

The DNER Technical Architecture: scoping the information environment

Andy Powell
Liz Lyon
UKOLN, University of Bath
18 May 2001

1. Introduction

1.1. The Distributed National Electronic Resource (DNER) [1] is a managed information environment for accessing quality assured Internet resources from many sources. These resources include scholarly journals, monographs, textbooks, learning objects, abstracts, manuscripts, maps, music scores, still images, geospatial images and other kinds of vector and numeric data, as well as moving picture and sound collections.

1.2. This study describes the technical architecture of the DNER. The intention is to underpin the development the DNER as a managed collection of resources in an integrated information environment. The DNER information environment will allow the end-user to interact with the DNER as a coherent whole, rather than as a set of individual collections and services. We describe the network services (and the standards and protocols that enable those services to work together) that must be put in place in order to achieve that aim.

1.3. This study describes a generic architecture: it does not say how or by whom the various services described here might actually be implemented. A further stage of this work will develop a DNER Service Delivery Architecture that will drill down to actual DNER service components and the relationships between them.

1.4. This study forms part of a range of materials concerning the architecture of the DNER, available through the DNER Architecture Web site [2].

2. DNER Scope

2.1. This study makes no attempt to provide a definitive view of what the DNER encompasses. However, it is useful to briefly consider the scope of the DNER, both in terms of the range of content to which the information environment supports access and in terms of the functionality that it supports.

2.2. DNER content is typically made available in the form of collections, where we define collection to be any aggregation of one or more items. There will be many kinds of collections, including collections of textual resources, images, data, learning materials, etc. There will be collections of metadata about other collections. There will be collections held locally within institutions, 'JISC' collections made available through DNER content providers and other external collections made available elsewhere.

2.3. The DNER information environment can be characterised in terms of three high-level user activities. The DNER supports the 'discovery' of resources of interest to the end-user. It enables 'access' to those resources. Finally, it facilitates their 'use'. The architecture presented here is primarily concerned with discovery and access. It is not so concerned with the use of resources except insofar as it will be desirable that tools that facilitate use will be closely bound up with tools that facilitate discovery and access and that use will often result in further phases of discovery, access and use.

2.4. It is worth noting that end-users are not just interested in digital resources. Many of the resources of interest to people are currently physical, and will remain so in perpetuity. The most obvious examples are books and journals. There will however be digital collections that provide metadata about such physical collections. At the point of 'access' and 'use', the DNER information environment is hybrid, in that the end-user may wish to use both physical and digital resources. Physical services make physical collections available at physical locations. Network services make digital collections available at digital locations. People access content through these services.

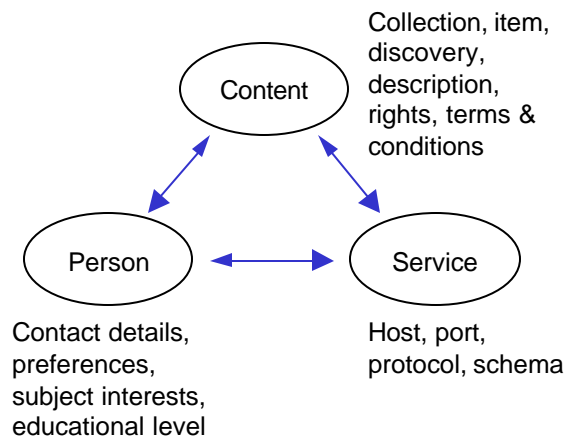


Figure 1 – High-level DNER entities

2.5. The DNER information environment is concerned with addressing two problems. Firstly, the problem of allowing the end-user to interact with multiple 'discovery' services in a seamless way – i.e. in a way that removes the need for them to learn lots of different user-interfaces in order to achieve their aim. We refer to this as the 'portal' problem. Secondly, the problem of guiding the end-user to the most appropriate copy of the resource that they have discovered. We refer to this as the 'appropriate copy' problem. At this stage we do not need to be too concerned about what defines an 'appropriate copy'. Suffice to say that it may be defined in several ways including those based on end-user access rights and restrictions, cost, institutional preferences, fastest delivery and network topology considerations.

2.6. Note that these two problems are fairly tightly coupled in practice because DNER portals, services that interact with other DNER services on behalf of end-users, will be concerned with facilitating both discovery and access. However, simplifications aside, by bringing together information (metadata) about people, content and services, the DNER information environment provides the basis for services that can begin to overcome these problems.

2.7. In summary, we can define the DNER information environment as a set of network services that enable people to effectively exploit a wide variety of resources for research and learning through their discovery, access and use, where such resources include:

- content and services,
- local and remote collections and services,
- primary and secondary information,
- data and metadata,
- digital and physical collections and services,
- JISC funded and non-JISC funded collections and services,
- policy controlled and non-policy controlled collections and services.

3. Functional model

3.1. In order to underpin the development of a technical architecture we need to begin by thinking about people's intended use of the DNER. What are people trying to achieve when they use it? This section presents an overview of a functional model for the DNER [3]. This

model builds on the functional model developed as part of the MODELS Information Architecture [4][5].

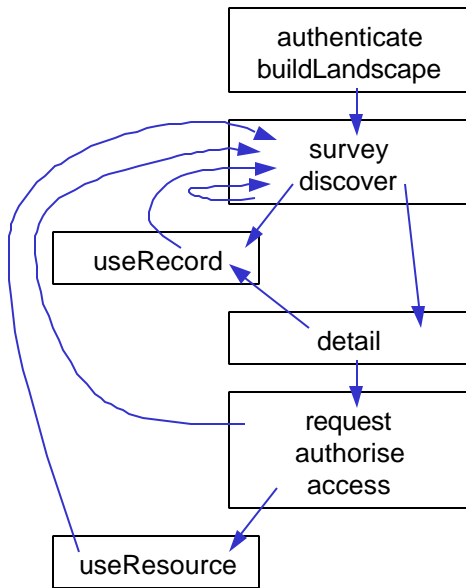


Figure 2 – High-level DNER process

3.2. As indicated above, in broad terms the DNER information environment supports discovery, access and use of a wide variety of resources. A slightly more detailed view of these activities is shown here.

3.3. What we see is an iterative process that moves from initial interaction with DNER services (authenticate and buildLandscape), through discovery of resources (survey, discover), through accessing those resources (detail, request, authorise, access), to their use (useRecord, useResource). It is worth noting that the discovery phase results in the end-user having some metadata about a resource in which they are potentially interested. That metadata record may be an end in its own right. For example, if the end-user is a lecturer, they may add that record to a course reading list and share it with their students. The access phase results in the resource itself.

3.4. The DNER architecture does not need to concern itself too much with what ‘use’ of the resource (or the record) actually means. Our minimal analysis shows ‘use’ to include:

- unpacking the resource (if necessary, for example in the case of an IMS package),
- viewing it (for example displaying the resource within a Web browser window) or listening to it,
- processing it (for example loading the resource into a spreadsheet),
- incorporating or assembling it into other (new) resources,
- storing, sharing or publishing it for use by others.

It is worth noting that the resources discovered and accessed using the DNER information environment will include information resources, learning objects and other kinds of resources. Learning objects will typically be accessed in a packaged form and will need to be unpacked, and possibly installed within a Learning Management System, prior to use.

3.5. We show a three-stage discovery process. The initial stage (buildLandscape) consists of the view of the DNER presented by the service with which the end-user is interacting – the user’s information landscape. This view may be built up based on knowledge of the end-user, their subject interests, institutional profile and other preferences, and knowledge of the collections and services that are available. The second stage (survey) is concerned with modifying the landscape by identifying the collections that are of interest for the particular activity in hand. This stage may narrow down the landscape presented by the first stage. However, it may also widen up the landscape through the discovery of new collections. This stage may result in permanent modifications to the user’s landscape, reflecting the end-user’s evolving practice and interests. The third stage (discover) is concerned with item level discovery, finding the particular resources of interest to the end-user. The ‘buildLandscape’ and ‘survey’ stages are collection level activities – they are concerned with collections of resources. The ‘discover’ stage is an item level activity.

3.6. Between the discovery of a resource and accessing it, the end-user must build up enough information to know how to request a particular instance of a resource. This often

requires the resolution of a resource identifier using one or more resolution services. For example, it may be necessary to resolve the ISBN for a book against a local library catalogue and one or more on-line or high street bookshop catalogues. Alternatively, it may be necessary to resolve the Digital Object Identifier (DOI) for a Web resource into a URL. This 'detail' process may also be concerned with obtaining information about the availability of different formats of the same resource (paper or electronic for example) and determining access rights and the terms and conditions associated with a particular instance of the resource.

3.7. At the end of the 'detail' process the end-user has enough information about a particular instance of a resource in order to 'request' it. For Web resources this is typically as simple as clicking on a link - effectively the item is downloaded by the end-user. In other cases, delivery of the resource is initiated by the content provider, in response to an inter-library loan (ILL) request for example. In some cases authorisation will be required before the item is delivered, typically based on the authentication credentials presented earlier or on the IP address of the end-user's browser. Where items are not available free at the point of use, payment using a credit card may be required before delivery. However, this study does not consider issues associated with e-commerce.

3.8. The information flow within the DNER information environment is iterative at all stages. Discovery, access and use seed more discovery, access and use. In some cases this is because the results of one person's discovery are passed on to others in the form of reading lists or shared bookmarks. In other cases it is because their usage results in new resources that become available for discovery by others. The DNER is not just a one-way flow of information from the provider to the user. End-users are both recipients and creators of primary content, secondary content and metadata. In the fullness of time the DNER architecture needs to support this variety of usage scenarios. However, this study focuses on the discovery and access aspects of the architecture.

4. Technical architecture

4.1. The current range of JISC funded collections are typically offered through their own Web-based services. Each service offers its own look-and-feel and end-users must learn and interact separately with a different service for each collection in which they have an interest. The situation is made worse because the services may offer a mix of 'discover' and 'access' functionality. In order to move from discovery to access, end-users may be required to navigate across several services manually, transferring information about the resources of interest as they go.

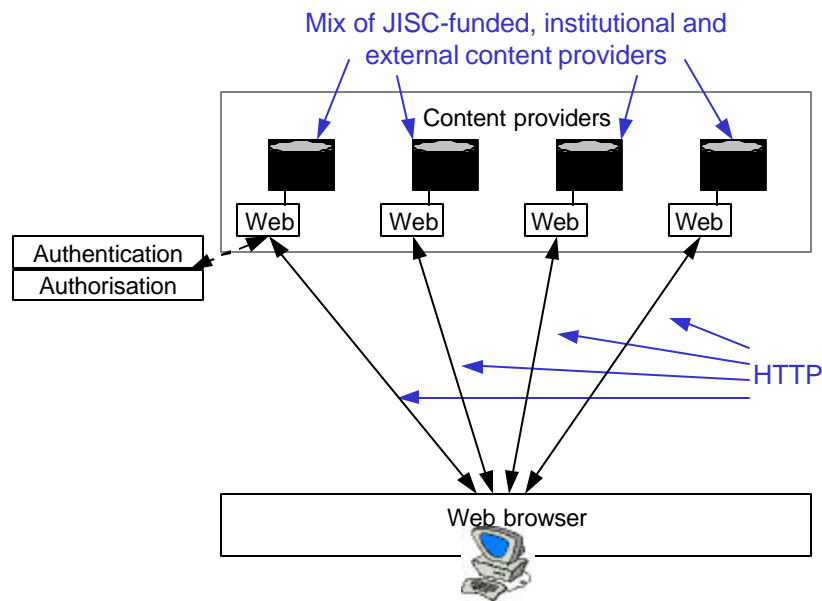


Figure 3 – Current DNER service provision model

4.2. The DNER technical architecture provides the framework that allows different services to interact with one another, allowing end-users to move between services in a more streamlined fashion than they are able to currently.

4.3. In order to develop the framework for a network of co-operating services, a set of shared infrastructure services may be required. Such services (sometimes referred to as middleware) remove the need for content provision services and portals to develop the same functionality independently. Across the current range of JISC funded services there is one such infrastructure service, the authentication and authorisation service offered by the ATHENS access management service.

4.4. The development of the architectural framework and associated shared infrastructure services facilitates the development of the DNER as a coherent whole, rather than as a set of stand-alone services. This study proposes that several new infrastructure services must be developed in order that this can happen: a collection description service, a service description service, one or more resolution services and an institutional profiling service. There will be additional shared services that are desirable including metadata registries, user preferences, thesauri and terminology, ratings and terms and conditions services. However, these are seen as being less critical for the early development of the DNER information environment.

4.5. The remainder of this section focuses on two aspects of the architecture in particular: discovery (finding resources across multiple content providers) and access (streamlining access to the appropriate copy of a resource). Detailed information about the protocols and standards described here is available in the DNER Standards and Interoperability Specification [6].

Discover

4.6. In order to allow end-users to discover resources across multiple collections offered though multiple content provision services, those services need to expose metadata for machine to machine (m2m) use. Metadata about the available resources needs to be exposed to facilitate searching, harvesting and alerting.

4.7. Searching involves sending a query from one service (the client) to another service (the server) with results being returned to the client. This must be done in such a way that both the client and the server share an understanding of what the query means and how results are to be structured.

4.8. Harvesting involves regular gathering of metadata from one service (the server) to another (the client). Again, the client and server must share an understanding of how the metadata being harvested is structured.

4.9. Alerting is conceptually different from harvesting, in that metadata is pushed from the server to the client. In practice however, alerting mechanisms are typically implemented using a regular (and frequent) harvesting process.

4.10. The DNER architecture supports the development of DNER portals – Web-based services that provide a single point of contact to a range of heterogeneous network services, local and remote.

4.11. Portals provide access to multiple network services. They will do this either by simply linking to each of the remote services (using the common Web hyperlink), by using one of the three mechanisms outlined above, or by some combination of these approaches. The use of searching, harvesting and alerting mechanisms supports greater functionality at the portal and provides a more seamless experience for the end-user because metadata from several different content providers can be presented within the consistent environment of the portal. A portal based on links to other services typically moves the end-user into the look-and-feel of each of the services that they are interested in. We refer to portals that primarily provide a set of shallow links to other services as ‘thin portals’ and portals that provide deeper linkages by making use of searching, harvesting and alerting mechanisms as ‘thick portals’.

4.12. It should be noted that we are outlining a model here. The model may be instantiated in several different ways, providing more or less functionality. For example, a simple institutional Web page containing links to external Web sites can be considered as the thinnest of portals.

4.13. It is anticipated that many kinds of portals will be developed. These will include:

- **Subject portals:** portals that provide a subject view of the DNER. For example, a social science portal will present a view of all the social science and related resources that are made available within the DNER and elsewhere.
- **Data centre portals:** portals that provide a view of all the material that is made available by a particular data centre.
- **Institutional portals:** portals hosted by institutions that provide a view of all the resources of interest to its members and to which it has negotiated access rights. Such portals may form part of the digital or hybrid library activities within an institution or may form part of its managed learning environment activities.
- **Virtual learning environments:** portals that are embedded into the virtual learning environment offered by an institution.
- **Personal portals (agents):** portals that act on behalf to a single end-user.

4.14. Each of these portal types may be thin or thick. Furthermore, we anticipate that simple mechanisms for embedding thick portal functionality within thin portals or other Web services will be developed. These may take the form of a ‘button’ link to the portal, CGI-based mechanisms such as that offered by RDN-Include [7], or other more complex alternatives. This will remove the need for all portals (particularly virtual learning environments) to support all of the standards and protocols described below.

4.15. End-users will typically access portals using their Web browser, though portals based on mobile technologies such as WAP may be developed in the future. It should be noted that DNER portals are not intended to replace the current Web-based services available from content providers. It is anticipated that end-users will value the ability to interact directly with content provider’s Web services alongside the use of the portals, depending on need, available functionality and personal preference.

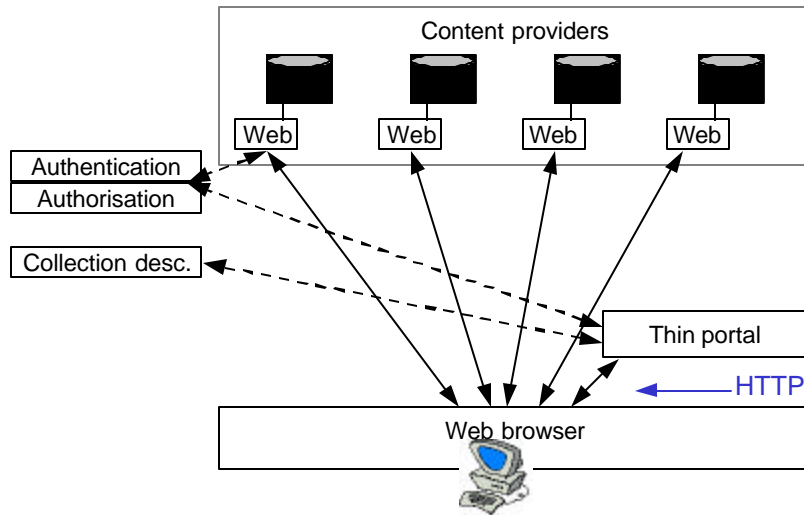


Figure 4 – Thin portal

4.16. Thin portals will present a view of available collections by gathering information from a collection description service. Such a service will provide machine-readable descriptions of available collections including information such as collection name, subject area and URL. The full range of available collections may be filtered against a list of portal subject areas and/or personal preferences. By clicking on the link associated with any one collection, the end-user will be passed directly to the Web service associated with that collection. Any further interaction with that Web service will by-pass the portal.

4.17. Thick portals will build on this simple functionality by including searching, harvesting and alerting functionality.

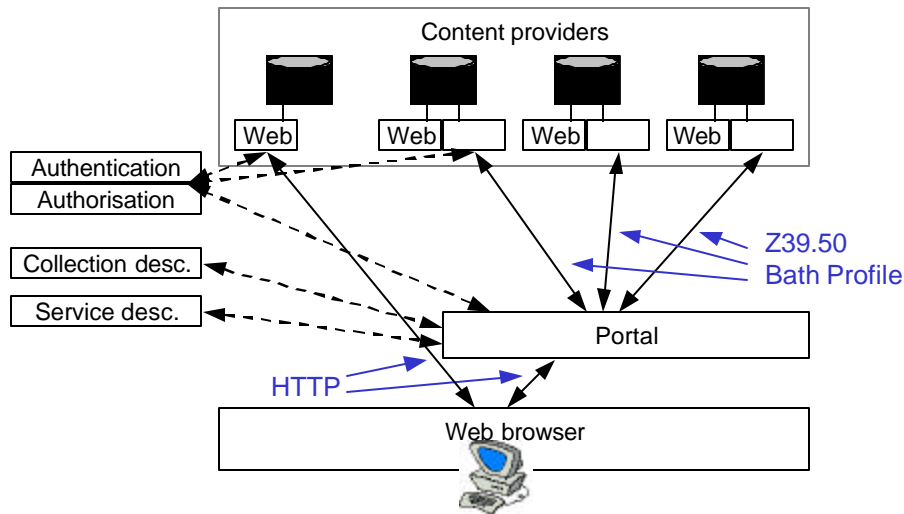


Figure 5 - Searching

4.18. Search functionality will be implemented using Z39.50 [8] and the Bath Profile [9]. This is a pragmatic choice based on currently available options. In the future other viable open standards for distributed searching will become available, for example the W3Cs XML Query Language [10]. At that time it will be appropriate to expand the DNER architecture to include those technologies. Z39.50 allows portals to send search requests to multiple DNER content provision services, getting back results formatted as Dublin Core XML records [11][12].

4.19. In order to understand the details of how to interact with each content provider's Z39.50 target, the portal will gather information from a service description service. Closely aligned

with the collection description service outlined above, the service description service will provide descriptions of the services associated with particular collections, providing protocol-specific information such as hostname, port number and database name.

4.20. The simplest cross-searching scenario is where the portal talks directly to multiple content provider Z39.50 targets. In practice broker services are also likely to be developed. These services will fan out search queries from portals to multiple targets, thus removing the need for the portal to talk to each target directly. Brokers may also offer other functionality such as query refinement or result processing. Brokers sit between the portal and the content provider target, interacting with both using Z39.50 and the Bath Profile.

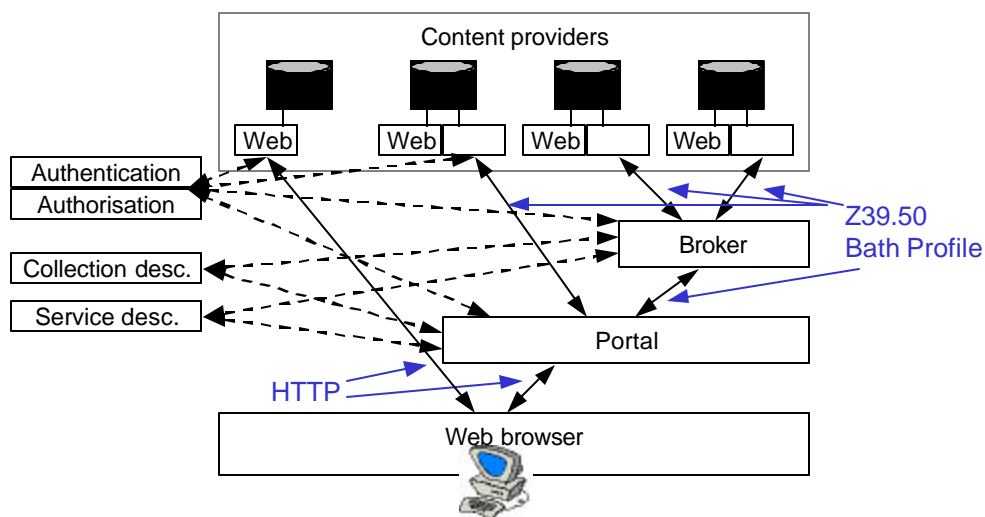


Figure 6 – Searching (with broker)

4.21. Whilst a cross-searching approach provides a fairly flexible solution there are some problems associated with it. For example, there may be performance issues associated with a cross-searching approach because response time tends to be limited by the worst performing target. Furthermore building a browse interface across multiple services based on a cross-searching approach is more difficult than with a locally held database of resource descriptions.

4.22. Portals are able to gather metadata records from remote content providers using the Open Archives Protocol for Metadata Harvesting [13]. This allows them to build local databases that contain copies of the records provided by remote content providers.

4.23. The OAI protocol provides a mechanism for sharing metadata records between services. Based on HTTP and XML, the protocol is very simple, allowing a client to ask a repository for all of its records or for a sub-set of all its records based on a date range. There is no search facility within the OAI protocol. By default, the exchanged records conform to Dublin Core. However, it is possible to define richer record syntaxes if necessary.

4.24. As in the case of searching, the portal will obtain protocol-specific details of available OAI repositories from the service description service.

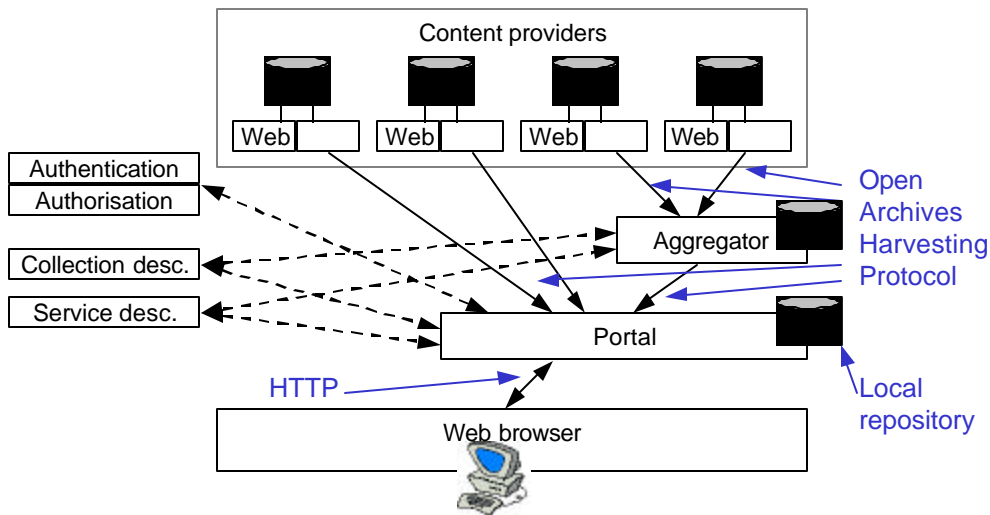


Figure 7 - Sharing (with aggregator)

4.25. It is also likely that we will see OAI aggregator services becoming available. Such services will aggregate metadata records from several repositories, making them available for harvesting by others.

4.26. Finally, portals will be alerted to new resources using RDF Site Summary (RSS) [14]. RSS is a Resource Description Framework (RDF) [15] application for syndicated news feeds on the Web. News items are described using Dublin Core-based descriptions and then exchanged as RDF/XML files. Although originally developed for 'traditional' news content, press-releases and the like, RSS can be used to exchange metadata about any frequently updated material. Again, RSS does not provide any mechanism for querying a remote service. It simply provides a mechanism for regularly gathering an RSS 'channel' from a content provider to the portal.

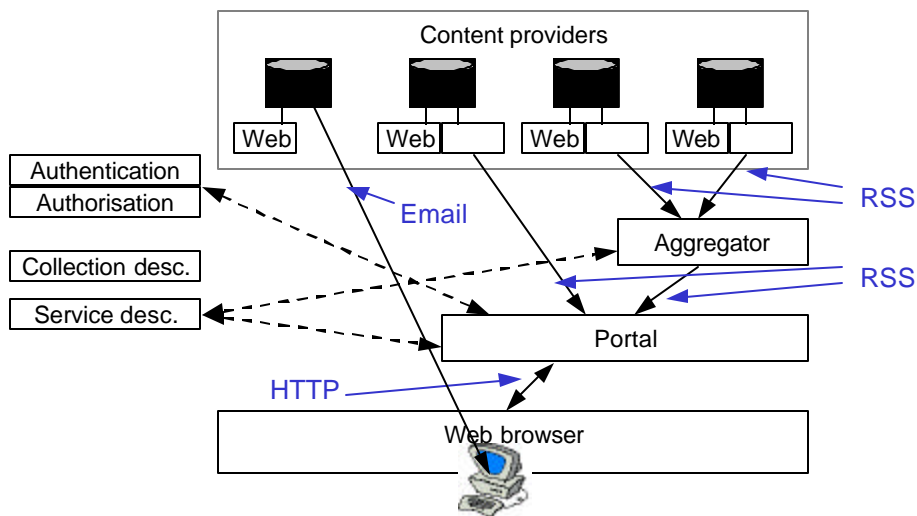


Figure 8 - Alerting (with aggregator)

Access

4.27. The discovery phase described above results in the end-user having some metadata about the resources that are likely to be of interest to them. Each metadata record will include some kind of identifier for the resource. In the case of Web resources, the identifier is most likely to be a URL. However, other identifiers may also be appropriate. For example, if the resource is a book, the identifier will be its ISBN.

4.28. It is important that the identifier obtained during the discovery process is both persistent and independent of the person who discovered it. This will allow its re-use in various ways. For example, a lecturer will be able to embed the metadata for the resource into a course reading-list or learning resource and pass it on to his or her students. Similarly, a student will be able to embed the identifier into a multimedia essay. In general terms, people will be able to provide a citation for the resource based on the identifier obtained by the discovery process. If the identifiers are not persistent, or if different identifiers are returned to different end-users, then these kinds of scenarios will not be possible.

4.29. It is also useful to consider what is identified. Consider a student searching for an image of daily life in First World War trenches. They discover an image made available by the Imperial War Museum. Does the identifier within the metadata record that they discover identify the digital image itself or the Web page that contains that image? In practice, several resources are likely to need to be identified including at least, the digital image, the original image (for example a photograph), a Web page containing the digital image, a machine-readable version of the metadata for the image and possibly a thumbnail version of the image. Depending on the usage envisaged by the end-user, any or all of these identifiers (and the associated resources) are likely to be of interest to them.

4.30. While there is no technical reason why identifiers based on URLs should not be persistent, we feel that some guidelines on good URL practice would be useful to content provision services within the DNER. We also recommend more investigation into the use of the Digital Object Identifier (DOI) [16] to facilitate persistent resource identification.

4.31. Irrespective of what kind of identifier is obtained during the discovery process, the identifier and/or its associated metadata may need to be resolved into either a locator (a URL) for a particular instance of the resource, or information about how to request the resource. A resolution service (or resolver) carries out this functionality. It is anticipated that a number of resolvers will be developed, some as shared infrastructure services others on an institutional basis. Resolvers find the most appropriate copy or copies of a resource. This may be based on knowledge of the network topology (finding the nearest mirrored copy of a resource), on access rights (what does this user's institution have a site license for), or on price (how much does this user want to pay), etc. To do this, the resolver must be aware of who the end-user is, where they are, what institution they are affiliated to and what they have access to. For this reason resolution may best be carried out locally to the end-user, where knowledge of the user's access rights and preferences are most likely to be available.

4.32. In choosing the most appropriate copy of a resource there is an interesting trade-off to be made between the preferences of the end-user, the preferences of the institution to which that person is affiliated and any associated access rights. This choice is made more complex given that users may have several roles associated with them, each of which may mean they are affiliated to a different organisation with different access rights. We refer to institutional preferences as an institutional profile and suggest that a shared institutional profile service might usefully be developed. An institutional profile could also contain information about any journal subscriptions (paper or electronic) held by the institution. The information stored in such a service could be made available to resolvers, providing them with guidance on the preferred appropriate copy for a given end-user.

4.33. The metadata or identifier obtained during the discovery phase effectively forms a citation for the discovered resource. The OpenURL [17] provides one way to encode citations for bibliographic resources as URLs. Although currently limited to bibliographic resources the scope of the OpenURL is expected to expand as it moves through the NISO standardisation process.

4.34. Put simply, an OpenURL consists of a BaseURL and a description. The BaseURL provides the location of an OpenURL resolution service and the description is typically either a global identifier (a DOI or ISBN for example) or a description of the resource (author, title, journal title, issue, etc.). The BaseURL is assigned dynamically by the service that generates the OpenURL (in the case of the DNER this will typically be a portal or content provision service) based on knowledge of the end-user's preferred OpenURL resolver. The OpenURL

specification doesn't prescribe how services maintain knowledge about a user's preferred resolver (typically this information is stored in a Web browser cookie). Neither does the specification say anything about how OpenURLs should be resolved. However, an institutional OpenURL resolver can reasonably be expected to resolve OpenURLs for books and journal articles using the local library catalogue and/or preferred online bookshops and full-text document supply services.

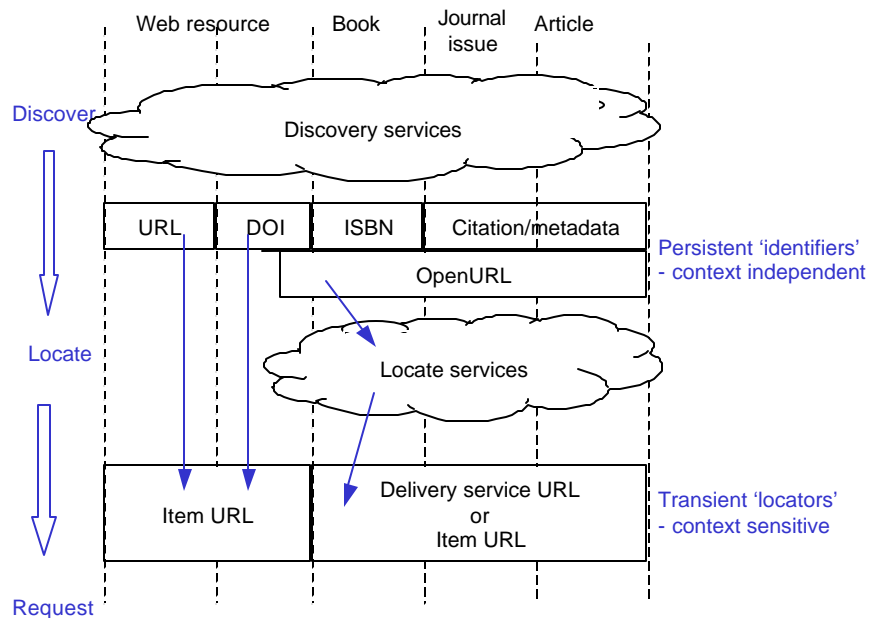


Figure 9 - Resolution and identifiers

4.35. The diagram above summarises the flow of identifiers and locators as the end-user moves from discovery, through location to accessing a resource. In the trivial case of Web resources there is typically no explicit resolving to be done – the URL obtained during the discovery process is sufficient for the purposes of accessing the resource. In the case of books, journal issues and articles, a citation is obtained (encoded as an OpenURL). This must be resolved into the URL for a particular instance of the resource or into the URL for a service that can be used to access a particular instance of the resource, as shown below.

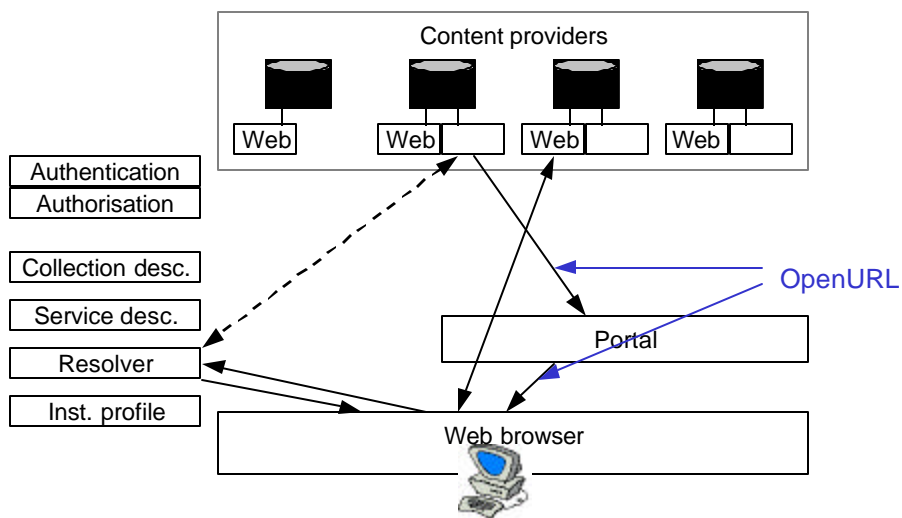


Figure 10 - OpenURL resolution

4.36. An OpenURL is returned to the end-user's Web browser, either directly from the content provision service or as part of the search results delivered by the portal. In either case, the

OpenURL contains the BaseURL for the end-user's preferred OpenURL resolver. When the end-user selects the OpenURL hyperlink a connection is made to the resolver. The resolver may optionally connect back to the service that generated the OpenURL, where it can request further information about the resource. Based on the identifier for the resource and any associated metadata, the resolver returns a list of 'deliver' (and related) services for that resource. This may be based on user-preferences and institutional profiles (which may include knowledge about the access rights of the end-user).

4.37. It is important to remember that the identifier obtained during the discovery process must be persistent and context independent in order that it can be shared with others and used as the basis of citations for the resource. The locator(s) returned by the resolution service can be transient and context dependent and only have to be valid for the particular end-user that is making the current resolution request.

5. Summary

5.1. This study proposes a technical architecture for the DNER based on portals, brokers, aggregators and content providers. These components need to call on a set of shared infrastructure services including authentication, authorisation, collection description, service description, resolver and institutional profiling services. The bulk of end-user's interaction with the DNER will either be directly with content provider Web services or through portals. Such traffic will be normal Web traffic, typically based on HTTP and HTML, using standard Web browsers. Portals, brokers, aggregators and content providers will interact using a mixture of Z39.50, Open Archives Metadata Harvesting Protocol and RSS with the majority of metadata transferred conforming to simple Dublin Core encoded using XML..

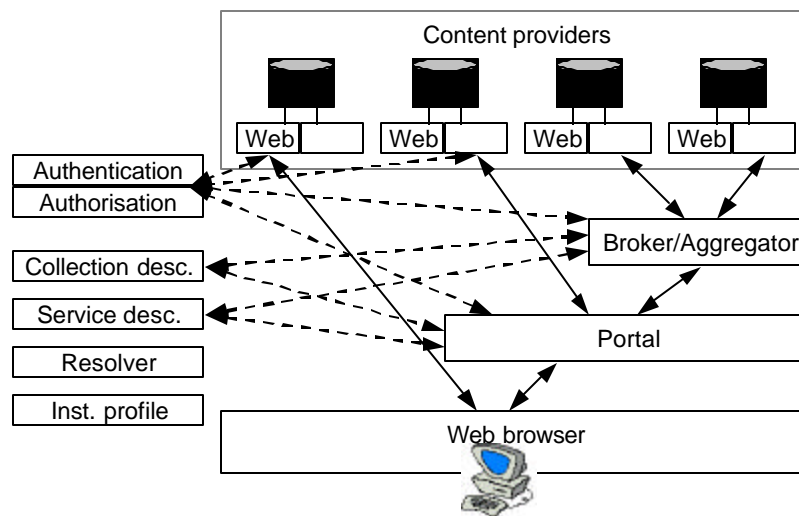


Figure 11 - Technical architecture components

5.2. We can separate DNER services into four broad classes of activity:

- **provision** - the storage and delivery of content,
- **fusion** - the bringing together of metadata about content from multiple providers,
- **presentation** - interaction with the end-user and
- **infrastructure** - shared services that support other activities.

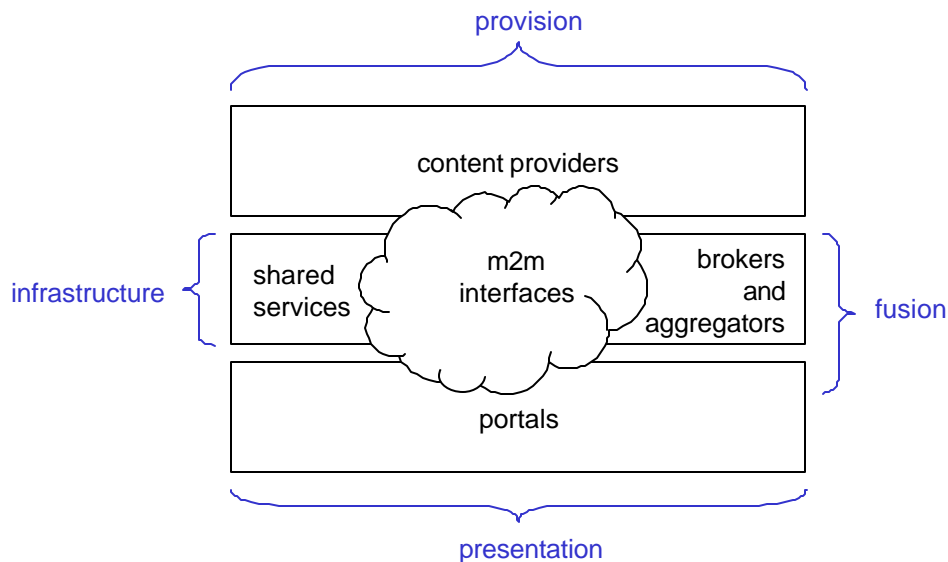


Figure 12 – DNER service categories

5.3. Typically, services will not fall cleanly into these four categories. For example, portals are likely to engage in both presentation and fusion activities, content providers will engage in provision and presentation, etc.

5.4. Separate studies will specify how collection description, service description, resolver and institutional profiling services will be operated and how they will interact with portals and other DNER services. It worth noting that portals are likely to be developed based on off-the-shelf software packages. Therefore it is likely that dynamic querying of these services will not be possible without modifications to that software being required. For this reason it may well be the case that batch-mode interaction between the infrastructure services and other DNER services may be the most sensible way forward at this stage. This will allow portal operators to update the configuration files for their portal software by obtaining information from the collection and service description services on a regular basis.

References

1. The Distributed National Electronic Resource (DNER)
<<http://www.jisc.ac.uk/dner/>>
2. DNER Architecture Web site
<<http://www.dner.ac.uk/architecture/>>
3. DNER Functional Model
<<http://www.ukoln.ac.uk/distributed-systems/dner/arch/functional/>>
4. The MODELS Information Logical Architecture
Tracy Gardner, Paul Miller, Rosemary Russell
<<http://www.ukoln.ac.uk/dlis/models/requirements/arch/>>
5. A Utopian place of criticism: brokering access to network information
Lorcan Dempsey, Rosemary Russell, and Robin Murray
Journal of Documentation, 55(1), 1999. p 33-70.
<<http://www.ukoln.ac.uk/dlis/models/publications/utopia/>>
6. Working with the DNER: Standards and Guidelines
<<http://www.jisc.ac.uk/dner/programmes/guidance/DNERStandards.html>>
7. RDN-Include
<<http://www.rdn.ac.uk/rdn-i/>>

8. Z39.50 Maintenance Agency
<<http://lcweb.loc.gov/z3950/agency/>>
9. Bath Profile
<<http://www.ukoln.ac.uk/interop-focus/bath/>>
10. XML Query
<<http://www.w3.org/XML/Query>>
11. Dublin Core Metadata Initiative
<<http://dublincore.org/>>
12. Extensible Markup Language (XML)
<<http://www.w3.org/XML/>>
13. Open Archives Initiative
<<http://www.openarchives.org/>>
14. RDF Site Summary
<<http://purl.org/rss/>>
15. Resource Description Framework
<<http://www.w3.org/RDF/>>
16. Digital Object Identifier (DOI)
<<http://www.doi.org/>>
17. OpenURL
<<http://www.sfxit.com/openurl/>>

Acknowledgements

The DNER technical architecture builds on work carried out previously during the e-Lib funded MODELS project. In particular it extends ideas first developed by Lorcan Dempsey (JISC), Robin Murray (Fretwell Downing Informatics) and Rosemary Russell (UKOLN) and more recently Tracy Gardner (UKOLN) and Paul Miller (UKOLN) as part of the MODELS Information Architecture (MIA).

This study was carried out using a consultative process that included individual and group face-to-face meetings and email discussion. The consultative group included representatives of JISC services (EDINA, MIMAS, BIDS, the Data Archive, the UK Mirror Service, the RDN, AHDS, BUFVC/MAAS, CETIS, TASI, UKOLN, UK Web Focus, UK Interoperability Focus, NISS, JISC Assist, VTAS and DISinHE), JISC committees (JCEI, JCAS, JCIEL, JCALT, JCN and JISC), other experts and interested parties (Keith Jeffery (RAL), Kevin Riley (IMS Europe), LTSN, British Library, Resource, DfEE, BECTA, NGfL, ILT and the JISC/JCIEL Technology and Standards Watch programme) and the DNER Programme Team. The authors would also like to thank all those who commented on previous versions of this document and who took part in the consultation process for the study. Full details of the consultation process are available through the DNER Architecture Web site.