CHAPTER 4.3

The Neuroanatomical Rat Brain Viewer (NeuARt)

Ali Esmail Dashti,[1] Gully A. P. C. Burns,[2] Shahram Ghandeharizadeh,[3] Shuping Jia,[4] Cyrus Shahabi,[3] Donna M. Simmons,[2] James Stone,[5] and Larry Swanson[6]

[1]*Computer Engineering Department, College of Engineering and Petroleum, Kuwait University, Al-Khaldya, Kuwait*
[2]*Department of Biological Sciences, University of Southern California, Los Angeles, California*
[3]*Department of Computer Science, University of Southern California, Los Angeles, California*
[4]*Intuit Inc., San Diego, California*
[5]*Neuroscience Program, University of California, Davis, California*
[6]*The Neuroscience Program and University of Southern California Brain Project, University of Southern California, Los Angeles, California.*

4.3.1 Introduction

A neuroanatomical atlas is an idealized three-dimensional map of brain tissue. Like any map, it can be used to navigate (by identifying the position of apparatus within the brain with stereotaxic coordinates within experiments), and it may be used to provide a standard "geographical" frame of reference (for data from neuroscientific experiments). This chapter describes an ongoing informatics-based project to simplify and optimize the process of superimposing neuroanatomical data onto the plates of a brain atlas through the use of multimedia databasing and network technology. The NeuroAnatomical Rat Brain Viewer (NeuARt) is based on neuroanatomical extensions to the NeuroCore database system, allowing neuroscientists to browse, compare, and query the complex spatially distributed patterns of label obtained from different experiments at their desktops through an network connection.

Consider the initial, primary form of neuroanatomical data as a large number of sections mounted onto slides providing a stacked three-dimensional map of different staining in the tissue at intervals through the brain. The physical characteristics of the physical preparation of these slides affect the viewing conditions of the data, so that one could almost think of the primary data as an analog data resource that only can be interfaced with through the use of a microscope. The primary data contains the full wealth of information that is present in the stained tissue and subsequent stages progressively simplify this information.

Neuroanatomists often depict their data by faithfully representing individual cases with photographs or *camera-lucida* drawings or by performing detailed examination of the tissue with computational tools (with products such as NeuroLucida from MicroBrightField, Inc., or the SEM Align tool developed as part of the Human Brain Project; Harris, 1999). Such representations are usually exquisitely detailed and very compelling and are becoming increasingly powerful when combined with high-powered computational analysis. However, their complexity makes them difficult to generalize from and computational methods to automate the process of transforming such results into a standard frame of reference (such as an atlas) cannot perform at the required level of accuracy required by expert anatomists.

Over a period of roughly 20 years, workers within the Swanson laboratory have amassed a wealth of original research data from anterograde tract-tracing experiments, retrograde labeling experiments, *in situ* hybridization and immunohistochemical staining (e.g., Risold and Swanson, 1997; Swanson, 1987; Swanson and Cowan, 1975;). The neuroanatomical data that will be considered are *Phaseolus vulgaris leuco-agglutinin* (PHAL) tract-tracing data (Gerfen and Sawchenko, 1984). These are by no means representative of all different types of neuroanatomical data, but they provide a starting point from which we intend to generalize.

Before workers in the Swanson laboratory publish their PHAL data, they transfer it to atlas plates of the rat brain by examining the tissue microscopically and drawing individual PHAL-stained fibers by hand onto a digital version of the atlas (Swanson 1998). This procedure requires a high level of expertise and patience because the orientation of the plane of section of the tissue cannot correspond exactly to that of the atlas and the cutting and fixing procedures cause unpredictable distortions of the tissue. The experimenter is forced to perform some degree of subjective interpretation while generating the drawings. If sufficient care is taken over this procedure, the end product is a highly detailed and accurate representation of the labeling pattern in the histological slide, but the procedure is extremely time consuming (Dashti *et al.*, 1997).

Fig. 1 illustrates an example of this sort of data. The left-hand drawing shows atlas level 14 of the Swanson 1998 atlas with the distribution of calcitonin gene-related peptide (CGRP) drawn in the left-hand hemisphere (lines = fibers, circles = pericellular baskets; see Risold and Swanson, 1997) and PHAL fiber labeling from injections into the posterior part of the anterior hypothalamic nucleus drawn in the right (AHNp; see Risold *et al.*, 1994). The space limitations of this document only permit a small "magnified view" of the high level of spatial detail of these data. The right-hand side of Fig. 1 shows the flatmap representation of the output connections of AHNp with a small "magnified view" of the detailed organization of this connection system. At present, the work in the Swanson laboratory emphasizes the use of the PHAL tract-tracing technique (Gerfen and Sawchenko, 1984; see also Chapter 6.3 concerning the NeuroScholar project), but any sort of anatomical data could be represented in this way (i.e., as overlays on atlas plates).

Before the NeuARt project was instigated, data were shared and examined in the Swanson laboratory by importing data by examining copies of the Adobe Illustrator files that make up the atlas with imported layers representing data from different experiments. When comparing data from two or three experiments, this was satisfactory, but when we wished to look for patterns in the data over large numbers of experiments this process became extremely cumbersome, requiring in excess of 30 to 40 layers to a single file. Reorganizing the data was impossible without laboriously copying layers between files, and the process of scanning data across atlas levels required the opening of several of these large Illustrator files and switching between them.

We reported much of the conceptual background and the proposed functionality of the NeuARt application in a paper published in the journal "*Neuroimage*" in May 1997 (Dashti *et al.*, 1997) and a more recent description of the architecture of all aspects of the system appears in the Neuroinformatics Workbench Manual (see the Website for the USC Brain Project for the most recent version). The current NeuARt application consists of a graphical user interface and neuroanatomical extension tables and relations for the NeuroCore Informix database. These tables hold details of the manipulations and protocols used in anatomical experiments and the drawings and photographs of the data, combined with index information describing how these data should be overlaid onto the atlas.

4.3.2 The NeuARt System

The basic functionality of the NeuARt viewer is to allow users to query a database of results from

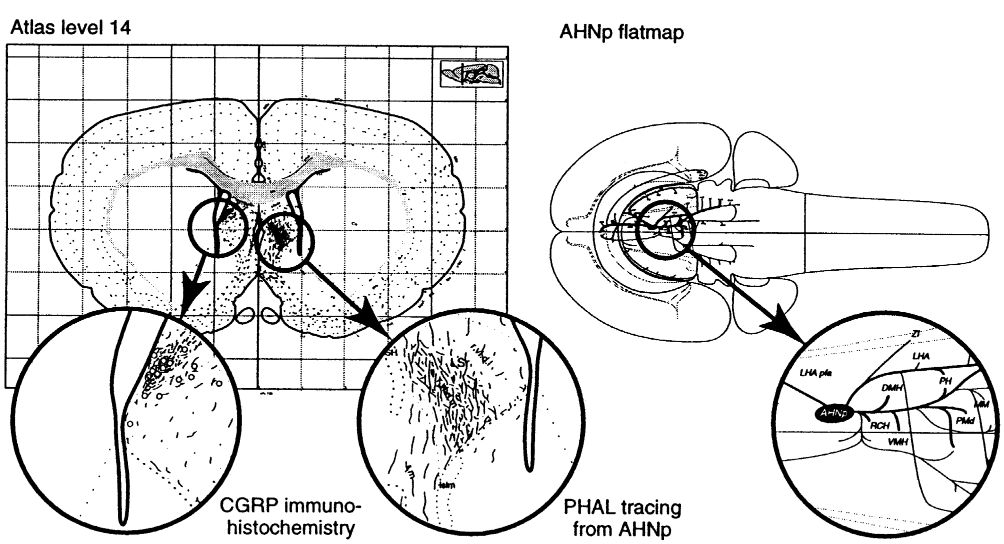

Figure 1 Illustration of detailed anatomical data from the Swanson laboratory; see text for description.

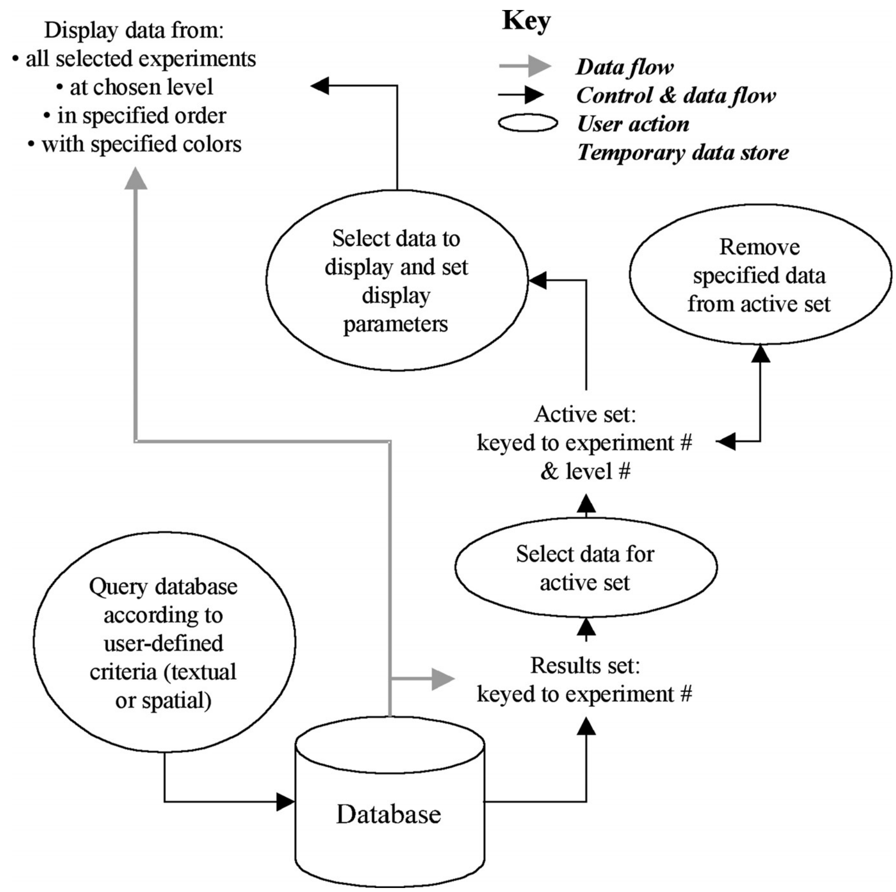

Figure 2 High-level flow diagram depicting the software requirement specification for the NeuARt system.

neuroanatomical experiments and to place results of queries into an *active set*. The display of the active set over atlas plates may be manipulated dynamically so that each individual result set may be displayed, hidden, recolored, reordered, or discarded. Users may browse data at different levels throughout the active set. The high-level organization of the system that underlies this functionality is shown in Fig. 2.

Fig. 2 illustrates the use of the system. At any time, users may perform any of the tasks shown in ellipses, assuming that the current state of the system provides those tasks with the necessary input data from either the database or temporary data stores within the application. This diagram illustrates the "two pass paradigm" method of gaining access to the database. The database serves data at two stages, first as low-bandwidth input to the results set as a simple identifying index, so users may choose to select it to add to the active set, or as the full data as input to the display. Segmenting the load on the database into these two sections means that the system does not serve full database sets unless the user actually wants to look at them. In the following sections we will discuss the schema of the Neuroanatomical database extensions to the NeuroCore system that we use to store the data and the programming design of the user interface itself.

Neuroanatomical Extensions for the NeuroCore Database

There are many kinds of neuroanatomical data. In fact, there are as many as there are different experimental procedures. An increasingly diverse number of sophisticated staining methods are available to neuroanatomical researchers. Data can be viewed in different ways and at different scales with light, confocal, electron, or magnetic resonance imaging microscopes. Neuroanatomical methodology is becoming increasingly more diverse, meaning that the schema used to capture the data must be extensible to be able to accommodate for new types of data.

The organization of the database underlying NeuArt is an extension to the NeuroCore system, meaning that not only is the basic design extensible, but non-neuroanatomical aspects of the data use the same schema as other NeuroCore databases. This creates a broad common substrate which makes the process of designing common tools more straightforward. The database can accommodate this variability by extending existing relations or adding new relations as the complexity of the data grows. Within this document, we will refer to the combination of the core section of the database (NeuroCore) and the neuroanatomical extensions (AnatExt) as the Neuroanatomical Repository of Data (Anat-RED).

The format of the data that are currently being stored in the Anat-RED are based on the methodology of a single laboratory, that of L. W. Swanson at the University of Southern California. Comparing data from different laboratories can be extremely difficult, if the underlying nomenclature of each laboratory is different, if different techniques are used, or if the data are presented in a different computational format. By presenting a way of viewing the data of a well-respected research laboratory with reference to a standard atlas (which, incidentally, was produced by the same group), we seek to illustrate the potential of this approach by example. If other laboratories find these methods useful, then NeuARt could provide a means to compare data between laboratories.

The central challenge presented to the designers of Anat-RED is to provide a unified format that captures the data of different laboratories but still represents the neuroanatomical data at a suitable level of complexity to allow the use of multiple atlases, nomenclatures, and imaging systems. The Swanson (1998) atlas has an extensive glossary, which will soon be incorporated into RED's lookup tables to provide a way of cross-referencing the nomenclature with structures that are defined as shapes in the atlas. At the present time, the neuroanatomical data under investigation are taken from adult rats. The general framework of the Anat-RED could be applied to any species or age, as long as a brain atlas for that animal could be incorporated into the database.

The tables of the NeuroCore section of the Anat-RED provide the general framework for neuroscientific experimental data. The extensions we describe here (AnatExt) provide the support to represent specific neuroanatomical research data. There are two classes of extensions: *neuroanatomical manipulations* (AnatManip) and *neuroanatomical research data* (AnatData). The AnatManip extension defines the possible manipulation techniques associated with anatomical protocols. The AnatData extension defines the spatial data associated with the anatomical data.

We have defined five classes of anatomical manipulations (Fig. 3):

1. *Chemical treatment:* These database classes describe drug treatments, specific diets, or experimental manipulations that have been performed on an animal prior to experimentation.
2. *Tracing:* These classes describe the procedures involved in performing a tract-tracing experiment; for example, the data in this section would describe the injection parameters and anesthetic regimen employed in a PHAL injection.
3. *Fixation:* These classes are used to describe the methods employed to fix the brain tissue. This would include a rigorous description of the perfusion methodology.
4. *Sectioning:* The classes of this section would be used to describe the precise nature of the sectioning techniques employed in the experiment, including a description of the type of cutting performed (e.g., freezing microtome, cryostat, etc.).
5. *Staining:* At the present time there are three separate subclasses to denote three different types of staining that are used to reveal the neuroanatomical properties of the tissue. They are listed below:

a. *Histochemical:* These are general histochemical stains such as Nissl or Golgi staining.
b. *Immunohistochemical:* These are immunohistochemical stains where the stained particles are attached to antibodies that recognize specific proteins.
c. *In situ hybridization:* These data refer to staining techniques that stain specific mRNA or DNA fragments.

A protocol can be made up of several manipulations, and these manipulation techniques are defined in isolation from specific protocols. Protocols can be associated with research data at a number of levels: at the level of individual experiments, sessions, segments, or subjects. On the other hand, manipulations are not associated directly with any of the research data.

The current implementation of AnatData assumes that anatomical data consist mostly of spatial data. At its most basic form, it consists of some sort of digital extraction of data from an actual experimental brain. This extraction could be either in two- or three-dimensional form, depending on the size of the brain, the experiment type, and the subject animal. The two-dimensional extraction methods usually constitute the slicing of the research brain and then using either a scanning or photographic technique to digitize the data. The three-dimensional extraction methods are based on MRI scanning technology. Our Anatomical-Research Data extensions provide the necessary framework to capture all of the above data types; however, because our lab generates mostly two-dimensional data, the focus of the AnatData extensions is based on two-dimensional data (Fig. 4).

User Interface Applet

We interpret the spatial structure of neuroanatomical data through visualization. Analysis of patterns of cellular properties in brain tissue with quantitative statistical methods can be used to make objective interpretations (Roland and Zilles, 1994; Zilles, 1985), but the use of these analyses is limited by the complexity of the tissue and the inherent difficulties of obtaining quantitative neuroanatomical data, and it is a well-known truism that the best pattern-recognition system available is currently the human visual system. Simple visualization is therefore unlikely to be superseded by quantitative statistical analyses among the majority of experimental

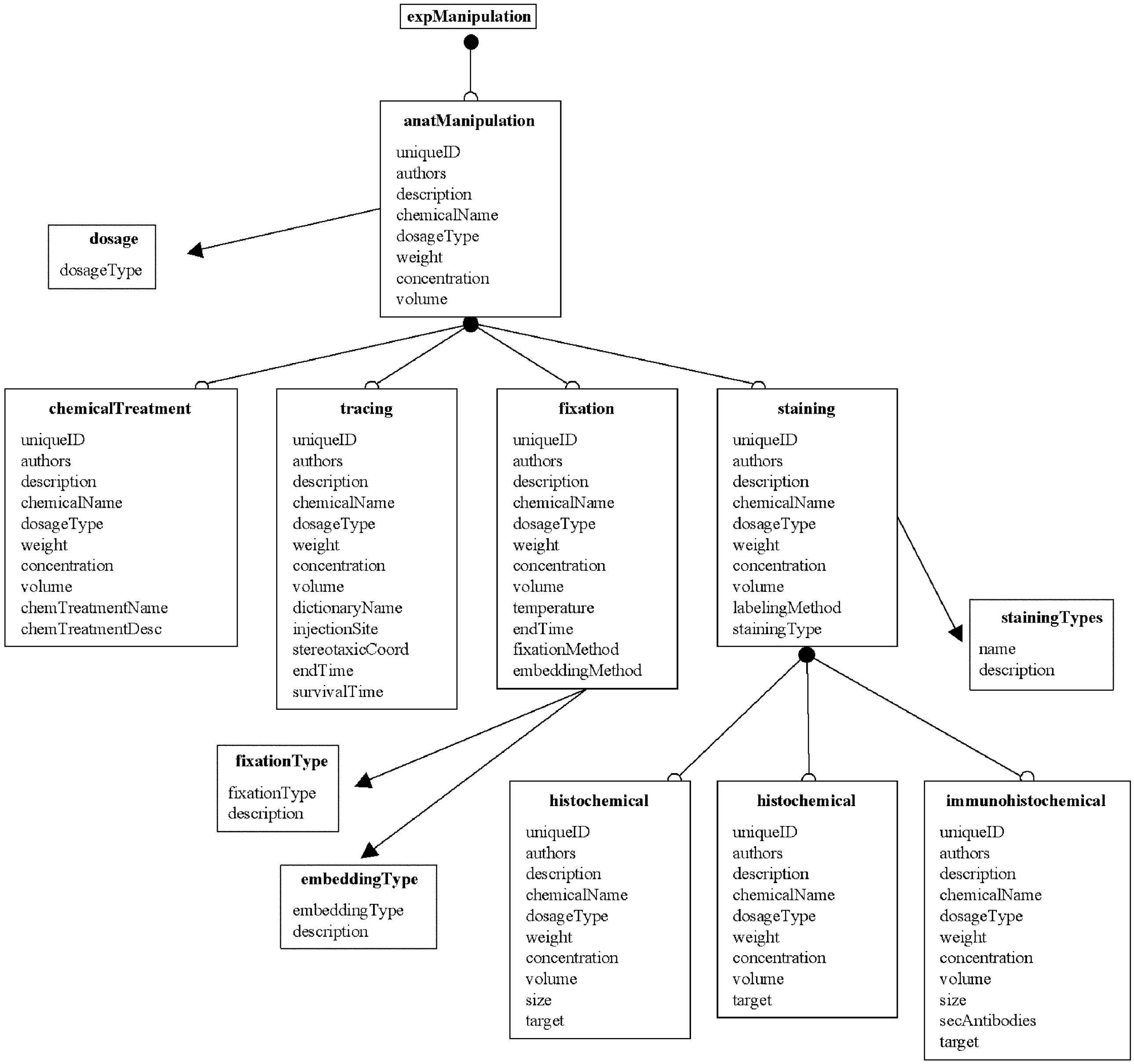

Figure 3 The database schema of the AnatManip extensions to the NeuroCore database.

workers. In any case, a visual representation of the data can be regarded as a standard requirement of any neuroanatomical storage system.

The architecture of the user interface consists of three types of modules: user-interface managers (UIM), a user-interface server (UIS), and a data server (DS). The UIM modules provide the direct interface for the user and communicate only with the UIS module, which coordinates their interactions. The UIS delivers queries to, and receives results from, the database via the data server. The specification of the database management system is contained with the DS and so can be changed, simply by replacing the DS module. This modular design will facilitate future modifications and extensions (e.g., modifying or adding UIM modules or changing the database implementation). The Swanson laboratory has a database of roughly 700 drawings from 43 separate studies in the form of layers in Adobe Illustrator.

The user interface consists of seven components: the Display Manager, the Query Manager, the Results Manager, the Active Set Manager, the Level Manager, the Viewer Manager, and the Inquire Manager. The Display Manager is where the data are visualized and may be considered the central focus of NeuARt. There are four types of interaction among the managers and the underlying data.

The first type of interaction (shown in Fig. 5A) involves the Display Manager, the Query Manager, the database (in this case, Informix Universal Server), and the Results Manager. In general, the user defines the textual and spatial query using the Query Manager and the Display Manager, respectively, and then submits this

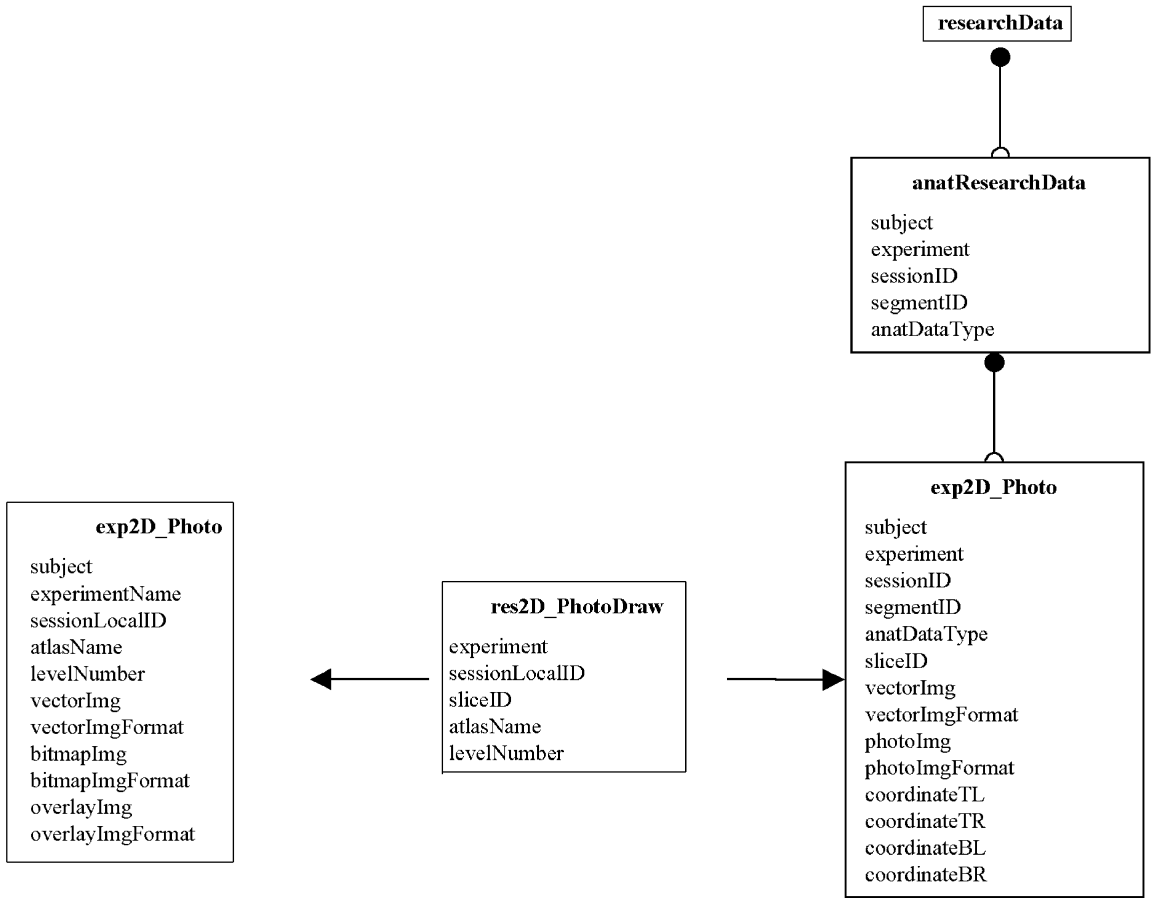

Figure 4 The database schema of the AnatData extensions to the NeuroCore database.

combined query to the database. The database returns the result to the Results Manager.

The second type of interaction (shown in Fig. 5B) involves the Results Manager, the database, the Level Manager, the Active Set Manager, and the Display Manager. After receiving the textual result of the query, the user may select a number of the experiments to be returned in their entirety for more elaborate study. This is sometime referred to as a two-pass paradigm (see above). The database returns these results into the Level Manager, the Active Set Manager, and the Display Manager.

The third type of interaction (shown in Fig. 5C) involves the Viewer Manager and the Display Manager. The user may customize aspects of the Display Manager using the Viewer Manager. The fourth type of interaction (shown in Fig. 5D) involves the Display Manager and the Inquire Manager. Using the inquire tool on the Display Manager, the user may query about the atlas structures. We have designed the underlying architecture of NeuArt to be as modular as possible. This was done in order to allow the various managers, each of which can be considered as a module, to be upgraded or replaced as new functionality is desired or as new data types become available.

The overall functionality is divided between user interface and data access, each of which is represented by an individual server. The user-interface server (UI-server) handles all communication between the managers described in the previous section. Because no manager communicates to another except via the UI-server, we were able to define clearly the interfaces between the managers and the UI-server. As long as a replacement manager conforms to these interface requirements, the system will still function correctly; however, if a new type of manager is implemented, then it is necessary to make changes to the UI-server (note that only one module has to be updated). Similar reasoning applies to the separation between the UI-server and the data server (D-server). To access data, the managers have to make the request to the UI-server, which alone communicates with the D-server. Currently, the data server is configured to communicate with an Informix database (see Fig. 13.3a). Although we have a number of future enhancements planned for NeuARt, we feel that the best enhancements will

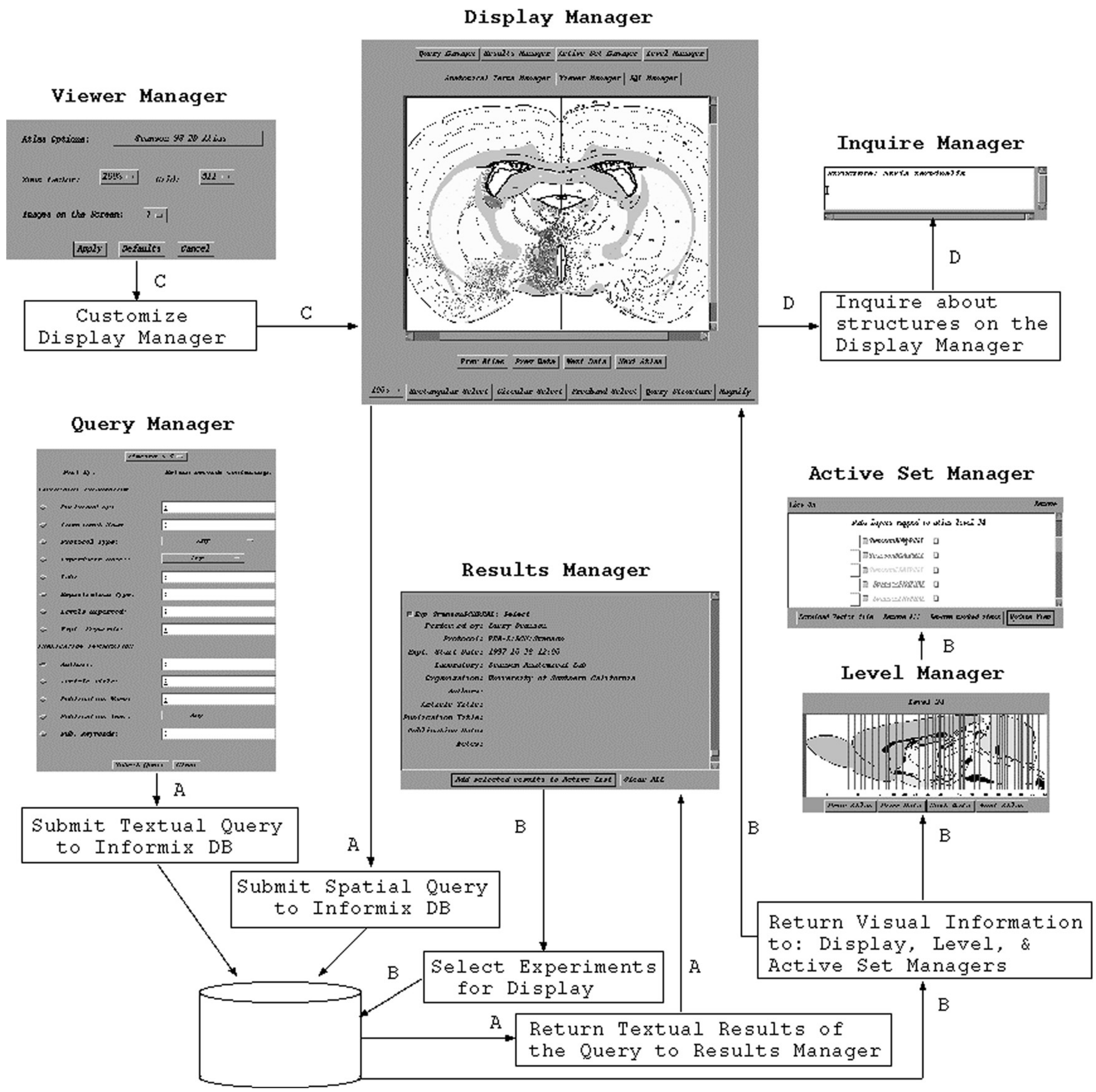

Figure 5 The organization of the NeuARt user interface, illustrating the global organization of the system.

probably result from user's comments, requests, and complaints.

DISPLAY MANAGER

The Display Manager is the focal point of most of the user interactions with the system (Fig. 7). These interactions take several forms, and the Display Manager can perform several tasks:

1. It can control the display of other managers.
2. It can control which atlas level is displayed.
3. It can enhance the ability of the user to specify spatial queries of the database.

The top two rows of buttons control the visibility of the other managers. So as not to clutter the user's view of NeuArt, the other managers will run even though they are not visible. Activating the appropriate button on the top two rows will make the corresponding manager visible. If the manager is obscured by another window, it will be raised to the top.

The third row of buttons control which atlas level is displayed. These buttons are of two types. The first type (Prev Atlas/Next Atlas) will switch the display to the previous or next atlas level as appropriate. If data layers are mapped to the newly displayed atlas level, their display will be controlled by the settings of the Active Set Manager, which we describe below. The second type of buttons (Prev Data/Next Data) will switch the display to the previous or next atlas level where data layers can be found in the active set. Any intervening atlas layers that do not contain data are skipped.

The final row of buttons enables the user to specify queries based on the spatial nature of the atlas or the data. The user may choose one of three selection tools (rectangular, elliptical, or freehand) with which to select an area of interest on the atlas. By such a selection, the

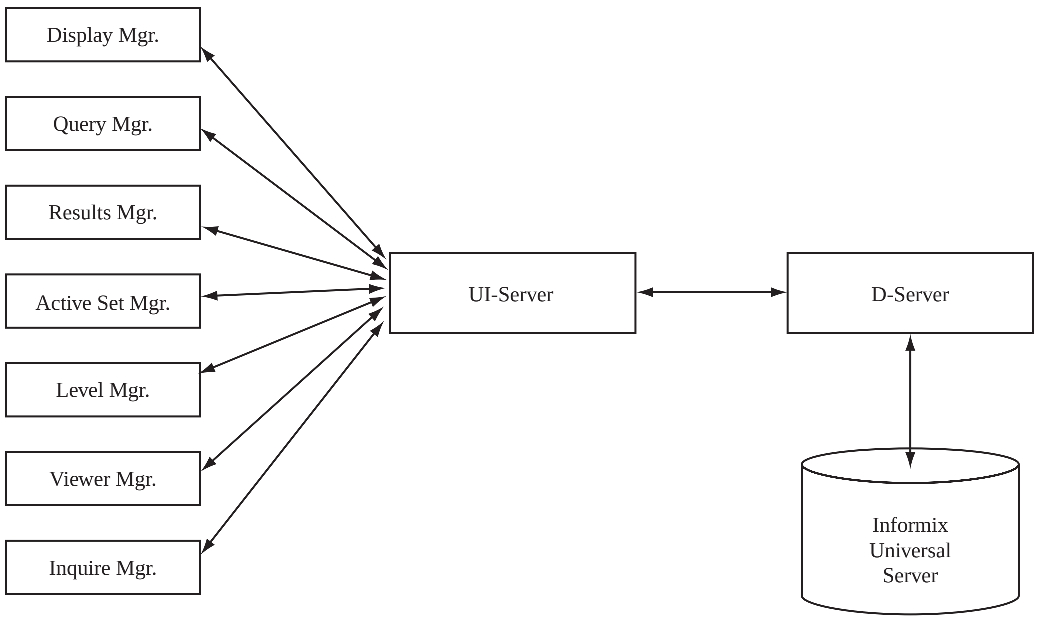

Figure 6 Relationship between NeuARt's managers, servers, and the underlying database.

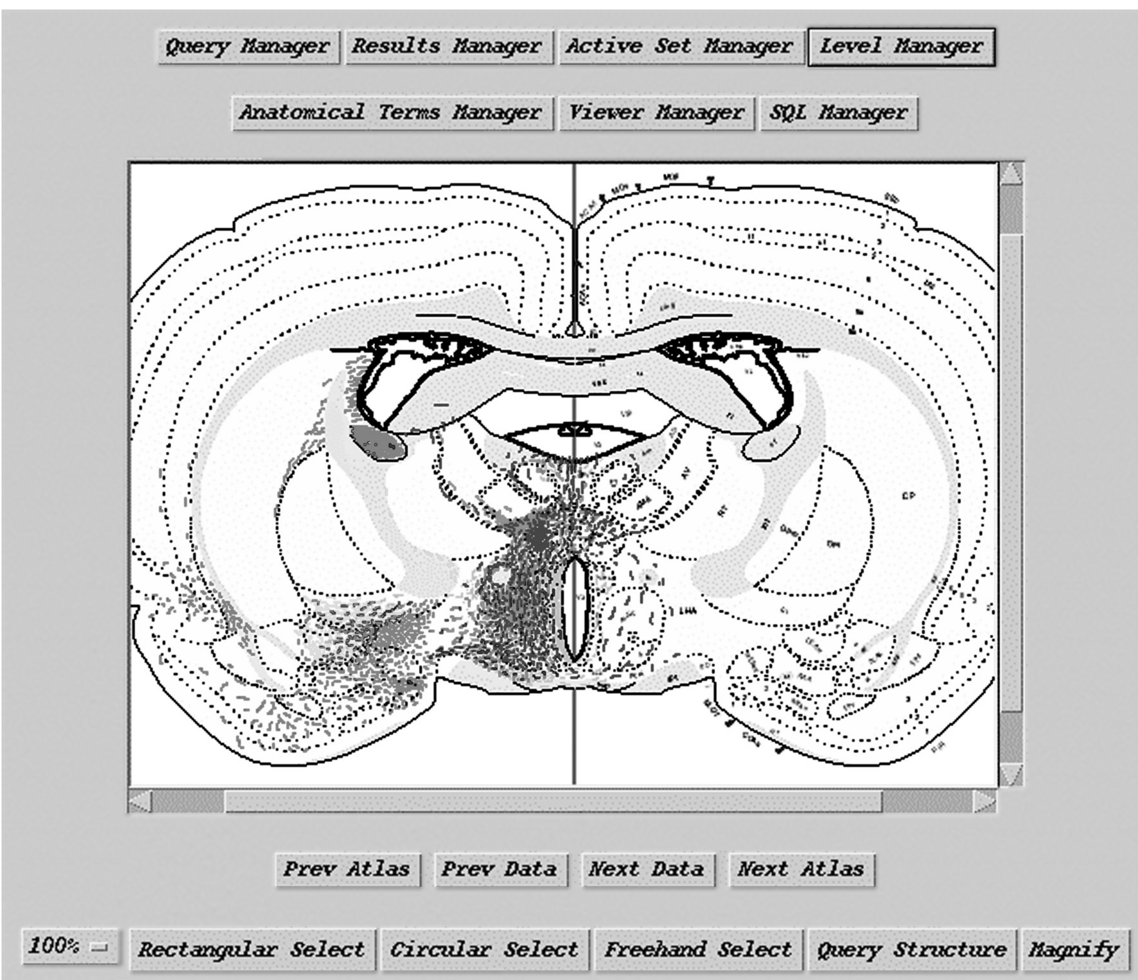

Figure 7 A screenshot of NeuARt's Display Manager.

user restricts the query so that only results that contain some data within the selected area are returned. An additional button is used to make name queries of the atlas. By the user's selecting the "Query Structure" button and then clicking on the atlas, the database will return the name of the anatomical structure in which the user clicked.

Query Manager

The purpose of the Query Manager (Fig. 8) is to enable the user to specify textual conditions that the results of the query must satisfy. Fields to specify various attributes of data from either published or unpublished sources, as well as from a variety of experimental protocols, laboratories, authors, etc., are provided. For cases in which the user will select from a limited number of cases, the user is presented with the possible choices and may select one. The operation of the Query Manager is that a blank field enforces no restriction upon the query; however, a non-blank field must be matched by the returned data. Multiple non-blank fields are combined by logical AND to determine the semantics of the query. To make multiple queries easier to perform, the Query Manager maintains a history of previously issued queries. This history may be used to fill in the fields as in a prior query; the user may then do any desired editing before issuing another query. The user submits a query to the database by activating the "Submit Query" button at the bottom of the Query Manager.

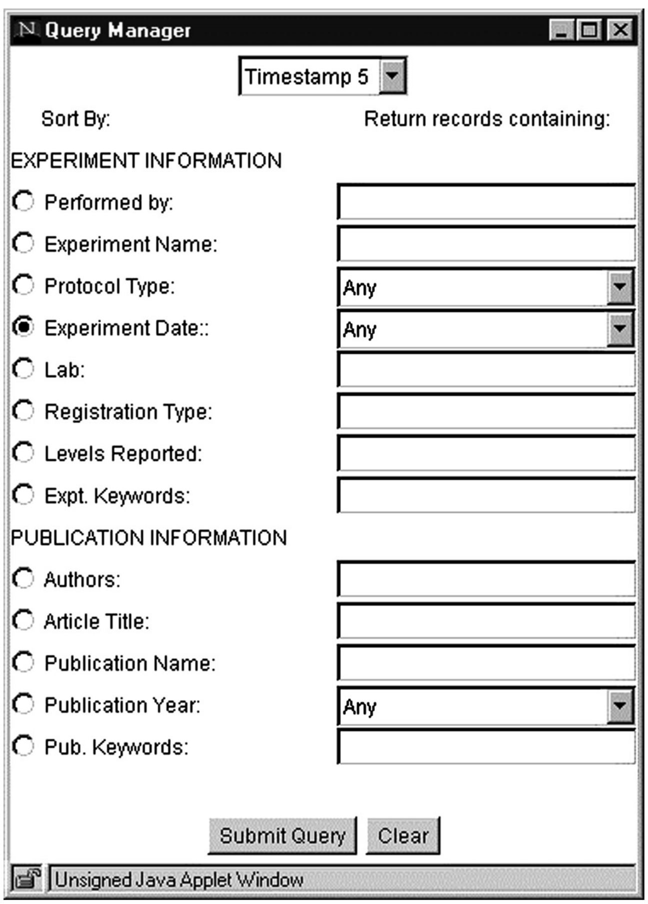

Figure 8 A screenshot of the Query Manager taken from NeuARt running as an applet over the World Wide Web.

An Additional, Related Application: The Spatial Index Manager (SIM)

The spatial organization of the regions in the Swanson brain atlas and of experimental data that can be overlaid onto them is valuable information that we wish to use within NeuARt. To this end, we have devised and implemented an application called the *Spatial Index Manager* (SIM) that permits polygons to be drawn over the atlas plates using a combination of automation and expert user intervention with a topological mapping program (Fig. 9). This process converts the atlas drawings into "intelligent templates" in which every point "knows" both the spatial extent and the name of the region that contains it. This "knowledge" is then inherited by any regional data registered against the atlas in the Neuroanatomical database and may thus support spatial queries anchored by references to particular brain regions, spatial features, or three-dimensional coordinates. The current version of SIM is implemented in the Java language. Similar to NeuARt's data server, it communicates to the Informix Universal Server via RMI. It stores the identified topological structures in Informix's spatial datablade format. Two major functions of SIM are

1. *Freehand drawing:*This function allows users to identify objects by freehand drawing polygons around them, labeling them, and storing them into the database. Through this function we can impose topological structures on both the atlas and the experimental data.
2. *Fill function and automatic boundary generation:* This function can semi-automatically identify objects with closed structures and store them as polygons into the database system. This is achieved by filling a selected closed structure with a certain color and then automatically detecting the convex hull of the colored region. Another function of this module is to check whether a freehand drawn polygon is closed or not.

Spatial Queries

The user may spatially query the atlas structures and/or query the combination of the atlas structures and the overlay data using the Display Manager. To specify spatial queries, the user may use SQM, which is designed in order to support spatial queries on both the atlas images and experimental data. SQM extends the NeuArt user interface permitting the user to: (1) point at a structure and see the name and corresponding information about the structure (including the list of publications with experiments on that structure), and (2) select an area (as a rectangle or a circle) and find all the experiments that are contained in, contain, or overlap with the

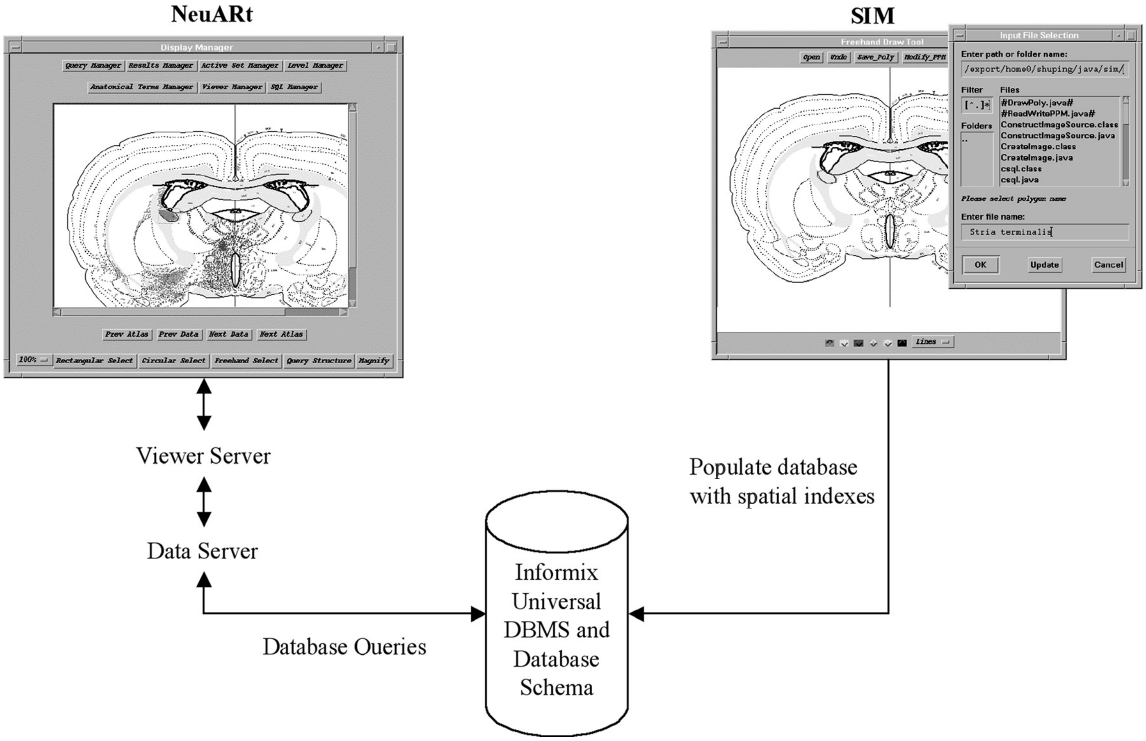

Figure 9 The computational architecture of the NeuARt and SIM systems.

selected area. SQM achieves its spatial query functionality by utilizing the Java 2D API on the user interface side and the Informix spatial datablade on the database server side. In addition, it utilizes topological information generated by the Spatial Index Manager (see above) for both atlas images and experimental data. The Query Manager utilizes the spatial description as an additional constraint to the textual attributes that it contains. After submitting a query, with the Query Manager and the spatial query tools of the Display Manager, the database returns results to the Results Manager (via the viewer server).

Results Manager

A description of all results matching a user's query are returned to the Results Manager (Fig. 10). While scrolling through these descriptions, the user selects results of interest by clicking on the "Select" checkbox for each result. When the user is satisfied that the selections are finished, clicking on the "Add selected results to Active List" button will inform the Active Set Manager (see below) that new data are being mapped to atlas layers. Should the user decide instead that the results are completely inappropriate, the user may select the "Clear All" button. This will result in the Results Manager being emptied of all results, and the user may initiate a completely new query.

Active Set Manager

The Active Set Manager enables the user to control the presentation of the data layers on an atlas level (Fig. 11). Its operation is tied closely to the Level Manager. Upon selection of a new data atlas level in the Level Manager, the Active Set Manager scrolls to show the data layers mapped to that level, and the background changes to white to indicate that it is selected. In the Active Set Manager, all of the other atlas levels containing data are also present and may be accessed by use of the scroll bars. The background for a non-selected atlas level will remain gray to indicate its status.

The entries in the Active Set Manager are the identification strings that were used to select these experiments in the Results Manager, providing a reminder of the data source. Each ID is displayed in a different color from other IDs which is the same color used to display the data layer in the Display Manager. In an atlas level with many data layers mapped to it, confusion will result if all data layers are present at all times. To aid the user in evaluating the different data layers, the Active Set Manager enables the user to change the order of display of the data layers, as well as to specify whether a data layer is displayed at all. Furthermore, the user may remove entries in the Active Set Manager by using the "Remove ALL" button and the "Remove Marked Items" button in conjunction with the checkboxes to the right of the IDs of the experimental data. The "Remove ALL" button removes all data layers mapped to the indicated atlas level, and the "Remove Marked Items" button removes only those items with selected checkboxes.

The display order of the data layers is the order in which they are listed in the Active Set Manager. The

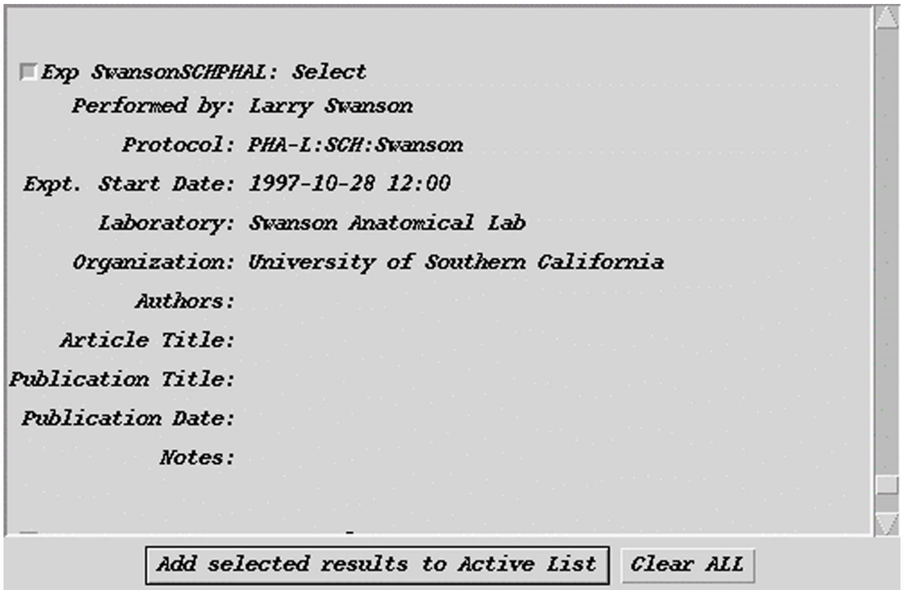

Figure 10 A screenshot of the Results Manager window, illustrating the results of a single experiment from the Swanson laboratory.

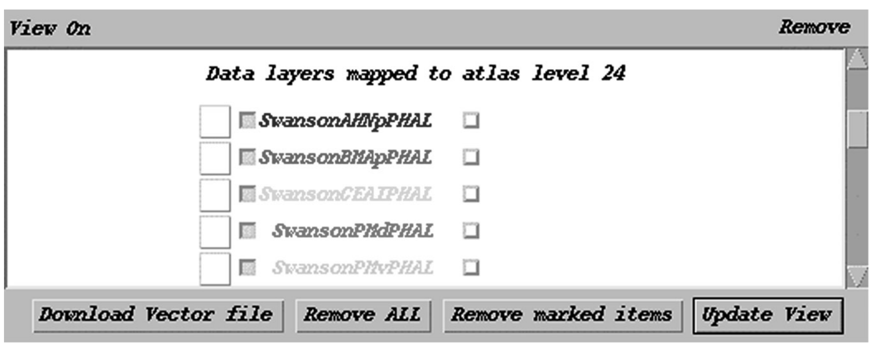

Figure 11 A screenshot of the Active Set Manager window, illustrating several datasets concurrently present on level 24 of the atlas.

uppermost-listed data layer is displayed on top of all data layers listed beneath it, and so on down the list. To change the display order, the user clicks on the square pad to the left of the ID and then moves the mouse cursor to the desired position while holding down the mouse button. Upon release of the mouse button, the list of IDs is rearranged to reflect the new display order.

In certain circumstances too many data layers may be mapped to the atlas level for the data to all be displayed simultaneously and still be understood. The user may deal with this situation by selectively displaying the data layers. This selectivity is specified by the checkbox to the left of the ID. If the checkbox is unchecked, the data layer will not be displayed, regardless of its display order. Should the user so choose, an Adobe Illustrator-compatible vector file may be downloaded to the user's system from the database, showing the selected atlas level with the data layers mapped to it. This is accomplished by clicking on the "Download Vector" file button. If the user wishes to use the atlas level in a publication, the quality of the bitmapped images displayed in the Display Manager will probably not be of sufficiently high quality.

Level Manager

It is the responsibility of the Level Manager to provide the user with feedback about which atlas levels have data mapped to them and which atlas level is currently displayed in the Display Manager and to enable the user to move from one atlas level to another. As shown in Fig. 12, the Level Manager shows a mid-saggital view of the brain with vertical lines superimposed on it. The position of the lines correspond to the position of the atlas levels. The vertical lines are drawn in one of three colors. A light gray line indicates an atlas level that has no experimental data mapped to it. Lines drawn in blue indicate that the corresponding atlas level has one or more data layers mapped to it. A red line indicates the current atlas level shown in the Display Manager. The position of the red line moves as the user changes which atlas level is currently displayed. The Level Manager has four buttons that operate in the same manner as the corresponding

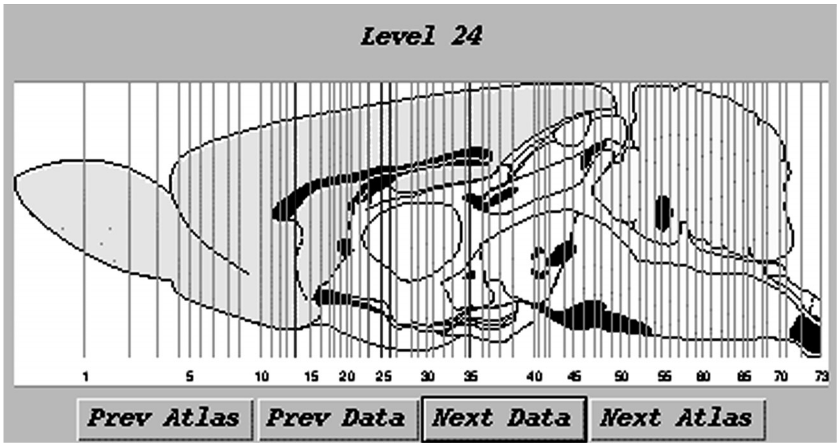

Figure 12 A screenshot of the Level Manager.

buttons in the Display Manager to change atlas levels. At the top of the Level Manager is a display indicating the number of the currently displayed atlas level.

INQUIRY MANAGER

Currently, the sole function of the Inquiry Manager is to display information returned as a result of a "structural query" in the Display Manager Fig. 13). The information returned is selectable text and may be pasted into the Query Manager if so desired in support of query specification.

VIEWER MANAGER

The purpose of the Viewer Manager (Fig. 14) is to control certain functions of the Display Manager, such as the number of Display Manager windows visible, the image magnification, the presence or absence of the physical-coordinates grid, and the chosen atlas. All of these choices may be specified by selecting relevant items from the lists shown, and the desired change will be brought about by clicking on the "Apply" button.

Figure 13 A screenshot of the Inquiry Manager.

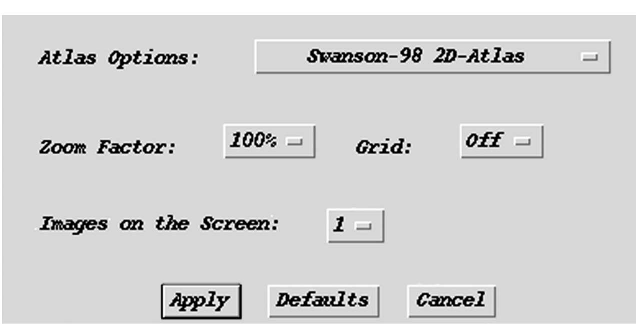

Figure 14 A screenshot of the Viewer Manager.

4.3.3 Discussion

This chapter has presented an introduction to a neuroanatomical informatics tool built on the NeuroCore architecture developed elsewhere within the USCBP. This tool provides a way of performing essential everyday visualization tasks within an anatomical laboratory without concomitant informational problems that we have found obstructive in the past. NeuARt is still a fragile prototype, requiring additional development in order to make it robust enough for general use, but it remains a coherent practical design and implementation that fulfills its original objectives well.

References

Dashti, A., Ghandeharizadeh, S., Stone, J., Swanson, L. W., and Thompson, R. (1997). Database challenges and solutions in neuroscientific applications. *Neuroimage* **5(2)**, 97–115.

Gerfen, C. R., and Sawchenko, P. E. (1984). An anterograde neuroanatomical tracing method that shows detailed morphology of neurons, their axons and terminals—immunohistochemical localization of an axonally transported plant lectin. *Phaseolus vulgaris* leukoagglutinin (PHA-L). *Brain Res.* **290**, 219–238.

Harris, K. (1999). Structure, development, and plasticity of dendritic spines. *Curr. Opin. Neurobiol.* **9**, 343–348.

Risold, P., Canteras, N., and Swanson, L. W. (1994). Organization of projections from the anterior hypothalamic nucleus: a *Phaseolus vulgaris*-leucoagglutinin study in the rat. *J. Comp. Neurol.* **348(1)**, 1–40.

Risold, P., and Swanson, L. W. (1997). Connections of the rat lateral septal complex. *Brain Res. Brain Res. Rev.* **24(2–3)**, 115–95.

Roland, P., and Zilles, K. (1994). Brain atlases—a new research tool. *Trends Neurosci.* **17**, 458–467.

Swanson, L. W. (1998). *Brain Maps: Structure of the Rat Brain*. Elsevier Science, Amsterdam.

Swanson, L. W. (1987). The hypothalamus. In *Handbook of Chemical Neuroanatomy.* Vol. 5. *Integrated Systems of the CNS* (A. Bjorkland, T. Hokfelt, and Swanson, L. W., Eds.). Elsevier Science, Amsterdam, pp. 1–124.

Swanson, L. W., and Cowan, W. M. (1975). Hippocampo-hypothalamic connections: origin in subicular cortex, not ammon's horn. *Science* **189(4199)**, 303–4.

Zilles, K. (1985). *The Cortex of the Rat: A Stereotaxic Atlas*. Springer-Verlag, Berlin.