Extracting Storm-Centric Characteristics from Raw Rainfall Data for Storm Analysis and Mining

Kulsawasd Jitkajornwanich
Computer Science and Engineering Department
Univ. of Texas at Arlington
P.O. Box 19015
Arlington, TX 76019
Tel. +1 682 429 5559
kulsawasdj@hotmail.com

Ramez Elmasri
Computer Science and Engineering Department
Univ. of Texas at Arlington
P.O. Box 19015
Arlington, TX 76019
Tel. +1 817 272 0067
elmasri@cse.uta.edu

John McEnery
Department of Civil Engineering
Univ. of Texas at Arlington
P.O. Box 19308
Arlington, TX 76019
Tel. +1 817 272 0234
mcenery@uta.edu

Chengkai Li
Computer Science and Engineering Department
Univ. of Texas at Arlington
P.O. Box 19015
Arlington, TX 76019
Tel. +1 817 272 0162
cli@cse.uta.edu

ABSTRACT
Most rainfall data is stored in formats that are not easy to analyze and mine. In these formats, the amount of data is enormous. In this paper, we propose techniques to summarize the raw rainfall data into a model that facilitates storm analysis and mining, and reduces the data size. The result is to convert raw rainfall data into meaningful storm-centric data, which is then stored in a relational database for easy analysis and mining. The size of the storm data is less than 1% of the size of the raw data. We can determine the spatio-temporal characteristics of a storm, such as how big a storm is, how many sites are covered, and what is its overall depth (precipitation) and duration. We present formal definitions for the storm-related concepts that are needed in our data conversion. Then we describe storm identification algorithms based on these concepts. Our storm identification algorithms analyze precipitation values of adjacent sites within the period of time that covers the whole storm and combines them together to identify the overall storm characteristics.

Categories and Subject Descriptors
H.2.8 [Database Management]: Database Applications – spatial databases and GIS, scientific databases; J.2 [Computer Applications]: Physical Sciences and Engineering – earth and atmospheric sciences

General Terms
Design, Algorithms

Keywords
Storm analysis, rainfall, precipitation, CUAHSI, ODM

1. INTRODUCTION
Most rainfall data is stored in formats that are not easy to analyze and mine. In these formats, the amount of data is enormous. In this paper, we propose techniques to summarize the raw rainfall data into a model that facilitates storm analysis and mining, and reduces the data size. The result is to convert raw rainfall data into meaningful storm-centric spatio-temporal data, which is then stored in a relational database for easy analysis and mining. The size of the storm data is less than 1% of the size of the raw data.

Previous storm analysis has mainly been location-specific (either site-specific or region-specific) [1, 2, 3, 4], meaning that each location is considered independently when analyzing a storm. An example would be determining how many storms occurred at site location 376501 in the year 2011. But in reality, a storm covers many locations over a period of time, so location-specific analysis is insufficient. In this work, we propose a climatic application framework that analyzes rainfall data in a storm-specific way. We consider all the locations over time for each storm, so we can determine storm-specific characteristics such as how big the storm is, how many sites are covered, and what is its overall depth and duration. Analyzing the whole storm can give more insight and information since it reflects how a storm actually behaves in nature. In particular, a storm can start at one location and end at another, and the storm typically covers multiple locations at each time point.

It is very difficult to analyze storms directly from the raw data for several reasons. First, the quantity of data is very large that it qualifies as big geospatial data [19, 20]. Second, the data is stored in a manner that makes it difficult to identify the storms. The data has been gathered as frequently as every five minutes and covers a huge area of observation fields. Traditionally, the data is recorded and stored in either printed or file/folder format. As a result, attempting to do storm analysis with such a large amount of data and the traditional way of storing the data will require manually combining all data across an enormous number of folders and processing them together. This makes it nearly impossible to do storm analysis. It was also documented that converting precipitation data from printed format into digital format (in file/folder), can take up to three years according to a TxDOT project [5].

Our framework allows big precipitation data to be analyzed using relational database by preprocessing, filtering, and discarding unimportant rainfall data and preserving only meaningful storm data that we are interested in. The framework provides a storm-specific approach for analyzing rainfall data by first formalizing storm-related concepts. We then incorporate hydrology concepts and design a customized database schema for storing storm data. This maximizes the ease for hydrologists to perform storm-
specific analysis and at the same time, follows the global standard of hydrological data model called CUAHSI ODM [10, 12, 23]. Algorithms are developed to identify the different types of storms as described in the formalization and store them using the proposed database schema. Scientists can perform custom storm analysis as needed by using (1) a standard SQL for non-visual (scalar) data analysis such as total rainfall of a storm or (2) our Storm Visualization component for visual (vector) data analysis, which cannot be determined by numerical results in the tables such as storm movement. In addition, data mining techniques such as clustering and classification can be used in storm analysis. Figure 1 is an overview of our framework.

The reduction in number of records allows faster querying and mining of the storm data. In our experiments, the number of storm records is less than 1% of the number of raw data records. In addition, each storm record summarizes the hydrological characteristics of one overall storm. Additional characteristics are available through the related sub storm records.

Since our framework is designed by keeping CUAHSI ODM in mind, it can then be adjusted to apply to other kinds of hydrological observation types as long as they are using the same CUAHSI ODM model, such as soil moisture or river gauge data [9, 10, 12, 23]. In addition, our framework is also backward-compatible with the original location-specific analysis of storms. It can still be used for location-specific storm analysis. That means, our framework can capture the complete spatio-temporal dimensions of storm characteristics: both location-specific and storm-specific. We are hoping that our framework can be a useful tool for hydrologists by helping them to analyze and visualize big precipitation data easier and more efficiently.

The organization of the paper is as follows. In section 2, we describe the raw data used for the system. In section 3, we formalize the concepts of local, hourly, and overall storms, design a database schema for storing them, and describe the algorithms for identifying these storms from the raw rainfall data. In section 4, we discuss some examples of the analysis that can be performed. Finally, we discuss related work in section 5.

2. DESCRIPTION OF RAW DATA
Our raw data comes from National Weather Service – West Gulf River Forecast Center (NWS – WGRFC) (NOAA) and contains 16-year (1996 - 2011) historical hourly precipitation data in Texas and some surrounding areas: Colorado, New Mexico, Louisiana, and part of Mexico, as highlighted in Figure 2 [7, 8]. There is a total of 69,830 site locations observed. The data is received hourly and consists of precipitation data for that particular hour for all 69,830 site locations. This means that the number of records inserted per hour, day, month, and year is 69,830, 1,675,920, 50,277,600, and 603,331,200, respectively. Site points are four kilometers apart to north, south, east, and west. The raw data is stored in the database using CUAHSI ODM [10, 23] and has 8.004123763 billion records. However, only 1.25 years of data (October, 2010 – December, 2011) was used in our initial analysis and only Texas was covered. This data has 405,450,691 records of historical hourly precipitation data covering 38,450 sites in Texas. We are now extending the analysis to cover all 16 years of data.

![Figure 1. An overview of our framework](image)

![Figure 2. The coverage of WGRFC observations](image)

CUAHSI (Consortium of Universities for the Advancement of Hydrologic Science, Inc.) [12] is a well-known research organization conducting research in the water science-related area since 2001, supported by National Science Foundation (NSF). Hydrologic observation data is gathered from various organizations and kept in various formats. To eliminate the ambiguities in sharing and interpreting hydrological information, CUAHSI ODM [10] was proposed in 2008 [23]. CUAHSI ODM provides a standard schema to store hydrological data in a relational database. In this paper, only the five main tables of CUAHSI ODM will be briefly discussed: Sources, Sites, Methods, Variables and DataValues tables. Figure 3 shows the star-schema [6, Chapter 29] of the five main tables. Appendix A describes these five tables as they were used in our analysis. For more details, please refer to [10, 23].

3. LOCAL, HOURLY, AND OVERALL STORMS: FORMALIZATION, DATABASE SCHEMA, AND IDENTIFICATION ALGORITHMS
3.1 Formalization of Storms
Before defining our storm-related concepts, we specify some predicates (relationships) and terminology that will be used in the definitions.

![Figure 3. Star-schema of 5 main tables of CUAHSI ODM](image)
The storm overall depth.

Local storm is a set of time points and associated rainfall data at a particular spatial site. Two distinct local storms are separated by at least h consecutive time points with zero precipitation, where h is called the inter-event time [1, 2, 17]. In our case, inter-event time (h) is set to 6 hours. There may be some consecutive time points with zero precipitation within a local storm, as long as it is less than h time points. For any local storm, there will not be a subsequence of h or more consecutive zeroes in the series. A local storm is a site-specific storm, as we see in most hydrology papers [1, 2, 3, 4] that considers each site independently, e.g. determining how many storms occurred at site location 376501 in 2011.

Additional terminology specifically for the local storm definition is as follows:

- storm depth: the amount of precipitation occurring throughout the storm duration at a particular site [16].
- storm intensity: the storm depth divided by the storm duration (inches per hour) [16].

Definition 1. A local storm is represented by \( L(s, T_j) \) where

- \( s = \) site id (We assume that the (x, y) HRAP coordinates [14, 10] of each site are available from a table and we refer to them as x, y.)
- \( T_j = \) the set of (time point, precipitation) that constitutes a local storm at site \( s \) and formalizes the concept that two different local storms at site \( s \) must be apart by at least inter-event time. That is, \( T_j = \{ (t_j, p_{j,i}) | t_j \text{ is time } i \text{ (hourly); } p_{j,i} \text{ or } p_{j,(i+1)} \text{ is precipitation from } t_{j,i} \text{ to } t_{j,i+1} \text{ (inches), } i = 1, 2, \ldots, n; t_{j,1} - t_{j,i} = 1; t_{j,n} = \text{ end time; } p_{j,n} = p_{j,(n+1)} = \ldots = p_{j,(n+h)}, h = 0 \text{ is false, } k = 1, 2, \ldots, (n-h)+1 \text{ and h = inter-event time; } p(t_{j,k-1}) = p(t_{j,k}+1) = p(t_{j,k}+2) = \ldots = p(t_{j,k}+h) = 0; \text{ storm duration } = n; \text{ storm depth } = \sum_{i=1}^{n} p_{j,i}; \text{ storm intensity } = \frac{\text{storm depth}}{\text{storm duration}} \}
- \( L(s, T_j) \)

3.1.2 Hourly Storm

Hourly storm is a set of adjacent sites of local storms at a particular hour. It is built up and has an orthogonal concept to local storm: instead of considering a site location independently we consider a time point (an hour) independently. Local storm fixes one site and covers its data over many time points, whereas hourly storm fixes a time point and covers its data over many sites.

Additional terminology specifically for the hourly storm definition is as follows:

- storm sites total: the total amount of precipitation occurring at a particular hour for the sites of an hourly storm.
- storm average: the average precipitation (per site) for an hourly storm.

Definition 2. An hourly storm is represented by \( H(t, S_j) \) where

- \( t = \) time point (hourly)
- \( S_j = \{ (s_j, p_{j,i}) | s_j \text{ is site id } i; p_{j,i} \text{ or } p_{j,(i+1)} \text{ is precipitation (inches) at site } s_j, i = 1, 2, \ldots, n; \text{ if } |s| = 1 \text{, then it contains a single site with no neighbors. Otherwise, every site in the set must have at least one neighbor that is also in the set. That is, if } |s| > 1 \text{, then a site } s_j \text{ of locations in } S_j \text{ must satisfy: for all } s_j \in S_j \text{ if } \text{neighbors}(s_j, S_{j,k}) \text{, then } s_{j,k} \in S_j; \text{ storm area } = |s|; \text{ storm average } = \sum_{i=1}^{n} p_{j,i}; \text{ storm total } = \sum_{i=1}^{n} p_{j,i}; \text{ storm coverage } = \text{storm total } / \text{storm area} \}

3.1.3 Overall Storm

Overall storm is a storm-specific concept, considering each storm individually as a union of hourly storms. Two neighboring hourly storms must be within a maximum time period \( g \), called the grouping-window, and must share at least \( n \) common sites, called the spatial-window. Grouping-window is the time interval within which storms will be considered to be part of the same storm. Spatial-window is the number of common site(s) shared between two hourly storms. The concept of overall storm covers both spatial and temporal characteristics of the storm.

Additional terminology specifically for the overall storm definition is as follows:

- storm overall depth: the total amount of precipitation occurring throughout the storm duration across the hourly storms.
- storm overall intensity: the storm overall depth divided by the storm duration (inches per hour).
- storm overall average: the average precipitation (per site) for an overall storm.

Definition 3. An overall storm is represented by \( O_j \) where

- \( O_j = \{ H_{j,i} | H_{j,i} \text{ is an hourly storm, } i = 1, 2, \ldots, n; H_{j,i} \text{ is the first hourly storm and } H_{j,1} \text{ is the last hourly storm; } t_{j,k+1} - t_{j,k} = \text{ grouping-window where } t_{j,k} \text{ is the time of hourly storm } H_{j,k} \text{ and } |S_{j,k+1}| \geq \text{ spatial-window where } S_{j,k} \text{ is sites of } H_{j,k}; k = 1, 2, \ldots, n-1; \text{ storm duration } = n; \text{ storm coverage } = |\bigcup_{i=1}^{n} S_{j,i}|; \text{ storm overall depth } = \sum_{i=1}^{n} \text{ storm sites total}(H_{j,i}); \text{ storm overall intensity } = \frac{\text{storm overall depth}}{\text{storm overall duration}}; \text{ storm overall average } = \frac{\text{storm overall duration}}{\text{storm coverage}}; \text{ storm area } = \sum_{a \in Z} \text{area}(a), Z = \bigcup_{i=1}^{n} S_{j,i}, S_{j,i} \text{ is sites of } H_{j,i} \}

The following terms are interchangeable in this paper:

- Event = Storm
- Sub = Hourly
- Main = Overall
3.2 Database Schema for Storms

Our database schema for storms was designed in such a way that the expressivity and usability features of SQL can be fully utilized in the analysis tasks. SQL and relational databases are proven tools in performing analysis [21, 22]. By designing database schema this way, we can preserve the advantages of SQL and at the same time, the big precipitation data can be analyzed in a relational database.

We store our output, which consists of the identified local, hourly, and overall storms, in relational database tables: LocalEvents, SubStorm, MainStorm, respectively. LocalEvents table stores local storms information for all sites. The information includes date, time, and precipitation depth (in inches) of the storm for a particular site. SubStorm table stores all hourly storms information. The information includes storm sites total, storm average, and storm coverage of an hourly storm. MainStorm table stores information of all overall storms consisting of all sites that were covered during each storm, storm overall depth, storm overall average, and storm overall intensity.

Additional tables were also created: LocationProximity and RA_Sites (Rainfall Analysis Sites table). These will allow us to use SQL during hourly storm and overall storm calculation and utilizes the fact that the raw data that we use resides in a relational database [8, 10, 23]. LocationProximity table stores neighboring site information for each site, which will be used along with LocalEvents table to calculate sub storms. Original CUAHSI ODM does not provide neighboring information in any tables. RA_Sites table stores all site information that we are interested in.

There is a total of five tables created, whose schema diagram is shown in Figure 4 [6, Chapter 3].

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Schema Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>LocalEvents</td>
<td>LocalEventID SiteID DateTimeUTC DataValue</td>
</tr>
<tr>
<td>RA Sites</td>
<td>SiteID Longitude Latitude LocalX LocalY County State</td>
</tr>
<tr>
<td>LocationProximity</td>
<td>SiteID SiteNeighborID Direction</td>
</tr>
<tr>
<td>SubStorm</td>
<td>SubStormID LocalEventID SiteID DateTimeUTC DataValue</td>
</tr>
<tr>
<td>MainStorm</td>
<td>MainStormID SubStormID</td>
</tr>
</tbody>
</table>

Figure 4. Schema diagram of the output tables

3.3 Algorithm Development

The algorithms for storm identification are designed by taking hydrology concepts into account and can be divided into 4 modules:

1. Event Separator
2. Location Proximity Creator
3. Sub Storm Identification
4. Main Storm Identification

The data flow among these modules is shown in Figure 5.

3.3.1 Event Separator

This module separates rainfall events (local storms) using 6-hour inter-event time as storm separators. The inter-event time (h) of 6 hours is suggested by Huff [1, 2]. The input for this module is the historical hourly precipitation raw data in Texas from the ODM DataValues table (405,450,691 records). The output will be stored in LocalEvents table. An example of LocalEvents table is shown in Table 2. Since the area of Texas is very large, there is a significant climatic difference in its various regions. As a result, USGS (U.S. Geological Survey) divides Texas into 10 regions based on their climatic and geographic characteristics and proposed a map, called Texas Climatic Regions [4], as seen in Figure 6.

To be consistent with USGS, we analyze each region separately. The experimental result (Table 1) indicates that East Texas has the most storm data whereas Trans-Pecos has the least storm data even though Trans-Pecos has more raw data compared to East Texas. This is consistent with the fact that Trans-Pecos is the driest region and East Texas is one of the wettest regions in Texas [24, 25].

<table>
<thead>
<tr>
<th>Region</th>
<th>Number of Raw Data</th>
<th>Number of Identified Storms</th>
<th>Reduction in Raw Data Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>East Texas</td>
<td>48,953,130</td>
<td>325,504</td>
<td>0.72%</td>
</tr>
<tr>
<td>Edwards Plateau</td>
<td>74,415,532</td>
<td>257,679</td>
<td>0.38%</td>
</tr>
<tr>
<td>Hill Country</td>
<td>31,713,527</td>
<td>97,327</td>
<td>0.34%</td>
</tr>
<tr>
<td>High Plains</td>
<td>24,905,521</td>
<td>89,834</td>
<td>0.39%</td>
</tr>
<tr>
<td>North Central</td>
<td>59,092,957</td>
<td>299,082</td>
<td>0.54%</td>
</tr>
<tr>
<td>South Central</td>
<td>32,120,334</td>
<td>130,083</td>
<td>0.43%</td>
</tr>
<tr>
<td>South Texas</td>
<td>26,091,109</td>
<td>70,580</td>
<td>0.42%</td>
</tr>
<tr>
<td>Lower Valley</td>
<td>11,142,385</td>
<td>41,820</td>
<td>0.42%</td>
</tr>
<tr>
<td>Trans-Pecos</td>
<td>65,136,238</td>
<td>155,453</td>
<td>0.26%</td>
</tr>
<tr>
<td>Upper Coast</td>
<td>22,883,789</td>
<td>127,843</td>
<td>0.68%</td>
</tr>
</tbody>
</table>

Table 1. An experimental result of our analysis
We used threads [26] to improve algorithm performance. For each region, sites are equally partitioned into \( p \) different disjoint subsets. Each subset is then assigned to one thread. The threads run concurrently and then the results are merged to form LocalEvents table. In our case, \( p \) is set to 4, assuming that each thread occupies each of 4 cores of our computer configuration.

To separate rainfall events for a particular site, a parameter, called \textit{inter-event-count}, is maintained to keep track of the number of consecutive zero precipitation (sorted by date and time). We use 6 hours inter-event time, as suggested in [1, 2]. In some situations or applications, a different inter-event time is needed and this can be achieved by changing the parameter in our algorithm to other values such as 24 hours.

### Table 2. An example of LocalEvents table

<table>
<thead>
<tr>
<th>LocalEventID</th>
<th>SiteID</th>
<th>DateTimeUTC</th>
<th>DataValue</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>654321</td>
<td>2012-04-01 09:00</td>
<td>0.2</td>
</tr>
<tr>
<td>2</td>
<td>654321</td>
<td>2012-04-01 10:00</td>
<td>0.9</td>
</tr>
<tr>
<td>3</td>
<td>60000</td>
<td>2012-04-01 09:00</td>
<td>0.8</td>
</tr>
<tr>
<td>4</td>
<td>45321</td>
<td>2012-04-01 10:00</td>
<td>0.1</td>
</tr>
<tr>
<td>5</td>
<td>45321</td>
<td>2012-04-01 11:00</td>
<td>0.6</td>
</tr>
<tr>
<td>6</td>
<td>45321</td>
<td>2012-04-01 10:00</td>
<td>0.3</td>
</tr>
<tr>
<td>7</td>
<td>50000</td>
<td>2012-04-01 10:00</td>
<td>0.8</td>
</tr>
<tr>
<td>8</td>
<td>50000</td>
<td>2012-04-01 11:00</td>
<td>0.5</td>
</tr>
</tbody>
</table>

### 3.3.2 Location Proximity Creator

This module creates the LocationProximity table containing neighboring sites information for each site. The input of this module is site information from ODM Sites table [8, 10]. The output will be stored in LocationProximity table. An example is shown in Figure 7.

We calculate neighboring sites information for each site using the HRAP coordinate information [14] labeled as LocalX and LocalY in the ODM Sites table [8, 10, 23]. A site \((x, y)\) will have the following neighboring sites: \(s_0(x, y + 1), s_0(x, y - 1), s_0(x + 1, y), s_0(x - 1, y), s_0(x + 1, y + 1), s_0(x - 1, y + 1), s_0(x + 1, y - 1), \) and \(s_0(x - 1, y - 1)\) where \((x, y)\) is a HRAP coordinate of site \(s\).

![Figure 7. Neighboring sites of site location 654321](image)

### 3.3.3 Sub Storm Identification

This module identifies hourly storms by finding neighboring sites that have precipitation during the same hour. The input of this module is LocalEvents and LocationProximity tables. The output will be stored in the SubStorm table. An example of SubStorm table is shown in Figure 9 and Table 3.

Based on the definition in section 3.1.2, hourly storm is a set of adjacent sites that contain non-zero rainfall in the same time point (hour). The naïve approach is to include all non-zero precipitation values of local storms at a particular hour to be the same hourly storm. However, practically, this could be too rigid. Unanticipated incidents can happen such as equipment malfunctions (not reporting data) or data misreading (reporting incorrect data) since the equipment (physical gauges) can be degraded over time. To keep accuracy of the system intact with small amounts of errors, we introduce a more “relaxed” approach, which incorporates the following two concepts:

#### 3.3.3.1 Space-Tolerance

Informally, \textit{space tolerance} is to allow non-zero precipitation sites to still be categorized as part of an hourly storm even if they are not in adjacent neighboring sites but are \textit{indirect} neighboring sites within a certain number of intermediate sites.

**Definition 4.** We say that \(b\) is an \textit{i- indirect neighbor} of \(a\) if:

\[
\text{neighbor}(a, x_1), \text{neighbor}(x_1, x_2), \ldots, \text{neighbor}(x_i, b)
\]

That is, when space-tolerance is set to \(n\), the neighbors of site \(a\) will include direct neighbors, as well as all \(i\)-\textit{ indirect neighbors} of \(a\) for \(i = 1, 2, \ldots, n\). Figure 8 compares the naïve approach and space-tolerance approach with \(n = 1\). With the same set of non-zero precipitation values, represented by dots at a particular hour, the naïve approach identifies 2 hourly storms whereas space-tolerance approach identifies 1 hourly storm, which is more practical in reality.

![Figure 8. Comparison between naïve approach and space-tolerance approach](image)

#### 3.3.3.2 Outlier Detection

The space-tolerance mentioned previously only considers if there is a precipitation value at a particular site regardless of whether or not the value is consistent with its neighbors. The space-tolerance concept is good at handling gauge malfunction problems (not reporting data). However, it does not take into account the data values. This outlier detection concept compares the value of precipitation of the current site with its neighbors to detect a potential outlier. This concept will help coping with the data misreading problem. Any outlier detection technique can be implemented. In our example, we only consider eight adjacent neighboring sites when detecting outliers. However, indirect neighboring sites can also be used.

The algorithm is based on recursion and depth-first-search. It checks for each hour to identify how many hourly storms there are, and the sites they cover. To be identified as part of an hourly storm, the sites’ location and precipitation value must satisfy the space-tolerance and outlier detection criteria.

### 3.3.4 Main Storm Identification

This module identifies all overall storms (which consist of hourly storms that are sharing some common site(s) \((\text{spatial-window s})\) within the specified \textit{grouping-window g} hour(s)) and their storm characteristics. The input of this module is the SubStorm table and
Table 3. An example of SubStorm table corresponding to Figure 9

<table>
<thead>
<tr>
<th>SubStormID</th>
<th>LocalEventID</th>
<th>SiteID</th>
<th>DateTimeUTC</th>
<th>DataValue</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>654321</td>
<td>2012-04-01 09:00</td>
<td>0.2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>60000</td>
<td>2012-04-01 09:00</td>
<td>0.3</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>45321</td>
<td>2012-04-01 09:00</td>
<td>0.1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>654321</td>
<td>2012-04-01 10:00</td>
<td>0.9</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>50000</td>
<td>2012-04-01 10:00</td>
<td>0.8</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>45321</td>
<td>2012-04-01 10:00</td>
<td>0.6</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>50000</td>
<td>2012-04-01 11:00</td>
<td>0.5</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>45321</td>
<td>2012-04-01 11:00</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Figure 9. Examples of hourly storms at 9:00, 10:00, and 11:00 on 4/1/2012

The output will be stored in the MainStorm table. An example of the MainStorm table is shown in Figure 10 and Table 4.

The algorithm has a similar concept as Sub Storm Identification algorithm mentioned previously. However, instead of checking neighboring sites, it checks if sub storms are sharing some common site(s) (spatial-window s) within the grouping-window g (in hours). In our analysis, grouping-window is 1 hour and spatial-window is 1 site. That is, if sub storms are within 1 hour difference and sharing at least 1 common site, they will be considered as part of the same overall storm.

Some of our algorithms are outlined in Appendix B.

4. EXAMPLES OF ANALYSIS PERFORMED ON THE STORM TABLES

The output of the storm identification algorithms is now stored in three tables: LocalEvents, SubStorm, and MainStorm. We can then perform SQL to do further analysis on these tables such as storm statistical analysis and storm classification. The following is an example of storm statistical analysis, which is to find the statistics of each type of the storms based on storm characteristics (SQL 1 - 3), e.g., storm duration, storm depth, storm intensity, storm sites total, storm average, storm coverage, storm overall depth, storm overall intensity, and storm overall average, whichever is applicable to each type of such storm as described in section 3.1.

Table 4. An example of MainStorm table corresponding to Figure 10

<table>
<thead>
<tr>
<th>MainStormID</th>
<th>SubStormID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>6</td>
</tr>
</tbody>
</table>

Figure 10. Examples of 2 overall storms on 4/1/2012

Because these three queries are statistical summaries of all three storm types, we pre-compute these three queries and store their results as three additional tables: LocalStormStatistics, OverallStormsStatistics, and HourlyStormsStatistics.
HourlyStormStatistics, and OverallStormStatistics, with the columns as specified in the query for more convenient analysis. Other storms' characteristics can be added later as attributes to these tables. Figure 11 shows a small sample analysis performed on the pre-computed OverallStormStatistics table, which is to find the fifty highest overall storms in term of storm duration.

Figure 11. Sample analysis performed on OverallStormStatistics table

For convenience, we also implemented a visualization component, called Storm Visualization, which processes our output storm data to project the resulted overall storms onto a map. Unlike the raster radar images, which are estimated rainfall values that might or might not actually occur, our storm visualization reconstructed the storm from the actual rainfall values. So, it gives more accurate information when doing analysis. The Storm Visualization allows us to capture different aspects of storm characteristics that could not be seen in the table (scalar) results such as storm formation, storm distribution, and storm movement. Figure 13 shows an example of how overall storm ID 863 in North Central region is formed and moves toward the southeast direction. The Storm Visualization component is implemented in C#, Javascript, HTML5, Google API [13] and ASP.NET.

Figure 12 shows the very first screenshot of the Storm Visualization, which projects the overall storm, 863, onto the map. After triggering by a user, the animation of overall storm (863) is shown hour by hour starting at 4/26/2011, 20:00 (Figure 13 (a)). The number in parentheses indicates the number of sub storms involved in that hour.

Figure 12. A screenshot of overall storm ID 863

5. RELATED WORK

Several studies suggest that storm characteristics analysis can be done in various ways, such as through its statistical properties, depth-duration frequency (DDF [15]), or focusing on its extreme precipitation values.

Asquith [1] studies storm statistical characteristics including the mean (average) of storm inter-event time, storm depth, and storm duration by analyzing hourly precipitation data retrieved from National Weather Service (NWS) [7]. The data contains 155 million values covering 774 sites in Eastern New Mexico, Texas, and Oklahoma. The storm characteristics results are used to help in designing and creating a new runoff control structure. The outputs are in two formats: maps and tables.

[1]’s raw data is stored in file and folder format which raises the difficulty in combining all data across an enormous number of folders and processing them together. Consequently, a huge manual effort is needed to do the analysis. In addition, its analysis has been location-specific (site-specific and regional-specific). So, the storm-specific information is lacking from the work.

For our work, on the other hand, the raw data is stored in the standard CUAHSI ODM database schema, which reflects to the future trend of using hydrological data in this standard format. Our framework can do the analysis in an automated way and process a much larger number of sites. Our algorithm is also customizable through parameters such as inter-event time so it does not need to be fixed to any set of inter-event times, in particular as seen in [1]’s work. Not only can our approach support location-specific analysis but it supports storm-specific analysis as well. So, the complete dimensions of storm characteristics can be analyzed. The following is a small example of how our approach can also be used in location-specific analysis. Suppose we want to find the mean storm depth for Tarrant County, Texas (region-specific storm analysis), we can then perform the following query (SQL 4) on one of our output tables, which store storm information (LocalStormStatistics), to get the answer.

**SQL 4. Find mean storm depth for Tarrant County, Texas**

```
1: SELECT AVG(StormDepth)
2: FROM LocalStormStatistics table
3: WHERE SiteID IN (SELECT SiteID FROM RA_Sites table
4: WHERE State = 'Texas' AND County = 'Tarrant')
```

In [2, 3], Asquith and Roussel study storm characteristics through its Depth-Duration Frequency (DDF [15]) property. [2] presents a procedure to develop a DDF at any location in Texas for the following 14 storm durations: 15, 30, and 60 minutes; 1, 2, 3, 6, 12, and 24 hours; and 1, 2, 3, 5, and 7 days with recurrence intervals ranging from 2 to 500 years. DDF is an estimated depth of the storm given its duration and frequency (recurrence time). It is very important when creating an efficient control structure such
as storm drains or parking lots. It is also used to design efficient river flow and flood prediction models. As a result, it has to be very accurate. To calculate DDF for a storm duration and frequency at any location, we need three storm depths (in inches) retrieved from three maps (location, scale, shape parameter maps) for that storm duration and a storm intensity (in inches per hour) retrieved from precipitation intensity-duration curve of that storm frequency. Then, plug all values into the equation given in the paper [2] and the result is an estimated storm depth for that particular storm.

[3] is an extension of [2]. However, it does not require users to do the calculation themselves. It provides pre-computed DDF maps, which are ready to use. The set of storm durations and storm frequencies, however, are different from [2]. The storm durations only include 15, and 30 minutes; 1, 2, 3, 6, and 12 hours; and 1, 2, 3, 5, and 7 days and the storm frequencies only include 2, 5, 10, 25, 50, 100, 250, and 500 years.

One of the key tasks of [2, 3] is to create location, scale, and shape parameter maps used in the approach. To create such maps, this work uses storm data from National Climatic Data Center (NCDC) [11]. However, only location-specific storm data (by county) is provided by NCDC. So, generating these required maps will be limited to location-specific storm data. In addition, even though NCDC stores storm data in a database, CUASHI ODM was not mentioned as its database schema. As a result, incorporating our storm data (storm-specific) into these two works may enhance their analytic capabilities.

Lanning-Rush [4] studies storm characteristics by focusing on its extreme precipitation (EP) values. The extreme precipitation depth refers to one that exceeds 100-year or greater storm depth. Unlike [1] that considers all storms, only extreme storms were taken into account in this work. Unlike [2, 3] that the inputs are storm duration, frequency and location, it only takes storm duration and area as inputs. The goal of this work is to create the extreme precipitation curve which can be used to estimate extreme precipitation depth for a particular storm duration and area. The EP curves are developed from 24 extreme storms out of 213 notable storms. They select storm durations to include 1, 2, 3, 4, 5, and 6 days and the areas include High Plains, Low Rolling Plains, North Central, Edwards Plateau, South Central, South Texas, East Texas, Upper Coast, and Lower Valley in Texas. Trans-Pecos area, however, was excluded due to the lack of its storm data.

6. CONCLUSION AND FUTURE WORK

6.1 Conclusion

In this paper, we presented a framework for converting the large amounts of rainfall data into storm-specific summaries. The resulting storm data is 1% of the original raw data, and can be easily analyzed and mined using SQL queries and other methods. We first formalized various types of storms that can be identified from standard raw rainfall data. We then developed a customized database schema and algorithms to automate the storm identification process, and to store the identified storms and their characteristics in relational database tables. Analysis tasks can be done in two ways: via SQL for scalar analysis or via our initial Storm Visualization for vector analysis.

6.2 Future Work

For future work, we will develop data mining techniques such as classification, clustering, time series mining, and association rules mining in order to find other interesting characteristics of storms as well as work on calculating other significant measurements such as storm area, storm center, and within storm variations [18].

7. REFERENCES


8. APPENDIX A

This appendix section briefly summarizes the five main tables of CUAHSI ODM tables: Sources, Methods, Sites, Variables, and DataValues tables.

The Sources table stores information about where the observation data comes from. Table 5 shows our Sources table in the database.

### Table 5. Sources table in the database

<table>
<thead>
<tr>
<th>ID</th>
<th>Organization</th>
<th>SourceDescription</th>
<th>SourceLink</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NOAA’s National Weather Service West Gulf River Forecast Center</td>
<td>Files containing MPE data from NWS-WGRFC</td>
<td><a href="http://www.srh.noaa.gov/wgrfc/">http://www.srh.noaa.gov/wgrfc/</a></td>
</tr>
</tbody>
</table>

The Methods table describes how the observation is collected. A brief explanation of the method along with its external link is also provided in this table. Table 6 shows our Methods table in the database.

### Table 6. Methods table in the database

<table>
<thead>
<tr>
<th>ID</th>
<th>MethodDescription</th>
<th>MethodLink</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The precipitation data are multi-sensor (radar, satellite, and rain gauge).</td>
<td><a href="http://www.srh.noaa.gov/fchshare/precip_about_hourly.php">http://www.srh.noaa.gov/fchshare/precip_about_hourly.php</a></td>
</tr>
</tbody>
</table>

The Sites table stores site information. The site information includes SiteID, Longitude, Latitude, LocalX, LocalY, County, State, etc. We have a total of 38,450 sites. Table 7 shows selected columns of our Sites table in the database.

### Table 7. Selected columns of Sites table

<table>
<thead>
<tr>
<th>ID</th>
<th>Latitude</th>
<th>Longitude</th>
<th>LocalX</th>
<th>LocalY</th>
<th>County</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>39672</td>
<td>31.0444</td>
<td>-97.9782</td>
<td>573</td>
<td>200</td>
<td>Tarrant</td>
<td>Texas</td>
</tr>
<tr>
<td>39673</td>
<td>31.0402</td>
<td>-97.9379</td>
<td>574</td>
<td>200</td>
<td>Tarrant</td>
<td>Texas</td>
</tr>
<tr>
<td>39674</td>
<td>31.0359</td>
<td>-97.8976</td>
<td>575</td>
<td>200</td>
<td>Tarrant</td>
<td>Texas</td>
</tr>
</tbody>
</table>

The next table is Variables table. The information about observation is stored in this table. Each variable represents different observation types and properties. The property information includes how frequent the observation is recorded (instantaneous or consistent) and what unit is used for the observation values.

That is, for example, hourly precipitation observation and 15-minute interval precipitation observation are considered different variables due to their properties even though they both are the same precipitation observation types.

We have one variable as demonstrated in Table 8, which is hourly precipitation data.
Table 8. Selected columns of Variables table

<table>
<thead>
<tr>
<th>ID</th>
<th>Code</th>
<th>Name</th>
<th>UnitsID</th>
<th>IsRegular</th>
<th>TimeSupport</th>
<th>TimeUnitsID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MPE</td>
<td>Precipitation</td>
<td>49</td>
<td>1</td>
<td>1</td>
<td>103</td>
</tr>
</tbody>
</table>

The last main table is DataValues table. This table stores numerical observation values for each site and variable as well as the method used and the source where they are from. Table 9 shows some samples of what DataValues table entries look like. The first row of the table states that we have no rain (precipitation value = 0) at site location 88814 from noon to 1 pm on October 1, 2011. As we can see that regardless of whether or not we have rain, the precipitation value is inserted into the table. As a result, the database grows rapidly and sparse.

Table 9. Some examples of DataValues table entries with selected columns

<table>
<thead>
<tr>
<th>ID</th>
<th>DataValue</th>
<th>DateTimeUTC</th>
<th>SiteID</th>
<th>VariableID</th>
<th>MethodID</th>
<th>SourceID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>2011-10-01 13:00</td>
<td>88814</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2011-10-01 13:00</td>
<td>88815</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>2011-10-01 13:00</td>
<td>88816</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

9. APPENDIX B

This appendix section briefly describes some of our algorithms.

Algorithm 1. Event Separator

**Input:**
- Rainfall data of a region (D)
- Inter-event time (v)
- Number of threads (t)

**Output:**
- Local storms stored in LocalEvents table

1: partition sites of region (D) into t subsets (S)
2: assign each subset to a thread
3: concurrently,
4: for each site x in S do
5:     r <- extract and sort (by time) records of site x
6:     for each record rj in r do
7:         if inter-event-count < v then
8:             include rj, precipitation and
9:             identified as part of local storm k
10:        else
11:            start new local storm k++
12:            reset inter-event-count
13:         end if
14:     end for
15: end for
16: merge results from each thread into LocalEvents table

Algorithm 2. Sub Storm Identification

**Input:**
- Local storm data (L)
- Location proximity data (P)
- Space-tolerance (n)
- Outlier detection technique (d)

**Output:**
- Hourly storms stored in SubStorm table

1: for each hour h in L do
2:     b <- extract all records of hour h
3:     for each site s in b do
4:         if s.precipitation <> 0 then
5:             identified as hourly storm i
6:             depthFirstSearch(s, i, b)
7:             start new sub storm i++
8:         else
9:             end if
10: end for
11: depthFirstSearch(s, i, b)
12: candidates set c <- expandNode(s, b)
13: if c <> ∅ then
14:     for each candidate cj in c do
15:         if cj ∈ indirectNeighbors(n, s, P) and
16:             cj.depth is not an outlier(d) then
17:             identified as part of hourly storm i
18:             depthFirstSearch(cj, i, b)
19:         end if
20:     end for
21: end if
22: merge results into SubStorm table