Identity Management System: Architecture

Table of Contents

1. Overview and Goals

The Identity Management (IdM) team at UNC Chapel Hill has identified a set of core requirements for a new system that will handle provisioning and de-provisioning in a timelier, more automated, and more efficient manner, as well as requirements for a user interface that will provide users with more effective and easier-to-use self-service options. These requirements are detailed in a separate set of documents.

This document details the high-level architectural design of the new system.

The system architecture should strive to meet these design goals:

- The system must be loosely coupled. That is, replacing or changing any one component of the system must not necessitate major changes to any other component of the system.
- The addition of new services to be provisioned and de-provisioned must require minimal modification to the system.
- The system should enable re-use of workflows and processes across services as much as possible.
- The system should use open protocols and standards (like SPML, DSML, LDAP, BPEL, and WS-*) as much as possible.
- We must support transitioning to the system as gradually as possible, and we must not lose any data during the transition process.
- The system should give service owners as much control as possible over the exact procedure for provisioning and de-provisioning those specific services. That is, the exact process for service provisioning should not be in the system directly, but rather be behind a service façade.
- If possible, the user-accessed UI and the system should be separate; as much as possible of the UI should be available through the current campus portal.
- The system should reduce duplication of data and of configuration, so that data and configuration are not stored in multiple systems unnecessarily.
- The system must be able to handle provisioning and de-provisioning services for applications, machines, and other types of identities as well as for person identities.

2. Design

Many of the design goals that IdM has for its new system are easily met if we choose to architect in a service-oriented fashion. Loose coupling, re-use, open protocols, and abstraction of implementation are all properties that services in a well-designed service-oriented architecture (SOA) possess.

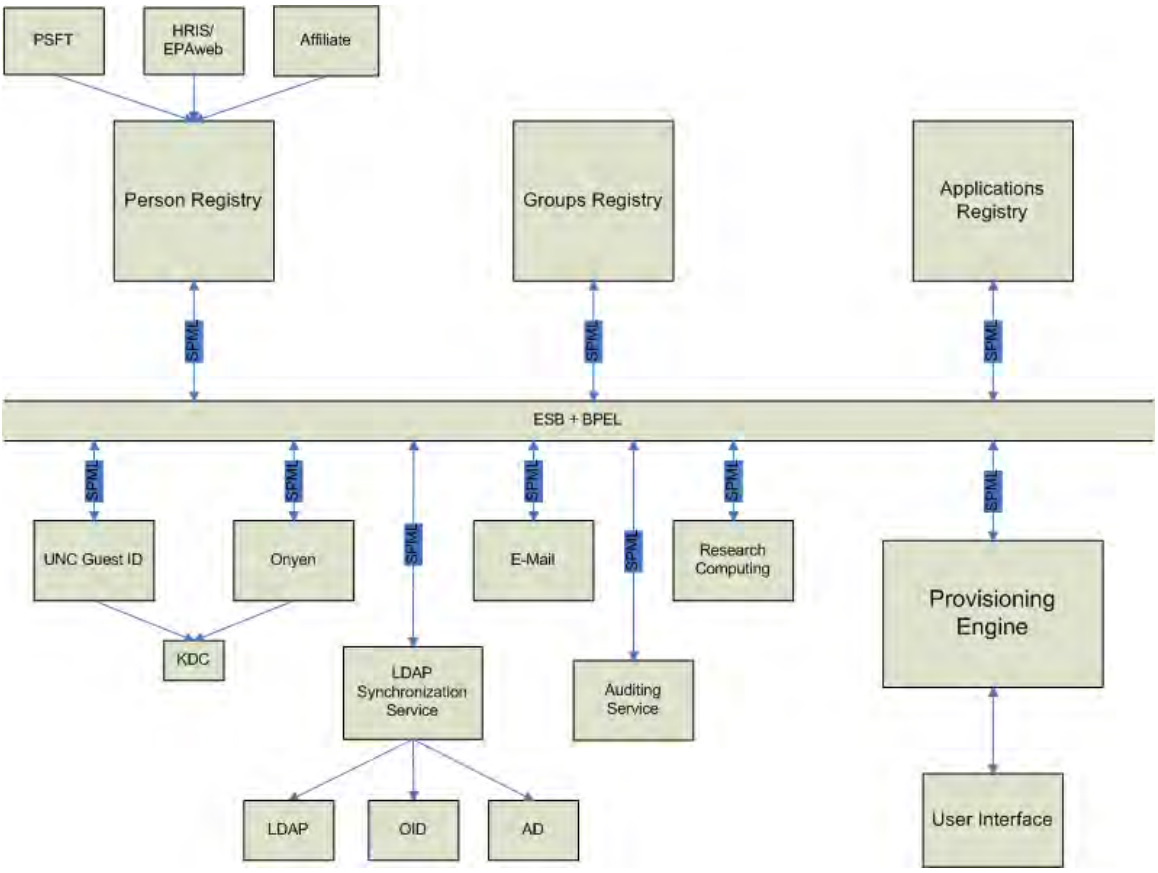
The new IdM system, therefore, will view each service to be provisioned and de-provisioned by the central system as a service in SOA terms. Each of these services should provide a SPML façade for the central provisioning service to call. All operations on a service are provided either by pre-defined SPML capabilities or by a small number of custom UNC capabilities that have been defined within the confines of the standard SPML capability framework.

The actual provisioning engine (the system that makes eligibility decisions, handles de-provisioning schedules, and coordinates communication with the user) can also be viewed as a service – it is the gateway through which external consumers, including the user interface, must enter.

Audit logging is a concern shared by most or all services. These are called crosscutting concerns in aspect-oriented programming (AOP) terminology. Rather than code audit logging into each and every service, this concern can easily be handled by BPEL processing, which can fill the role of AOP interceptors. This BPEL processing can direct copies of request/response messages to an auditing.

The new architecture is represented visually in Figure 1.

Figure 1


PSFT
HRIS/ EPAweb
Affiliate
Person Registry
Groups Registry
Applications Registry
SPML
ESB + BPEL
UNC Guest ID
Onyen
LDAP Synchronization Service
E-Mail
Auditing Service
Research Computing
Provisioning Engine
KDC
LDAP
OID
AD
User Interface

3. Scenarios

In order to see how we would use the architecture above in order to handle some real-world situations, we present some sample high-level message flows through the provisioning/de-provisioning system. All data flows refer to Figure 1.

3.1. Person-related scenarios

All changes to person data will be external to this architecture and will be reflected in the person registry. The person registry is responsible for publishing messages into the provisioning/de-provisioning system for processing.

3.1.1. Bio-demo information change

Bio-demo information will rarely be used in provisioning and eligibility decisions, but it is synchronized to the LDAP directories and it may be required data for some provisioned services.