



The Enterprise Directory Implementation Roadmap

Introduction

The Enterprise Directory Implementation Roadmap is a collection of resources that campuses can use to assist in building enterprise directory services. In this Roadmap, we offer a sample implementation process that has been gleaned from work and experiences of many campuses, and couple it with technology, policy, and management information and resources. As noted by one of its reviewers:

"The Directory Roadmap is a generally useful introduction to concepts of large scale enterprise transformation, to institutional system and data architecture, and to the maturation of information technology in support of business processes in higher education. It is one of the most useful documents I have ever found for introducing technical folks to non-technical issues and vice versa."
- Michael Conlon, Director of Data Infrastructure for the University of Florida.

This Roadmap was developed through the efforts of the [Internet2 MACE-Dir](#) Working Group.

A Word about Identity and Access Management and Interoperability

When implementing a directory, the policy/process and technologies should be considered in a broader identity and access management context. Assembling the identity-related information from the systems of record, joining them together so that the data associated with an individual is listed together, then making that available to applications for authentication and authorization is what lies at the heart of an identity management infrastructure. This Directory Roadmap provides guidance for collecting, storing, and making this information available, but stops short of discussing authentication and authorization services. For more information, refer to the [Enterprise Authentication Implementation Roadmap](#) (HTML) and [Identity and Access Management and Security in Higher Education](#) (PDF).

Campus identity and access management infrastructures built with similar process, policy, and technical standards can be leveraged to enable your faculty, staff, and students to access applications and services managed by other institutions, consortia, government agencies, or vendors. Higher-education practices and tools, such as the eduPerson directory schema, provides a built-in, common corner of infrastructure that functions as a predictable platform to be used by federating technologies such as the [Shibboleth®](#) system. This Directory Roadmap also includes pointers to the emerging set of common architectures and practices needed to ensure interoperability.

How to Use the Directory Roadmap

This Roadmap is organized into policy/management and technology/architecture tracks as indicated to your left, and highlights the main points in each. Click on the boxes to review more detailed information, tools, articles, slide decks, and other resources to help you with your implementation.

General understanding of the functions of a directory, its use, and related concepts are assumed. Refer to [Introductory Resources for Lightweight Directory Access Protocol \(LDAP\) Directories](#) for background resources. For unfamiliar concepts or terms, we recommend visiting the Johns Hopkins University [Enterprise Services Glossary](#). In addition to the planning, design, data, and implementation materials on the Roadmap site, refer to the [Resource & Bibliography Collection](#) for a compendium of resources from each Roadmap section.

In general, readers should keep the following in mind:

- The institutional business needs must drive the directory project. The goal should be to deploy institutional directory-enabled applications and services, not to implement a directory.
- Much of the process of implementation is iterative. Even after deploying a directory services, adding a new application, for instance, may require a review of your architecture and possibly a change in schema, business, and data-feed processes. In addition, many of the steps can and probably should be done concurrently. Consider the included process as more of a functional checklist than a serial requirement.
- Because implementation depends on your local situation, support, and constraints, not all of these process stages may need to be addressed in the prescribed order or even at all. Directories are a reflection of the technology, data, policy, and political environments in which they are implemented. This Roadmap provides steps that have been used successfully at other institutions, and reflect current practices. However, the requirements at your campus may be different enough that you may need to add (or skip) a few steps. Focus on implementing the functions outlined in a way that best suits your institution and accommodates future inter-institutional application requirements.

This Roadmap is a work in progress. Comments are encouraged and welcome.

NSF Middleware Initiative
internet2-mace-dir-directory-implementation-roadmap-200510.html
[Roadmap Change Log](#)

Ann West, Editor
EDUCAUSE/Internet2
Michigan Technological
University

Copyright © 2005 by Internet2 and/or the respective authors
October 2005
Comments to: nmi-support@nsf-middleware.org

Acknowledgments

This website is a compendium of many people's experience and knowledge. Many thanks are offered to the MACE-Dir working group for content and to the campus liaisons of the NMI Integration Testbed who provided much encouragement for the completion of this project. In addition, the Internet2 Early Adopters Project members contributed the Business Case information. Several individuals, including Mike Conlon from the University of Florida, Brendan Bellina from Notre Dame, Paula Vaughan from University of Colorado-Boulder, Andrea Beesing from Cornell University, Jessica Bibbee from Internet2, Matthew Buss from Michigan Tech, Art Vandenberg from Georgia State University, and Mike Stockwell from Cranking Graphics, contributed additional content and design expertise. All errors, misrepresentations, and confusions are solely owned by the person responsible for the compilation.

This work was supported in part by the NSF Middleware Initiative - NSF 02-028.

Project Planning, Preparation, & Requirements

Components critical to the success of a directory services project include educating oneself and the stakeholders, securing support from campus leadership, developing a strong business case, and planning the stages of the project up front.

To plan the project budget, management and directory analysis staff will need to work together to ballpark the infrastructure and initiative needs. This requires that the directory analysis phase be done somewhat concurrently with the project planning.

Policy / Management

Develop business case and secure support

- Educate yourself and organization on need for middleware
- Assemble drivers for campus
- Assess strengths, weaknesses, and critical success factors
- Develop business case
- Secure support

Develop project plan

- Decide on implementation strategy, timing, and organizational approach
- Develop communications and promotion plan
- Discuss with stakeholders, when appropriate
- Develop project specifics

Assemble resources

- Decide on funding model and secure funding
- Develop technical, policy, executive and organizational project structure
- Begin communication plan

Develop Business Case and Secure Support

Policy / Management

Educate yourself and organization on the need for middleware

The first step in a directory services project is to understand why it is important to the institution. You need to make your own business case, preferably with quick, to-the-point delivery, so that when you're on the spot later in the project, the explanation comes easily and confidently.

The University of Florida, for instance, suggests that you reduce the selling points to roughly three compelling reasons in communicating with sponsors, stakeholders, and the community. For example, their points include: the directory provides identity management (a key component of their security strategy), a single system of record for information about people, and a common method for systems to use for accessing information about people.

The first people to approach are the technology stakeholders and possible implementation

team members, who are usually staff in the information technology department and could include the CIO, department management, and technical staff.

Assemble drivers for campus

Next, revisit the campus strategic plans and collective knowledge regarding future application requirements. Fit the directory services implementation into the context of campus needs. It is recommended that you visit stakeholders outside IT to ensure the strongest, most applicable case.

Assess strengths, weaknesses, and critical success factors

Consider the possible pitfalls of the project and have ready answers and alternatives. Review the current organizational and political issues that may provide project roadblocks. Review the policy and business process structure as well. Do you have a data oversight and stewardship policy? You may be able to use an existing cross-functional group assembled for your Enterprise Resource Planning (ERP) system to inform the directory implementation.

Develop the business case

A business case includes both costs and benefits to the institution as a whole. Develop the benefits with the functional users and consumers of the directory services. Assemble the costs with the help of the technology staff, who will need to give input on overall technology strategy and requirements. The result may be a white paper that uses scenarios to explain benefits and provides an estimate of project cost and duration. For an example, see the [University of Florida's Proposal for Enterprise Directory Services](#) white paper.

Secure support

Next, discuss the business case with campus stakeholders to determine the first possible directory-enabled applications and find out how you can assist others with the project. It is advantageous to secure a champion outside IT who can talk to others about the importance of the project to campus. This stage can help you refine your business case for presentation at the next level.

Lastly, make sure you secure executive support to help secure funding and develop policy.

Tools and Resources

Articles

[Beyond Bandwidth...](#) (PDF) provides an interesting perspective on the next challenges of our growing reliance on global networked computing.

[Directory Services: The Foundation for Web Portals](#) (PDF) discusses the importance of directory services to web portal implementations.

[Gaining the President's Support for IT Initiatives at Small Colleges](#) (PDF) helps IT project managers, directors, and CIOs present sound and coherent business cases to their upper administration.

[Identity and Access Management and Security in Higher Education](#) (PDF) provides an overview of identity, access, and security issues, and other aspects involved with implementing these important infrastructures; includes next steps for campuses.

[Middleware: Addressing the Top IT Issues on Campus](#) (PDF) provides background and rationale for middleware deployments as critical new infrastructures.

[Middleware: The New Frontier](#) (PDF) describes why middleware is important to the national research and engineering agenda and the National Science Foundation.

Documents

[University of Florida's Proposal for Enterprise Directory Services](#) (PDF) is an example of a business case/white paper UFL used to make a case for their middleware infrastructure.

[Middleware Business Case](#) (PDF) is a sample business case written by the Internet2 Early Adopters Project Participants.

[Middleware Business Case: Writer's Guide](#) (PDF) is the accompanying guide to help institutions understand and develop their own case.

Slide presentations

"Introduction to Middleware" ([PDF](#)) ([PPT](#)) slide presentation can be used to explain what enterprise middleware is and why it is important.

Develop Project Plan

Policy / Management

Decide on implementation strategy, timing, and organizational approach

The nature of a directory implementation project concerns the following overall tasks:

- Clarify relationships between individuals to be in the directory and the institution. When does an admitted student become able to access online library resources?
- Determine who manages, who can update, and who can see common data. How does an address get changed? Who is responsible for its accuracy?
- Structure information access and use rules among departments and central administrative units. Who can download directory information? Who can access the directory information in real-time? What security roles or levels will be used for various directory data?
- Reconcile business rules and practices. What needs to happen in the systems of record in order to have new student accounts added? Who must initiate this?

Project methodology

Once a decision to go forward has been made, the next set of decisions involve fitting the project into the current political and business context of the institution. There are a number of ways to approach this:

- **Campus strategic project.** This approach entails creating campus-wide support for the implementation and selling the idea strategically.
- **Application requirement.** This approach ties a directory service implementation to the deployment of an application, such as a portal. The costs are then rolled into the portal costs, and political issues are resolved within the framework of a campus-wide application need. Experience has shown that in order to leverage the enterprise directory in a broad fashion, the strategic work has to be done at some point.
- **Stealth.** Many implementations are done without campus buy-in, and instead the business case is made and the project is done inside central IT. This approach requires the necessary data, systems, and network infrastructure groups to be cooperative, and a degree of trust to be present between the technical staff and data stewards. The drawback to this method is the lack of concurrent policy development, which is important strategically to avoid duplicated systems of record, and to achieve economies in data access made possible by a directory service. However, many institutions have used this approach at first to demonstrate value to campus

stakeholders and provide incentive to engage them in the necessary strategic institutional policy and business process discussions.

A bit about risk

A strong middleware project plan will include a discussion of major risks in the resources and applications the directory is serving as well as the project itself. For example project risks and contingency plans, refer to the "[Internet2 Early Adopters Participants, Profiles, and Scope Documents](#)."

Develop communications and promotion plan

Managing expectations and publicizing quick wins is critical to acceptance of directory-enabled applications. Like ERP systems, middleware cuts across divisions and requires broad support and needs a champion, a shared vision, and support from the executive levels.

Through a combination of face-to-face conversations and presentations and web/hard copy communications, many campuses have handled this aspect well. The former allows the presenter to tailor the message to audiences such as the financial officer, data stewards, and technical staff, and the latter keeps the overall message consistent. If possible, identify ways to involve stakeholders in the decision and policy-making process.

Project managers and staff may find they need to reiterate the overall goals and business case many times before the directory is deployed and applications are enabled. Experience has shown that it is not possible to over-communicate.

Discuss with stakeholders when appropriate

When selling the idea to stakeholders, some presenters have mentioned the terms "identity management" or "directory services" and some have focused more on the outcome for the audience. Whichever method is chosen, tailor the message to the audience, do not overwhelm them with details, and emphasize the points most important to them, such as security, privacy, less time to deploy, or increased service offerings with decreased risk.

Consensus is important, but not critical to the project if you have enough support elsewhere. Report progress as often as you can, and keep a consistent, constant message. Allow plenty of time up front to work with data stewards, and include them in the policy/business process discussions, if possible.

The technical staff are also stakeholders in this project. Expect at least some disenfranchisement as some audiences will happily accept, some will reluctantly accept, and some will need dragging along in order to understand the benefits and needs. Look for ways to make IT services easier and better for building internal support.

Develop project specifics

The [Internet2 Early Adopter's Scoping documents](#) offer a good outline of how to structure a project plan and what to include when developing one. Below are a few highlighted items that are especially important to consider:

- Quick wins should be planned into the project early in the process to demonstrate value. Middleware's benefit is often found in productivity gains or through self-service. Identify ways to measure this ahead of time.
- Success enables more success. If the first few enabled applications are accepted, the directory team will be approached with more. Make sure later requests can be accommodated to help keep enthusiasm high.
- Over-provision the first infrastructure to accommodate growth, both in the use of the first applications and the addition of new ones. Do not skimp on hardware or redundancy.
- Develop overall guidelines for the directory and project, such as criteria for adding

data to the directory and how those decisions are made, and so forth. This helps in decision-making later, when the project members can get bogged down in details.

- Be prepared to redefine responsibilities of people as the workload changes. The initial development team might not be the best to support this once it is in production.
- Treat the directory as a formal application development project, and provide for a life-cycle of support and management.

For a non-scientific assessment of project readiness, refer to the [Identity Management Project Readiness Self-Assessment Checksheet](#) (PDF).

Staff requirements

Below are the common functions needed for a directory service deployment. These roles can be done by a few or many team members:

- **Technical architect** grasps the breadth of databases, applications, security, and their interrelationships. Understands organizational needs and values, and can map these into functional and security requirements for middleware.
- **Project manager** has a level of influence equal to being near the top of the central IT organization chart, and manages overall project progress and policy issues. Could be the same as the technical architect.
- **Policy developer** works with stakeholders and university administrators to develop policy for the directory services.
- **Systems analysts and interpersonal communication specialists** interact with data stewards, ensuring that detailed designs mesh with real practices in business and academic offices.
- **Systems, database, and application developers** implement the selected technologies and understand the details of how they must be integrated into the existing infrastructure.

Institutions use a number of ways to assemble this expertise. Some train staff-in-residence, some hire new staff, and some use consultants. Others have traded staff for open positions with other IT departments and hired new staff. Be creative, share the vision often, offer incentives to staff to participate, and keep in mind outsourcing as an option, if that is acceptable. Whether using in-house or outsourced expertise, the technical staff should understand the strategic value of the project and focus on application development and deployment.

Tools and Resources

Articles

[Directory Services: The Foundation for Web Portals](#) (PDF) by Albert DeSimone discusses the importance of directory services to web portal implementations. This document may assist in making the business case for directory services, if a portal implementation is the main driver.

[Identity and Access Management and Security in Higher Education](#) (PDF) demonstrates how core middleware, including enterprise directories, addresses security and access management issues in an institution.

[The Middleware Connection](#) (PDF) offers information about and short business case for middleware tailored to your institution's financial and business officers.

Documents

[Identity Management Project Readiness Self-Assessment Checksheet](#) (PDF) assessment is intended to identify factors at your school that other campuses have found to be important in their Identity Management projects.

[Identifiers, Authentication, and Directories: Best Practices for Higher Education](#) offers an up-level technical overview of the core middleware pieces and how they fit together.

Project plans

[Internet2 Early Adopter's Scoping documents](#) offer examples of how several institutions structured and planned their projects, including risks, project drivers, and communications.

Slide presentations

"Introduction to Middleware" ([PDF](#)) ([PPT](#)) slide presentation can be used to explain what enterprise middleware is and why it is important.

To learn more about Project Planning, review the "Middleware Planning and Deployment 102: Mapping Out Your Strategy" ([PDF](#)) ([PPT](#)) slides.

For a business-case presentation that speaks to institutional Registrars, refer to "On the Internet, We Should Know Who's A Dog: Security, Privacy, and Personalization of Online Services." ([PDF](#)) ([PPT](#))

Assemble Resources

Policy / Management

Decide on funding model and secure funding

For most campuses, the bulk of the cost to deploy a directory is related to staff time. This is incurred across the institution in acquiring data, establishing policy, and implementing the technical infrastructure. However, it is also important to consider the capital and operational costs to fully understand the impact this type of project can have as a foundation component within an enterprise. The methods of securing funds for this project can vary from campus to campus, and will depend largely on the existing staff, their expertise, available resources for outsourcing, and level of commitment to other production systems.

It is possible to absorb the cost of this project into existing initiatives that are underway, or within ongoing operational budgets. It can be submitted to management for funding as a standalone project, as well. This is a case-by-case decision, and will depend largely on local considerations, such as funding availability and willingness to take on new initiatives.

The [Sample Middleware Business Case](#) (PDF) provides details for both scenarios - seeking funding as a standalone project, and for a project using existing resources - and includes a budget for a "new initiative", standalone project. A full-fledged return on investment model has been included in the companion [Sample Middleware Business Case: Writer's Guide](#) (PDF) to assist the reader with estimating an institutional ROI for his or her own needs (see Appendix F).

Develop technical, policy, executive, and organizational project structure

The combination of data, people, and business practices found within and between a university's traditional silos of information, function, and technology forms the basis for subsequent decisions regarding the processes and business rules that drive the design, implementation, and management of the project and resulting directory services.

Thus the project structure should reflect a concerted effort between IT, administrative, and academic cultures to proactively foster collaborative relationships and broad participation throughout the project. Investing in a strong project structure will help build a cohesive

enterprise-wide foundation of technology, expertise, and culture; it is this multifaceted foundation that will lay the groundwork for a successful project.

Each element within the project structure should have a clear purpose and the appropriate resources for achieving this purpose. See the article below for an example and details of how the University of Colorado-Boulder set up such a structure. Below is a synopsis of their recommended elements:

A Project Champion, who:

- Recognizes the importance of the project and keeps the momentum going by reinforcing the importance of the directory initiative at every opportunity, particularly to those impacted directly by the project and to those doing the hands-on project work;
- Serves at a level of authority that would enable him or her to act as a conduit to the top ranks of the administration, and to departments and system administrators;
- Champions the project to all parts of the institution, establishing the groundwork for university-wide commitment to the Enterprise Directory Services; and
- Engages in behind-the-scenes work to move the community from passive interest to active engagement with the project.

A Project Manager, who is involved with all aspects of the project, from its inception and:

- Is trusted by the community (preferably through a proven project track record);
- Has a collaborative work style;
- Possesses technical acuity;
- Is a proactive communicator, willing and able to represent the project to a broad spectrum of audiences and forums; and
- Can manage many tasks simultaneously.

A Core Team, who forms the heart of the project and guides the implementation process from conceptual design through project implementation details. Core Team characteristics include:

- A small and nimble group that is large enough number for fostering ideas, yet small enough to move quickly to action;
- Team members who are chosen for their technical and functional expertise, representing major technical and functional areas (e.g., central IT, Libraries, Housing, Computer Science, Registrar);
- Frequent meetings to help ensure continued forward progress and strong communication;
- Detail-oriented issue resolution of implementation and design decisions, and specific task assignments ranging from technical exercises to documentation; and
- A source of recommendations for the Steering Team for their review, comments, and hopefully, approval.

A **Big Team**, who may not ever meet as a team, but will bring the following to the table:

- A broad representation of requirements and interests, because of the breadth of this team's composition. It should include representation from every area that could contribute to or be impacted by the implementation of a Directory Service. This might include system administrators, human resources, university communications, business administration, faculty, athletics, admissions, etc.
- The language needed to perform effectively in a wide-ranging project, such as an Enterprise Directory implementation. This language is acquired through individual interviews during the early design stages of the project, from which critical information about data, systems, business functions, and culture should be gleaned;
- Iterative confirmation of requirements and direction through consultation whenever their individual expertise is needed;
- The foundation-through their imparted knowledge-for the analysis, design, and project implementation strategy; and Enhanced information flow and collaboration through interactions focused on understanding and educating the campus, rather than focusing on technical details.

A **Steering Team** - the smallest and most powerful of the teams, the most significant characteristics of which include:

- A membership composed entirely of key decision-makers, all at levels of senior manager, director, or above, representing the primary constituencies of the campus (faculty, staff, and students) and with the authority to approve process change and institute campus-wide policy;
- A small enough size to foster intimate and intense discussions;
- A role as communication conduits to constituencies and superiors;
- Regular monthly meetings that establish a culture of communication and understanding among and between these constituencies and with the technical side of the project; and
- Responsibility to lay the foundation for a post-implementation Directory Governance Board, which acts as the policy oversight group of the Enterprise Directory.

A **Technical Team**, who tackles the nuts and bolts of the project tasks. This workhorse team, made up of technicians (such as programmers, web designers, database administrators, IT Architects, directory administrators, and the project manager) - shoulders the responsibility for all of the development, testing, and integration of the technical components of the Enterprise Directory. When choosing members of this team, consider who will need to be involved with ongoing directory operations (for example, the Directory Manager, the lead programmer, and/or lead administrator). Continuity of expertise in the transition from project to operations is an essential component to the success of the directory implementation.

This broad participation in the project implementation establishes channels of cross-campus communication and buy-in, and helps define the political and cultural structures to be negotiated. More important, the structure sets a precedent for tackling the interplay of technical, political, and cultural issues that might have the potential to disrupt - or even derail - the project. Constant communication and collaboration between the technical side of the project and the Core, Steering, and Big Teams sets in motion an iterative process that will serve the project and the post-implementation directory equally well. Each step of the design and implementation will generate comments, ideas, and issues that percolate throughout the

project, and will evolve into a system design and implementation stamped with institution-wide authorship.

The reward for participation in the project is the opportunity to influence the design and use of the directory service - and to witness firsthand the results of their influence on the project. This has the added benefit of building a level of confidence in the directory services, which then encourages further adoption of the technology approach throughout the campus.

Begin communication plan

After assembling the resources needed to begin the project, it is time to begin formally rolling out the project using your methods outlined in the communication plan.

Tools and Resources

Articles

[On Beyond Z: Building a Directory Service](#) (PDF) by Paula J. Vaughan, Deborah Keyek-Franssen, and Marin Stanek, EDUCAUSE Quarterly, volume 25, number 4, 2002.

Documents

[Sample Middleware Business Case](#) (PDF) is a sample business case written by the Internet2 Early Adopters Project Participants.

[Sample Middleware Business Case: Writer's Guide](#) (PDF) is the accompanying guide to help institutions understand and develop their own case.

Slide presentations

"Introduction to Middleware" ([PDF](#)) ([PPT](#)) slide presentation can be used to explain what enterprise middleware is, and why it is important.

Directory Architecture Design & Initial Policy Development

This stage pairs the technology development and policy planning and development, so that each can inform the other. Because a decision in one may place limits on the other, it is important that these functions are accomplished in a coordinated and communicative fashion.

Technology / Architecture

Develop campus identifier strategy

- Create an identifier inventory
- Decide on an identifier strategy

Research directory services architectures

- Understand the components and how they interact
- Review campus infrastructure and requirements
- Research current higher-ed practices
- Research security issues and models
- Review and decide on products

Research and design: system architectures

- Design software, hardware, and networking infrastructure
- Work with project management and others to assemble needed technology components

Policy / Management

Educate policy staff about directories

- Discuss and encourage support for the business and project
- Begin discussion of roles and long-term opportunities and challenges

Review policy structure and begin development

- Work with project/technical staff to identify gaps and develop overall requirements
- Determine policy approval mechanism

Develop Campus Identifier Strategy

Technology / Architecture

Create an identifier inventory

Directories (and in particular, person registries) are where different types of identifiers are correlated and mapped to a single unique ID. As a result, it is important to understand such relationships among them.

To understand the complexity and begin the design process, campuses must first research the various identifiers used in systems and applications across campus, and understand their characteristics. For a list of the types of possible identifiers used in source, person registry, consumer and application services, see the [Identity and Name Space Considerations](#) (PPT). Next, create a mapping of the identifiers to help with planning of next steps. There are several examples available in the [Early Adopters Identifier Mappings](#), which also provide a sample template.

More important than the technical details, is the establishment of ongoing relationships between the architecture and those assigning and using fundamental identifiers. Researching identifiers on one campus revealed fourteen different departments maintaining name, birth date, and SSN for the same students.

Decide on an identifier strategy

After creating an identifier inventory of the campus, decide on one campus unique identifier that other identifiers will map to within the enterprise directory. The campus unique identifier, or UID, is the primary internal identifier, and is typically used behind the scenes and not known by most users. The UID is centrally provided, perhaps with distributed online clients, is assigned to all current active users of campus electronic resources, and all other identifiers should be either directly or indirectly linked to the UID. It is valuable to have this be human-unfriendly, to discourage its inappropriate use.

Tools and Resources

Documents

For more information on identifiers within the context of core middleware, refer to [Identifiers, Authentication, and Directories: Best Practices for Higher Education](#).

For a more in-depth explanation of identifiers and a list of [questions](#) to use in doing a campus inventory, see the [Internet2 Identifiers page](#).

To review other campus mappings and a sample template, go to the [Early Adopters Identifier Mappings](#) page.

Research Directory Services Architectures

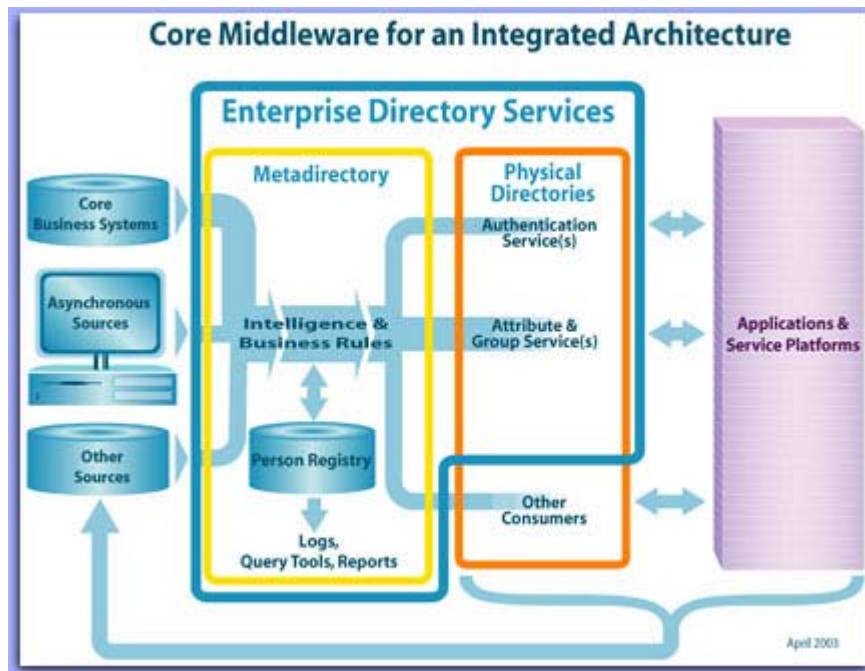
Technology / Architecture

Understand the components and how they interact

There are three common components of an enterprise directory architecture: 1) the registry, which is a database of information about each entity of significance - current and historical; 2) the interface to consumer applications - usually an LDAP directory or authentication service, such as Kerberos; and 3) the metadirectory infrastructure, which controls the flow of information between systems of record, the enterprise directory components, and the consumer applications.

An enterprise directory is generally not a stand-alone service. Rather, it is a means of publishing institutional data in an easily accessible manner. As such, one or more systems of record will provide data for import into the directory. There may also be data that only exists in the directory. There will certainly be a number of users of the directory.

Below is a diagram of the core middleware in an integrated architecture. As noted in the picture, an enterprise directory comprises a number of services and processes, and is typically more than one physical system.



Data enters from the left, passes through a "join" process to merge the information under the correct identifiers and is written to the person registry, which is a database whose primary functions are identity management, reconciliation ("Is this person the same as that person?"), and cross-indexing ("Given this person's ID on system X, find their ID on system Y.") The person registry can also serve as a reference identifier for other systems. Other types of registries, such as organization registries or group registries, may also exist; registries in general are also referred to as metadirectories. Both directory and metadirectory products often come with person registries.

Finally, not all institutions have a physical person registry. Some smaller schools or those with simpler data feeds, may not need to address identifier reconciliation, or they can do it within the metadirectory intelligence before loading into the directory. Disadvantages to this approach include:

- Where are the unique identifiers assigned? In simpler cases, campuses use the identifier from the system of record as the UID.
- How will the campus assign identifiers and offer extended services to broader audiences, such as summer youth camp attendees or theater ticket purchasers? Having a separate person registry can be a place to keep these additional audiences separate from the systems of record and apply different aging/retaining rules to their data, if necessary.

For more information about person registries, see the Early Harvest [Best Practices for Higher Education](#).

The data are then loaded into the physical directories used for authentication and attribute and group services (represented in green) and are served out to the applications. The other consumers could be application or NOS-specific directories.

There are a number of questions to be considered, and they include:

- What are the data sources?
- How will the data be received (batch, real-time)?
- What are the data definitions?
- Are there attributes in the directory that will be updated from more than one source? If so, what are the rules to do the "join"?
- Will a metadirectory product be required to support the various data definitions and joins
- How will directory services be made available to campus? Will there be a centralized service? Or will there be a series of distributed services?
- How much traffic is expected? Will the directory need to be replicated to support the traffic?
- How will the consuming applications or systems access the data? Do they need to be directly provisioned or can they access the directory directly?

For more information on this and schema design, see the [A Recipe for Configuring and Operating LDAP Directories](#).

Review campus technical infrastructure and requirements

Once the various directory service architectures are reviewed, the next step is to look at the current campus infrastructure. Which hardware platforms are already supported by the campus infrastructure? Obviously, supporting a new one will require significant human capital investment, through additional staff or training, or both. Will the network infrastructure be able to handle the required traffic? How big will the Directory (database) be, and how many copies will be in production simultaneously? Is the necessary OS and disk technology to support the high availability need of the Directory, and the expertise in configuring/using the technology, already on campus? If an open-source system is considered, can the campus developers provide the level of support required? All of these will be factors that will impose limitations on the choice of Directory Server.

The nearly universal acceptance of LDAP v.3 means that many of the major email and address book clients will communicate with any compliant directory product. However, there may be older clients that want to use the ph protocol, or finger, to read information from the directory.

There are products available to translate these older protocols to LDAP, but they must be included in the overall project specification.

Research current higher-ed practices

Higher education institutions share a common set of problems, and have a need for some amount of collaboration and data sharing to provide services to one another, as well as to our vendors and contractors. This implies the need for a common schema definition, specific to the needs of higher education. A directory schema called [eduPerson](#) begins to address this. Make sure that eduPerson is defined in your schema for each person entry in the directory to enhance interoperability. Further examples include [eduMember](#) for including group membership and [eduCourse](#) for describing courses and course components. Every institution has local needs, so create a local schema with attributes defined by local requirements. For more information about local attributes other campuses have implemented, refer to the [LocalDomainPerson Object Class Study](#).

Look for other campuses with a similar size population and funding model that have already implemented a directory, or are in the process of doing so. Networking with others on some of the tough problems often helps. The vendors of the chosen product(s) should be able to provide references. There are campus affiliations by location (such as state organization) and by commonality of purpose. Consult the national organizations - the user groups for various vendor products, efforts such as Internet2 and EDUCAUSE, both of which have middleware resources.

Research security issues and models

Campuses need to protect the directory service systems as well as the data contained in those systems from security breaches. Data must also be secured during transit. The LDAP specification allows for authenticated access to the directory, but does not make any statement about the encryption of data or passwords into or out of the directory. Will there be graduated levels of access (that is, will anonymous/public access entities see a certain set of attributes; certain authenticated users see an additional set of attributes; other authenticated users see a different additional set of attributes, etc)? How will users of the directory be authenticated? Is there an existing authentication system (certificates, Kerberos, etc.) that can be leveraged? For more information about appropriate configurations of directories, see [A Recipe for Configuring and Operating LDAP Directories](#).

Review and decide on products

There are a number of commercial and open-source directory-service products available. In many cases, the actual directory server software is just one piece or portion of the overall solution. Other products may be implemented in addition to the directory server software, such as delegated administration tools. Decide on required additional functionality, such as multi-master replication. In some cases, products may require additional products in order to install (for example, a particular compiler version might be required for an open-source product) or to interoperate (will a tool be needed to do password sync between the existing security system and the directory?). Finally, hardware and OS platforms for the products will need to be selected. Some products will run on more than one platform. There may be constraints on performance/functionality if a second tier platform is considered.

After all these issues are taken into the consideration, the choice of products may be self-evident, based on issues of cost (purchase and support), functionality, and performance. It should be noted, however, that the selection committee should include representatives from the policy, data administration, and functional offices, as well as technical personnel. Consider that there may be product-loyal staff that will not be happy with some of the choices made by the project team.

Tools and Resources

Documents

For general information on middleware components, see [Identifiers, Authentication, and Directories: Best Practices for Higher Education](#).

[A Recipe for Configuring and Operating LDAP Directories](#) outlines specific practices for directory design in the higher education sector.

[Practices in Directory Groups](#) offers ideas and methodologies for managing groups in directories, which is for many campuses entry-level authorization.

Directory Schemas

[eduCourse](#) offers guidance for institutions interested in expressing course and course components in an LDAP directory.

[eduMember](#) offers a way to express groups in an LDAP directory.

[eduPerson Object Class](#) and accompanying LDIF files offers a directory person schema that, once installed, can be leveraged to serve inter-campus applications.

[eduOrg Object Class](#) and accompanying LDIF files offers a directory organization schema that, once installed, can be leveraged to serve inter-campus applications.

[LocalDomain Person Object Class Study](#) highlights common attributes added to local directories across higher education.

Research and Design: System Architectures

Technology / Architecture

Design networking, hardware, and software infrastructure

Networking Infrastructure is critical to directory service design. The analysis of the network infrastructure needs to be done in terms of risk management, tolerance for service failure, and recovery times. Then one can review the cost/benefit considerations, such as establishing redundant paths to the network from the directory server where each path leads to a separate switch on the same network.

Another cost/benefit solution involves the implementation of load-balancing routers. This will enable the institution to withstand certain directory server outages, as well as have the ability to expand the service as requirements change. Having directory server replicas to meet demand is built into most directory servers. Using this, you will be able to establish multiple directory servers, all serving the same data on the same or different networks.

As global requirements grow, directory replicas can be added onto other networks positioned around the campus, the region served by the institution, or around the world - and can utilize DNS servers that will resolve names to servers closest to the client. This allows everyone using the directory service to use the same hostname and get the quickest response possible.

Hardware diversity and the accompanying operation-systems choices will impact the directory choice. On one end of the spectrum, there are highly redundant systems that can be bought for

larger sums of money, or smaller, less redundant systems that cost much less money. Your institution needs to find the happy medium.

Since directory servers are "light-weight," it is nice to be able to put the entire database in memory. The ability to respond to requests rapidly is critical, so directory servers should use disk as little as possible - the less overhead, the better. This is where Direct Attached Storage (DAS) is utilized. Most of the time, directory servers deal with small bits of information, such as password verification; multiprocessor machines work well for this type of service.

If you have implemented load-balancing routers, growing your directory service to meet demand is very simple. Just add another replica host, and have the load-balancing router start giving it requests. If this service is contained in one machine, it needs to be as fault tolerant as possible. It will soon grow to be one of the most important machines on campus, as well as an important single point of failure for other campus services.

See the [diagrams](#) for simple and more robust examples of how to structure a systems infrastructure to support middleware.

Software, or the directory server package, can be a complex decision. All directory servers are not the same. Find out which directory server has the features you would like to implement. Determine how the server software adheres to directory standards, as well as how easily it can grow to meet increasing demand. Ease of maintenance should also be considered.

Educate Policy Staff about Directories

Policy / Management

Discuss the enterprise directory, and encourage support for the business and project plan

Meet with key business unit heads and project teams to discuss the role of the enterprise directory (see sample presentation below.) As part of your communication plan, now is the time to create communication forums to facilitate developer initiatives involving the directory.

Hold focus group meetings with campus developers and central IT Directory Services staff. Plan to meet regularly with campus IT groups to provide status updates, and obtain input for planning program initiatives.

Determine the services or applications that you can provide to the critical audiences that assist them in their work or serve their constituents. If you need the support of a certain stakeholder, negotiating support in exchange for a mutual benefit may be the best approach. See if there is "low-hanging fruit" in the form of directory-enabled applications, which they want deployed and are easy for you to do so, once the directory is built.

Begin discussion of roles and long-term opportunities and challenges

To ensure a smooth transition between existing policy and political environments, make sure that the individuals critical to the operational process buy into your plan. Data stewards, for instance, should be involved not only in the initial data discussions and implementations of data feeds, but also in the longer term oversight and management of the directory. If your campus does not have a data administration policy for the various systems on campus, you will need to address this to build the directory. Begin discussing big policy issues such as this early on, as they will take significant time to develop and resolve.

Tools and Resources

Presentations

This sample presentation from Cornell University, "Integrating Applications with the Directory," ([PDF](#)) ([PPT](#)) can be used to explain issues to audiences who are new to directory services.

Review Policy Structure and Begin Development

Policy / Management

Work with project/technical staff to identify gaps and develop overall requirements

Beginning with your existing policy structure, determine the following:

- Is data viewed as a strategic resource?
- What are your data access and use policies, and how are they determined?
- Which data will you keep in the directory, why, and for how long? Will you delete entries? What types of services will you use it for?
- [What are the guiding principles for your directory?](#)
- Who are your data stewards for your systems of record? Typically, the data steward for the system of record is the same one for corresponding attributes in the directory.
- Who should be the data steward for information where the directory is the system of record? Will there be any data that fit this category? Why or why not?
- Who can create, read, update, and delete the data in the directory? Is IT a steward for any new information, such as e-mail addresses or UNIX UIDS?

Can a developer access non-public directory data? If so, what is the approval process to do so? How do you get in touch with the developer in the future, if the data policies change?

Determine policy approval mechanism

If you believe that a number of policy/procedure changes will be needed to support the directory, determine what you can do to streamline the approval process. Some campuses have set up a group to review and approve these as the project progresses. See the introductory section of the [Project Planning, Preparation & Requirements](#) section for how to structure your project to support this.

Data Flow & Business Process Review: Continue Policy Development

Populating the directory can expose data and business-process integrity problems. Management and technical staff typically work together closely during this stage to address the administrative and technical challenges. In many cases, this stage is done concurrently with the architecture stage.

Technology / Architecture

Policy / Management

Review chosen application requirements

- Design for the first applications, keeping the longer term in mind

Continue communication plan

Review and design business processes

- Review business and application requirements
- Work with technical implementation team to determine where current practices need alteration and new ones need development
- Work with data custodians to develop data update, flow, and oversight approaches

Develop data flow from source systems and to consumer systems

- Consider and architect the three major processes involving consolidation/identity matching, intelligence, and application/system consumers
- Work with data custodians to populate the directory with the correct data and work with management on getting clean data

Continue policy development

Develop technical processes according to business and architecture requirements

- Develop person registry and data load processes

Review Chosen Application Requirements

Technology / Architecture

Design for the first applications, keeping the longer term in mind

The goal of an enterprise directory service is to meet the needs of consumer applications. The only metric for success is utility. Unlike directories designed to meet the needs of a single application, such as a campus white pages or an email directory, enterprise directories are designed to meet both the immediate needs of a multitude of existing applications as well as the future needs of applications and services yet-to-come. As a result enterprise directories require a greater focus on the processes for migrating data from systems of record and providing that data to other services, systems, and application directories.

However, an institution does not need a lot of data in the directory to begin deploying applications. The directory structures should be well thought out and implemented for the long run, but the directory can be populated sparsely at first and expanded as the applications are added.

Tools and Resources

Documents

For further information on populating directories, see [Metadirectory Practices for Enterprise Directories in Higher Education](#).

[Practices in Directory Groups](#) offers ideas and methodologies for managing application groups in directories.

Develop Data Flow from Source Systems and to Consumer Systems

Technology / Architecture

Consider and architect the three major directory processes: 1) consolidation/identity matching, 2) intelligence, and 3) provisioning of data consumers.

There are three major directory processes. The first is consolidating data from all systems of record, such as human resources information systems, student information systems, email address tables, UNIX account information, campus telephone directories, physical office locations, etc. All information is then "joined" to produce a single master record for each individual. This identity matching process resolves records that appear to be related to an individual, determining definitively whether they are related or not. The resultant collection of resolved master records is referred to as a "registry", and may be stored in a single data store (database table, indexed file, etc.) In essence, this process reviews all of the relevant institutional sources of data, and joins them together.

The second process, or "intelligence", manages how data is inserted, modified, and deleted from the registry, based upon the business rules of the institution. This process is mindful of both the data providing source systems and the applications that will consume the transformed data.

The third process considers all the applications and systems using the directory, and provisions them accordingly. For example, directory-enabled applications, such as calendaring, may require an LDAP directory presentation of the data. Non-directory-enabled applications may require ODBC presentations of just a few of the attributes. Resource provisioning and account management systems track additions, removals, and changes of status, and perform tasks accordingly.

Work with data stewards to populate the directory with the correct data and work with management on getting clean data.

Identifying the systems of record for each directory attribute is a critical component in the analysis of the enterprise directory services. One of the biggest challenges reported by institutions is maintaining the integrity (or correctness) of the data in the enterprise directory, given that some portion of the data in their systems of record is out-of-date, contains mistakes, and/or is not consistently formatted. Most institutions prefer not to fix bad data within the enterprise directory and instead develop a policy stipulating that corrections must be applied at the source. While such policies reduce the amount of transformations required to handle erroneous data, they may undermine the usability of the directory for consumers, if the administrators of the source systems are less than responsive.

Another common problem is deciding which pieces of source data to use in the enterprise directory. Faced with the likelihood of multiple values for commonly required attributes (such as names, addresses, phone numbers, and job titles), analysis may be required to determine which values are the most appropriate ones to reside in the directory. Such analysis requires

applying knowledge of the business rules, policies, and the source systems (as well as in some cases, allowances for personal preference), all of which are different within each institution.

It is common to transform data before putting values into a registry. Standardizing format, case in names, and attribute contents are the most common, along with removing duplicate names coming from different systems. Directory-project planners should bear in mind that data transformation can require a significant investment in time and energy.

Tools and Resources

Documents

For further information on the functions of metadirectories and implementation guidelines, see [Metadirectory Practices for Enterprise Directories in Higher Education](#).

The [Local Domain Person Object Class Study](#) summarizes the results of a survey of institutions and the institution-specific attributes that have been added to their directory.

Develop Technical Processes According to Business and Architecture Requirements

Technology / Architecture

Develop person registry and data load processes

Building the registry entails extracting, transforming, and loading the data. Many institutions write their own scripts or code to manage these processes. Commercial products may also be an attractive option to institutions starting now and for those who wish to move away from scripted solutions.

The recommended method for storing and managing the registry data is to use a relational database. The size of the institution and amount of data to be stored in the registry are two factors to consider. Registries can be "fat" or "thin", depending on how much data is put into the registry. If the source systems are capable of being accessed by a variety of applications (perhaps using LDAP or SQL) and are highly available themselves, then building a thin registry with just enough data to perform the identity resolution might be appropriate, since the applications or consumers can get the identity from the registry and other data from source data stores. Most campuses choose to build a fat registry, in order to supply information to consumers. Fat registries simplify processes required to meet application and consumer requirements and can help to avoid issues regarding the technical or procedural inaccessibility of source systems.

Tools and Resources

Documents

For further information on the "join" process of metadirectories, see [Metadirectory Practices for Enterprise Directories in Higher Education](#).

For additional information on person registries, see [Identifiers, Authentication, and Directories: Best Practices for Higher Education](#).

The [LocalDomainPerson Object Class Study](#) summarizes the results of a survey of institutions and the institution-specific attributes that have been added to their directory.

Business Process Design

Policy / Management

Review business and application requirements

Take a look at the existing policies and procedures used for managing the system(s) of record and determine if they can be leveraged to serve the directory infrastructure. If you are planning to switch from using a social security number as the primary identifier to another identifier, now is the time to determine if there are policy and/or architecture changes needed to accommodate the new approach.

Work with technical implementation team to determine where current practices need alteration and new ones need development

For the most part, you will need to develop new policies, procedures, and service agreements for the new enterprise directory. For example, the backup of a directory is problematic because the 24X7 service expectations do not allow for clean snapshots. How often should the data be updated? Once per day? Every ten minutes? As transactions occur? These service targets and policies will affect the applications to be deployed, and should be documented for developers and data stewards, at a minimum.

Confusion can exist regarding the differences between a data warehouse and a directory. Each is a data repository, but they are structured for different purposes. To ensure good communication about the use of a directory, decide on a directory-use philosophy that outlines which data can be added and for what purposes. This will go a long way towards helping future developers know which service to use for each application.

Additional considerations include the following:

- Users should be able to read a statement about how their data is being used, where, and for what purposes. If your campus does not have a privacy or personal data use policy, you are strongly encouraged to develop one.
- If your campus has decided to implement a new identifier or identity management system, there will be significant policy development associated with your project.
- As application-integration best practices emerge, develop a checklist for RFPs to ensure compliance with future products, whether purchased or public-domain.

Work with data stewards to develop data update, flow, and oversight approaches

The goal of this step is to develop a shared understanding, and collectively develop a document outlining the directory attributes, their source, and ownership permissions. Consider creating a CRUD table (Create, Read, Update, and Delete) for each piece of data. You could also include information about the metadirectory processes that were performed on the source data to arrive at the directory attribute value. In many cases, project teams spend an extended period discussing the definitions of terms such as faculty, student, and staff, discovering that people on campus have all three affiliations, and then deciding which classification takes priority and for what purposes. Grinding through this definition process is often the best way to expose and address these terminology differences.

When applications are deployed, data and process issues are exposed more acutely. For example, a student worker in the payroll system appears in the on-line white pages with an on-campus office location. Once the student leaves the position, if the payroll system is not updated with a termination date, the student's entry retains the office location. The student complains to the directory administrator, typically, and the process issue is pursued and fixed. There will be *many* similar examples, not only when you first deploy the directory, but also as you add new applications that use the data in different ways.

Because of these ongoing issues, it is a good idea to develop an oversight process to ensure the data custodians, developers, application owners, policy-makers, and directory administrators communicate on a regular (if infrequent) basis. New developers, then, request the data use from the data stewards, rather than the directory administrator.

Tools and Resources

Documents

To address ongoing governance issues associated with the directory use, campuses should consider establishing an oversight process. See [Enterprise Directory Oversight Process](#).

Directory & Applications: Implementation & Deployment

Project teams bring the entire process together with application deployment. As in previous stages, the technical and functional teams must work together to assemble and test the pieces and assure that the technical, policy, and business requirements are met.

Technology / Architecture

Install network, hardware, and software systems to support the deployed services

Implement directory, security, and data flow architecture

- Implement person registry and data flow functions
- Populate directory and test
- Prototype first applications and work with stakeholders on testing
- Include mechanisms for implementing institutional privacy policy/approach

Deploy monitoring and operational tools

Policy / Management

Continue communication plan

Participate in testing initial applications and directory service

- Review initial applications for functionality, data integrity, and policy/legal compliance

Implement oversight mechanism

Publish project success, and thank participants

Implement Directory, Security, and Data Flow Architecture

Technology / Architecture

Implement person registry and data flow functions

As noted in the design phase, a person registry is a directory or database whose primary functions are identity reconciliation.

The implementation of the person registry has three stages:

1. Determine if a new person entering the registry is actually new or if he/she already exists in the registry. This is usually done by comparing key elements, such as name, date of birth, city of birth, and mother's maiden name.
2. If the person is new, assign an identifier to them and enter them into the registry. If the entrant is an existing person, the source system may be notified and provided with the existing person's identifier. If there is uncertainty, the system that initiated the new entrant is notified and an arbitration process is begun.
3. Determine if a person's information needs to be updated, such as a

name change. Since the person registry holds very little volatile information, this is an infrequent and straightforward activity.

The person registry may be operated by a central IT organization or by a sponsoring campus unit, such as the Registrar or Personnel. This unit may handle the arbitration process as well. Although there is a real cost in labor to this work, there are major institutional efficiencies to having this focused approach.

Because query resolution speed is not important, the person registry may be implemented as a database, rather than a directory. Unexpected benefits of a person registry may include cleaning up after student information system errors, such as miss-typed names. One should be careful about combining person registry entries into a single entry. Once merged, separation is difficult.

Populate directory and test

Populate the directory server with the clean data set created in the data phase. This data will be the result of the feed process that extracts or generates data from your systems of record.

Test the data by comparing it with your source. Work with the stakeholders and data stewards to review the data in the directory server to make sure it is correct.

Next, perform load testing, if possible. There are benchmarking tools available that are useful for load testing a directory server. Some are free, while others are vendor products. Of course, it is difficult to mirror "real world" load, but it will provide you with a general idea. Your access-control approach may lead to performance problems. This would be a good time to verify that your access-control rules are working as expected.

Prototype first applications, and work with stakeholders on testing

Develop a test plan for the application(s) to communicate target metrics and facilitate user testing. Remember that stakeholders and data stewards may not have a great deal of time to spend on testing applications, so try to make it as easy for them as possible to make progress.

Deploy your first application(s) in a test environment, and allow people, especially stakeholders, to begin using them. You may wish to limit this testing to just the appropriate stakeholders and data custodians, until they are comfortable and "sign off" on the application(s) and/or use of the directory server in general.

Access control rules, schema changes, and even data sets can change at this stage, depending on the results of your widened testing.

Include mechanisms for implementing institutional privacy policy/approach

There are at least two approaches to implementing institutional privacy policies:

1. Develop a policy, and enforce it with the application developers using the data in the directory server responsible for upholding that policy. This could involve writing an API in multiple computer languages for application developers. Most directory servers come with an ability to handle access control. One could develop access control rules in the directory server itself that enforce policy rules.
2. Develop access control rules that specify which data are available to which authorized users. Using this approach, it will not be necessary to develop and maintain one or more APIs.

Tools and Resources

To check your directory schema and design against current higher education practices outlined in the [LDAP Recipe](#) and popular schemas, such as [eduPerson Object Class](#), use the [LDAP Analyzer](#).

Deploy Monitoring and Operational Tools

Technology / Architecture

The replication capabilities of LDAP directories are often used to provide a high-performance, redundant directory environment, in support of directory-enabled mission-critical applications. While updates to a directory service may be infrequent, replication logs should be monitored to ensure that updates are as expected.

Systems and processes may fail to operate perfectly from time to time and bad data might be moved into the directory, causing a corresponding impact on services. The risk for such an occurrence can be minimized by the implementation of batch flow models and monitoring thresholds. A batch flow model could afford an operator the chance to review proposed updates before they are released. Reasonable thresholds could be established, such that batch updates are permitted to proceed automatically if the number of changes in the update is less than the configured threshold. Otherwise, an operator could be notified so that manual review could be performed.

In order to meet the authentication and authorization needs of applications, directories need to be highly available and highly responsive. Monitoring and graphing the performance of LDAP operations using utilities, such as Look, can help in quickly ascertaining operational deficiencies. Doing so can also provide useful information for troubleshooting unexpected problems, and help with capacity planning to ensure operational requirements continue to be met as more applications integrate with the directory service. Because of the open and public nature of LDAP directories, directory administrators may not know that a new application is performing LDAP queries, so operational monitoring is critical to the support of the service.

Tools and Resources

[LDAP Operational ORCA "k"ollector](#) (Look©) is a Perl utility that gathers LDAP performance data at periodic intervals, and generates a file of summary results in a format compatible with the open-source [ORCA](#) web graphing product.

Participate in Testing Initial Applications and Directory Service

Policy / Management

Review initial applications for functionality, data integrity, and policy/legal compliance

Work with the data custodians and owners of the initial applications to assess data accuracy and directory performance. As mentioned in the Technology/Architecture section, develop a testing plan and require approval of the directory, data, and app owners before going live.

At this time, you also may want to consider how users report problems with the applications and/or data, and who is responsible for troubleshooting. Sometimes it is difficult to know if the issue is a directory, application, or data problem. If possible, make sure the users know how to change their information.

Tools and Resources

Documents

University of Wisconsin-Madison has a well-developed [LDAP requirements list](#) that they include in the RFPs to vendors.

Implement Oversight Mechanism

Policy / Management

As noted in the initial Policy Development section, an oversight group should be established to ensure long term compliance of the service to institutional policy, national and state law, user needs, and changes in the systems of record.

When new data elements are considered for the directory service, it is important to include both the data stewards and the directory architects in the decision-making process. Determining access to data is the steward's responsibility. Securing the data and providing capacity for new demand is the responsibility of the directory architect.

In the long term, it is very useful to create a similar CRUD (create, read, update, and delete) matrix for applications. Who owns the applications? Who can update and decommission them? This can take a lot of work up front, but it retains the directory as an infrastructure service that leverages institutional data to serve stakeholders and their applications.

Tools and Resources

Documents

A sample oversight document was developed to assist campuses in establishing this function. See the [Enterprise Directory Oversight Process](#).

Publish Project Success, and Thank Participants

Policy / Management

When the project plan and steps designed at the outset are accomplished, it is time to thank the participants in a visible way, and notify the campus of the service success.

Even though the directory will continue to be updated and changed in accordance with new application and institutional requirements, mark the end of the development and the beginning of the oversight/production phases. You should also formally release the time commitments of those who have contributed to the project, but who will not be involved at an ongoing level. Sending a formal letter of thanks and contribution highlights to their supervisor is a tangible way of doing this.

Lastly, it is best to end the process with the team enjoying themselves, wanting to work together again in the future (when you start your campus authentication project), and feeling good about your collective work and outcomes.

Resources & Bibliography

Project Planning, Preparation, & Requirements

Articles

[Directory Services: The Foundation for Web Portals](#) (PDF) by Albert DeSimone discusses the importance of directory services to web portal implementations. This document may assist in making the business case for directory services, if a portal implementation is the main driver.

[Identity and Access Management and Security in Higher Education](#) (PDF) demonstrates how core middleware - including enterprise directories - addresses security and access management issues within an institution.

[The Middleware Connection](#) (PDF) offers a short business case and information about middleware tailored to your institution's financial and business officers.

[Sample Middleware Business Case](#) (PDF) is a sample business case written by the Internet2 Early Adopters Project Participants.

[Sample Middleware Business Case: Writer's Guide](#) (PDF) is the accompanying guide to help institutions understand and develop their own case.

[Gaining the President's Support for IT Initiatives at Small Colleges](#) (PDF) helps IT project managers, directors, and CIOs present sound and coherent business cases to their upper administration.

[Beyond Bandwidth...](#) (PDF) provides an interesting perspective on the next challenges of our growing reliance on global networked computing.

[Directory Services: The Foundation for Web Portals](#) (PDF) discusses the importance of directory services to web portal implementations.

[Identity and Access Management and Security in Higher Education](#) (PDF) provides an overview of identity, access, and security issues, along with other aspects involved with implementing these important infrastructures; includes next steps for campuses.

[On Beyond Z: Building a Directory Service](#) (PDF) by Paula J. Vaughan, Deborah Keyek-Franssen, and Marin Stanek, *EDUCAUSE Quarterly*, volume 25, number 4, 2002.

Documents

[University of Florida's proposal for Enterprise Directory Services](#) (PDF) is an example of a business case/white-paper, which UFL used to make a case for their middleware infrastructure.

[Identity Management Project Readiness Self-Assessment Checksheet](#) (PDF) assessment is intended to identify factors at your school that other campuses have found to be important in their Identity Management projects.

[Middleware: Addressing the Top IT Issues on Campus](#) (PDF) provides background and rationale for middleware deployments as critical new infrastructures.

[Middleware: The New Frontier](#) (PDF) describes why middleware is important to the national research and engineering agenda and the National Science Foundation.

Slide presentations

"Introduction to Middleware" ([PDF](#)) ([PPT](#)) slide presentation can be used to explain what enterprise middleware is, and why it is important.

Directory Architecture Design & Initial Policy Development

Documents

For general information on middleware components, see [Identifiers, Authentication, and Directories: Best Practices for Higher Education](#).

[A Recipe for Configuring and Operating LDAP Directories](#) outlines specific practices for directory design in the higher education sector.

For a list of the types of possible identifiers used in source, person registry, consumer, and application services, see the [Identity and Name Space Considerations](#) (PPT).

[Practices in Directory Groups](#) offers ideas and methodologies for managing groups in directories, which is, for many campuses, entry-level authorization.

For more information on identifiers within the context of core middleware, refer to [Identifiers, Authentication, and Directories: Best Practices for Higher Education](#).

For a more in depth explanation of identifiers and a list of [questions](#) to use in doing a campus inventory, see the [Internet2 Identifiers page](#).

Sample core principles for an enterprise directory are listed on [Strategies: Core Principles](#).

See these [diagrams](#) for simple and more robust examples of how to structure a systems infrastructure to support middleware.

To review other campus mappings and a sample template, go to the [Early Adopters Identifier Mappings](#) page.

Directory Schemas

[eduPerson Object Class](#) and [eduOrg Object Class](#): accompanying LDIF files offers a directory person schema and a directory organization schema that, once installed, can be leveraged to serve inter-campus applications.

[eduCourse](#) offers guidance for institutions interested in expressing course and course components in an LDAP directory.

[eduMember](#) offers a way to express groups in an LDAP directory.

Data Flow & Business Process Review: Continue Policy Development

Documents

To address ongoing governance issues associated with the directory use, campuses should consider establishing an oversight function. See [Enterprise Directory Oversight Process](#).

For further information on the functions of metadirectories and implementation guidelines, see [Metadirectory Practices for Enterprise Directories in Higher Education](#).

For further information on the "join" process of metadirectories, see [Metadirectory Practices for Enterprise Directories in Higher Education](#).

For additional information on person registries, see [Identifiers, Authentication, and Directories: Best Practices for Higher Education](#).

For further information on populating directories, see [Metadirectory Practices for Enterprise Directories in Higher Education](#).

[Practices in Directory Groups](#) offers ideas and methodologies for managing application groups in directories.

The [Local Domain Person Object Class Study](#) summarizes the results from a survey of institutions and the institution-specific attributes, which they have added to their directory.

Directory & Applications: Implementation & Deployment

Documents

A sample oversight document was developed to assist campuses in establishing this function. See the [Enterprise Directory Oversight Process](#).

University of Wisconsin-Madison has a well-developed [LDAP requirements list](#) that they include in the RFPs to vendors.

Tools

[LDAP operational ORCA "k"ollector](#) (Look©) is a Perl utility that gathers LDAP performance data at periodic intervals, and generates a file of summary results in a format compatible with the open-source [ORCA](#) web graphing product.

To check your directory schema and design against current higher education practices outlined in the [LDAP Recipe](#) and popular schemas, such as [eduPerson Object Class](#), use the [LDAP Analyzer](#).

Roadmap Change Log

[November 2003] Introduction page revised.

[November 2003] Internal pages proofread.

[November 2003] [LDAP Introduction](#) added.

[November 2003] [Resource/bibliography page](#) added.

[December 2003] [Site PDF](#) added.

[October 2005]

- Overall, changed data custodians/owners to data stewards and used systems of record as preferred terminology throughout. Edited text on each page for grammar and readability.
- Introduction - Updated and added identity management context for enterprise directories.
- Develop Business Case and Secure Support - Added an article about making the business case to small-college presidents. Added also the University of Florida's Enterprise Directory Services white-paper and their strategy for making a compelling business case.
- Develop Project Plan - Added the Identity Management Project Readiness Self-Assessment Checklist. Also added a policy developer to the list of staff requirements and removed ineffective example. Updated description and drawbacks to the Stealth approach.
- Research Directory Services Architectures - Added the LocalDomainPerson Object Class Study and the eduMember and eduCourse directory schemas. First paragraph is new and was moved from the Develop Data Flow section.
- Research Systems/Designs Architects - Refocused design aspect on using a cost/benefits analysis.
- Implement Directory, Security, and Data Flow Architectures - Updated discussions of person registries regarding performance requirements and of privacy approaches.
- Participate in Testing Initial Applications and Directory Service - Added University of Wisconsin-Madison's LDAP requirements list, which they include in the RFPs to vendors.
- Implement Oversight Mechanism - Rewrote section on shared responsibility.
- Resources & Bibliography – Updated according to resources added above.