**<People page on RD> Intro**

#(Alex) Intro# Hi, my name is Alexander Stuy, this is Nodin Weddington. We have been asked today to talk about a project we have worked on for quite a few years now, with much effort and sweat, and in the beginning a fair amount of trepidation. (Trepidation is gone now, but the effort and sweat continues) The project is ReplicationDomain and is an online database resource for storing, sharing and visualizing DNA replication timing and transcription data, as well as other numerical epigenetic data types. Dr. Gilbert wanted to make sure I mentioned that ReplicationDomain can handle any numerical data that maps back to a set of base pairs on a chromosome.

<Main page on RD>

**#(Alex) Background on ReplicationDomain**

Dr. Gilbert came to the department around 2006. As part of being hired he wanted to have a website created.

This was his idea of a website that could visualize and analyze the genomic data he and his lab collected.

It would also provide access to his data to other scientists and the public. Public access to data is a requirement for being published by some publications. The department agreed as part of Dr. Gilbert's hiring to provide hours to his project for the first 2 years, after that he would provide funding for the hours.

**#(Alex) Gathering of system requirements for ReplicationDomain**

One of the first steps to creating software is to gather the system requirements. Nodin and I met with Dr. Gilbert and his postdocs and we setup regular biweekly meetings We discussed what they envisioned the site would look like and what it's capabilities should be, both in short term and long term. Of course as Dr. Gilberts group and others used the site the capabilities requested have changed and expanded over time. We looked at similar sites all ready in existence. And we looked at sample datasets. Then we started coding.

Current capabilities of ReplicationDomain

- Storage of Epiginetic data sets

- Dynamic visualization of multiple epigenitic datasets, with zoom and move of visualization window

- Downloading of whole and partial datasets

- Granular sharing of datasets between users and user groups

**<Show website>**

**#(Alex) Architecture of System, Software used. FreeBSD OS, Apache, PHP, MySQL, PHPMySQLAdmin, GraphicsMagic**

So after we got the system requirements and we were comfortable that we had a good idea of what the system should do we had to come up with an architecture that could accomplish the requirements. We have done other database driven websites, including www.bio.fsu.edu and herbarium.bio.fsu.edu. The main difference here was the on the fly image creation from the data. Our other websites architecture consisted of:

<FreeBSD Webpage>: http://www.freebsd.org FreeBSD® is a Unix based operating system for modern server and desktop computers. FreeBSD provides advanced networking, impressive security features, and world class performance and is used by some of the world's busiest web sites including Yahoo, Apple and Cisco

<Apache Webpage>: http://httpd.apache.org An open-source HTTP server for modern operating systems including UNIX, Windows and MacOS.

<MySQL Webpage>: http://www.mysql.com MySQL is an opensource relational database management system (RDBMS), originally maintained by Sun, now Oracle. Oracle has not killed or maimed it yet. If so it will be forked by the opensource community, like the OpenIndiana fork of OpenSolaris

<PHP Webpage>: http://www.php.net PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML.

<PHPMyAdmin Webpage>: http://www.phpmyadmin.net phpMyAdmin is a free software tool written in PHP intended to handle the administration of MySQL via a web interface.

To do the on the fly images we used

<GraphicsMagic Webpage>: http://www.graphicsmagick.org (API and Libararies)

GraphicsMagick is the swiss army knife of image processing. It provides a robust and efficient collection of tools and libraries which support reading, writing, and manipulating an image in over 88 major formats including important formats like DPX, GIF, JPEG, JPEG-2000, PNG, PDF, PNM, and TIFF.

**#(Alex) Overview/discussion of relational databases.**

As I mentioned ReplicationDomain uses a MySQL database to store the epigenetic data. MySQL is a relational database management system (RDBMS) that uses the Structured Query Language (SQL) to provide access to the data. The first databases were flat databases. In a flat database all data is stored in one table. In 1970 E. F. Codd, a researcher at IBM, invented the relational database.

The relational database organizes data in a series of tables. Each table has its own fields specific to the type of data being stored.

The tables are tied together using relational keys or fields in the tables. This makes the database both smaller and more efficient than a flat table architecture. As with most good inventions in retrospect the idea seems obvious.

**#(Alex) Database Structure of ReplicationDomain. Changes in database structure as data has grown in size**

**#(Alex) Programming of ReplicationDomain backend. Users and groups as related to access permissions. Data input scripts.** **Dataset security levels.**

Groups

- User capabilities on ReplicationDomain website controlled by groups.

- Every function on the website has a corresponging group.

- Groups are implemented via the People, Groups and groupsXuserid tables.

- Allows granualar access to website functions.

Data upload and input scripts

- Check to make sure user is logged in and in group that allows data upload

- Uses PHP file upload library and functions to upload data file to the server

- Does extensive data validation to before inserting data into the database

- Originally all data points were stored in one table. Now the datapoints from each dataset are stored in a separate table.

Dataset security levels:

- Public: Dataset is publicly viewable.

- Private: Only registered users can view the dataset.

- Uber Private: Only you can view the dataset unless specific users are chosen.

Example data file (Replication Timing)

GeneID GeneName CHR Start End Data

1 NA chr1 32108 32168 0.38345398166424

2 NA chr1 36566 36626 0.364192185307867

3 NA chr1 37196 37256 0.361495417337534

4 NA chr1 38109 38169 0.357598361745252

5 NA chr1 39532 39592 0.351550655778936

6 NA chr1 49283 49343 0.310969710092480

7 NA chr1 52308 52368 0.298685934100916

8 NA chr1 55650 55710 0.285281234664487

9 NA chr1 59489 59549 0.270095756722014

10 NA chr1 60884 60944 0.264633874328716

11 NA chr1 72260 72320 0.221203716120895

12 NA chr1 241735 241795 -0.207085578195046

13 NA chr1 357503 357563 -0.291335217941659

14 NA chr1 389471 389531 -0.288986772899941

15 NA chr1 403676 403736 -0.283373882928247

16 NA chr1 443361 443421 -0.246078544630576

17 NA chr1 530358 530418 -0.089271577517763

18 NA chr1 532718 532778 -0.0842186226371571

19 NA chr1 547649 547709 -0.0519017953505649

Example data file (Transcription)

Gene ID Gene Name CHR Start End Data

NM\_009834 Ccrn4l chr3 51312375 51339502 8517

NM\_023502 Elf2 chr3 51343599 51413916 11284

BC022740 4930583H14Rik chr3 51476344 51484438 1927

NM\_025523 Ndufc1 chr3 51493407 51496884 5071

NM\_053089 Narg1 chr3 51504070 51561953 22257

BC065076 Rab33b chr3 51571894 51584141 737

BC050190 Setd7 chr3 51603246 51648752 11948

NM\_174995 Mgst2 chr3 51749121 51770604 105

BC049812 Maml3 chr3 51777363 51944212 1163

NM\_019739 Foxo1 chr3 52356302 52437467 7700

NM\_026225 Cog6 chr3 53070094 53105122 6398

NM\_175386 Lhfp chr3 53129475 53349607 1307