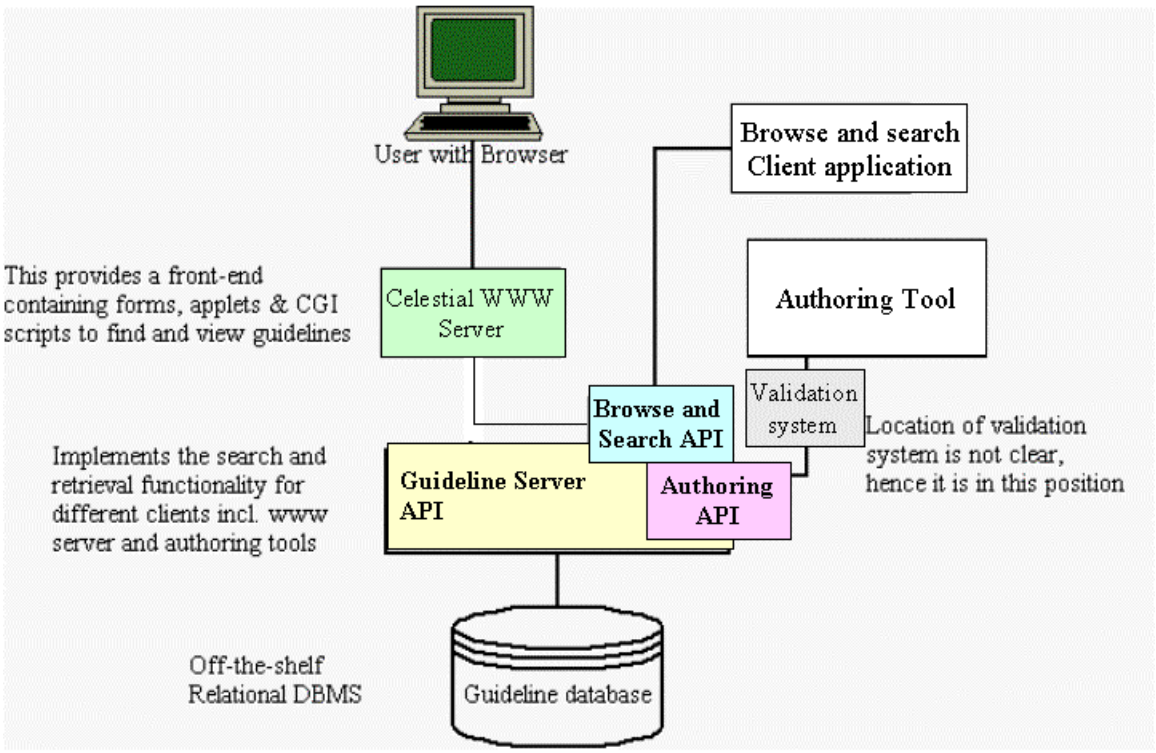
The Guideline Server: Summary of InterMed Meeting December 20th, 2000

The architecture of the guideline server is outlined below.



The diagram makes it clear that we are committed to building the API's and the celestial server, but not the client applications (except for our own authoring tools, of course).

With this diagram, one can imagine that the www server can provide the applets to render the guideline flowchart on the users screen, without putting undue burden on the guideline server.

In the architecture described in the diagram, the guideline server does not provide support for graphically viewing/browsing a guideline. That is left to client applications and to the www server. The latter can provide an applet-based flowchart viewer or can render gifs or can have little elves draw the flowchart on paper, scan it, and push it out on the network.

The "browse and search application" is not something we need to build. However, having the GL Server API available makes it possible for other application builders to connect directly to the guideline server without having to use the user-interface we impose on them through the "celestial www server".

PubMed has a similar architecture. You can use the web site at http://www.ncbi.nlm.nih.gov/PubMed/. However, when I am writing a paper, I (Aziz) access PubMed through EndNote, pull-in all the references I need into the latter application, and then cite them in my word document.

The Guideline Server has two API's sitting on top of it, one called the "Browse and search" API and the other the "Authoring" API. Then both the Celestial WWW server and the "Browse and search client application" would interact with the GL server through the "browse and search" API. External authoring tools would interact through the "authoring" API.

Issues that we agreed upon:
1) The near-term goal is to have one guideline server. In the future we may want to have multiple servers. The design of the first server does not need to take into account the future goal.
2) The server will not need to have execution capabilities.
3) Retrieval of the encodings (in RDF and in the future, maybe also XML that goes with an XML-schema) will be by keyword+ classification, as well as by eligibility criteria + patient-states that serve as entry points into the guideline. This will be done by content analysis techniques.
4) Submission will be done through the web. A submitted guideline will not be automatically available to server users. Instead, an off-server process of inspection will take place, by assistance of validation tools. After that process the guideline encoding will be tagged with one of possible validation statuses. The different validation statuses should be defined. Possibilities are: encoding fits with its declared GLIF Schema version, encoding is complete and correct by toolX, encoding is medically validated.
5) The server should not provide on-line validation and on-line authoring.

Unresolved Issues:
1) Maintaining explicit connection between the original text guideline and the GLIF encoding. We all agree that this is important. We talked about the possibility of integrating GEM-cutter into our authoring environments, although no one is really working on this. It is not an easy task. The mappings from text to GLIF are not 1:1. (For example, how do you map the recommendation "add a second anti-hypertensive drug from another drug-class", where the guideline also has tables that specify contraindications, relative contraindications, relative indications, compelling indications, and information about drugs that affect the same body system or biological process? Decision steps that make reference to domain knowledge (the info in the tables), which is expressed as concept relationships represent the information in GLIF.) In short, I think that this should be developed as a separate tool from the server. Part of the validation will be to run this tool on the encoded guideline.
2) Should browsing of the guideline flowcharts in read-only mode be part of the server functionality or not.

Issues that were not discussed yet:
1) Should the server provide comparison between guidelines? How? Any two guidelines according to the classification attributes?
2) Should we put authoring tools, browsing tools, validation tools, and execution tools on the server so people can download them?
3) Should we put the GLIF RDF schemas of different versions of GLIF (e.g., 3.1) on the server?
4) Should the classification attributes be made part of GLIF3? Should this be integrated as part of the process of integrating GEM attributes?