

# Towards Interactive Exploration of Images, Meta-Data, and Analytic Results 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

*The collection, management, and analysis of microscopy data present numerous informatics challenges. Appropriate user tools are needed to support acquisition, creation, analysis, annotation and interpretation of microscopy data. The Open Microscopy Environment (OME) is a database-driven system for the storage, analysis, management, and interpretation of microscopy data. An examination of these tasks in the context of regularly occurring structures in OME's data model motivates a framework for constructing these tools. Initial implementations of several components, linked together in a common environment, provide a partial realization of this framework.*

## 1. Introduction

The collection, management, and analysis of microscopy data present numerous informatics challenges. A robust data model for storage and description of images, acquisition, and related meta-data must be provided. Analysis tools based on the use of machine vision and related techniques are needed to automatically extract biologically relevant understanding from raw images. Workflows composed from these tools will support the development of standardized analysis protocols. The complete data lineage of all analyses should be stored along with the images. Appropriate user tools are needed for creation, analysis, annotation and interpretation of data at every stage of the process. As types of data, research questions, and analysis techniques are likely to both vary across experiments and evolve over time, the data model and the analysis tools must be extensible and customizable.

The Open Microscopy Environment (<http://www.openmicroscopy.org>) is a database-driven system for the storage, analysis, management, and

interpretation of microscopy data [8]. The OME Data Model includes detailed descriptions of image acquisition parameters, experimentally driven description of sets of images, and analytic components such as computational modules (executable implementations of computational algorithms) and workflows. The OME Data Model is represented in an XML file and within a database schema [17]. OME provides methods for augmenting the model to support the local requirements of individual laboratories or facilities. With these extensions, the OME Data Model can be used to create a detailed, customized view of microscopy data.

To make the most of this flexible and extensible data model, scientists will need powerful tools for data exploration and interpretation. A discussion of typical data exploration tasks will motivate a framework for constructing tools that support scientists in their work. Initial implementations of several user tools for OME data, linked together in a common environment, provide a partial realization of this framework.

These tools and data models are similar in spirit to related proposals for genomic and microarray data. Microarray data models like MGED/MIAME [2] and classification systems like the Gene Ontology [7] provide common vocabularies and structures for discussing microarray data and gene function respectively. Interactive tools for the interpretation of microarray results (e.g., [9,15]) sequence analyses (e.g., [4,12]), and other biological data support examination of large data sets. Tools for creating and executing analysis workflows have been also been proposed [14].

While many of these systems attempt to support one aspect of the process of acquiring, analyzing, and interpreting data, OME is an “end-to-end” system, covering all of these tasks. As a result, analysis tools for OME must support the tasks of managing and identifying large datasets and interacting with them over extended periods of time.

## 2. The Open Microscopy Environment

The Open Microscopy Environment (OME) is the combination of a data model for describing 5D (three-dimensional space, time, and spectral wavelength) microscopy experiments in cell biology and a client-server suite of software tools for storing, manipulating, retrieving and visualizing elements described in the data model [8].

The OME Data Model describes the structure and content of microscopy data and experiments on that data. The structure of the experiments is defined in terms of *projects*, each of which contains one or more *datasets*, each containing one or more *images*. These relationships are non-exclusive, forming a “quasi-hierarchy”: a dataset can be associated with multiple projects, and an image with multiple datasets. An image can contain one or more *features* – OME’s term for sub-image items of interest. Features can be hierarchical.

The content of microscopy data is defined via an XML meta-data model. The model defines a rich vocabulary for describing instruments used to collect microscopy data - including objectives, filters, etc. These components and others are defined as *semantic types* – structured data definitions. OME users can extend the core vocabulary by declaring their own semantic types. Semantic types that divide images into experimentally meaningful sets - such as screens, plates, and wells - can be particularly useful for screening applications. These types can create additional quasi-hierarchies: a plate might participate in more than one screen.

OME also provides facilities for computational analysis of images. A *module* is a computational unit, with *formal inputs* and *formal outputs* describing the parameters and the products of the computation. Each input and output has a *semantic type* that describes how its content is to be interpreted. Modules can be composed into workflows known as *analysis chains*. An output from one module can only be used to provide value for an input of another module if they have the same semantic type. As a chain contains multiple modules, which may themselves be found in many chains, the analysis framework defines an additional “quasi-hierarchy”. Given a specification of an analysis chain and a set of inputs, OME’s *analysis engine* will use the inputs to execute the modules in the chain.

The concept of *data lineage* – the components and processes that led to the derivation of data – has been identified as an important component of scientific databases [3]. OME’s analysis engine stores the results of executing a module as instances of the semantics types corresponding to the module’s outputs. Derivation links between these module executions are

also stored, providing a complete data history for analytic results.

OME data is stored in a relational database and accessed via an object-relational mapping layer written in Perl. Images can be retrieved by a stand-alone image server. A web interface and a suite of standalone tools written in Java provide user access to OME data. Data exchange between OME installations is facilitated by import/export tools that translate data between external XML representations and the OME database. Additional details about the OME data model, the OME image file, and the OME software suite can be found in Goldberg, et al. [8].

## 3. Information Exploration Tasks in OME

Biologists and others using OME to interpret microscopy data will likely engage in two main types of exploration tasks – navigation and visualization.

### 3.1 Navigation

The primary goal of navigation is the identification of relevant subsets of a large data store. OME’s representation of experiments provides an obvious starting point for this task. Identification of the appropriate project and dataset may be the first steps in an analysis session. Selection of one or more images of interest within a dataset may be a subsequent task. These tasks are complicated somewhat by the many-to-many relationships between projects, datasets, and images.

Analysis chains, modules, and semantic types provide an alternate perspective on the data. When approaching microscopy data from an analytic point of view – for example, in the hopes of identifying images that have had certain types of statistics calculated – this approach may be more useful than the selection of projects, datasets, and images. Similar perspectives can be created through the definition of semantic types.

Other views based on meaningful organizations of the data might also prove useful. For example, timeline or history views can help users leverage their memory of past events to help find items of interest.

A secondary goal of this task will be the identification of connections and relationships between items. Although this task is closely related to visualization and interpretation, appropriate visual feedback for navigation tasks might help users place data items in a context that will help them properly interpret data.

### 3.2 Visualization and Interpretation

Interpretation of microscopy data begins with rendering of the images and associated features derived from analyses. Depending on their form, image analysis results can be either displayed as overlays on

rendered images, or in a more abstract view that lacks a direct visual correspondence with the underlying image.

The display and availability of relevant contextual information can be a powerful interpretative tool. For example, interpretation of analysis results for an image might require examination of results for other images. The identification of this contextual information might require additional navigation. Given analysis results for an image, a scientist might be interested in exploring a list of other analyses that have been conducted for that image, or perhaps in other images that have similar results.

OME's data history model provides an additional opportunity for visualization and interpretation. The lineage of an analysis result – both in terms of its ancestors and descendants - provides context that aids in interpretation. A lineage visualization might also serve as a navigation aid.

## 4. A Design Framework

An information exploration tool designed to support these tasks will have several interacting components. These components must individually support specific tasks, with interactions between them providing additional feedback and context that can assist users with navigation and interpretation.

### 4.1 Quasi-Hierarchical Browsers

The quasi-hierarchical subdivisions of OME data provide both a challenge and a starting point for the design of exploratory tools. Although the navigation of graphs and trees is a well-studied topic in the Human-Computer Interaction literature [16], most approaches are either fully hierarchical (tree browsers) or fully general (network browsers). Quasi-hierarchical data might best be served by a hybrid that combines the simplicity of a tree viewer with additional contextual information. Thus, a project, dataset, image browser includes views that show not only the lists of datasets that might be selected from a given project, but also an indication of which other projects each dataset is associated with.

### 4.2 Simultaneous Menus

The hierarchical structures described above provide alternative means of selecting items of interest. Independent and simultaneous navigation of these hierarchies can be useful [6,10]. Starting from a complete view of a user's data, selection of an item in any of the available hierarchies would restrict the view to the subset of the data that falls under the selected item. Subsequent selections – in any hierarchy – would conjunctively filter the result set.

### 4.3 Filtered Views and Semantic Zooming

As users apply filters to select data of interest, semantic zooming – the use of different displays for an object, depending on the scale of interest [1] – will be used to ensure effective utilization of available screen space. An overview display of thousands of images will show iconic representations of each image, whereas thumbnails will be shown for hundreds of images, and feature overlays on each image will be displayed for a few tens of images.

### 4.4 Coordination and Multiple Views.

Multiple, linked views of data can prove helpful for building understanding of relationships in complex data sets [13]. For OME's relational data model, this means views that present data in different contexts. A module can be depicted both in a catalog of available analysis modules, and in one or more analysis chains. Coordination of views – selection (via mouse click or mouse over) of the item in one view leading to the highlighting of related items in other views – illustrates these relationships. For example, mousing over a dataset leads to highlighting of all analysis chains that have been applied to a dataset.

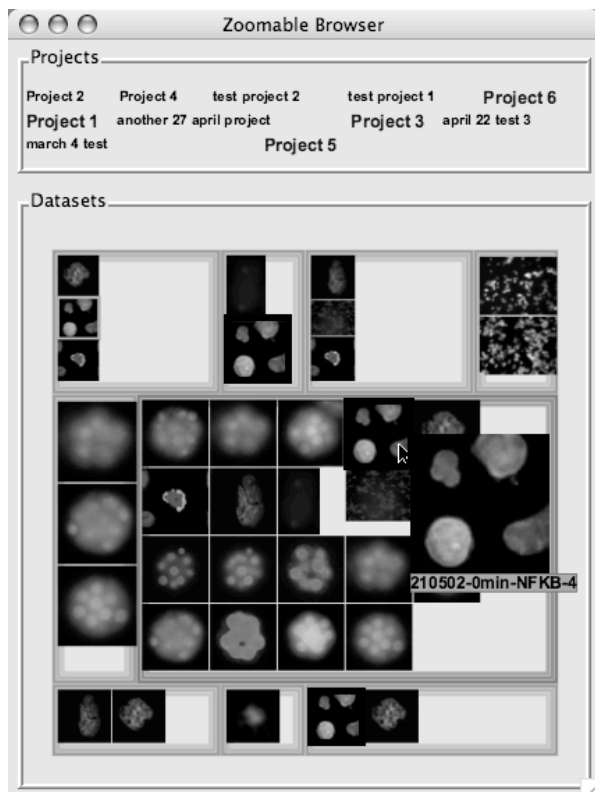
### 4.5 Visualization of analysis results

Different types of analysis results will require different visualizations. Like other views, these should be coordinated: Given a dataset that has been analyzed in various ways, linked views of all results could be used to compare results for one image to results for others in the same dataset, both for any single result type and across related result types.

## 5. Examples

The *zoomable browser* contains a pane for projects, and one for datasets. The projects pane lists available projects by name, while the dataset pane shows a rectangular space divided up between the available images. The quasi-hierarchical relationships are illustrated by linked highlighting. When the mouse enters a dataset, corresponding project names are drawn in a larger font. Similarly, selection of an image in one dataset leads to highlighting of that image in other datasets in which it is contained. Images in a dataset are displayed as thumbnails. For large datasets, zooming facilities are provided to focus on individual thumbnails (Figure 1).

The *chain builder* supports interactive examination of analysis chains and modules. Chains and modules are displayed in two separate windows, with modules being divided into categories. New

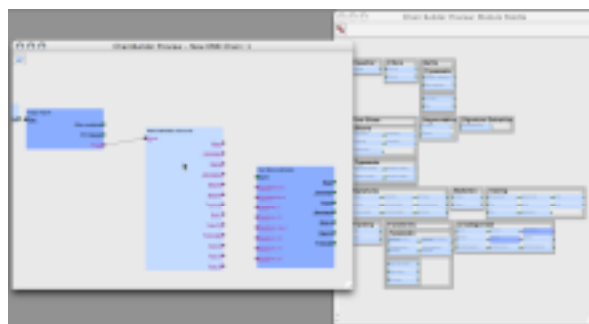


**FIGURE 1: OME’s zoomable browser. The top pane list active projects, with datasets shown in individual rectangular regions below. Projects containing the selected dataset are shown in larger font.**

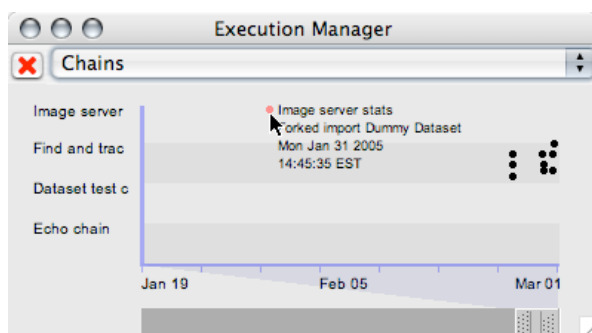
chains are created by dragging modules or existing chains to blank canvases. Links are created by click-dragging between inputs and compatible outputs. Coordinated highlights are used to show possible connections between modules: when the mouse moves over a module, other modules that might be connected to the selected module are highlighted. Semantic zooming facilities display modules as large rectangle with all inputs and outputs shown explicitly, or as small boxes containing only one output or input (Figure 2).

The *data manager* provides a classic tree-view of the project/dataset/image hierarchy, and also provides tools for annotating and classifying images. This tool can be used to launch many of the other OME Java clients.

OME’s *image viewer* includes facilities for navigating and rendering 5\_D images, image zooming, and magnifying specific image regions. In addition, the *ROI analysis* tool allows user to define regions of interests (ROIs) across space, time and channel and



**Figure 2: The chain builder, with a chain in progress on the left and the module palette on the right. Modules that can be linked to the selected module are painted dark blue.**



**Figure 3: Execution History Viewer. Analysis chain executions are arranged by time in a 2Dplot. Executions are organized vertically by datasets.**

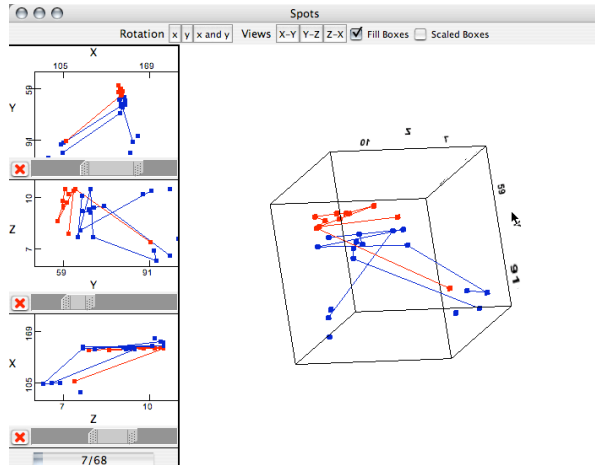
obtain quantitative measures of image pixel values, both in raw numeric and graphical form.

An *execution history* viewer displays all executions of analysis chains on a timeline. Designed to take advantage of the power of temporal relationships in interpreting data [5], this view shows displays a dot for each chain execution on a timeline. The vertical axis of the display is used to separate executions by chain and by dataset, via nested bands. A double-thumb slider can be used to limit the display to a subset of the overall time range (Figure 3).

Finally, a *3D trajectory viewer* demonstrates the possibilities for displaying results of specific analyses. OME’s “Find and track spots” chain locates regions in an image and tracks their position across multiple time points. The viewer displays these trajectories as chains of points in a 3D region, along with 3 2D-projections. The 3D display can be rotated arbitrarily.

Each 2D projection has a double-thumb range slider that can be used to adjust the displayed range in one of the three dimensions (Figure 4).

These displays are coordinated through linked selection and highlighting. Each display is implemented as a separate “agent” in OME's Java client architecture, which supports message passing for



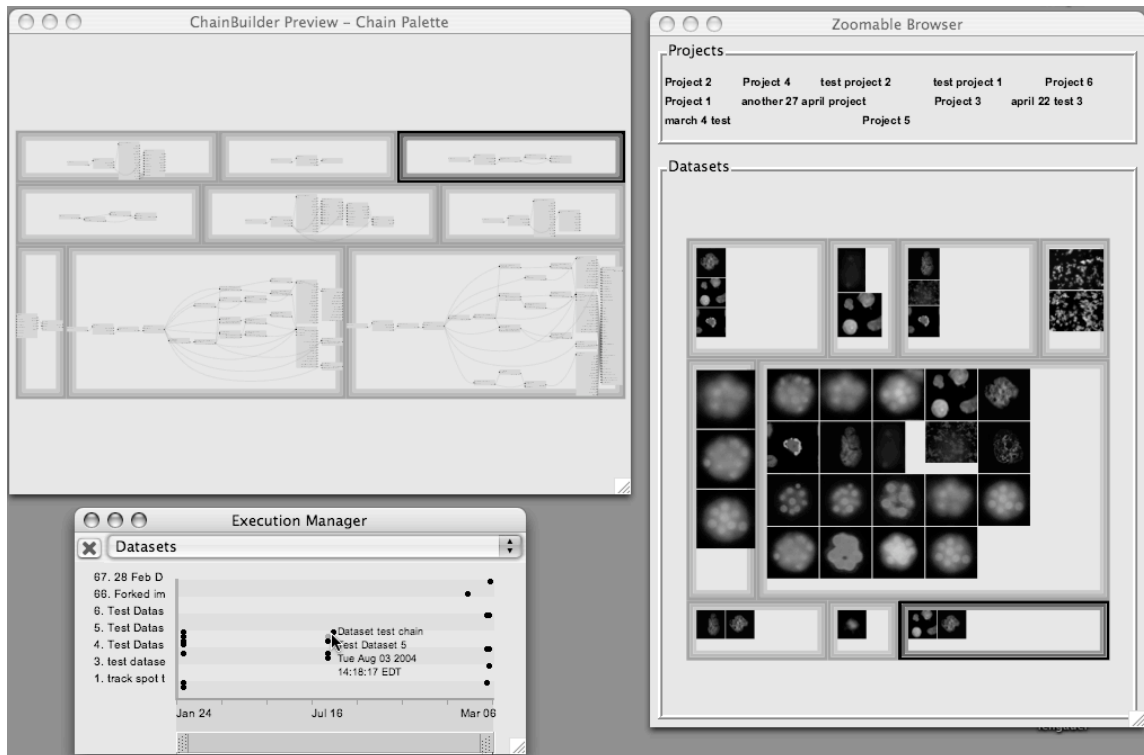
**Figure 4: A viewer for 3D trajectories as derived by an OME analysis chain.**

coordination between agents. These messages are used for coordinated highlighting between views. Thus, selection of an execution in the execution history viewer might lead to the highlighting of both the corresponding dataset in the zoomable browser, and the corresponding chain in the chain palette. (Figure 5).

## 6. Generalization

A generalized hierarchy browser will support any of the hierarchies described above. Combinations of two or more of these browsers – each for a different hierarchy – will support simultaneous menu navigation. These browsers would be used together with a window that defines an active dataset. Initially, this window would provide a high-level view of the contents of the data store. Selections made in the various hierarchies would filter the active data set, which would display more detail as the size of the set decreases (semantic zooming - Section 4.3). Coordinated highlighting of the various hierarchy browsers would provide further feedback.

A mockup of this general navigation system (Figure 6) shows how these ideas might be implemented. Initially, the browser would show a pair of hierarchy browsers surrounding a data overview. To



**Figure 5: Coordinated views: selecting a chain execution in the chain history viewer leads to highlighting of the associated dataset and chain in other viewers.**

the left of the overview, one browser would allow for selections in projects and datasets. A second browser across the top of the overview could be used to select analysis chains, modules, and semantic types. As selections are made in the hierarchy browser, the active dataset shrinks. Eventually, individual image thumbnails might be shown. Note that the hierarchy browsers are themselves semantically zoomed, as rectangles are used for datasets, modules, and semantic types when the current sets are large.

The hierarchy browser should be general enough to accommodate unforeseen hierarchies created by the addition of new structures of semantic types. This might extend to the data model.

Generalized coordination between views can involve event responses beyond simple highlighting. Aggregation and disaggregation facilities might be helpful for moving between summary statistics that describe all of the images in a dataset and details of any given image. Other coordination facilities include refocusing existing views based on user action in another view, or launching new views.

Individual data viewers for specific analysis results will be needed. Generalization of these tools will involve construction of interfaces and specification of function as needed for extending existing visualization tools to newly defined semantic types when appropriate.

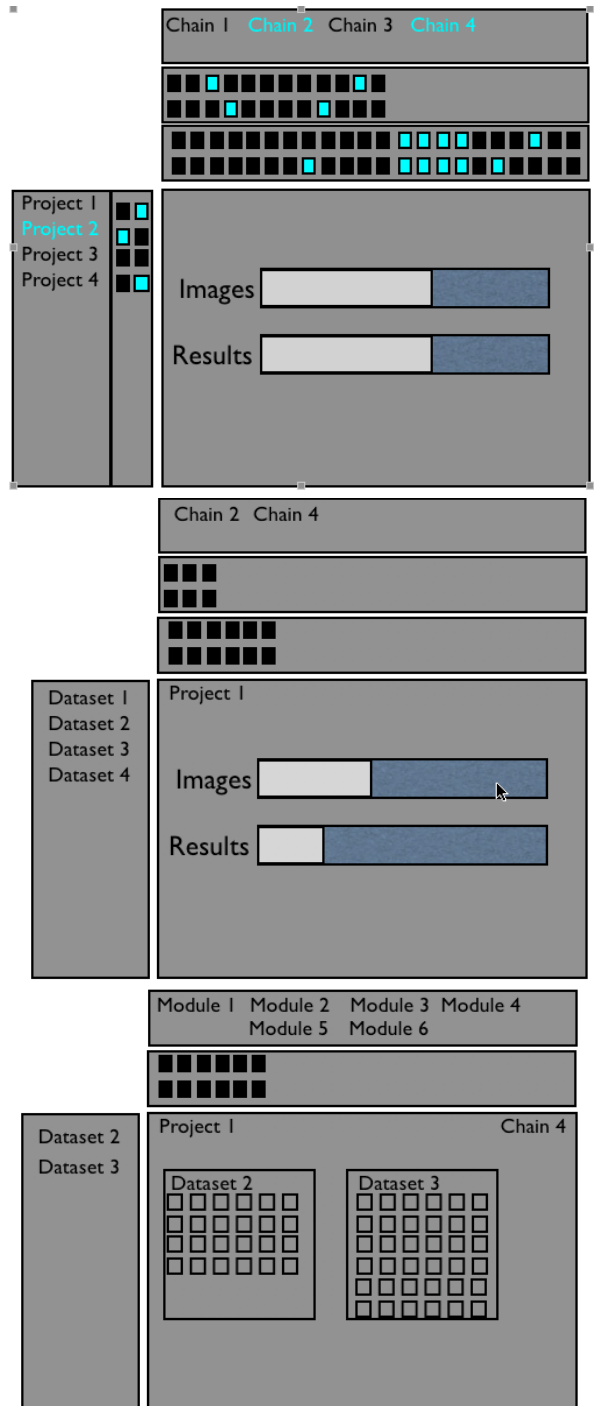
## 7. Conclusions and Future Work

Biologists interested in interpreting data in a microscopy database will need flexible tools for interactive exploration. These tools should offer multiple ways to identify data of interest and view analysis results.

Visualizations of other components of the OME data model will provide additional context for interpretation. Image acquisition metadata can be used to compare images and identify systematic problems with acquisition equipment. Manually created user annotations – whether free text or structured – might provide the basis for display and search tools.

Appropriate software engineering infrastructure, including class libraries and frameworks, will be necessary for generalization and reuse of these tools. Code re-use will be particularly important for visualization of analysis results. A suite of generic visualizations (scatter plots, graph/network displays, etc) will provide the basis for customization for specific analysis results. Similarly, the construction of a generalized hierarchy browser will require the definition of appropriate programming interfaces.

As the interpretation of scientific data is an ongoing process, these tools should include facilities for managing exploration across multiple sessions.



**Figure 6: Mockup illustrating possible implementation of simultaneous hierarchy navigation with a data browser. As selections are made in the project/dataset menu (left) and the chain/module menu (top), the set of active data shrinks. When the set is small enough, individual image thumbnails are shown.**

Bookmarks, ongoing annotations, and other tools for describing insights and describing scientific arguments [11] can be constructive in this regard.

## ACKNOWLEDGEMENTS

Jean-Marie Burel, Andrea Falconi, Josiah Johnston and Jason R. Swedlow provided useful design suggestions and assistance.

## REFERENCES

- [1] Bederson, B. and Hollan, J. 1994, Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics. *Proc. UIST '94 Conference: User Interface Software & Technology*, 17-26.
- [2] Brazma, A. et al. 2001, Minimum information about a microarray experiment (MIAME) – towards standards for microarray data. *Nature Genetics* 29(4), 365-371.
- [3] Bose, R. and Frew, J. 2005, Lineage Retrieval for Scientific Data Processing: A Survey. *ACM Computing Surveys*, 37(1), 1-28.
- [4] Clamp, M. et al. 2003. Ensembl 2002: accommodating comparative genomics. *Nucleic Acids Research*. 31: 38-42.
- [5] Freeman, E. and Gelertner D. 1996, Lifestreams: A Storage Model for Personal Data, *ACM SIGMOD Record*, 25(1), 80-86.
- [6] Furnas, G. and Zacks, J. 1994, Multitrees: Enriching and Reusing Hierarchical Structure. *Proc. CHI '94 Conference: Human Factors in Computing systems*, 330-336.
- [7] The Gene Ontology Consortium: 2000, Gene Ontology: Tool for the Unification of Biology. *Nature Genetics*, 25, May 2000, 25-29.
- [8] Goldberg, I.G. et al. 2005, The Open Microscopy Environment (OME) Data Model and XML File: Open Tools for Informatics and Quantitative Analysis in Biological Imaging. *Genome Biology*, 6:R47.
- [10] Hochheiser, H. and Shneiderman, B. 1999, Performance Benefits of Simultaneous over Sequential Menus as Task Complexity Increases. *International Journal of Human-Computer Interaction* 12(2), 173-192.
- [11] Kuchinsky A., Graham, K., Mo. D., Adler, A., Barbara, K., and Creech, M. 2002, Biological Storytelling: a software tool for biological information organization based upon narrative structure. *Proc. Advanced Visual Interfaces '02*.
- [12] Lewis, S., et al. 2002, Apollo: A Sequence Annotation Editor. *Genome Biology*, 3(12).
- [13] North, C. and Shneiderman, B. 2000, Snap-Together Visualization: A User Interface for Coordinating Visualizations via Relational Schema. *Proc. Advanced Visual Interfaces 2000*, 128-135.
- [14] Oinn, T., Addis, M., Ferris, J., Marvin, D., Greenwood, M. Carver, T., Pocock, M.R. Wipat, A., and Li, P. 2004, Taverna: A tool for the composition and enactment of biological workflow. *Bioinformatics* 20(17), 3045-3054.
- [15] Seo, J. and Shneiderman, B. 2002, Interactively Exploring Hierarchical Clustering Results. *IEEE Computer*, 35(7), 80-86.
- [16] Shneiderman, B., and Plaisant, C. 2004, *Designing the User Interface*, Fourth Edition. Boston: Addison-Wesley.
- [17] Swedlow, J.R., I. Goldberg, E. Brauner, and Sorger, P. K.. 2003, Informatics and Quantitative Analysis in Biological Imaging. *Science*, 300, 100-102.