

QUASI-HIERARCHICAL, INTERACTIVE NAVIGATION OF IMAGES AND META-DATA IN THE OPEN MICROSCOPY ENVIRONMENT

Harry Hochheiser and Ilya G. Goldberg

Image Informatics and Computational Biology Unit, Laboratory of Genetics, National Institute on Aging, National Institutes of Health, 333 Cassell Drive, Baltimore MD 21224, USA
{hsh,igg}@nih.gov

ABSTRACT

Databases containing microscopy images and associated meta-data present users with the challenge of interpreting complex interrelationships between objects. Effective user tools must support users in the task of interactively navigating and filtering data to identify items of interest and understand relationships between these items. In the Open Microscopy Environment, these relationships often take the form of “quasi-hierarchies”: tree-like structures where nodes can have multiple parents. A compact browser for selecting items in a quasi-hierarchy illustrates relationships between items and supports selection of items from any level in the quasi-hierarchy. Multiple interoperating browsers for orthogonal quasi-hierarchies can be coordinated to form a powerful filtering interface useful for simultaneously drilling down from multiple perspectives. These hierarchy browsers can be combined with information-specific displays, including summary views containing detailed information about image sets and image thumbnail displays.

1. INTRODUCTION

The complex data models used in image databases present scientists with significant challenges in identifying and interpreting data. Large numbers of images are typically collected under differing experimental conditions and subjected to a range of analytic procedures. For example, images must be examined to identify similarities, draw connections between dissimilar results, and generate hypotheses for further study. Although systems that use machine learning and similar approaches to automatically classify images or identify useful patterns (e.g., [6]) have substantial potential, automated approaches are most powerful when combined with the direct involvement of expert scientists. Powerful interactive tools that support open-ended exploration and navigation of large data sets are needed to support this process.

Given the large volumes of data and the complexity of the underlying data model, these tools must present information in a manner that emphasizes connections and relations between different data items, while minimizing the cognitive load required to interpret the data.

Interfaces that provide multiple perspectives on complex data can give users alternate paths to the same result. By combining flexible selection of items of interest with feedback that clearly

indicates the current state of selection, these interfaces can help users approach data from multiple viewpoints.

This strategy can be used to navigate “quasi-hierarchical” data decompositions found in the data model of the Open Microscopy Environment (OME). The combination of multiple browsers – one for each quasi-hierarchy – provides a powerful tool for quickly drilling-down to items of interest.

2. THE OPEN MICROSCOPY ENVIRONMENT

OME is a data model and a software suite for the storage, management, and analysis of biological microscopy data. OME’s data model describes images, acquisition parameters, annotations, and analysis results. Experiments can be defined by combining images into *datasets*, which can in turn be collected in larger *projects*. An image can be included in multiple datasets, and a dataset in multiple projects.

OME supports the automated analysis of images for the extraction of scientifically useful information. Analyses are conducted by executing *analysis chains*, which are strongly typed, directed, acyclic workflows composed of individual *analysis modules*. Inputs and outputs of the various modules are instances of various *semantic types*: information structures that describe the results of specific computations. The data model and analysis framework are extensible: semantic types can be defined as needed to support the addition of new modules. The analyses performed by these chains and modules are conducted against a specific dataset or image, and each result is associated with a dataset, an image, or a feature derived from an image.

The OME software suite includes facilities for instantiating the data model in a relational database; tools for importing images, analysis modules, and other data via descriptions in XML, an engine for executing analysis chains, and both web-based and Java-based user interfaces. For more background on OME, see Goldberg, et al. [4].

3. QUASI-HIERARCHIES

The experimental and analytic components of the data model described above provide a natural starting point for interactive navigation of an OME data store. Specifically, the non-exclusive relationship between projects and datasets is neither a strict hierarchy (as objects may have multiple parents) nor a fully general tree (projects do not directly contain images). This “quasi-hierarchical” structure is also found in relationships between analysis objects. Each analysis chain contains one or more modules, and modules can be contained in multiple chains. Inputs

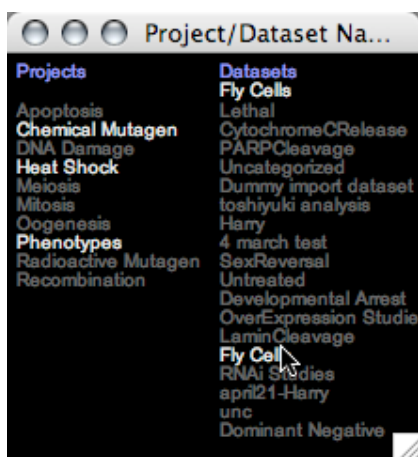


Figure 1: The OME quasi-hierarchical navigator. Projects are listed in the left column, datasets in the right. Mousing over a dataset on the right causes the highlighting of all associated projects. The name of the currently highlighted dataset is displayed at the top of the column.

and outputs of the modules are instances of semantic types, which can be produced and consumed by multiple modules.

Extensions to the data model can create additional quasi-hierarchies. For example, a data model for screening might contain many-to-many relationships between screens and plates.

These quasi-hierarchical decompositions are largely orthogonal: decomposition of data into projects and datasets is independent from data from any divisions based on screens, plates, and wells. This orthogonality provides an opportunity for coordination of multiple browsers. Given n orthogonal browsers, selection in each of them can be made independently, forming a conjunctive query. The increased flexibility of selection can help users quickly drill-down to find items of interest. The result of the query can be displayed in type-specific displays. For example, a list of images can be rendered using thumbnails, while a list of sub-image features and their associated information (e.g., positions, intensities, etc) can be displayed in a multi-dimensional interactive viewer.

4. QUASI-HIERARCHICAL BROWSING

Quasi-hierarchical structures present a challenge for interface design. Human-computer interaction research has generated a substantial body of work on navigating both strict hierarchies and fully general trees [7], but neither of these approaches is ideally suited for quasi-hierarchical navigation.

Tools that use a traditional hierarchical browser (similar to those used for exploring file systems) might do so by creating multiple representations for each item, each indicating membership in one parent. Thus, a dataset found in several projects would have a separate listing in the dataset list for each project.

This approach runs the risk of obscuring relationships that may be potentially interesting. For example, although the list of datasets associated with a given project may be easily identified, the inverse question – finding the projects associated with a dataset – is potentially very difficult to answer in a fixed hierarchy browser.

Graph browsers are similarly inappropriate for quasi-hierarchical data. These browsers generally provide a clear picture

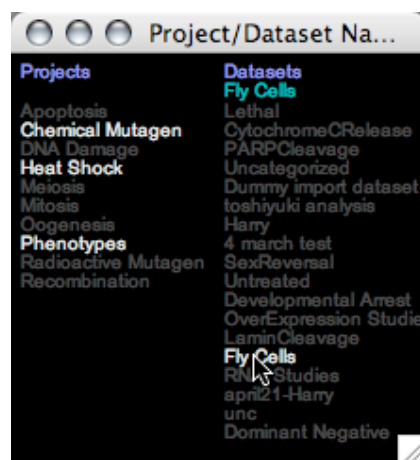


Figure 2: A dataset has been selected in the right-hand column. This highlights all of the associated projects and "locks" the browser on the selected dataset. The dataset can be unselected by clicking on it.

of the relationships between items, but this clarity often comes at the price of the extensive screen real estate needed to explicitly render links. Furthermore, general graph browsers might not take advantage of the rooted nature of the quasi-hierarchies that have been identified in OME.

An effective browser for quasi-hierarchical data structures will provide compact views of the relationships between items in a manner that clarifies (rather than obscures) the many-to-many relationships present in the data. Given the extensible nature of the OME data model, this browser must also be general enough to handle novel quasi-hierarchies based on user-defined semantic types.

The use of OME's quasi-hierarchical browser for exploring the relationships between projects and datasets is shown in Figure 1. Projects are given in the left-most column, and datasets in the right. Relationships between items are illustrated on mouse-over: when the mouse moves over a project/dataset, the corresponding datasets/projects are highlighted. The name of the currently highlighted item is also displayed at the top of the corresponding column. Items in either column can be selected via a mouse click. These selections lead to the highlighting of corresponding items in the other column, and the dimming of items that do not correspond (Figure 2). These selections serve to "lock-down" the search space to the selected axis and result in a query component that can be combined with other, similarly locked-down axes to drill-down to a specific result. Individual selected items can be deselected by a mouse click. Mouse clicks on the background clear all selections.

This browser uses a tradeoff between screen space and user time to present a compact view of the relationships between the two levels in the quasi-hierarchy. Instead of showing all of the relationships at all times – via graphically depicted links that occupy significant space and add visual clutter – interactive highlighting is used to display links associated with selected objects.

A similar quasi-hierarchical browser for OME's analysis hierarchy of chains, modules, and semantic types is given in Figure 3. This browser displays chains in the left-most column, with modules and semantic types in subsequent columns. Due to the large number of modules and semantic types, they are displayed as rectangular icons. When a selection is made in a given column, the selected item is displayed in its entirety. Mouse-over tooltips

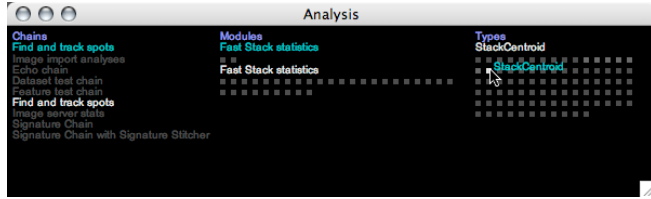


Figure 3: A quasi-hierarchical browser for OME's analysis structure. Analysis chains, modules, and semantic types are shown in the three columns. Unselected modules and types are shown as rectangular icons. A mouse-over tooltip shows the name of an unselected type.

identify the iconified items in either column. Alternatively, a shift-click on the background of the browser invokes a terse display mode, in which only those items that are currently active (i.e., the modules and types associated with a selected chain) are displayed.

The analysis browser in Figure 3 is an instance of the widget shown in Figures 1 and 2 applied to a new quasi-hierarchy: the use of the iconified representations is managed automatically by the widget layout algorithm.

5. DATASET OVERVIEW DISPLAY

The project/dataset browser provides a compact means of selecting a dataset of interest, at the expense of limited information: selections must be based only on the name of the object. A second window containing detailed information about the various choices can provide further context that may prove helpful in locating appropriate image sets.

The dataset overview display is a window that contains a rectangular region for each of the datasets that is currently active. This region contains the name of the dataset, graphical and textual indications of the number of images in the dataset, descriptions of analytic metadata, and a button that can be clicked to view the images in the dataset. Currently, the analytic data summary is limited to indicating the number of analysis chains that have been executed against the given dataset. Eventually, detailed summaries describing analytic results, including the number of results of different types, will be included in this area as well.

The dataset overview window is subdivided via a treemap decomposition [2], which allocates space to these datasets in proportion to the number of images in the dataset. The display initially shows overviews of all of the available datasets. When selections made in quasi-hierarchical browsers change the set of active datasets, the layout of the dataset display is updated to contain only the active datasets.

The dataset display is coordinated with the mouse-over highlights found in the quasi-hierarchical browser. When a dataset is highlighted in the project/dataset browser, the region for that dataset is highlighted in the dataset display. Direct interaction with the dataset display can be used to navigate through the project/dataset hierarchy: mousing over a dataset region leads to a highlight for that dataset, and clicking in the dataset region selects the dataset.

6. SIMULTANEOUS MENUS

As the multiple quasi-hierarchies are orthogonal, multiple browsers (one for each quasi-hierarchy) might be combined to support

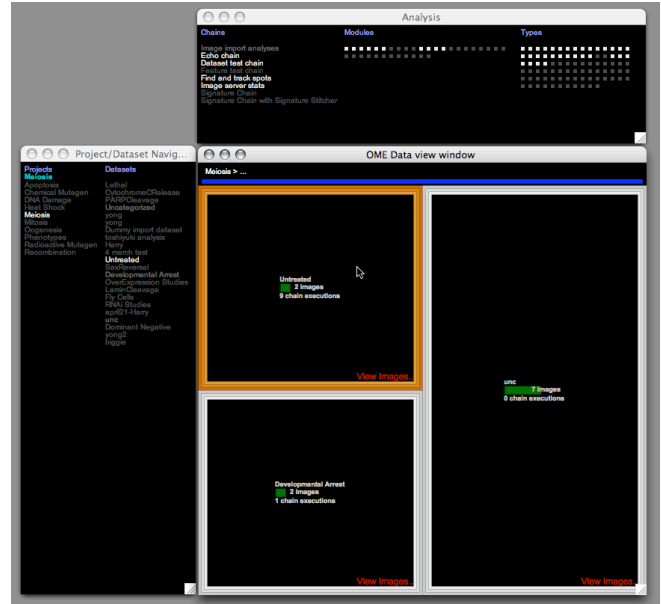


Figure 4: Coordinated browsing via simultaneous menus: Selection of the "meiosis" project leads to display of three associated datasets in the dataset overview display. Mousing over the "Untreated" dataset leads to highlighting of both that dataset and the chains that have been executed against that dataset.

browsing from multiple perspectives. Building on the examples given above, a user might begin by selecting an analysis chain, which will restrict the dataset browser to display only those datasets that the select chain has been executed against. The subsequent selection of a project will further restrict the set of active datasets to include only those that are associated with that project, in effect creating a conjunctive search. Known as "simultaneous menus" this approach has been applied to web browsing and arbitrary tree structures [3,5].

Mouse-over highlights can also be coordinated across different instances of the quasi-hierarchical browser. For example, mousing over a dataset in the project/dataset browser leads to the highlighting of the analysis chains that have been executed against that dataset (Figure 4).

7. THUMBNAIL VIEWERS

The underlying goal of these navigational tools is to help users locate images of interest. Once a dataset has been identified as being particularly interesting, the "view images" text in the appropriate dataset overview can be used to open a new window containing thumbnails of all of the images in that dataset. For large datasets, multiple level-of-detail zooming facilities can be used to focus in on thumbnails of interest. Multiple thumbnail displays – each for a different dataset – can be used to compare and contrast the various datasets.

Interaction techniques can provide further illustration of the relationships between items in these datasets. As images can be contained in multiple datasets, mousing over a thumbnail for a given image will lead to a highlight of thumbnails for that image in all datasets (Figure 4).

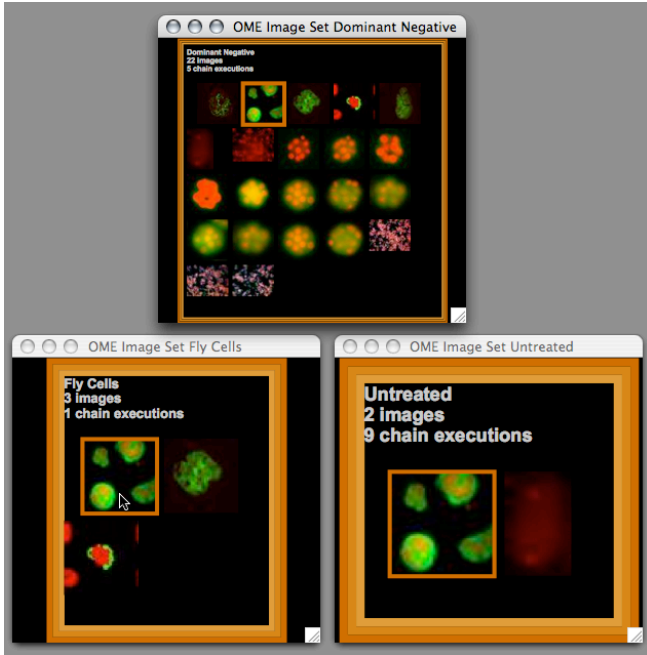


Figure 5; Image thumbnail viewers for three datasets. Mouse-over of a thumbnail in one dataset leads to highlighting of the thumbnails for that image in other datasets.

The thumbnail display viewers are also coordinated to exchange mouse-over highlight events with the dataset overview window and the quasi-hierarchical browsers.

Earlier prototypes of this browser displayed all of the thumbnails in all of the datasets in a modified version of the dataset overview window. Inspired by the PhotoMesa image browser [1], this design had the advantage of showing all of the thumbnails in a single window. However, the thumbnails overwhelmed the meta-data currently shown in the dataset overviews, and the overhead of loading large numbers of thumbnails that may not be of interest was deemed unacceptable.

8. DISCUSSION

User interfaces for complex data models define and constrain the tasks and analyses that can be conducted with the underlying data. Tools that provide multiple perspectives on data provide users with flexibility that may be useful for finding items of interest and building understanding of relationships between items.

The browser described above is a general widget for making selections from “quasi-hierarchies”; tree-like graphs with internal nodes and leaves that can have multiple parents. The combination of compact presentation of the levels of the quasi-hierarchies with mouse-over highlights and coordination between multiple browsers provides users with rich feedback regarding the relationships between items in the data model, without the risks of disorientation often found in more general tree browsers.

The combination of multiple coordinated browsers providing alternative paths through the data store can be a powerful aid for navigation. Any given path may be more or less appropriate for the user at any specific moment: a task best suited for project/dataset selection might be followed by another that requires navigation of the analysis hierarchy. In still other cases, the user might not initially be aware of the appropriate path to take. Alternative

browsers eliminate the need for an up-front decision. Coordinated highlights provide visual connections between items of interest, illuminating correspondences that may have been otherwise overlooked, presenting opportunities for serendipitous discovery of interesting data.

Although the quasi-hierarchical model and browser are specifically motivated by the presence of these structures in OME’s project/dataset/image hierarchy and analysis hierarchy, the model is, in theory, flexible enough to be extended to other quasi-hierarchies that might arise in OME and other related data models. As new semantic types that extend the OME data model are defined, relationships between instances of those types may form quasi-hierarchies. As these structures may evolve over time, full support will require run-time mechanisms for specifying the types that interact in a quasi-hierarchy (via additional annotation in the data model or explicit user interaction), and the relationships that will allow interactions with that hierarchy to be coordinated with others.

The scalability of this approach to browsing quasi-hierarchies remains an open question. While browsing tens of items in each level of a quasi-hierarchy may work well, this approach may break down as the dataset grows to contain hundreds or items (or more) at a given level. Alternative approaches, such as the addition of intermediate levels of organization, might be needed to accommodate larger data stores, but this problem is not unique to the quasi-hierarchical browser presented in this paper.

9. REFERENCES

- [1] B.B. Bederson. “PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps” UIST 2001, ACM Symposium on User Interface Software and Technology, CHI Letters, 3(2), ACM Press, New York, pp. 71-80.
- [2] B.B. Bederson, B. Shneiderman, and M. Wattenberg “Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies” *ACM Transactions on Graphics* 21(4) ACM Press, New York, pp. 833-854, October 2002.
- [3] G. Furnas, G. and J. Zacks “Multitrees: Enriching and Reusing Hierarchical Structure” *Proc. CHI '94 Conference: Human Factors in Computing systems*, ACM Press, New York, pp. 330-336, 1994.
- [4] I.G. Goldberg, et al., “The Open Microscopy Environment (OME) Data Model and XML File: Open Tools for Informatics and Quantitative Analysis in Biological Imaging.” *Genome Biology*, Biomed Central, London, pp. 6:R47, 2005.
- [5] H. Hochheiser and B. Shneiderman, Performance Benefits of Simultaneous over Sequential Menus as Task Complexity Increases. *International Journal of Human-Computer Interaction* Lawrence Erlbaum Associates, New Jersey, 12(2), pp. 173-192, 1999.
- [6] N. Orlov, J. Johnston, C. Wolkow, and I.G. Goldberg, “Pattern Recognition Approaches to Compute Image Similarities: Application to Age Related Morphological Change” *2006 IEE International Symposium on Biomedical Imaging*.
- [7] Shneiderman, B., and C. Plaisant, *Designing the User Interface, Fourth Edition*. Addison-Wesley: Boston, 2004.