THE Open GROUP
Making standards work®

Business Scenario: Identifiers in the Enterprise

This Business Scenario was produced by The Open Group for the Core Identifier Work Group, an informal group of members of the Distributed Management Task Force (DMTF), the Network Applications Consortium (NAC), and The Open Group.

The Open Group would like to thank all those that have contributed to this Business Scenario through participation in the workshop or submission of comments. The views expressed in this Business Scenario are, however, not necessarily those of any particular member of The Open Group or any particular contributor to the Business Scenario.

Business Scenario: Identifiers in the Enterprise

Document No.: K061

Published by The Open Group, December 2006

Any comments relating to the material contained in this document may be submitted to:

The Open Group
44 Montgomery St. #960
San Francisco, CA 94104

or by email to:

ogspecs@opengroup.org

Contents

Management Summary

How should an enterprise identify people and things to optimize its operation and facilitate collaboration with other enterprises?

The problems are technical, but they have implications at the business level. The purpose of this Business Scenario is to explain the implications at the business level, in order to get traction for a solution at the technical level.

The technical problems are that:

- Different products and systems require different forms of identifiers.
- Some products and systems allocate identifiers themselves or constrain the choice of identifier for individuals or system resources so that the user organization cannot give an individual or system resource a single identifier that does not change over time and can be used in all instances of the product or system within the organization.
- Technical limitations require identifier changes that are unrelated to business reasons, making the process of identifier management unnecessarily complex.
- Usage of identifiers across organizational boundaries is difficult, complex, and unreliable.

The business implications are that:

- There is a significant cost overhead in translating identifiers between systems and to user-friendly forms.
- Forced identifier changes carry significant cost overheads.
- Service provision is more complex and difficult than is necessary.
- Collaboration between departments and organizations is inhibited.
- Audit and event tracking is more difficult, making it harder to maintain business standards and comply with regulation.

The solution to the problem as stated sounds simple – to let the user organization specify the identifiers. But there are difficulties with this. The need for interworking among systems means that organizations cannot assign identifiers in isolation, and the need for system performance constrains the form of internal system identifiers, so that users cannot assign them arbitrarily.

This Business Scenario puts forward a solution to the problem and lists specific requirements for the components of that solution.

The solution has three components.

1. A documentary framework for existing identifier forms that will help enterprises to manage identifier complexity and to reduce that complexity over time
2. A common identifier form to which existing identifiers can be mapped algorithmically that will enable standardization of system components and interface mechanisms, simplifying enterprise IT architecture
3. A global standard common core identifier for each person or thing that an enterprise needs to identify that will:
    a. Simplify identifier mappings

b. Provide a persistent identifier for security principals that enables responsibility for actions to be established clearly, and as long after the time of the actions as necessary

c. Enable sharing of identifiers across an organization's internal and external boundaries

Standards for the common identifier form and common core identifiers should be implemented in new software by 2008. Retrospective application to legacy software will probably be impossible, but the natural equipment refresh cycle will ensure that they are eventually implemented throughout all systems in every enterprise.

Background to the Business Scenario

This Business Scenario was developed by the Core Identifier Work Group: a joint initiative of The Open Group, the Network Applications Consortium (NAC), and the Distributed Management Task Force (DMTF) in order to develop an understanding of the requirements for core identifiers. It loosely follows the established Business Scenario format (see Part IV of The Open Group Architecture Framework (TOGAF™) [TOGAF]. This Business Scenario builds on an earlier draft Business Scenario developed by the Identity Management Work Area of The Open Group in order to ascertain the requirements for identity management products to support a standard representation of core identity. It is largely based on material from the following sources:

- The Identity Management Business Scenario, published by The Open Group, July 2002 [IDMSCEN]
- The presentations by Lockheed Martin to The Open Group Identity Management Work Area at its conferences in February and April 2004, and the discussions that followed those presentations
- The Core Identity Requirements workshop held at The Open Group Conference in Boston, MA, 22 July 2004
- Presentations and discussions at The Open Group Conference: Architecting Identity Management, January 2005
- The meeting and teleconferences of the Core Identifier Work Group during 2005, especially the requirements gathering teleconferences

The Identity Management Business Scenario captures some of the background, business processes, and actors related to the use of identifiers within an enterprise.

The presentations and discussions in February and April 2004 related to a specific potential solution to the problem. The points that were made about the requirements for that solution are captured in this Business Scenario. The arguments for and against the solution that was proposed are omitted.

The Boston workshop developed the pain points that arise from lack of a core identity representation, and the specific objectives that a core identity representation should meet. The participants in that workshop were: Chris Apple (GlaxoSmithKline), Ilya Burdman (NASA-SEWP), Ian Dobson (The Open Group), Nicki Habluetzel (KSU), Chris Harding (The Open Group), Eva Kuiper (Hewlett-Packard), Mike Litchfield (Iron Mountain), David McCaskill (Procter & Gamble), Shafiq Rahim (Boeing), Skip Slone (Lockheed Martin), and Steve Whitlock (Boeing).

The meeting and teleconferences of the Core Identifier Work Group added major new perspectives from the NAC and the DMTF. Contributors to the requirements work at this stage included Paul Agbabian (Symantec), Greg Blana (Boeing), Merl Ferguson (Progress Energy), Don Hirst (ABN AMRO), Jim Hosmer (Lockheed Martin), Marty Schleiff (Boeing), Skip Slone (Lockheed Martin), and Andrea Westerinen (Cisco).

Business Scenario

Identifiers in the Enterprise

Business Scenario Problem Description

How should an enterprise identify people and things to optimize its operation and facilitate collaboration with other enterprises?

There are too many different ways of identifying people and things; and processes and systems relating to identity have grown up haphazardly and without linkage. This imposes a major overhead on the operation of enterprises today.

There are many different ways of representing identities, and there is a proliferation of name forms across different computer systems. And proliferation can be a problem even within a single one of these categories. For example:

- Some US Government agency employees have to carry about six or seven badges for asserting their identity on different systems in different government agencies.
- Many companies purchase most of their applications. They find that each application handles identity differently and these differences are not readily customizable.

Each person or thing that an enterprise deals with typically has many different identifiers. A person may have a name, an employee number, several computer system user IDs, multiple "systemic" identities (UIDs, SIDs, etc.), an X.500 directory name, several email addresses, and so on. A piece of equipment, a building, or other object that needs identification may have several kinds of formal identifier such as serial number or asset number, plus descriptive characterizations such as "John Doe's PC". A number of identifier forms, including those in common use within enterprises today, are described in Appendix A: Forms of Identifier.

Different products and systems require different forms of identifiers; directories require X.500 names, one kind of operating systems requires UIDs, another requires SIDs, and so on.

In some cases the formats of these identifiers are standardized (X.500 directory name, for example), but in many cases they are not, and the standards that do exist apply only within particular contexts. In particular, there is a lack of standards for identifiers for things (as opposed to people).

As devices become more intelligent, they increasingly need to be treated in a similar way to people. For example, a software program may be a security principal. But systems for managing non-human identifiers generally do not exist. Authoritative sources of information are often lacking, and there is inappropriate use of human identifier systems to manage non-human identifiers.

Processes and systems that depend on identifiers have grown up around the naming conventions in use within enterprises. They have differing levels of maturity in handling identifiers, particularly across organizational boundaries. And they have no systematic way of treating identifiers or relating them to each other.

Some products and systems allocate identifiers themselves. This can be a convenience for the user, especially if the identifiers are propagated automatically over a set of networked products. However, it may mean that the user organization cannot give an individual or system resource a single identifier that is invariant over time and can be used in all instances of the product or system within the organization. Other products and systems constrain the choice of identifier for individuals or system resources; for example, by insisting on numbers within a particular range or on a particular style of directory name. This too can prevent the assignment of persistent, enterprise-wide identifiers.

Technical limitations (such as scope of uniqueness) can require identifier changes that are unrelated to business reasons. This introduces further complexity into the process of identifier management.

It is difficult to manage identifiers consistently within a single enterprise. Usage of identifiers across organizational boundaries is even more difficult, complex, and unreliable.

The complexity results in much confusion; there is no visibility or clarity. It is frequently impossible to tell whether two identifiers refer to the same person or thing.

This confusion translates to cost. The cost of defining and managing human identifiers is known to be large. It is difficult to quantify the cost for other identifiers, but it is significant and increasing. In particular:

- There is a significant cost overhead in translating identifiers between systems and to user-friendly forms.
- Forced identifier changes carry significant cost overheads.

In addition to cost, the complexity and confusion have a major impact on scalability, business process enabling, deployment of systems and services, regulatory compliance, event tracing, and security, and they hamper collaboration between departments and organizations.

Identity management products and standards can make the confusion more manageable, and hence reduce the cost and other adverse impacts. But the greatest benefit will be obtained by dealing with the root of the problem, and reducing the complexity. This can be achieved by introducing standard, common forms of identification for people and things.

Objectives

An enterprise should be able to identify each entity within its range of operations, individually; whether that entity is a person, place, security principal, hardware asset, software asset, or information asset.

While there will be multiple identifiers, and multiple forms of identifier, for a single entity, an enterprise should be able to relate all of the identifiers for a specific entity to each other easily, using an automatic system such as a directory.

Identifiers used for different purposes need different characteristics. There should be a common documentary framework for these identifiers. It should be possible to tell from an identifier and its context how it fits into the framework, and what characteristics it has.

An enterprise should be able to identify every security principal using a standard format that is common to all of the other organizations that it deals with, and with a single identifier in that format, which persists over time. An enterprise should be able to use common authorization systems for people, devices, and applications.

Achievement of these objectives will lead to:

- Lower IT budgets
- Improved IT project times
- Lower production/operation costs
- Improved production quality
- Better compliance with regulation
- Increased customer satisfaction

Standards that enable the above objectives to be realized should be implemented in new software by 2008. Retrospective application to legacy software will probably be impossible, but the natural equipment refresh cycle will ensure that they are eventually implemented throughout all systems in every enterprise.

Views of Environments and Processes

Major Trends

Most of the difficulties of managing identifiers within the enterprise today can be traced back to the major business and technical trends of the last 30 years: the weakening of organizational boundaries, then growing requirements for collaboration among organizations; the introduction of information technology; and the development of system-to-system communications.

Taken together, these trends have led to a massive increase in the use of identifiers and in the role of information technology in handling identifiers. This has made the management of identifiers within the enterprise so complex and difficult that existing systems and frameworks are no longer adequate.

The Boundaryless Organization

The old idea of an organization with a hard boundary, within which resources could be accessed and information could be exchanged freely, and subdivided on a hierarchical principal, has gone. The boundaryless organization, pioneered by Jack Welch and others in the 1970s (see [BNDLESS]), is now the norm. The vertical boundaries between subordinate and superior within the organization, the horizontal boundaries between different functions and departments, the external boundaries between organizations, and the national boundaries between countries have all become much more permeable.

This means that the scope within which an individual person or other entity is known, and therefore must be identified, is very much wider. Many people and computer processes now need to use an individual entity's identifiers.

The need to restrict access of individuals to services and information, for business reasons and for other reasons such as national security, has not gone away, but meeting that need has become more difficult. Access to services and information must be granted on the basis of the attributes of the individual concerned, rather than on the basis of membership of an organization or department.

The Information Revolution

Computerization is now a fact of organizational life. The typewriter has been replaced by the word processor; the filing cabinet has been replaced by the database; the control lever has been replaced by the keyboard; and the gauge has been replaced by the VDU. This has led to the need for people to identify themselves to the computers that they use. In some cases, particularly as computers take over functions and responsibilities of people, there is, as well, a need to identify computers, and computer programs.

Unfortunately, the different computer systems that have been developed for different functions often use different information formats and do not interoperate very well. They are so-called *information silos*. In particular, they often use different forms of identifiers. This means that an individual must have different identifiers, in different forms, for use with different computer systems.

The information revolution is not yet complete. As Drucker points out [DRUCKER], we have automated things that we were doing before, but it has not yet really changed what we do – as was true of the first 50 years of the industrial revolution. As the revolution develops,

rationalization of identifiers will be needed if we are to see radical changes in enterprise processes and decision-making.

The Networked World

The third major trend that has complicated the management of identifiers in the enterprise is the explosion in system-to-system communications. The Internet and the World-Wide Web are technical phenomena that have revolutionized modern life.

The Internet enables systems of all kinds, and in all places, to communicate with each other. Business processes such as exchange of contracts and purchase orders can take place automatically, and technical processes such as computer-aided manufacturing can be integrated. This means that the systems must understand each other's identifiers, and authenticate them automatically.

The Web enables access to all kinds of information by anyone, anywhere, and is becoming the leading vehicle for business-to-customer communication. This means that a Web-enabled system may need to identify an enormous number of individuals and, conversely, an individual may need to provide identifiers to a large number of systems.

Business Drivers

The main business drivers for organizations to manage identities are identified in the Identity Management Business Scenario [IDMSCEN]:

- Efficiency and competitive advantage
- Security
- Support for mobility
- Consistent treatment of the individual
- Conformance to regulation

Identifier simplification is not a revenue generator; it is a cost reducer.

Organizations need to manage the identities of several different kinds of people. For example, one area of a major computer vendor is concerned with:

- Internal identities and being able to switch provisioning for them on and off
- Business partner identifiers for allocation of fine-grained permissions and privileges
- Customer identity information

Identity representations are used by organizations in the course of all of their business processes. Typically, one or more identity representations are assigned when an individual joins an organization, or comes into contact with it. Further representations may be assigned as an individual acquires a new role, or needs to use new equipment or software in order to act.

The same person often has many different roles in a particular organization. He/she may have relationships with several different departments of the organization. This naturally leads to a proliferation of identifiers for the same person and thus a need for mapping and conversion between identity representations.

Organizations generally wish to recognize the same person in different contexts in order to deal with them efficiently and consistently. They may also need to do this to provide an audit trail, for legal or security reasons.

Identity management is a significant issue for large corporations. One major aerospace corporation estimates that 15-20% of its large program development costs center on identity management issues. In a recent project, $2.5 million was devoted to identity management from a total budget of $12.5 million.

Ambiguous and unstable names make it hard for enterprises to manage communications with all the individuals that they have to deal with, both within the enterprises themselves and with their business partners.

Inconsistency and instability in name syntax and semantics is a major contributor to these costs. It makes the establishment of interoperability between and among IT systems a difficult, time-consuming, and costly process.

Inconsistency and instability arise for several reasons. The most important are:

- The need to do business with many different business partners

  Even in the context of business communications where are all the relationships are well-established, and it is not a case of dealing with "people off the street", the number of identities to manage can be enormous. The major aerospace corporation mentioned above has approximately 65,000 trading partners. Even with some optimization and consolidation, with federation, and with the potential use of commercial identity brokers, they are unlikely to reduce the number of identity authorities much below 1000, representing as many as 200,000 security principals. Given that most standards permit options with regard to an authority's choice of identity representation, it is likely that there will be dozens, if not hundreds of variations in approach. Some of their most important identity management enablers are becoming extremely difficult, if not impossible, to implement cost-effectively. In the absence of overarching standards guidance, they fear that the problem is likely to spiral out of control.

- Globalization, and the need to do business in different countries with different cultures

  For example, in America people do not have common root identities, because no one will take the responsibility of issuing them. This contrasts with the situation in Europe, where people living in or visiting countries that have signed the Schengen agreement are obliged to carry identification documents and produce them when requested by the authorities.

- Changes over time.

  Some programs last for many years; for example, there is one military aircraft program that started in 1950, and will probably last until 2050.

- The natural tendency of product designers to do things differently in the absence of a standard

Less important than the desire to improve efficiency, but nonetheless a significant driver, is the desire to mitigate risk of non-conformance to regulation and legislation.

For example, corporations are expected to take reasonable levels of security measures, and there is indeed a European directive that refers to "adequate security". But there is no standard definition of what is reasonable or adequate. An accepted standard would mean that people would not have to take exceptional measures to prove that what they have done is reasonable. Similar considerations apply to the need for responsible use of personal information.

Business Processes

These include:

- The headline business operation processes, such as production, sales, and marketing
- Processes relating to the collaboration of the enterprise with other organizations
- Processes relating to change in the organizational structure of the enterprise
- Support processes that are often carried out by service departments to enable the business units to focus on their business objectives, such as:
    - Human Resource Management
    - Asset Management
    - IT Systems Management
    - Technology Risk Management
    - Permissions Management
    - Security

Business Operation

Each organization carries out a range of activities. Commercial organizations may carry out sales, production, and accounting activities, for example. Any of these activities may require management of identities to be effective; for example, effective sales may depend on management of customers' identities.

Risk management requires identification of assets and processes, people, places, and things.

Collaboration

As illustrated in Figure 1, each enterprise is part of a multi-dimensional collaborative matrix. It serves individual customers. It uses the services of other organizations, including financial, legal, and government organizations. It is part of a number of supply chains. Increasingly, it is likely to collaborate with peer organizations in consortia.

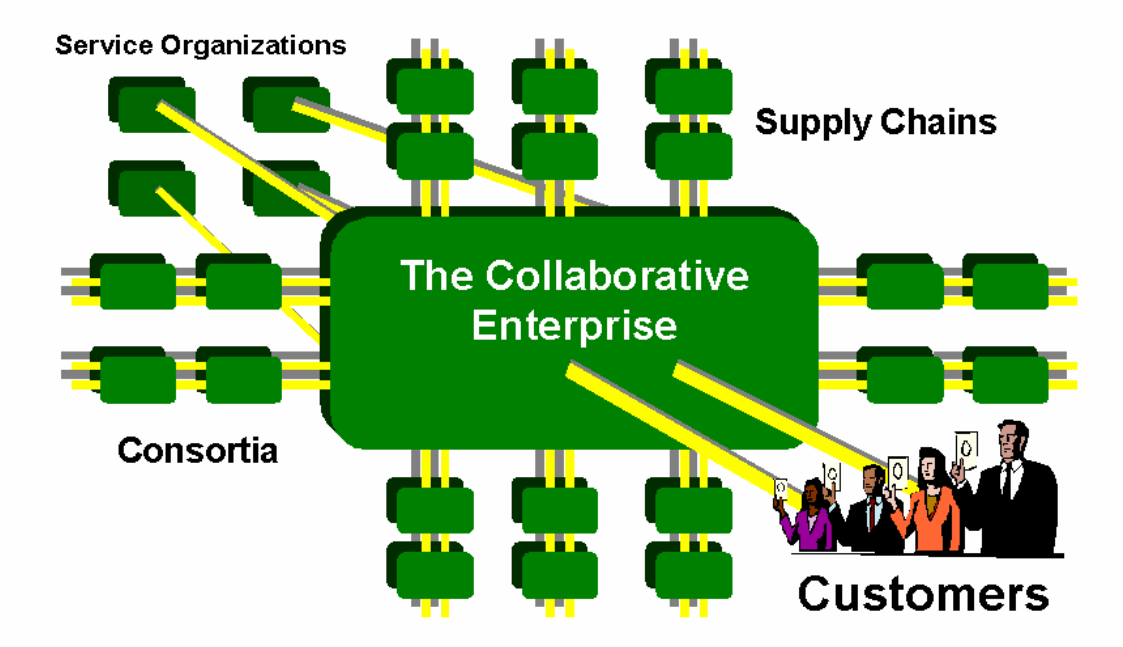

Figure 1: The Collaborative Enterprise

The number of organizations with which an enterprise collaborates can be very large. For example, a major aerospace corporation might have 60,000 trading partners. It needs to deal with employees of those trading partners individually, but does not wish to have to manage identity records for those employees. Management of these records requires a regular check (say, every six months) that the information is still current and correct. Even though such a check might take only a couple of minutes, with (say) 150,000 records to check, the management overhead is substantial. And a couple of minutes per record is only time for a minimal check, while there is a substantial risk associated with having incorrect information that might enable unauthorized access to systems. There is also a problem that the information is constantly changing and, with checks only every six months, it will contain some inaccuracies most of the time.

Because of the difficulty of managing other organizations' identifiers, an enterprise may fail to maintain effective and accurate identity records for many of the people with which it interacts. This implies serious security and business risks.

Employee numbers are often used as identifiers within enterprises, and many enterprise applications use them. When collaborating with other organizations, the enterprise may desire to use the same applications, but it will not want to issue members of these organizations with employee numbers. This presents a problem.

Organizational Change

The lifecycle of an organization may involve the following processes:

- **Formation**. Companies are founded, voluntary organizations form, government administrative areas are created, and so on.
- **Merger**. Companies and other organizations sometimes merge. This may imply combining identity management stores; for example, companies that merge may wish to merge their personnel records, their customer databases, and so on.

- **Split**. An organization can break up into two or more parts. There can be de-mergers as well as mergers. This will generally imply that the organization's identity management stores must be split also.
- **Dissolution**. A company or other organization can be wound up, and cease to exist.

These processes can have major impacts on use of identifiers. A merger or splitting of enterprises can require a major reorganization of certain kinds of identifier: employee numbers, asset identifiers, product identifiers, etc. For example, there was a four-year period of consolidation in the Aerospace sector where the number of companies reduced from 32 to 9. Mergers resulted in collisions in the employee number space, and left some employees in at least one company with two employee numbers.

This can affect not only the enterprises concerned but also their business partners that use identifiers assigned by them.

Dissolution of an enterprise can make it hard, or impossible, to use identifiers that it has assigned.

Support Processes

It is generally the service departments that are most concerned with the problems of identification. They need to identify individual entities of various kinds, especially people, items of equipment, programs, services, plant and machinery, and resources. They need common identifiers to support multiple business operations. With collaboration, these identifiers are increasingly used across organizations.

In one major pharmaceuticals corporation, the impact of identity affects all departments, especially the HR area, which performs account management across many systems. Application developers and HR and the core IT functions are all affected by inefficiencies in identity management. These lead to irritations and wasted time. Greater automation could help significantly.

Service departments may be organization-wide or embedded in individual business areas. For example, some enterprises have a department that is responsible for asset management that is a shared service organization with enterprise-wide scope and reports to the CIO. In other enterprises, asset management is decentralized, and individual business units have responsibility for managing their own assets. In yet other organizations, asset management is broken down by type of asset. For example, in one major communications equipment manufacturer there is one group responsible for chassis, cards, etc. and another responsible for power supplies; fragmentation is not by business unit but by application.

Human Resource Management

This involves the management of people joining and leaving the company, benefits, pensions, and so on. It requires identification of the people in the organization.

For an individual, membership of an organization typically involves the following processes, as described in the Identity Management Business Scenario [IDMSCEN].

- **Join Community**. For example: a new employee joins a business, a new customer buys a product online, a new citizen is born.
- **Acquire Role**. Within a community, each individual may take on various roles. An employee can be appointed as a salesman, production manager, HR director, or whatever. A citizen can become a voter.

- **Act in Role**. A role can convey rights to access information and services (and, of course, can also include duties). A salesman can access the customer database; the HR director can modify personnel records. A voter may (in the future, in most places) be able to vote electronically.
- **Give up Role**. Roles are temporary. People quite often change their jobs and other roles.
- **Leave Community**. Employees resign, customers stop buying products, citizens die.

Organizations are concerned with the above "individual" processes, playing a complementary part to that of the individual.

Managing identity representations when people leave an organization or change roles is a major problem in many organizations. It is difficult to set up procedures that work well in all situations. For example, in one area of a major computer vendor they found that when someone changed status (e.g., temporary to permanent staff; partner to reseller) their employee number had to change, and they were issued with another ID even though there was no change in their use of systems and services.

Asset Management

This requires identification of assets, which may include software applications as well as equipment and other physical assets.

The number of software applications used by an enterprise can be very large. For example, there is a business division of one major aerospace corporation that has 5,000 applications. (It is trying to reduce the number to about 700.)

An item might have different user-friendly names, but the goal is not to record it differently in different contexts for asset management purposes. Having a single identifier as far as possible is desirable, but it is reasonable to have multiple identifiers provided so that it is possible to map between them when necessary.

IT Systems Management

Software license management requires identification of software products in use.

Suppliers of software products need identifiers against which to charge.

Currently, people may look for patterns, presence of files, registry entries, etc. in order to track software assets.

Technology Risk Management

Standardized system names simplify the tasks of technology risk management and incident recognition and reporting.

Standardized domain name structures are required for systems addressing and name/address conversion. Such structures simplify the recognition of rogue systems and attackers.

Standard user ID formats simplify the recognition of valid system users and discrimination against rogue users or attackers.

Standardized identifiers simplify the linkage of IT assets to owners, reducing operational risks associated with IT management.

Use of Identifiers by Business Processes

Identifiers are used by business processes for many purposes. The main ones are:

- Authorization
- Information Provision
- Management
- Event Tracing
- Signature

Different information about an entity may be known or recorded at different times. However, an identifier by which to retrieve an instance is needed as soon as that instance is created.

The different contexts for use of identifiers mean that different forms of identifier with different characteristics are needed.

No identifier format guarantees the association between the identifier and a particular subject, or that a particular organization has issued the identifier, or anything else about the identifier. It is up to the organization using the identifier to satisfy itself on these points. Some identifier formats (such as HIP) may assist the user organization to satisfy itself, and the manner in which an identifier is conveyed (for example, within a PKI certificate) may assist also. But the ultimate responsibility lies with the user organization.

Authorization

Authorization to access or use resources, programs, and services is granted on the basis of roles, permissions, or other attributes that are associated with entities (and therefore with their identifiers). The setting up of authorization systems, and the management of attributes associated with access to resources (permissions management) can be complex and difficult. A particular aspect of this is establishing relations, and federation, with business partners.

Figure 2: Authorization

Increasingly, authorization must cater for devices, as well as for people. For example, one major aerospace corporation has a big permissions management system for people, and is starting to develop one for devices. But it does not want different systems and therefore wants common identifiers.

Some companies do not need to track events or keep an audit trail, but they do need to authenticate and authorize users. They may also need permissions management, which is much more than access control, involving transfers of liability, permission to launch, transfer funds, and the like. Authentication, authorization, and permissions management can be very difficult, both for the organization and the individual, when each individual has many identifiers.

For example, one major information management corporation needs to know to what applications customers have access. Their present inability to support single sign-on makes it very difficult for them to provide acceptable service that supports credentials in a holistic way.

Information Provision

Ability to provide information about something generally requires that the entities associated with that something are identified. For example, providing information about a campus often requires the buildings on the campus to be identified, as in Figure 3.

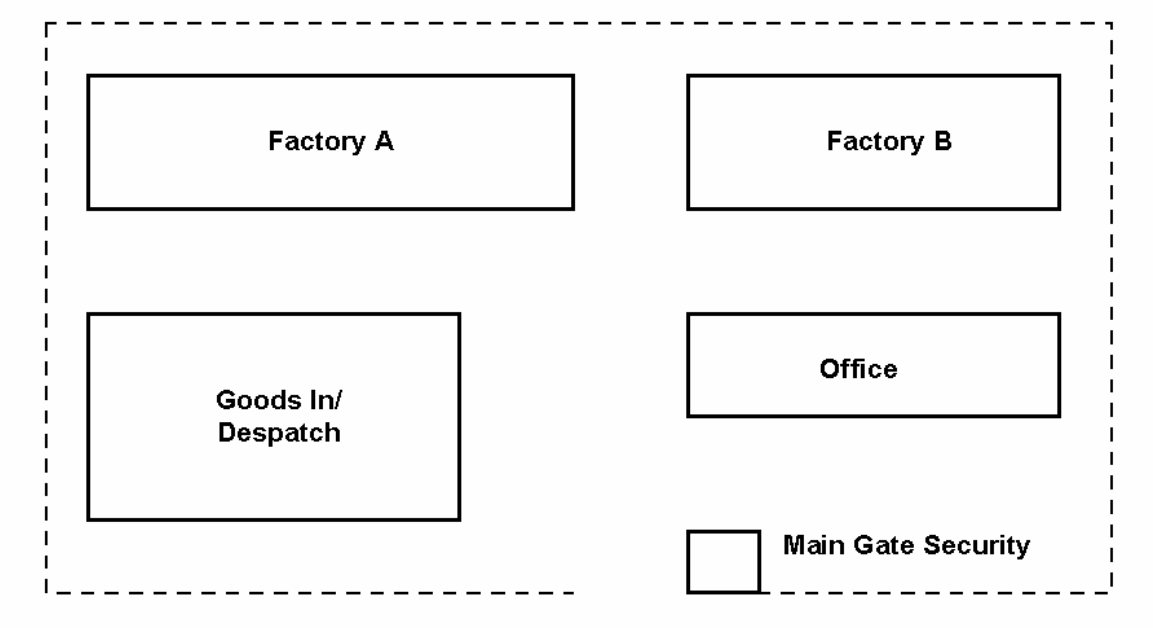

Figure 3: A Campus Map

Management

If something cannot be named, then it cannot be measured, and it cannot be controlled. This presents a management problem. And if the costs associated with it cannot be controlled, then there is a serious business problem.

Event Tracing

An enterprise may need to trace a trail of events over time, and identify the entities that were affected and responsible, for several reasons. These include analyzing the causes of a disaster, and tracking a security breach.

There is need for traceable and trusted identity in many areas, such as financial services and healthcare.

The US government currently uses directories to authorize rights to perform operations. It tracks these operations, both for people inside each agency and for external people (business partners).

A company supplying equipment to the US Government may have to track all operations and events. This is vital for traceability on critical operations where something may fail, and tracking back to trace the source and supplier of the component or operation at the point of failure is essential so that appropriate measures can be taken to correct the problem.

Tracking people or things across different systems and departments is cumbersome. A department of one major computer vendor, for example, needs to keep track of each customer as that

customer makes deals with different departments. This kind of tracking incurs significant extra cost and time, which can be a major issue.

It is not just a case of dealing with point-to-point transactions. Transactions can involve multiple parties, and may require traceability over several years. Also, credentials issued by varied mechanisms do not deal with reconciliation of trust. It is not sufficient to do reconciliation at the perimeter of the organization; the communications must pass through to internal systems, and it is necessary to trace transactions across domains.

The events in a trail may relate to multiple enterprises. This is illustrated in Figure 4.

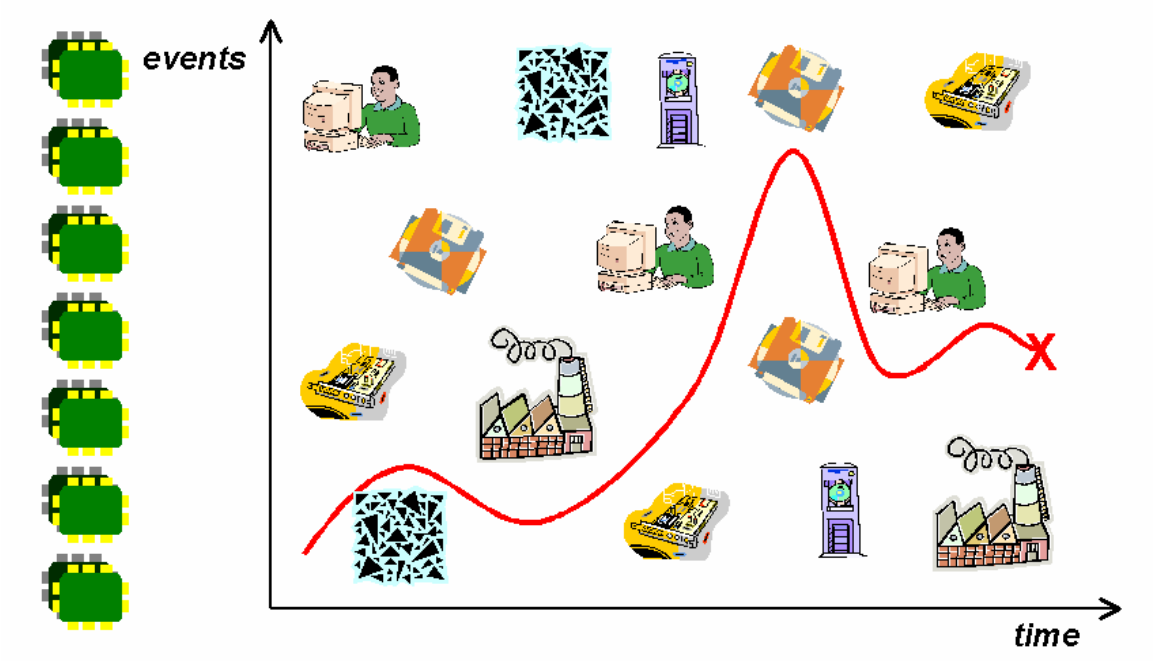

Figure 4: Event Tracing

Signature

Signatures provide reliable assertions of authority. Identifiers may be used in digital signatures.

Note that the value of a signature may be lost when the identifier is translated.

Technical Processes

There are many technical processes related to identifiers.

Some of them require exchange of identity information between different systems, sometimes in different organizations. Often, the standard protocols for these exchanges (for example, the Security Assertion Mark-up Language, SAML) use particular identifier forms. As these are not necessarily the forms within use in the communicating systems, these exchanges can introduce further complexity to the problem of identifier management within the enterprise.

Identifiers are often used as keys for database or directory searches. Ability to support efficient indexing algorithms is therefore important.

Identity Management

The Open Group Identity Management White Paper [IDMWP] contains further information on identity management technical processes.

Registration

Registration is the process of making an individual known to a system. It includes validation of the individual's identity. For example, registration of a customer at a bank may include verification of the customer's address, and checks with people that know the customer of the customer's financial standing and *bona fides*.

Provisioning

Provisioning is the process of configuring accounts and identity information in resources for the purpose of controlling access to them.

There are three key aspects of provisioning:

- *Account provisioning* deals with identity-related information associated with individuals, their personal attributes, affiliations, and so on.
- *Resource provisioning* deals with business assets such as computers, databases, applications, and with the management of permissions associated with those assets.
- *Account de-provisioning* deals with the termination of access rights to systems and services and re-allocation of those systems and services.

Provisioning a new employee can be a complex process, and automating it is a common requirement. For example, a company might want a new employee to be able to go to a PC and follow a simple set of instructions that will within a few minutes give him/her access to all the employee facilities that he/she should have – including payroll, pension, parking, other employee benefits – as well as IT access to the applications and data required to do his/her job. They might also want to be able to follow a simple keying process to just as easily de-provision them within minutes of them leaving the organization.

Identity Information Update

Since identity information is used by many applications and system processes, and particularly by the security infrastructure, it is important to keep this information up-to-date. Authoritative sources should be determined, and systems put in place to ensure that current information from those sources is held by the identity management infrastructure.

For much identity information (home address and telephone number, for example) the authoritative source is the individual concerned, and it is often good administrative policy to give individuals the ability to update their information directly, without having to go through HR or system support departments. Other information (such as permission to access particular services) must, however, be the responsibility of these departments.

Identity Information Access

Access to identity information is needed by individuals (for example, to look up contact details for their colleagues) and by applications and infrastructure services. A common way of providing such access is through the Lightweight Directory Access Protocol (LDAP), used directly by applications and infrastructure services, and via directory clients by people.

Identity Information Archive

Identity information that is not in current use may still need to be kept; for example, to satisfy legal constraints, or to enable responsibility for past events to be established. Information relating to an individual may need to be retained after that individual leaves the organization; even after the individual has died, in some cases.

Synchronization

Identity *synchronization* is the process of copying identity information (especially passwords) between identity stores and resources in order to create a consistent set of identity information. It includes both replication of information between identity stores and projection of identity store information onto resources.

Identifier Mapping

The most significant problem that organizations have relating to identifiers is that of name proliferation. An organization has many names for a thing or a person, and they need to be mapped together for the technology to be used effectively. Mapping works in a bilateral situation, but in multilateral and multilevel multilateral situations it becomes unmanageable, especially when many business partners are involved.

Where an enterprise has a large number of different identifiers, the effort of defining mappings between them can be so great that the enterprise simply cannot undertake the task.

Determination of whether two identifiers refer to the same entity should be possible using a simple computer process, of the order of complexity of a directory look-up or evaluation of an arithmetic expression.

Determination of the identifier in a particular class that corresponds to a particular identifier of another class (for example, of which employee number corresponds to a particular email address) should be equally straightforward.

Federation

Identity *federation* is a standard way of allowing enterprises to provide services directly for people registered at other (partner) enterprises. It can also be used within an enterprise between departments or divisions with different identity management systems. And it can apply to non-human entities, such as applications, as well as to people.

Within a federation of services, an enterprise (or department) can obtain trusted information about a user from the user's home organization (or information-providing service). The enterprise does not need to register and maintain that user's identity, and the user is spared from having to obtain and remember a new login in order to interact with the enterprise.

A federation system creates associations between sets of identity information – including information held by different organizations – to enable authentication and access control systems to support this kind of federated operation.

Permissions Management

Permissions management refers to the management of information about what entities should be allowed to do. Appropriate use of resources is assured through the management and enforcement of permissions associated with those resources. Permissions include access permissions and more: they include permission to read, compare, write, modify, create, destroy, execute, copy, print,

forward, delegate, purchase, authorize, approve, sell, sublease, assign, transfer, hire, fire, promote, and so on.

Security

Access Control

Access control refers to the control of access to resources. It includes the determination of whether an entity may use a resource (authorization) and the enforcement of the result of that determination (access permission enforcement). It may be carried out by dedicated access control components, by access control functionality within resources, or by a combination of these two methods.

Different kinds of access – for example, read access and modify access – may be subject to access control, and an entity may be granted some kinds of access and denied others, to the same resource.

Different kinds of access apply to different resources. A file system might have read, write, and modify access, while an application might have user and supervisor access, for example.

Access control is closely related to authorization and permissions management. The meanings of these terms overlap, and they are sometimes used interchangeably.

Authorization

The term *authorization* has two distinct meanings.

1. *Authorization* is the process of determining whether an entity should be allowed to do something. In this respect, authorization to access resources is an aspect of access control.
2. *Authorization* is the process of assigning permissions to entities.

Because of the potential for confusion arising from the term having two meanings, it is not used in The Open Group Identity Management Technical Reference Model.

Authentication

Authentication is the process of establishing confidence in the truth of some claim. In the context of identity management, an authentication system provides an understood level of confidence that an identifier refers to a specific individual (*individual authentication*) or identity (*identity authentication*), or that an attribute applies to a specific individual (*attribute authentication*).

Authentication may be carried out by dedicated authentication components, by authentication functionality within resources, or by a combination of these two methods.

Communications Confidentiality and Integrity

Encryption mechanisms are used to ensure confidentiality and integrity of computer communications. These mechanisms may require the identification of the security principals and/or systems involved. For example, in the commonly-used SSL protocol a digital certificate is generally supplied by the server to the client, and may be used by the client to verify the server's identity.

Audit

Generation of audit trails during transactions can enable the cause of security breaches to be identified and corrective action to be taken.

It is important that the security principals concerned are correctly and consistently identified in audit trails.

Digital Signature

Digital signature provides verifiable indication of authorship or responsibility. It typically uses public key technology, often sharing a common Public Key Infrastructure (PKI) with security components.

Application Development

Application developers should be able to exploit operating system calls, such as GSS-API, to request security services, which will then just work.

In the absence of a core identity standard, developers will create their own solutions. This gives problems for the development group and implies further problems for users downstream and (where the developers are developing an organization's products) for customers.

Infrastructure Evolution

When an enterprise wants to change its technical infrastructure it will often set up a cross-functional team to address this. Cross-functional teams focus on particular topics, such as setting up systems to support federated identity, or systems that deploy multiple platforms. Members of such a team must worry about identifiers.

Actors and their Roles and Responsibilities

Human Actors and Roles

The human actors and their roles are listed in Table 1.

Table 1: Human Actors and their Roles

Human Actor	Role(s)
Individual	Has identifiers.
Business Manager	Manages business risks: credit, banking, etc. Needs to identify the parties dealt with.
HR Person	Manages the interaction of employees with an organization: people joining and leaving the company, benefits and pensions, etc. May be responsible for some identity-related information, especially a person's status as an employee.
IT Operations Manager	Responsible for operation of an organization's communication and information infrastructure. May be a dedicated support person, or someone with another "day job" who supports equipment part-time.
Facilities Manager or Asset Manager	Responsible for management of buildings, plant, and equipment. Needs these things to be identified in order to manage them.
Technology Risk Manager	Ensures security.
Developer or Maintainer of Tools and Applications	Designs and implements tools and applications that use identifiers.

Computer Actors and Roles

The computer actors and their roles are listed in Table 2.

Table 2: Computer Actors and their Roles

Computer Actor	Role(s)
Individual System Component, such as a card	Has identifiers. (These might not necessarily be global, but unique within a context.)
Resources, Services, and Applications	Including operating systems, database management systems, web services, and enterprise applications. Use identifiers for access control or to provide functionality.
Identity Stores	Including stores holding information about people (directories) and stores holding information about other kinds of entity, such as application registries used by asset management systems. Hold items of information associated with identities. Such items can be an identifier, a credential, a permission or role, or an item related to a specific identity-enabled resource.
Identity Management System	Including registration, provisioning, identity information update/ access/archive, identity synchronization, identifier mapping, and federation. Manipulates or manages identity information, and supports identity management processes.

Computer Actor	Role(s)
Security System	Including authentication, authorization, and access control systems, and systems responsible for communications confidentiality and integrity. Uses identifiers to identify the objects being secured, and the security principals that use them.
Secure ID Device	Helps user establish his/her identity. Examples are: • Challenge/response devices that generate time-dependent identification codes • Certificate-bearing smart cards • Magnetic stripe cards • Biometric characteristic (e.g., Fingerprint) readers
PKI System	Manages certificates that include identifiers and give assigned bindings to them.
Regulatory Compliance System	Uses identifiers to identify the people, information items, and systems affected by the regulation (e.g., HIPAA or SOX).
IT Management System	Uses identifiers to identify the systems and components being managed.
HR System	Uses identifiers to identify the people being managed.

Organization Actors and Roles

The kinds of organization that are relevant to the problem are listed in Table 3.

Table 3: Organization Actors and their Roles

Organization Actor	Role(s)
Partner Enterprise	Has employees, and possibly things, that the enterprise needs to identify.
Standards Body	Defines/maintains standards that include or refer to identifiers (especially the IETF, The Open Group, the DMTF, the NAC, and the OMG).
Product Supplier	Designs, develops, and sells products such as operating systems that use identifiers. Needs to identify instances of products in use for charging purposes.

Technical Solution

The objectives set out in this Business Scenario can be met by:

1. A documentary framework for existing identifier forms that will help enterprises to manage their complexity and to reduce that complexity over time
2. A common identifier form to which existing identifiers can be mapped mechanically that will enable standardization of system components and interface mechanisms, simplifying the enterprise IT architecture
3. A global standard common core identifier for each person or thing that an enterprise needs to identify that will:
    a. Simplify identifier mappings within the enterprise by enabling all other identifiers to be mapped to the core identifier (a "2n problem") rather than being mapped to each other (an "n^2 problem")
    b. Provide a persistent identifier for security principals that enables responsibility for actions to be established clearly across the enterprise, and as long after the time of the actions as necessary
    c. Enable sharing of identifiers across an organization's internal and external boundaries

Documentary Framework

The documentary framework will:

- Provide a common understanding of identifiers, removing confusion
- Describe guidelines, algorithms, and common semantics
- Be a reference point for identifier classifications and how they are used
- Enable simplification over time

Common Identifier Form

The common identifier form will also enable simplification. In addition, it will:

- Support mappings between identifiers
- Avoid the need to rename resources, as most resources already have identifiers
- Help clients to identify resources correctly
- Enable the evolution of more efficient ways of using identifiers in systems and processes
- Enable exchange of identifiers between collaborating organizations
- Enable interoperability

Applications will need to understand the identifiers of this form, and parse them in order to determine how to interact with them.

Identifiers of this form will be:

- Able to be associated with roles, permissions, and other attributes for authorization purposes
- Usable for information provisioning, management, and event tracing

- Able to serve as the subjects of digital signatures

Existing identifier forms will continue to exist, but will be correlated with the common identifier form.

XRI (see [XRI]) is an appropriate standard for the common identifier form, and its adoption for this purpose is recommended.

Common Core Identifiers

A *core identifier* is an identifier that has the irreducible minimum of attributes, sufficient to distinguish its subject within the scope of a naming authority, and to identify that authority. A *common core identifier* is one that can be used between different organizations.

Common core identifiers can:

- Reduce the number of identifier mappings needed by an organization, since it is possible to map a core identifier to all other identifiers for the same entity
- Serve as persistent identifiers for security principals that can be embedded in operating systems and used for access control, without users needing to know they are there
- Improve traceability, including across organizational boundaries
- Be a basis for large-scale federation and simplified sign-on
- Reduce double-counting of assets

The aim is not to assign a unique identifier to each individual at birth that will remain with him/her forever, and be used in all dealings with other individuals and organizations. This is not practical, or even desirable. Even within a single organization, the proposal is not for a sole identifier, but for a core identity to which aliases can be mapped. (There will usually, however, be advantages in using the core identity rather than an alias.)

An enterprise will usually have a single common core identifier for a subject at any given time, but:

- Different enterprises may have different common core identifiers for the same subject at the same time.
- The same enterprise may have different common core identifiers for the same subject at different times.
- Identifiers other than common core identifiers will often be exchanged between enterprises.

The concept of a common core identifier does not remove the need for identity federation between organizations. An individual may have a different common core identifier for each organization to which he/she belongs or relates. Organizations may still wish to develop trust relationships and link each other's identity representations by federation.

Common core identifiers will simplify identity management within user organizations. The significant benefits will come when a standard common core identifier format is adopted by vendors of identity-enabled products. This will improve compatibility and interoperability between products, reduce the need for "glue" software that converts between different products' identity representations, and make customers' identity management systems less complex. The result will be systems that are easier to manage and more secure.

Requirements

Documentary Framework

The documentary framework must:

1. Comprehend all important existing identifier forms used by enterprises
2. Allow for the definition of new forms
3. Explain identifier characteristics and attributes
4. Include the common identifier form and core identifiers
5. Be an authoritative reference
6. Be easy to read and understand.

Common Identifier Form

The common identifier form must:

1. Allow an entity to have multiple identifiers
2. Be able to be handled by computer programs that do not require direct participation of people in the processes (except possibly in exceptional circumstances)
3. Map algorithmically (*not* including table lookups, and in conformance with agreed standards) to existing syntaxes for identifiers in use within enterprises, such as:
    a. User-friendly identifiers
    b. Short-form identifiers that can be conveyed verbally
    c. Long-form identifiers that are guaranteed unique
    d. Systemic identifiers
    e. Identifiers that support specific requirements; e.g., HIP identifiers for Secure Mobile Architecture (SMA)
4. Allow for new identifiers that support innovative built-in functionalities
5. Enable some attributes of the identified entity to be determined by inspection of the identifier, where appropriate, but also allow for opaque identifiers to protect privacy
6. Comprehend identifiers with different characteristics, and enable some characteristics of the identifier to be determined by inspection of it where appropriate, including:
    a. The authority responsible for issuing the identifier
    b. The process by which the identifier can be resolved to discover further information about its subject and its issuing authority
    c. Whether the identifier is static (e.g., to support personalization) or dynamic (e.g., to avoid profiling)
    d. Whether the identifier is permanent or re-assignable (e.g., for finite or dynamic namespaces)

7. Have a standard process for resolution to discover further information about its subject and its issuing authority, noting that:
    a. Determination of the issuing organization cannot be guaranteed (for example, it may have been issued by a company that has gone out of business and no longer exists).
    b. It must be possible to control the amount of information about the subject that can be discovered.
8. Be portable (capable of being issued by one organization and used by others) based on cross-organization standards
9. Be independent of how the subject is accessed (for example, the identifier for a file should not depend on whether the file is accessed via a file manager or via the web)

Common Core Identifiers

Core identifiers must:

1. Be portable – able to be issued by one organization and used by others – based on cross-organization standards
2. Have a clear, unambiguous name form
3. Convey no meaning other than that they identify someone or something – there should be no need to parse names
4. Impose no constraints on directory namespace
5. Be easily generated without reliance on complex interactions with some central authority
6. Not be tied to any language or cultural environment
7. Be flexible enough to accommodate different business models
8. Be able to be integrated into single sign-on systems where security and privacy of the identifier information is critical
9. Allow for the fact that an individual is usually represented by some authority that holds sway over him – his credit card company, his government, etc.
10. Be compatible with federated identity standards
11. Be applicable to things as well as to people – anything that needs to be subject to access control policy, not just a person, can be a security principal
12. Be applicable to groups as well as to individuals
13. Allow for anonymity – there is a need for "friendly handles" that can be used to refer to people in transactions without revealing their real identities; anonymity can be a requirement in some cases
14. Provide for processing efficiency (for example, fixed length identifiers are more efficient in some situations)

Common core identifiers must, in addition:

1. Be persistent over time
2. Uniquely distinguish an entity within a global scope

3. Uniquely distinguish the issuing authority, which is within the same scope
4. Be capable of representation in common identifier form syntax
5. Be assured of interoperability among domains or systems, according to agreed standards and related policy

The definition of common core identifiers should leverage existing technology where feasible.

Fixed length would be a desirable characteristic.

Appendix A: Forms of Identifier

This appendix describes forms of identifier in common use in enterprise IT systems plus some other identifier forms that, while not in common use at this time, are important because of their underlying concepts or the possibility that they will become commonly used in future.

Further information on the identifiers described in this appendix can be found from the following sources:

- Internet Engineering Task Force (IETF) Requests for Comments (RFCs) can be obtained from the IETF at: www.ietf.org/rfc.html.
- The Single UNIX® Specification and other UNIX system documents can be obtained from The Open Group at: www.opengroup.org/bookstore/catalog/un.htm.
- Microsoft Windows documentation can be obtained from the MSDN Library at: msdn.microsoft.com/library.
- DCE documentation can be obtained from The Open Group at: www.opengroup.org/bookstore/catalog/dz.htm.
- Recommendations of the International Telecommunications Union Telecommunications Standardization Section (ITU-T) can be obtained from the ITU at: www.itu.int/publications.
- Information on the Unique Identification (UID) system of the US Department of Defense (DoD) can be obtained from the US DoD Defense Procurement and Acquisition Policy website at: www.acq.osd.mil/dpap/UID.
- Information on the Extensible Resource Identifier (XRI) is available from the OASIS XRI Technical Committee website at: www.oasis-open.org/committees/tc_home.php?wg_abbrev=xri.

Commonly Used Identifier Forms

The following are some of the identifier forms for people and other entities that are currently in frequent use in enterprises. Note that they are all identifiers for instances of classes. Identifiers for classes of item, such as Universal Product Code (UPC), are not included.

Personal Names

The original way of identifying people; “Thomas Atkins”, “Jane Doe”, etc.

Employee Numbers

Because common names are often not unique, many businesses also identify their employees with assigned numbers that are unique within the organization.

Asset Identifiers

Many enterprises identify individual assets, including such things as computers used by staff, by character strings that are unique within the enterprise and in a format defined by the enterprise.

Product Identifiers

Manufacturers and product vendors often identify individual product instances by serial numbers or other kinds of character strings, in formats that they define. This can include applications and other software products, as well as physical products such as computers, chassis, cards, etc.

Operating System User Names

In most operating systems, users are identified by user names that are unstructured alphanumeric strings, arbitrarily assigned (typically, chosen by the user or by the system administrator).

While user names are employed at the user interface, other identifiers that are not "user-friendly" may be employed internally. For example, UNIX has numeric user identities (UIDs) and group ids (GIDs), and Windows™ has 128-bit GUIDs (see under "UUIDs" below).

UNIX UIDs and GIDs are unique only within the scope of the directory domain and/or Kerberos realm in which they are issued, but common usage generally does not include reference to domain or realm. This means that there is a high probability of overlap, and no possibility of globalization.

Email Addresses

An Internet email address (as defined in [IETF RFC 822]) consists of a local part and a domain, separated by an "@" symbol.

The local part is an unstructured alphanumeric string (typically, chosen by the address owner or by his/her service provider).

The domain part is a DNS name. This consists of a sequence of zero or more sub-domain names, followed by a domain name and a generic top-level domain name, and separated by period characters. The generic top-level domain names are assigned by the Internet Corporation for Assigned Names and Numbers (ICANN). Domain names are arbitrary character strings, assigned by the domain owners, and registered with ICANN to ensure uniqueness within the generic top-level domains. Sub-domain names are arbitrary character strings, assigned by the domain owners.

The process of registering domain names ensures uniqueness of email addresses, provided that the domain owners assign sub-domain names uniquely and take steps to ensure uniqueness of local parts within their domains and sub-domains.

The addresses are unique at any point in time, but re-use of local parts, sub-domain names, and domain names means that different principals may have the same address at different times.

Universal Principal Names (UPNs)

UPNs are user-friendly unique identifiers used in Microsoft® products. Their format is based on IETF RFC 822: a UPN is composed of the user logon name and the UPN suffix joined by the "@" symbol. It can be used to log on to a Windows domain (Windows 2000 or later).

IETF RFC 822 Name or Universal Principal Name is a popular workaround in PKI, but is unstable. When an account is moved into another account domain, all issued long-life credentials must be re-issued. More importantly, companies will not wish to expose their UPN domains, which will make trust verification difficult or impossible.

X.500 Distinguished Names

The ITU-T X.500 recommendations define a model for directory services, with a hierarchical namespace. The model and name format is also assumed by the Lightweight Directory Access Protocol (LDAP) defined by the IETF.

Each directory entry has a number of attributes, and one or more of these attributes define its Relative Distinguished Name (RDN). Their values uniquely identify the entry at its level of the hierarchy, by distinguishing it from its siblings. The Distinguished Name (DN) of an entry is the chain of RDNs that starts with the root of the hierarchy and ends with the entry.

Typically, the root would be a country (e.g., “c=US”), the next level would be an organization (e.g., “o=Acme Computing”), and the next level (or levels) would be organizational units (e.g., “ou=Sales”). Within an organizational unit, a person might be identified by his/her common name (e.g., “cn=John Doe”). So a DN might be “cn=John Doe, ou=Sales, o=Acme Computing, c=US”.

A person (or other entity with a directory entry) is identified by the DN(s) of its directory entry(ies).

The original X.500 recommendations envisioned a single, distributed global directory, with the DN forming a unique global identifier. This has proved impractical, and modern practice, particularly with LDAP, is to regard each directory, or related set of directories within an organization, as being separate. DNs are thus not guaranteed to be unique, although they are likely to be so.

In practice, DNs are highly unstable and change whenever an object is moved in the directory. Directories are restructured from time to time, new branches of the tree are created for new policy differentiations, and items are moved between branches. Also, many organizations find that they need to give each individual two names: a “flat” one for public consumption, and a “hierarchical” one for internal consumption. A further complication is that there are two commonly used variants of this form: the original X.500 organizational variant, and DC naming (see below). All of this means that the DN is not a reliable identifier.

Domain Component (DC) Names

DC naming is a variant of the X.500 naming structure that is often used in the context of LDAP. It follows the X.500 naming rules, but instead of using the “country”, “organization”, etc. attributes that are normal for X.500, it used the DC attribute whose values are Internet domain components. So a DN in DC naming might be “cn=John Doe, dc=acmecomputing, dc=com”.

Universal Unique Identifiers (UUIDs)

A UUID is a 128-bit identifier that is generated in accordance with an algorithm that is designed to ensure that each UUID is unique in space and time. UUIDs were originally used in the Network Computing System (NCS) and later in the Distributed Computing Environment (DCE) defined by the Open Software Foundation (OSF). The DCE definition of the UUID concept (see www.opengroup.org/onlinepubs/9629399/apdxa.htm) became generally accepted, and is the basis of definitions in ISO/IEC 11578:1996 [ISO/IEC 11578] and in IETF standardization work (the current draft is at www.ietf.org/internet-drafts/draft-mealling-uuid-urn-03.txt). There are, however, other variants, including Microsoft Global Unique Identifiers (GUIDs).

There are two main components of a UUID that help to ensure uniqueness. The first relates to where the UUID was generated, and the second to when it was generated. In the DCE version, the IEEE 802.1 node identifier (Ethernet MAC address) of the generating device forms the first

component, and a timestamp forms the second component. There are other ways of defining the first component, including randomly-generated numbers and hash values derived from system names. There are rules for timestamping to ensure that the timestamp component is unique within a given system.

The UUID generation process does not mathematically guarantee uniqueness, but the probability of a duplicate is negligible for practical purposes. The advantage of the process is that it does not assume a central registration authority.

Security Identifiers (SIDs)

SIDs are variable-length names used in Microsoft products to uniquely identify users or groups. A SID includes a revision level component, a 48-bit identifier authority value that identifies the authority that issued the SID, and a variable number of sub-authority or relative identifier (RID) values that uniquely identify the trustee relative to the authority that issued the SID.

The combination of the identifier authority value and the sub-authority values ensures that no two SIDs will be the same, even if two different SID-issuing authorities issue the same combination of RID values. Each SID-issuing authority issues a given RID only once.

While a SID reflects the source of authority and "subject", the source of authority is not globally visible, and the format is a proprietary one.

DCE Names

The Distributed Computing Environment (DCE) incorporates a concept of federated naming, with a hierarchy of composite namespaces. The global namespace is at the top level. It provides for a universally unique root and contains cell namespaces as subordinates. Particular entries within the global namespace identify *cell namespaces*, which are subordinates of the global namespace. Cell namespaces can also be subordinate to other cell namespaces in a configuration called a *cell hierarchy*. The top-level cell in a cell hierarchy, the *parent cell*, is always catalogued in the global namespace. The namespaces of the *child cells* are subordinates of the namespace of the parent cell.

Principal identities are represented by both user-friendly cell and principal (string) names and by Universal Unique Identifiers (UUIDs). Group identities are represented in a similar fashion.

The user-friendly cell and principal (string) names are derived as follows. A parent cell is identified either by an X.500 Distinguished Name or by a DNS name. A child cell is identified within its parent by an alphanumeric atomic name. A subordinate is identified by adding an alphanumeric atomic name to the name of its superior (separated by a slash character). A principal is identified within a parent cell by adding an alphanumeric atomic name to the name of its cell (separated by a slash character). A principal is identified globally by the combination of this name and its parent cell name.

UUIDs are used internally to identify principals, and other entities, within DCE. Note, however, that for authorization purposes, identities in DCE are represented not by a *single* UUID, but by a *pair* of UUIDs: <Cell UUID, Subject UUID> (where "subject" is "principal" or "group"). This provides greater protection to the overall system in the event that security in a cell becomes compromised.

Uniform Resource Identifiers (URIs) and Internationalized Resource Identifiers (IRIs)

URIs (see [IETF RFC 3986]) and their internationalized version, IRIs (see [IETF RFC 3987]) are identifiers for abstract or physical resources that are commonly used in the context of the World-Wide Web. In particular, they are used to identify web pages and web services. Uniform Resource Locators (URLs) are a subset of URIs that, in addition to identifying their subjects, enable them to be located on the Web. (Note that the term "URL" is now deprecated by W3C, even though it is still in common use. The official term is now "http URI"; see [IETF RFC 3305].)

Uniform Resource Names (URNs) are another subset of URIs; they are required to remain globally unique and persistent even when the resource ceases to exist or becomes unavailable.

The syntax of URIs is defined in IETF RFC 3986. A URI contains the name of the scheme being used (<scheme>) followed by a colon (":") and then a string (the <scheme-specific-part>) whose interpretation depends on the scheme. The scheme-specific-part need not have any general structure or set of semantics which is common among all URIs, but a subset of URIs do share a common syntax that is a sequence of four main components: <scheme>://<authority><path>?<query>, each of which, except <scheme>, may be absent from a particular URI. The <authority> component identifies a naming authority, such that the namespace defined by the remainder of the URI is governed by that authority. The path component contains data, specific to the authority (or the scheme if there is no authority component), identifying the resource within the scope of that scheme and authority. The query component is a string of information to be interpreted by the resource.

For example, *https://www.opengroup.org/cgi-bin/dbcgi?TPL=pub_reg* is a URI that identifies The Open Group web page through which a user of The Open Group's website can request a username and password. It is also a URL, enabling the page to be located by the https protocol. In this example:

- "https" (Secure HyperText Transfer Protocol) is the name of the scheme being used.
- "www.opengroup.org" is the authority component, identifying The Open Group web server.
- "cgi-bin/dbcgi" is the path component that identifies a Common Gateway Interface (CGI) script on that server.
- "TPL=pub reg" is information to be interpreted by that script (the information in this case being that the user is requesting a username and password).

URIs are generally used in the form of *URI References*. A URI reference may consist of a URI as described above (an *absolute URI reference*) or may be a *relative URI reference* in which part of the URI is omitted, its value being assumed relative to a *base URI*. For example, cgi-bin/dbcgi?TPL=pub_reg is a relative URI reference relative to the base URI https://www.opengroup.org/. When a URI reference is used to perform a retrieval action on the identified resource, it may also contain a fragment identifier, appended to the URI and separated from it by a crosshatch ("#") symbol, that consists of additional reference information to be interpreted by the user agent after the retrieval action has been successfully completed. The fragment identifier is often used to identify a particular point on a web page that the browser should display to the user when the page is retrieved.

US DoD UIDs

The UID program of the US Department of Defense (DoD) is intended to enable easy access to information about DoD possessions that will make acquisition, repair, and deployment of items faster and more efficient (see www.acq.osd.mil/dpap/UID). The DoD requires unique

identification for many items, including all those costing $5,000 or more. The identifiers are marked on the items in machine-readable form.

UID data will be stored in a central registry, maintained by the Defense Logistics Information Service (DLIS), and populated as new items are acquired, or as legacy items are assigned UIDs.

A *Unique Item Identifier (UII)* has one of two forms:

- Enterprise Identifier plus unique serial number (construct 1)
- Enterprise Identifier plus original part, lot, or batch number plus unique serial number (construct 2)

The enterprise identifier relates to the entity responsible for assigning the UII to the item, and identifies an entity location that has its own unique, separate, and distinct operation. It consists of a code identifying a recognized issuing agency plus a code uniquely assigned to the enterprise by that agency. Examples are: Dun & Bradstreet Data Universal Numbering System (DUNS) number; Uniform Code Council International Company Prefix (UCC/EAN); Allied Committee 135 Commercial and Government Entity (CAGE) number; Department of Defense Activity Address Code (DoDAAC); and Coded Representation of the North American Telecommunications Industry Manufacturers, Suppliers and Related Service Companies (ANSI T1 220) number.

The serial number is a combination of numbers and letters assigned by the enterprise which must be unique within the context of the enterprise (for construct 1) or the context of the original part, lot, or batch (construct 2).

For construct 2, the item is identified in the context of an original part, or in the context of a lot or batch. In the case of a part, the original part number is a combination of numbers and letters assigned by the enterprise to a class of items with the same form, fit, function, and interface. A lot or batch number is a number assigned by the enterprise to identify a group of items manufactured under identical conditions.

Other Important Identifier Forms

The following identity representations, while not in common use at this time, are important because of their underlying concepts or the possibility that they will become commonly-used in future.

SPKI/SDSI Names

The IETF Simple Public Key Infrastructure (SPKI) working group was active between 1996 and 1999. It incorporated the work of the Simple Distributed Security Infrastructure (SDSI) MIT Cryptography and Information Security Group Research Project. Its aim was to develop a public key certificate format and associated protocols that are simple to understand, implement, and use.

SPKI/SDSI defines a *fully qualified name* as a local name together with a global identifier that identifies the namespace in which that local name is defined. The *global identifier* is a globally unique byte string, which can be a public key, a collision-free hash of a public key, or a fully qualified name.

This definition permits a hierarchy of naming authorities, each defining its own local name format, and possibly having other naming authorities at the next level beneath it. Uniqueness of local identifiers is the responsibility of the naming authorities. Uniqueness of top-level global identifiers depends on the uniqueness of the public/private key pairs assigned to the naming authorities, and hence ultimately on the quality of the key-generation process. Most key-

generation processes in use today are of sufficient quality that the probability of duplication is negligible.

Extensible Resource Identifiers (XRIs)

XRIs are abstract structured identifiers based on URIs and IRIs, with a new syntax and resolution protocol. Standards for XRIs are being developed by the Extensible Resource Identifier (XRI) Technical Committee of the Organization for the Advancement of Structured Information Standards (OASIS); see www.oasis-open.org/committees/xri. The XRI 2.0 suite of specifications provides the following features:

- Ability to define both human-friendly and machine-friendly identifiers
- Ability to syntactically define and distinguish both persistent and re-assignable identifiers
- Ability to create identifiers using all kinds of characters (not just English)
- Ability to convey the global context of an identifier (is its subject a person, an organization, a generic concept, or a specification?)
- Ability to create identifiers for "abstract" subjects ("Paris", "the planet Jupiter")
- Ability to share identifiers, including standard identifiers and identifiers from other organizations, within an organization's identifiers ("cross-referencing")
- Ability to include standard types of identifier metadata (identifier type, language, date, and version) within an XRI using XRI cross-references
- Ability to delegate authority for issuing identifiers at all levels of the path; i.e., not just to organizations, but inside of them and between them
- Ability to assign and resolve XRI "synonyms", particularly persistent synonyms to reduce the need for pairwise mappings and reduce the number of identifiers and addresses needed to reach a given object or person
- Ability to describe and extend in XML the metadata available about a resource via the XRI resolution protocol
- Ability to control access to communication with or information about the subjects of identifiers
- Extensibility of both the XRI namespaces and the XRI resolution protocol to cope with unknown future developments

XRI syntax is an extension of IRI syntax, which in turn is based on URI syntax with the addition of Unicode characters for internationalization. An XRI identifier takes a similar form to a URI:

xri:// authority **/** path **?** query **#** fragment

The *authority* component identifies the naming authority. XRI syntax supports the same types of *IRI authorities* as generic URI syntax (DNS names and IP addresses). In addition, it supports *XRI authorities* of two types: global context symbols (single-character prefixes that establish the global authority type) and cross-references (encapsulated identifiers, such as URIs, that can identify any independent identifier community).

The *path* component is similar to a URI path component, except that it is not completely opaque. It is composed of segments that are syntactically distinguished as being either *re-assignable* or *persistent*. Re-assignable segments are used for identifiers that may be re-assigned by an identifier authority to represent a different resource at some future date (like a domain name).

Persistent segments are never re-assigned; i.e., they are effectively URNs (Uniform Resource Names).

The *query* and *fragment* components are both identical to the URI/IRI query and fragment components, except that they allow the full XRI character range.

The XRI Syntax specification also defines an unambiguous transformation from XRI normal form into IRI normal form for applications that expect IRIs, and in turn the IRI specification defines an unambiguous transformation from IRIs into URIs. Thus, XRIs can be used anywhere IRIs or URIs are accepted.

While XRI provides a framework within which organizations can define common core identifiers, and enables an organization to characterize those identifiers as persistent, it does not prescribe how the organization should generate those identifiers. Once an organization has obtained an authority identifier component of any of the types supported by XRI syntax (IP address, DNS name, GCS symbol, or private cross-reference), it can generate its identifiers easily and without regard to a central authority. It can ensure that those identifiers are unique in space and time, subject to its authority identifier component being unique. By using a persistent GCS registry or a persistent privacy cross-reference as a root, it can also issue globally persistent XRIs that satisfy the requirements for URNs.

The XRI specifications include a simple, flexible resolution protocol based on HTTP(S) and XML documents. This protocol enables any type of XRI to be resolved into an XML document that describes the XRI synonyms and service endpoints associated with the target resource. The protocol supports both native and proxy resolution, including defining a standard HTTP URI format in which all XRIs can be expressed, so XRIs can be used immediately with legacy HTML and HTTP infrastructure.

Lastly, the XRI specification suite also includes standardized XRI metadata for indicating identifier type, language, date, and version. The XRI Types specification further defines standard XRI metadata for explicitly declaring the type of an identifier such that it can be understood and resolved both within and between different organizations. This “XML for identifiers” approach can significantly increase identifier interoperability.

Following are examples of various XRIs that meet the persistence requirements for Common Core Identifier. The characteristic of persistence is recognized as being a claim (syntactically represented in XRI by an exclamation point) from the issuing authority that it manages the identifiers in a fashion that guarantees persistence and does not allow re-assignment of identifiers. Because numeric identifiers can be less prone to changing than character-based identifiers, the authority segment in these examples is represented as a numeric *inode*.

- Example of OS Username using syntax indicating/claiming persistence:
  xri://!!1000!1234.a1b2/!UserID
- Example of email address using syntax indicating/claiming persistence:
  xri://!!1000!1234.a1b2/!(mailto:mailbox@domain)
- Example of UUID using syntax indicating/claiming persistence:
  xri://!!1000!1234.a1b2/!($t*uuid*6ba7b810-9dad-11d1-80b4-00c04fd430c8)
- Example of Host Identity Tag using syntax indicating/claiming persistence:
  xri://!!1000!1234.a1b2/!($t*hit*a76f4e9c083de7a23b3deac46b98f7c3)
- Example of OID using syntax indicating/claiming persistence:
  xri://!!1000!1234.a1b2/!($t*oid*1.2.3.4)

Host Identity Protocol (HIP) Identifiers

The Host Identity Protocol is being defined by a group of concerned individuals within the IETF to provide a new protocol layer between the internetworking and transport layers. HIP can provide internetworking mobility and multi-homing at a low infrastructure cost, and can protect against certain security attacks. It is a key feature of The Open Group Secure Mobile Architecture (SMA).

The proposed Host Identity namespace consists of Host Identifiers (HI). An HI is a consistent name for a system regardless of how it connects to the Internet. Each Host Identity will uniquely identify a single host. Each host will have at least one Host Identity, but it will typically have more than one.

A Host Identifier is the public key of an asymmetric key-pair. As with SPKI/SDSI global names, uniqueness depends ultimately on the quality of the key-generation process; it is highly probable statistically but not guaranteed mathematically. A Host Identifier may be given in full, but will often be represented by a 128-bit cryptographic hash – the Host Identity Tag (HIT) – and within a particular context a 32-bit Local Scope Identifier (LSI) may represent a Host Identifier.

Abbreviations

CAGE	Allied Committee 135 Commercial and Government Entity
CGI	Common Gateway Interface
DC	Domain Component
DCE	Distributed Computing Environment
DLIS	Defense Logistics Information Service
DN	Distinguished Name
DoD	US Department of Defense
DoDAAC	Department of Defense Activity Address Code
DUNS	Dun & Bradstreet Data Universal Numbering System
GSS-API	Generic Security Service Application Program Interface
GUID	Global Unique Identifiers (Microsoft)
HI	Host Identifier
HIP	Host Identity Protocol
HIPAA	Health Insurance Portability and Accountability Act
HIT	Host Identity Tag
HTTPS	Secure HyperText Transfer Protocol
ICANN	Internet Corporation for Assigned Names and Numbers
IETF	Internet Engineering Task Force
IRI	Internationalized Resource Identifier
ITU-T	International Telecommunications Union Telecommunications Standardization Section
LDAP	Lightweight Directory Access Protocol
LSI	Local Scope Identifier
NCS	Network Computing System
OASIS	Organization for the Advancement of Structured Information Standards
OSF	Open Software Foundation
PKI	Public Key Infrastructure

RDN	Relative Distinguished Name
RFC	Request for Comments
RID	Relative Identifier
RPC	Remote Procedure Call
SAML	Security Assertion Mark-up Language
SDSI	Simple Distributed Security Infrastructure
SID	Security Identifiers
SMA	Secure Mobile Architecture (The Open Group)
SOX	Sarbanes-Oxley
SPKI	Simple Public Key Infrastructure (IETF)
SSL	Secure Sockets Layer
UCC/EAN	Uniform Code Council International Company Prefix
UID	Unique Identification (US DoD)
UII	Unique Item Identifier
UPC	Universal Product Code
UPN	Universal Principal Names
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
UUID	Universal Unique Identifier
XRI	Extensible Resource Identifier

References

[BNDLESS] The Boundaryless Organization: Breaking the Chains of Organization Structure (Revised and Updated), Ron Ashkenas, Dave Ulrich, Todd Jick, Steve Kerr, published by Jossey-Bass, 2002

[DRUCKER] Managing in the Next Society, Peter F. Drucker [ISBN: 0-312-28977-4], published by St Martin's Press, 2002

[IDMSCEN] Business Scenario: Identity Management, July 2002 [K023], published by The Open Group; refer to: www.opengroup.org/bookstore/catalog/k023.htm

[IDMWP] White Paper: Identity Management, March 2004 [W041], published by The Open Group; refer to: www.opengroup.org/bookstore/catalog/w041.htm

[IETF RFC 3305] Report from the Joint W3C/IETF URI Planning Interest Group: Uniform Resource Identifiers (URIs), URLs, and Uniform Resource Names (URNs): Clarifications and Recommendations, August 2002

[IETF RFC 3986] Uniform Resource Identifier (URI): Generic Syntax, January 2005

[IETF RFC 3987] Internationalized Resource Identifiers (IRIs), January 2005

[IETF RFC 822] Standard for the Format of ARPA Internet Text Message, August 1982

[ISO/IEC 11578] ISO/IEC 11578:1996, Information Technology – Open Systems Interconnection – Remote Procedure Call (RPC)

[TOGAF] The Open Group Architecture Framework (TOGAF); refer to: www.opengroup.org/public/arch

[XRI] The OASIS Extensible Resource Identifier (XRI) Technical Committee; refer to: www.oasis-open.org/committees