Agile Information Acquisition and Management System for Magnetized Dusty Plasma Experiment Project

Chih-Jye Wang
Dept. of Computer Science
and Software Engineering
Auburn University
Auburn, AL 36849, USA
wangchj@auburn.edu

Edward Thomas, Jr
Dept. of Physics
Auburn University
Auburn, AL 36849, USA
thomaed@auburn.edu

Darrick Artis
Dept. of Physics
Auburn University
Auburn, AL 36849, USA
doa0003@auburn.edu

ABSTRACT

An agile information acquisition and management system for the Magnetized Dusty Experiment is presented. The data storage of the system is divided into two categories. The first is the research data which include time-series pulse data and video data from the experiment. This bulk research data is stored as normal files on a file system. The second category is the research metadata, which precisely describes the setup information as well as instrumental set-points of an experiment. We use a relational database to store this data. Using a relational database allows us to impose relationships across different runs of experiments at the same time allow us to store mixture of flat and hierarchical data. In addition the database allows the research to specify precise queries to find the data of interest. This report describes the structure of the research data as well as the metadata database. In addition, a web interface to the database is also presented.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Application—*spatial databases and GIS*

General Terms

Algorithms and Experimentation

Keywords

Location-based services, location privacy and spatial cloaking

1. INTRODUCTION

This report describes information management system for the Magnetized Dusty Plasma Experiment (MDPX) [2, 4] project at Auburn University. The primary purpose of the system is to store and retrieve experiment calibration, setup, and summarized research data. The following are the design motivation of the database system:

- Fast retrieval of stored data that has not been a goal of older database systems .
- Sophisticated data search and filtering capability, which allows researchers to form relations and hypothetical research proposals across multiple experimental setups and scenarios.

In addition, the system was designed to be agile and modular. New features can be easily added to the database, and database can be accessed through any software package or language that supports data access via SQL[1].

The system is roughly divided into 3 major parts as shown in Figure 1: relational database management system, LabVIEW[2] control and data acquisition subsystem, and web-based data manager. Our LabVIEW control software interfaces with (for the lack of proper technical term) the dusty plasma experimental hardware and stores setup and experiment data into the database. The web interface allows researchers and operators to monitor and search acquired data from experiment runs. This report will cover the design of the database and overview the capability of the web interface. The LabVIEW control

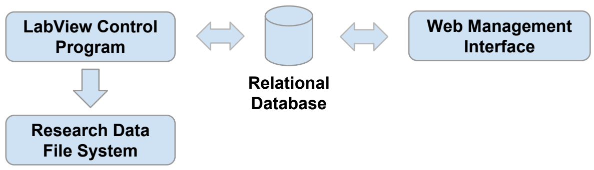

Figure 1: Data acquisition system overview.

[Report outline]

2. COMPARATIVE DATABASE SYSTEMS

3. DATA STORAGE

The acquired experiment research data is split into two categories: experiment metadata and bulk research data. Experiment metadata precisely describe repeatable experimental parameters. This information contains:

- Calibration (?)
- Parts

[1]http://en.wikipedia.org/wiki/SQL
[2]http://www.ni.com/labview/

- Part setup
- Parametric set-points
- If experiment is a systematic scan from a control program
- Recorded summarized research data
- Storage location of bulk research data
- User management

The metadata is stored in a MySQL[3] relational database. There are three main reasons that we use MySQL to store this data. The first is that MySQL allows us to impose relationships among experiments. A simple relationship would be two experiments sharing the same part setup (but not necessary the same set-points). Second, MySQL supports broad range of data searching capabilities, which facilitates researchers to quickly find experiments of interest. The final reason is that MySQL has great performance of data searching. MySQL is the most used relational database management system with proven data storage and retrieval performance. In addition, according to a publication by Princeton Plasma Physics Laboratory on data management for Plasma Physics Databases [1], a relational database is several factors faster than MDSplus [3] database with reasonable data compression ratio. The metadata can be thought as a research data index to find proper experiments.

The second broad category of data is the bulk research data. These include dense time-series pulse measurements, and video files. Much of these files are large files and they are stored on file system outside of the relational database. The location for each experiment is easily searchable via metadata database.

3.1 Metadata Database

The database is a relational database that is used to store and to perform complex analytical query operations of the metadata. Figure 5 in Appendix A shows the database schema that contains all the tables.

Note: In addition to the tables shown in Figure 2, the database also contains a set of user defined views that are currently used by the web interface to show data merged from different tables.

The database tables are divided into 4 groups that are described in this section.

3.1.1 Catalog

The *catalog* consists of a set of tables that keeps an inventory of parts and materials that we have at our facility and are used in the experiments. Table 1 summarizes the purpose of the tables in this group.

3.1.2 Setup

The *setup* is the group of table shown in the middle section of the schema in Appendix A (colored in blue). This group holds the bulk of the searchable metadata of the experiments and contains information of how an experiment is setup – which machine parts are used, how parts are connected, what are the set-points for instruments, and the grouping of experimental studies.

Table 2 summarizes the purpose of the tables in this group.

3.1.3 Summarized Measurements

The *Measurements* table is the only table in this category. For each run of experiment, there is a corresponding set of aggregated measurement data in the this table. The purpose of this table is to allow the researchers to perform queries to find the research data of interest.

[3] http://www.mysql.com/

Table 1: Tables in Catalog

Table	Description
PartCategories	A hierarchy of parts. An example of a category is flanges, with side and top flanges as subcategories.
Parts	A catalog of experimental hardware parts. For example a flange, a hexagon chamber, or a camera.
ChamberSides	Enumerates the number of sides for part in Parts table that has the category of *chamber*.
GasTypes	A catalog of gases used for experiments, e.g. Argon.
DustTypes	A catalog of particles used for experiments, e.g. Silica 0.5.

Table 2: Tables in Setup

Table	Description
VesselSetups	Each record of this table denotes a set of parts that makes a vessel, and the date the setup was constructed.
SetupParts	This table contains a list of parts for a vessel setup. This table references the parts in *Parts* table.
SetupCameras	This table contains a list of parts and parameters for a vessel setup that are cameras.
SetupProbes	This table contains a list of parts and parameters for a vessel setup that are probes.
PIVSetup	Reserved table currently not used.
LIFSetup	Reserved table currently not used.
ExperimentSetups	This table contains the set-points of instruments.
Experiments	This is a table for grouping a set of related runs. Each run would have set of experiment set-points. In essence, this table groups a set of entries in the ExperimentSetups table.

3.1.4 Access Control and User Management

The last group are the user management and access tables. This category contains the *Users* table and the tables under the group named *Web UI Permissions*. These tables defines user roles and controls which role has access to and what action can be performed to different part of the database. Currently only the web interface is using this access control mechanism. Description of these tables follows (in Table 3).

3.1.5 Relationships

3.2 Structure of Research Data

To be completed by Auburn University Plasma Science Laboratory Group (Dr. Thomas).

4. WEB INTERFACE

The web interface allows users of the system to view and, if permission granted, modify the information stored in the relational database without having to write complex data query sentences. A perfect example of the power of the web UI can be demonstrated via the *catalog*. Due to the hierarchical nature of the Part Categories (where one categories could have subcategories), it is cumbersome to insert or modify a category directly using SQL. For ex-

Table 3: Tables in Setup

Table	Description
Users	Registered user that are allowed to login through web interface.
Roles	A list of user roles.
UserRoles	A list of role the users are assigned to. A user could be assigned to multiple rows.
RolePermissions	Action that can be performed on the web interface. For example, only users assigned with role of Admin can show the registered users ini the database.

ample, if a new part category is to be inserted using raw SQL, the user will first have to write a query sentence to find the identifier of the desired parent category; then, another query sentence to insert the new category, all the while there is room for human copying error.

Using the web interface, the user can easily see the hierarchical structure of the data as shown in Figure 2, and to modify a category, just right click on a tree node, and a list of supported actions are shown.

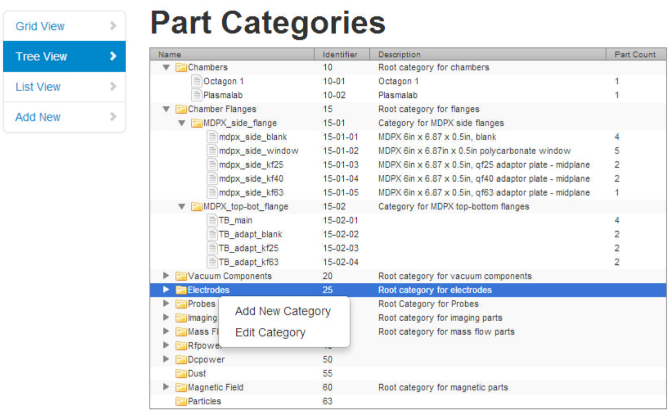

Figure 2: Tree view of the part categories.

The web interface is implemented via PHP 5.3 with Yii Framework 1.1. The framework provides agnostic database access; therefore, RDBMS from different vendors can be used as the data store. The RDBMS that are supported include MySQL(what we use), PostgreSQL, SQLite, Microsoft SQLServer.

4.1 Inserting and Modifying Data

The interface facilitates entering new data into the database as shown in the Figure 3.

Two ways that the interface make entering and modifying data easier and less error-prone.

The first is that data input form provides selection drop-down menu for data item when possible. The content of these selection menus are usually populated from tables in the database. For example, to insert a new entry into the Experiments table (a experiment group), a researcher and operator user ID is needed. The data entry interface provides a selection menu for these fields so that the user does not have to perform additional step of looking up user IDs. When the data is submitted, the interface inserts the correct user IDs.

The second is that, the interface provides data validation. If a data item does not match a format or validation rule, the data is

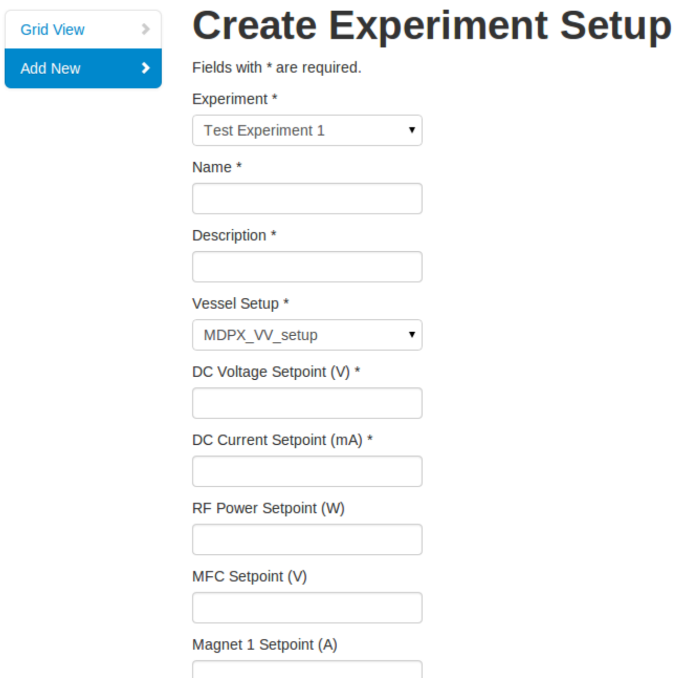

Figure 3: Data entry form and data validation.

rejected and will not be saved to the database. The database also enforces data integrity rules. The interface adds a second layer of data integrity and constraint checks.

4.2 Data Filtering and Searching

Data filtering is supported in grid views. Figure 4 shows the grid view of parts in the system. Users can specify filtering criteria on the columns at the top of the grid view. For string columns, the user can enter the complete or partial search string. For example, to search and researcher with the last name of "Thomas", the user can enter either "Thomas" or just "Thom" as filtering string. For numerical columns, comparison operators (<, >) can be used in filtering criteria.

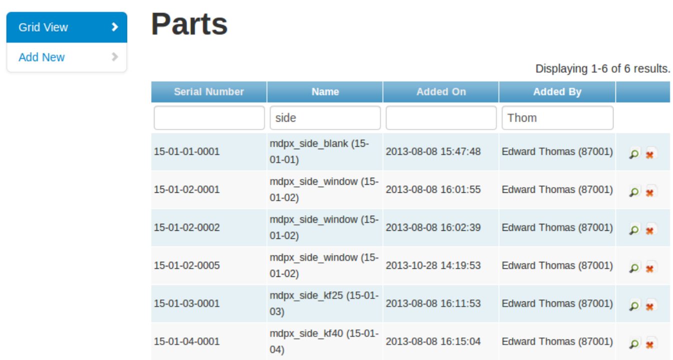

Figure 4: Data Filtering.

APPENDIX

A. DATABASE TABLES

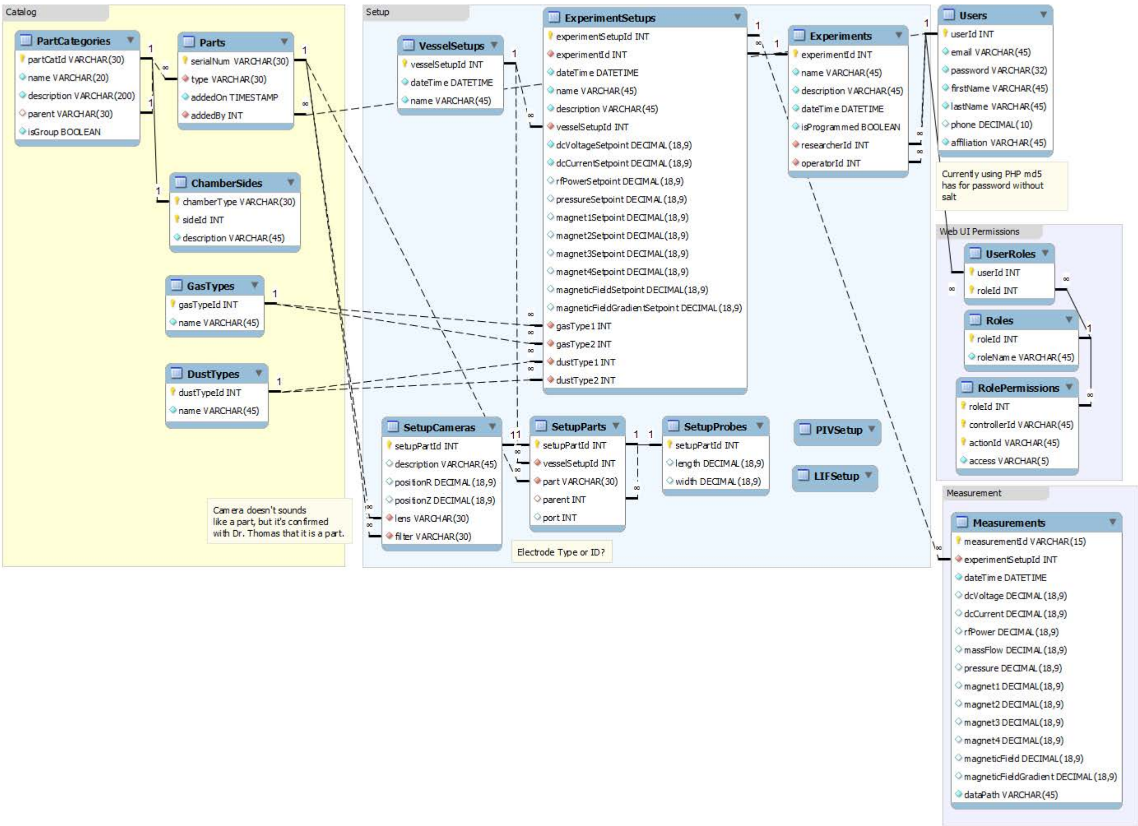

Figure 5: Database schema.

B. DATABASE COLUMN DESCRIPTION

PartCategories Table

Column	Description
partCatId	Identifier of part category.
name	Display name of part category.
description	A textual description of this part category.
parent	Contains Part Category ID of the parent part category node.
isGroup	Denotes if a Part Category is an intermediate node or a leaf node.

Parts Table

Column	Description
serialNum	Unique identifier of each part. TWo physical parts of the same type in the catalog will have the same category ID, but different serial number.
type	Part Category ID.
addedOn	The date and time when the part is added to the catalog.
addedBy	The ID of the user who added the part to the catalog.

GasTypes Table

Column	Description
gasTypeId	Unique integer identifier.
name	Name of the gas, e.g. Argon.

DustTypes Table

Column	Description
gasTypeId	Unique integer identifier.
name	Name of the particle, e.g. Silica 0.5.

VesselSetups Table

Column	Description
vesselSetupId	Unique integer database record identifier.
dateTime	Date and time when this setup was created.
name	A descriptive name of this setup.

SetupParts Table

Column	Description
setupPartId	Unique integer database record identifier.
vesselSetupId	The ID of the vessel setup to which this part is attached.
part	Serial number of this part.
parent	The ID of the setup part to which this part is attached.
port	The location on the parent part on which this part is attached.

SetupCameras Table

Column	Unit	Description
setupPartId		The ID of the setup part of which this record is describing.
description		Description of the camera.
positionR	Degree	The R position of the camera.
positionZ	Centimeter	The Z position of the camera.

SetupProbes Table

Column	Unit	Description
setupPartId		The ID of the setup part of which this record is describing.
length	Centimeter	The length of the probe.
width	Centimeter	The width of the probe.

ExperimentSetups Table

Column	Unit	Description
experimentSetupId		Unique integer database record identifier.
experimentId		ID of an record in Experiments table.
dateTime		Date and time on which this setup was created.
name		Descriptive name.
description		Description.
vesselSetupId		Database ID of the vessel setup of this experiment.
dcVoltageSetpoint	Volt	
dcCurrentSetpoint	Milliamp	
rfPowerSetpoint	Watt	
pressureSetpoint	Volt	
magnet1Setpoint	Amp	
magnet2Setpoint	Amp	
magnet3Setpoint	Amp	
magnet4Setpoint	Amp	
magneticFieldSetpoint	Tesla	
magneticFieldGradientSetpoint	Tesla/Meter	
gasType1		
gasType2		
dustType1		
dustType2		

Experiments Table

Column	Description
experimentId	Unique integer database record identifier.
name	Descriptive name of the experiment group.
description	Description of the experiment group.
dateTime	The date and time on which this group was created.
isProgrammed	A boolean flag that signifies if this group of experiment is conducted by a program that iterates through a range of values for a set of paramenters.
description	Description of the experiment group.
researcherId	Foreign key to Users.userId. This means each record in this table has a researcher.
operatorId	Foreign key to Users.userId. This means each record in this table has an operator.

Measurements Table

Column	Unit	Description
measurementId		Primary key of this table.
experimentSetupId		Foreign key to ExperimentSetups.experimentSetupId. This relationship means every measurement is linked with one experiment setup, and every experiment setup has multiple measurements.
dateTime		Date and time when a record was recorded.
dcVoltage	Volt	Monitored average DC voltage of ...?
dcCurrent	Milliamp	Monitored average DC current of ...?
rfPower	Watt	
massFlow	Volt	
pressure	?	
magnet1	Ampere	
magnet2	Ampere	
magnet3	Ampere	
magnet4	Ampere	
magneticField	Tesla	
magneticFieldGradient	Tesla/Meter	
dataPath		Where the corresponding data is stored.

Users Table

Column	Description
userId	Primary key for this table.
email	Email address of this user.
password	Password of the user hashed using md5 without salt.
firstName	First name of the user.
lastName	Last name of the user.
phone	Contact telephone number of the user.
affiliation	Associated organization of the user.

Roles Table

Column	Description
roleId	Primary key for this table.
roleName	Name of this role, e.g. Admin.

RolePermissions Table

Column	Description
roleId	Foreign key to Roles.roleId. This means for reach
controllerId	Controller identifier of the web interface.
actionId	Action identifier of the web interface.
access	Have value of either ALLOW or DENY. This column specifies whether an operation identified by pair controllerId, actionId can be performed by a role.

Acknowledgments

This research has been funded in part by the National Science Foundation grants CNS-0831502 (CT) and CNS-0855251 (CRI).

C. REFERENCES

[1] W. Davis and D. Mastrovito. Dbaccess: interactive statistics and graphics for plasma physics databases. *Fusion Engineering and Design*, 71(1âĂŞ4):183 – 188, 2004. <ce:title>4th {IAEA} Technical Meeting on Control, Data Acquistion, and Remote Participation for Fusion Research</ce:title>.

[2] E. T. Jr, R. L. Merlino, and M. Rosenberg. Magnetized dusty plasmas: the next frontier for complex plasma research. *Plasma Physics and Controlled Fusion*, 54(12):124034, 2012.

[3] J. Stillerman, T. Fredian, K. Klare, and G. Manduchi. Mdsplus data acquisition system. *Review of Scientific Instruments*, 68(1):939–942, 1997.

[4] E. Thomas, R. Merlino, and M. Rosenberg. Design criteria for the magnetized dusty plasma experiment. *Plasma Science, IEEE Transactions on*, 41(4):811–815, 2013.