ARTICLE IN PRESS

Computers & Operations Research ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at SciVerse ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

Community-based collaborative information system for emergency management

Juan Li [a,*], Qingrui Li [a], Chao Liu [a], Samee Ullah Khan [a], Nasir Ghani [b]

[a] North Dakota State University, Fargo, ND 58108, USA
[b] University of New Mexico, Albuquerque, NM 87131, USA

ARTICLE INFO

Keywords:
Emergency management
Information system
Distributed systems

ABSTRACT

Natural and man-made disasters, such as tsunamis, earthquakes, floods, and epidemics pose a significant threat to human societies. To respond to emergencies in a fast and an effective manner, Multi-Criteria Decision Making (MCDM) is very important for the decision-making process. The provision of information concerning the "ground-zero" situation to the emergency management stakeholders is an essential prerequisite for MCDM. In this paper, we propose a strategy to form a community-based virtual database, which connects local resource databases of suppliers that provide information and human resources for emergency management. Such a virtual database enables collaborative information sharing among community-based NGOs, public, and private organizations within a community. Moreover, to mobilize resources, the aforementioned process raises awareness within the community and aids in assessing local knowledge and resources. In our work, we present the design, implementation, and evaluation of such a community-based database, which maximally utilizes all of the available information and network resources of a community to better manage natural and man-made disasters.

1. Introduction

Decision making is a crucial issue in emergency management [1]. Multi-Criteria Decision Making (MCDM) [2,3] is a subdiscipline of operations research that explicitly considers multiple criteria in decision-making environments, which can help the stakeholders: (a) assess the current situation, (b) find satisfactory solutions, and (c) take appropriate responses in a timely manner [1]. Emergency MCDM requires the provisioning of accurate and updated information, such as the geographical data of the affected area, data about shelters and the available transportation means, data about victims and relief personnel, available resources, and scientific field measurements. The aforementioned data belongs to multiple autonomous organizations within a community, such as government organizations (GO), non-governmental organizations (NGOs), international non-governmental organizations (INGOs), individuals, communities, and industries. The above mentioned factor motivated us to maximally utilize all of the available information pertaining to a community so that valuable information can be made available to heterogeneous user groups. This in turn enhances the current information systems to better manage natural and man-made disasters. In particular, our work will provide a virtual database that connects local resource databases of suppliers providing information and human resources for use in emergency management. Examples of suppliers may include: construction equipment and operators; medical facilities and personnel; transportation; food, housing, and shelter; and animal shelters. Such a virtual database enables collaborative information sharing among GOs, NGOs, public, and private organizations within a community. The information/data in the community raises awareness within the community and aids in assessing local knowledge and resources.

A large amount of information may be readily available when gathered from the previously mentioned sources. However, when collective emergency-related information is generated on a societal scale and shared across the general public, information collection, integration, storage, and queries must be performed on an unprecedented scale. Managing information on such a large-scale is challenging due to diversity, large amount, dynamic behavior, and geographical distribution. Moreover, it is also challenging to integrate information originating from totally different domains with heterogeneous representative formats. Furthermore, during anomalous events, network disconnections and data failures become the rule rather than exceptions. The system must also respond to the users as quickly and as efficiently as possible with the most updated information. The aforementioned challenges compel us to rethink how to manage, that is to say, how to store,

* Corresponding author.
E-mail addresses: tisini@gmail.com, j.li@ndsu.edu (J. Li), qingrui.li@ndsu.edu (Q. Li), c.liu@ndsu.edu (C. Liu), samee.khan@ndsu.edu (S. Ullah Khan), nghani@ece.unm.edu (N. Ghani).

0305-0548/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2012.03.018

retrieve, explore, and analyze this abundance of relevant, useful, and publically available data.

Current emergency management systems only use and manage relatively limited information, such as within an organization [4,5], with few organizations [6,7,8], for a particular social media (Twitter [9,10]), and inside the emergency management local database [11,12]. The information technologies required to manage large-scale data is still in its infancy. Moreover, existing information systems have difficulties in achieving both rich information retrieval and good scalability. The proposed work will provide practical and applicable solutions to the aforementioned challenges by investigating the problem of the management of information for large-scale natural and man-made emergencies. By utilizing the collaborative power of citizens, settlements, data gathered from communities, the proposed system will create a computationally mediated community which will effectively assist the emergency management process. In particular, the system will provide

- Timely information as current and as detailed as possible from a broad variety of content to satisfy the information need of different individuals and organizations.
- Best-effort automatic information integration to improve interoperability between different information sources and make the integrated knowledge available as fast as possible.

The proposed framework aims to utilize our previous work on information management and large-scale distributed systems [41,42,55,56,57] to emergency management. The proposed framework is not to invent any new computer science methodology; however, it delivers a seamless integration of our past work into practical and applicable solutions for emergency management. Therefore, we must understand that a significant effort has been made to merge two separate domains (data management and large-scale distributed systems) that have great potential to resolve issues related to emergency management.

The rest of the paper is organized as follows. We review the current state of the art in information and communication technologies (ICT) for emergency management in Section 2. The survey primarily focuses on the technological trends and missing features. Our proposed information system is presented in Section 3, in which we utilize all of the available emergency-related information inside the affected community to construct a virtual information repository for emergency management use. Specifically, a virtual community database will be constructed by connecting, integrating, and indexing distributed data sources from different organizations within the affected community. In Section 4, we evaluate the proposed methodologies and show their effectiveness with a comprehensive set of simulations. Finally, we conclude the paper in Section 5.

2. Related work

2.1. ICT for emergency management

Over the past years, a variety of information and communication technologies (ICT) have been proposed for managing national, regional, natural, and man-made disasters [13–22]. ICT can be applied during different stages of an emergency, including emergency prevention, mitigation, preparedness, emergency response, and emergency recovery. Basically, ICT technologies can be used for (a) effective warning of emergencies using different communication channels; (b) integrating information on necessary supplies and other sources; (c) coordinating disaster relief work; (d) encouraging social, institutional, and public responses; (e) evaluating the damages caused by a disaster and the need for disaster relief.

The development of ICT over the last few years has facilitated emergency management with numerous collaborative tools at different levels. In particular, some open source emergency management tools have become very popular [23], such as Ushahidi [12], Sahana [11], and SwiftRiver [24]. Ushahidi [12] was developed to report on the violence during the 2008 Kenyan general election. The idea behind the website was to harness the benefits of crowd-sourcing information (using a large group of people to report on a story) and to facilitate the sharing of information in an environment where rumors and uncertainties were dominant. Since then, Ushahidi has been deployed more than 20 times around the world to cater for similar situations where little or no support is provided by governmental authorities responsible for emergency management. Sahana [11] is a web-based collaboration tool that addresses common coordination problems during a disaster, such as finding missing people, managing aid, managing volunteers, tracking disaster relief camps and the victims. SwiftRiver [24] is a free and open source platform that complements Ushahidi's mapping and visualization products. The goal of the project is "to democratize access to the tools for making sense of information" [24]. Therefore, SwiftRiver helps users to understand and act upon a stream of massive amounts of crisis data.

Our proposed work advances several fundamental design issues that set this work apart from current practices in developing emergency information systems for organizational use. First, this research addresses information needs for a wider audience that include: GOs, NGOs, communities, organizations within a community, and the general public. Most of the existing emergency management information systems [7,8,25] are designed for limited users and organizations, such as the emergency management professionals. Second, by utilizing existing community facilities, such as network, storage, and data the proposed system would be scalable, robust, sustainable, and easier to deploy compared to existing systems such as Refs. [4,5].

2.2. Information integration technologies

To share heterogeneous information from various data sources, effective information integration mechanism is a crucial entity. Information integration has received steady attention over the past two decades, and has now become a prominent area of research. We can roughly classify the integration schema into four categories based on their different treatment of mappings and query answering. The first category is the data warehouse-based information integration [26,27]. The data warehousing schema need to gather all of the data from their distributed sites to a central location. Due to the large amounts of data and security related issues, it is impractical to be applied in our case. The second category, the data exchange-based information integration [28], materializes the global view and allows for query answering without accessing the sources. The third category, the Peer Data Management System (PDMS) [31,32], extends the autonomous data sharing of a peer-to-peer (P2P) system from file exchange to the exchange of semantic rich information. PDMS is built on pairwise mapped network to achieve high flexibility and scalability. In this category, data sources are more freely mapped together and form a graph topology. A query is translated along the pair-wise schema mappings between data sources. Normally it takes multiple hops to translate a query from its originating data source to a "faraway" data source where answers are retrieved. Therefore, the query latency is relatively high. Moreover, query rewriting based on multiple pair-wise schema mapping may cause information loss especially when the mapping pairs are not semantically similar enough. The last category, the mediator-based data integration [29,30] is the integration approach we adopt in our work. In this approach, all the databases map their schemas to the mediated schema so that a query to one database can be translated to queries on other schemas with the help of the schema mediation. The benefit of such a system is that a query translation always takes 2-hops from one database to any another database in the system. The difficulty of this architecture is the creation, maintenance and synchronization of mediated schemas among distributed data sources.

2.3. Ontologies related to emergency management

Employing ontology and semantic web technologies can improve information interoperability across the stakeholder functions within the area of emergency management. Ontology definition for emergency management is still immature. Although there are many example ontologies related to disaster/emergency [38,39,40,54], such ontologies have not been widely adopted in the industry or public organizations. Among the existing works, the W3C Emergency Information Interoperability Framework [54] is one of the most prominent one and as such, our proposed work adopts all of the concepts reported in [54]. The W3C Emergency Information model represents the concepts and relationships that define an overall context for sharing of coordination information in an emergency. The model uses the scenario "Who (organizations or people) does What (activity) Where" as a basis to derive high-level concepts and relationships, and is developed based on data schema from existing emergency information systems. In our work, we reuse the existing ontologies and at the same time extend them with our new findings.

3. System design

3.1. Overview

The proposed project aims to utilize the massive amount of computing and information resources inside a community to construct a collaborative social computing architecture to support community emergency management. We use a P2P architecture to manage distributed datasets of the affected community to form a P2P-based networked database (as depicted in Fig. 1). In the architecture, each organization maintains its own dataset and also connects to one or more of the organizations within a community. As a consequence, data sharing will be performed in the community network. The P2P model will allow a dataset to easily join and leave the community network; it also will allow owners of data sources to fully control access to and sharing of data without relying on a centralized server (a potential bottleneck). Moreover, the failure of one dataset will not affect the functionality of the whole system. The aforementioned advantages make the proposed P2P architecture an ideal choice for emergency management systems in which data sharing must be set up quickly and easily with limited resources but the availability of a centralized server cannot be guaranteed.

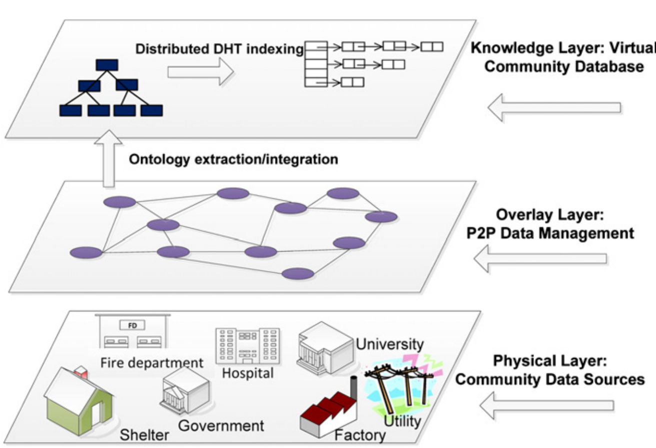

Fig. 1. System architecture.

To overcome data heterogeneity that originates from having different organizations involved in the community, we will extend the distributed datasets with a semantic dimension. In particular, we will utilize ontologies to unify the semantically related data in different sources. In an emergency affected community, it is impractical to assume that there is a global ontology defined for all participating data sources. We will take advantage of the special property of emergency-related query, which focuses on concepts and relations about location, time, weather, and other emergency-related properties, to simplify the data integration. Therefore, we propose a more feasible approach that allows for the different data organizations to maintain their own ontologies locally while specifying how emergency-related concepts in their own ontology correspond to concepts in the ontology of the emergency-related domain. To enable sharing and integrating between participating datasets, ontologies of each dataset will be extracted as a conceptual view over the data. This will also enable the access and query of the underlying data with vocabularies of ontologies. To locate desirable data to construct mappings between datasets and to resolve queries efficiently, we utilize a distributed hash tables (DHTs)-based P2P network [33–37] to implement a distributed ontology repository for storing, indexing and querying ontology knowledge. DHTs are a class of decentralized distributed systems that partition ownership of a set of keys among participating nodes, and can efficiently route messages to the unique owner of any given key. The indexing on the distributed repositories will speed up the searching process by only pushing down queries to information sources we can expect to contain an answer.

Security is a vital issue for such systems. Although it is out of the scope of our research, we must understand that security mechanisms, such as authentication and authorization that provide different access control to different organizations and individuals, and encryptions/decryption may be applied to guarantee the security of the system.

3.2. Emergency-related ontology definition

Inside a community, various organizations develop their own datasets without fully understanding each other. Therefore, data heterogeneity becomes the primary problem that must be solved when designing an information system. To address the need for information interoperability in emergency management, we extend our distributed datasets with a semantic dimension using semantic web technologies. The semantic web is a framework specifically designed to foster information sharing and multidisciplinary use of informational resources in collaborative and distributed environments. In particular, we exploit the benefits provided by semantics through ontology. Ontology is defined as "an explicit specification of a conceptualization" [29]. It provides a common understanding of a domain by defining a controlled vocabulary of concepts and the relationships between them. Therefore, ontology supports the exchange of semantics not just syntax. However, in a large-scale community, it is impractical to assume that there is a global ontology defined to integrate all of the data owned by different participants. From a practical point of view, the information need for emergency management focuses on emergency-related information, we can merely use emergency-related ontology to integrate

Please cite this article as: Li J, et al. Community-based collaborative information system for emergency management. Computers and Operations Research (2012), http://dx.doi.org/10.1016/j.cor.2012.03.018

information from different sources and ignore other semantics. Therefore, a viable practice is to adopt foundational emergency-related ontologies, i.e., conceptual models of common, cross-domain notions, such as spatial–temporal ones to identify and associate semantically corresponding concepts in the disaster-related information, so that the heterogeneous data can be integrated and ingested.

We revised and enhanced existing emergency-related domain ontologies [38,39,40,54] in our work. In particular, we adopt the "who-what-where" model proposed by W3C Emergency Information Interoperability Framework [54] and extend it with two other important directions, "When" and "Where". To define the ontology, we collaborate closely with the relevant agencies providing emergency services. The definition is based on the knowledge inferred from the interviews with the relevant partners, papers, manuals, and emergency proceedings. In particular, we focus on some main subjects, namely: types of hazards and emergencies, and meteorology, i.e., weather issues that might trigger an emergency situation, date and time, and most importantly, geographic concepts that can describe geographical regions affected by the emergency. To cope with the openness and extensibility requirements, we adopt two W3C recommendations: the Resource Description Framework (RDF) and the Web Ontology Language (OWL) as our ontology language. Fig. 2 presents part of the ontologies we adopted from the W3C information framework.

3.3. Ontology-based integration

To enable sharing and integrating between participating datasets, termed *peers*, ontologies of each peer will be extracted as a conceptual view over the data. This will enable the access and query of the underlying data with ontology vocabularies. The mapping between the relational (or XML) database and the ontology is captured by associating a view over the source data to each element of the ontology. At each data source, all of the source schemas are integrated into a single local ontology, which is represented in RDF schema. Based on our previous research [41], which maps relational schema to ontologies, our principles are based on best practices on relational schema design from ER diagrams. Specifically, we consider ER models that support entities with attributes, relationships, and inheritance hierarchies between entities. We reverse the process of translating ER model to relational model. In this way, we convert entities, keys, foreign keys to entities and relationships (including hierarchical relationship) in ontology. If the data source stores XML data, then we can analyze the XML Schema to find all the elements, attributes, and their hierarchies, which can easily be mapped to ontology. Once the schema has been mapped to a particular ontology; thereafter, we can map the data instances. *Join* operation will be performed on data from multiple tables connected with foreign keys.

We adopt the mediator-based integration [29,30] to integrate distributed data sources. The emergency-related domain ontology described in the previous section will work as a mediated schema providing a conceptual representation of the domain (i.e., a globally shared vocabulary and a set of constraints). Each individual data source is described by a source ontology extracted using the aforementioned approaches. To integrate these individual data sources, each data source is mapped to the global domain ontology by relating its emergency-related objects to the defined emergency domain model. This process normally needs human intervention. The mappings clarify the semantics of the source objects and help find semantically corresponding objects.

3.4. P2P-based indexing

After ontology extraction, each peer's local ontology repository makes flexible statements about the dataset. However, putting an ontology document in a peer's repository does not mean that others are able to find it. The system needs a scheme to locate desirable data to construct mappings between dataset and to resolve queries efficiently. For this purpose, we propose a DHT-based P2P overlay to implement a distributed ontology repository for storing, indexing and querying ontology knowledge. Distributed hash tables (DHTs) are a class of decentralized distributed systems that partition ownership of a set of keys among participating nodes, and can efficiently route messages to the unique owner of any given key. The indexing on the distributed repositories speeds up the searching process by only pushing down queries to information sources we can expect to contain an answer.

Previously, we have proposed an efficient index scheme to index semantic web data with a DHT overlay [42]. The basic idea is to divide a resource's RDF description into triples and index the

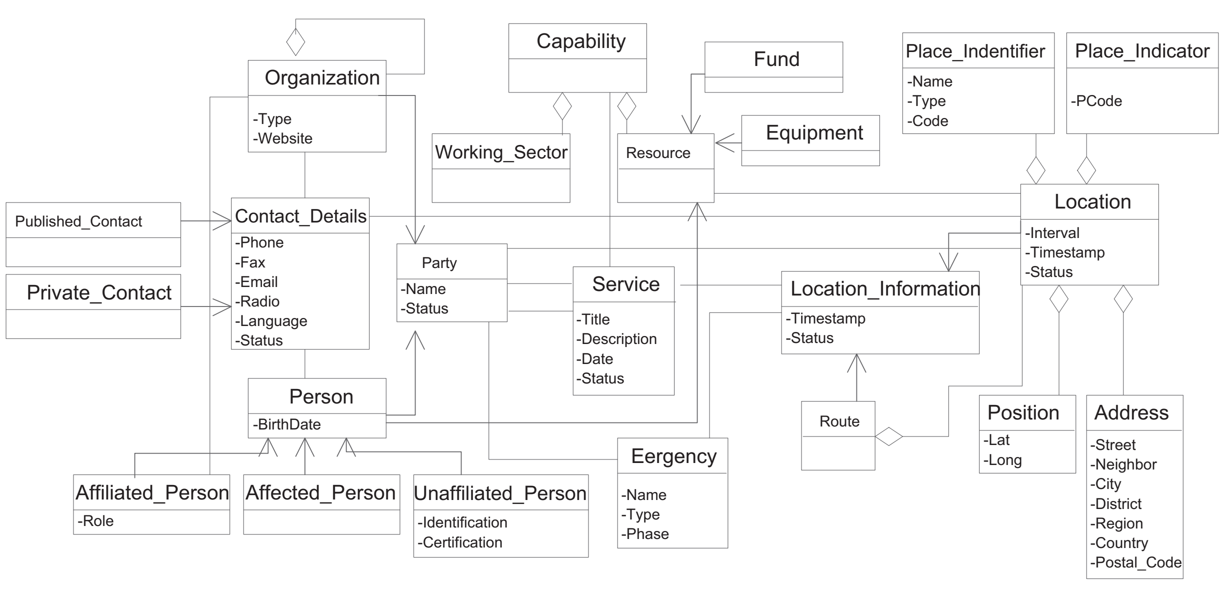

Fig. 2. Ontology adopted from the W3 Coordination Use Case Information Model [54].

Please cite this article as: Li J, et al. Community-based collaborative information system for emergency management. Computers and Operations Research (2012), http://dx.doi.org/10.1016/j.cor.2012.03.018

triples in a DHT overlay. We store each triple three times by applying a hash function to its *subject*, *predicate*, and *object*. The insertion operation of a triple t is performed as follows:

$$\text{Insert}(t) \equiv \text{Insert}(\text{SHA1Hash}(t.\text{subject}),t),\text{Insert}(\text{SHA1Hash} \times (t.\text{predicate}),t),\text{Insert}(\text{SHA1Hash}(t.\text{object}),t)$$

In this fashion, a query providing partial information of a triple can be handled. We adopt this idea to index the semantic data extracted from each peer (in the global ontology) to the network. Each data source publishes its data to the DHT network against the global schema type. The DHT network then maintains the sources capability lookup for all sources. The ontological indexing scheme distinguishes schema-based ontology (TBox) and instance-based ontology (ABox). In this way, indices can be created based on these two types of ontology. Given the combination of these two indexing schemes an application can choose which scheme fits the needs of the system best. As illustrated by our prior studies [42] and experimental results re-verified in this study, the DHT overlay will be able to scale to thousands of nodes and to large amounts of ontology data and queries.

3.5. Semantic query evaluation

When issuing a query, users do not need to know where a particular piece of information resides. The system behaves as if all the information is available from a single source. The query answering system can locate relevant information, retrieve it, and combine the individual answers. We use SPARQL [44] as the query language, but the query evaluation approach is not limited to a specific query language.

The query evaluation process begins with the parsing of a user's query to the SPARQL format. After that, the query in the user's local ontology will be first reformulated into the shared domain ontology based on the mediated mappings. Following the above mentioned step, the query will be further rewritten into sub-queries using the semantic mapping axioms. Thereafter, each of the sub-queries will be executed at different sources (in parallel) and the query engine will collect returned answers from the sources and combine them (if needed). The system supports two categories of queries, schema-based queries and instance-based queries, for querying abstract structural knowledge and concrete instance knowledge. A solution to a SPARQL graph pattern with respect to a source RDF graph G is a mapping from the variables in the query to RDF terms such that the substitution of variables in the graph pattern yields a sub-graph of G [45]. More complex SPARQL queries are constructed by using projection (SELECT operator), left join (OPTIONAL operator), union (UNION operator), and constraints (FILTER operator) [46]. The semantics for these operations are defined as algebraic operations over the solutions of graph patterns [47].

The simplest query is to ask for resources matching a single triple pattern. In this query pattern, there is only one triple pattern and at least one part of the triple is a constant. Because we store each triple three times based on its hashed subject, predicate, and object values, we can resolve the query by routing it to the node responsible for storing that constant. Thereafter, the responsible node matches the triple against the patterns stored locally and returns results to the requesting node.

If the graph pattern is more complex containing multiple triples or the query contains a group graph pattern, then each triple pattern will be evaluated by one or two different nodes. These nodes form a processing chain for the query. The first triple pattern is evaluated at the first node, the result is then sent to the next node for further processing. The aforementioned process continues until the last triple pattern is processed. An alternative approach is to process patterns in parallel, and all results are sent to one node to do the final processing. A system should choose the appropriate approach according to its application. In our work, we use the sequential approach since sequentially joining intermediate results saves the traffic for transferring large amounts of unrelated data. The sequence to evaluate the triple patterns is crucial. Many database researchers have worked on it [48,49]. Here, for simplicity, we assume that we evaluate the query with the original triple pattern order, in which adjacent triple patterns share at least one common variable.

For a query q that has k conjunctive triple patterns $(t_1, t_2, \ldots t_k)$, the query evaluation proceeds as follows: First, t_1 is evaluated using the single triple pattern processing method mentioned previously. The result is projected on the variables with values that are needed in the next query evaluation. Thereafter, the query together with the next triple sequence number and the intermediate result is sent to the node responsible for the next triple pattern. When a node n_i receives the query request, n_i evaluates the i-th triple pattern t_i of the query using its local triple index and the intermediate result from previous nodes. Thereafter, n_i computes the intermediate result and projects the result on columns that are needed in the rest of the query evaluation (i.e., variables appearing in the triple pattern t_{i+1} of q). The aforementioned is a nested loop join on the common column for the inner relation. The process recursively repeats until the last triple pattern t_k of q is evaluated. Then, the last node n_k simply returns the result back to the querying node. We use an example to explain this process. The query to find victims who live in Fargo area is listed below:

```
SELECT?victims
WHERE {
   ?victims:locatesIn?location.
   ?location:belongsTo?region.
   ?region:label Fargo
}
```

The query evaluation process is illustrated in Fig. 3. Each event in this figure represents an event in the network, i.e., the arrival of a new query request. The query request consists of three parts: (1) the original query, (2) the triple pattern to be processed in this node, represented with that triple's sequence number in the original query's triple lists, (3) the intermediate result from previous nodes. Initially, the intermediate result is empty (Ø).

We have studied how to evaluate different SPARQL queries (Single triple pattern, Conjunctive patterns, Value constraints, and Optional patterns, Disjunctive patterns) using DHT indexing. The readers are strongly encouraged to find more details about query evaluation from our previous publication [42].

In our previous description of query evaluation, we assume that the overlay maintains instance (A-Box in logic terminology) indexing. In that scenario, instance triple patterns are indexed in the network and queries for instances can be accurately

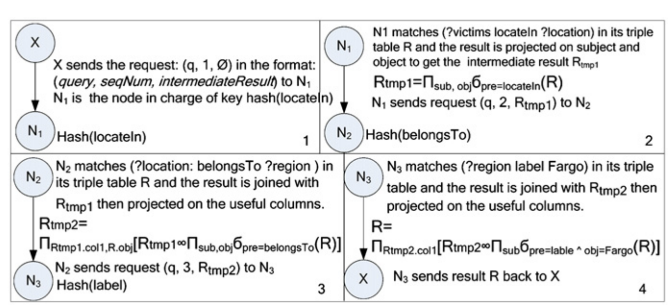

Fig. 3. Processing a query with a conjunctive pattern (Results are represented as relational algebras. Π:Projection, б:Selection, ∞:join).

Please cite this article as: Li J, et al. Community-based collaborative information system for emergency management. Computers and Operations Research (2012), http://dx.doi.org/10.1016/j.cor.2012.03.018

forwarded to the right peers in charge of the triples. If an application only maintains schema (T-Box in logic terminology) indices, the evaluation process is different. For queries on T-Box indexing, the evaluation process is similar to the query evaluation process we just explained, because T-Box indexing is detailed enough to answer the schema query. T-Box indexing cannot be used directly to evaluate queries at the instance level; however, it can restrict the query to a small set of nodes which are ontologically related to the query. These nodes have the T-Box knowledge to understand the query, thus are capable of answering the query. When a node issues an instance-level query, the T-Box concepts related to the query are extracted in the form of a keyword list, and these keywords are used as parameters to retrieve the relevant peers.

3.6. *Prototype Implementations*

We have implemented a prototype system in the form of a social networking site (SNS). The prototype provides users with an interface to access the information repository to give them a good situational awareness view and expand the capability of sharing information with emergency partners at all levels. Besides providing information, it also provides users with communication and collaboration tools to deal with disaster situation. We choose SNS over the traditional classical ICT system, because SNSs create Inter-organizational networking that can play an important role in facilitating the flow of information across organizational boundaries in emergencies [50], and increases interaction among organizations [51]. Moreover, SNSs allow the community/group as a whole to engage in overall higher levels of risk-taking [52], and solve collective action problems more easily [53]. Furthermore, SNSs facilitate the rapid dissemination of information and improve access to resources among network members. Fig. 4 shows a screenshot of a web-based user interface.

We plan to deploy our system through a website targeting the residents of Fargo–Moorhead area. Here, the unique case of the annual Red River cresting will be used to assess the community response towards repeated possibilities of large-scale natural disasters. Namely, we will examine the overall users' experience levels with our system and use the feedback to improve our design, i.e., in terms of disaster warning, relief, and recovery. In particular, our evaluation efforts will use both quantitative and qualitative social science research methods.

4. Experimental evaluations using simulations

In this section, we evaluate the performance of the proposed system with simulation experiments. We created an experimental scenario to show that our community-based information system has good performance in different aspects, such as query recall, system scalability, overhead, etc.

4.1. *Experimental setup*

We use both the publicly available International Disaster Database (EM-DAT http://www.em-dat.net/) and artificially generated data to provide reasonable approximation to evaluate the performance of the system. EM-DAT emergency database contains essential core data on the occurrence and effects of over 18,000 mass disasters in the world from 1900 to present. The database is compiled from various sources, including UN agencies, non-governmental organizations, insurance companies, research institutes and press agencies. EM-DAT includes fields related to a disaster, such as a unique disaster number for each disaster event, countries in which the disaster occurred, disaster group, disaster sub-group, disaster type and subset, the dates when the disaster occurred and ended, number of people confirmed dead and number missing and presumed dead, number of people suffering from issues, such as physical injuries, trauma or an illness requiring immediate medical treatment as a direct result of a disaster. Fig. 5 shows part of an ontology implicitly defined in EM-DAT. Data from EM-DAT are relatively simple compared to the potential data in our complex community information system; therefore, we also artificially generated a more complex data set to evaluate the system.

In the simulation experiments, all of the EM-DAT data was converted to RDF triple instances. Because EM-DAT has a single schema, this can be easily performed. To artificially generate data, we use a small-sized vocabulary set to define the ontology. We have generated the test data in multiple steps. The algorithm starts with generating the ontology schema (T-Box). Each schema includes the definition of a number of classes and properties. The classes and properties may form a multilevel hierarchy.

Fig. 4. Screenshot of the prototype interface.

Please cite this article as: Li J, et al. Community-based collaborative information system for emergency management. Computers and Operations Research (2012), http://dx.doi.org/10.1016/j.cor.2012.03.018

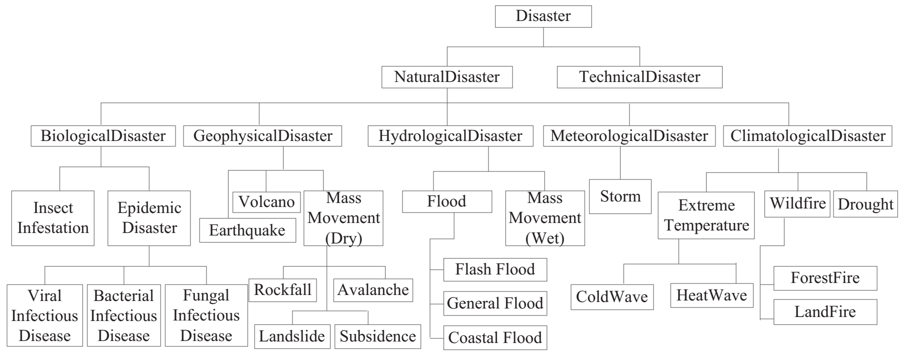

Fig. 5. Part of an ontology implicitly defined by EM-DAT.

Thereafter, the classes are instantiated by creating a number of individuals of the classes. To generate an RDF instance triple t, we first randomly choose an instance of a class C among the classes to be the subject: $sub(t)$. A property p of C is chosen as the predicate $pre(t)$, and a value from the range of p to be the object: $obj(t)$. If the range of the selected property p are instances of a class C', then $obj(t)$ is a resource; otherwise, it must be considered as a literal.

For both the EM-DAT data and the artificial data, the queries are generated by randomly replacing parts of the created triples with variables. In our experiments, we use single-triple-queries and conjunctive-triple-queries. To create the conjunctive-queries for artificial data, we randomly choose a property p_1 of class C_1. Property p_1 leads us to a class C_2, which is within the range of p_1. Thereafter, we randomly choose a property p_2 of class C_2. This procedure is repeated until the range or the property is a literal value or we have created n ($n \leq 3$) triple patterns.

We implement a simulator based on Pastry [35] in Java on top of which we developed our indexing and routing algorithms. Each peer is assigned a 160-bit identifier, representing 80 digits (each digit uses 2 bits) with base $b=2$. Once the network topology has been established, we randomly assign each node a data set (a partial EM-DAT data set or an artificially generated data set) and then they publish their data on the overlay network. Thereafter, a mixture of joins, leaves/failures, and queries are injected into the network based on certain ratios. Inserted nodes start functioning without any prior knowledge. Each experiment is run ten times with different random seeds, and the results are the average of these ten sets of results.

4.2. Experimental results

In this section, we discuss the experiments performed to evaluate the system performance in terms of scalability, latency, overhead, and fault tolerance.

The first set of experiments is to verify the scalability of the system and efficiency of answering typical lookup requests. We vary the number of Pastry nodes in the network from 2^9 to 2^{12}. We run two trials of experiments (one trial issues only single-triple-queries, while the other trial issues conjunctive-triple-queries) on two sets of data—the EM-DAT data and the artificial data, respectively.

Figs. 6 and 7 show the average number of routing hops taken as a function of the network size for both query patterns. $\log_2 b\, N$ is the expected maximum number of hops required to route a key in a network containing N nodes (In our experiment $b=2$), therefore, in the figures, "$\log_4 N$" is included for comparison. The experiment shown in Fig. 6 was performed on artificial data, while the experiment shown in Fig. 7 was performed on EM-DAT data. We can see that the performances on these two data sets are very similar. The results show that the number of route hops scales with the size of the network as predicted: for the single triple query, the route length is below $\log_4 N$. For conjunctive queries, the number of routing hops is below $3 \log_4 N$ as expected. Because our experiments performed on the two kinds of data set (artificial and EM-DAT) demonstrate similar results, in the following experiment, we do not present the two sets of results; instead, we only show the results on the artificial data set.

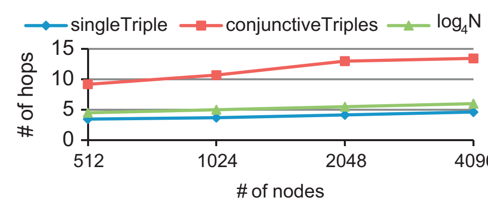

Fig. 6. Average number of hops per query vs. network size on artificial data set.

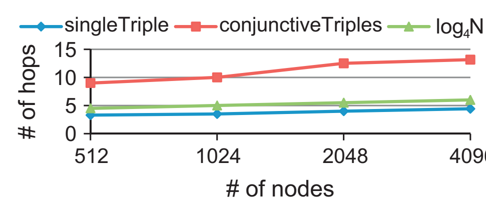

Fig. 7. Average number of hops per query vs. network size on EM-DAT data set.

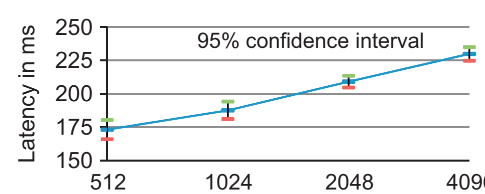

Fig. 8. Query latency vs. network size (using DHT-based forwarding).

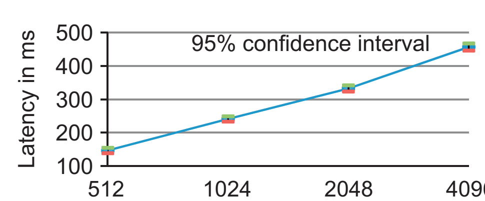

Fig. 9. Query latency vs. network size (using multicast-based forwarding).

Figs. 8 and 9 plot the mean query latency and the 95% confidence interval as the network size increases. Fig. 8 presents the query latency based on our DHT-based query forwarding, while Fig. 9 illustrates the query latency on traditional multicast-based query forwarding. It is noteworthy to mention that to show the 95% confidence interval (more clearer), the two figures are plotted with different scales. As can be seen, query using our DHT-based forwarding outperforms multicast-based forwarding in query latency.

Figs. 10 and 11 compare the mean bandwidth consumption and the 95% confidence interval of query over our DHT-based forwarding and query over multicast-based forwarding. Moreover, to clearly plot the 95% confidence interval, Figs. 10 and 11 are plotted with different scales. It can be observed that our DHT-based query evaluation scheme consumes much less bandwidth compared with the multicast-based query forwarding.

The next set of experiments evaluates the fault tolerance of proposed system. We use an Information Retrieval (IR) standard, recall, as the performance metrics. Recall refers to completeness of retrieval of relevant items, as per Eq. (1). The "document" in the IR definition represents a semantic entity in our experiment.

$$\text{recall} = \frac{|\text{relevant Documents} \cap \text{retrieved Documents}|}{|\text{relevant Documents}|} \quad (1)$$

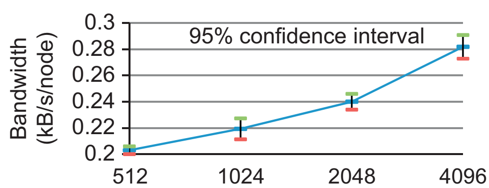

Fig. 10. Bandwidth consumption vs. network size (using DHT-based forwarding).

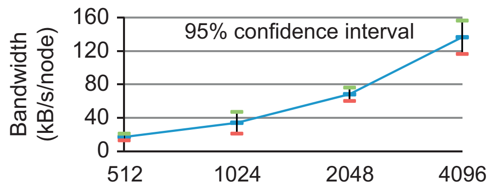

Fig. 11. Bandwidth consumption vs. network size (using multicast-based forwarding).

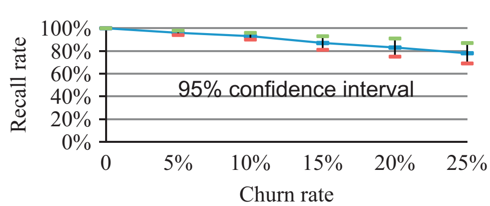

Fig. 12. Recall rate vs. network size.

Fig. 12 presents the mean query recall rate and the 95% confidence interval under different node churn rate that represents the collective effect caused by independent arrival and failure of thousands of nodes in the network. As can be seen from the figure, the system performance degrades gracefully with the increase of churn rate. Even a quarter of nodes leave/join the network, the system still can answer about 80% of the queries on average. The aforementioned demonstrates that our system is resilient to intermittent network connections.

The last set of experiments studies the performance of the T-Box and the A-Box indexing in terms of indexing overhead and query overhead. Each node may randomly choose n ($n<3$) ontologies from 100 distinguished ontologies, and instantiate each class with m ($m<10$) instances. For simplicity, each query uses the simple single triple pattern. With this configuration, we see from Table 1 that A-Box indexing incurs much more overhead than T-Box indexing, and the discrepancy increases as the network size increases. For example, A-Box indexing causes several orders of magnitude higher overhead than what TBox indexing creates when the network size is 4096. On the other hand, if the system can afford the cost of maintaining the large index, then A-Box indexing can improve searching efficiency. Table 2 shows the query overhead in terms of cumulative query messages. It is clear that with A-Box indexing, processing a query requires much less message forwarding overhead than that based only on T-Box indexing. Therefore, there is a trade-off between A-Box indexing and T-Box indexing. There are many factors to consider that can determine the right indexing scheme, for example, the storage capacity of the participating nodes, the nature of the major queries, and even the organizations' policy. Another important factor is the degree of heterogeneity of the system's ontology.

5. Conclusions and future work

Disaster management normally involves different authorities and organizations, such as central government, local authorities, police and fire department, health and ambulance services, utility companies, monitoring and observatory centers. Moreover, the popularity of social media and the ubiquity of mobile wireless devices may facilitate the general public's involvement in disaster management. Therefore, we believe that when provided with appropriate tools, a large number of people, communities, and organizations can be effectively utilized managing large-scale disasters. To realize this envision, we proposed an information system that maximally utilizes all of the available information and human power of a community to better manage natural and man-made disasters. The proposed research will develop the next generation of disaster management systems by advancing the current state of the art in information system—theory and practice. It combines the power of community, citizen, information and computing platform to attack critical disaster management problems. It brings novel

Table 1
Cumulative indexing storage load of T-Box indexing and A-Box indexing.

Network Size	256	512	1024	2048	4096
Cumulative A-Box index (bytes)	10472400	20732040	39491280	81690660	1.63E+08
Cumulative T-Box index (bytes)	365497	370939	381086	403625	446060

Table 2
Cumulative query overhead based on T-Box index and A-Box index.

Network Size	256	512	1024	2048	4096
Cumulative Query messages on A-Box index	17880	22080	21840	24780	26880
Cumulative Query messages on T-Box index	66120	105360	170880	302940	573960

Please cite this article as: Li J, et al. Community-based collaborative information system for emergency management. Computers and Operations Research (2012), http://dx.doi.org/10.1016/j.cor.2012.03.018

solutions (such as: (a) information collection, integration, indexing, and querying, and (b) mass collaboration) to fundamental issues underlying virtually any large-scale information efforts.

After a successful completion of the emergency information system, the system will be deployed to a website targeting the residents of the Fargo–Moorhead area. The unique case of the repeated Red River cresting will be used to assess the community response towards repeated possibilities of large-scale natural disasters. Examination of users' experience with it in the disaster warning, relief, and recovery will be performed to examine the effectiveness of the system in helping people gaining timely, accurate, and trustworthy information related to disasters. We hope that through an extensive evaluation of the system, we can continuously improve the developed technology. The studies that focus on evaluations of the proposed system will encompass both quantitative and qualitative social science research methods.

References

[1] Peng Y, Zhang Y, Tang Y, Li S. An incident information management framework based on data integration, data mining, and multi-criteria decision making. Decision Support System 2011;51(2):316–27.
[2] Köksalan M, Wallenius J, Zionts S. Multiple criteria decision making: from early history to the 21st century. Singapore: World Scientific; 2011.
[3] Ergu D, Kou G, Shi Y, Shi Y. Analytic network process in risk assessment and decision analysis. Computers and Operations Research 2011.
[4] HICS: Hospital Incident Command System. 〈http://www.emsa.ca.gov/hics/〉.
[5] National Incident Management System. 〈http://www.fema.gov/emergency/nims/〉.
[6] White C, Hiltz SR, Turoff M. United we respond: one community, one voice. 2008.
[7] DisasterLAN Incident Management System. 〈http://www.semo.state.ny.us/programs/training/Courses/disaster_lan.cfm〉.
[8] E-SPONDER. 〈http://www.e-sponder.com/overview.html〉.
[9] Vieweg S, Hughes AL, Starbird K, Palen L. Microblogging during two natural hazards events: what twitter may contribute to situational awareness. 2010.
[10] Hughes, AL Palen, L. Twitter adoption and use in mass convergence and emergency events. 2009.
[11] SAHANA Website. 〈http://sahanafoundation.org/〉.
[12] Ushahidi Website. 〈http://www.ushahidi.com/〉.
[13] O'Brien G, Read P. Future UK emergency management: new wine, old skin? Disaster Prevention and Management 2005;14(3):353–61.
[14] Wattegama C. ICT for disaster management, Asia-Pacific development information programme. e-Primers for the Information Economy, Society and Polity; 2007.
[15] Fiedrich F, Burghardt P. Agent-based systems for disaster management. Communications of the ACM 2007;50(3):41–2.
[16] Mendonca D, Jefferson T, Harrald J. Collaborative adhocracies and mix-and-match technologies in emergency management. Communications of the ACM 2007;50(3):45–9.
[17] Wybo J-L, Lonka H. Emergency management and information society: how to improve synergy? International Journal of Emergency Management 2002;1(2):183–90.
[18] Carver L, Turoff M. Human-computer interaction: the human and computer as a team in emergency management information systems,. Communications of the ACM 2007;50(3):33–8.
[19] Palen L, Liu S. Citizen communications in disaster: anticipating a future of ICT-supported public participation2007.
[20] Fabbri K, Weets G. ICT for environmental risk management in the EU research context.Geo-information for disaster management. Berlin Heidelberg: Springer; 2005 pp. 51–56.
[21] Becker S. Emergency communication and information issues in terrorist events involving radioactive materials. Biosecurity Bioterrorism: Biodefense Strategy, Practice, and Science 2004;2(3) 195–20.
[22] Hagar C. The information and social needs of Cumbrian farmers during the UK 2001 foot and mouth disease outbreak and the role of information and communication technologies. In: Döring M, Nerlich B, editors. The socio-cultural impact of foot and mouth disease in the UK in 2001: experiences and analyses. Manchester University Press; 2009.
[23] Currion P, Silva CD, Walle BV. Open source software for disaster management. Communications of the ACM—Emergency response information systems: emerging trends and technologies CACM Homepage table of contents archive 2007;50(3):61–5.
[24] SwiftRiver Website. 〈http://swift.ushahidi.com/〉.
[25] WebEOC: Emergency Services integrators. Incorportated (ESi) 〈http://www.buffalocomputergraphics.com/content/pages/dlan-crisis-info-mgmt-system〉.
[26] Ahmad H, Kermanshahani S, Simonet A, Simonet M. Data warehouse based approach to the integration of semi-structured data. 2010.
[27] Halevy AY, Ashish N, Bitton D, Carey MJ, Draper D, Pollock J, et al. Enterprise information integration: successes, challenges and controversies. SIGMOD (Special Interest Group on Management of Data) conference 2005:778–87.
[28] Fagin R, Kolaitis P, Popa L. Data exchange: getting to the core. ACM Transactions on Database Systems 2005;30(1):174–210.
[29] Bergamaschi S, Castano S, Vincini M, Beneventano. D. Semantic integration of heterogeneous information sources. Data and Knowledge Engineering 2001;36(3):215–49.
[30] Lenzerini M. Data integration: A theoretical perspective. 2002;233–46.
[31] Tatarinov I, Halevy A. Efficient query reformulation in peer data management systems. 2004.
[32] Ng WS, Ooi BC, Tan K-L, Zhou A. Peerdb: A p2p-based system for distributed data sharing. 2003.
[33] Li Y, Feng Y. Study of data search in DHT P2P networks. Application Research of Computers 2006.
[34] Zhao BY, Kubiatowicz JD, Joseph AD. Tapestry: an infrastructure for fault-tolerant wide-area location and routing. Technical Report. UCB/CSD-01-1141, April 2000.
[35] Rowstron A, Druschel P. Pastry: scalable, distributed object location and routing for large-scale peer-to-peer systems. 2001.
[36] Stoica I, Morris R, Karger D, Kaashoek MF, Balakrishnan H. Chord: a scalable peer-to-peer lookup service for internet applications. 2001;149–60.
[37] Ratnasamy S, Francis P, Handley M, Karp R, Shenker S. A scalable content-addressable network. 2001;161–72.
[38] Xu W, Zlatanova S. Ontologies for disaster management response. Lecture Notes in Geoinformation and Cartography 2007:185–200.
[39] Wang W, Dong C, Yang P. Ontology modeling of emergency plan systems. Fuzzy Systems and Knowledge Discovery 2009.
[40] Little, E, Rogova, G. Ontology meta-model for building a situational picture of catastrophic events. 2005.
[41] Li J, Su Y. A framework for efficient query answering on semantically heterogeneous grids. The International Journal of Computational Science 2009;3:4.
[42] Li J. Building distributed index for semantic web data. 2009;660–7.
[44] Harris S Shadbolt N. SPARQL query processing with conventional relational database systems. 2005;235–44.
[45] Hayes P. RDF semantics. W3C Recommendation 〈http://www.w3.org/TR/owl-semantics/〉; 2004.
[46] Sirin E, Parsia B. SPARQL-DL: SPARQL query for OWL-DL. 2007.
[47] Perez J, Arenas M, Gutierrez C. The semantics and complexity of SPARQL. 2006.
[48] Jarkr M, Koch J. Query optimization in database systems. ACM Computing Survey 1984;16:2.
[49] Sellis T. Multiple-query optimization. ACM Transactions on Database Systems 1990;12(1):23–52.
[50] Kapucu N. Interorganizational coordination in dynamic context: networks in emergency response management. Connections 2005;26(2):33–48.
[51] Coleman J. Foundations of social theory. Cambridge, Mass.: Belknap: Harvard University Press; 1990.
[52] Fukuyama F. Trust: the social virtues and the creation of prosperity. New York: The Free Press; 1995 pp. 1995.
[53] Palen L, Hiltz SR, Liu SB. Online forums supporting grassroots participation in emergency preparedness and response. Communications of the ACM 2007;50(3):54–8.
[54] Emergency Information Interoperability Frameworks. 〈http://www.w3.org/2005/Incubator/eiif/XGR-Framework-20090806/〉.
[55] Li J, Li Q, Khan SU, Ghani N. Community-based cloud for emergency management. 2011.
[56] Li J, Khan SU, Li Q, Ghani N, Min-Allah N, Bouvry P, et al. efficient data sharing over large-scale distributed communities. In: Bouvry P, Gonzalez-Velez H, Kolodziej J, editors. Intelligent decision systems in large-scale distributed environments. New York, NY, USA:Springer; 2011.
[57] Li J. Grid resource discovery based on semantically linked virtual organizations. Journal of Future Generation Computer Systems—Elsevier 2010;26(3).

Please cite this article as: Li J, et al. Community-based collaborative information system for emergency management. Computers and Operations Research (2012), http://dx.doi.org/10.1016/j.cor.2012.03.018