12

A Web-Based Data Management and Analysis System for CO_2 Capture Process

Yuxiang Wu and Christine W. Chan
Energy Informatics Laboratory, Faculty of Engineering, University of Regina
Regina, Saskatchewan, S4S 0A2,
Canada

1. Introduction

Fossil fuels constitute a major energy resource for Canada. In 2002 alone, the production of oil, gas and coal contributed over $30 billion to the Canadian economy.

Fossil fuel is presently the world's most abundant, economical and reliable fuel for energy production. However, the industry now faces a major challenge because the production of fossil fuels including coal, crude oil and gas, and the processes currently used for energy production from such fuels, can have adverse environmental consequences. Hence, along with the positive economic advantages of energy production using fossil fuels come the responsibility of mitigating the consequent adverse environmental and climate-change impacts (Harrison et al., 2007).

Carbon capture and storage (CCS) is an approach for reducing carbon dioxide (CO_2) emissions to the environment by capturing and storing the CO_2 gas instead of releasing it into the air. The application of CCS to a modern conventional power plant could reduce CO_2 emissions to the atmosphere by approximately 80-90% compared to a plant without CCS (IPCC, Metz, & Intergovernmental Panel on Climate Change Working Group III, 2005). CO_2 capture technologies mainly include: chemical absorption, physical absorption, membrane separation and cryogenic fractionation. Among these technologies, chemical absorption of CO_2 is one of the most mature technologies because of its efficiency and low cost.

The highly complex CO_2 absorption process generates a vast amount of data, which need to be monitored. However, industry process control systems do not typically incorporate operators' heuristics in their intelligent control or data analysis functionalities. Our objective is to construct an intelligent data management and analysis system that incorporates such human experts' heuristics. The Data Analysis Decision Support System (DADSS) for CO_2 capture process reported in (Wu & Chan, 2009) is a step towards filling this gap in automated control systems. However, the DADSS is a standalone PC-based system with limited flexibility and connectivity. In this paper we present a web-based CO_2 data management and analysis system (CO_2DMA), which overcomes these limitations.

The system presented in this paper was built based on data acquired from the Pilot Plant CO_2 capture process of the International Test Centre for CO_2 capture (ITC), located at the University of Regina in Saskatchewan, Canada. The CO_2 capture process at the ITC is monitored and controlled by the DeltaV system (Trademark of Emerson Process

Source: Decision Support Systems, Book edited by: Chiang S. Jao,
ISBN 978-953-7619-64-0, pp. 406, January 2010, INTECH, Croatia, downloaded from SCIYO.COM

Management, U.S.A), which adopts the technology of Object-Linking and Embedding (OLE) for Process Control (OPC). OPC standards are widely used in industry process control and manufacturing automation applications ("OLE for process control," n.d.). More detailed information about OPC will be provided later.

The paper is organized as follows: Section 2 presents some background literature on decision support systems used for problem solving in engineering. Section 3 gives some background on knowledge acquisition and knowledge representation in the process of developing the web-based carbon dioxide data management and analysis system called CO_2DMA. Section 4 discusses software engineering techniques used in system development. Section 5 presents some sample test runs of CO_2DMA. Section 6 concludes the paper and presents some directions for future work.

2. Background

2.1 Amine-based CO_2 capture process

The purpose of CO_2 capture is to purify industrial gas streams before they are released into the environment and produce a concentrated stream of CO_2. Current post-combustion CO_2 capture technologies mainly include: chemical absorption, physical absorption, membrane separation, and cryogenic fractionation (Riemer, 1996). The selection of a technology for a given CO_2 capture application depends on a variety of factors, such as capital and operating costs of the process, partial pressure of CO_2 in the gas stream, extent of CO_2 to be removed, purity of desired CO_2 product, and sensitivity of solutions to impurities (i.e. acid gases and particulates, etc.) (White et al., 2003).

In recent years, chemical absorption has become the most commonly used technology for low concentration CO_2 capture, and amine solvents are the most widely used for chemical absorption. In the amine-CO_2 reaction, CO_2 in the gas phase dissolves in an aqueous amine solvent; the amines react with CO_2 in solution to form protonated amine (AMH+), bicarbonate ion (HCO3-), carbamate ($AMCO_2$-), and carbonate ion (CO32-) (Park et al., 2003). The system described in this paper is constructed based on ITC's CO_2 capture process, which primarily implements this chemical absorption as an industrial process for pilot run and research purposes.

Fig. 1 shows a process flow diagram of the CO_2 capture plant. Before the CO_2 is removed, the flue gas is pre-treated in the inlet gas scrubber, where the flue gas is cooled down, and particulates and other impurities such as sulfur oxide (SOx) and nitrogen oxide (NOx) are removed as much as possible. The pre-treated flue gas is passed into the absorption column by an inlet-gas feed blower, which provides the necessary pressure for the flue gas to overcome the pressure drop in the absorber. In the absorber, the flue gas and lean amine solution contact each other counter-currently. With the high temperature steam provided by the boiler, the amine selectively absorbs CO_2 from the flue gas. The amine solution carrying CO_2, which is called CO_2-rich amine, is pumped to the lean/rich heat exchanger, where the rich amine is heated to about 105 °C by means of the lean amine solution. The heated CO_2-rich amine enters the upper portion of the stripper. Then the CO_2 is extracted from the amine solution, which is now the lean amine solution. Most of the lean amine solution returns to the lean amine storage tank and then recycles through the process for CO_2 absorption. A small portion of it is fed to a reclaimer, where the degradation by-products and heat stable salts (HSS) are removed from the amine solution. The non-regenerable sludge is left behind in the reclaimer and can be collected and disposed. The CO_2 product

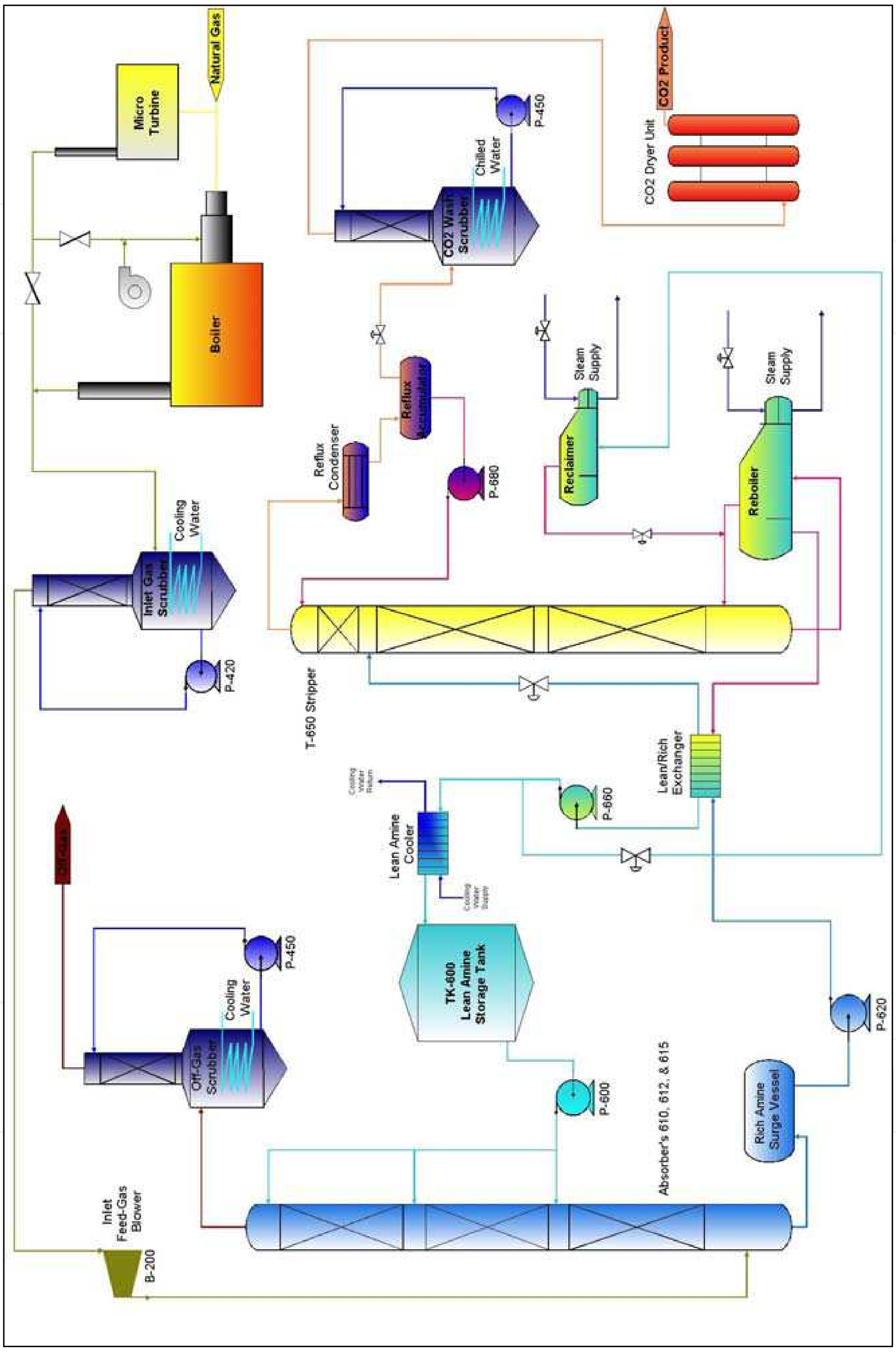


Fig. 1. Process flow diagram of the CO_2 capture process

and water vapour from the top of the stripper is passed through a reflux condenser. Most water is condensed inside the condenser, and the residual amine solvent is passed back to the stripper column reflux section so as to desorb CO_2 again. The CO_2 product enters a CO_2 wash scrubber, where the CO_2 gas is cooled to the desired temperature of 4 °C. From there, the CO_2 can be vented into the atmosphere or passed through a dryer and purification unit to produce food grade quality CO_2.

2.2 Decision support system

The decision support system (DSS) is a computerized system built for aiding the process of decision making (Turban, 1993). Decision support systems can supplement human cognitive deficiencies by integrating different sources of information and providing intelligent access to relevant knowledge. They can also help people select among well-defined alternatives based on formal or theoretical criteria and methods from engineering economics, operations research, statistics and decision theory. For problems that are intractable by formal techniques, artificial intelligence methods can also be employed (Druzdzel & Flynn, 1999).

Decision support systems (DSS) have been widely used in diverse domains, including business, engineering, the military, medicine and industrial areas. Some applications of decision support systems in process industries are described as follows. Geng et al. (2001) presented a knowledge-based decision support system that aids users select petroleum contaminant remediation techniques based on user-specified information. Szladow and Mills (1996) described a DSS that presents the application work-flows used for training new operators in five heavy industries including iron and steel, cement, mining and metallurgy, oil and gas, and pulp and paper. Flores et al. (2000) described an intelligent system that links multiple anaerobic systems for wastewater treatment to a common remote central supervisor via wide area networks. The local control systems have a hybrid structure, which is comprised of algorithmic routines for data acquisition, signal preprocessing, and calculation of plant operation parameters. Kritpiphat et al. (1996) developed an expert DSS for supervisory monitoring and control of a water pipeline network in a prairie city in Canada.

2.3 Web-based system

In the past decade, the World Wide Web has successfully demonstrated how the internet technologies can support information sharing and knowledge exchange. Technically, a web-based system can be defined as an application or service which resides in a server remotely or locally. The application or service can be accessed using a web browser from anywhere via the internet.

In the system requirement analysis and design stage of this project, a stand-alone application was considered but not adopted because of a number of reasons: (1) The stand-alone system relies on a particular data file as its data source. This is a limitation because it is not a flexible format that can be useful in future data analysis. (2) Knowledge and data sharing through a specific file will be difficult. (3) System and data access is limited to the station in which the system has been installed.

Due to these limitations, a web-based system was considered and adopted due to the following benefits (Liu & Xu, 2001):

- A web-based system has a reduced product development cycle time because of the increased collaboration among all areas of an organization and its supply chain, and the easy access to system information;

- A web-based system can make use of a full library of utilities, which help the developers avoid tedious coding and enable sharing of code common to various modules;
- A web-based system facilitates the uses in accessing the information of data source. When properly implemented, the web-based system can simplify many day-to-day user operations by managing and automating routine tasks, such as searching and completing data reports;
- Management of a project for constructing a web-based system is easier because a web-based system allows the system developers or maintainers to track the status of the system more effectively and facilitates validating the development work.

Therefore, a web-based CO_2 data management and analysis system was designed and developed; the knowledge engineering process for the system is described as follows.

3. Knowledge engineering of CO_2DMA

The knowledge engineering process (Jack Durkin & John Durkin, 1998) for building CO_2DMA involves the two primary processes of knowledge acquisition and knowledge representation.

3.1 Knowledge acquisition

The knowledge useful for developing a knowledge-based system refers to the problem solving activities of a human expert. Knowledge acquisition (KA) is the process of elucidating, analyzing, transforming, classifying, organizing and integrating knowledge and representing that knowledge in a form that can be used in a computer system (Druzdzel & Flynn, 1999) (Geng et al., 2001).

The objective of knowledge acquisition in this project is to obtain specific domain knowledge on how to filter and analyze CO_2 data. The knowledge was acquired during interviews with the expert operator, who is the chief engineer of the ITC pilot plant. Relevant knowledge on the existing process of filtering and analyzing CO_2 data includes:

- Data points or tags used: data obtained for 145 tags need to be analyzed for monitoring the CO_2 capture process.
- Steps or methods for filtering data: The original CO_2 data captured by the DeltaV system suffer from the deficiencies of being: (1) incomplete (lacking attribute values or certain attributes of interest, or containing only aggregate data), (2) noisy (containing errors, or outlier values that deviate from the expected), and (3) inconsistent (containing discrepancies in the tag names used to label attributes). In other words, there are reported errors, unusual values, redundant values and inconsistencies in the data recorded for some transactions. The data was pre-filtered by filling in missing values, eliminating noisy data, identifying or removing outliers, and resolving inconsistencies. The pre-filtering procedure involves the following four steps:

Step 1: IF Gas flow rate into absorber <= 4.4 OR
Gas flow rate into absorber >= 4.6,
THEN Delete this row of data

Step 2: IF Heat Duty <= 20000 – (20000 * 0.1) OR
Hear Duty >= 100000 + (100000 * 0.1),

	THEN	Delete this row of data
Step 3:	IF	Input CO_2 fluid gas <= 9.0 OR Input CO_2 fluid gas >= 12,
	THEN	Delete this row of data
Step 4:	IF	Rebuilder steam flow rate >= 70000 – (70000 * 0.05) AND Rebuilder steam flow rate <= 70000 + (70000 * 0.05) AND Heat Duty >= 70000 – (70000 * 0.2) AND Heat Duty <= 70000 + (70000 * 0.2),
	THEN	Keep this row of data
	ELSE IF	CO_2 production rate >= 0.7 – (0.7 * 0.1) AND CO_2 production rate <= 0.72 + (0.72 * 0.1),
	THEN	Keep this row of data
	ELSE	Delete this row of data

Where	*Gas flow rate into absorber*:	Absorber inlet gas flow rate (1000 m^3/day);
	Heat Duty:	Energy used (BTU/lb-mole CO_2);
	Input CO_2 fluid gas:	Absorber input fluid CO_2 gas (CO_2%);
	Rebuilder steam flow rate:	Steam from rebuilder flow rate (kg/h);
	CO_2 production rate:	CO_2 production rate (tones/day).

3.2 Knowledge representation

The knowledge obtained from the KA process was represented in a number of classes, which are referred to as knowledge components. The components are organized into the class hierarchy shown in Fig. 2. Most of the classes were implemented as tables in a database.

The class of Project represents the concept of a CO_2 data analysis project. A project contains a Profile, Data, and Subsets. When a new analysis of the data is initiated, a new project is created.

The class of Profile stores the tags and data filtering steps the user uses in one analysis case. The class of tag represents a component from which parameter values are obtained. A tag has a name, the area it is in, the path, a description, the units, minimum/maximum values, and a flag for whether or not the tag is being used. The class of Step represents a filtering step. Each step has a name, a description, and a flag for whether or not it is used.

The class of Data is represented as a row of values which is identified by the date. The values are augmented with an inUse flag and a comment field to record if a value has been filtered and the reason for performing the filtering.

The class of Subset is a portion of the data. A subset has a name, a list of tags selected by the user, and the actual data. The class of SubsetData is very similar to the class of Data, except that it consists of only subsets of data. SubsetData is a row of data in the subset and only contains the date and parameter values.

4. System development of CO_2DMA

This section presents the structure of the CO_2DMA, and several software engineering technologies that were used during system development.

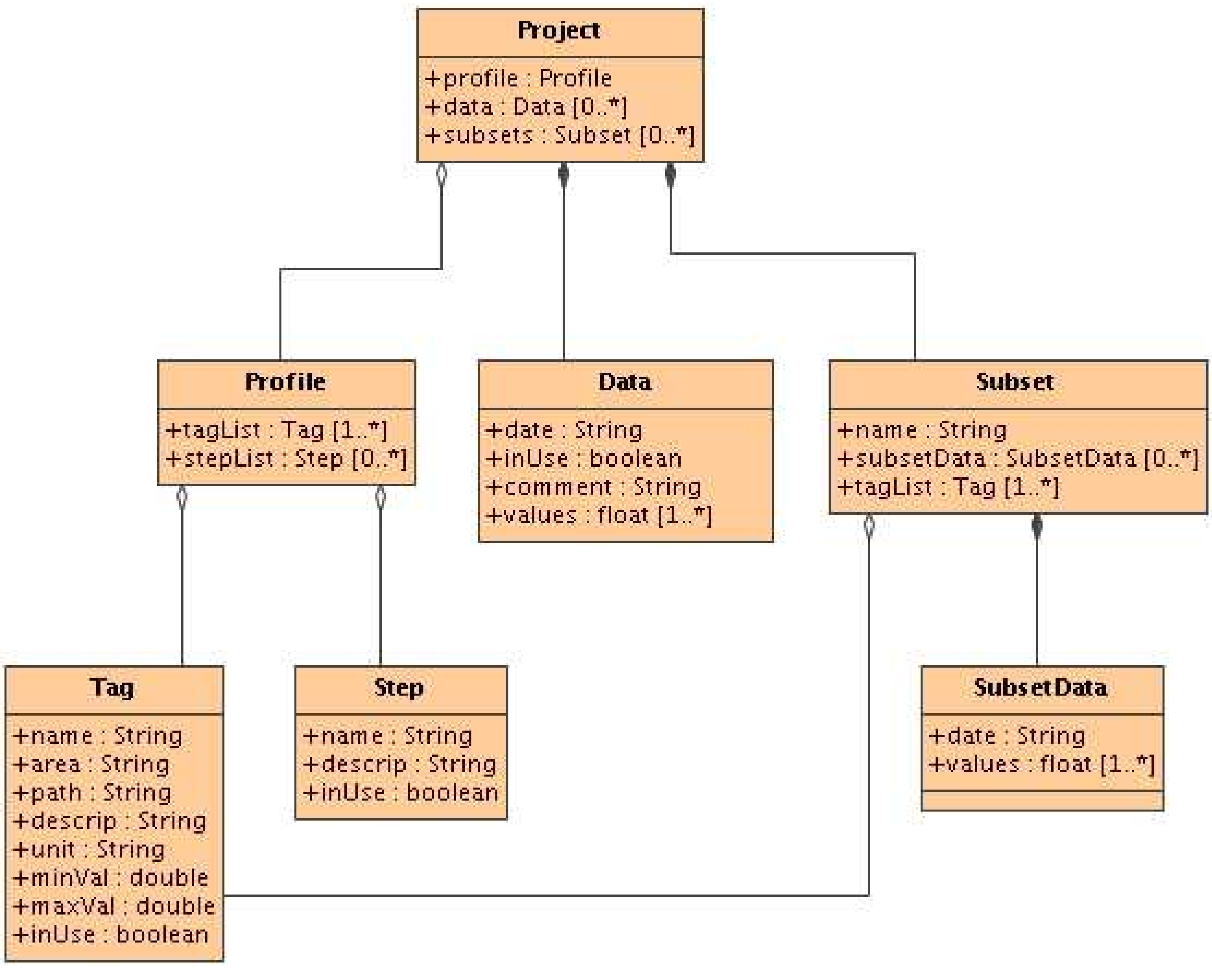


Fig. 2. Class Hierarchy of Knowledge Components

4.1 System structure

The CO_2DMA system consists of four main modules (see Fig. 3): (1) OPC Historical Data Access (HAD) Server module, (2) OPC Data Transporter module, (3) Database Server module, and (4) Web Server module. The OPC HAD Server usually resides in the same computer as the process control system, which is implemented in the DeltaV system at ITC. It is the repository where process data are stored, and which can be accessed only by programs built according to the HDA standards. The OPC Data transporter is a C# (Microsoft® software) program that runs along with the OPC HDA Server in the background. It continually reads data from the OPC HDA Server and converts the data into the appropriate types in order to transfer them into the Database Server. The Web Server component of the system is responsible for communicating with clients through the internet. The clients send request to and retrieve data from the Web Server. Both communication and data transfer are based on the HyperText Markup Language (HTML).

4.2 OPC and OPC transporter

OPC, which stands for Object-Linking and Embedding (OLE) for Process Control, is basically a series of standard specifications ("The OPC Foundation - Dedicated to Interoperability in Automation," n.d.). The OPC standard specifications support

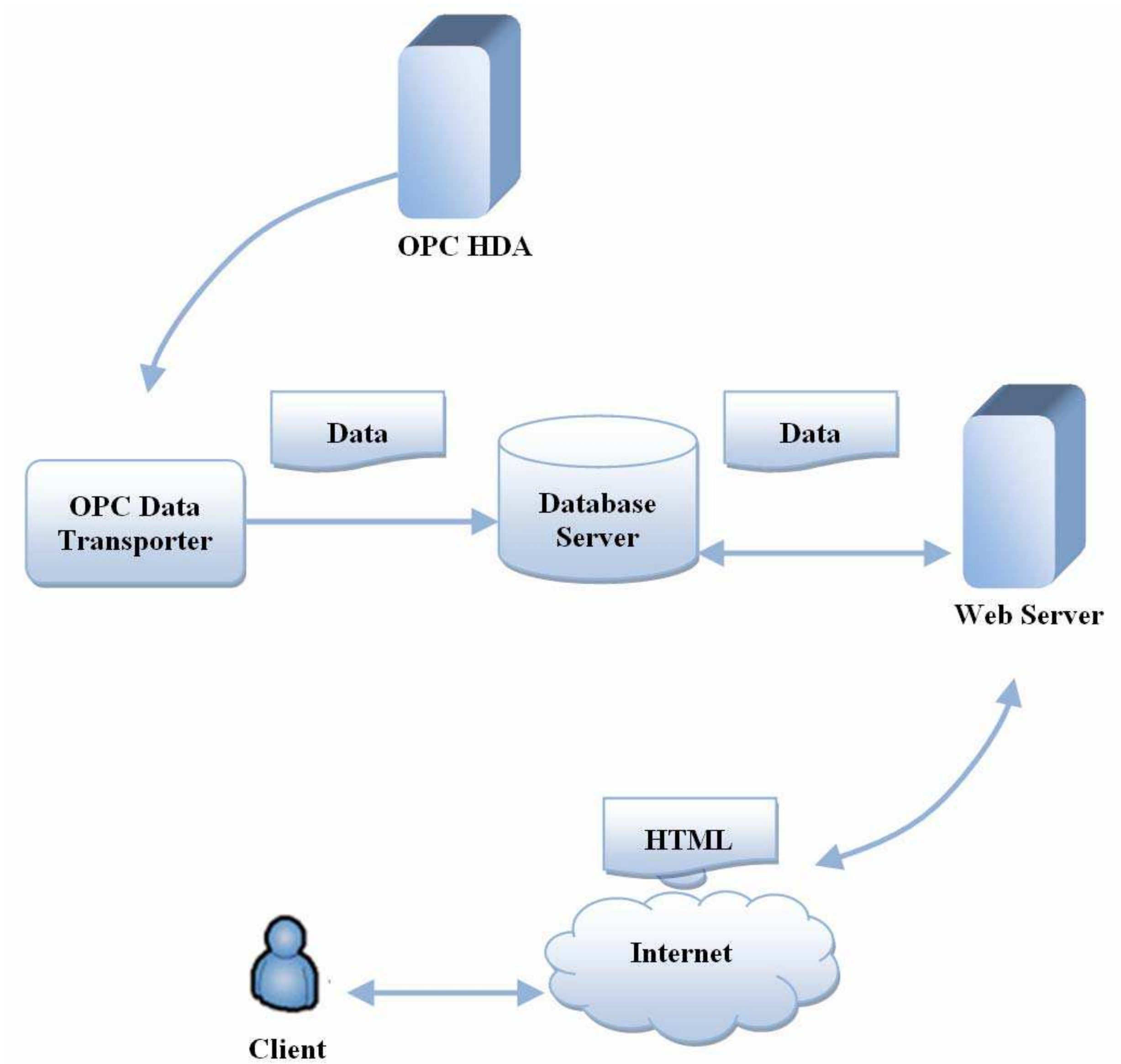


Fig. 3. Structure of CO_2DMA

communication of real-time plant data between control devices from different manufacturers ("OLE for process control," n.d.). The OPC Foundation maintains the standards. Since the foundation was created, more standards have been added.

The purpose of using OPC was to bridge Windows (Microsoft® software) based applications with process control hardware and software applications because the open standards support a consistent method of accessing field data from plant floor devices. The OPC servers define a common interface which can support different software packages to access data derived from the control devices.

Despite its advantages, the OPC technology suffers from two main weaknesses, which the ITC operators found can impede smooth operations of the CO_2 capture process:

The OPC technology, which includes the OPC servers and client applications, are developed based on the windows platform. This presents a problem when data and knowledge need to be shared with an application that is developed on a non-windows platform. In other words, interoperability among different platforms is not supported.

Only applications that support OPC protocols can access the data in the OPC Historical Data Access (HDA) Server, which is where the DeltaV data reside. Hence, data manipulation and analysis had to be handled by OPC client applications, and data cannot be reused by other computational tools that do not share OPC interfaces.

A generic database can address these limitations because it enables retrieving data from the real-time control system and storing them. Since we believe a generic database can render our system more flexible and the data reusable, the component called OPC Data Transporter was constructed for accessing, converting and sending data from the OPC HDA Server to the generic database. The OPC Data Transporter is an OPC client application written in C# (Microsoft® software) using the Historical Data Access (HDA) common library. Currently the transporter runs as a background program within the same machine as the DeltaV control system. It can also reside in a remote machine which physically connects to the control system. In either case, the data will be periodically captured from the OPC HDA Server and converted to the correct data type, then stored in the generic database. This approach supports isolating and protecting the process control system from outside interference, while enabling sharing of data and other useful information from the control system through the data repository implemented as the database.

4.3 Web server development

The Web Server plays a key role in the enhanced version of the DADSS because it acts as an intermediary between the database component and the user on the internet. The server was constructed using the LAMP software bundle, which includes:

- Linux, a Unix-like computer operating system.
- Apache, an open source HTTP Server.
- MySQL (Trademark of MySQL AB), multi-user SQL database management system (DBMS).
- PHP (Hypertext Preprocessor), a computer scripting language originally designed for producing dynamic web pages.

This LAMP bundle has become widely popular since its inception by Michael Kunze in 1998 because this group of free software could provide a viable alternative to commercial packages ("LAMP (software bundle) - Wikipedia, the free encyclopedia," n.d.). Therefore, the LAMP bundle has been adopted for developing the Web server.

Usually the most time consuming part of building a web server is to program the entire site including design of the user interface as well as construction of the background logical layer. This process was often conducted in an ad hoc manner, based neither on a systematic approach, nor quality control and assurance procedures. Recently, different types of web application frameworks supporting different languages have been built. A web application framework is a software framework that is designed for supporting the development of dynamic websites, web applications and services; the framework is intended to simplify the overhead associated with common activity procedures in web development. The general framework usually provides libraries for database access, template frameworks, session management and code reuse.

In development of the web server, CakePHP (trademark of Cake Software Foundation) was adopted as the basic framework because of its detailed documentation and ease of use. Based on CakePHP, the system structure of the web server was designed and developed as shown in Fig. 4.

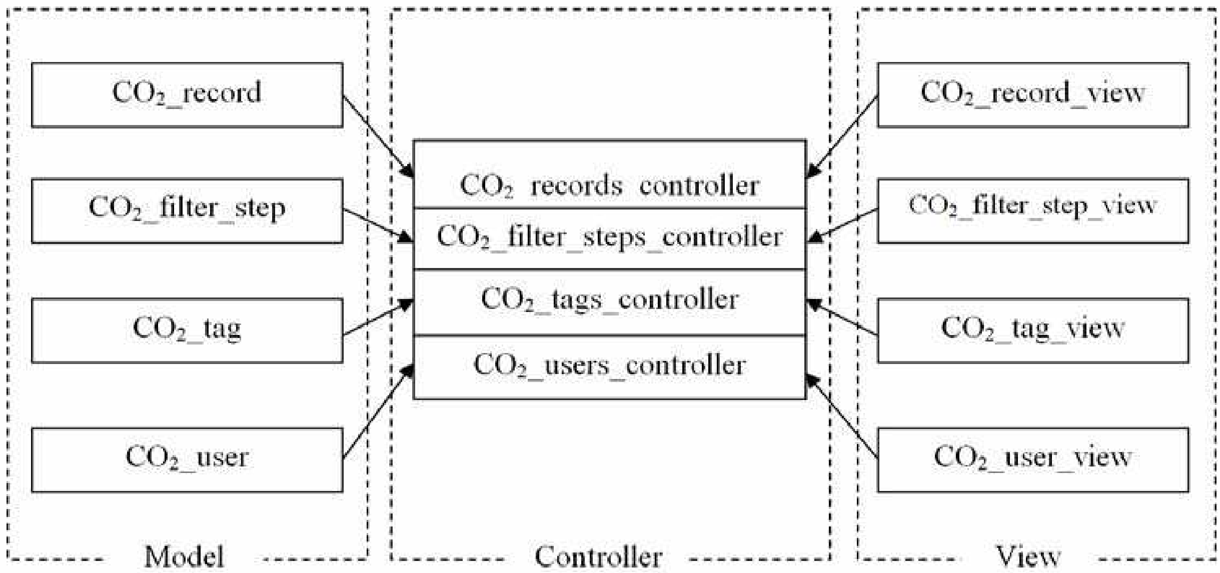


Fig. 4. Web Server Structure

As shown in Fig.4, the structure of the system follows the Model-View-Controller (MVC) architectural pattern (Gamma et al., 1995), which is one of the most commonly adopted application structural models in software development. Recently the model has become widely used in web application development. In the web server system, a model represents a particular database table, and its relationships to other tables and records. The Model also consists of data validation rules, which are applied when the model data are inserted or updated. The View represents view files, which are regular HTML files embedded with PHP code. This provides users with the web page display. The controller handles requests from the server. It takes user input which includes the URL and POST data, applies business logic, uses Models to read and write data to and from databases and other sources, and lastly, sends output data to the appropriate view file ("Basic Principles of CakePHP," n.d.). This system structure has the advantages of (1) modularizing the code and making it more reusable, maintainable, and generally better; and (2) encapsulating the knowledge captured from the expert operator, which was translated into procedures and methods using an object-oriented representation.

4.4 System security

With the proliferation of web-based applications, security is now one of the most crucial considerations in system development. This is also true for the CO_2DMA because it needs to be protected from outside interference. A number of steps were taken to ensure security:

- The website can only be accessed by particular users with the correct user name and password. Accesses are filtered by IP address.
- Data transferred between the user's device and the web server will be encrypted by Secure Sockets Layer (SSL).
- The web server only has privileges to view data from the database server. The connection from the Web Server to the Database Server is read-only; therefore the Web Server cannot do any modifications to the Database.
- The OPC Data transporter module is responsible for transferring data from the DeltaV Server to the Database Server, but it can only access the HDA server and not the DeltaV process control system.

- Hardware Firewalls are configured between:
  a. the CO_2 capture process control system and the Database module
  b. the Database module and the Web Server module

5. Sample session

A sample session of running the system demonstrates how the CO_2DMA assist the operator in accessing and filtering data generated from the CO_2 capture process. When the user enters the website of the system, he/she can select either to display or download the data in order to obtain information about the CO_2 capture process. Hitting the button for display or download would trigger a display of the calendar, as shown in Fig. 5. The user can select the date and time range for which data are required. In response to the user's request, the system displays the entire set of tags, which stand for all the equipment in the process. The tag selection table, as shown in Fig. 6, also includes detailed descriptions such as area, path and unit of each tag. After selecting the tags, the user chooses the pre-filtering steps that are applicable to the data, as shown in Fig. 7. Finally the system displays the filtered data, which are presented in either the browser's viewable format (Fig. 8) or in CSV format and can be downloaded to the user's local machine.

The difference between the unfiltered data and filtered data can be revealed by examining the two sets of data on the sample variables of (1) CO_2 production rate, represented by 'Wet CO_2 out from v-680' and (2) 'Heat Duty', which were selected from the 145 variables monitored by the system. Two trend lines that approximate the data are drawn as shown in Fig. 9 and Fig. 10. The points in the plot of the unfiltered data in Fig. 9 are more scattered because of the high volume of noisy data. After filtering by CO_2DMA, more than 60 rows of noisy data were filtered out from the 590 rows, and the data points are closely clustered as shown in Fig. 10.

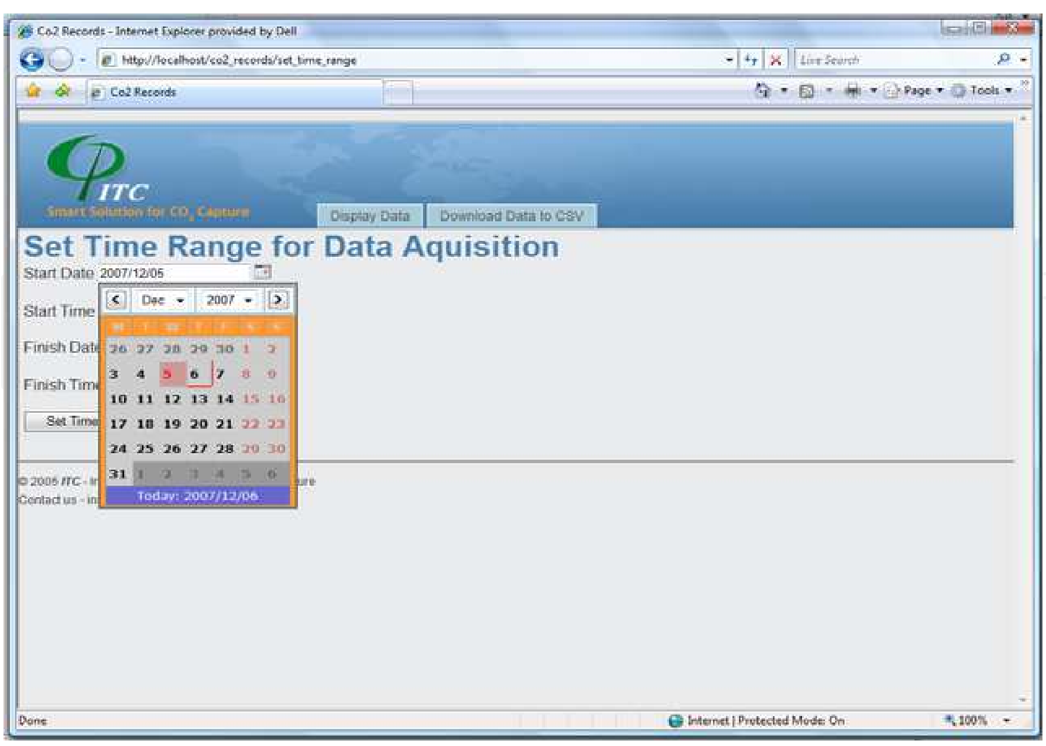


Fig. 5. Date range selection

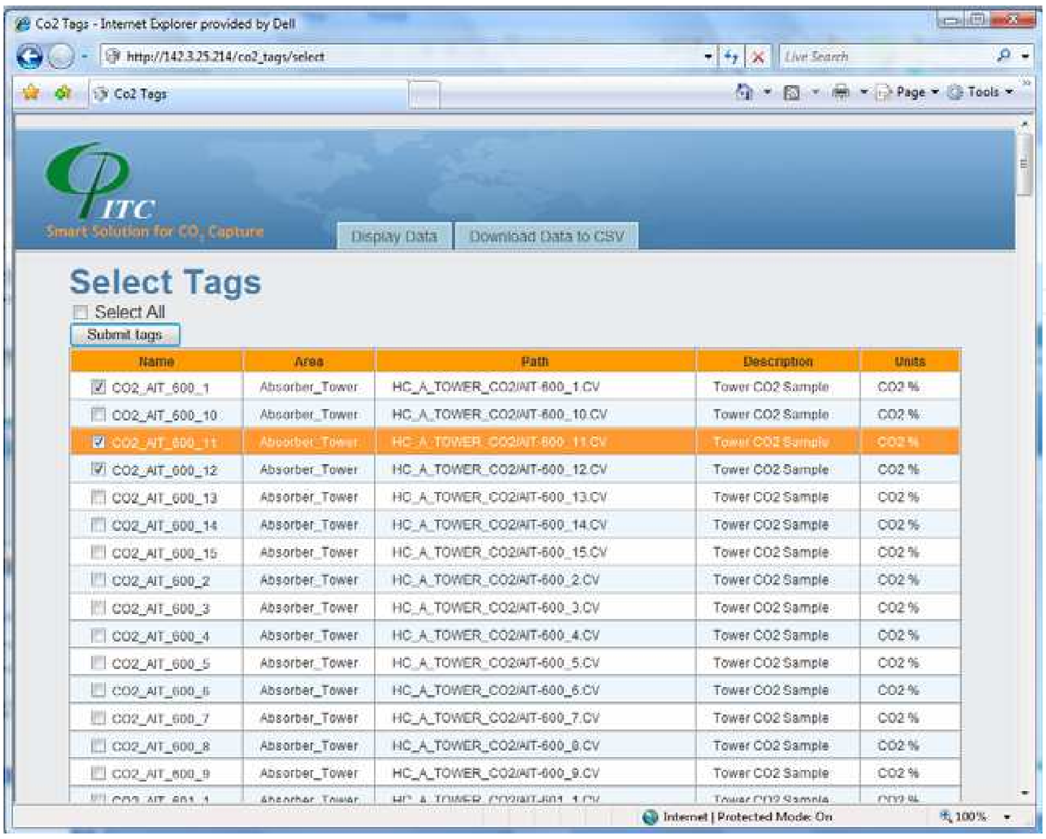


Fig. 6. Tag Selection

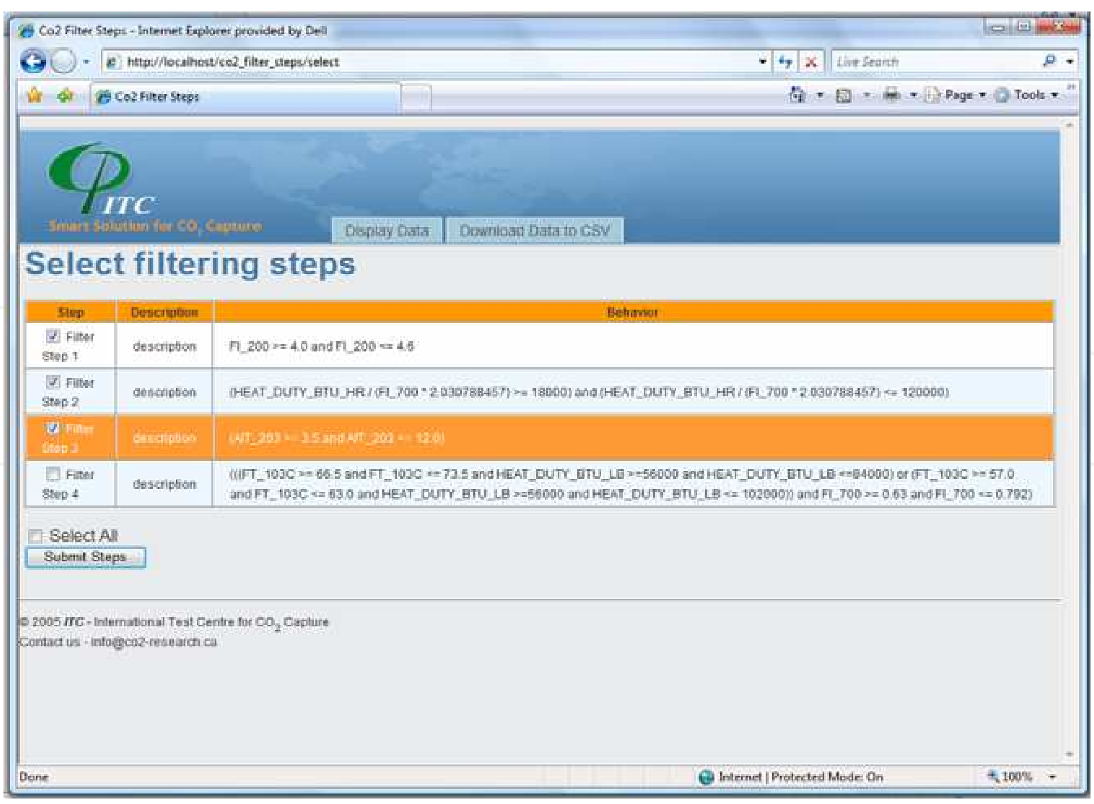


Fig. 7. Selection of filtering steps

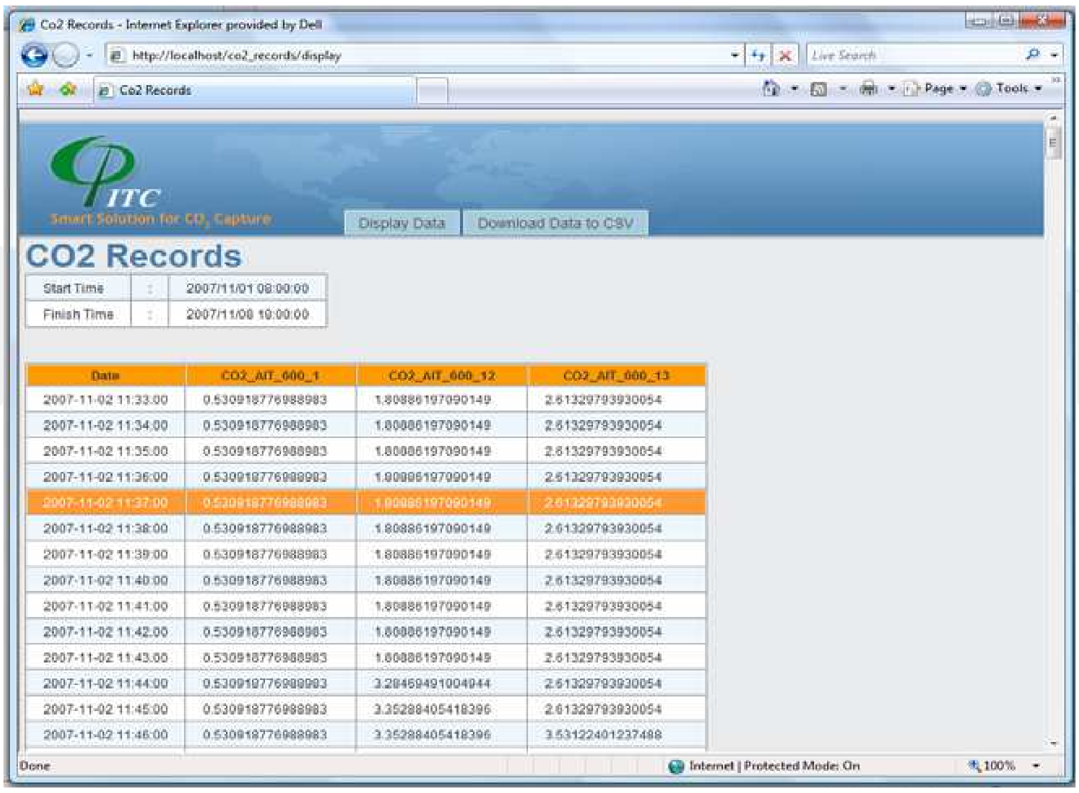


Fig. 8. Sample of filtered data

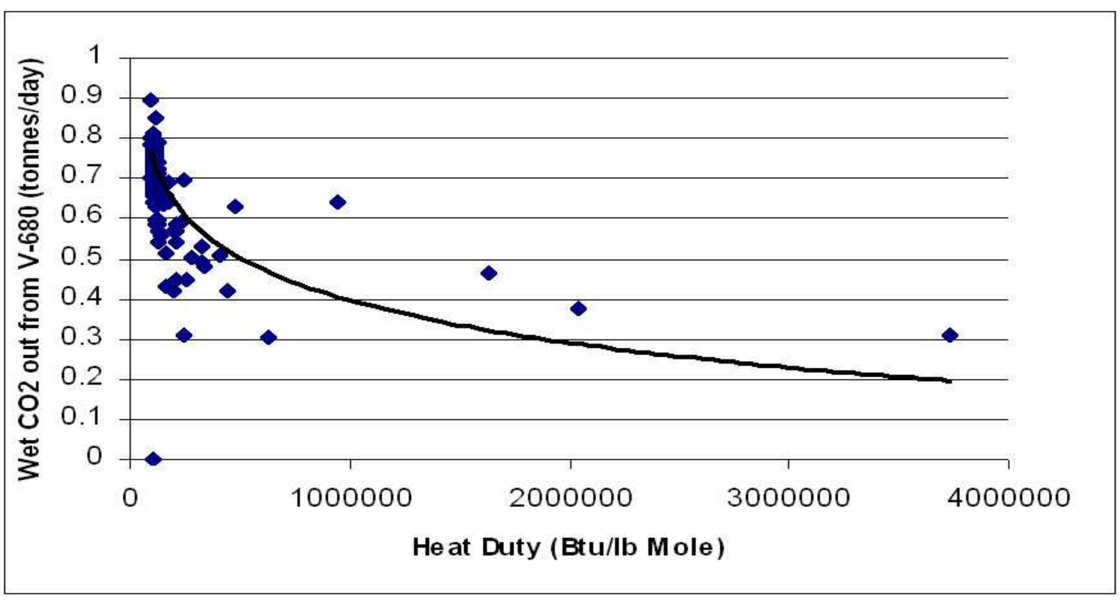


Fig. 9. Plot of data before filtering

6. Conclusion and future works

A web-based data management and analysis system for the CO_2 capture process called CO_2DMA has been developed. The system has a user friendly interface and therefore does not require a steep learning curve for the user. Since the system is built as a web service application, there is no need to install any software in the user's computer. By automatically

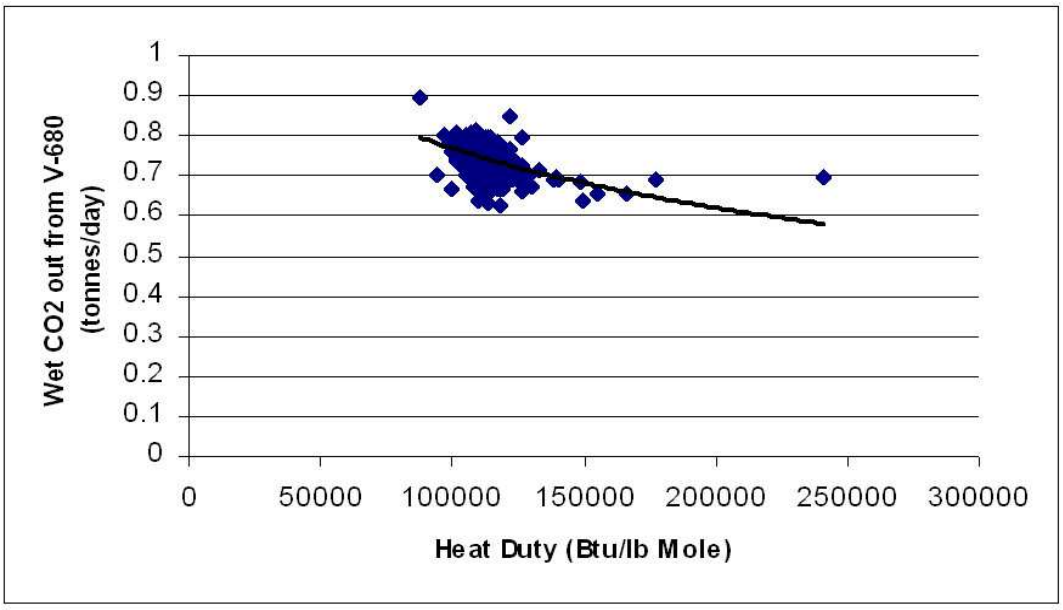


Fig. 10. Plot of data after filtering

filtering and processing hundreds of fields of raw data, the CO_2DMA frees users from having to perform data filtering manually; hence, it improves efficiency of the data filtering process.

Future work for enhancing system efficiency involves saving the user's preferred filtering procedures in a historical configuration file. With this enhancement, the user can simply retrieve their preferred configurations from the configuration file, thereby avoiding the step of selecting the filtering criteria. We also plan to add curve fitting and graphing functions to the system so that the filtered data can be processed for visual displays inside the system instead of being exported to Microsoft Excel (Trademark of Microsoft Office) for further charting. Automation of the data filtering step is only the first step in our research agenda. Future objectives include building system modules for analyzing the data for prediction, planning and control of the CO_2 capture process using artificial intelligence techniques.

7. Acknowledgement

The authors are grateful for the generous support of grants from the Canada Research Chair Program and Natural Science and Engineering Research Council of Canada. We would like to acknowledge the help from members of the Energy Informatics Lab, Robert Harrison and Chuansan Luo. We also would like to thank Dr. Paitoon Tontiwachwuthikul, Don Gelowitz and Dr. Raphael Idem for their valuable suggestions, technical guidance and insights.

8. References

Basic Principles of CakePHP. (n.d.). . Retrieved March 29, 2009, from http://book.cakephp.org/view/13/Basic-Principles-of-CakePHP.

Druzdzel, M. J., & Flynn, R. R. (1999). *Decision Support Systems. Encyclopedia of Library and Information Science. A. Kent*. Marcel Dekker, Inc.

Durkin, J., & Durkin, J. (1998). *Expert Systems: Design and Development* (p. 600). Prentice Hall PTR. Retrieved March 27, 2009, from http://portal.acm.org/citation.cfm?id=551328.

Flores, J., Arcay, B., & Arias, J. (2000). An intelligent system for distributed control of an anaerobic wastewater treatment process. *Engineering Applications of Artificial Intelligence*, *13*(4), 485-494. doi: 10.1016/S0952-1976(00)00015-4.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns. Elements of reusable object-oriented software*. Retrieved March 27, 2009, from http://adsabs.harvard.edu/abs/1995dper.book.....G.

Geng, L., Chen, Z., Chan, C. W., & Huang, G. H. (2001). An intelligent decision support system for management of petroleum-contaminated sites. *Expert Systems with Applications*, *20*(3), 251-260. doi: 10.1016/S0957-4174(00)00063-4.

Harrison, R., Yuxiang Wu, Nguyen, H., Xiongmin Li, Gelowitz, D., Chan, C., et al. (2007). A Decision Support System for Filtering and Analysis of Carbon Dioxide Capture Data. In *Electrical and Computer Engineering, 2007. CCECE 2007. Canadian Conference on* (pp. 1380-1383). doi: 10.1109/CCECE.2007.347.

IPCC, Metz, B., & Intergovernmental Panel on Climate Change Working Group III. (2005). *IPCC special report on carbon dioxide capture and storage: Special Report of the Intergovernmental Panel on Climate Change* (p. 431).

Kritpiphat, W., Tontiwachwuthikul, P., An, A., Chan, C. W., & Cercone, N. (1996). SUPERVISORY AND DECISION-SUPPORT SYSTEM FOR INTELLIGENT MONITORING AND CONTROL OF A PIPELINE NETWORK. In *First International Conference on Intelligent Systems in Process Engineering: Proceedings of the Conference Held at Snowmass, Colorado, July 9-14, 1995* (p. 355). American Institute of Chemical Engineers.

LAMP (software bundle) - Wikipedia, the free encyclopedia. (n.d.). . Retrieved March 29, 2009, from http://en.wikipedia.org/wiki/LAMP_(software_bundle).

OLE for process control. (n.d.). .

Park, J., Yoon, S. J., & Lee, H. (2003). Effect of Steric Hindrance on Carbon Dioxide Absorption into New Amine Solutions: Thermodynamic and Spectroscopic Verification through Solubility and NMR Analysis. *Environmental Science & Technology*, *37*(8), 1670-1675. doi: 10.1021/es0260519.

Riemer, P. (1996). Greenhouse gas mitigation technologies, an overview of the CO_2 capture, storage and future activities of the IEA Greenhouse Gas R&D programme. *Energy Conversion and Management*, *37*(6-8), 665-670. doi: 10.1016/0196-8904(95)00237-5.

Szladow, A. J., Ltd, L. T., & Mills, D. (1996). INTELLIGENT SYSTEMS IN HEAVY INDUSTRY. In *First International Conference on Intelligent Systems in Process Engineering: Proceedings of the Conference Held at Snowmass, Colorado, July 9-14, 1995* (p. 237). American Institute of Chemical Engineers.

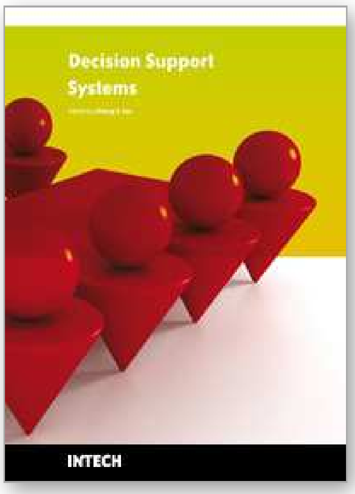
The OPC Foundation - Dedicated to Interoperability in Automation. (n.d.). . Retrieved March 27, 2009, from http://www.opcfoundation.org/.

Tony Liu, D., & William Xu, X. (2001). A review of web-based product data management systems. *Computers in Industry*, *44*(3), 251-262. doi: 10.1016/S0166-3615(01)00072-0.

Turban, E. (1993). *Decision Support and Expert Systems: Management Support Systems* (p. 960). Prentice Hall PTR. Retrieved March 27, 2009, from http://portal.acm.org/citation.cfm?id=541815.

White, C. M., Strazisar, B. R., Granite, E. J., Hoffman, J. S., & Pennline, H. W. (2003). Separation and capture of CO_2 from large stationary sources and sequestration in geological formations--coalbeds and deep saline aquifers. *Journal of the Air & Waste Management Association (1995)*, *53*(6), 645-715.

Wu, Y., & Chan, C. W. (2009). A data analysis decision support system for the carbon dioxide capture process. *Expert Systems with Applications*, 36(6), 9949-9960. doi: 10.1016/j.eswa.2009.01.064.



Decision Support Systems

Edited by Chiang S. Jao

ISBN 978-953-7619-64-0

Hard cover, 406 pages

Publisher InTech

Published online 01, January, 2010

Published in print edition January, 2010

Decision support systems (DSS) have evolved over the past four decades from theoretical concepts into real world computerized applications. DSS architecture contains three key components: knowledge base, computerized model, and user interface. DSS simulate cognitive decision-making functions of humans based on artificial intelligence methodologies (including expert systems, data mining, machine learning, connectionism, logistical reasoning, etc.) in order to perform decision support functions. The applications of DSS cover many domains, ranging from aviation monitoring, transportation safety, clinical diagnosis, weather forecast, business management to internet search strategy. By combining knowledge bases with inference rules, DSS are able to provide suggestions to end users to improve decisions and outcomes. This book is written as a textbook so that it can be used in formal courses examining decision support systems. It may be used by both undergraduate and graduate students from diverse computer-related fields. It will also be of value to established professionals as a text for self-study or for reference.