การบูรณาการการเฝ้าติดตามน้ำในบรรยากาศจากเรดาร์ตรวจอากาศ โดยใช้ไทยกริด

Integrated Precipitation Monitoring from Weather Radar using Thai Grid

แซลลี่ อี โกลดิน และ เคิร์ท ที รูเดิล

Sally E. Goldin and Kurt T. Rudahl

Geoinformatics Laboratory at KMUTT (GlaK)

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

sally@cpe.kmutt.ac.th และ kurt@cpe.kmutt.ac.th

บทคัดย่อ

การเฝ้าติดตามและตรวจวัดน้ำในบรรยากาศอย่างต่อเนื่องเป็นสิ่งที่สำคัญอย่างยิ่งสำหรับประเทศไทย ความแม่นยำของข้อมูลตำแหน่งและความหนาแน่นของการเกิดฝนตกจำเป็นสำหรับการคาดคะเนและรับผิดชอบต่อภาวะความแห้งแล้งและอุทกภัย หรือการคาดคะเนการเกิดดินถล่ม หรือเป็นแนวทางสำหรับกิจกรรมการกำเนิดเมฆฝน

ประเทศไทยมีสถานีเรดาร์ตรวจอากาศติดตั้งอยู่ 20 สถานีโดยประมาณทั่วประเทศ อย่างไรก็ตามยังเป็นการยากที่บูรณาการข้อมูลของแต่ละพื้นที่มาเป็นข้อมูลในมุมมองทั้งประเทศ เมื่อฐานเรดาร์มีรัศมีครอบคลุม 240 กิโลเมตรโดยรอบสถานี และแต่ละสถานียังผลิตแผนผังข้อมูลต่างชนิดและต่างมาตราส่วนทั้งในเชิงภูมิศาสตร์และในการวัดเชิงปริมาณอีกด้วย

รายงานนี้แสดงการทำงานโดยการใช้การประมวลผลภาพเพื่อทำการแยกการวัดปริมาณความหนาแน่นของฝนจากผลผลิตข้อมูลเชิงภูมิศาสตร์ซึ่งได้จากฐานเรดาร์ การเก็บข้อมูลการวัดเหล่านี้ลงในฐานข้อมูลเชิงภูมิศาตร์ และสังเคราะห์รูปภาพใหม่ซึ่งแสดงข้อมูลน้ำในบรรยากาศทั้งในปัจจุบันหรือในช่วงเวลาที่ผ่านมาแล้วสำหรับพื้นที่ส่วนใดของประเทศไทยก็ตามที่ถูกเลือกขึ้นมา ระบบข้อมูลทางภูมิศาสตร์แบบเฉพาะทางนี้เป็นการคำนวณและการเก็บข้อมูลในปริมาณมาก ดังนี้ระบบจึงถูกออกแบบมาเพื่อใช้ความสามารถในการประมวลผลแบบขนานของกลุ่มคอมพิวเตอร์ของศูนย์ไทยกริดแห่งชาติ

คำสำคัญ : เรดาร์ตรวจอากาศ, การเฝ้าติดตามน้ำในบรรยากาศ, การประมวลผลแบบกริด, การบูรณาการข้อมูลทางภูมิศาสตร์

ABSTRACT

Continuous measurement and monitoring of precipitation is very important for Thailand. Accurate, timely information on rainfall location and intensity is necessary to predict and respond to droughts and floods, to predict landslides and to guide cloud seeding activities.

Thailand has approximately twenty weather radar installations in various locations throughout the country. However, it is difficult to integrate regional information into a national perspective, since each radar site covers a radius of only 240 km around the station, and different stations produce plots of different types with different geographic and measurement scales.

This paper reports on our work using image processing to extract rainfall intensity measurements from the graphical products supplied by the radar sites, storing these measurements in a geodatabase, and then synthesizing new images that display current or historical precipitation information for any selected region in Thailand. This specialized geographic information system is both computation- and storage-intensive. Thus we have designed it to use the parallel processing capabilities of the multi-computer Thai National Grid.

KEY WORDS : Weather radar, Precipitation monitoring, Grid computing, Geographic data integration

1. Introduction to the problem

Continuous measurement and monitoring of precipitation is very important for Thailand. Accurate, timely information on rainfall location and intensity is necessary to predict and respond to droughts and floods, both of which are serious problems for the country. This data is also potentially useful for landslide prediction and can be used to guide cloud seeding activities.

Thailand has approximately twenty weather radar installations in various locations throughout the country, operated by the Bureau of Royal Rainmaking and Agricultural Aviation (BRR) and the Thai Meteorological Department (TMD). Each radar site captures information on precipitation approximately every 5 to 15 minutes in a radius of up to 240 kilometers around the station.

The Bureau of Royal Rainmaking and Agricultural Aviation evaluates and disseminates the data gathered by these stations. It maintains a website at *http://www.royalrainmaking.thaigov.net*, which presents frequently updated sample plots of data gathered from various stations as either Plan Position Indicator (PPI) or Constant Altitude Plan Position Indicator (CAPPI) plots, which are conventional data products in this domain.

However, the utility of this data is limited in several ways:

- The software associated with these stations produces only processed graphical results. The raw measurements are not currently available.

- It is difficult to integrate regional information into a national perspective. Each radar installation produces plots for a radius of only 240 km around the station. In addition, different stations produce plots of different types and with different geographic and measurement scales. Thus there is no easy way to assess precipitation on a country-wide basis.
- Since radar data are available in graphical form only, researchers do not have the numerical, historical data they need to create more sophisticated models and decision support systems.

We originally became aware of these problems when we were contacted by a staff member from BRR who needed numerical precipitation data from the weather radar stations for his research. We visited BRR headquarters in Bangkok and one of the radar facilities, in Phimai, to gather information and investigate the parameters of the problem. Apparently the company which sells the radar hardware and software can supply a module that outputs raw data, but this software component is prohibitively expensive.

We undertook this project to provide a less expensive solution to the raw data problem as well as to provide integrated visualizations of precipitation that combine information from multiple radar sites. We have designed and implemented a software system that can automatically extract numerical precipitation readings from the graphical products currently available, store these in a database, and then use the stored information to synthesize new precipitation maps that integrate data from multiple radar stations for any selected region in Thailand.

Our current system is in prototype form and runs on a single server. However, it is designed to be deployed on a grid architecture. The storage and computational demands of an operational system would quickly exceed the resources of a single server, for several reasons.

First, the problem is data-intensive. Processing one image per hour from each site produces approximately 260 images per day. (This varies because sites are not always functional.) A single image can generate as many as 250,000 data points, although most contribute far fewer. In our test database, processing of 811 images generated 20,711,491 records, or an average of about 25,500 points per image. At this point, the database used nearly four Gigabytes of disk space. Using these figures, we estimate that an operational system would add about one Gigabyte of data per day. There is significant potential for improving performance by creating a distributed data base using the facilities of a grid.

Second, both data extraction from the images and data integration to produced regional visualizations are computationally intensive processes. A grid architecture would distribute extraction and integration requests to different grid nodes in order to achieve parallelism and speed response for each user.

2. Introduction to grid computing

Grid computing is distributed computing using networked resources (processors, storage, peripherals) which may be of heterogeneous types. Open standards for communication, management and application control allow computational units with very different specifications to share work in a transparent way. For an end-user, grid computing provides a transparent portal to a set of virtual services which might be physically located anywhere in the grid.

The individual computers linked into a grid are called *nodes*. Grid nodes may have varying capabilities and may be running different operating systems. *Grid middleware* provides facilities for system administrators and programmers to monitor and control the allocation of processing to different nodes

Thailand has been in the forefront of grid computing research and application development. The Thai National Grid Center (TNGC) was established in 2005 by the Software Industry Promotion Agency (SIPA) to develop a Thailand-based grid infrastructure and fund research and development efforts in grid computing, both fundamental technology and applications.

Fourteen institutes throughout the country, including the top universities in each region, have been linked by high-bandwidth network connectivity to create the Thai Grid. Meanwhile the TNGC headquarters at Kasetsart University has built a 200 node cluster (the "TeraGrid") capable of delivering supercomputer-level computational capabilities.

The present work was intended to take advantage of the Thai Grid infrastructure in order to deal with the issues of storage volume and computational load highlighted in the previous section.

3. Architecture and implementation of the Precipitation Radar Analysis System

3.1 Technologies

The Precipitation Radar Analysis System (PRAS) software was developed at the Geoinformatics Laboratory at KMUTT (GLaK). The software is written in Gnu C and in Perl, with a small amount of JavaScript used in the web-based user interface.

The system makes extensive use of various open source packages to provide necessary functionality, including the following:

- **PostgreSQL:** (***http://www.postgresql.org/***) PostgreSQL is a popular open source relational database management system. We use the PostgreSQL C language library, **libpq** and the **DBI ::Pg** package to read and write data and metadata.
- **Proj4**: (***http://trac.osgeo.org/proj/***) Proj4 is a C language library for transforming geographic coordinates into and out of various projections.
- **Gocr**: (***http://jocr.sourceforge.net/***) gocr is an open-source Optical Character Recognition (OCR) library that we used in the analysis of the source PPI images.
- **Gd**: (***http://www.libgd.org/***) gd is a library for creating image files. The gd API allows the programmer to create a blank image of the desired size and then "draw" into the image, creating lines, circles, rectangles, text, filled areas and so on.
- **Shapelib:** (***http://shapelib.maptools.org/***) Shapelib is an open source library for reading and writing geographic data in the ESRI *shapefile* format.
- **ImageMagick**: (***http://www.imagemagick.org/***) ImageMagick is an open source application for image manipulation that we use to convert the source PPI images from GIF to RGB (24 bit per pixel) image format.

3.2 Core processes

PRAS involves three core processes, which operate asynchronously.

The first process, *data acquisition*, involves gathering radar images from various sites, converting them to a more convenient format, and recording their metadata in a database. In an operational system, this process would be triggered from the radar sites. Periodically, they would "push" PPI images to a grid node, where the images would be pre-processed and recorded in the database. In the PRAS prototype, we simulate this by using a "pull" process which queries the

Bureau of Royal Rainmaking website to retrieve the most current PPI images from each station that is on-line.

The second process, *data extraction*, uses image analysis techniques to locate pixels in the image which represent radar returns and records data values for those locations. Each radar image includes a legend, which indicates the colors associated with different radar values. The data extraction module analyzes the legend for each image in order to determine the correspondence between pixel color and value, which differs across different radar sites. Extracted data values are stored in the database along with their geographic coordinates and metadata such as date, time and source radar station.

The third process, *image synthesis,* generates integrated CAPPI images based on user requests from a web page. The user must select a rectangular region to view (interactively or by specifying latitude/longitude coordinates), a date and time, and an elevation. The process queries the database to locate radar readings that satisfy these criteria. It then assembles a synthetic CAPPI image, adding reference data such as country and changwat boundaries and water features, and displays this image in the web page.

The database serves both as the repository for collected radar data and as a communication mechanism for coordinating these three main processes. The data acquisition process records the availability of new PPI images in the database. The data extraction process examines the database to discover which PPI images have been uploaded but not yet been analyzed. Data extraction processes these images and adds their radar readings to the database. After an image has been processed, the data extraction process updates the database to indicate that fact. Finally, the image synthesis process accesses radar readings as well as contextual geographic features stored in the database, and uses the query results to create an integrated image for a region of interest.

Figure 1 shows the abstract architecture in a single server environment. The process that is probably of most interest to practitioners in the area of geoinformatics is image synthesis. Figure 2 presents a simplified flowchart of this process.

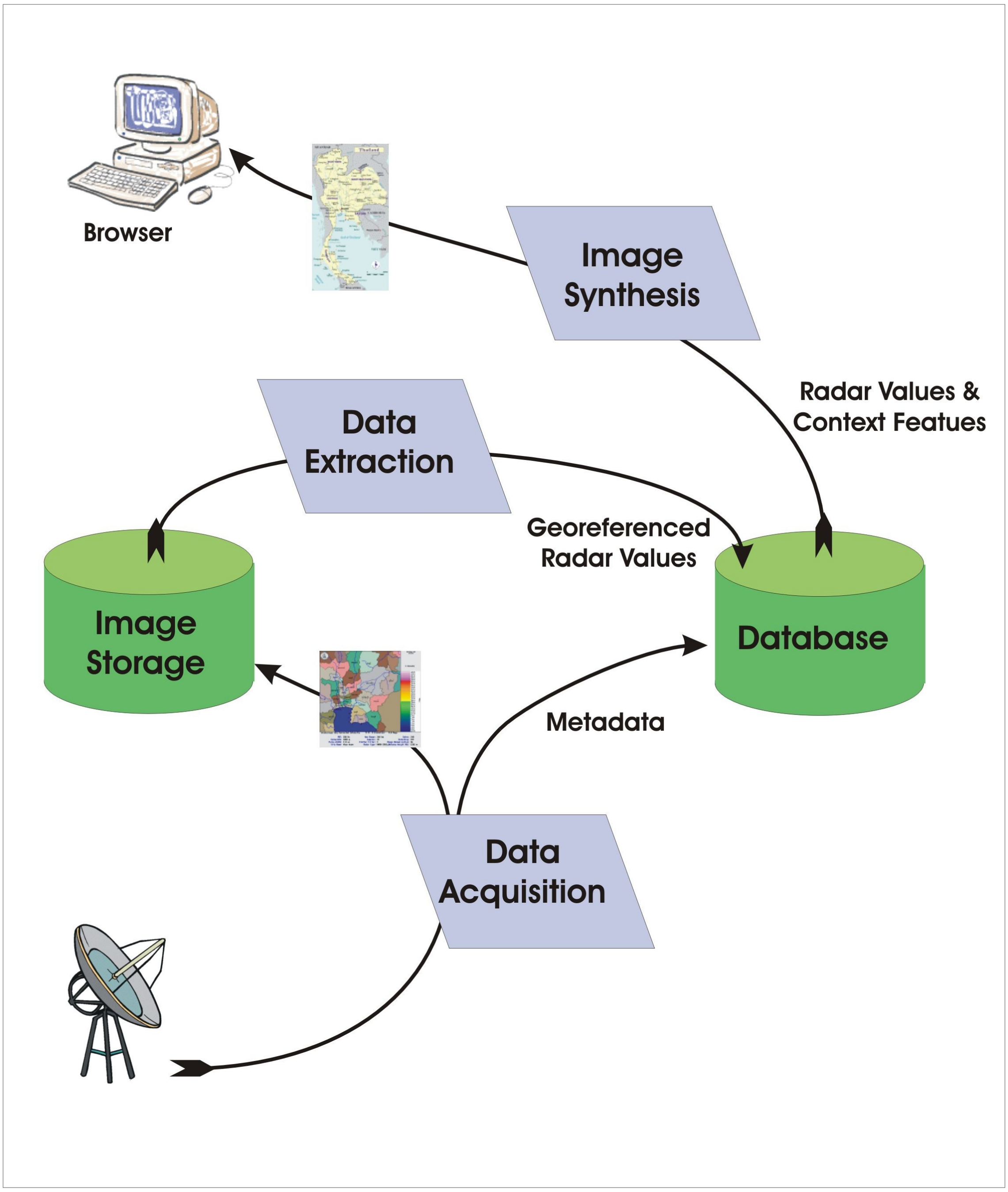

Figure 1. Abstract architecture of PRAS

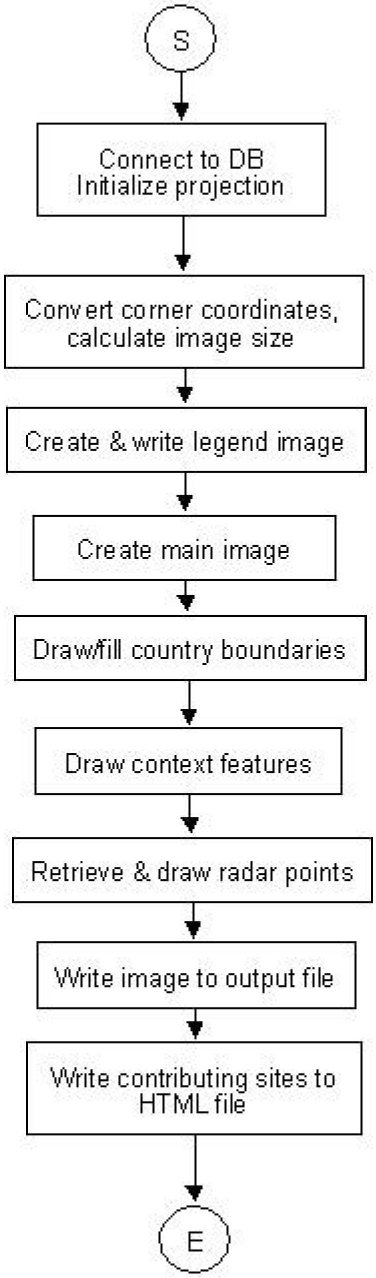

Figure 2. Overview of image synthesis process

When we extract data from the PPI images, we assign each reading a set of geographic coordinates in a custom projection that includes all of Thailand. (We cannot use UTM coordinates because Thailand is covered by more than one UTM zone.) Image synthesis creates a new, integrated precipitation image for all data readings whose coordinates are within a region of interest (specified by the user in latitude/longitude) and which represent a particular date and time.

The program begins by connecting to the database and initializing the projection for conversions. The projection conversion is required in order to transform the coordinates defining the geographic region to be displayed from latitude/longitude to our custom all-Thailand projection. The program converts the corner coordinates and determines the overall size of the region in meters. Based on the image width, it chooses a scaling factor of 1, 2, or 4, so that the resulting image will fit on the web page where it is to be displayed.

The resolution of the original images is approximately 960 m per pixel. Radar readings in the database are located to the nearest half-kilometer. A very wide region will be scaled by 4, that is, each pixel will represent 2 km. A medium-sized region will be scaled by 2 (each pixel represents 1 km) while the smallest regions will use 1:1 scaling (each pixel represents 500 m). If the resulting image would be smaller than 400 by 400 pixels, the program will report an error, since this is significantly smaller than the range of one radar station (nominally 480 km, or 960 pixels at 1:1 scale, in diameter).

Next, the program creates a legend image file. We use a value range that covers all possible values from all radar sites. This is broader than the range supplied by any single site.

The program then creates a blank in-memory image of the size determined in the earlier step, and begins to write graphics content into that image. First, the program draws country boundaries and fills them appropriately to distinguish Thailand from its neighbors and from the ocean. The coordinates for the boundaries are retrieved from the database and are clipped to fit into the requested region. Next the program retrieves and draws "context features", that is rivers and changwat boundaries.

Next, the program retrieves all the radar data readings that fall into the requested region and which were gathered within one hour of the requested time. If the scale is 1:1, the program simply looks up the appropriate color in the legend table and paints the corresponding pixel in the in-memory image. If the scale is 2:1 or 4:1, it is possible that there could be multiple readings corresponding to one image pixel. In this case, we find the largest reading associated with an output pixel and use that value to determine the color of the output pixel.

Finally, the program writes the in-memory image to a PNG format file. It also writes an HTML file which contains a list of the three letter codes for stations who contributed readings to the image. The web-based user interface displays this file in a frame to give the user more information about the result being displayed.

3.3 Grid-based architecture

Figure 3 illustrates how PRAS architecture has been mapped to a grid platform.

In a multi-computer grid or cluster environment, the data extraction and image synthesis processes are distributed to grid nodes as they are triggered. The executive for data extraction assigns nodes on a random basis, selecting one node from those currently on-line. The back-end script for servicing web requests uses a similar strategy. A more sophisticated load balancing technique might be necessary if the web page transaction rate were high, but for a prototype, this simple algorithm appears to be adequate.

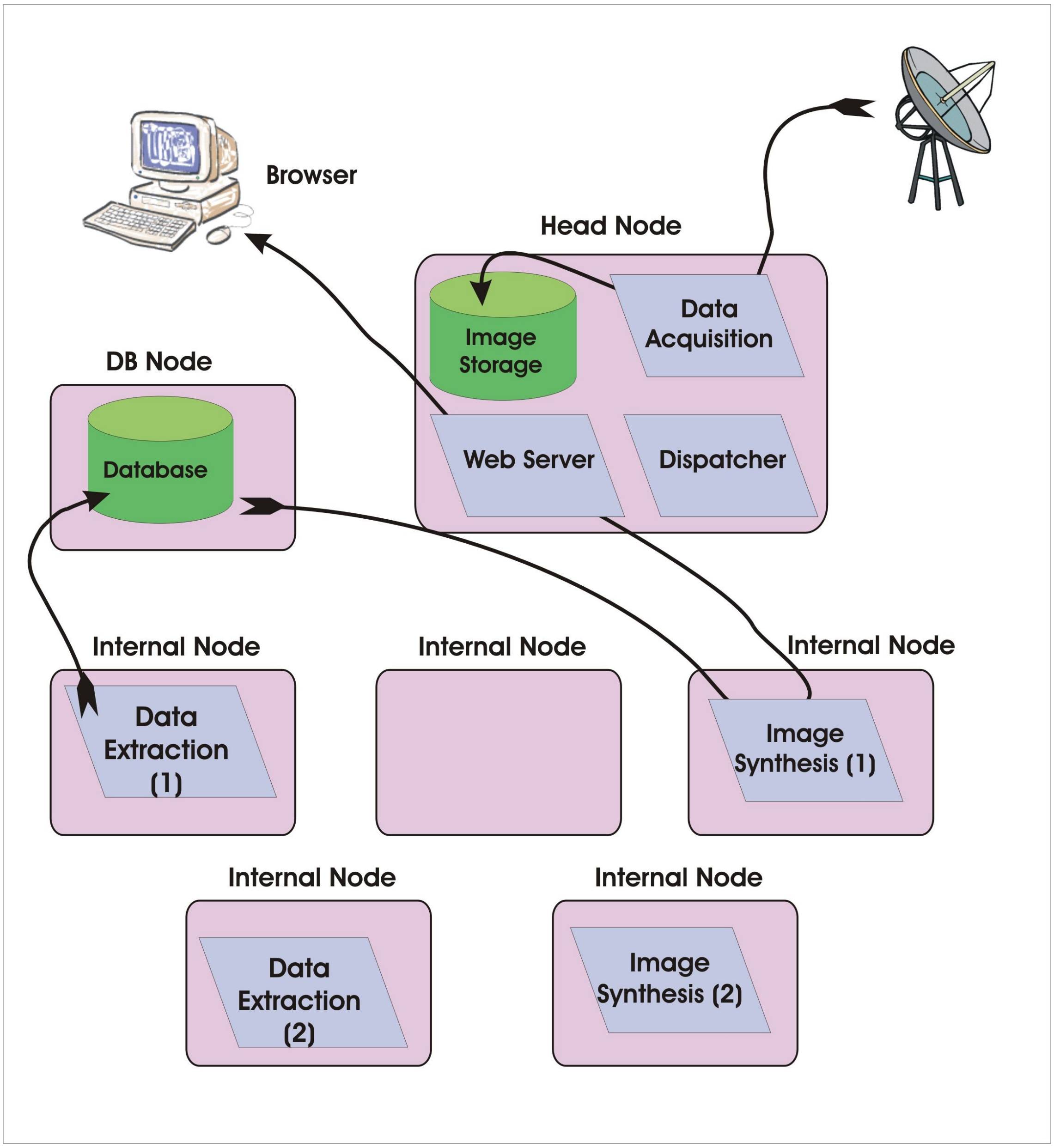

Figure 3. PRAS architecture in a grid environment

The data acquisition process currently resides on the head node of the Thai National Grid Center TeraGrid cluster. This process needs access to the public Internet in order to pull images from the BRR site. Only the head node can do such network access. In an operational system, we assume that each radar site would send images to the head node of one of the sub-clusters comprising the Thai National Grid. In this case, there would be multiple copies of the data extraction executive, one on each sub-cluster, resulting in a broader distribution of processing load.

In the current implementation, there is a single database, which resides on a dedicated grid node. It would be possible to create multiple distributed databases located on the different sub-clusters. A distributed database strategy would have the advantage of reducing disk space requirements on any one sub-cluster, but with the possible costs of increased processing complexity and/or execution time.

There are two possible ways to distribute the data storage, geographically or temporally. In the first case, each database would maintain data for a subset of the radar stations. If radar images were being uploaded to geographically-distributed sub-clusters, geographic distribution of databases would make sense. However, this strategy would significantly increase the complexity of the image synthesis process, since this process would need to relay data requests to the other sub-clusters and then receive the requested data in response. In order to synthesize an integrated image, all relevant data readings, from all radar stations, must be available to the image generation program.

The temporal distribution strategy would group data by time. All readings extracted during one twenty-four hour period would be stored in a single database. Readings from other time periods might be in a different database. This arrangement would not significantly complicate image synthesis, since all the readings for a particular image represent the same time period. However, additional control mechanisms would be needed during data extraction to determine where data should be stored, and during image synthesis to decide which data base should be queried.

4. Results

We have successfully implemented and deployed PRAS on a single server. Figure 4 shows an example of an integrated precipitation images generated by the Precipitation Radar Analysis System.

The prototype version of PRAS is publicly available at

http://www.cpe.kmutt.ac.th/glak/radarproject/demo.html **.**

This URL provides instructions plus a link to the application. The prototype allows the user to request integrated images for anywhere in Thailand. However, the software for gathering new images (data acquisition and data extraction) has been turned off because we do not have adequate disk space to handle the data involved. The database available holds readings from only two dates. However, it allows users to experiment with the application.

We have begun the work of deploying the software to the Thai National Grid Center TeraGrid cluster but this work has not been completed due to the limits of our project funding.

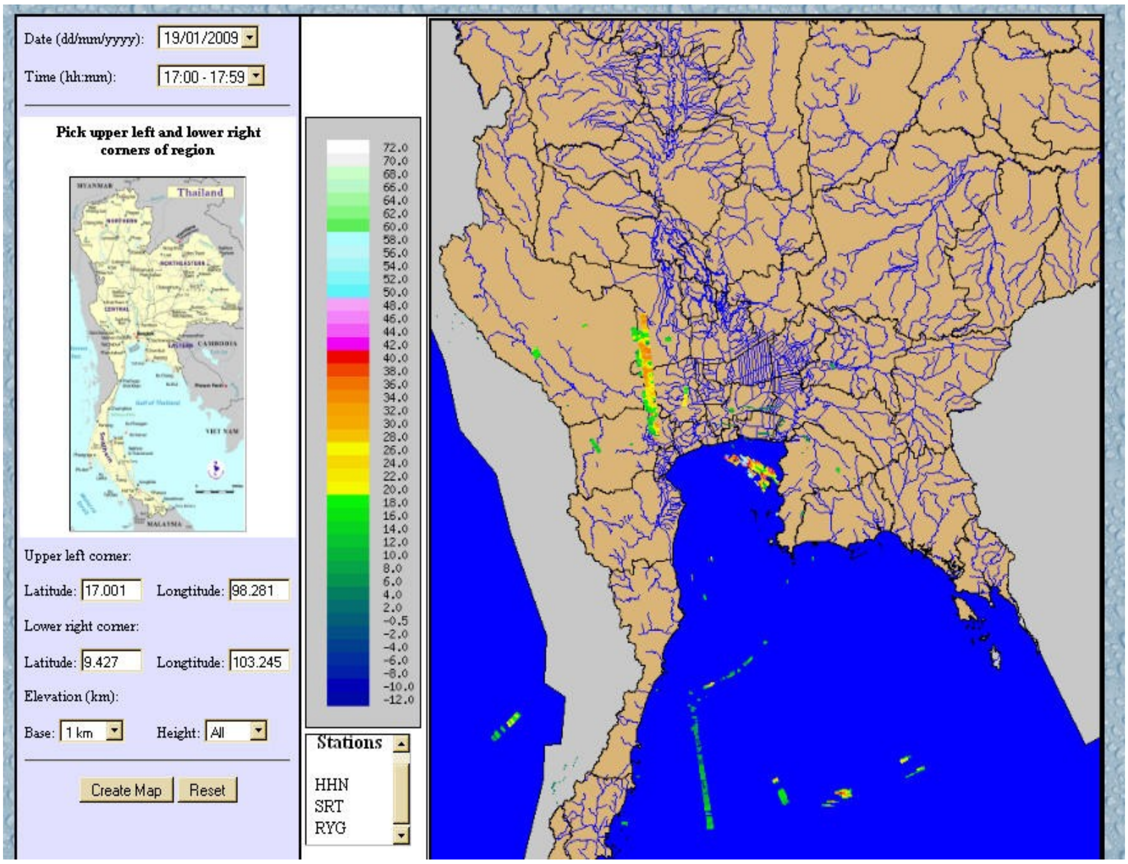

Figure 4. Integrated precipitation image created by PRAS

5. Conclusions and future work

We are encouraged by our success in extracting numerical precipitation data from the graphical products currently created at Thailand's weather radar stations. However, significantly more work remains to be done.

- We need to implement a data distribution strategy to resolve the issue of the huge data volume generated by this application.
- Our "pull" strategy for acquiring data is not very robust since it depends on the continued existence of BRR's radar web page. In fact it appears that since we completed the first phase of this work, BRR has changed its web site and the data we need are no longer available.
- Our application cannot process the PPI images created by all radar stations. The images produced by some stations do not include an integrated legend. Thus we have no key for assigning data values based on pixel color.
- Given that researchers at BRR would like to use the raw data for use in modeling and forecasting, we need to implement simple data search and extraction tools that will create output files in convenient formats.

We plan to seek additional funding from the relevant agencies in order to move this work from the prototype phase into an operational phase. An operational Precipitation Radar Analysis System will provide an economical solution to the limitations of existing radar station software, support new research, and assist Thailand in dealing with urgent precipitation-related problems such as drought, floods, and landslides.

Acknowledgments

This work was supported by a research grant from the Software Industry Promotion Agency via the Thai National Grid Center.

We also want to thank Dr. Prasert Aungsuratana from the Bureau of Royal Rainmaking and Agricultural Aviation for his assistance in clarifying the problem and for making it possible for us to visit the weather radar station in Phimai.