacts

TIER Entity Registry Update

- redirect_urls: list of redirect_url (if OAuth supported)
- host: if known and constant.

- In addition the CSR maintains a long-term authentication credential to itself for each client in its registry. A service will generally authenticate to the CSR with its InCommon certificate.

Note: In addition to the active data regarding an entity the registry would maintain a complete Audit trail or similar mechanism is required to be able to review and look at all changes to data.

SCIM standard will be used to provide a standard for the restful design. It provides built in extensibility methods. You could expect the base SCIM, a TIER extension and if desired an institutional extension as needed for your implementation.

Web services can front or wrap the SCIM based components as needed to provide a flexible input for an institutions SORs. Likewise, a data repository can be interfaced by providing micro services behind the SCIM API to provide as much flexibility and robustness as needed to feed a component or to maintain or retrieve the repository data. The data can fill or sync into or out of any of the components of that repository based on the institutions need. The API approach provides piping within the ecosystem and for flows into (SOR sourcing) or out of the ecosystem (provisioning/deprovisioning).

SCIM provides a mean of extensibility and thus expect that TIER and Institution specific extensions will exist. An abbreviated version of the SCIM information is below. See the TIER API workgroup wiki for a more complete version of the User resource schema TIER is currently refining.

API components like Get Person, Maintain Person, Is Member Of and so on are being defined and developed. Several APIs are being distributed with the current Grouper install.

SCIM "User" Resource Schema

SCIM provides a resource type for "User" resources. The core schema for "User" is identified using the following schema URI: "urn:ietf:params:scim:schemas:core:2.0:User". The following attributes are representative of the core schema attributes:

4.1.1. Singular Attributes
  userName
  name
    familyName
    middleName
    honorificPrefix
    honorificSuffix
    displayName
    nickName
profileUrl
URIs
title
userType
preferredLanguage
delegated
locale
timezone
active
password
4.1.2. **Multi-Valued Attributes**
  The following multi-valued attributes are defined.
  emails
  phoneNumbers
    "display"
    "type"
  ims
    "type"
    "aim",
 photos
addresses
type
    streetAddress
    locality
    region
    postalCode
    country
  groups
    membership,
    role-based
  entitlements
  roles
  x509Certificates

4.3. Enterprise User Schema Extension - The following SCIM extension defines attributes commonly used in representing users that belong to, or act on behalf of, a business or institution. The enterprise User extension is identified using the following schema URI: "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User".

The following singular attributes are defined:
  employeeNumber, costCenter, organization, division, department, manager, etc.

Support for VOs and collaborations

A short note on support for VOs and collaborations with respect to Minimal Entity Registry Data follows below.

The TIER stakeholder requirements reflect that research activity at Higher Ed institutions need support for collaborations. These can broadly be segmented into two categories.

"Simple" collaboration needs cover researchers on campus (ie: those who have campus NetIDs) collaborating with others on campus via access to campus managed services (email lists, documentation spaces, etc). In general, this functionality can be provided with a combination of (existing) group and person registry services, often with a minimal "service enablement" layer to allow authorized individuals to define the collaboration groups and map them into enabled services.

"Advanced" collaboration needs expand to include researchers not affiliated with the campus (ie: those who would leverage federated identity to participate), complex enrollment procedures (invitation, self-signup, approval, etc), larger collaborations with delegation requirements, and finer grained service management. Meeting these needs often implies solutions like COmanage or extensions to include sponsored collections or groups in your grouping toolset.

For more information on the workgroup activities visit the TIER WORKGROUPS HOME PAGE in the Internet 2 wiki.

TIER Entity Registry Update

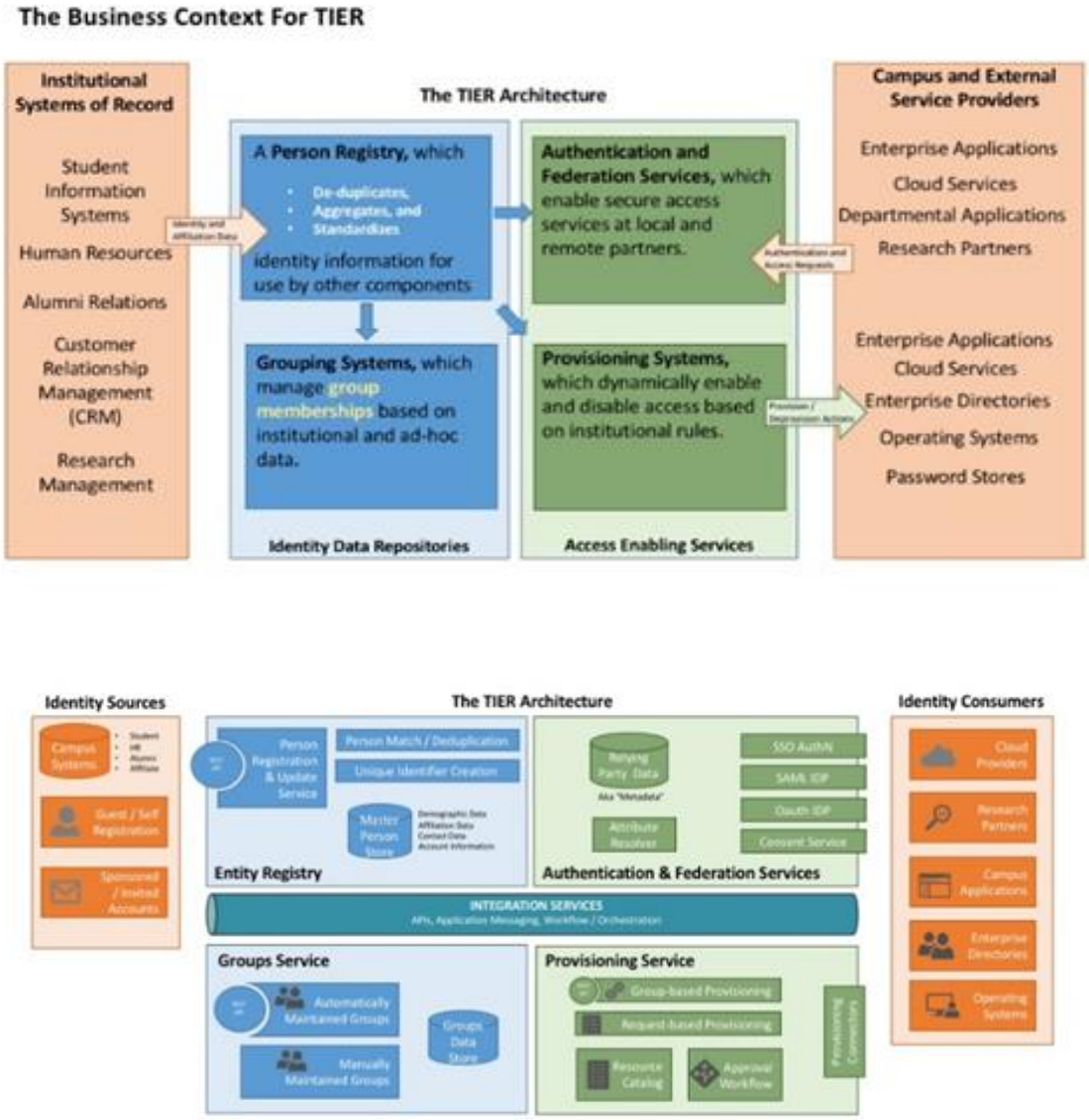

Figure 2 – The TIER Architecture