



## Authenticated Networked Guided Environment for Learning

### What is middleware?

Internet 2 describes middleware as 'glue', and this is a fairly accurate description. Middleware is a layer of software that sits between two applications and helps provide services such as authentication and authorisation, and services for combining metadata through cross-searching, harvesting and alerting. This enables applications that would not normally be able to communicate to do so without having to extend either application. This is important for disparate sectors such as HE where communication between applications may not have been considered when purchase decisions were made.

### What does it offer?

Some of the middleware structures and services that ANGEL is concerned with are:

**Authentication** - the authentication process makes sure that a user is who he claims to be by asking for proof. This is typically achieved by asking for a password but could be established through the use of smartcards, or digital certificates or by asking the user to answer a series of questions. Current developments for authentication are looking for solutions to the single sign-on, where a user only has to complete the authentication process once per session (e.g. by logging on to the local network).

**Authorisation** - authorisation is perhaps the least mature middleware service, and therefore still very much in development. At a basic level, authorisation establishes what an authenticated

identifier (an accepted username / certificate etc.) is permitted to do / access. Complexity arises from the problem associated with passing this information between services in an understandable format, and with issues surrounding storing the information such as privacy and security.

**Directory services** - directories hold the key information needed by middleware services. In an HE environment this will typically be the Student Record System for user information and the Library Management System for resource data, but information can also be gathered usefully from other directories such as portals and service directories. Information held in directories can differ greatly from institution to institution, but may typically hold personal details, photos, e-mail aliases, and permissions. A directory is formed of a schema or data model (which defines the relationship of the fields), a namespace (which defines the names of the fields) and attributes (the content of the fields). Directories also can have indices to increase information retrieval speed.

**Identifiers** - we all use identifiers in a variety of ways on a daily basis, and a typical user can have a range of identifiers. A user may have a network ID, which they use as a username to log-on (e.g. smith, ja), but which is different from the assigned e-mail alias (e.g. j.a. smith@uni.ac.uk). A key issue for middleware is the context in which an identifier is used. This might include information about how long an identifier is used, who is responsible for assigning identifiers, whether they can be reused and dependency on other identifiers.



## Authenticated Networked Guided Environment for Learning

### Why use middleware?

Many of the services described here are offered by diverse applications in daily use in UK HE. They are offered by Library Management systems, VLEs, Library Portals, Content Management systems, intranet software and database solutions. They may also be offered by solutions used by members of institutions, but not provided by the institution such as web search engines.

The problem with these services is that the products that provide them are generally proprietary, or competing, and may not make use of existing recognised standards. The services may not work with all content providers, and it can often be expensive to plug-in different services. For example, a standard purchase of a Library Management System may not include a Z39.50 module. The library would have to pay extra to have this highly essential 'add-on'. Even one specific type of service may have products offered by different vendors that do not interoperate. Even when there is a recognised interoperability standard (such as IMS for VLEs) vendors may implement this standard in different ways, to different levels.

Offering separate middleware, especially through open-source channels, creates an opportunity to utilise and promote standards and interoperability. It also offers freedom of choice to educational institutions, by allowing them to use these tools with a variety of interfaces and content providers.

### Understanding the lingo.

Middleware applications have a variety of different names attached to them, depending on the functions they perform, and the way they perform them.

Most of the ANGEL applications are **sockets**. ANGEL uses sockets to communicate between a client (front-end) program and a server (back-end) program. Requests or **calls** are sent to the socket across the internet in a format it understands. The socket then translates the request into a form that will be understood by another protocol. This makes it possible to translate between protocols without much difficulty and without having to write specific software for each pairing that might be desired.

A **resolver** essentially takes an easy to remember or unlikely to change piece of information and matches it to something which can be used to track down a resource. ANGEL makes use of **OpenURL resolvers**. An OpenURL carries with it resource **metadata**. When this is passed to an OpenURL resolver, the resolver can point the user to a variety of resource locations.

The terms **broker** and **aggregator** are generally applied with their traditional meanings to services that act as agents, or intermediaries, for other services (the ANGEL RM is a broker in this sense). The term **broker** is often used to refer to services that provide an interface to multiple network services (such as Z39.50 cross searching services). The term **aggregator** is often used to refer to services that gather content, and store and / or distribute it to users (such as OAI harvesting and RSS alerts).

**Metadata** is very important to middleware services. At a basic level, it simply means 'data about data' but it is generally used to refer to highly structured schemas and namespaces for providing information about a resource. A frequently used format for passing metadata is **XML** (eXtensible Markup Language), which is a popular format for passing metadata because it allows customisation of the data description beyond what can be supported by a language such as HTML. It uses tags, in the same way that HTML tags are used by web developers, but allows creation of tailored sets of tags for use in specific situations.