

# Technical Breakout Notes

These are notes from our technical breakout. This page should serve to record ideas and action items. We will create separate pages for this after the meeting.

## Goals of the Breakout Session

1. Produce a list of use cases
2. High-level IT architecture design

## List of Use Cases

The purpose of these use cases are to gather requirements for EdUnify and illustrate how the service can be used by different categories of users.

1. Analysts looking for services to solve an integration problem by specific standard name.
  1. Exploratory capabilities inquiries - Web application designers looking for services to invoke at academic institutions to retrieve academic history, transcript, or other data. (Richard Moon)
2. Integration developers looking for services to solve an integration problem without knowledge of a standard, but just the business...like data field or business process names.
  1. Same as above, but without knowledge of existing standards. (Richard Moon)
3. Domain experts want to publish a new ontology to facilitate integration. Who are the domain experts we are trying to support? For example, EdUnify/PESC will be authoring ontologies. Anyone else? SIF, Department of Education, (Arnie Miles, Steve Wheat)
  1. EdUnify or PESC working group develops an ontology (object and behavioral model) for a domain and publish it.
  2. one or more government agencies
  3. one or more vendors
4. Someone who wants to publish a new service.
  1. Academic institution develops and registers a deployed academic history web service.
  2. Academic institution develops and registers a deployed course information web service (for use by course articulation services).
  3. State government develops and registers a deployed web service to send and receive transcripts. Universities, junior colleges, high schools, employers, and vendors also do this. (Note, do we want multiple use cases for this or address it all somehow in one?)
  4. System vendor develops and registers a web service capability for its student information system. (Note: this is not deployed, it is a deployable feature of the product)
  5. EDFACTS develops and deploys a web service to accept summarized data from state education agencies.
  6. A clearing house like NSC develops, deploys, and registers web services to send and receive data like transcript, health, or immunization data.
  7. FSA develops, deploys, and registers their services to receive and send financial aid data to and from academic institutions.
  8. Testing agencies develop and deploy web services to receive Pre-Id data from school districts and send test scores back to school districts.
5. Application looks up a services
  1. EdUnify's core services themselves (directory, ontology, feedback, etc.) will all be web services so anyone can implement applications that use them. For example EdUnify will provide registration, ontology, and feedback services, but any vendor, open source project, or anybody can implement these applications that conform to EdUnify's policies as well.
  2. Google-like semantic web crawling applications to index the web of services and ontological equivalencies for better use. EdUnify will provide some search facilities, but we should design to allow and expect anyone to build search and index utilities.
  3. Application searches for all services that meet certain criteria for research purposes...to determine availability of data. For example, a government agency looking for performance metrics looks for the availability of certain data and how much it will cost to get it.
  4. Notification services for new services, ontologies, applications for the semantic web, etc...

## High-level IT architecture design

1. Distributed, trusted service registry of services described with WSDL - make sure the service meta data is provided by legitimate users
2. Ontology service for managing standard vocabularies
3. Feedback, rating, and monitoring services
4. Architecture patterns and reference implementations. While no requirements will be introduced other than describing and registering services with WSDL, EdUnify will maintain best practices patterns that allow registrants and users to get the most out of EdUnify.
5. Will use SAML 2.0 for EdUnify core services where authentication and authorization is required

## Questions and Follow-ups

What registration artifacts will the directory accept...both technically and from a policy perspective? Will we have people register WSDL? Anything that could be registered in the directory technically? Like anything that UDDI would or the directory technology would support?

Higher education has a hard-won standard based on SAML 2.0 and, going forward, where authentication is required, we will look to leverage this. For example, higher education institutions have slowly, but consistently implemented federated authentication with SAML using frameworks like Shibboleth.

Study and review caGrid infrastructure for design, overlap, and potential reuse. These grid directory, ontology, security, and monitoring services were developed as part of government-funded research.

We should collaborate with Internet2 and semantic web experts, because many of these technology problems have already been solved.

We need to look at UDDI, EBXML, caGrid, NASA echo, supply chain/barcode example

Review <http://service-repository.com>

## Assumptions

We will describe the governance process to govern the policies of EdUnify. We need to provide some mediation of problems and data quality issues for EdUnify core service. Note that EdUnify is not responsible for mediation between users of services registered in the direction. As an example, EdUnify would need policies to mediate a feedback or monitoring dispute.

## Constraints

UDDI is something we should look at closely, but there has been limited success and in fact many failures in the implementation and use of UDDI beyond the scope of a single enterprise.