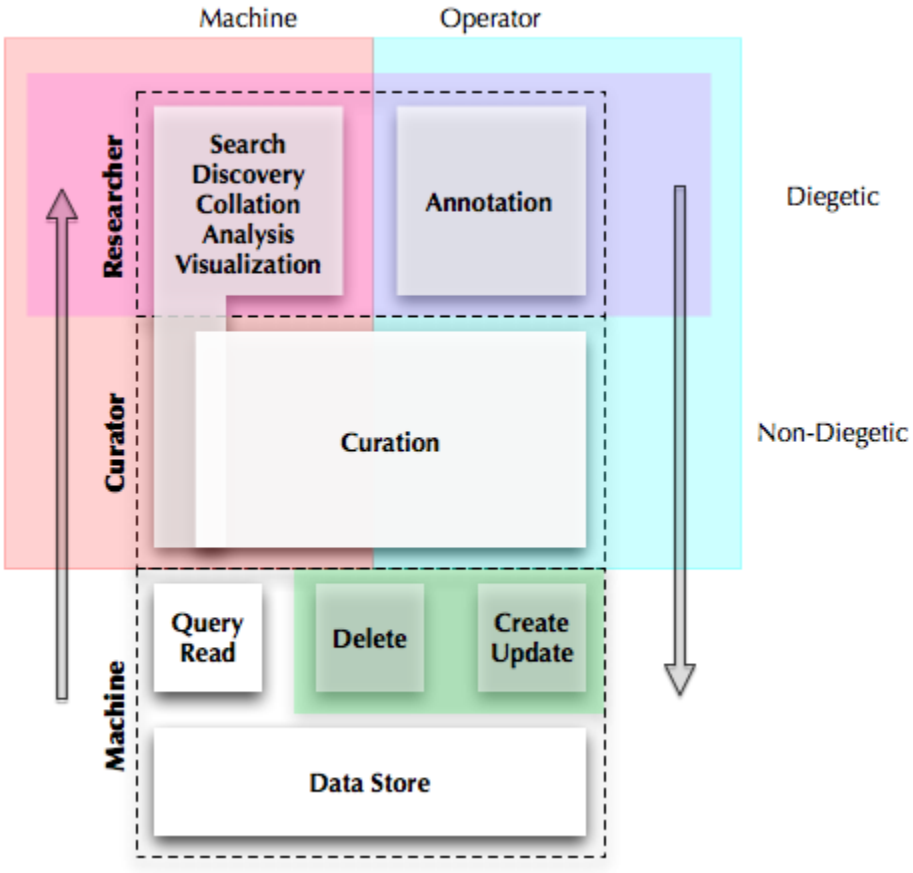
Corpora Space Workshop II Architecture Design

Corpora Space Workshop II Architecture Design

Overview



In this round of architecture considerations and planning, we want to move a bit higher up the code stack. The figure to the right gives us a new way of looking at the data flow and human interaction in Corpora Space. The architecture we tested at Corpora Camp I (the Utukku libraries, agents, etc.) provided an interface at the Machine level (Query, Create Read, Update, Delete, or Q-CRUD) that focused on Query and Read though it can be extended to include the other basic operations. Query and Read are considered data sources or data transformations while the other operations are considered data sinks. The CCI platform is designed to allow various tools and environments to access data and functionality across multiple collections.

In this document, the "Compute Engine" conflates the switchboard and compute engine components from the CCI platform. For this round, we are considering the amount of computation to be a tunable. Minimal computation is simply a mechanism for providing remote procedure calls without requiring a different protocol for each API composed of a set of remote procedures. Allowing remote pre- or post-processing requires more extensive support for remote computing beyond a simple RPC mechanism.

The stack is designed to insulate the base data store from the non-expert user by passing the data through a curation layer which can vet and normalize the data for use by non-experts. The researcher may have access to read directly from the data store, but we assume that the available data has been curated. The arrows on either side represent this data flow through the layers.

Curation is done by the curator or through a mechanized process that captures the curation rules and expertise. This could be a system by which the curator approves information contributed by the researcher or a reputation system that automatically accepts information from researchers who have demonstrated their curatorial expertise through practice.

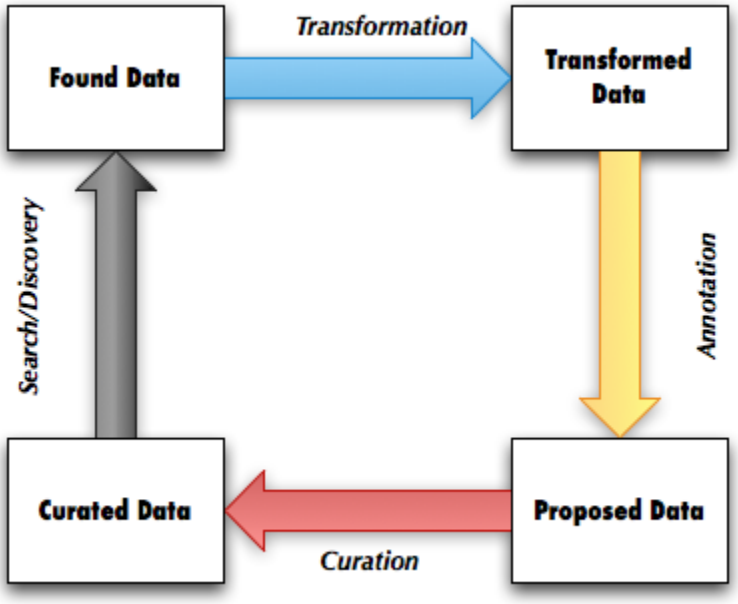
The left side of the graphic is concerned with data provided by the machine. The right side is concerned with data provided by the operator, either curator or researcher. The top layer is considered the "diegetic" layer because it is immersed in the data. The middle layer is the "non-diegetic" layer because it stands outside the data and operates on the data. It's a meta view into the data. The Delete, Create, and Update operations represent assertions while the Query and Read are interrogations.

The distinction between "diegetic" and "non-diegetic" comes from "Gamic Action, Four Moments," an essay that Alexander Galloway included in his book *Gaming: Essays on Algorithmic Culture* (U of Minnesota P: 2006). Exploring the data is similar to exploring a game world while curating the data is similar to curating the game world. For example, in World of Warcraft, the game designer might have a similar interface into the world as the player, but the designer can change the landscape while the player can not. The researcher can propose modifications while the curator can impose modifications.

The primary architecture problem facing Corpora Space is where to draw the interface between Corpora Space and the data, the tools, the curator, and the researcher. This means that Corpora Space needs to decide where to draw the boundary on this image. Everything within the boundary should be the responsibility of Corpora Space and eventually the Bamboo Project. Everything outside the boundary is the responsibility of tool and data providers. The boundary itself needs to be defined as a set of APIs or protocols. This doesn't preclude Corpora Space or Project Bamboo from creating a reference set of tools and data stores, but there needs to be a core set of APIs that tool builders and data providers can use in order to interoperate with anything else that uses the same APIs to work with Corpora Space and/or Bamboo.

Dataflow

Based on the prior diagram, we can imagine a dataflow as illustrated in the image to the right. For our purposes, we can begin by conducting a search of the curated data to produce a set of documents that we want to process with some tools. The tools might provide transformations, analysis, or other ways of creating a derivative data set. This transformed data can then be contextualized through annotation and become proposed data for inclusion in the curated data set. Once the proposed data has gone through a curation process, it can become part of the curated data that feeds the next round of searching and discovery.



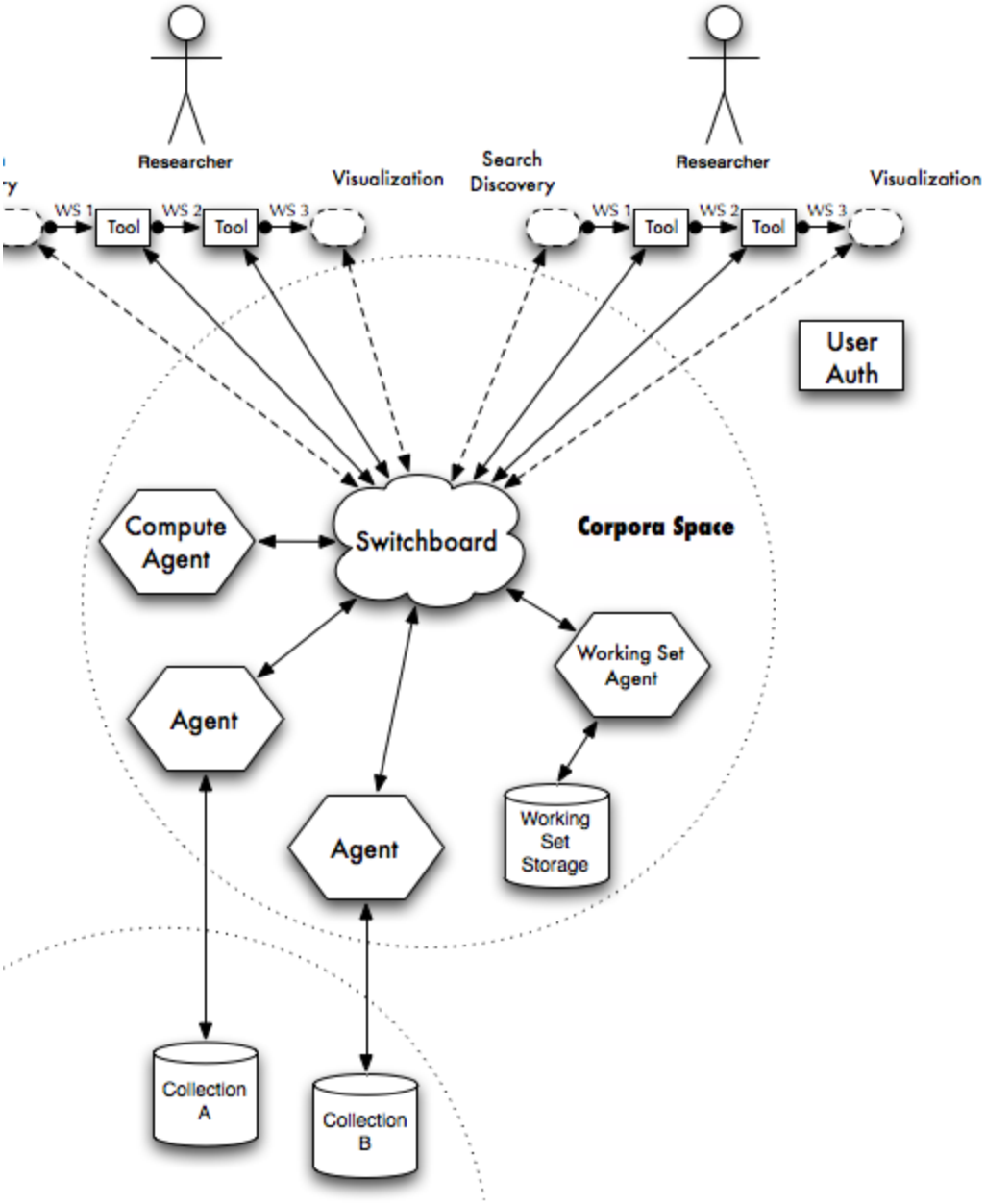
We don't expect any one person or one tool to participate in all parts of this data cycle. We also don't expect data to go through all four processes. The data transformation could be done through a series of tools working in turn on the results produced by the prior tool. Not all transformed data will be proposed as modifying the curated data. Not all proposed modifications will be accepted.

We expect that any of these four types of data could be useful. Curated data is the basis for further scholarly work. Sets of documents can form a working corpus for further specific work in an area. Transformed documents can provide a basis for visualization or further analysis. Annotated data provides scholarly information about documents and can contextualize new information for inclusion in a curated data set.

Where is Corpora Space?

The previous images provide a very broad view of how a data exploration system might be structured. This third image shows how the tools that implement the exploration and curation aspects might be connected, resulting in the dataflow from the previous section.

The details surrounding the switchboard can be found in Corpora Camp Platform Architecture and API from the first Corpora Camp. This platform provides a foundation on which we can build the connections between the switchboard, the tools, the working set manager, and the collection agents. These lines connecting the various components are primarily lines of control. They do not imply particular protocols for interacting directly with resources (e.g., HTTP might be used to retrieve a document directly from a collection, but the switchboard might mediate the search that pointed to the document).



The flow is modeled after the UNIX command line in which pipes connect components, allowing the output of one component to feed into the input of the ne

In the image, the Search/Discovery on the left in the flow is a placeholder. It could be that the tool is getting data from another tool, or that search/discovery is managed by the working environment of the user. The WS 1, WS 2, etc., lines are the exporting/importing of working sets from one tool to the next mediated by the Corpora Space switchboard. The working sets are managed by the Working Set Agent, allowing somewhat persistant storage between tools. The Visualization on the right is another placeholder. It could be that another tool is slotted in at that location. If not, the user environment might want to intelligently display the last result, or at least offer the capability.

The Corpora Space part can be done using the architecture from the first corpora camp augmented with user and group authentication/management. The Agents and Working Set Agent are just agents providing certain capabilities to the ecosystem. We may want to provide some standard libraries for various development environments to let tools easily take advantage of the input/output possibilities with working sets in Corpora Space (perhaps mirroring the standard Open/Save dialogs), at least for those tools which are standalone applications.

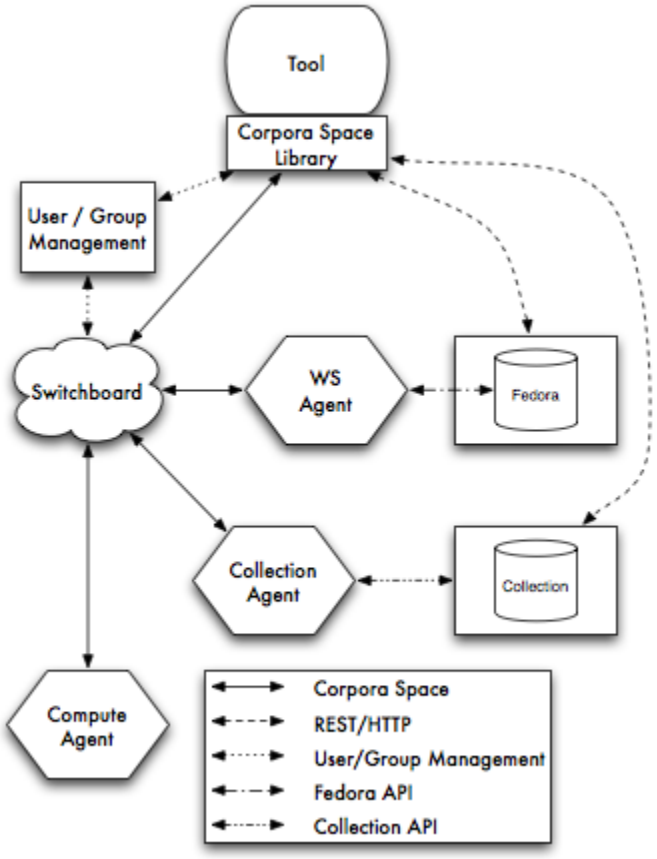
In this picture of the Bamboo ecosystem, Corpora Space becomes the repository of research results, both intermediate and final (or as final as research results are during active research) as well as a means by which to pass those results from one tool to another without having to email, ftp, copy, or otherwise manage the transport for files. There are a lot of details that would need to be worked out (e.g., how long should a working set be maintained and how many should a user be allowed to have if they aren't paying for storage?), but those are policy issues that inform aspects of the system outside the architecture itself.

How well a tool might take advantage of this system depends on what the tool does. Some tools might work well with a simple configuration and then operate on each packet of information coming down the pipeline (e.g., a document classifier). Other tools need hands-on work (e.g., a graphics editing program used to retouch photos).

The Corpora Space ecosystem can support one-off pipelines in which each tool provides an interface for users to select incoming working sets as well as outgoing working sets (analogous to opening and saving files) as well as establishing well-defined workflows across tools. The first style will probably be most useful for exploration while the latter will probably be most useful for curation. Exploration demands an agile computing environment while curation typically requires a stable, well-documented flow from raw data to the final, curated database.

Connecting Tools with Corpora Space

The figure to the right shows a typical set of interconnected components. At the top is a tool that uses the Corpora Space library to interact with the Corpora Space ecosystem. Various protocols are used based on the context, but the primary ones from the CS library perspective are the messaging protocol with the switchboard and REST/HTTP for accessing documents exposed via RESTful interfaces. User authentication is typically done at the beginning of a session with the tool, especially if the tool is a standalone tool and not embedded within an environment that provides user management for the tool.



In this figure, the tool can communicate through the switchboard with the working set agent to discover working data in Corpora Space that can be used as input into the tool. The actual data can be read via HTTP. In this case, we picture the data to be stored in a Fedora repository since Fedora provides almost all of the semantics we need: accessibility, management, metadata, resource forks, and the ability to track provenance of the data. While the working set agent might communicate directly with Fedora, we may introduce a thin server-side layer to provide easier access for the particular needs of the tools.

The tool may also interact with a collection, using the collection agent to perform a query on a collection. The returned set of document references and metadata can be used to retrieve particular documents via the collection's REST/HTTP interface. The tool also may access some remote compute service represented by the Computer Agent.

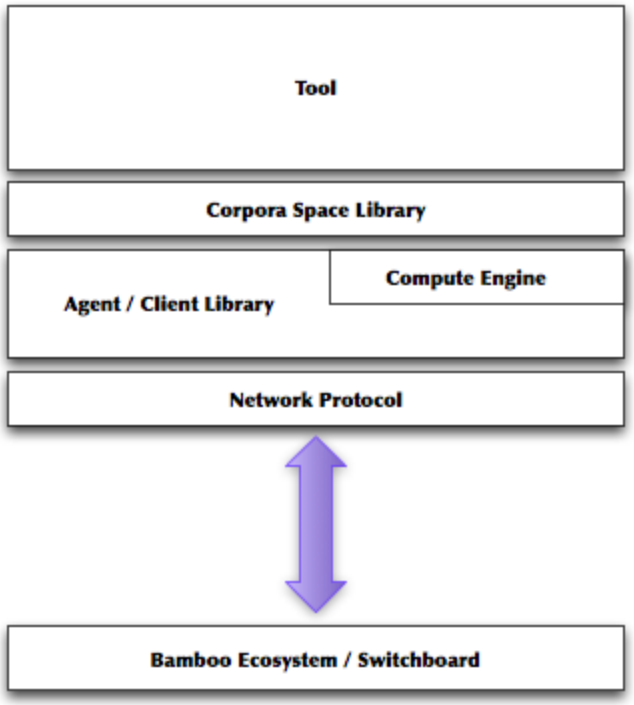
The messaging platform divides functionality up as follows:

	function	mapping	reduction	consolidation	action
Source	X	X			
Transformation	X	X	X	X	
Sink					X

Corpora Camp I explored the use of Source and Transformation (see Corpora Camp Platform Architecture and API for details). The current architecture is exploring Sinks with tool interconnection mediated through working sets. The actual protocol details are still being worked out, but a likely solution will be to use the messaging protocol to set up the appropriate data receptacle and then use REST/HTTP to place data in the receptacle.

Tool Software Stack

Tools should be able to take advantage of Corpora Space with minimal work. Additional work may be required to take advantage of more advanced Corpora Space capabilities. We expect to provide a set of libraries implementing the base protocol as well as tool-specific components in a number of languages. Our initial focus is expected to be those languages needed in order to support the largest number of tools. Likely, this will be Java, JavaScript, and one or two other languages. Since the platform will be open in all aspects (open source, open protocol specification, openly documented), anyone can implement a set of libraries for their development environment.



The basic Corpora Space tool profile supports retrieving and depositing research results from and to Corpora Space. We don't expect this to be any more complicated or difficult than using the system open and save functions if using the Corpora Space tool libraries. All of the interactions with Corpora Space will be handled behind-the-scenes by the library, including user authentication if necessary.

More advanced interactions with Corpora Space may entail tracking changes from the input to the output, essentially providing a record of the provenance of the result. We expect to provide easy-to-use APIs in the Corpora Space tool library to enable this, but tracking changes such as this does require more extensive interaction with the library in the tool's core code.

This right-side column is an artifact of "old-style" navigation. **Please remove sections, columns, and content of the right-side column when this page is next edited.** Cf. the Wiki navigation changes - July 2011 page for a 'how-to' on removing old-style page markup.