Editor: Geoffrey Fox, gcf@indiana.edu

DATA AND METADATA ON THE SEMANTIC GRID

By Geoffrey Fox

IN THE LAST ISSUE ("INTEGRATING COMPUTING AND INFORMATION ON GRIDS," JULY/AUGUST 2003, PP. 94–96), WE COVERED THE IMPORTANT ROLE THAT THE GRID IS EXPECTED TO HAVE IN INFORMATION SYSTEMS, FOCUSING PRIMARILY ON SENSOR AND database Grids. We contrasted this with the major original Grid focus of metacomputing, or the Grid's linking together distributed computers. Whatever type of Grid you have, metadata—data about data—is important.

One way to think of metadata is as information presented in "small, high-value records." Examples of Grid metadata include user information (name, address, profiles, and preferences) and Grid resource specifications, which could include CPU, memory, and number of nodes (if parallel) for each computer; for software, this includes location, compiler options, and, perhaps, needed input data specifications.

In previous articles, we discussed the underlying service model developed for the Web and the Grid. This groundwork lets us identify *service metadata* as a critical feature of the evolving architecture. Computers and software are services in the Grid model, but so also are databases, sensors, and network and user information systems. Here, we'll describe some of the many different sources and approaches to metadata.

Semantic Grid

In the current Web service-based approach, each service exhibits input and output channels, or ports, specified in practice by a URL, such as http://gridserviceNNN.niftyWebsite.org:80XY. This emphasizes that Web services are Web pages; accessing a Web page is the most important example of Web service data. Thus, studying Web-page metadata helps us understand Grid metadata. This observation is at the heart of the Semantic Grid (www.semanticgrid. org),[1] which builds on the W3C Semantic Web (www.w3.org/ 2001/sw/)[2] initiative.

The Semantic Web strives for an intelligent Web searching and linking structure based on including rich metadata in Web pages. An article by Tim Berners-Lee, James Hendler, and Ora Lassila uses a healthcare example to suggest how metadata-enriched Web pages could let you identify an illness, a preferred treatment, healthcare specialists, and even an ambulance service.[3] A more technical review of the Semantic Web describes it as an approach to distributed artificial intelligence (AI), with tools to generate the metadata attached to Web pages (resources) and to reason about their significance—especially when different resources can combine and individual metadata can guide what to combine and the significance of their combination.[2] Combining healthcare Web pages to generate an optimal diagnosis, doctor, and ambulance services could use a rather loose coupling of text pages that match illness descriptions between doctor, diagnosis, and patient as well as address information between patient and doctor. Combining services becomes more complex as you match precise input and output service ports corresponding to a message-based remote procedure call implementation in the Web or Grid service model.

A Web service specifies the procedure's syntax in its Web Service Definition Language (WSDL) instance. Metadata adds method semantics, or meaning, in each service. Such enriched function calls (Grid service ports) are characteristics of the Semantic Grid. Today, we have services (perhaps programmed in C++ or Java) with overlays of metadata arranged so that we can use it to deduce important consequences of linking services together. This combination of distributed AI and conventional programming represents a potentially significant new programming paradigm. We can think of the semantic, or metadata, information as a richer form of the documentation traditionally attached to programs. This semantic information is built in terms of keywords (in chemistry, for example, molecule, atom, and binding energy) and relationships (for example, molecules are composed of atoms), which experts in each and every field would need to develop. We call these domain-specific semantic frameworks *ontologies*.

 Copublished by the IEEE CS and the AIP 1521-9615/03/$17.00 © 2003 IEEE

Workflow and a Programming Model

In the Grid community, we usually call *workflow* the linking of services; rich service metadata usually is essential for any dynamic workflow in which real-time decisions determine which services tie together to solve a particular problem. As you might imagine, many contenders could provide a given service. As an example, consider yourself as a bioinformatician who needs to extract gene data from one or more databases and send it to a computer for matching. You might use metadata to choose the database source (the metadata could describe your view of the database quality), the computer resource (whose dynamic metadata includes its cost and availability), and a visualization engine (whose metadata could include functionality and suitability for client display). You then might use agents to match metadata between resources, which leads to a three-level programming model, as illustrated in Figure 1.

The lowest level is the traditional code (SQL, Fortran, C++, or Java) that implements a service. Next is agent technology that uses metadata to choose which services to use. Finally, the third level links the chosen services to solve the distributed bioinformatics problem. Each level has distinct programming models and tools to add (annotate) the metadata and some logic (AI) to reason about it. The Semantic Web community developed these tools, and the Grid is now beginning to use them. MyGrid (http://mygrid.man.ac.uk/myGrid/web/home/) and DiscoveryNet (www.discovery-on-the.net/) are two UK e-Science (www.escience-grid.org.uk/) projects exploring these ideas for bioinformatics. This discussion shows how the Grid requires, and can use, distributed AI and agent technologies and how the three-level programming model integrates conventional programming, AI, agents, and coarse-grained application integration or workflow—four often disparate and rival threads of computer-science research.

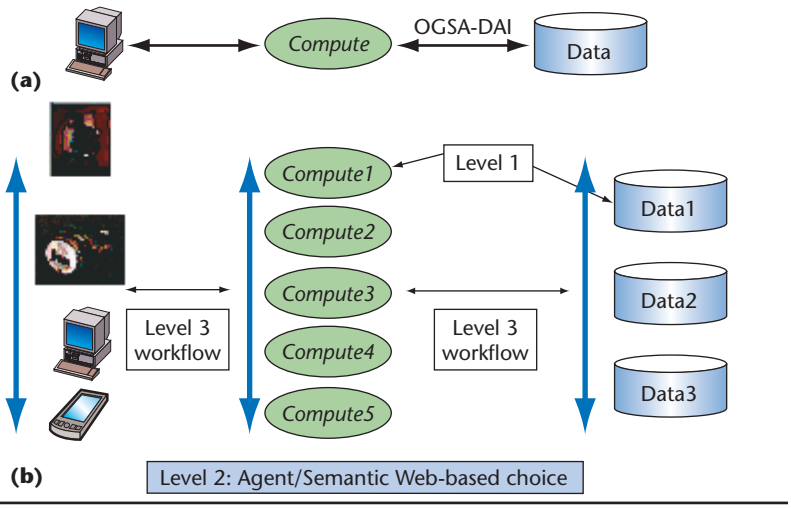

Figure 1. Workflow comparison. (a) A simple two-level workflow links database, compute, and visualization resources. (b) The three-level workflow adds an agent (broker)-based choice of resources at each workflow stage.

Usually, we consider that information or data Grids provide pathways that transform data into information and then into knowledge. Figure 1 shows the paradigm. In this regard, metadata is rather loosely defined because it includes metadata, meta-information, and metaknowledge. In the following sections, we touch on the many types of metadata, their search and retrieval, and how to gather them.

Types of Metadata

So far, we have discussed several metadata forms from different perspectives, including

- *registry*, which provides URLs (locations) in terms of unique identifiers (URIs) for each Grid or Web resource;
- *resource*, basic metadata that specifies computers, databases, and users;
- *Web services*, basic listings and simple lookups of available services relating function and location without significant rich context; and
- *Semantic Grid*, sophisticated ontologies and metadata to provide intelligent lookup and matching of services.

However, many other related small-data types share support technologies or use the service and resource metadata just listed. These include

- *monitoring*, information streams that record the Grid resources' operation, including network performance and job-status data;
- *collaboration and caching*, state changes in services that support coherence between different copies and versions of the same basic resource;
- *application fields*, specialized metadata to manage the data and information in each field (typically, these would reside in metadata catalogs maintained in a manner and format idiosyncratic to each application); and
- *provenance and data-curation*, metadata that describes data's life cycle and ownership (the topic of linking Grid and digital library communities needs more discussion).

Each of these topics deserves a deeper discussion[4] but here we will continue with our broad overview and a discussion of how we will gather and access metadata.

Gathering Metadata

How to generate metadata is a major issue today. We know how hard it is to document software and processes, so

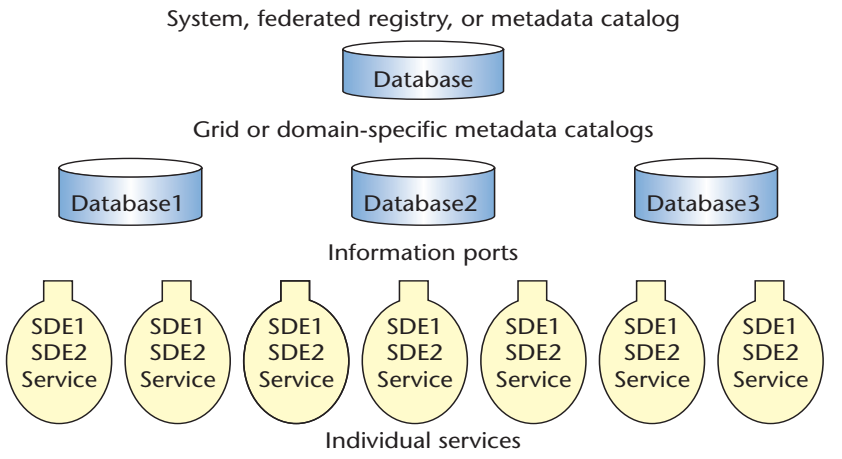

Figure 2. Three-level distributed metadata repository hierarchy. The top level covers all Grid services. The middle shows domain- or Grid-specific catalogs, each covering a services subset. The bottom shows metadata held separately in each service.

we might expect that generating metadata also would be challenging. The Semantic Web project has generated several annotation tools with convenient user interfaces to add resource-describing metadata. For example, *curation*, which is common in bioinformatics, sets up teams to manage databases, validate the added information's quality, and generate some of the needed metadata. Many scientific fields, including astronomy and environmental science, have similar processes with specific organizations responsible for metadata generation, among other things.

Data or other services connected to a particular resource automatically will generate a lot of metadata. Some computers will use existing metadata and other data to produce important new metadata. One simple example is provenance information defining the processing and ownership of data that corresponds somewhat to the preservation of computer-generated auditing metadata. More controversially, we can use search engines such as Google to automatically find metadata by comprehensively sieving through complete data sets.

Metadata Storage, Search, and Access

We could dismiss the treatment of storage, search, and retrieval of Grid metadata as a "database problem" and refer to last issue's article with the discussion of Open Grid Services Architecture-Data Access Information (OGSA-DAI) to integrate databases with the Grid. In fact, this is required if we view a database as stretching from a flat file to the latest Oracle release. However, Figure 2 emphasizes a key difficulty as some metadata becomes distributed and some is held in a system-wide database; we could realize this scenario by federating several different individual repositories. Alternatively, some metadata also will reside in the service itself, exemplified by Service Data Elements (SDE) in the new Open Grid Service Infrastructure (OGSI) Grid service model (Global Grid Forum's OGSI Working Group,[15] www.gridforum.org/ogsi-wg/) or by using embedded metadata tags in HTML Web pages. In OGSI, information ports let you query and retrieve such embedded metadata.

The middle layer of Figure 2 shows an intermediate situation in which some metadata resides in limited-scope Grid or domain-specific databases that often use diverse technologies and policies. For any given service, some metadata will reside in internal repositories and other metadata in external repositories. For example, as we add metadata to the world's Web pages, it usually is not practical to add it to an actual page. We must place it somewhere outside the resource it describes. Thus, managing metadata requires attention to distributed repositories with various degrees of scope and completeness. Furthermore, a repository at one level might be (partially) generated by injecting more distributed metadata at a lower layer, as shown in Figure 2. This complexity currently has no complete solution, but we can expect substantial progress in the near future as more sophisticated database technology deploys.

Metadata is essential for the Grid and suggests new programming models that combine workflow, conventional languages, and distributed AI; this is the Semantic Grid vision. The core database technology that generates, stores, searches, and accesses metadata must address difficult federation and distributed-system issues. Today, we are far from mature, agreed-upon solutions to these issues. There are large research activities in core technology[2] and in applications. Metadata is important; we must learn how to use it even while Semantic Grid technologies and our experiences with it are rapidly evolving. CISE

References

1. D. De Roure, N. Jennings, and N. Shadbolt, "The Semantic Grid: A Future e-Science Infrastructure," *Grid Computing: Making the Global Infrastructure a Reality*, F. Berman, G. Fox, and T. Hey, eds., John Wiley & Sons, 2003, www.grid2002.org.
2. The Semantic Web theme Issue, *IEEE Intelligent Systems*, vol. 16, no. 2, 2001, pp. 24–79; www.computer.org/intelligent/ex2001/x2toc.htm.
3. T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," *Scientific Am.*, vol. 248, no. 5, 2001, pp. 34–43.
4. G. Fox and D.Walker, Section 7.6 of *e-Science Gap Analysis*, National e-Science Centre report UKeS-2003-0, 2003; www.nesc.ac.uk/technical_papers/UKeS-2003-01/index.html.

Geoffrey Fox is director of the Community Grids Lab at Indiana University. Contact him at gcf@indiana.edu;www.communitygrids.iu.edu/IC2.html.