Extended Kepler Grid-based System for Diabetes Study Workspace

Fawaz AL Hazemi*, Chan-Hyun Youn*
*Dept. of Electrical Engineering, KAIST
Email: {fawaz,chyoun}@kaist.ac.kr

Abstract

Chronic disease is linked to patient's' lifestyle. Therefore, doctor has to monitor his/her patient over time. This may involve reviewing many reports, finding any changes, and modifying several treatments. One solution to optimize the burden is using a visualizing tool over time such as a timeline-based visualization tool where all reports and medicine are integrated in a problem centric and time-based style to enable the doctor to predict and adjust the treatment plan. This solution was proposed by Bui et. al. [2] to observe the medical history of a patient. However, there was limitation of studying the diabetes patient's history to find out what was the cause of the current development in patient's condition; moreover what would be the prediction of current implication in one of the diabetes' related factors (such as fat, cholesterol, or potassium). In this paper, we propose a Grid-based Interactive Diabetes System (GIDS) to support bioinformatics analysis application for diabetes diseases. GIDS used an agglomerative clustering algorithm as clustering correlation algorithm as primary algorithm to focus medical researcher in the findings to predict the implication of the undertaken diabetes patient. The algorithm was Chronological Clustering proposed by P. Legendre [11] [12].

1. Introduction and Background

One of the crucial chronic diseases is diabetes, which is taking attention of recent development in e-Healthcare system. The development of a diabetes patient condition from normal diabetes to diabetes type 2 is becoming one of the vital transitions of a disease condition on the global healthcare.

Prior giving any decision related to a diabetes patient, a medical doctor must screen varies types of the patient's historical data, starting from blood sugar, fat, and cholesterol to HBA1c and Heart ECG; and could be more [13]. Despite the fact that medical doctors have the essential analysis processes of any type of chronic diseases, it is (assumed) very hard to consolidate the analysis of all types of medical reports at once without help from automated machines. For example, it is possible to review the trend of blood sugar over period of time but it begins to be difficult when it is correlated with fat or cholesterol to find out which one was the real cause. Another example, not all historical diabetes patient's data are collected at the same time, for instance blood sugar measurements are taken in a daily based (which is important for monitoring the patient) while measuring the HBA1c is normally checked every three months (which is the vital sign of diabetes condition of a patient)[13]. These difficulties are important to be check accurately during medical doctor's screening especially when the patient is using different types of treatments (diet or using drugs) to control his/her body condition.

Many proposed solutions tickled chronic diseases visualization [2][3]. In[2], the proposed solution was based on problem centric visualization over the time. The solution was good for finding the syndrome and adjusting the treatment plan according to the observation. Unlikely, [3] was another research that used order-entry technique to keep tracking the order-entry. Apparently, none of these solutions showed concerns on backend computation developments. On the other hand, there were several research solutions which focused on the involvement of the backend computation developments to improve the services such as in[1] [5]. [5] is one of the Grid platform solutions that supporting radiologists to diagnosis cancers with mammography. And[1] was a Grid based platform that utilized the real-time image processing of Magnetic Resonance Imaging (MRI) intra-operatively. The focus of this research was to achieve a very short time of image processing and feedback to doctors in the operation surgery, therefore a load balancing and fault tolerance were considered in the development.

However, diabetes disease has many computer aided diagnosis tools (CAD), the huge amount of historical data for the diabetes patient was necessary to be processed all at once with different CAD tools and into the same workspace. This was important to assure analysis all targeted data over a period of time; and this could be achieved by content relation enforcement policy defined by the medical researcher.

Our proposed solution was a Grid-based Interactive Diabetes System (GIDS) which was a system to automate the analysis of diabetes patients' conditions. It was providing an optimized timeline view about patient's condition based on agglomerative clustering algorithms which depends mainly on the enforced relation polices by the medical doctor to correlate patient's medical reports contents. GIDS used the benefits of the Extended Kepler Profile to facilitate the implementation of the agglomerative clustering algorithm. Finally, the results of the analysis would be displayed to medical researcher as trends over time. These trends were representing the relations among patient's medical data.

2. Grid-based Interactive Diabetes System

Grid-based Interactive Diabetes System (GIDS) was a web-enabled Grid-based application that included software, hardware, and middleware systems to support bioinformatics analysis application for diabetes disease. GIDS was to help medical researchers who were interested in finding the relationship between diabetes patient's condition and diabetes' factors such as blood sugar over period of time, which could infer them to predict the implication in patient's condition. To find and measure this type of relationship in time domain, GIDS included a well-known agglomerative clustering algorithm called *Chronological Clustering Algorithm* (introduced by P. Legendre[11][12]) to find the similarity among data over time and demonstrate varies of changes in diabetes patient's condition by controlling the *Chronological Clustering Algorithm* parameters (in particular the alpha and the connectedness). The adoption of the Extended Kepler Profile helped to adjust these parameters smoothly when it was needed without recreate the whole study from scratch.

The main theme of GIDS was the optimized viewer of patient's historical information; which freed medical researcher from the burden of reviewing laboratory tests, blood sugar tests (HBA1c), fat, cholesterol, and others. This work was a challenge task when there were multiple factors for example blood sugar and potassium which were on focus during a specific study on a diabetes patient. GIDS helped to display any interesting changes (based on *Chronological Clustering Algorithm* parameters) in the focused factors into an understandable chart graphs that easily helped in the study.

2.1 GIDS Architecture

The internal architecture of GIDS is illustrated in Figure 1. The GIDS architecture was composed of three main layers. These layers were *Application Layer*, *GIDS Core Layer*, and *Grid Middleware Core Layer*. In addition, there was a *Grid Resources* which acted as computing resources and was not part of GIDS architecture.

2.1.1 Application Layer

This layer was the interface of GIDS which was basically web portal module that enabled GIDS's who were interested in studying diabetes conditions; to compose, run, and review diabetes patients' studies. This layer was a simple website which allows GIDS's user to select which diabetes patient to be undertaken in the study, and to specify the primary elements of that study such as the starting and ending points of a period of time when the diabetes patient where under supervision; and defined the interested diabetes factors for example blood sugar in GIDS analysis; (reminder, all data have to be available in Medical Databases located in Medical Resources Layer as shown in figure 1.)

The application layer was responsible for displaying the outcome of GIDS's computation for GIDS's user's composed study. The display was initially combination of understandable chart graphs, and there was ability for the GIDS's user to review the study by editing the primary elements of the study, which will be explained later in this paper.

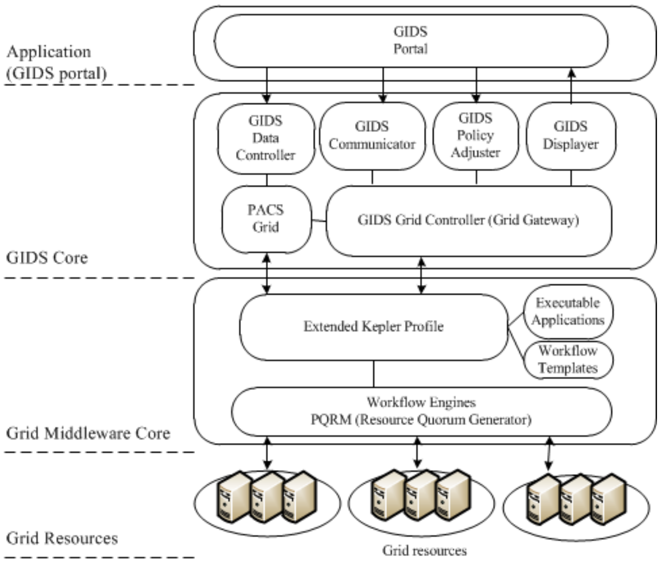


Figure 1 GIDS Architecture

2.1.2 GIDS Core Layer

This layer was the core layer in GIDS which was responsible of the coordination between GIDS's user, medical data, and computation procedure (or workflow process) of bioinformatics analysis application for diabetes within the grid infrastructure. In this layer, there were six modules with different roles; and they were: *GIDS Data Controller* module was an input module from the *Application Layer* boundary (upper bound) which received GIDS's user's inquiries regarding patient's history, and reformatted these inquiries for collecting the required patient's history to *PACS Grid* module [4] as *PACS Grid* requests and these requests considered to be internal communication in *GIDS Core Layer*; *PACS Grid* module was a grid-based application that supported medical data exchange between medical databases, which its design and role in medical data exchange were explained in [4], *PACS Grid* module was adopted as an internal module with bidirectional communication with *Grid Middleware Core Layer* (lower bound) which received internal requests from *GIDS Data Controller* module and, as it had its own *Grid Controller* or Grid gateway, it communicated directly with an *Extended Kepler Profile* module in the *Grid Middleware Core Layer* to retrieve the requested medical data, and the outcome of the retrieved data would be forwarded to *GIDS Grid Controller* module; *GIDS Communicator* module was an input module from the upper bound which was responsible of organizing the annotations made by GIDS's users (medical researchers) such as tags on curious findings, in addition to that, it was responsible for opening, editing, and terminating diabetes study workspaces; *GIDS Policy Adjuster* module is was an input module which was receiving from GIDS's user the adjusting parameters such as the *Chronological Clustering Algorithm* parameters (alpha and connectedness) or selecting the diabetes factors (for example blood sugar) to be on focus in the analysis; *GIDS Displayer* module was an output module from the upper bound which was the responsible module for gathering the outcome of the study from *GIDS Grid Controller* module and displaying them on *GIDS Portal*; and the last but the most important module was *GIDS Grid Controller* or *GIDS*

Grid Gateway module which was the interpreter between the modules located in this layer (except *PACS Grid* module) and *Grid Middleware Core Layer*, it collected the specified study with its data, passed them to *Extended Kepler Profile* module at lower layer (*Grid Middleware Core Layer.*)

2.1.3 Grid Middleware Core Layer

This layer was fundamental grid middleware layer in any grid-based infrastructure. In this layer, there were four basic parts that were needed to support *GIDS Core Layer*, and they were the following: *Extended Kepler Profile*, *Workflow Engines*, *Workflow Templates*, and *Executable Application*. Both *Workflow Templates* and *Executable Applications* were simply pre-stored files at *Grid Middleware Core Layer*. *Workflow Engines* could be any grid-based workflow engine but in this work Kepler[9] was used to communicate with Condor-based[6] grid resources (more details will be provide in the experiment section in this paper). We added the *Extended Kepler Profile* module in this layer to manage the communication occurred between the grid computing systems' *Workflow Engines* if there were multiple *Workflow Engines* working at the same time such as Pegasus[8], DAGMan [7].

2.1.4 Grid Resources

This layer was the computing layer where the Grid computing resources located. It contained the physical computing resources which were computing and analysis diabetes patient's history as specified in the composed study by GIDS's users. The computation would be done based on predefined workflow templates managed by *Workflow Engines*.

3. Evaluation and Results

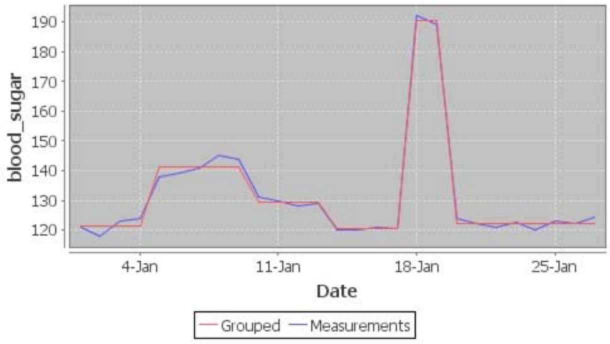
GIDS was tested on small size computer lab to evaluate its usability and functionality. The lab was consisted of three physical computers each was 3.2 GHz CPU; and one laptop used as user's PC to access *GIDS Portal* website. *GIDS* system was installed on one computer; and Kepler was installed on the same computer. Because there was a need of having multiple of computers work as Condor-based Grid resources, Rocks Cluster[10] was installed on the other two computers as one master node and one slave node. Three virtual machines were created on top of Rocks Cluster, and each with Condor enabled.

A diabetes study workspace was created for anonymous patient, who was under supervision for 27 days starting from 1st of January, 2010; a blood sugar test was taken daily. Two adjustments were applied on the study to examine the usability of the system to help medical researches. First, a study workspace was created; then an adjustment was applied on the *Chronological Clustering Algorithm*; after that a change was applied on time domain of the supervision period.

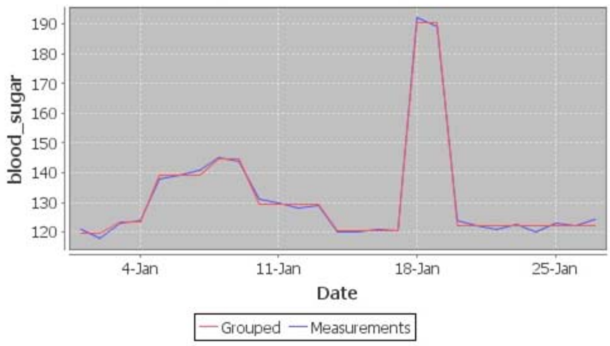
At first, the initial configuration of *Chronological Clustering Algorithm* was kept to observe an initial outcome from the study for the whole supervision period (27 days) that was applying the initial configuration for the full period of study. Then, an outcome graph was shown as in figure 2.a. The graph was containing two lines namely *Grouped* (red) and *Measurements* (blue). Both lines demonstrated the patient's blood sugar during the supervision period. The *Measurements* (blue) line was showing that the blood sugar was changing in a daily base in the range of 118~192 mg/dl. On the other hand, the *Grouped* (red) line was compressing those daily changes of the blood sugar into straight lines by grouping the slight changes in blood sugar over connected period of time within the supervision period.

After that, the feature of *GIDS* that allow user to re-edit the study was checked. The *Chronological Clustering Algorithm* parameter alpha was reset from 0.1 to 0.25, and the study was recomputed again. The outcome of the recomputed study was displayed back in *GIDS Portal* as in figure 2.b. As it was shown in the first part of the experiment, that the blood sugar was grouped (red line) in figure 2.a, in this part of the experiment the blood sugar was grouped (red) but in smaller period of times compared to the first experiment, and this change came from the behavior of *Chronological Clustering Algorithm* parameters. That is, if the value of alpha was increased, the view of the blood sugar measurements presented in the computed grouped (red) line would be much similar to the original view of the daily measured blood sugar presented in the blue line.

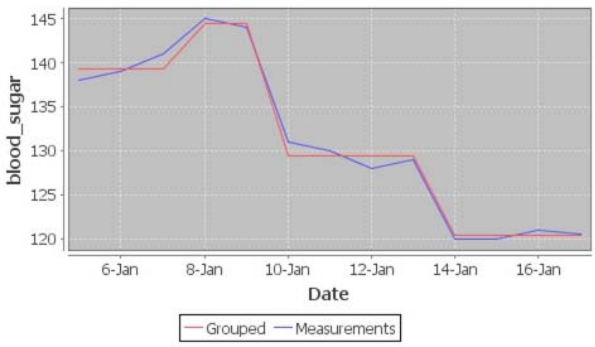
Next, to observe other functionality in *GIDS* which is the zooming in over the time domain, a change of starting and ending points in time for the supervision period was forced in the study. That was done by reducing the full view of the supervision period to be focused on what happened from 5th to 17th of January, 2010. An outcome of the results was shown as in figure 2.c. This focusing was clearly shown that the grouped (red) line; which represented the optimized view of blood sugar changes over the newly selected period of time; was providing netted view of blood sugar measurements than the measurements (blue) line.



(a)



(b)



(c)

Figure 2 GIDS study's outcomes for different adjustments in the study for a diabetes patient's blood sugar measurements over 27 days: (a) the outcome of the initial study for the full period, (b) the outcome of the readjusted chronological clustering alpha variable to 0.25, (c) the outcome of the zooming in smaller period of the supervision over the time domain.

4. Conclusion

Grid-based Interactive Diabetes System (GIDS) is a web-enabled Grid-based application that main goal was to assist medical researchers; who are interested in studying diabetes disease; to have a broader view of diabetes patient's condition and predict the future implication based on current developments in some diabetes factors such as blood sugar, fat, or potassium. GIDS was using base algorithm called *Chronological Clustering Algorithm* for analysis diabetes records. GIDS was tested to study a basic history of a diabetes patient who was under supervision for less than a month. The test was performed to check two functions provided by GIDS which are changing the basic algorithm that GIDS used (*Chronological Clustering Algorithm*) and changing the full view of the supervision period in the time domain in the study.

References

[1] N. Chrisochoides, A. Fedorov, A. Kot, N. Archip, P. Black, O. Clatz, A. Golby, R. Kikinis, and S. Warfield, "Toward Real-Time Image Guided Neurosurgery Using Distributed and Grid Computing," Proceedings of the 2006 ACM/IEEE, SC\06 Conference (SC'06.)

[2] Bui, A A.T.; Aberle, D.R.; Hooshang Kangarloo, "TimeLine: Visualizing Integerated Patient Records," *Information Technology in Biomedicine, IEEE Transactions*,vol. 11, no. 4, pp. 462-473, July 2007.

[3] Toyoda, S.; Niki, N.; Nishitani, H., "SAKURA-Viewer: Intelligent Order History Viewer Based on Two-Viewerpoint Architecture," *Information Technology in Biomedicine, IEEE Transactions*, vol. 11, no. 2, pp. 141-152, March 2007.

[4] Yong-Jie; Chan-Hyun Youn; Hyewon Song; Byoung-Jin Kim; Youngjoo Han, "A PACS-Grid for Advanced Medical Services based on PQRM," *Intelligent Sensors, Sensor Networks and Information, 2007. ISSniP 2007. 3rd International Conference*, pp. 625-630, 3-6 Dec. 2007.

[5] S. R. Amedolia, F. Estrella, "MammoGrid: A Service Oriented Architecture based Medical Grid Application," Proceedings of the 3rd International Conference on Grid and Cooperative Computing. Wuhan. China, 2004.

[6] Douglas Thain, Todd Tannenbaum, and Miron Livny, "Distributed Computing in Practice: The Condor Experience" *Concurrency and Computation: Practice and Experience*, Vol. 17, No. 2-4, pages 323-356, February-April, 2005.

[7] Peter Couvares, Tevik Kosar, Alain Roy, Jeff Weber and Kent Wenger, "Workflow in Condor", in *In Workflows for e-Science*, Editors: I.Taylor, E.Deelman, D.Gannon, M.Shields, Springer Press, January 2007 (ISBN: 1-84628-519-4)

[8] Ewa Deelman, James Blythe, Yolanda Gil, Carl Kesselman, Gaurang Mehta, Sonal Patil, Mei-Hui Su, Karan Vahi, Miron Livny, "Pegasus : Mapping Scientific Workflows onto the Grid ," Across Grids Conference 2004, Nicosia, Cyprus.

[9] Altintas and C. Berkley and E. Jaeger and M. Jones and B. Ludascher and S. Mock, "Kepler: an extensible system for design and execution of scientific workflows" (2004) Scientific and Statistical Database Management, 2004. Proceedings. 16th International Conference on. Pages: 423--424.

[10] Federico D. Sacerdoti, Sandeep Chandra, and Karan Bhatia, "Grid Systems Deployment & Management Using Rocks", September 2004, IEEE International Conference on Cluster Computing, San Diego.

[11] P. Legendre, S. Dallot, and L. Legendre, "Succession of Species within A Community: Chronological Clustering, with Applications to Marine and Freshwater Zooplankton." *Am. Nat.I 1985. Vol. 125, pp. 257-288..*

[12] M. Bell and P. Legendre, "Multicharacter Chronological Clustering in A Sequence of Fossil Sticklebacks", *Syst. Zool., Vol. 36 (1), pp. 52-61, 1987.*

[13] Standards of Medical Care in Diabetes – 2007 by the American Diabetes Associantion, *DIAbETES CARe, VOL. 30, pp. S4-S41, Supplement 1, January 2007.*