

**Animal Models of Diabetic Complications Consortium  
(U01 DK60966-01)**

**Annual Report  
(2004)**

**“Coordinating and Bioinformatics Unit of the AMDCC”  
Medical College of Georgia  
Augusta, Georgia**

**Richard A. McIndoe, Ph.D.  
Program Director**

**Richard A. McIndoe, Ph.D.  
Medical College of Georgia  
Center of Biotechnology and Genomic Medicine  
Augusta, GA 30912  
Phone: (706) 721-3542 Fax: (706) 721-3688  
Email: rmcindoe@mail.mcg.edu**

## **Table of Contents**

	<b><u>Page</u></b>
<b>Part A: Principal Investigator's Summary</b>	
<b>1. Project Accomplishments (2004)</b>	<b>4</b>

**Animal Models of Diabetic Complications Consortium  
(U01 DK60966-01)**

**Part A:**

**Principal Investigator's Summary**

**NOTE: Due to size constraints, most of the figures for this report are located in the Appendix at the end of this document and will be appropriately referenced in the document.**

## 1. Program Accomplishments:

The Coordinating and Bioinformatics unit is responsible for the creation of the informatics infrastructure of the consortium as well as facilitating the efforts of the mouse engineering centers. Our programming paradigm is to develop software systems based on an n-tier architecture, where we create the presentation layer, business logic and data layer into separate software systems. Figure 1 presents the system we have created for the consortium. These systems have been developed to minimize maintenance, but provide a robust scalable model for future growth and interactions at the national level with other organism databases.

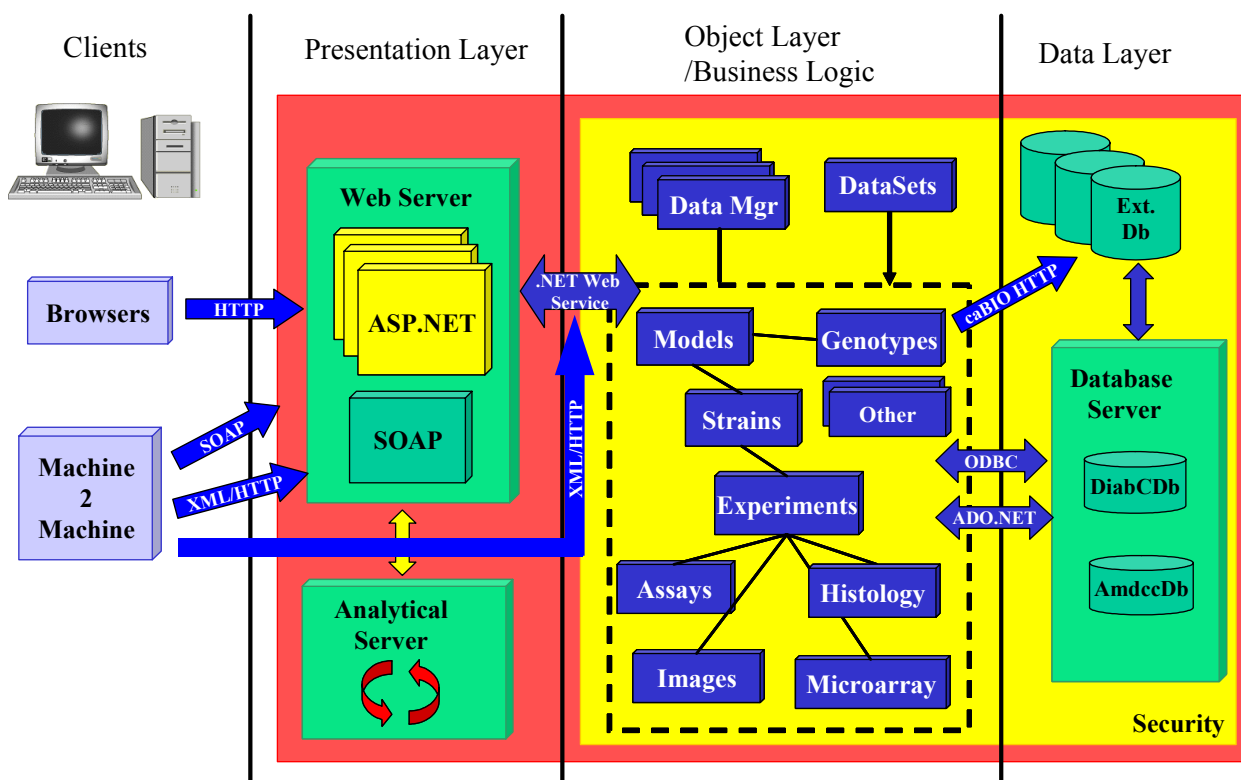


Figure 1. AMDCC n-tier design architecture

## AMDCC Web Portal

During this last year, the AMDCC website has undergone many changes. Some of these changes have been technological advancements for the website as a whole while others have been more functional additions and enhancements. The following sections will describe these changes in more detail with example figures presented when appropriate.

**ASP to ASP.NET** The most significant change we undertook during the last year was to convert the entire website to the .NET framework. The original website application was written using ASP. While this website was quite functional, there were a number of limitations to the ASP application model that made maintenance and future enhancements more cumbersome. The ASP programming paradigm is a scripting model where server side code is written using a scripting language such as javascript or vbscript. These languages, while functional, have limitations with respect to code execution, reuse and development. The .NET framework is a high performance scalable programming paradigm that provides technologies to easily create web applications (ASP.NET) and Web Services (HTTP/SOAP). Unlike classic ASP, which supports only interpreted VBScript and JScript, ASP.NET supports more than 25 .NET languages (including built-in support for VB.NET, C#, and JScript.NET). My laboratory exclusively writes our web applications in C#. The most significant advantage of C# and ASP.NET are that these technologies are strongly-typed fully object-oriented languages with full support for inheritance and polymorphism, making code re-use very simple and easy. In fact, even the ASP.NET web pages themselves are compiled classes that can have their own properties and methods.

In order to implement this transition, we had to re-write the entire web application. As a testament to the increased speed in application development, we were able to rewrite the entire web application (including the API, see below) in six months. The original ASP web portal took almost 10-12 months to create. Additionally, while we were re-writing the web portal, we redesigned the page structures to take advantage of the object oriented nature of ASP.NET and added a number of user interface enhancements and analytical tools for consortium members. Although we are constantly working on the web portal, this initial transition effort required writing 144 ASP.NET pages (26032 lines of code), which replaced the 199 ASP pages from the original web application (an approximate 30% reduction).

**API changes: COM to .NET DLL** As stated above, during this last year we converted the entire AMDCC web application to the .NET framework. This transition also included converting the original COM API to a .NET DLL. The original COM DLL we created for the AMDCC could have been used by the ASP.NET application, but would have required the use of a wrapper class to interpret the COM calls and translate them to interpretable .NET calls (Interop classes). Because of this and the fact that the VB6 language used to write COM is not an object oriented language, we made the decision to completely rewrite the API. During this re-write we decided to revisit the class structure of the API and took advantage of the object oriented nature of C# to better reflect the phenotype data collected by the consortium. The new AMDCC API was completed during the same time frame as the ASP.NET web application. This was necessary since the web application uses the API. To date, the AMDCC API has 185 object classes that fully describe the activities of the consortium (37,964 lines of code).

Because we have implemented the API as a .NET DLL we can easily expose specific methods and properties of these classes to a dedicated AMDCC web service, providing a mechanism for machine to machine transfer of AMDCC data. We have started to implement a web service that uses the various manager classes to extract data from the system via SOAP or HTTP calls. A brief example of the current state of the AMDCC Web Service is provided later in this document.

## Web Portal Enhancements

Over the last year we have added a number of enhancements and improvements to the website with respect to the user interface, analytical methods and phenotype exploration. These enhancements were only possible due to the fact that we converted the site and API to the .NET framework.

**Use of multidimensional grids.** All of the phenotype data uploaded to the system is in the context of an experiment that contains both assays tested and experimental factors (independent variables). The traditional approach to viewing this data is to create a 2 dimensional (row by column) representation of the data where each row of data comes from one animal with the columns being the independent variables and assays measured for that experiment. Flattening multidimensional data into a 2 dimensional table poses a number of problems for the user. If there are multiple independent variables, the number of rows in the table dramatically increases to accommodate each unique combination of these variables. For example, if an experiment took blood glucose measurements over 4 time intervals for two different doses of STZ, the resulting 2D table would require 8 rows per animal tested. If one tested 20 animals each for 4 strains, the resulting table would have 640 rows. While this may not seem large, the table generated for the web page would require the end user to repeatedly scroll to access and view the data. A better approach would be to provide the end user with a way to conveniently create multidimensional grids of the data. In the above example, the user might want to group first by strain then by STZ dose and sort by animal ID and then by time. This organization would provide the end user with a quick way to move around the data.

During this last year, we implement these multidimensional grids throughout the website where the user would benefit from a hierarchical display of the data. Figure 2 (see Appendix) is a screen capture from the experiment browser interface. The user has grouped first by strain then by sex and sorted the data by animal ID then age. The user interface to generate these hierarchical grids is very intuitive and simple. To group by a specific data type, the user simply drags and drops the appropriate column header to the area above the grid. One can drag multiple column headers in any order. With this interface the user can not only group the data, but also sort and move the column order as well. This interface provides a very flexible way for investigators to organize the data for their particular interest. Finally, all of the hierarchical grids generated by the investigator can be downloaded to an Excel spreadsheet while maintaining the multidimensional structure.

**AMDCC Datasets.** Last year we were working on developing a data structure that allowed investigators to query the entire AMDCC database in a flexible and powerful way. We had completed the data model, but needed to finish the object model and user interface. This year, we completed the entire AMDCC Dataset concept. The data and object models used to represent a dataset have been completed as well as the user interface necessary for the members and public to query the system. AMDCC investigators can save these dataset definitions and data to the database. They can then be used by consortium members, their laboratory and the general public. Any dataset can be saved and optionally marked for public use.

A few key features of the Dataset concept should be mentioned. First, all datasets are essentially a snapshot of the data at the time of the query. This includes datasets that have been saved. When an investigator saves a dataset, a copy of the data returned for that query is also

saved. This guarantees that the data will always be the same while an investigator works with a particular saved dataset, even if more data is added to the system by another member. Obviously, an investigator may want the latest data for a given dataset definition. Therefore, we also added an optional flag that can be set by the investigator to force a refresh of the dataset to the latest data in the system. Second, investigators can build complex queries based on strains (male/female/both), investigators, experiments, assays or experimental conditions. For assays and conditions, one can define ranges of values or categories. Additionally, one can create groups of query parameters that can be mixed together to form even more complex requests. Third, one does not have to save a dataset to use it for phenotype exploration. Because the public can not save datasets, we had to create a mechanism to maintain adhoc queries during a user's session. This is principally accomplished by the use of serialization of our object model. Users have the ability to create an adhoc query and then use the result set of that query to create charts or perform statistical analysis in other parts of the website. Finally, all datasets (adhoc or saved) can be downloaded by the users as Excel spreadsheets. If a user has analytical needs that are not met by the web site, they can always download the data and analyze it locally using their own software. Figure 3 (see Appendix) illustrates an example query using the Dataset interface of the AMDCC web portal.

**Microarray data uploads.** During this last year, we completed the object model and user interfaces necessary to allow consortium members to upload their microarray data to the system. The AMDC Microarray object model is MIAME compliant. In order to maintain this compliance, the investigators supply specific information during their upload process. For example, the labeling strategies, sample information, data collection information and hardware/platform information are all required information. Because this amount of information can be overwhelming when entering one array at a time, investigators have a choice to either upload one array at a time or use a bulk upload strategy to send many arrays at once. For bulk uploads, two files are created by the user. One is essentially a file-of-filenames where the experimental conditions and file names for each hybridization are given. The other is a ZIP file containing all the files for the microarrays. Adding microarray data to any database is always a cumbersome process, but we have tried to make it as simple as possible for the AMDCC membership.

**Data Downloads to Microsoft Excel Spreadsheets** All of the data generated by the consortium can now be downloaded into Excel spreadsheets. This is true for both the public and members. However, the public will only be able to download data that has been released. We have added this functionality to both AMDCC Datasets as well as the individual experiments. In order to download data, all one has to do is either create a dataset or browse an experiment and an "Export to Excel" button will appear once data is returned. Figures 2 and 3 (see Appendix) are screen shots from the Experiment Browser and Dataset query (respectfully) with the Export to Excel button visible.

**Analytical Tools.** One of the major goals of this last year was to add more analytical tools to the web portal in order to facilitate both data analysis and data visualization. We wanted to provide not only the tools, but to provide automated analysis of the data where appropriate. One of the common features to all of the interfaces for these tools is that one can filter the data to be analyzed or viewed. Each of these interfaces uses common options to filter based on assay,

strain, sex, age and any experimental factor values available for that data. The options are tailored to the kind of tool being used. As an example, Figure 4 presents a screen capture of the filter options for one experiment in the Basic Statistics user interface.

The following sections describe the list of tools we created during the last year and, when appropriate, a description of any analysis that is done automatically.

**Basic Statistics** This interface provides the users and membership with an easy and flexible way to calculate basic statistical information for Adhoc and saved datasets or individual experiments. Users select the assays and strains they want to analyze as well as any filter criteria (Figure 4). The system will automatically calculate the mean, standard deviation, median, min, max, variance, skewness and kurtosis for each assay/strain combination. If the query contains both male and female data, the system will automatically do these calculations for the combined sexes as well as males and females separately. Figure 6 (see Appendix) is a screen shot from an analysis of two assays for five strains with both male and female data from one experiment. This interface provides a convenient way for investigators to analyze many assays for many strains quickly.

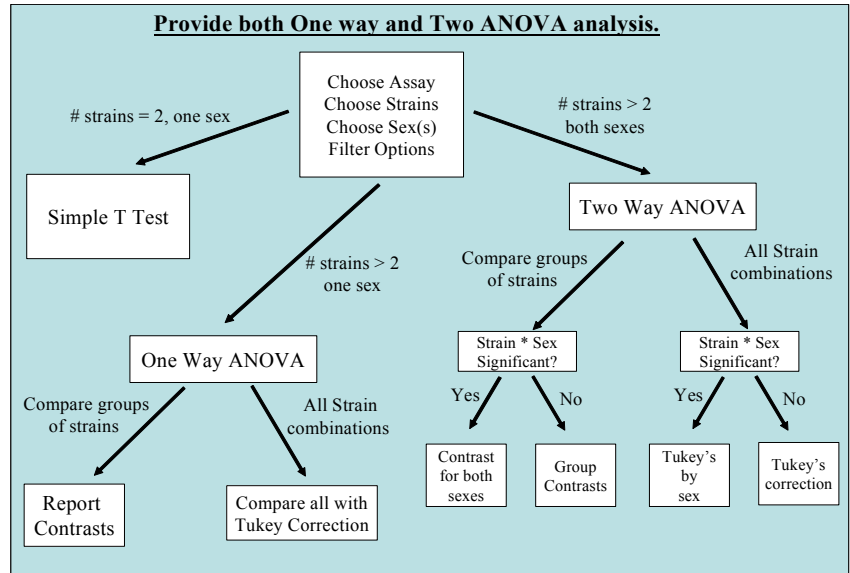
**Figure 4.** Example filter options from Basic Statistics user interface.

**T Tests** This interface provides a simple way for users to perform an unpaired T Test to assess whether two groups of animals are significantly different from each other. Similar to the Basic Statistics interface, users select the assay and then strain and filter criteria for each group to be tested. Based on these criteria, a Student’s T value, p value, upper/lower confidence limits and group means, standard deviations and number are all calculated.

**ANOVA Analysis.** Analysis of variance (ANOVA) is the multigroup generalization of the t test. Like the t test, ANOVA assumes that samples are randomly drawn from normally distributed populations with the same standard deviations. If differences between the observed means of the samples are larger than one would expect from the underlying population variability, estimated by the standard deviations within the samples, you can conclude that at least one of the samples has a different mean than the others. The AMDCC ANOVA analysis interface supports automated One Way and Two Way ANOVA analyses for both balanced and



unbalanced experimental designs. Like the other tools, the user can filter the data to analyze based on strain, sex, age and available experimental factor values. Once selected the user is presented with the list of assays to analyze and two additional options. These options allow the user to decide what strain comparisons they want to make once the ANOVA is completed. The first option lets the user contrast two groups of non-overlapping strains to determine if they are significantly different from each other and the second option tells the algorithm to do all pairwise comparisons of individual strains calculating p values for each. Once these options are selected, the system automatically prepares the data and performs the appropriate ANOVA analysis (Figure 5).



**Figure 5.** Workflow decision tree for automated ANOVA analysis. Based on the data being analyzed the system automatically chooses the appropriate ANOVA analysis.

There are two key points to these analyses. First, the Two Way ANOVA currently only looks for Strain by Sex interactions. If there is a significant interaction between Strain and Sex, the subsequent analysis (contrast or pairwise) is conducted separately for each sex. If the interaction is not significant, then the subsequent analysis is done with males and females grouped together. Second, we use a Tukey correction when calculating the p values for all pairwise comparisons if there is more than two strains being compared. This correction will account for the multiple tests being done.

The algorithm automatically determines whether a One Way or Two Way ANOVA analysis is appropriate based on the data being analyzed. For example, if after preprocessing the data for analysis (eg. removing missing data, removing data with only one sex) there is only one strain left that has both male and female data, a One Way ANOVA will be done on males and females separately. Figure 7 (see Appendix) shows a screen capture of the sample output from a Two Way ANOVA analysis looking at Albumin/Creatinin ratios between five different strains.

**Correlations** This interface provides a way for users to measure the relationship between two or more assays. Again, users can filter the data based on strain, sex, age and experimental factor values. The user selects their filter criteria and the two assays they want to assess. The algorithm calculates the Pearson correlation coefficient, R squared and covariance as well as provides a plot of the data with the regression line. Figure 8 (see Appendix) shows the generated data plot and correlation data looking at the relationship of blood glucose and rodent weight.

**Charting Data** During this last year we have significantly increased the visualization tools available to the membership and public. We have concentrated on augmenting the charting capabilities of the web portal. Specifically, we have developed a number of user interfaces to allow users to create charts on the fly based on any filter criteria they want. Like the analytical tools described above, we allow the users to filter the data to be plotted based on strain, sex, age and experimental factor values. We currently have three charting options for consortium members. All of the charts listed below generate error bars when appropriate and provide editable titles and axis labels. Additionally, we use tooltips extensively so that details of each data point can be seen simply by hovering the mouse pointer over the data point.

**Column Charts** The column charts provide the users with a convenient way to visualize the differences between strains. The users select the assay and filter criteria to use in the plot and click a button to generate the chart. If the data used to generate the chart contains both male and female data, one can preview the details of the male and female distribution by placing the mouse pointer over the data column. Figure 9A shows the chart of 5 strains for blood glucose. A preview of the male and female plot is also shown. One can drill down into the data by clicking on the column. Figure 9B shows the detail male/female chart after the user clicks on the column shown in Figure 9A.

**Line Charts** The line charts provide a convenient way for users to visualize strains over a range of values. These values can be age or any other experimental factor (eg. time since diabetes or time since glucose injection). However, categorical experimental factors are excluded from the available options (eg. mouse diet). Each strain is plotted as an individual series with a unique marker and line color. The user simply selects the strains and filter criteria to use and generates the chart. Figure 9C is a line chart of body weight versus time since Type 1 Diabetes onset for five strains.

**XY Charts** The XY chart interface provides a convenient way for users to visualize the individual data points and their distribution. Data is plotted for an assay and any experimental factor value, including categorical data types. The mean for each cluster is plotted as a horizontal line across the cluster. This chart provides a nice way for users to compare groups of animals as well as look for outliers in defined categories, times or ages.

**Chart Customization** Currently, all of the charts that are generated can have their axis labels and chart titles customized by the end user. Figure 10 (see Appendix) illustrates two charts generated using two different filter criteria, one for animals on the STZ dosage plan and the other for control animals. In this figure, the user has added chart titles to reflect these differences and edited the axes labels. Future versions of these interfaces will include more options such as color schemes, background colors and different borders. However, because the amount of screen space available is limited, we are trying to design a simple server control that can be activated by a right mouse click over the chart itself. The challenge in this feature is to make it cross browser and OS compatible.

## Data Schema and Technological Advances

Based on a number of conversations with the membership, we made some significant changes to the data schema in order to better reflect the data capture by the consortium. In addition, we have worked on some software components that provide increased flexibility and data sharing capabilities.

**Share Laboratory Animals** During the last year, the phenotyping cores for neuropathy, uropathy and retinopathy were established and began to accept animals from other laboratories in the consortium. In addition, many of the mouse engineering groups began to send animals to be cross assayed for other diabetic complications. These activities uncovered a design issue in the original data schema. We had not anticipated that animals would be shared between laboratories. The original data schema provided for animals within a laboratory to be used in any number of experiments. But once another investigator used that animal in their experiments, there was no guarantee the data collected for that animal would be maintained across the consortium. Meaning that the data schema did not reflect the fact that these were the same animal and not two separate animals in two separate laboratories that happen to have the same external animal ID.

In order to fix this design issue, we had to completely redesign the concept of animals in the data schema as well as the data upload interface. This included table additions and updates, store procedure modifications and modifications to the web portal. We completed these additions during this last year. The system now provides a mechanism for investigators to share animals across the consortium. The data collected for one animal can now be tracked across the consortium. This means that if an animal is assayed for diabetic nephropathy in one lab and the tissues are sent to the neuropathy and retinopathy phenotyping cores, the data collected for this animal will be correctly linked and maintained in the data schema. In order to use an animal from another laboratory, all the member has to do is correctly identify it in the data upload templates as coming from another investigator. This is accomplished by appending the investigator's user id (email address) in brackets to the external animal ID. For example, to add data to the system for the animal ENU1234 from Matt Breyer's laboratory, the Animal ID column in the Data tab of the template would have ENU1234[mathew.breyer@vanderbilt.edu].

In addition to the ability to share animals, we also created the user interface to query the system for any animal being used in the consortium. This interface will provide all of the information about that animal (e.g. sex, DOB, strain) including what experiments it has been used in.

**Genotype Definitions Improved** The initial data schema and object model used to define the genotypes of the various strains used by the consortium had limited functionality. For example, we did not take into account the Genomic Segment mice. We have fixed this model by letting the genotypes be defined as a collection of Gene Modifications, Transgenes and Genomic Segments. Each of these entities has a corresponding data model and object model equivalent. We can now define the genotypes of each mouse strain with better granularity. For example, transgenes can now have one or more regulatory elements associated with the transgene (e.g. promoter, enhancer) and genomic segments can be described by the proximal and distal loci that define the genomic organization of the intervals. These modifications include the ability to define a Cre-Lox system.

We have also redesigned the Show Strain page to reflect these changes. Because these genotypes can be complicated, we decided to create a tabbed interface where each of these objects has its own tab where the information is displayed. Users can then easily switch between the different kinds of genetic manipulations for a particular strain.

**caBIONet** The National Cancer Institute's Center for Bioinformatics (NCICB) has developed a java based data management and information system called caCORE. One component of this software suite is the object oriented API (caBIO) used to access the rich biological datasets collected at the NCI. This API can access the data using native java classes, SOAP requests or HTTP calls. Non-java based clients wanting to use this API have to use the SOAP or HTTP interfaces with the data being returned from the NCI servers as an XML data stream. While the XML can be read and manipulated using DOM or SAX parsers, one loses the convenience and usability of an object oriented programming paradigm.

The caBIO software suite and databases can be downloaded and installed on a local server. However, this will require local curation, maintenance and update of these data warehouses. This can be quite time consuming and complicated. Investigators wanting to take advantage of the NCI database infrastructure can access the genomic information in caBIO using NCI's caBIO SOAP/HTTP API

The AMDCC web portal uses genomic information for the description of the genes that make the genotypes of the animal models developed by the consortium. Rather than maintain external databases locally, we decided to use caBIO to retrieve genomic and biological data for the genes under investigation. However, because caBIO is Java based, we needed to create the software components that would translate the data retrieved from NCI to local .NET objects. Therefore, we created caBIONet, a set of .NET wrapper classes (managers, genes, chromosomes, sequences, etc.) capable of serializing the XML data stream into local .NET objects. The software is able to search NCICB databases and provide local objects representing the data that can be manipulated and used by other .NET programs. The software was written in C# and compiled as a .NET DLL. The current version of caBIONet has 29 classes with 313 properties and methods.

**AMDCC Web Services** One of the goals of the AMDCC is to share the data generated by the consortium with the scientific community. This would include both individual investigators as well as national data repositories such as the NCBI. Currently, individual investigators can download all the data generated by the AMDCC via the web portal. However, this strategy is not optimal for data repositories or other web sites that would like to retrieve our data. A better approach for these entities is to provide a web service that can be used for machine-to-machine transfer of data.

The term Web services describes a standardized way of integrating Web-based applications using XML, SOAP, WSDL and UDDI open standards over an Internet protocol backbone. XML provides structure to the data, SOAP is used to transfer the data, WSDL is used for describing the services available and UDDI is used for listing what services are available. Web services allow different applications from different sources to communicate with each other without time-consuming custom coding, and because all communication is in XML, Web services are not tied to any one operating system or programming language. For example, Java can talk with Perl, Windows applications can talk with UNIX applications.

One of the benefits of converting the API to the .NET framework is one can easily expose specific methods and properties of the API as a web service. During this last year, we have begun the process of creating the AMDCC web service. The primary transport is SOAP; however one can also use an HTTP POST passing specific parameters in the URL. We will provide access to all of the data manager classes used in the API to retrieve data as well as pertinent SearchCriteria classes. We will publish the WSDL and documentation of how to use the web service to retrieve data.

## **2. Address previous EAC comments:**

- *We look forward to progress toward the outlined objectives for the next year including better charting options as well as background statistical analysis.*

We have made significant progress towards the stated goals. The automated ANOVA analysis is complete as well as the improved charting options. We have also added the ability to analyze and visualize adhoc queries, which was not in place at that time of the last meeting.

- *Additional training for AMDCC sites is encouraged if only to ensure more data is uploaded.*

The current plan was to do another round of training sessions once we had completed the statistical and charting interfaces. We are not prepared to revisit the centers and continue our education efforts.

- *More sophisticated web tracking tools would be a benefit. Not only how many times the site is accessed, but by whom and to what level.*

The current tracking tool we use can tell us much about how the users interact with the website. For example, we know how often each hyperlink and button on a page is clicked and by whom (users vs members). We can also track the average amount of time spent on a page as well as what page they came from and which page they went to. However, this particular tool does not give us browser and location information. We will look into other products that focus in on these aspects of web site tracking.

- *You are encouraged to take the lead in organizing an AMDCC subgroup focused on mining the existing data within the website.*

We are now prepared to take on that role. I have added Dr. Robert Podolsky, a biostatistician, as co-Investigator on the grant specifically to work on the meta analysis and data mining efforts. He was instrumental in helping us design the automated ANOVA analysis to provide the correct analysis given the data. Over the next 6 months, I will begin to organize this effort with the PIs.

- *Please ensure that all “public” data is easily visible and accessible on the front page of the website.*

All of the statistics and metrics published on the home page are based on publicly released data. If members log in, they will notice that the availability and numbers adjust to reflect that all data is being considered. From the home page, all users can directly link to any of the data objects in the web portal.

- *Continued efforts must be made to facilitate the addition of the AMDCC link to other websites.*

We agree the AMDCC link needs to be added to other websites. We will continue this effort. In addition, we have been submitting abstracts/posters to national and international meetings in order to promote the website as well as speaking to investigators at those meetings.

- *Please continue to send a “scorecard” of uploaded data to both the Steering Committee and the External Advisory Committee every 3 months. These updates should include the previous and current numbers of experiments, models, strains, etc. for each individual laboratory.*

I apologize for this last round as I was under the impression we were going to coincide the Site Activity Reports with the Steering Committee meeting. From now on, we will supply the Site Activity Reports every three months. The current reports only show the information that has been uploaded. We will add all investigators to that report whether or not there has been any activity since the last query. We will also add the previous numbers to the activity report.

- *How will the microarray data be validated? RTPCR? Will everyone use the same chip?*

Currently the microarray data is only uploaded into the system. We only validate the data content and file structures, not whether or not a particular gene is correctly identified post analysis. We are only a repository for the raw data. The individual investigators are responsible for analyzing, interpreting, validating and using their results to generate hypotheses. The public will also be able to download and analyze the microarray data using their favorite analytical package.

As for the chip platforms, we provided a microarray questionnaire asking details about the platforms, equipment and software each laboratory used for their microarray experiments. Of those who answered the questionnaire, the majority of the laboratories who do microarray experiments use the Affymetrix chips. There was one lab that also used custom spotted arrays and two color experiments, but they have since told me they are moving more towards the Affymetrix arrays.

## APPENDIX: FIGURES

**Export to Excel**

Strain ▲  
Sex ▲

Strain : 129S6/SvEv (2)  
Sex : F (84)  
Sex : M (84)

Animal ID ▲	Age ▲	IP STZ dosage plan	time since Type 1 Dia	W rodent (g)	blood glucose (mg/dl)	ualb/uc
344	8	stz1	-1	23.71		
344	11	stz1	2	26.84	168	
344	17	stz1	8	27.86	172	
344	21	stz1	12	28.7	157	
344	24	stz1	15			26.2303
344	25	stz1	16	30.24	191	
345	8	stz1	-1	21.98		
345	11	stz1	2	25.83	177	
345	17	stz1	8	27.16	187	
345	21	stz1	12	27.92	278	
345	24	stz1	15			33.8129
345	25	stz1	16	29.47	306	
346	8	stz1	-1	21.11		
346	11	stz1	2	24.15	223	
346	17	stz1	8	25.07	236	
346	21	stz1	12	25.45	370	
346	24	stz1	15			68.7599
346	25	stz1	16	28.18	317	

**Figure 2.** Example hierarchical grid from the experiment browser interface. The grid can be grouped by any column by dragging the columns to the top of the grid. Columns can be sorted by simply clicking the column header. In this example, the grid is grouped first by strain then by sex with the Animal ID and Age columns being sequentially sorted.

**Animal Models of Diabetic Complications Consortium** Logged in as: **Richard McIndoe**  
Edit profile

• MAIN • DATA SEARCH • DATA ANALYSIS • DATA SUBMISSION • ABOUT AMDCC • CONTACT • MEMBER AREA • LOGOUT

---

### Datasets

**Welcome to the Datasets page!** We are currently testing this feature and welcome your input.

Here you can query phenotype data by strain, investigator, experiment, experimental condition, and assay. You can filter by sex for selected strains. For selected assays and conditions, you can specify bounded or unbounded ranges of values. AMDCC investigators and their lab members can save criteria for later review.

Click **Add Group** to start a new criteria group (indicated by =====OR===== in the display). When you specify more than one group of criteria, you are searching for data that matches all the criteria in group 1 OR all the criteria in group 2, and so on for each group. Within a group, criteria from the same category are ORed together (for example, including Strains 129.A and C57BL6 in the same group will find data for both strains).

**NOTE: The selection lists for strains, assays, and experiments show those items that have associated data, some of which may not be publicly available. Public users may not get any results even with minimally restrictive criteria.**

Select a dataset from your lab [No selection]

or select a dataset from another lab [No selection]

Criterion Include Strain 129.SVJ;B6;FVB(STOCK) Tg(alpha-MHC-Cre)-GLUT4<lox/lox> Sex M+F

Include Assay blood glucose (>200)  
Include Condition IP STZ dosage plan=stz1

Animal ages expressed in week(s) post-natal

**Options**  
 Public users can view this dataset definition (although they are still restricted to released data)

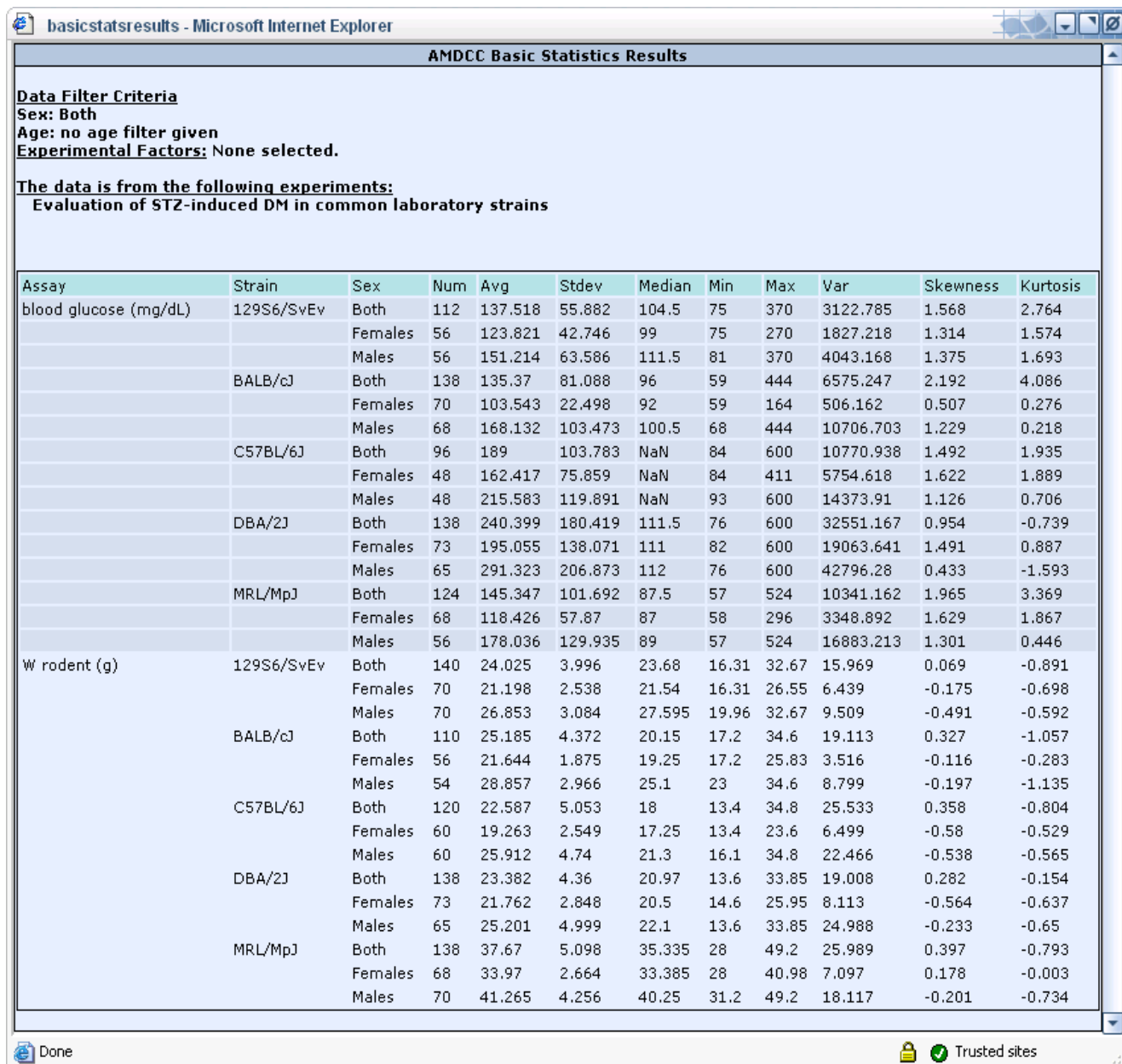
Save criteria as

*Note: Using an existing dataset name will overwrite that dataset.*

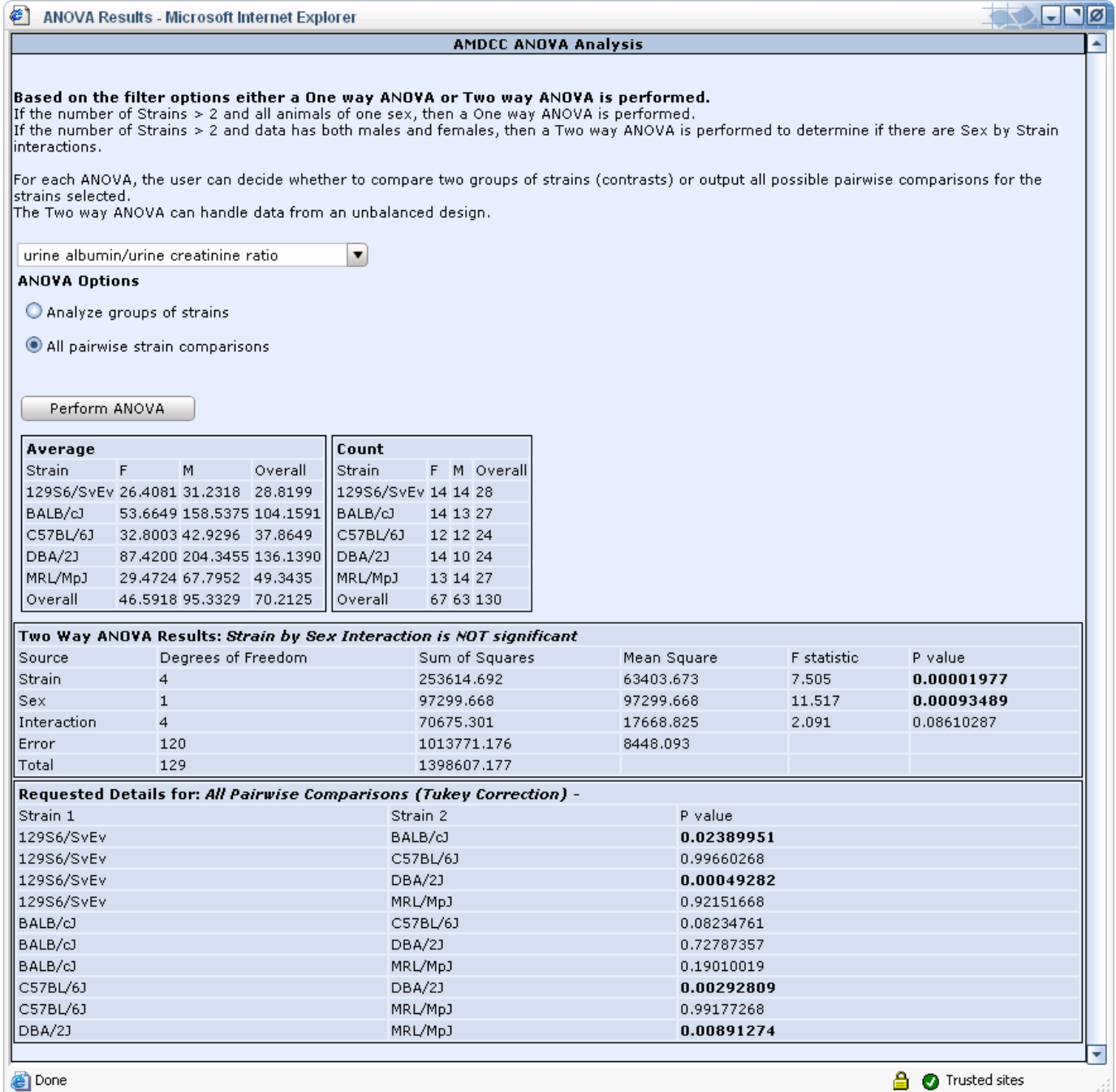
Investigator	Experiment	Animal ID	Strain	Sex	Age	IP STZ dosage plan	time since Type 1 Diabetes onset (wk)	blood glucose (mg/dL)	MeasID
Coffman	Evaluation of STZ-induced DM in common laboratory strains	100	BALB/cj	M	20	stz1	12	205	16972
Coffman	Evaluation of STZ-induced DM in common laboratory strains	100	BALB/cj	M	24	stz1	16	201	16974
Coffman	Evaluation of STZ-induced DM in common laboratory strains	101	BALB/cj	M	20	stz1	12	290	16987
Coffman	Evaluation of STZ-induced DM in common laboratory strains	101	BALB/cj	M	24	stz1	16	390	16989
Coffman	Evaluation of STZ-induced DM in common laboratory strains	101	BALB/cj	M	27	stz1	19	373	16991
Coffman	Evaluation of STZ-induced DM in common laboratory strains	101	BALB/cj	M	16	stz1	8	321	16986
Coffman	Evaluation of STZ-induced DM in common laboratory strains	102	BALB/cj	M	16	stz1	8	290	17001
Coffman	Evaluation of STZ-induced DM in common laboratory strains	105	BALB/cj	M	16	stz1	8	270	17016
Coffman	Evaluation of STZ-induced DM in common laboratory strains	106	BALB/cj	M	20	stz1	12	368	17032
Coffman	Evaluation of STZ-induced DM in common laboratory strains	106	BALB/cj	M	24	stz1	16	383	17034
Coffman	Evaluation of STZ-induced DM in common laboratory strains	106	BALB/cj	M	27	stz1	19	444	17036
Coffman	Evaluation of STZ-induced DM in common laboratory strains	106	BALB/cj	M	16	stz1	8	317	17031
Coffman	Evaluation of STZ-induced DM in common laboratory strains	107	BALB/cj	M	20	stz1	12	345	17047
Coffman	Evaluation of STZ-induced DM in common laboratory strains	107	BALB/cj	M	24	stz1	16	416	17049
Coffman	Evaluation of STZ-induced DM in common laboratory strains	107	BALB/cj	M	27	stz1	19	411	17051
Coffman	Evaluation of STZ-induced DM in common laboratory strains	107	BALB/cj	M	10	stz1	2	243	17044

**Figure 3.** Example of a Dataset query and result.. This dataset looks for all data where blood glucose > 200 mg/dL and the IP STZ dosage plan is STZ1. Adhoc datasets can be used in subsequent statistical analysis or charting activities.

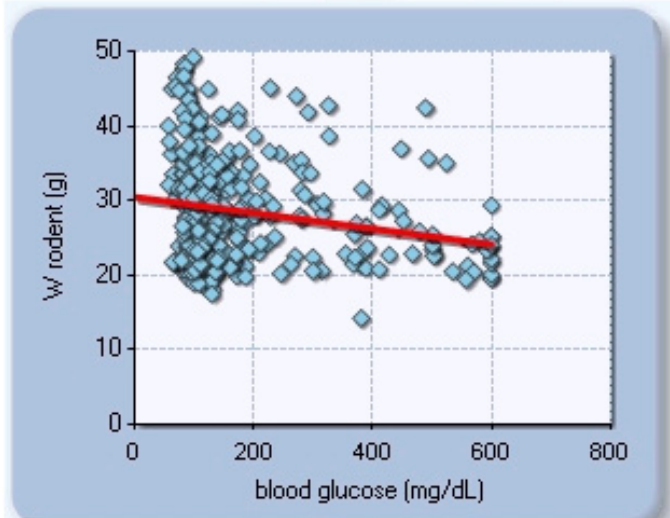




**Figure 6.** Example of basic statistics automatically calculated for two assays and five strains. If the data contains both male and female data, the algorithm automatically analyzes grouped as well as individual sex data.



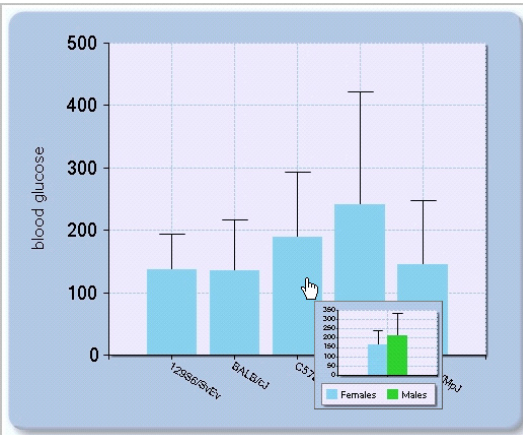
**Figure 7.** Example Two way ANOVA results. The algorithm automatically prepares the data for analysis and correctly decides the appropriate analysis. In this example, all the data analyzed has both male and female data. Because there are not significant Strain\*Sex interactions, we can group the sexes together when calculating all pairwise comparisons. The P value calculation use a Tukey correction for multiple tests.



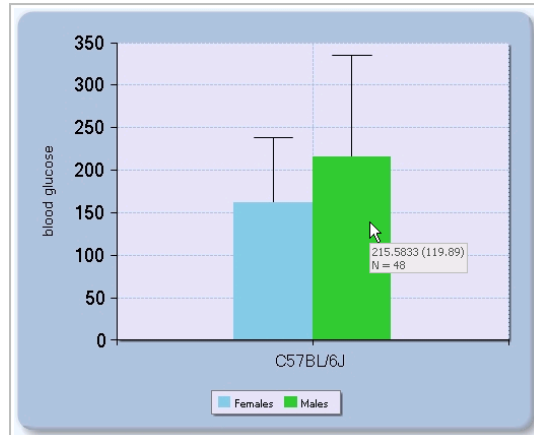
Correlation:	-0.2
R Squared:	0.04
Covariance:	-191.1
Total Points Assay 1:	366
Total Points Assay 2:	366

**Figure 8.** Example output from the correlation analysis interface. In this example, one wants to determine if there is a correlation between blood glucose and body weight.

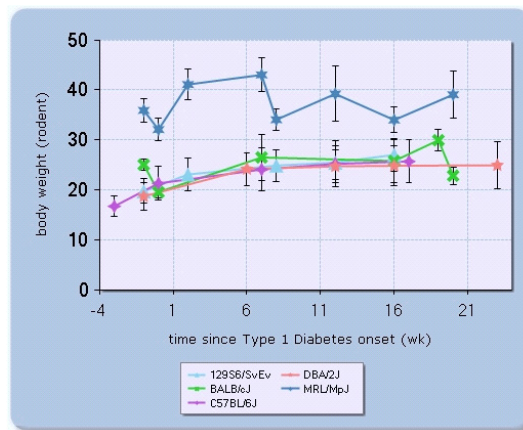
A



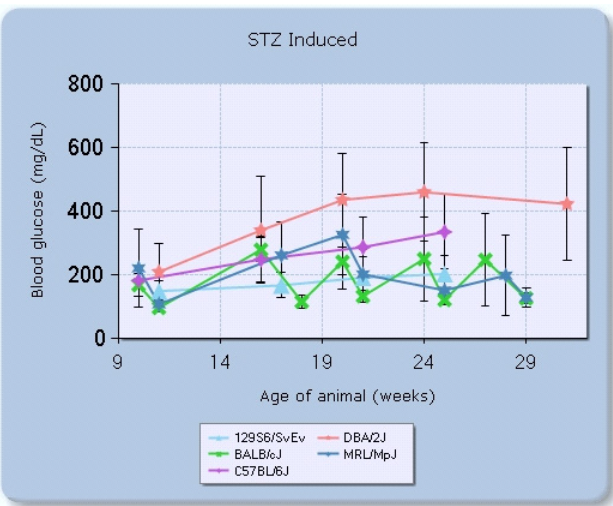
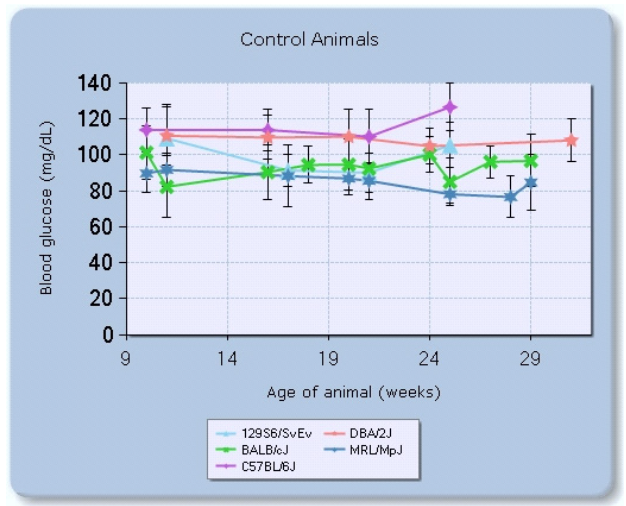
B



C



**Figure 9.** Example charting options available. **A)** Column charts for comparing strains. If the data contains male and female data, a preview is shown during mouse over. **B)** Detail male/female data from previous. Tooltip shows mean, SD and N. **C)** Line chart where each line is a different strain.

**A****B**

**Figure 10.** Two line charts filtered based on STZ treatment comparing strains over time for blood glucose concentrations.. Titles for each chart were dynamically created. All axis labels can be customized as well.