Assuring Information Quality in Sharing Platform for Disaster Management

Ying Su and Jie Peng
Institute of Scientific and Technical Information/Center for Resource Sharing Promotion, Beijing, China
Email: { suy.rspc; pengj } @istic.ac.cn

Zhanming Jin
Tsinghua University/School of Economics and Management, Beijing, China
Email: jinzhm@sem.tsinghua.edu.cn

Abstract—**Disaster management is a multifaceted process aimed at minimizing the social and physical impact of these large-scale events. In this paper, we propose a quality-mind assuring approach in which web services are discovered, composed and executed considering both functional and QoS requirements. We prescribe a QoS management framework that defines fundamental principles, concepts and mechanisms which can be applied to evolve an effective distributed resource sharing platform for quality-mind assuring of web services - the so-called quality enable web service architecture. The paper also defines an extensible QoS model and a fuzzy algorithm for services evaluation and selection. The prototype system is designed and used to demonstrate the applicability of the prescribed framework to support user level QoS requirements for disaster management.**

Index Terms— ***Information Quality; Web Service; Disaster Management; Resource Sharing; Platform; Fuzzy Algorithm***

I. INTRODUCTION

Disasters are non-routine events in societies, regions, or communities that involve conjunctions of physical conditions with social definitions of human harm and social disruption. Natural, technological, and willful (terrorist initiated) sources of disasters all cause dramatic losses of life and property[1]. Disaster management is a multifaceted process aimed at minimizing the social and physical impact of these large-scale events. It is thus not surprising that IT has become a critical tool for facilitating the communications and information-processing activities in managing disasters[2]. Especially important is the role of Web Services and the Service Oriented Architecture (SOA) for computer to computer communication in helping to support this capability[3]. SOA is a conceptual architecture that specifies interoperable, modular, loosely-coupled, self-contained and self-describing applications, systems or services that interact only at well-defined interfaces[4]. Analogously, web services are loosely-coupled, interoperable and modular network addressable application modules which can be invoked across heterogeneous networks to access and process information[5].

Loose-coupling means that atomic web services can be deployed, deprecated, or reconfigured independently, for example to adapt to changes in technology or to achieve performance targets, without need to modify other web services or infrastructural components. Interoperability and modularity enable dynamic chaining of disparate web services to create more elaborate composite web services that are capable of value-added services[6]. Web services offer a framework for modular composition of evolvable information systems.

However, most of current web service implementations do not guarantee the levels of information quality delivered to their users. At present, UDDI is just a registry database that allows clients to look for web services based on their functionality but not information quality (IQ) property. IQ, which defines quality of information such as completeness, accessible, timeliness and accurate, is vitally important to disaster response situations that the answers sought are expected in near real-time. Moreover the information should be accurate otherwise it will lead to suboptimal decisions and, as a consequence unnecessary loss of life, property and livelihoods. The lack of adequate IQ support has impaired the adoption of these web services[3].

In this paper, we build a distributed resource sharing platform that supports emergency officers in digital communication and coordination for emergency management and response, and focus in particular on incident command and inter-responder communications. The need for such a system is dramatic, driven by the earthquake of Wenchuan County in Sichuan province of China and the inadequacy of current emergency response systems. This paper presents a platform for better disaster management by assessing information quality provided by web services using the geospatial data

doi:10.4304/jsw.4.9.915-924

quality metrics, which is novel step towards the building geospatial web of trust. Then IQ assessing model for disaster management is given in detail. These will enable county emergency officers to make more accurate, timely and all-around decisions before, during, and after an earthquake situation.

The significance of our work is to promote IQ support in web services so that clients may receive a consistent service level regardless of other competing requests on the same server. The fuzzy algorithms presented for services evaluation and selection can be applied in both legacy geo-information systems and IQ-enable web services.

The rest of the article is organized as follows. Section II reviews related work on IQ and disaster management. Section III presents the IQ-enable web service (IQEWS) architecture which is defined In our previous work[7] and disaster services of sharing platform. Section IV describes a QoS model for IQ metrics and proposes the assurance algorithm to implement QoS metrics in quality-mind web service architecture. Finally, the prototype system that can improve IQ of remote sensing images is reported.

II. . RELATED WORK

Information Quality Dimensions

Snavely constructs a four level hierarchical framework of IQ assessment by the American Accounting Association [8]. Feltham develops formalisms for three quality related attributes of information systems based on Statistical Decision Theory and Information Economics [9]. Taylor proposes five quality dimensions [10].

The most systematic approaches to categorizing data quality dimensions are the ones proposed in papers [11, 12], they propose a two level framework of data quality dimensions comprised of 4 data value quality dimensions. Motro and Rakov use a theoretical approach and proposed two general dimensions for estimating the quality of data in relational databases [13]. English defines an information quality dimensions taxonomy consisting of two categories and fifteen dimensions [14]. Huang, Lee, and Wang distinguish between three different approaches: the intuitive, systematic, and empirical one [15].

Zhu & Gauch take an intuitive approach and propose a model of Web page quality assessment [16]. Loshin proposes four major categories of data quality dimensions [17]. Rieh and Belkinlist criteria of information quality measurement [18]. Eppler proposes his framework contains four levels and sixteen dimensions [19]. Su and Jin present a definition approach of IQ that is based on characteristics of enterprise activities precedence relationship between them [20, 21].

We summarize the discussion above in Table 1.

TABLE I.
CLASSIFICATION OF IQ DIMENSIONS

Lite	Taxonomy	Category			
[22]	Hierarchical	Intrinsic	Contextual	Representational	Accessibility
[23]	Ontological	External	Internal	System related	Data-related
[24]	Source for IQ metadata	Subject	Object	Process	
[25]	Semiotics	Syntactic	Semantic	Pragmatic	
[26]	Product and service	Sound	Dependable	Useful	Usable
[27]	Sequence of using data	Accessibility	Interpretability	Relevance	Integrity

. An Overview of Disaster Management

Disasters are normally categorized according to the cause. For example natural disasters are caused by naturally occurring phenomena e.g. earthquakes, landslides, etc. Similarly, technological disasters are caused by design and management failures in technological artifacts. Other categories of disasters can similarly be defined. Nonetheless, disasters share a number of common features:

- Disasters are a threat to life, property and livelihoods
- Disasters are rapid onset events i.e. the time between the moment it becomes apparent that a disaster event is eminent and the onset of the event is rather short
- Disasters occur with intensities that demand emergency response and external intervention
- A greater proportion of the direct loss occasioned by a disaster is suffered within a relatively short time after onset of the disaster event.

Disaster management concerns the organized efforts focused on eliminating or reducing the risk of a disaster and minimizing the impact of the disaster when it happens. The process of disaster management consists of four broad phases; disaster mitigation, disaster preparedness, disaster response, and disaster recovery[2]. These phases can generally be grouped into pre-disaster and post-disaster phases. Pre-disaster phases are disaster mitigation and preparedness and generally concern activities that take place before a disaster event happens. In contrast, post-disaster phases concern activities that take place after a disaster event. The post-disaster phases are disaster response and disaster recovery.

SOA & Web Services

Service-Oriented Architecture (SOA) is a way of designing, developing, deploying, and managing systems. SOA is gaining momentum today due to its unique nature of enabling organizations to make their emergency decisions first and have those decisions supported by technology, rather than making emergency decisions determined or constrained by technology[28].

One of the many distinctions between traditional systems and SOA-based systems is that in traditional software development, functionality is built into the systems so when a particular functionality is used in a number of different applications, the code is often rewritten. Therefore slightly different versions of a piece of code exist around an organization and when there is a requirement change, all of this code needs to be updated. SOA lets you take this functionality and put in a service and make the service available across a number of applications.

The most common way of implementing SOA is using web services. However, there are other mechanisms as well, such as RPC, DCOM, and CORBA. In this paper, web services are considered as the standard underlying technology to implement SOA. The distributed nature of web services, having XML coding of data passing between services using SOAP, and the stateless feature of HTTP protocol all make this form of SOA-based systems different than traditional component-based systems -and these differences all have advantages for disaster management[29].

Table 2 depicts the web services protocol stack. At the lowest level of the protocol stack is the TCP/IP network service; the other web services build upon it. The next level above that is the transport protocol, which is mainly Hyper-Text Transfer Protocol (HTTP). This means that all messages are transported via text. Above the transport based service is the XML Messaging service, which allows web services to communicate with each other. SOAP is the most common message protocol nowadays, although another protocol called REST is gaining momentum. Above that is a description layer of WSDL (web service description language). This is the piece that will give a reasonable explanation of the functionalities of the web service, and is will be the interface by which the programmers will be able to talk to the services.

TABLE II.
WEB SERVICES PROTOCOL STACK

Metadata Exchage	Composition	BPEL, WS-Notification
	Quality of Service	SLA, WS-Security, WS-Reliable Messaging
UDDI	Description	WSDL, WS-Policy
	Messaging	XML, SOAP, WS-Addressing
	Transports	HTTP , HTTPS, SMTP
Network Level	Application	TCP/IP

A few of research activities have been done in the area of how to realize the IQ support during disaster management. There is no web service based architecture to dealing with the information quality assurance in sharing platform. Consequently, the assurance algorithm to this problem is, to the best of our knowledge, novel.

III. Architecture and Web Services

Architecture of Sharing Platform

In this section, we introduce an information quality enable web service (IQEWS) architecture. It has three entities: Server, IQ Broker and Client (Fig. 1). They work together to accomplish the desired quality of information that a user wants.

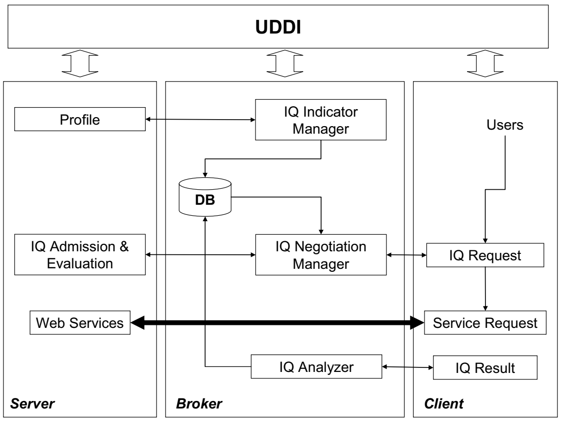


Figure 1. IQEWS Architecture

1) Server

a) Profiie: Besides providing service functionalities information to brokers, a server also provides IQ information about its services. Currently there are two classes of servers providing web services.

Services from the first class are not built with IQ support, referred as legacy servers in this article. There is no information quality level concept in legacy servers; all clients are treated equally and scheduled using native schedulers in the operating system. Many of today's Web servers belong to this class. For these legacy servers, the only IQ information a broker can get is the current server load.

The other class of servers are built with IQ support and called IQ servers. They have the ability to assign different amount of system resources to different clients according to their IQ requirements. The IQ information provided by IQ servers includes quality levels with the timeliness, accuracy and authority in each quality level.

b) IQ Admission & Evaluation: Another important function of a server is the admission control. After a broker selects a service (and quality level if available) for a client, it sends a request to the server for confirmation (admission). The server admits a request only when it can reserve a sufficient amount of resource to achieve the desired quality. If *not, the broker should contact another server to make the request.*

Servers are designed with the IQ evaluation mechanism in order to prevent irrespective clients from consuming more resources than what they have asked for. Clients can only consume their reserved share of system resources.

2) IQ broker

An IQ broker receives clients' functional and IQ requirement requests and identifies qualified services for them. Its main components include IQ information manager, IQ negotiation manager and IQ analyzer.

a) IQ Information Manager: The IQ information manager collects the information required for IQ negotiation and analysis. It checks with the UDDI registry to get the server information and contacts servers for IQ information such as server contend and quality levels. This work is done periodically to keep the broker's information consistent and up to date with UDDI. The collected information is placed in the broker's database. It also maintains the historical IQ information generated by the IQ analyzer.

b) IQ Negotiation Manager: The IQ negotiation manager is the core of IQ Broker. It conducts service selections and service establishments for clients. After receiving the client's functional and IQ requirements, the IQ negotiation manager searches through the broker's database to look for qualified services (and quality levels if available). If more than one candidate is found, a decision algorithm is used to select the most suitable one. The IQ information from both server and IQ analyzer will be used to make the decision.

c) IQ Analyzer: After a web service task is completed, a client may summarize the IQ experience and send them to IQ analyzer in the broker. The IQ analyzer inspects the raw data to produce the statistical information about the service such as information completeness, accessible, timeliness and accurate. All this information will be stored in the broker's database as the historical IQ information, and used by the IQ negotiation manger during the service selection and decision phase.

3) Client

As the end users of web services, clients send their service request and IQ requirements to a broker and ask the broker to select a server for them. Clients also collect the IQ result information after each service call and send them to the broker.

Disaster Service Application

Following several interviews with *spatial data* holders in China, it was decided to focus the application on a real past disaster situation: "Great Sichuan Earthquake" which caused 69,181 known deaths on 12th May 2008. To present the application, we follow the SOA introduced in the previous section.

4) Legacy System Layer

The Institute of Scientific and Technical Information of China (iSTIC) aggregates data and functionalities from three structurally independent and heterogeneous, real world sources:

- *WeatheMan*: a national weather satellite center which provides environmental resources and in particular weather forecast data.
- *DataPool*: collaboration between National Disaster Mitigation Center and Resources Satellite Application Center, which has created a single corporate spatial data warehouse. As can be expected DataPool contains a wide range of data including data for roads, administrative boundaries, buildings, and Ordnance survey maps, as well as environmental and social care data. Within the application we used building related data to support searches for suitable rest centres.
- *InfoAgent* is an Instant Messaging client facilitating lightweight communication, collaboration, and presence management built on top of the instant messaging protocol Jabber. The InfoAgent client can be accessed on standard PCs, as well as on PDAs and on mobile phones which in an emergency situation may be the only hardware device available.

5) Service Abstraction Layer

We distinguish between two classes of services: data and smart. The former refers to the three data sources introduced above, and are exposed by means of web service:

- *Weather service*: this service provides weather information over a specific rectangular spatial area.
- *Emergency Planning services*: using the *DataPool* data each service in this set returns detailed information on a specific type of rest centre within a given circular area. For example, the 'getHospitals' web service returns a list of relevant hospitals.
- *InfoAgent services*: these services allow presence information on online users to be accessed.

Smart services represent specific emergency planning reasoning and operations on the data provided by the data services. In particular, we created a number of IQ assurance services that manipulate GIS data according to emergency specific requirements semantically described (e.g. emergency service centers with air conditioning system, hotels with at least 40 beds, easier accessible hospital, etc.). The criteria used were gained from our discussions with the emergency governors.

6) Semantic Web Service Layer

The following ontologies reflecting the client and provider domains were developed to support Web Service Modeling Ontology (WSMO) descriptions:

- Domain Ontology: it represents the concepts used to describe the services attached to the data sources.
- HCI Ontology: it is composed of Human-Computer Interaction (HCI) and user oriented concepts, and allows lowering from the semantic level results for the particular interface which is used.
- Archetypes Ontology: it aims to provide a cognitively meaningful insight into the nature of a specialized object.
- Spatial Ontology: it describes GIS concepts of location, such as coordinates, points, polygonal areas, and fields. It also allows describing spatial objects as entities with a set of attributes, and a location.
- Context Ontology: it allows describing context *n–tuples* which represent a particular situation. In the emergency planning application, context *n-tuples* have up to four components, the use case, the user role, the location, and the type of object.

7) Presentation Layer

The application user interface is based on Web standards. XHTML and CSS are used for presentation and JavaScript is used to handle user interaction together with AJAX techniques to communicate with IQA.

. *WSMO descriptions for Sharing Platform*

A small portion of emergency management workflow represented in terms of SWS descriptions is shown in Figure 1.

Get-Polygon-Spatial-Data-with-Inspector-Goal represents a request for available images within a disaster area. The user specifies the requirements as a target area, a sequence of at least three points (a polygon), and a shelter type (e.g. hospitals, tents). As mentioned above the set of Emergency Planning services each return potential images of a specific type with a circular query area. The obtained results need to be inspected in order to return only images correlated to emergency-specific requirements (for example an earthquake). From a SWS point of view the problems to be solved by this particular portion of the SWS layer included:

- Discovering the appropriate Emergency Planning service; each definition is linked to the Get-Spatial-Data-Goal by means of a unique Get-Web-Service Agent (shown as gWS).
- Meditating the difference in area representations (polygon vs. circular) between the goal and web services;
- Composing the retrieve and inspect data operations. Below we outline how the WSMO representations in Fig. 2 address these problems.

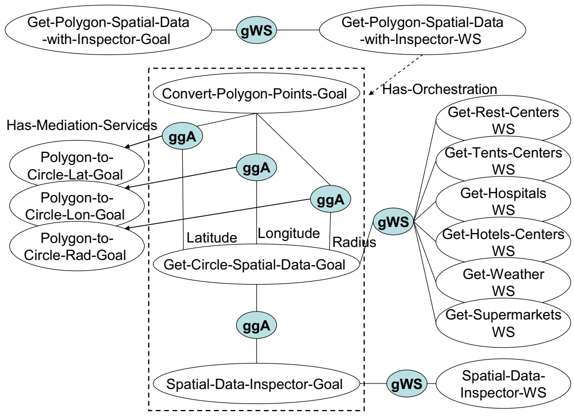


Figure 2. WSMO descriptions for Sharing Platform

Get-Polygon-Spatial-Data-with-Inspector-Goal is associated with a unique web service that orchestrates, by simply invoking three sub-goals in sequence. The first gets the list of polygon points from the input; the second is Get-Circle-Spatial-Data-Goal; finally, the third invokes the smart service that inspects the list of spatial data. The first two sub-goals are linked by means of three get-goal Agent (depicted as ggA) that return the centre, as a latitude and longitude, and radius of the smallest circle which circumscribes the given polygon.

WS offers a number of promises such as cost-efficiency, reusability, flexibility, agility, and adaptability. However, a significant barrier to the widespread adoption of WS-based systems for disaster management is how to cope with the problem of information quality[30].

IV. . QOS MODEL AND PROPOSED ALGORITHM

Quality of service (QoS) is the totality of characteristics of a service delivered by collaborating web services that bear on its ability to satisfy stated or implied needs in an application context.

We have proposed a set of taxonomy of IQ dimensions that would allow us to evaluate the QoS variance caused by these causes in a systematic and meaningful way, to incorporate into our base framework (i.e., IQEWS). They augment the generic QoS metrics to allow emergency officers more precise control over their query. There are many advantages in using this approach. Traditional web services provide the modularity but take away the ability to precisely control the use of the info. To get around this problem, one can retrieve a large amount of info from a server and perform offline filtering or various types of modifications themselves. However, this is a very inefficient and time-consuming procedure since a lot of processing is done post hoc. The QoS metrics allow clients to restrict the types of servers it is interested in before any processing on the info is done. These QoS metrics are added as WSMO descriptions to our Resource Description Framework (RDF) described in our previous paper[28].

. QoS Metrics

Quality metrics are organized into a hierarchy allowing clients to visualize them at different levels of detail. Quality data is aggregated into the dimension hierarchy from the most detailed level to the more general ones. Members of this dimension (i.e., quality indicators) can either provide information regarding syntactic (e.g., format accuracy), semantic (e.g., thematic accuracy) or pragmatic (e.g., data producer reputation) aspects of the dataset. For instance, members can be horizontal positional accuracy, completeness, timeliness of acquisition or accessibility (see Table 3).

TABLE III.
TAXONOMY OF IQ DIMENSIONS

Quality	Information Quality	
	Indicator	***Description***
Syntactic Level	Omission	Is the data omitted in metadata?
	Concept	Is the concept of metadata accuracy?
	Domain	Is the information processed and delivered without domain?
	Format	Is the metadata in right format?
	Topology	Is the data in right toplogy?
	Attribute	Are the attributes of metadata accuracy?
Semantic Level	***Completeness***	Is the scope of information adequate?
	Consistency	Is the information to the point, void of unnecessary elements?
	Accuracy	Positional
		Temporal
		Thematic
Pragmatic Level	ISO	Can the information be directly applied? Is it useful?
	Expert	Is the information understandable or comprehensible to the target group?
Ontological Level	Image reputation	Is there a continuous and unobstructed way to get to the information?
	Resolution	Have both image and vector data appropriate resolution properties.
	Spatial extent	Can all of the information be organized and updated on an on-going basis?
	Temporal extent	Is the information processed and delivered rapidly without delays?
	Timeliness	Can the infrastructure match the user's working pace?

Based on the above definitions, we select four typical quality metrics for a perfect tuple T. $|T|$, $|T_A|$, $|T_I|$ $|T_M|$, and $|T_C|$ denote the cardinalities of the sets T, T_A, T_I, T_M, and T_C, respectively.

Accuracy of T, measured as $\alpha_C = |T_A|/|T|$, is the probability that a tuple in T accurately represents an entity in the real world.

Inaccuracy of T, measured as $\beta_C = |T_I|/|T|$, is the probability that a tuple in L is inaccurate.

Mismembership of T, measured as $\gamma_C = |T_M|/|T|$, is the probability that a tuple in T is a mismember.

Incompleteness of T, measured as $\chi_C = |T_C|/(|T| - |T_M| + |T_C|)$, is the probability that an information resource in the real world is not captured in T. Because T_A, T_I, and T_M constitute T, we have $0 \le \alpha_C\ , \beta_C, \gamma_C \le 1$ and $\alpha_C + \beta_C + \gamma_C = 1$.

. QoS Assurance Step

During the first Phase, the query profile of the user is submitted to a broker for determining the functional matches from the set of published services. The Broker returns a set of functionally similar services if the query to be solved involves single server; otherwise returns a dynamically composed service if the query requires service orchestration.

To incorporate the QoS values, we add a step to the IQ assurance. The step operates as follows.

1) Servers publish profiles to Broker

2) Generate query profile

3) Find semantically similar services for the query using the functional parameters: input and output parameters

4) If there is no such service from step 3, dynamically compose complex service using the services registered using IQEWS Fuzzy Computing [7]

5) Sort the Functionally Similar Semantic Services using the QoS Algorithm (see Fig. 3)

6) Return the URI of the best Service from step 5 to user

We will describe the approach developed by us for performing the Step 5 of the service discovery algorithm. The QoS selection differs when we have a dynamic composition that involves computing the aggregate QoS values of the services dynamically, which is also one of our contribution in this paper.

QoS semantic algorithm

The Environment is comprised of registered Servers $S_1, S_2 \dots S_j$, Clients $C_1, C_2 \dots C_i$, Brokers $B_1, B_2 \dots B_k$. In our interaction model we assume only one broker. We employ special assuring services which get the user results on QoS relevance feedback which are called Quality Analyzers QA_1, QA_2 ... QA_l. Broker can also additionally act as Quality Analyzer.

User Query List UQ = $\{(uq_1, r_1), (uq_2, r_2)\ldots(uq_n, r_n)\}$
TargetMatch // Number of content matches required
G_{val} = 0 for all services
1. $\forall$ S_j in Functional Match Set F
2. dist = 0.0
3. $\forall$ qi:qi=quality content in uq
4. If q_i matches with a content in sp_j
5. content-match = content-match +1
6. dist += $|r_i - q_i|$
7. If content match >= TargetMatch then
8. G_{val} = diff/content-match
9. Return F sorted by ascending order of G_{val} scores.

Figure 3. QoS semantic algorithm

Servers publish their QoS values (sp_1, q_1), (sp_2, q_2) … where (sp_i, q_i) are vector pairs of contents and their values. Users provide the QoS requirements for every query as (uq_1, r_1), (uq_2, r_2) … where (uq_i, r_i) are vector pairs of contents and user required values. QoS vector values q_i, r_i are fuzzy values which are in the range [1, 9]. 1 is the worst QoS support available and 9 is the best support available for that QoS parameter.

In the First Phase, for each registered Server j in the functional match set F of the Query Q, a G_{val} is evaluated using the advertised QoS parameters. G_{val} is the Manhattan distance averaged over the number of quality content matches between the user requirement and the server advertised QoS values.

The QoS Similarity Match Algorithm is illustrated in Fig. 3 to select a set of services. All the Servers are set with G_{val} = 0 and the target content matches between query and server content is set to a constant (is 5 in our platform). In Step 1 for every service Sj returned from Functional Set F returned from Broker. The similarity between r and q vectors is measured using Manhattan Distance. For every quality content q_i in Vector uq, if there is a content-match (exact, subsumes) with content in sq_j, content-match is incremented. The diff is updated for this match, In step 7 we check if there are at least target number of matches for meeting the user requirement, we compute the G_{val} as average distance over the content matches in step 8. Step 9 returns the F in ascending order of G_{val}.

In the second phase of the QoS measurements, we use the user feedback to update the advertised QoS parameters of the selected service S_i as follows. All the user reports pertaining to the similar query Q posed is aggregated here in this phase. The user feedback list UF of every user is evaluated as shown in Table 4.

TABLE IV.
QOS FEEDBACK ALGORITHM

No	Steps
1	Aggregate Feedback Vector FV; For every Service Provider S_j
2	Read every User Feedback List U_i. F received = $\{(uq_1, f_1), (uq_2, f_2) \ldots (uq_n, f_n)\}$ where i=1:n
3	FV = FV + $\{(uq_1, f_1), (uq_2, f_2) \ldots (uq_n, f_n)\}$
4	End For
5	FV_{avg} = FV / n
6	Update each QoS parameter sp_j of S_j as $q_j = q_j (1 - Fv) + q_j$

In our model, user results are considered to be credible as only authenticated users can log on to the platform for service discovery. We assume that the Servers who publish their service descriptions to the broker do not cancel their registration during the interaction for at least a certain number of iterations to facilitate the catching of ropy providers.

V. DEVELOPING THE PROTOTYPE OF DISASTER MANAGEMENT SYSTEM

Based on the IQEWS architecture, web services, QoS model and semantic algorithms to assure IQ, we developed a prototype to build Spatial data sharing platform for disaster management. The prototype implements several functions such as Product Management, Information Assurance, Data Download, Retrieval and Query, User Behavior Management. Fig. 4 illustrates the framework.

Prototype framework

The prototype was developed using ESRI ArcSDE software driven by a multiple users interface developed in Java.

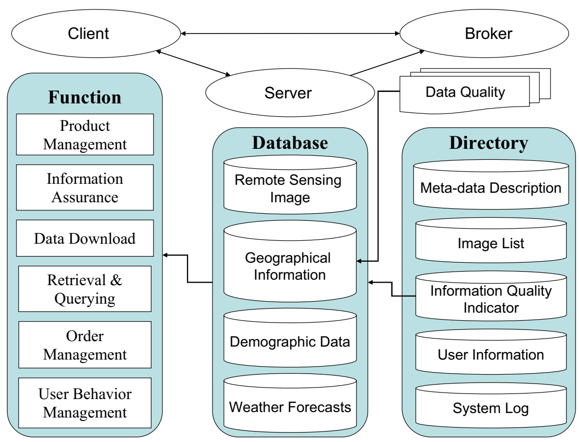


Figure 4. Prototype Framework

Indicator management

Not all users evaluate quality based on the same criteria. Based on the ISO 19113 standard, quality indicators were defined and stored hierarchically within a relational database. The user can eventually adapt some items further or add more metadata about the indicators. One may select among different graphical representations to illustrate each indicator (See in Fig. 5 and Table 3.)

Indicator values are based on the spatial extent of the map being displayed to the quality expert. Indeed, if the user zooms in or pans towards a particular region of interest, quality indicators are recalculated for objects located within the disaster area.

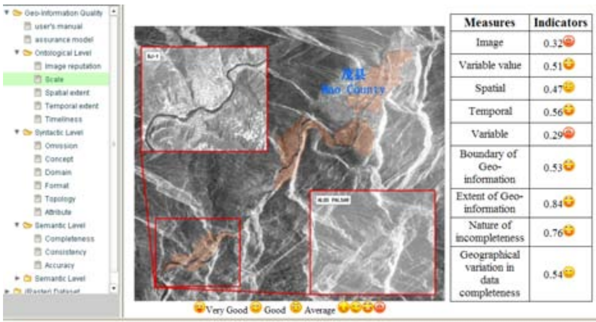


Figure 5. Indicators selection tool with Image in disaster area

Navigation into IQ Data

Using the prototype described in the previous section, information quality experts can improve their knowledge of IQ through the use of different navigation tools. Table 5 illustrates the benefits of such a system through different questions a user may have regarding IQ and the different tools offered by the system to help in answering those questions.

TABLE V.
MAP QUALITY PROBLEMS

Problems	%	Quality problem incidents counted
Ambiguity	89	What is the avarage quality of the image displayed on your screen?
Inaccuracy $\beta_R =$	25	What image quality characteristics can be a problem according to the task defined for disaster management
Incomplete ness	100	Positional accuracy looks problematic, but is it spatially heterogeneous?
Inconsisten cy	26	Inconsistent formatting or representation of the same elements
Resolution	92	How good is the positional accuracy of this specific object class?
Expert	86	What about the quality of the school building in particular?

In our prototype, the quality panel can include up to six indicators. From Fig. 6, the IQ metric values of different layer for assure images are shown in the prototype system. In addition to the levels of detail within the image, this approach also allows users to explore image quality along a quality indicator hierarchy. For instance, in the example of Table 6, a user looks first at the higher-level indicators. He realizes that 'General Quality' is only average (i.e. yellow) because of the lower 'Internal Quality'. He then drilldown into the 'Internal Quality' to see its sub-indicators

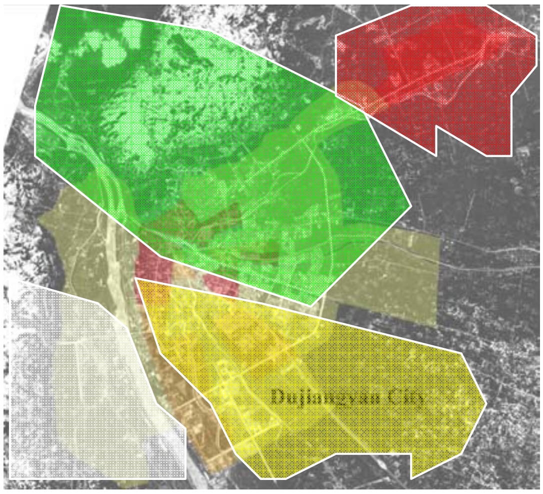


Figure 6. Different Color Representations for Image Quality in Disaster Management

TABLE VI.
IQ METRICS OF ASSESSED IMAGE

Dimensions	Dataset	Band	Attribute Value	Object Instance
accuracy	0.51	0.32	0.54	0.46
		0.19	0.62	0.38
Resolution	0.68	0.66	0.78	0.83
			0.88	0.83
		0.89	0.48	0.94
completeness	0.67	0.53	0.59	0.74
		0.84	0.63	0.56
consistency	0.79	0.86	0.88	0.77
		0.64	0.73	0.66

At this second level, he can wonder why the 'Positional Accuracy' indicator is only average, and then drill-down on 'Positional Accuracy' to obtain more detail. He finally arrives at the last level of detail available in our prototype and sees that the problem comes from the 'Spatial Resolution'.

VI. CONCLUSION

In this paper we defined quality of service and presented a QoS model for disaster management in the context of broker- based web service. We defined QoS as comprising of qualities related to the operational behavior of a service and the information quality that the service delivers. We introduced the notion of an IQEWS architecture that is a middleware platform that enables QoS-driven composition of web services. The QoS assurance step and semantic algorithm enables evaluation, selection and execution of web services in compliance with user requirements. They provide the distributed resource sharing platform with QoS-driven composition of web services. The paper illustrated how desired QoS in prototype system of disaster management can be supported.

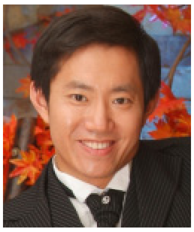
ACKNOWLEDGEMENT

We would like to thank NNSFC (National Natural Science Foundation of China) for supporting Ying Su with a project (70772021, 70831003).

.REFERENCES

[1] E. L. Quarantelli, "Where We Have Been and Where We Might Go," in What Is A Disaster?New Answers to Old Questions E. L. Quarantelli, Ed. London: Routledge, 2005, pp. 234-273.
[2] R. R. Rao, J. Eisenberg, and T. Schmitt, Improving Disaster Management: The Role of IT in Mitigation, Preparedness, Response, and Recovery. Washington, DC: National Academies Press, 2007.
[3] L. Bahler, F. Caruso, C. Chung, G. Collier, B. Falchuk, J. Micallef, and G. Patton, "Improving emergency management with Web services," Orlando, FL, USA, 2004, pp. 162-7.
[4] Y. Shen, J. Zhang, and Y. Fan, "Multi-index cooperative mixed strategy for service selection problem in service-oriented architecture," Journal of Computers, vol. 3, pp. 69-76, 2008.
[5] C. Zhou, L.-T. Chia, and B.-S. Lee, "QoS-Aware and Federated Enhancement for UDDI," International Journal of Web Services Research, vol. 1, pp. 58-85, Apr-Jun 2004.
[6] C. K. Fung, P. C. K. Hung, R. C. Linger, G. Wang, and G. H. Walton, "A Service-Oriented Composition Framework with QoS Management," International Journal of Web Services Research, vol. 3, pp. 108-124,126-127,129-132, Jul-Sep 2006.
[7] Y. Su, j. Peng, and Z. Jin, "Assuring Information Quality in Web Service Composition," in 5th International Conference on Service Systems and Service Management (ICSSSM'08), Melbourne Australia, 2008, p. 4598537.
[8] H. Snavely, "Accounting Information Criteria," The Accounting Review, vol. 42, pp. 223-232, 1967.
[9] G. A. Feltham, "Information Evaluation," Studies in Accounting Research, vol. 5, 1972.
[10] R. S. Taylor, Value-Added Processes in Information Systems. Norwood, NJ: Ablex Publishing, 1986.
[11] C. Fox, A. Levitin, and T. Redman, "The Notion of Data and Its Quality Dimensions," Information Processing & Management, vol. 30, p. 9, Jan/Feb 1994.
[12] A. Levitin and T. Redman, "Quality Dimensions of A Conceptual View," Information Processing & Management, vol. 31, p. 81, Jan/Feb 1995.
[13] A. Motro and I. Rakov, "Estimating the Quality of Data in Relational Databases," in 1996 Conference on Information Quality, 1996.
[14] L. P. English, Improving Data Warehouse and Business Information Quality Methods for Reducing Costs and Increasing Profits New York: Wiley, 1999.
[15] K.-T. Huang, Y. W. Lee, and R. Y. Wang, Quality Information and Knowledge. Upper Saddle River, N.J. : Prentice Hall PTR, 1999.
[16] X. Zhu and S. Gauch, "Incorporating Quality Metrics in Centralized Distributed Information Retrieval on the World Wide Web," in the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2000.
[17] D. Loshin, Enterprise Knowledge Management: the Data Quality Approach. San Diego: Morgan Kaufmann, 2001.
[18] S. Y. Rieh and N. J. Belkin, "Understanding Judgment of Information Quality and Cognitive Authority in the WWW," Journal of the American Society for Information Science, vol. 53, pp. 145-161, 2002.
[19] M. J. Eppler, Managing Information Quality Increasing the Value of Information in Knowledge-Intensive Products and Processes. Berlin: Springer, 2003.
[20] Y. Su and Z. Jin, "A Methodology for Information Quality Assessment in the Designing and Manufacturing Processes of Mechanical Products," in 9th International Conference on Information Quality MIT, USA: ICIQ, 2004.
[21] Y. Su and Z. Jin, "A Methodology for Information Quality Assessment in the Designing and Manufacturing Process of Mechanical Products," in Information Quality Management: Theory and Applications, L. Al-Hakim, Ed. USA: Idea Group Publishing, 2006.
[22] R. Y. Wang and D. M. Strong, "Beyond Accuracy: What Data Quality Means to Data Consumers," Journal of Management Information Systems, vol. 12, p. 5, Spring 1996.
[23] Y. Wand and R. Y. Wang, "Anchoring Data Quality Dimensions in Ontological Foundations," Association for Computing Machinery. Communications of the ACM, vol. 39, pp. 86-95, Nov 1996.
[24] F. Naumann and C. Rolker, "Assessment Methods for Information Quality Criteria," in International Conference on Information Quality (MIT IQ Conference), MIT, 2000, p. 10.
[25] M. Helfert and R. Radon, "An Approach for Information Quality Measurement in Data Warehousing," in International Conference on Information Quality (MIT IQ Conference), MIT, 2000, p. 10.
[26] B. K. Kahn, D. M. Strong, and R. Y. Wang, "Information quality benchmarks; product and service performance," Association for Computing Machinery. Communications of the ACM, vol. 45, p. 184, Apr 2002.
[27] M. Bovee, R. P. Srivastava, and B. Mak, "A conceptual framework and belief-function approach to assessing overall information quality," International Journal of Intelligent Systems, vol. 18, pp. 51-74, Jan 2003.

[28] Y. Su, Z. Jin, and J. Peng, "Assuring information quality in e-Science," in International Parallel and Distributed Processing Symposium - IPDPS 2008, Miami, Florida USA, 2008, p. 4536206.
[29] T. Parveen and S. Tilley, "A research agenda for testing SOA-based systems," Piscataway, NJ 08855-1331, United States, 2008, pp. 355-360.
[30] Y. Su, Z. Jin, and J. Peng, "Modeling Data Quality for Risk Assessment of GIS," Journal of Southeast University (English Edition), vol. 24, pp. 37-42, June 2008.



Ying Su is an associate professor in the Center for Resource Sharing Promotion in Institute of Scientific and Technical Information of China. He received a B.S. degree at the Department of Mechanical Engineering in 1992 from Hohai University, Jiangsu Province, China. In 1999, he received an M.S. degree in mechanical engineering from Southeast University, Nanjing, Jiangsu Province. In 2006, he received a PhD degree from the Department of Precision Instruments and Mechanology at the Tsinghua University, Beijing. From 1992 to 1996, he was an engineer and technologist at an iron and steel plant. From 2006 to 2008, he has worked at School of Economics and Management as postdoctoral fellow in Tsinghua University. Dr. Su is an author of many scientific and research papers. His current research interests include Information Quality, Discrete-Event Simulation, Green Supply-Chain Management, Business Process Modeling, and Enterprise Resource Planning.

Jie Peng is a professor of information resource management in Institute of Scientific and Technical Information of China (ISTIC). She received her BS from South China University of Technology in 1986, her MBA from the City University of Seattle in 2007. She worked in ISTIC from 1986. Her research interests include Information Resource Management, Information Integration, Scientific and Technical Resource Management. She has been in charge of many research and development projects.

Zhanming Jin is a professor of strategic management in Department of Business Strategy and Policy at Tsinghua University. He received his BS from Jilin University in 1980, his MS from the Chinese Institute of Agriculture and Mechanic in 1986, and a PhD from Chinese University of Mining and Technology in 1989. He worked as postdoctoral fellow from Tsinghua University in 1991. His research interests include Strategic Management, Strategic Selection under Green supply-chain management, Military Strategy and Enterprise Competition. Prof. Jin has published the results of his research in over 120 refereed journal and conference papers.