Learning the Umlaut

In September 2006, Ross Singer's Umlaut service was awarded during the Second Research Software Contest, held by the Online Computer Library Center (OCLC). Umlaut is an OpenURL link resolver and works as a middleware layer which transparently integrates a range of information systems, e.g. library catalogs, link resolvers and web services. Umlaut requests these distributed resources in advance and presents all available information in one homogeneous user interface. The result is very impressive and brings together "traditional" services (e.g. links to available full texts) and extended functionalities (e.g. by providing "Closest web results")[1].

Umlaut's vision is to "improve access to library collections by contextualizing citations and available holdings more accurately for a given user", but wasn't that actually the initial reason to setup an institutional OpenURL link resolver? So let's take some curious glances at the Umlaut services to see which of them would be feasible in the SFX environment as well:

Idea 1: Check external knowledge bases to present relevant information directly to the user.

Good news - the SFX software provides a mechanism to perform this kind of pre-fetches: It's the Plug-In feature which is mainly supposed to define thresholds based on a value returned by an external program[2]. For example, most libraries only wish to offer a link to their catalog if a record for the particular book or journal is available. This mechanism can also be used to enhance the information gained from the original OpenURL request by additional data obtained from the external system. There are some convincing examples - such as the integration of location and status information from the library catalog of the Wageningen University and Research Centre into their SFX server[3].

Idea 2: Provide a shortcut URL for the service menu.

Sometimes patrons wish to bookmark a specific service menu, but unfortunately, OpenURLs appear to be longish as well as cryptic to the non-librarian. Umlaut converts each request into a very short URL in the form "http://findit.library.gatech.edu/go/XXX", displays it at the top of the service menu and thus provides a very straightforward possibility to store the link in any personalized tool. Until now, SFX does not offer any comparable feature, but there are some implementations by individual SFX customers[4] and we look forward to having this type of feature added to the SFX code.

Idea 3: Consider the user's context.

In a more and more interconnected world, OpenURL link resolvers are confronted with user requests from very different contexts, for example because the requester is located at an associated institution or just accidentally stumbled across a link to the resolver. Displaying the institutional view on available collections is not helpful in this case because users may have

---

[1] More information about Umlaut, including a short description of the workflow and access to the service itself, is available at http://www.oclc.org/research/announcements/features/umlaut-about.htm
[2] The Plug-In feature is documented in chapter 9 of the SFX User guide part 1
[3] Frank Waajen from Wageningen UR presented the implementation on the 2005 SMUG conference in London, see http://library.wur.nl/sfx_local?isbn=089054218x
[4] Neil Verkland from the MacEwan College in Edmonton Alberta posted a self-contained solution to the sfx_supp mailing list in March 2006. A very trivial alternative is to offer a link to the TinyURL service with each menu, see http://tinyurl.com/27y3o3

access to completely different resources in their current working environment. A possible approach to solve this problem would be an architecture where link resolvers know one another and exchange relevant services. According to the documentation available, Umlaut has already taken a big step towards this direction: It checks OCLC's OpenURL resolver registry[5] for a link resolver associated with the requester's IP address and - in some cases - requests additional services from this resolver. ExLibris, on the other hand, has always pointed to the fact that the requirement of bridging distributed resources is a good reason to choose the SFX software. And indeed the product provides an XML interface (SFX API) which theoretically enables the exchange of service information between link resolvers. Apart from that I'm not aware of any SFX installation which really dynamically integrates services taken from a remote link resolver into their service menu. As a simple solution to offer users a transversion point to their local institutional link resolver, we added a "Check for a local link resolver via OCLC" target which only appears if the request comes from an external IP address.

Is it worth while? While writing this article, I realized that over the last few years the notion of SFX has changed from a software which provides context sensitive services to a tool which primarily generates links to full texts available online. Anyway, the initial vision was to serve our users with a well chosen set of "right-to-the-point" resources for a particular request - and we still believe this means much more than just links to full texts. Taking a look at Umlaut and other innovative services may help us to rediscover some of our ideas we have disregarded while focusing on problems like full text licensing and providing links to appropriate copies.

---

[5] OCLC OpenURL resolver registry see http://www.oclc.org/productworks/urlresolver.htm