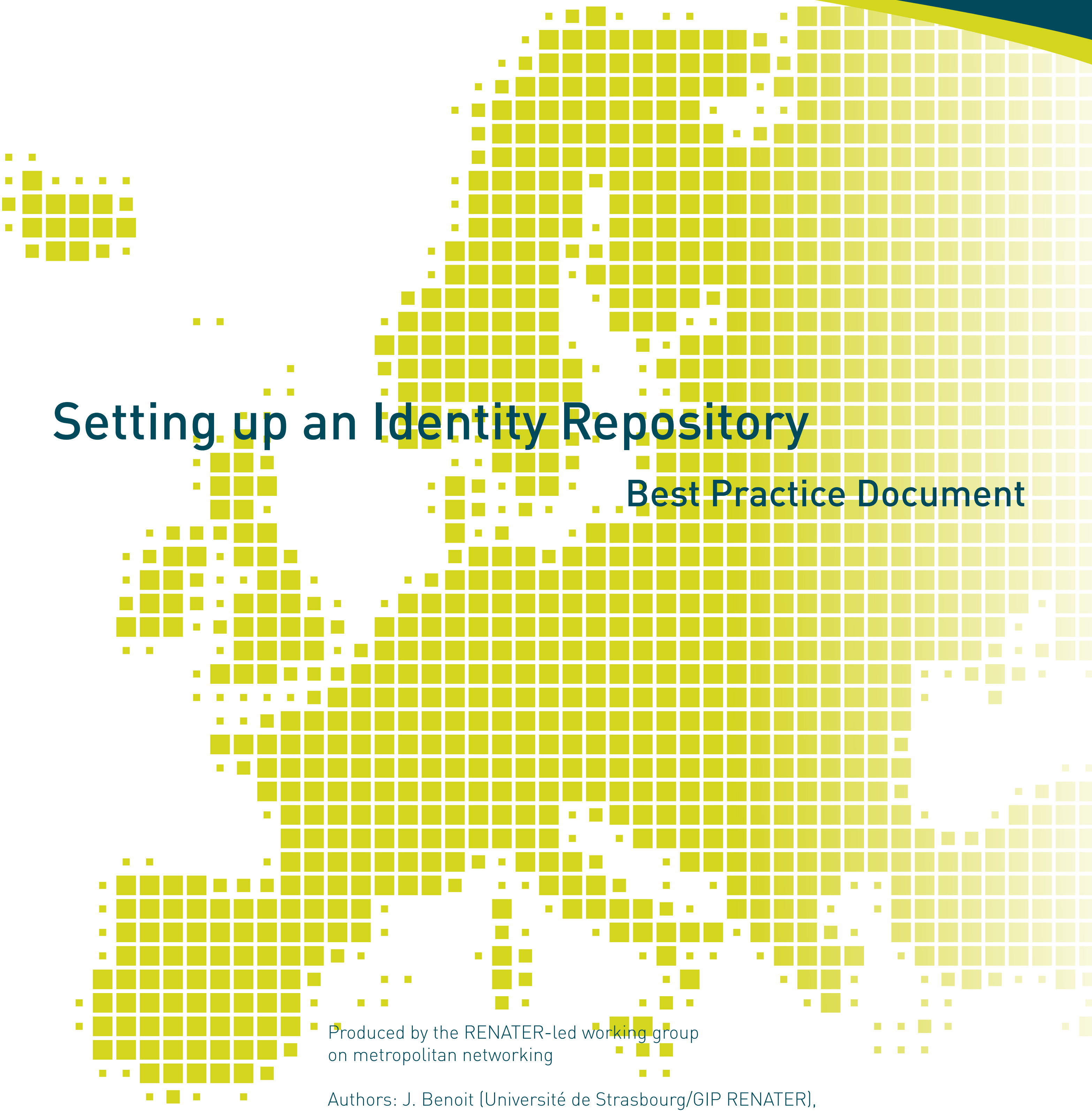

Setting up an Identity Repository

Best Practice Document

Produced by the RENATER-led working group
on metropolitan networking

Authors: J. Benoit (Université de Strasbourg/GIP RENATER),
P. Colombani (Ministère de l'éducation nationale/GIP RENATER),
J-P. Floret (Aix-Marseille Université/GIP RENATER), N. Romero
(Ministère de l'éducation nationale/GIP RENATER), O. Salaün (GIP
RENATER), A. Zamboni (Université de Strasbourg/GIP RENATER)

April 2015

Document No:	GN3plus-NA3-T2-R4.1
Version / date:	Version 1.0 / April 2015
Original language :	French
Original title:	“Constitution d'un référentiel d'identités ”
Original version / date:	V1 - 27/11/14
Contact:	cbp@listes.renater.fr

RENATER bears responsibility for the content of this document. The work has been carried out by a RENATER led working group on metropolitan networking as part of a joint-venture project within the HE sector in France.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 605243, relating to the project 'Multi-Gigabit European Research and Education Network and Associated Services (GN3plus)'.

Table of Contents

Table of Figures

Table of Tables

Executive Summary

This document suggests an approach for setting up the identity repository for a Research and Education (R&E) institution.

The identity repository is located between the applications producing the identities upstream and the applications consuming the identities downstream. We recommend a functional split of the identity repository into two layers. On one hand, the people repository and the organizational unit repository, which aggregate the data coming from source applications and allow this data to be cross-referenced, reconciled and consolidated; on the other hand, the account repository and group repository, which are used directly by the applications consuming the identities.

Setting up the identity repository requires analysis of the existing system: we suggest a method for inventorying the populations, cataloguing the applications managing each population and evaluating the population overlaps. The next stage consists of defining a data model, including information describing the people and the associated meta-information (expiration date, status, data source). An LDAP directory is an appropriate choice for implementing the account repository.

Identity management is intended to automate the provisioning of identities. Several types of synchronization can be envisaged between the repositories and their data sources: full batch synchronization, differential batch synchronization or event-based synchronization. This automation will be coupled with the definition of processes for activating accounts, updating them and arranging for their expiration.

Handling identities involves the use of identifiers allowing reference to be made to the user. We present different formats of identifiers that can be used (name-based, opaque and mixed), mentioning the advantages and disadvantages of each. The login alias is presented as a solution for reconciling user ergonomics and the use of opaque identifiers. Insofar as they are associated with application rights, identifiers must not be reassigned to another user.

1 Introduction

Digital identity is defined as the totality of information about a user: the elements that allow the user to be identified and authenticated (login alias, password, and certificate), the information used for access control (identifier, user profile, privileges, and groups) and more generally information that describes the user (surname, given name, email address).

Each application manages some of this data. In the context of an institutional information system, it is necessary to guarantee the consistency of these identities at the overall level in order to avoid duplication, to define user profiles and to monitor the lifespan of the identities.

This document suggests an approach for setting up an identity repository for a research and teaching institution. We also raise questions related to the identity lifecycle (provision, account expiration, format of the identifiers).

2 Elements of Identity Management

Identity management is a complex and cross-departmental problem. It clearly involves setting up a dedicated technical infrastructure, but the major challenge in the set of problems it poses consists in managing these identity management processes: the creation, update and deletion of identity data have major impacts beyond the identity repository alone.

The following diagram illustrates this idea, showing the different functional elements of an identity management system as well as the major theoretical flows of information between them. These elements will be explained further in the remainder of this chapter.

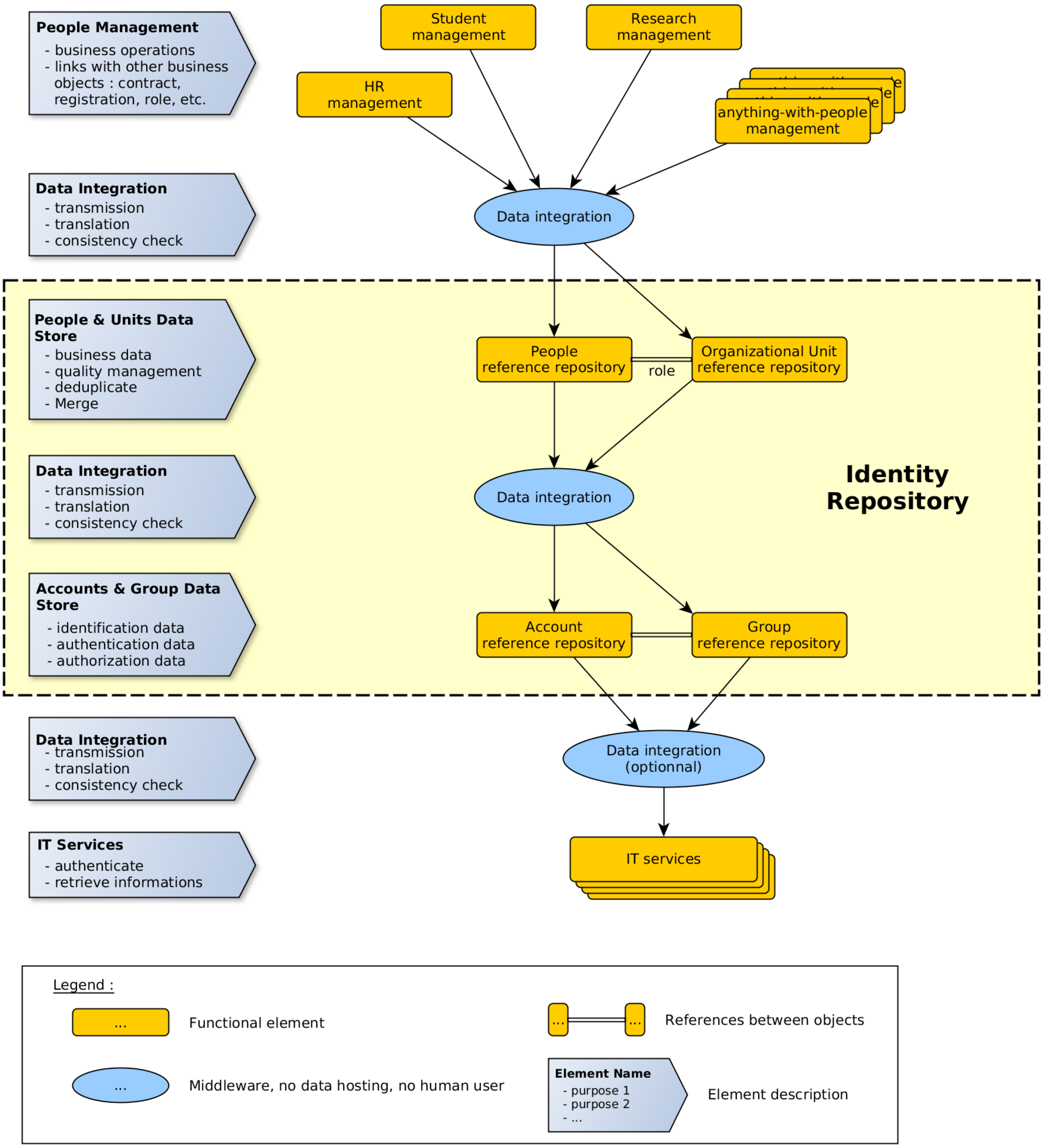

Figure 2.1: Functional elements of Identity Management

2.1 People Management applications

People management applications produce reference information about a person. They are of major importance in that they will be the sources of the data that allows numerous identity management processes to be automated.

Authoritative data sources

In an ideal information system, there should only be one authoritative source of data for each type of person managed.

Examples of populations:

- Students.
- Teachers.
- Administrators.
- Researchers.
- Alumni.

2.2 People Repository

The people repository is the functional element guaranteeing the quality of the data about people within the institution. It contains business data, but its job is essentially to improve the overall operation of information systems. It is therefore an element requiring close cooperation between the IT and business departments.

A few examples of the functionality of a people repository:

- Deduplication of entries about a single person, coming from different people management applications.
- Cross-referencing and reconciliation of data produced by the people management applications.

When should a specific application be deployed as a “people repository”?

A dedicated people repository becomes appropriate when the following two conditions are met:

- When managed by different applications and sealed off from one another, a number of different populations have a significant degree of overlap.
- The institution needs to cross-reference people between these applications: to offer single access to authenticated services; to have a “people file” listing all personal data.

In this situation, the approach to setting up a people repository involves the business departments to a much greater extent even than simple single-source provisioning. The quality and consistency of the data no longer simply have to be guaranteed between two parts (account repository and a management application), but between all parts (people repository and all management applications). It is therefore a cross-departmental institutional project mobilizing all the people management processes within the institution. It is therefore liable to suffer from major inertia. For that reason, we generally recommend initiating this approach with the deployment of an account repository that will allow an operational service to be offered more quickly.

2.3 Organizational Unit Repository

This functional element has the same objective as the people repository, but for the institution’s organizational units. It facilitates identity management by allowing the establishment of people’s roles within the institution to the extent that these roles relate to a unit. These roles may involve authorizations for access to user services. It interacts with the people repository to a major extent and as a result is often managed by the same application.

2.4 Account Repository

This element acts as a centralized user database for authenticated services offered by the institution. Its main mission is to authenticate and identify users connecting to the services. It may also supply information that grants access authorization to restricted services. It is intended to be used by as large a number of services as possible.

Other means of authentication

To make it easier to understand, this document will deal primarily with an account repository based on identification via login and password, but its content remains valid for any other type of authentication.

Using the account repository

Advantages:

- Simplifies the usage of user accounts in the IT services.
- Simplifies the user experience and security: a single password to be remembered, so fewer Post-It notes.
- Reduces direct exchanges between IT services and people management applications.

Disadvantages:

- The unavailability of the account repository blocks operation of all applications that use it,
- A compromised account has a potentially major impact, granting access to a set of applications.

2.5 User Group Repository

This function is intended to centralize user groups and make them available to final services. It is dependent on the account repository because it makes reference to identities. Depending on the strategies of the institution, this element may not exist, or can even exist alongside business information presented by the account repository (or directly by the people repository).

⚖ Using the group repository

Advantages:

- The IT services do not need to offer group management functionality;
- Guarantees consistency of user groups between applications;
- The abstraction of business information specific to the institution into a technical data item, while complying with an interoperable standard. Example (Unix group): “Jean Dupont is a 3rd year undergraduate student in mathematics” becomes “u3math: jdupont”.

Disadvantages:

- Loss of flexibility of and independence in IT services.

2.6 IT Services

These business applications are consumers of the account and group repositories. They do not produce any reference data about users. If that were to be the case, the services would also have to become “people managers”.

2.7 Examples of Implementation

It should be noted that the elements described here are functional and not software components. In theory, it would seem to make sense for each functional element to be implemented by a single application. But, depending on the institutional context (size, existing system, organization, IS strategy, etc.), the architecture can vary while remaining effective. A single application may also have several roles in identity management. The following diagram presents some examples of possible implementations.

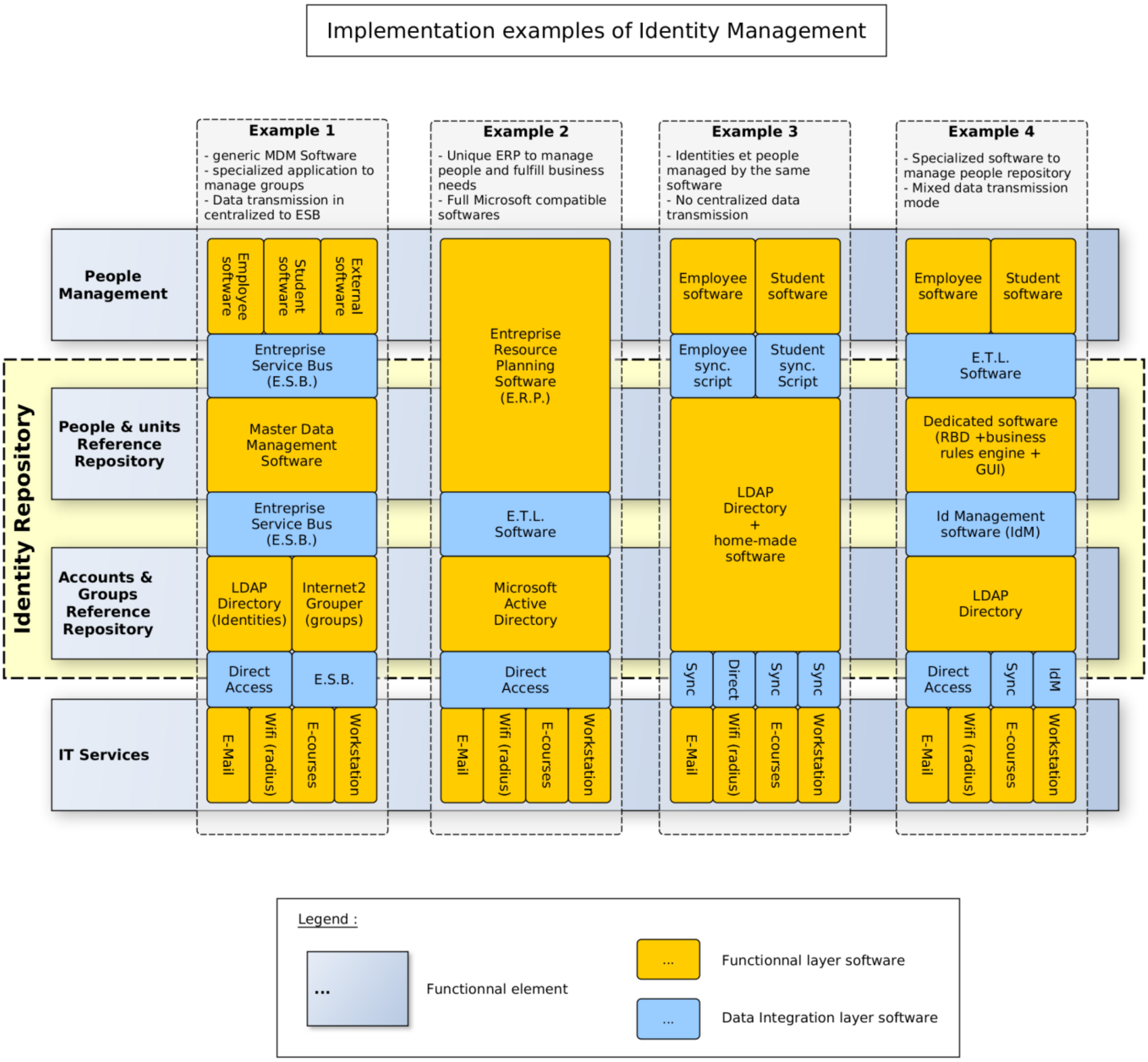
Implementation examples of Identity Management
Example 1
- generic MDM Software
- specialized application to manage groups
- Data transmission in centralized to ESB
Example 2
- Unique ERP to manage people and fulfill business needs
- Full Microsoft compatible softwares
Example 3
- Identities et people managed by the same software
- No centralized data transmission
Example 4
- Specialized software to manage people repository
- Mixed data transmission mode
People Management
Identity Repository
People & units Reference Repository
Accounts & Groups Reference Repository
IT Services
Employee software
Student software
External software
Entreprise Service Bus (E.S.B.)
Master Data Management Software
Entreprise Service Bus (E.S.B.)
LDAP Directory (Identities)
Internet2 Grouper (groups)
Direct Access
E.S.B.
E-Mail
Wifi (radius)
E-courses
Workstation
Entreprise Resource Planning Software (E.R.P.)
E.T.L. Software
Microsoft Active Directory
Direct Access
E-Mail
Wifi (radius)
E-courses
Workstation
Employee software
Student software
Employee sync. script
Student sync. Script
LDAP Directory + home-made software
Sync
Direct
Sync
Sync
E-Mail
Wifi (radius)
E-courses
Workstation
Employee software
Student software
E.T.L. Software
Dedicated software (RBD +business rules engine + GUI)
Id Management software (IdM)
LDAP Directory
Direct Access
Sync
IdM
E-Mail
Wifi (radius)
E-courses
Workstation
Legend :
...
Functionnal element
...
Functionnal layer software
...
Data Integration layer software

Figure 2.2: Implementation examples of Identity Management

Choosing your applications

Applications that allow identity management to be implemented can be categorized according to two criteria:

- Generic software compared to dedicated software: the theoretical advantage of the generic software is that it offers broad functionality and thus allows a rationalization of technologies and competencies. On the other hand, there is a risk it may be less able to respond to certain very specific needs.
- Off-the-shelf product vs. in-house development: off-the-shelf products allow a quick start to be made while limiting human investment. In-house development guarantees that the needs of the institution will be properly met and offers significant control, but imply higher development costs **[homegrown]**.

There is no best option. These choices depend strongly on specific requirements and the strategy of the institution.

3 Setting up an Account Repository

As mentioned above, the deployment of an account repository should be favoured over that of a dedicated people repository. The methodology will therefore allow the set-up of an account repository as described in examples 2 or 3 of the diagram "Implementation examples of Identity Management", i.e. concentrating on provisioning of identities and deprioritizing the problems related to the functional management of people.

The set of problems related to centralized group management, which can prove to be a complex subject, is also ignored here. We will assume that the account repository offers enough information for the departments to set up their own local groups.

3.1 Analyzing the Existing System

The aim of this first stage is to list certain information about the existing system that will then allow appropriate choices to be made. It must allow identification of the positive points of the current situation to serve as a working basis, while noting discrepancies that may exist and that would benefit from correction.

3.1.1 Cataloguing its populations

This table lists the user populations of IT services and the application used to manage each of them.

Population name	Population management application	Volume
Administrative		
Teacher		
Researcher		
Student		

etc.		

Table 3.1: Population census model

Details of the columns:

- Population name: name given to this category of users.
- Population management application: main application in which information about this kind of people are stored and managed.
- Volume: approximate number of people of this type in the institution.

Identifying the populations

Some useful rules and advices to identify the populations:

- The population/data source pairing must be unique
  - The same population cannot come from two sources. If this is so, you must be able to create two distinct populations based on a specific criterion of differentiation.
- The same application can manage several populations. You will have to distinguish them in the inventory table if several of the following conditions are met:
  - The management of these populations differs greatly: business department management, lifecycle, etc.
  - The services offered differ significantly.
  - There is little or no overlap between these populations.
- The aim of this first-level typology is to determine a set-up strategy, not to specify precise authorizations. Consequently, it can remain at a high level.
- This first-level typology should as far as possible be based on the activity of the person in the institution.
- One or several sub-categories will have to be established during the design of the model in order to precisely define the authorizations. For example, the “Teacher” model could be subdivided into “Permanent teachers” and “Temporary teachers”.
- In the examples, we use the indication “local” to indicate that a population is not managed by a business application, but directly in the authentication database of the applications offering services to this population.

3.1.2 Cataloguing the services that are potential users of an account repository

This table identifies the following information:

- The services that will potentially make use of the account repository,
- The current methods of supplying account data for these services.

Service/application	Authentication database	User populations: source of provisioning (from Table 3.1; specify if subcollection)	Volume of users by population
Electronic messaging			
Institution Intranet			
Wi-Fi connection			
etc.			

Table 3.2: Service census model

Fill in the table

- Some services use several authentication databases (chaining). In this case, use as many lines as there are user account databases, specifying each of the populations involved.
- The services/applications still at the project stage must be taken into account.

Examples of completed tables:

Population name	Population management application	Volume
Administrative	Harpège	2,500
Teacher	Harpège	3,500

Researcher	Graal	2,500
Student	Apogée	20,000
Library reader	local	100

Table 3.3: Population census model

Service/application	Authentication database	User populations: sources of provisioning	Volume of users by population
Electronic messaging	LDAP directory	Administrative: manual Teacher: manual Researcher: manual	2,500 3,500 2,500
Electronic messaging	local	Student: Apogée	20.000
Wi-Fi connection	LDAP directory	Administrative: manual Teacher: manual Researcher: manual	2,500 3,500 2,500
Wi-Fi connection	local	Student: Apogée Library reader: manual	20,000 100
Online course	local	Teacher: manual Student: Apogée	3,500 20,000
University administration (Apogée)	local	Administrative (university administration): manual	400
Workstation connection	Active Directory	Administrative: manual Teacher: manual	1,200 800

Table 3.4: Example census of services

3.2 Analyzing and Prioritizing

Setting up an end-to-end identity management system is an initiative of significant scope. The development should be iterative and each stage should be completed with a change that significantly improves the quality of the service or rationalizes resources. These stages should be carefully considered from the outset and should give rise to an overall roadmap.

The aim of the first stage is to put the foundations in place. Later development will hinge upon its success. It is appropriate therefore to focus on setting up a robust and progressive infrastructure. Work that requires major organizational changes that run the risk of delaying the time at which the account repository is brought into service should be planned for a subsequent iteration.

The censuses should reveal the paths that will facilitate this first stage. It will be appropriate to favour:

- The supply of identities for high-volume populations managed by identified people management applications.
- The integration of the services offered to a large number of potential users.
- The designation of an authentication database already used by several services as an identity repository.

Clearly, these rules can be loosened depending on technical possibilities. For example:

- If it proves extremely difficult to extract information on the largest population from the management application, it would be reasonable to start the initiative with a more accessible population.
- If the quality of business data for a population is known to be poor, it is also preferable in that case to ignore this population at the initial stage.
- If the authentication database already used cannot be extended beyond its current scope, it would be appropriate to set up another repository. At a later stage, this account database must be slaved to the repository.

Once the account repository has been deployed, with its initial populations, its initial consumer services and depending on the priorities of the institution, subsequent iterations may consist of:

- Extending the management to new populations.
- Extending usage with new services.
- Developing the data model to add options for categorizing identities (cf. section 2.3, “Creating the Account Repository”).
- Checking and improving the quality of the data.

The following paragraphs present the major stages in the creation of the account repository. Its development in subsequent stages will require similar considerations, but from the point of view of change rather than creation.

Example – analyzing the existing system

Based on example Table 3.3 and Table 3.4, the following objectives could be identified at the initial stage:

- Electronic messaging and Wi-Fi access are offered to all populations. These services must therefore be prioritized.
- These two services already use a shared authentication database (LDAP) that benefits from following a standard and open protocol. This account database therefore has to be designated the institution's account repository.
- Student accounts are provisioned by extracting data from the student management application (Apogée). So it would probably require little effort to supply data to a shared accounts repository in this way.
- The administrative and teaching employees come from a shared application (Harpège) and represent more than 20% of the population. Setting up automatic provisioning from the Harpège application would offer a good return on investment.

A second implementation stage might involve the following subjects:

- Extending the use of the account repository to online teaching or the workstation connection.
- Automating the provisioning of researcher accounts.

The management application for the university administration or the management of "library reader" accounts involves smaller populations; handling them therefore appears to be less of a priority.

3.3 Creating the Account Repository

3.3.1 Setting up the model

On a technical level, the account repository consists of a database listing the users of the information system. It is therefore appropriate to define which information is required for modelling an identity.

This information can be of several types:

- Identification information,
- Operational data for managing the identity lifecycle,

- Contact data,
- Business data to allow the assignment of access rights for services.

Here is some basic user data that makes for a good starting point. However, this list should be adapted depending on the needs of the services using the identity repository, particularly the business data they use.

Information	**Description**
Login alias	Identifier known by the user and used to identify themselves. Preferably user-friendly. Potentially re-assignable, therefore must not serve as a reference for storing personal data. Also known as "login alias"
Password	Authenticator
Unchangeable identifier	If different from the login alias. Can be opaque, unchangeable for the same identity and not reassigned. To be used for references to personal data or synchronizations of data.
Expiration date	Date on which the identity should no longer be usable. Depending on the lifecycle defined, several dates may be necessary (deactivation, deletion, purge, etc.)
Status	The identity may go through several statuses in its lifecycle, for example: new, activated, locked, suspended. These statuses define the actions that it will be possible to perform on the identity and indicate if the account can be used. Depending on the processes chosen by the institution, it may be possible for this idea to be implemented with a single attribute taking several values, or several Boolean attributes, or even a mix of the two.
Data source	Defines the source database for the identities created through automatic synchronization from the business applications. Indispensable for guaranteeing data synchronization.
Identifier in the data source	Identifier referencing the person in the source database. Indispensable for guaranteeing data synchronization.
Given name Patronymic Preferred name Date of birth	Minimum data required to connect the digital identity with a physical person.
Email address Telephone number(s) Postal address(es)	Contact data.

Population(s)	Allows identification of the population(s) (as listed above) to which the identity belongs. Defining a “main” population may prove necessary for services that cannot handle belonging to several populations simultaneously.
Unit(s) assigned to	Displays the organizational units to which the individual is attached. These units may be at different levels (institution, service department, component, centre of activity, laboratory research team). Consequently, depending on needs and the implementation choices, several different attributes could potentially be used.
Employee status	Information about the professional status of an employee. Examples: type of contract, team/grade, end of contract date.
Educational/university course	Information on the academic history of the student. Examples: registration for degrees or courses, grant status, etc.

Table 3.5: Example of basic user data

Favouring the use of standard models

The research and teaching community has defined some standard data models for account management. Using them is strongly recommended. If the information modelled is insufficient, it is always possible to extend the model.

There are several advantages to using standards. A quicker start can be made, and traps into which predecessors in this field have fallen can be avoided. It also facilitates cooperation between institutions in terms of sharing, dissemination or joint design of services.

In respect of disadvantages, it should be noted that if the standard model is not appropriate for consumer services, work to convert the intermediate data may prove necessary. If a majority of the consumer services are neither adapted nor adaptable, this might suggest the use of a private model in your repository. But even in this extreme situation, it is strongly recommended that you have the capability to export the identities in the standard format, to be able to benefit from the choices offered by the community.

The standard recognized formats within the education community are:

- eduPerson (Internet2): LDAP for the representation of people within higher education.
  - http://www.internet2.edu/products-services/trust-identity-middleware/eduperson-eduorg/
- SCHAC (TERENA): scheme complementing eduPerson to facilitate the exchange of data at the European level.
- National directory schema. In the French case, this is Supann (CRU/RENATER) **[supann]**: French supplement to eduPerson.

3.3.2 Choosing the technology

The use of an LDAP directory has now established itself as a standard in the education/research community for building the user account repository. This type of directory is natively designed for managing objects of the type “individual” and in practice offers several advantages:

- Excellent read performance.
- The possibility of extending its schemes.
- Options for distribution across several servers.
- Support from many LDAP client applications.

- A vast choice of implementations, particularly in the world of free software.

The OpenLDAP implementation today seems to predominate among the implementations in teaching institutions, although other free alternatives exist.

Active Directory as identity repository?

From a technical point of view, Microsoft Active Directory is an LDAP directory managing user accounts. It therefore seems appropriate for implementation of the account repository function. However, due to its initial objective (managing stock of Microsoft computers) and proprietary nature, it imposes a certain number of constraints on the model and its organization. For that reason, it is not often chosen to become the central account repository in large institutions. However, it remains a relevant choice in small institutions and/or those using a large number of specific, compatible products.

3.3.3 Technical infrastructure

The design of the technical infrastructure to support the account repository service does not present any major specific challenges compared to other applications. However, it is necessary to insist on the fact that this element will be central and vital for the proper operation of other Information System services. Consequently, it is indispensable to set up a redundant architecture that allows any possible failures in some components to be withstood. It is also recommended to have a disaster recovery plan to put the service back into operation in the case of a major accident (fire, flood, natural catastrophe, etc.).

4 Lifecycle and Provisioning of Identities

4.1 Supplying the Account Repository

Once the account repository is technically operational, identities will have to be created to be consumed by the applications. Although a manual management mode is vital, one of the objectives of identity management is to automate the provisioning of identities from the information based on people management business databases.

Controlling the data

In all cases, it is important to properly control the repository and to always know where the data comes from, when it was created and the date on which it may potentially no longer be there. At the very least, plans must be made for deprovisioning (or deactivation) of the identity based on the management application data.

4.1.2 Choice of exchange type

Beyond considerations of tools and languages, which are specific to each institution, a crucial choice in determining how identity repositories are supplied with data is the type of exchange used. Three major scenarios are presented here. When creating an identity repository, we recommend choosing the most appropriate method for the capacities of the business source databases.

Full batch synchronization

- Description: a periodically executed program reads all the people data in the source, then determines the modifications that need to be made in the identity repository
- Prerequisite:
  - Access to database data (or via an API)
- Advantages
  - Simple to implement,
  - Few prerequisites,
  - Guarantees regular verification of data consistency.
- Disadvantages
  - Larger volume of data processed,
  - Longer processing time,
  - Limits the frequency of synchronizations: two simultaneous executions are prohibited.

Differential batch synchronization

- Description: a periodically executed program reads the source for data modified as of a given date (e.g. the previous execution), then determines the changes to be made in the identity repository.
- Prerequisite:
  - Access to database data (or via an API),
  - The source database must be able to date stamp the changes made to the data.
- Advantages
  - Optimized processing volume,
  - Allows more frequent execution (within a certain limit).
- Disadvantages
  - No automatic correction of information in the repository (requires regular consistency check in parallel)

Event-based synchronization
• Description: changes to the data in the source trigger a call to a method that transmits the data to the identity repository. • Prerequisite: ○ The producing application must have a change detection system. • Advantages ○ Optimized processing volume. ○ Consequences of real-time changes. • Disadvantages ○ Increased software complexity. ○ No automatic correction of the information in the repository (requires regular consistency check in parallel). ○ Error management: requires a system that allows changes to be replayed (e.g. buffer). ○ Initiates a general update of all the identities to correct any errors.

4.1.3 Development/reworking of business processes

The design of an identity repository based on business data creates new restrictions on those businesses. User access to IT services becomes conditional on the user's existence in the information system and their role.

Consequently, the implementation of an identity repository supplied with data from people management applications may highlight problems that had not previously come to light.

The most common difficulties are:

- Incomplete data.
- Obsolete data.
- Administrative files entered late (arrival or departure of people).
- Differences in the way the administrative applications are used according to people, for example:
  - time-based nature of the input,
  - different practices in the same input fields,
- The risks of duplication.

If some of these anomalies can be detected and resolved by setting up a people repository, some basic work between the IT and administrative departments is required to process them at source. We recommend the drafting of clear procedures that enjoy total compliance from all the parties concerned. The array of measures includes adding application constraints, organizational changes and staff training.

This process must be led by the governing body of the institution.

4.2 Account Repository Management

4.2.1 Activation processes

It is important that the person concerned is notified of the account creation.

Example – notifying the user

One way of implementing this action consists of sending an email to the personal address of the person concerned (which he or she will have supplied when their human resources file was created) at the time of the provisioning of the identity.

It is generally not desirable for an account to be activated automatically. The activation of the account by the member of staff is a stage that can enable:

- The IT charter of the institution to be accepted,
- The entry of a password that meets the security requirements imposed by the institution,
- The collection of certain information that will allow an automatic lost password recovery mechanism to be set up (secret phrase, mobile number, etc.),
- The confirmation of the existence of the account, to prevent it being recycled.

4.2.2 Updates

The relevance of the identity data is an extremely important element from the point of view of both security and accessibility. Indeed, many accesses to applications (or to premises) may be based on identity elements (for example belonging to the IT department will grant access to the application for creating temporary Wi-Fi accounts).

The update of the data depends on at least two processes:

- Business process of updating staff data (for example by HR),
- Technical process of regular updates by means of correctly defined data flows (cf. section 3.1.1).

4.2.3 Duration and expiration

Identities are generally linked to creation and deletion (or deactivation) processes, either automatic (when entering staff into an HR database) or manual (guest accounts), as in 3.1 above.

It is relevant to have a permanent solution allowing accounts to be deleted or deactivated when they are no longer required (for example on retirement). This function is indispensable in terms of security and IS control.

For this purpose, local strategies can be based on information from business databases (end of contract date, etc.), intermediate databases (containing guest accounts) or even the institution's LDAP (local end of account attribute for example).

Example – input-output process of an identifier in the IS of an institution

- Each new staff entry in the HR software involves the creation of an identity.
- Each exit from the same software (retirement, change, etc.) involves the deactivation of the identity, either immediately or delayed (to the expiration date).

4.3 Identifiers

The identifiers can be formed in several ways. Most of the time, they should meet the following specifications:

- Controlled lifetime.
- Invariability across the entire lifetime of the identity.
- Controlled length.
- Not reassignable.

In this section, we list all of the different possibilities, as well as their advantages and disadvantages.

4.3.1 Length of identifiers

⚖ Using an identifier of moderate length (a maximum of 12 characters for example)

Advantages:

- Easy for the user to learn,
- Limited risk of incompatibility with some applications (SAP limit of 12 characters for example)

Disadvantages:

- If the form of the identifier is name-based, recycling the identifier will be difficult due to the limited number of characters.
- Some given names/surnames must be truncated due to their length,
- Identifiers with a numerical suffix to avoid duplicates.
- Blockage points between the users impacted and the Helpdesk (“why isn’t my full name used, why do I have the number 2, etc. …).

4.3.2 Format of identifiers

Depending on the strategy adopted by the institution, three types of identifier format can be envisaged: name-based format, opaque format (sequence of characters other than that based on the name) or hybrid format (hybrid of name-based and opaque).

4.3.2.1 *Name-based format*

The name-based identifier can consist of the surname and given name of the person according to an order decided by the institution. Several combinations can be chosen, in order to find alternatives to potential duplicates.

Example - format of a name-based identifier

“surname.first letter of given name” then “surname.first and second letter of given name”

Name-based identifier

Advantages:

- Easier to recognize the owner of the identifier.
- The identifier can be easily memorized.

Disadvantages:

- The algorithm for assigning an identifier to avoid duplicates is not simple; some identifiers can have meanings that are undesirable (Irina Diot could produce “idiot” as an identifier).
- The management of name changes (divorces, marriages) risks creating complications with IT Helpdesk services.

4.3.2.2 *Opaque format*

Opaque identifiers can be created using a clearly defined algorithm in line with the rules of the institution.

Example - format of an opaque identifier

“a+last two numbers of the year'+a+4-figure increment” which could give: “a14a1298”

Opaque identifier

Advantages:

- The assignment algorithm for an identifier is simple and there is no risk of duplication (the namespace is very large).
- These identifiers do not need to be recycled.
- It is easier to avoid the risk of re-assigning an identifier to another person, even many years later.

Disadvantages:

- Impossible to know who the person is without a look-up table (identifiers/names).
- Difficult for the user to learn (memorization).

4.3.2.3 *Hybrid format*

The hybrid identifier can consist of a name-based part and an opaque part. For example, the name of the person followed by 4 figures (last 3 numbers of the year and an increment)

Example - format of a hybrid identifier

loche0141 (Christophe Loche for the year 2014 and with an increment of 1)

Hybrid identifier

Advantages:

- The assignment algorithm for an identifier is simple and there is no risk of duplication (the namespace is very large),
- These identifiers do not need to be recycled,
- It will be easier to avoid the risk of re-assigning an identifier to another person, even many years later,
- Relatively easy to recognize the owner of the identifier,
- Memorizing the identifier remains easy.

Disadvantages:

- Concerns may remain regarding communication with the IT Helpdesk when a person changes their name (divorce, marriage), unless the person's patronymic or given name is used.

4.3.3 Dedicated login alias

We have seen that the format of an identifier can present a readability problem for the user, particularly if the identifier is opaque.

Assigning an identifier for use exclusively in the user connection, commonly called a login alias (attribute supannAliasLogin of the Supann 2009 scheme **[supann]**) allows this disadvantage to be mitigated. In this case, the institution's SSO authentication system allows authentication either with the uid or the alias login (while sending the uid to the client application).

⚖ **Use of a login alias**

Advantages:

- The user can have a personalized login alias (nevertheless, uniqueness in the case of homonymy must be managed).
- The login alias can be changed without impacting the client applications (unlike a change to the uid).
- The login alias can be recycled indefinitely.
- The length of the alias is not subject to the same constraints as the uid.

Disadvantages:

- The login alias is difficult to use outside the institution's SSO. Many commercial applications cannot handle this notion and assume that the login alias that has been entered is the unique identifier and use it directly to reference the user.
- Confusion for the user who no longer knows which identifier to use when.

4.3.4 Recycling

An identifier is generally stored in applications, or in the centralized group management system. This means an identifier, when activated, starts a "social life" in the IS: the associated rights change.

The difficulty with complex and heterogeneous IS lies in ensuring that the deletion of an identifier will result in the deletion of all the associated rights for all client applications. It must be noted that it is rare indeed these days to have this certainty. Reassigning an identifier to a new person therefore involves potentially allowing them to inherit the rights and data associated with the former owner of the identifier. The risk of losing rights then becomes very significant.

In the absence of the certainty of being able to delete all the rights associated with an identifier, the solution is therefore to deactivate an identifier and never reassign it to anyone other than its initial owner (even in this case the same question of access rights is raised again).

However, this raises the problem of managing the identifiers on the IS. The right idea would be to put in place a regular monitoring system covering all systems on which identifiers are stored (rights, accounts). Out-of-date identifiers can then be deleted and reassigned as necessary. Of course this only applies if there is certainty that these identifiers cannot ever be used by external suppliers using the identity federation. Otherwise, re-assignment is completely prohibited.

Conclusion

Setting up an identity repository is an initiative that has a structuring effect on an organization, requiring iterative development and realistic aims. The principles described in this document enable an understanding of each stage in an iteration:

- Choice of a target population,
- Identification of the data source(s) and associated services,
- Definition and implementation of an architecture including the functional elements of identity management, i.e.:
  - adapted to requirements,
  - compatible with the existing system,
  - progressive,
- Possible adaptation of the services.

In the course of the iterations, the granularity and the scope of the identity information will be changed to take into account the sets of problems to be handled, for example:

- Access control based on user characterization: a minimum of a stable and valid identifier, ideally attributes and groups that will allow them to be profiled more precisely.
- Single-Sign-On (SSO) along with consideration of the means of authentication to be supplied to users, depending on the criticality of the services being accessed: the level of reliability of the authentication (password, OTP, certificate) may then constitute identity information.
- Opening towards the outside, via the identity federation or guest management, raising the question of the identity information to be transmitted or requested and its validity.

In conclusion, the identity repository constitutes the basis of a general identity and access management project; the quality of the identity data is therefore crucial. As in any iterative process, the final stage of an iteration must be to verify and establish corrective measures, if appropriate:

- At the level of data supply from data sources, by instilling a sense of responsibility into the managers in charge of these applications regarding the need to enter correct information.

- At the data level, by standardising values (example: deleting spaces and special characters, adopting standardised usage of upper/lower case, maintaining consistency in character encoding).
- At the data integration level, by improving processes for the synchronization and processing of information (for example, deduplicating the data, sending notifications of discrepancies to be dealt with at the level of the data source).

At the user level, by offering to verify and update all or part of the information that relates to the user (if possible with a validation circuit that allows a third-party to accept or reject the changes).

References

[homegrown] Thread about "Homegrown identity management systems", IDM mailing list by EduCause, March 2014. http://listserv.educause.edu/cgi-bin/wa.exe?A2=ind1403&L=IDM&P=3845

[supann] SupAnn, schéma d'annuaire pour l'enseignement supérieur [directory diagram for higher education]. https://services.renater.fr/documentation/supann/index http://www.cru.fr/documentation/supann/2009-en/index (English version of SupAnn2009)

[iamcomponents] Higher Education – Key IAM Components and Requirements, Internet2, 2011 https://spaces.internet2.edu/display/iamtep/Higher+Education+-+Key+IAM+Components+and+Requirements

[landscape] Identity Management in Higher Education – A View of the Landscape, Internet2, 2013. https://spaces.internet2.edu/display/idlandscape/Identity+Management+in+Higher+Education+-+A+View+of+the+Landscape

[idmshoestring] Identity Management on a Shoestring, Architectural lessons from Real-world Implementation, Ganesh Prasad and Umesh Rajbhandari, March 2012. http://www.onlineprogrammingbooks.com/identity-management-shoestring/

[idmodernisee] Vers une gestion d'identités modernisée [Towards a modernized identity management], Pascal Aubry, 2013. https://conf-ng.jres.org/2013/document_revision_1476.html?download

Glossary

Authorization or permission Process for assigning rights within a repository or an application.

Access control Process for verifying rights upstream of a service. This process can be based on authorizations (or permissions).

Login alias Sequence of characters, associated with a password, used by users to authenticate themselves (source [supann]).

ESB (Enterprise Service Bus) IT middleware technology aimed at managing exchanges between heterogeneous applications. Development of EAI (Enterprise Application Integration) (source: http://fr.wikipedia.org).

ETL (Extract-Transform-Load) IT middleware technology that allows large-scale synchronizations of information from one data source (usually a database) to another (source: http://fr.wikipedia.org).

Identifier Sequence of characters allowing a user to be distinguished unambiguously.

Identity federation Technology interconnecting an institution offering an application (service supplier) and the institution that an external user originates in (identity supplier), allowing the latter to be delegated to handle the verification of identities and the propagation of attributes up to the application; in education, mainly based on the SAML2 protocol.

Identity repository Set of databases used within an identity management framework.

Management of identities Process for managing users, their authentication and their privileges within an organization.

MDM (Master Data Management) Management of reference data. IT field that defines a set of concepts and processes aiming to define, store, maintain, distribute and impose a complete, reliable and up-to-date view of the repository data within an information system (source: http://fr.wikipedia.org).

People management application Software for storing the data held about a person in relation to the institution. Example: employees of a company are registered in the human resources management software.

Complete BPDs are available at http://services.geant.net/cbp/Pages/Home.aspx

campus-bp-announcements@geant.org