JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 35, 487-505 (2019)
DOI: 10.6688/JISE.201905_35(3).0001

Disaster Scenario and Record Capture System

HAN WANG[1], CHEN-YEN TANG[1], CHIN-JUNG TAO[2] AND HENG-CHIH HSIEH[1]
[1]*Institute of Information Science*
Academia Sinica
Taipei, 115 Taiwan
[2]*Department of Computer Science*
National Tsing-Hua University
Hsinchu, 300 Taiwan
E-mail: {hollywang; kennethtang; hunter205}@iis.sinica.edu.tw[1];
kevintao5566@gmail.com[2]

Typical reports on past disasters contain valuable lessons that can help us improve our preparedness and response strategies and operations, especially with the help of modern analysis, simulation and visualization tools. However, disaster historical records are typically written in natural language for human consumption. They sometimes contain inconsistent data and information, which take time and effort to uncover. Extracting from the reports for further analysis by tools is almost impossible. DiSRC (Disaster Scenario and Record Capture) system is being designed as a solution of these problems. The system contains a cloud-based authoring system together with Disaster Record Capture Tools (DiReCT) running on mobile devices. Targeted users of DiReCT are government agents who are responsible for contributing data to be included in disasters records and volunteers trained by the responsible agencies. They can use DiReCT to capture observation data on the dynamics and effects of the disaster on places and people in machine readable form. The quality of the captured observation data is controlled and assured by both the capturing tools in real-time and by tools in the cloud. The authoring system can assist authors to write reports on the disaster based on the captured data and information. Disaster records produced with the help of DiSRC system can be read and processed by search, analysis, simulation and visualization tools for purposes such as developing better response strategies, providing decision support during emergencies, post-disaster analysis, and so on.

Keywords: disaster, electronic data capture, authoring system, quality control, mobile device

1. INTRODUCTION

In recent years, one sees the emergence of models and tools for capturing business scenarios, authoring patient records, storybooks, e-learning course wares, and so on. These and related information technological (IT) advances have not been exploited to ease the tasks of capturing data and information on disasters, assuring and controlling data quality during preparedness and response times, and authoring and use of disaster records afterwards. Today, records on disasters, including records on earthquakes from Central Weather Bureau [1] and other sources and records on weather-related disasters provided online by NOAA Climate Data Center [2], are still written mostly for human consumption. They are in natural language text accompanied by tables, graphs, *etc.* Such records stitch together data and information contributed by government agencies, NGO's,

Received June 9, 2018; accepted November 29, 2018.
Communicated by Chi-Sheng Shih.

mass media, and general public and drawn from multiple databases and web services. One can easily see from reading the records the immense efforts required to produce them.

Disaster records in their current forms have at least two shortcomings. A serious shortcoming is that it takes immense effort to make good use of the data provided by them. Historical reports on past disasters can teach us valuable lessons. In particular, records on preparedness and response strategies and operations and data on their effectiveness during similar disasters in the past can be used as the basis to develop and tune preparedness and response strategies and action plans for an imminent disaster. The chance of putting in place the best plans in a timely fashion can be significantly improved with the help of modern analysis, simulation and visualization tools. Unfortunately, it is nearly impossible to extract from the existing reports machine readable input needed to use the tools. This shortcoming would not exist if the records were machine readable and data on lessons could be extracted automatically. Another serious shortcoming is that data and information in the records may be inconsistent and incomplete, and such defects typically take time and effort to uncover. Today, quality assurance of observation data is often overlooked [3]. Up until 2013, only 17% of disaster record databases have applied some quality control procedure, and some of them are verified on a semi-annual basis [4]. Without data quality assurance, the validity and reliability of vulnerability and risk assessment based on data are questionable.

These facts motivated the DiSRC (disaster scenario and record capture) system described in this paper. The system contains as components mobile Disaster Record Capture Tools (DiReCT) and a cloud-based authoring system. The former aims to be used during preparedness and response times to capture and record data and events/conditions on the dynamics and development of each disaster and its effects on threatened and affected places and people. The latter aims to assist authors to write reports on disasters based on the captured data and information.

Specifically, the primary users of DiReCT are trained staffs of organizations responsible for contributing data and information included in disaster historical records/ reports. Within the duration of a disaster, their duties include recording observation data and related information. We call these users recorders. (In general, some recorders may be volunteers who were recruited and trained to contribute observation data. They will not be mentioned hereafter since their presence is not important for most of the discussion here.) In all use scenarios, each DiReCT is used by one recorder at a time but multiple recorders may use a DiReCT at different times. Data recorded via each DiReCT is uploaded periodically and opportunistically to the authoring system. The primary users of the authoring system are authors, who write disaster historical records for human readers. The authoring system provides them with a library of diverse tools to enable the integration and analysis of all observation records uploaded from all DiReCT and presents the data to them via web-services, file systems, and so on. The authoring system maintains the records while providing access to past historical records and support the use of tools that generate decision support data, performance of post-disaster analysis, and so on, in short making diverse usages of the available data.

A distinguishing capability of DiSRC is quality assurance (QA) and quality control (QC) of data in disaster historical records. The QC module in DiReCT detects on real-time basic errors such as incorrect range and poor time and space resolutions while data are captured. Whenever possible, the tool helps the recorder to take follow-up actions, such as repeating the observation and revising the capture record or marking the record as defective. The defective records are taken care of by the failure handler in ways specified by the SOP (standard operating procedure) of the QC module. Once a record is uploaded to the authoring system, the system checks the record for global consistency by comparing the record with records captured by other recorders during the current disaster event and records captured during similar past events. In addition, the authoring system provides the system administrators with data and information need to prepare each DiReCT as a part of pre-collection data quality assurance so that the mobile device surely will contain related background information and definitions of all SOP specified by the user's organization for the current disaster.

Following this introduction, Section 2 presents related work. Sections 3 presents an overview of DiSRC design rationales. Sections 4 and 5 present details on DiReCT and the cloud-based authoring system respectively. Section 6 presents the current status of DiSRC system and future work to complete it.

2. RELATED WORK

As described above, DiSRC is consisted of two kinds of components: mobile systems of data capture tools and a cloud-based authoring system. Unlike many systems for similar purposes, DiSRC provides the distinguish capability of controlling of data quality while data is captured and supporting data quality assurance before and after data capture. This section presents an overview of existing technologies in areas closely related to the work on data capture system, authoring system and data quality assurance and control.

2.1 Data Capture System

Over the past few years, a growing number of digital data capture system had been developed. The majority of such systems aim to support crowdsourcing data collections by volunteers and crowd of people in general. Based on different purposes and data source, there are two types of system: crowdsourcing data capture system and electronic data capture system. DiSRC system integrates benefits from both and resembles more closely to systems that are used by professionals for purposes such as capturing data during clinical and field trials, science and engineering experiments, *etc.*

(A) Capture of Crowdsourcing Data

DiReCT is similar to form-based data capture systems. Typically, such system provides tools [5-25] that project administrator can use to generate questionnaires with the help of web-based user interface and send the questionnaires to data collectors' mobile devices. Data contained in completed questionnaires returned by collectors is stored on the system. Through analysis and visualization tools provided by the system, data is presented on maps or in charts and can be easily analyzed. Some of the systems are hazard-specific [15, 16, 20, 21, 23] and others can be used for any type of disasters depending on the questionnaires used by data collectors.

As examples, Damage Tracker [23] is a mobile application that allows the user to capture and annotate scenes of the tornado damages with such things as damage ratings and descriptive note. Ushahidi is a well-known open source crowdsourcing platform [24] that has been used time and again to communicate issues and collect data. Guillén *et al.* [25] utilized Ushahidi to visualize traffic flow in real-time based on data uploaded by drivers.

(B) Electronic Data Capture

Electronic Data Capture (EDC) systems are designed for the data collection in electronic format [26]. Most of them are used to capture data generated from field trials and market surveys by enterprise (*e.g.*, [27, 28, 30, 31]) and medical organization (*e.g.*, [29, 32, 33]). These systems typically provide a user-friendly interface and tools to set up mobile EDC systems and make them ready for data capture. Some of the systems have the functionality for the team leader to manager and check team members' progress, organizing the team's sale activity in a report format and giving the control and visibility to the work the team does in the field [28, 31]. For examples, Socialbungy [28] provides tools for building marketing campaigns and capture leads when running an event or managing an experiential marketing team. With a few expectations, these systems provide sufficient storage on devices so that data collection can proceed without Internet. DiReCT is similar to these systems. Fundamental differences between DiReCT and them include real-time data quality control and the support by authoring system.

2.2 Authoring System

Today, there are authoring systems for numerous types of information products. Examples include multimedia titles, cartons, stories and novels, business records, and disaster records. Typically, programming features of authoring systems are built in but hidden from the user behind buttons and other use interface feature [34, 35]. As examples, authoring tools for development of educational software allow non-programmers to easily create lessons with programming features. E-learning authoring tools allow its users to create multimedia education materials including training scenarios, gaming elements, videos, audios, *etc.* The well-known authoring system, Articulate360 [34], includes Storyline and Rise, plus a slew of other authoring apps. They allow users to develop customized, interactive and create presentation-based course in PowerPoint. Smartbuilder [35] lets users create varieties of actions in action panel by combining triggers, conditions and responses that can be associated with any object. Users can fully customize course interfaces to create unique look and feel and rich scenarios and interactions using an intuitive, icon-based flow chart.

Fig. 1 lists other examples of authoring systems. These tools assist authors in structuring their books and also help authors manage data such as characters, locations, scenes, items, tags and idea in one place. Specifically, E-book authoring tools typically allow the user to add and organize book content with an easily-accessible and intuitive user interface. Documentation tools and content management system enable user to include multiple content versions in different output formats and for different purposes all in the same project. Novel-writing tools [38, 39] for creative writers, novelists and authors help authors to organize their books.

For the purpose of discussion here, one can think of DiSRC authoring system as a framework that provides tools and information for connecting fragmental descriptions of individual scenarios together and putting them in historical background and context, and thus transforming them into publications and information products. As it will be evident in subsequent chapters, it differs from the authoring systems mentioned above in many fundamental ways. An example is its capability for data quality assurance.

Category	Reference	Functionality
E-book	Pressbooks [36]	▪ Provide a user interface for organizing the content. ▪ Edit book in all the formats to publish and deliver.
Documentation tool	Help and Manual [37]	▪ Provide intuitive working environment that supports single and multiple authors editing. ▪ Help authors to generate and edit documentation file with powerful features.
Novel writing tool	Storybook [38] yWriter [39]	▪ Assist author to control multiple plot lines. ▪ Keep overview on the specific key points, such as location, characters,etc.

Fig. 1. Summary of diverse authoring system.

2.3 Data Quality Control and Assurance

Again, DiSRC has the capability to ensure the good quality of captured observation data. Real-time quality control (RTQC) during data captures and the quality assurance (QA) of pre-and-post collection are two main concerns.

(A) Real-time Quality Control

Many tools offer customized field-based verification and basic workflow for real-time quality control. Project administrator can specify input type of each field (*e.g.*, integer, string, or multi-choice) and the time when questions should be visible or invisible during questionnaire design process. These tools automatically check whether each record meets specified quality requirements, including whether each answer is a right data type or any answer is missing. So does DiReCT.

Some tools (*e.g.*, Fulcrum [17] and GDACSmobile [22]) allow administrators to program their own custom quality control logic through REST APIs and code libraries or review manually the observation data and, if needed, request user to take actions such as extending the questionnaire, clarifying uncertain answers, or asking for further information in order to improve data quality. Currently, DiSRC system does not provide these capabilities.

(B) Quality Assurance

Many data mining methods are effective in detecting abnormal information and data to achieve the goal of post-collection data quality assurance [40]. Among exiting methods, outlier detection methods can be used to implement a part of the QA module. As Wand and Wang [41] suggested that outlier should be isolated from other data. The defective data can be detected by this way.

DiSRC system will adopt the outlier detection method proposed by Lin [42]. The method combined two outlier detection algorithms: Local Outlier Factor (LOF algorithm) and Local Distance Outlier Factor (LDOF algorithm). Both LOF and LDOF algorithms calculate a value for each record indicating the likelihood of the record being an outlier. The combined method aims to identify outliers of observation data by checking whether the value of each record calculated by both algorithms is larger than a specific threshold. Depending on how the value for each record is computed and whether values of a large number of records captured are required, the method can be used as a way for the real-time quality control on DiReCT or post-collection quality assurance by the authoring system.

3. DESIGN RATIONALES

To help us explain our design objectives and choices in subsequent sections, Fig. 2 shows a configuration of the proposed DiSRC system. Without loss of generality and for sake of concreteness, the system contains as components a large number of DiReCT (Disaster Record Capture Tools). Each DiReCT is a system of software tools running on a tablet, smart phone or some other mobile platform. The other major component is the authoring system (AS) that runs on a cloud-based server. DiSRC does not support crowdsourcing data collection. However, as the left corner of the figure illustrates, the system and its users can access crowdsourced sensor data such as earthquake intensity data and other types of participatory sensor data [43]. As stated earlier, an objective of DiSRC is to make captured data directly readable by tools. The authoring system collects and organizes observation data captured by DiReCT. Authors who write disaster historical records and reports for human readers can access the data, as well as related scientific and historical data and information via web-services, visualization tools, file systems, and so on. The system provides them with tools for translating data into forms (*e.g.*, spread sheets, tables and graphs) that can be included directly as parts of their reports and the digital formats (*e.g.*, JSON, XML, and RDF) for the tools used for further analysis, simulation and animation.

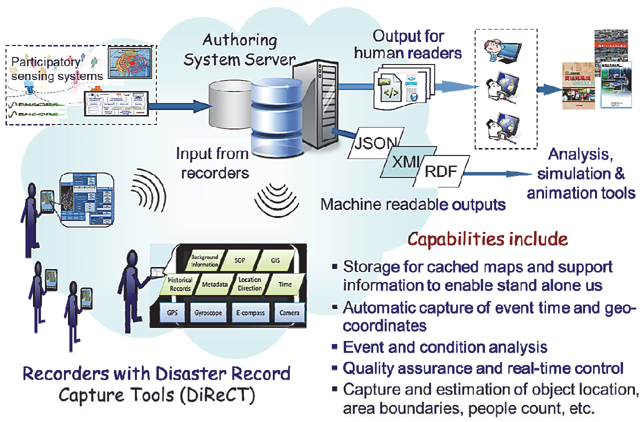
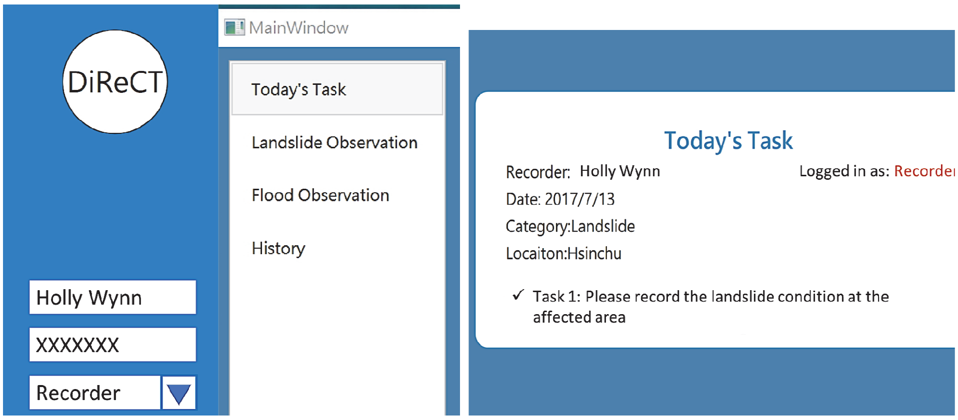


Fig. 2. DiSRC system framework and overview.

3.1 Use Scenario and Graphical User Interface

We take a user-centered approach to design, development and evaluation of the DiSRC system. A way to determine the functionalities and requirements of each system components is to start with likely use scenarios of the component. Functional and non-functioned requirements of the component can be determined through analysis of the scenarios.

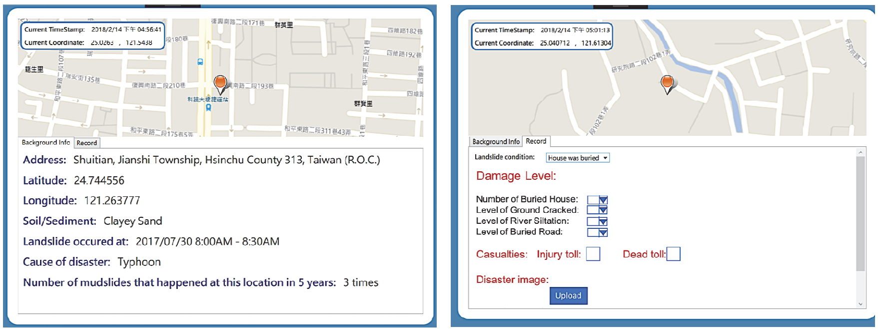
We now use the use scenario depicted in Figs. 3 and 4 to illustrate this point. In this example, staff member Holly Wynn is assigned the task of recording observation data on a landslide caused by a typhoon, Since DiReCT she uses for this task may have been and will be used by other staff members, the device requires her to login as the first step. Fig. 3 (a) shows DiReCT login page. As shown, DiReCT supports role-based access control. The page offers the user choices of recorder, administrator and event manager roles. After being authenticated as a recorder by the Authentication and Authorization (AaA) component of DiReCT, which will be described later, the user is presented with the screen shot shown in Fig. 3 (b).



(a) Login page menu. (b) Task for user.

Fig. 3. Screen shots of Login page menu and tasks for the recorder on DiReCT.

In this scenario, the current event is in the "Landslide Observation" category. Accordingly, contextual and background information related to landslide in the affected area by the current disaster is downloaded to DiReCT from the authoring system as a part of pre-collection quality assurance. The recorder can peruse background information needed without having to access the authoring system illustrated by Fig. 4 (a). Fig. 4 (b) shows a questionnaire-based screenshot. It is a popular format used to lead and remind the recorder to fill in all the data items required to describe the condition of the area affected by the disaster. In this example, the observation area is affected by a landslide. The specific data items to be recorded include the estimated number of buried houses, damage level, estimated and observed numbers of casualties, *etc.*



(a) Background info. of landslide. (b) Record page for landslide.
Fig. 4. Screen shot of tasks for recorder on DiReCT.

3.2 Supporting Data/Information and Tools

Again, each DiReCT provides its user with background and scientific data and contextual information to enable him/her to enter observation data and record occurrences of events and phenomena (*e.g.*, rain stopped, debris flow starts, *etc.*) in location-specific and context-specific ways. Fig. 5 shows examples: Background (*e.g.*, census data), scientific, and GIS (Geographical Information System) information are types of information about the region and the type of emergency event which the recorder may need. Historical disaster records are data on similar and related types of disasters that have occurred in the region and data on their impacts on the region. These records are cached to the device. The selections of the downloaded records are specified by the Standard Operating Procedures (SOP) for the disaster type. Metadata are data about disaster data type and format and parameters for data quality control, as well as data and information needed to provide the user with on-line helps at all times.

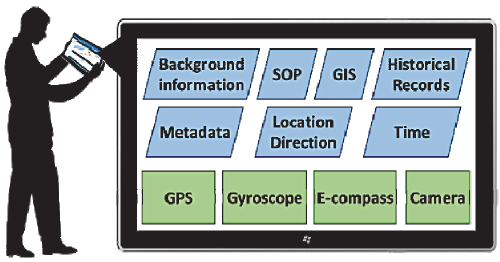


Fig. 5. Local data/information and tools on DiReCT.

Tools such as location/navigation tools, calendar time and timers, and several hardware sensors provide essential data/information for data capture and quality control. Similarly, units of DiReCT used during some types of disasters by some organizations may have special purpose tools such as people counter and remote location pin pointers mentioned in the previous section.

Also stated earlier, data quality control is a distinguishing feature of DiSRC. Following published recommendations such as [44-48], DiSRC divides data quality into quality assurance (QA) and real-time quality control (RTQC) as illustrated by Fig. 6. The former refers to activities that take place before and after data collection. Pre-collection QA involves making sure that correct and sufficiently completed supplement information such as historical record and background information related to the current disaster/emergency and possible affected areas are been downloaded from the authoring server to each DiReCT. Post-collection quality assurance detects the outlier from the records and checks and fixes inconsistencies records. The latter (*i.e.*, RTQC) refers to activities that take place during data collection. QC provides DiReCT with the capability to monitor all collecting process. To be specific, the system provides supports including monitoring the capture of observation data, detecting errors in real-time, alerting the user of the errors and overseeing corrections are made, and so on.

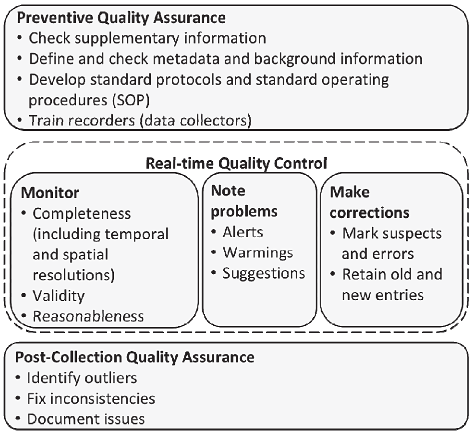


Fig. 6. Data quality assurance (QA) and quality control (QC) for disaster data recording.

3.3 Required Functionality

Based on the previous discussion, we conclude that the capabilities of DiReCT and the authoring system (AS) should include the following:

- DiReCT provides an authentication and authorization mechanism so that a user can login and assume a role among all the roles the user can play;
- DiReCT provides local storage sufficient for storing background information for likely scenarios and for saving observation records in the case of unstable Internet connection;
- DiReCT automatically captures event time and geo-coordinates;
- DiReCT supports real-time data quality control. The AS supports pre-collection and post-collection data quality assurance;
- DiReCT supports advanced capture tools for estimation of object location, area boundaries, people count…*etc.*;
- DiReCT automatic alerts each recorder of violation of time and spatial resolution requirements missing data or conflicting data to be tracked down, resolved, and corrected whenever possible;
- AS integrates data and information captured by all recorders;

- AS provides interface and tools for accessing data and information on past disaster residing in databases maintained physically by multiple organizations;
- AS provides APIs, tools and web services for data and information discovery and retrieval; and
- AS provides translation of data to different data format to input to the tools.

4. DESIGN AND IMPLEMENT OF DIRECT

Fig. 7 depicts the software structure of DiReCT and Table 1 lists the full names of the modules together with the corresponding abbreviated names. As shown in Fig. 7, front-end modules that support the interaction between the user and the device and back-end modules that organize captured data and manage storage and facilities upload and download are described below.

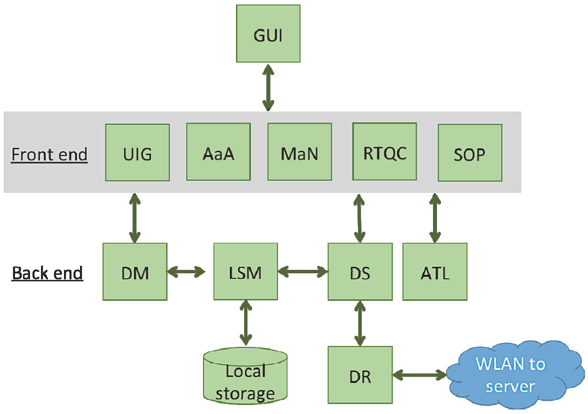


Fig. 7. Software architecture of DiReCT.

Table 1. Modules name lookup table.

Abbreviated name	Full name
GUI	Graphical User Interface
UIG	User Interface Generator
AaA	Authentication & Authorization
MaN	Monitor and Notification
RTQC	Real-time Quality Control
SOP	Standard Operating Procedures
DM	Data Manager
LSM	Local Storage Manager
DS	Data Synchronizer
DR	Data Refresher
ATL	Assistant Tool Loader

4.1 Front-end Support Modules

As stated in the previous section, the users' needs and limitations are the first and foremost factors to consider. During the design and development process of DiReCT, we followed commonly accepted guidelines in user interface and interactive tool design [49-51]. The goal is to produce a system of tools that are easy to use, configure and customize for typical users. In this subsection, the modules that support the GUI are described such as User Interface Generator (UIG), Standard Operating Procedures (SOP), Authentication and Authorization (AaA), Monitor and Notification (MaN) and Real-time Quality Control (RTQC).

(A) UIG and SOP

The configurability of the device is achieved mostly by providing it with a GUI composed of a WPF (Windows Presentation Foundation) plus Gmap.NET component and a state-machine workflow component. The former presents to the user with display and user interface elements, including input controls (*e.g.*, checkboxes, buttons, list boxes and text fields), navigational elements (*e.g.*, sliders, search fields, tags and icons), information elements (*e.g.*, icons, progress bar, notifications, message boxes, and modal windows) and containers. The state-machine workflow defines formally interaction between user and device (*i.e.*, changes in display in response to user's choices of controls). The GUI then operates as specified by the workflow.

A dedicated GUI thread is used to keep the system responsiveness. In addition, the interaction between GUI and other components in DiReCT are all asynchronous so as to avoid blocked user interface. Specifically, the GUI is supported by User Interface Generator (UIG) that dynamically generates and sequences the displays presented to the user according to the GUI configuration file, the SOP component and the state machine workflow. The SOP component provides are one or more standard operating procedures that define and enforce best practices during data capture processes. They are downloaded from the authoring system prior to and during initialization of the device and in response to change notifications from the authoring system as a part of quality assurance.

(B) AaA, MaN and RTQC

Authentication and Authorization (AaA) module authenticates the user during login. The user is authorized to download event data and upload record data to and from DiReCT according to the access control policies enforced by AaA. A user may have multiple roles and may access different types of background information and the records captured by the other users.

The Monitor and Notification (MaN) and Real-time Quality Control (RTQC) modules automatically alert the user of anomalies (*e.g.*, missing data or conflicting data) to be tracked down, resolved, and corrected as much as possible during the data capture process. Specifically, MaN module is a runtime-monitoring tool. It enforces the SOP governing time and spatial resolutions of the capture observational record. In addition, it provides the capability of the real-time monitoring, capture and analysis of events and conditions that indicate the potential for occurrences, or actual occurrences, of errors, and issuing alerts and notifications to trigger error handling or prevention actions. It also provides notification functions that other modules can call to send reminders and alerts.

The RTQC module examines the record metadata and input data during data collection. When it detects defects in data, it may alert the user of the errors and oversee the corrections, or mark the record as defective, or even ignore the defect, and so on in ways defined by the SOP governing how RTQC is required to be done.

4.2 Back-end Support Modules and Record Process

As depicted in Fig. 7, back-end module includes Data Manager (DM), Local Storage Manager (LSM), Data Synchronizer (DS) and Data Refresher (DR). They are described below.

(A) Data Management

Observation records captured by the user are stored, maintained and transferred under the control of Data Manager (DM) module. DM's functions include overseeing the examination by RTQC of each record stored temporarily in the GUI record buffers, putting records found free of defects in one or more collections of records captured by the user, handling defective records in ways specified by SOP, and storing the collections in the local storage. The defective records are stored in a temporary storage space. Whether and when a defective record is to be stored in the local storage and whether and what follow up actions are to be taken is specified by the SOP for data management. Other modules of DiReCT access the event data, user data and record data in the local storage using the API functions of the DM module. DM module works with the Data Synchronizer (DS) to upload record collections stored in the local storage to cloud server during or after data collection and download event data (work descriptions, GUI configuration, SOP configuration, map data, user data, historical records, and quality control data) and user data (profile and settings) before data collection. DS module takes in charge of the information update by matching the data in local storage to the data downloaded supported by Data Refresher (DR). If the data is updated, it would be stored back in the local storage to wait for a call of uploading to the server after DR makes sure there is enough storage and the network is enabled.

Local Storage Manager (LSM) provides APIs for DM module and DS module. The API service includes storing and loading all the record data managed by DM module and DS module and downloading the files required for this recording task to the device's storage space. In order to get better system performance, any storage action (add, update, delete) are enqueue in the queue, waiting for the local storage module to complete the current task, and then dequeue the queue to execute. In terms of storage of record, the data collection would be converted to JSON format. Each recorder has his/her own independent file.

(B) Record Transmission

Fig. 8 depicts in detail the process of record transmission. Storing the observation data on DiReCT captured by the user requires multiple phases. As shown in the figure, when the user clicks the button to save record, the save record procedure starts. The saved data from the user temporarily stored in a GUI record buffer that is managed by I/O helper. These, the records await the DM module to do RTQC validation: RTQC module invokes the validation function from the SOP module to verify the records in the

GUI record buffer. Validated data is moved to the space provided by the DM module. The data that fails to be validated is managed by the failure handler from SOP module. As Fig. 8 shows, the save record work is done in the background by worker thread(s).

Once observation data is validated and ready to be upload to the authoring system, the DR module transmits the data from DiReCT to the AS. When the network is available, the DR module guarantees reliable data transmission using TCP protocol and specified quality of service. The DR module is responsible for the completion of all data transmissions requested during network outage as well as yet-to-be-completed transmissions requested before network outage as soon as network is available again. DiReCT allows user to upload data in any format, and usually, not only text, but also image and video in common data formats.

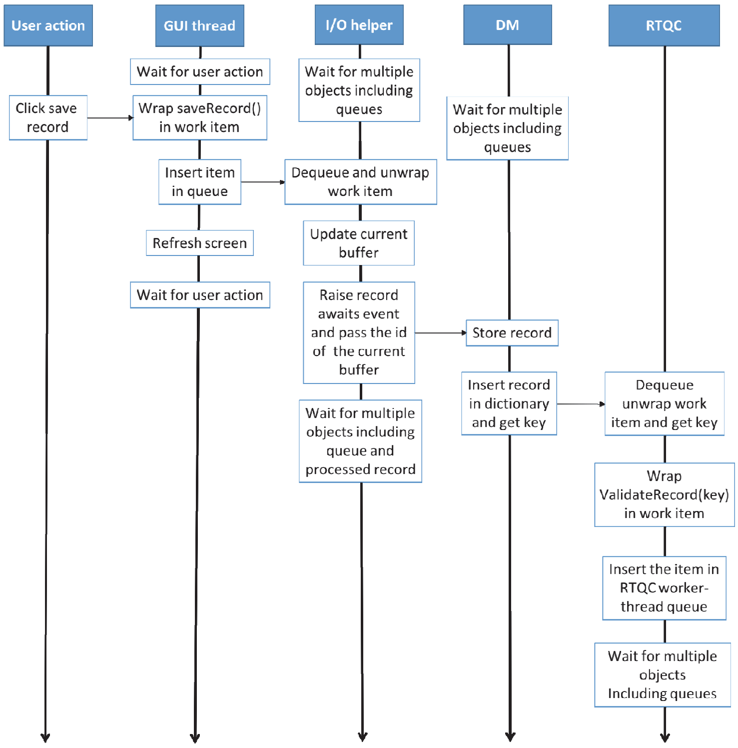


Fig. 8. The process of record transmission.

5. AUTHORING SYSTEM

The authoring system (AS) has four major components. The virtual repository contains data and information on past disasters, geographical information, related scientific information, and real-time data and information on recent emergencies. By providing tools to ease the access and use of the data, the repository hides the fact that datasets presented to users by the repository in fact reside physically in data centers operated by government and non-government institutions in Taiwan and abroad. AS also provides an open data database/repository. In the best-case scenario, AS will be adopted and widely used. It should be one of the sites where disaster history records will be maintained. The virtual repository and open data repository are operational.

The AS also has an ODSD (operation/decision support data) database that holds libraries of SOP guiding observation data capture process, data for quality assurance, and real-time quality control for each of the types of disaster/emergency event supported by DiReCT. This database also maintains the specifications and other parameters needed for configuration and customization of DiReCTs used for capturing data during each type of emergency. The WIP (Work-in-Progress) database holds records uploaded from DiReCTs during and after emergencies. The records on each emergency event are held here until they are processed and used to generate history records and reports. The ODSD and WIP databases and tools provided by them are being designed and built.

A purpose of AS is to help authors generate historical records in human readable and machine readable forms making use of observational data captured by all recorders during and after disaster events. AS also is where disaster-event-specific and user-specific SOP governing data capture practices and RTQC are maintained, and preventive data quality assurance and post-collection data quality control are carried out. It aims to support downloading to DiReCTs SOPs and contextual data and information as a part of preventive data quality assurance and opportunistic upload of captured data from DiReCTs; integrate data uploaded from all DiReCTs and does post-collection data quality control; provides access to an extensible library of search, analysis, simulation, visualization and data mining tools and tools for translating captured data into input formats of the tools; provides a virtual repository of available disaster historical records with APIs and web services for data and information discovery and retrieval by human users, and an open data repository for maintaining future historical records generated by DiSRC system.

5.1 Components of Authoring System

The authoring system (AS) has five major components as shown in Fig. 9. First, it presents to authors and tools a virtual repository of data and information on past disasters, geographical information, related scientific information, and real-time data and information on recent emergencies. Datasets containing these data and information reside physically in data centers operated by government agencies and non-government institutions in Taiwan and abroad, including Central Weather Bureau and GIS Center in Taiwan, NOAA in USA and numerous universities worldwide. Tools such as web crawler, format converters, access control and interoperability services shown in the leftmost box in the figure can significantly reduce the efforts required in accessing widely distributed data and make use of them.

The second major AS component is an open data database/repository. In the best-case scenario, AS will be adopted and widely used. It should be one of the sites where disaster history records will be maintained. This repository is being designed and implemented like an open data site for creation and maintenance disaster historical records in the future.

The third component is a database that holds libraries of SOPs guiding observational data capture process, data for quality assurance, and real-time quality control for each

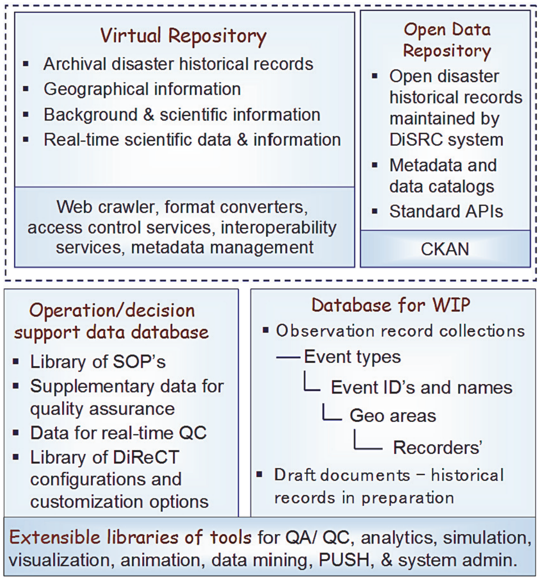


Fig. 9. Major components of authoring system.

of the types of disaster/emergency event supported by DiReCT. The DiReCTs used for capturing data on each type of emergency by each government agency and department are configured and customized according to the requirement specification of the devices. Ideally, requirement specifications of DiReCTs are executable, in terms of workflows, for example. This database maintains the specifications and other parameters needed for configuration and customization of DiReCTs.

The fourth component is the WIP (Work-in-Progress) database that holds observation records uploaded from DiReCTs during and after emergencies. The records on each emergency event are held here until they are processes and used to generate history records and reports. At that time, parts of them will be moved to the open data repository and other archives depending on the choice of the government agency who owns the data. As Fig. 5 shows, within the observation record database, the records are organized hierarchically. For example, the tree of the record collections starts from event type, individual event, geographical region and time interval, and then collections of records contributed by individual recorders. Information on recorders can be stored here optionally. This database also maintains documents generated from the records, including capabilities version control and collaborative authoring.

The fifth component contains extensible library of tools. Fig. 9 lists examples. The system has capabilities for authors to use libraries of tools remotely, assisting them to write reports without the need of installing tools on their devices.

5.2 AS PROTOTYPE

Currently, the prototype authoring system contains only the virtual history record repository and a local open data repository, *i.e.*, the first two AS components mentioned above. Author can search data in these repositories. Search results are managed in Data

Catalog. These data can be further searched by users via APIs, which provides data in JSON format, and web user interface, which visualizes data into table format. For example, the virtual historical record repository now provides access to data from P-Alert, a disaster record website developed by Institute of Earth Science, Academia Sinica and the CWB website. During a searching process launched by a user, the system searches data with specified filter via APIs provided by P-Alert and via HTML webpage analysis based on web crawler tools to obtain data that match the filter condition. In this way, data are presented to the user together as if the data were from the same database.

The local open data repository is built on CKAN [52], an open source data portal software for data management. Data that have been reviewed for data quality control issue can be uploaded, categorized, deleted, and more, by administrators of each event. When a user launches a search process via APIs or web user interface, the system will search through the local database by calling CKAN API, and then return the search results.

6. SUMMARIZATION AND FUTURE WORK

In the previous sections, we described the proposed a DiSRC system. A DiSRC system contains numerous DiReCTs and an authoring server system. The former enables the capture and recording observational data during disasters, while the latter provides functions to help author analyze and use the capture data to generate their reports. The proof-of-concept system demonstrates that the DiReCT application can executes based on pre-defined SOP. The current SOP includes a flood scenario state-machine workflow, simple data validation functions, and a questionnaire. The prototype of authoring system demonstrates the feasibility of the discovery and gathering data from multiple websites and databases, hiding the physical distribution of the data from the users. Our next step is to develop a more polish and complete DiReCT system and its assistant tool library. We will complete the authoring server system, in particular completing a basic part of the tool library containing methods for validating and merging inconsistent data to ease data managing and analyzing process, and tools for analysis, visualization, and simulation.

ACKNOWLEDGEMENT

The authors wish to thank Prof. Jane W. S. Liu of Academia Sinica for the guidance and supervision of development of the DiSRC system. Implementation of DiReCT and Authoring System were done with the help of many graduate students, undergraduate workers and summer interns. The authors thank them sincerely.

REFERENCES

1. Central Weather Bureau, https://translate.google.se/#en/zh-CN/Central%20Weather%20Bureau.
2. NOAA Climate Data Center, https://translate.google.se/#en/zh-CN/NOAA%20 Climate%20Data%20Center.

3. M. Pittore, M. Wieland, and K. Fleming, "Perspectives on global dynamic exposure modelling for geo-risk assessment," *Nature Hazards*, Vol. 86, 2017, pp. 7-30.
4. UNDP, "A comparative review of country-level and regional disaster loss and damage databases," http://www.undp.org/content/undp/en/home/librarypage/crisis-prevention-and-recovery/loss-and-damage-database.html.
5. CommCare, https://www.commcarehq.org/home/.
6. Magpi, http://home.magpi.com/.
7. EPICollect, http://www.epicollect.net/.
8. Fieldata, http://fieldata.org/.
9. Microsoft Data Gathering, https://www.microsoftdatagathering.net/ home.
10. Open Data Kit, https://opendatakit.org/.
11. OpenXdata Kit, http://www.openxdata.org/.
12. PoiMapper, http://www.poimapper.com/.
13. Surveybe, http://surveybe.com/.
14. Formhub, https://formhub.org/.
15. DARMsys, http://qldreconstruction.org.au/about/darmsys.
16. Rover http://www.roverready.org/.
17. Fulcrum, http://www.fulcrumapp.com/.
18. Orion, http://www.orionprotected.com/.
19. KoBoToolbox, http://www.kobotoolbox.org/.
20. F. Ballio, D. Molinari, G. Minucci, M. Mazuran, C. Arias Munoz, S. Menoni, F. Atun, D. Ardagna, N. Berni, and C. Pandolfo, "The RISPOSTA procedure for the collection, storage and analysis of high quality, consistent and reliable damage data in the aftermath of floods," *Journal of Flood Risk Manage*, Vol. 11, 2018, pp. S604-S615.
21. J. Bevington, R. Eguchi, C. Huyck, H. Crowley, F. Dell'Acqua, G. Iannelli, C. Jordan, J. Morley, M. Wieland, S. Parolai, M. Pittore, K. Porter, K. Saito, P. Sarabandi, A. Wright, and M. Wyss, "Exposure data development for the global earthquake model: Inventory data capture tools," in *Proceedings of the 15th World Conference on Earthquake Engineering*, No. 5057, 2012.
22. D. Link, A. Widera, B. Hellingrath, T. de Groeve, G. Eidimtaite, and M. K. Limbu, "GDACSmobile − An IT tool supporting assessments for humanitarian logistics," in *Proceedings of International Annual Conference on Emergency Management Society*, 2015.
23. C. Hodapp, M. Robbins, J. Gray, and A. Graettinger, "Damage tracker − A cloud and mobile system for collecting damage information after natural disasters," in *Proceedings of the 51st ACM Southeast Conference*, Article No. 28, 2013.
24. C. Macdonell, "Ushahidi – A crisis mapping system," *SIGCAS Computers & Society*, Vol. 45, 2015, p. 38.
25. K. I. L. Guillén, U. F. Mendoza, and L. W. Santos, "Crowdmap and Ushahidi: to obtain and visualize traffic congestion information in Mexico city," in *Proceedings of the 4th ACM SIGSPATIAL International Workshop on Computational Transportation Science*, 2011, pp. 24-27.
26. Electronic Data Capture Wiki, https://en.wikipedia.org/wiki/Electronic_data_capture.
27. AKKROO, https://akkroo.com/.
28. Socialbungy, http://www.socialbungy.com/.

29. Formic Fusion for Tablet PC, http://www.formic.com/surveysoftware/tablet-survey/.
30. iCapture, http://www.icapture.com/.
31. Repsly, https://www.repsly.com/.
32. OpenClinica, https://www.openclinica.com/.
33. Clinical Studio, http://www.clinicalstudio.com/.
34. Articulate360, http://www.articulate.com/360.
35. Smartbuilder, http://www.smartbuilder.com/.
36. Pressbook, http://pressbooks.com/.
37. Help and Manual, http://www.helpandmanual.com/index.html.
38. Storybook, http://storybook.software.informer.com/.
39. yWriter, http://ywriter.en.softonic.com/.
40. Wikipedia Anomaly Detection, https://en.wikipedia.org/wiki/Anomaly_detection.
41. Y. Wand and R. Y. Wang, "Anchoring data quality dimensions in ontological foundations," *Communications of the ACM*, Vol. 39, 1996, pp. 86-95.
42. W. Y. Lin, "Data quality," MS Thesis, Department of Computer Science, National Tsing-Hua University, 2016.
43. Did you feel it, https://earthquake.usgs.gov/data/dyfi/.
44. N. C. Tang, Y.-Y. Lin, M.-F. Weng, and H.-Y. M. Liao, "Cross-camera knowledge transfer for multiview people counting," *IEEE Transactions on Image Processing*, Vol. 24, 2015, pp. 80-93.
45. A. B. Chan, *et al.*, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *Proceedings of IEEE 11th Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1-8.
46. National Weather Service, "Observational quality control – General, national weather service instruction 10-1305," http://www.nws.noaa.gov/directives/sym/pd01013005 curr.pdf, 2014.
47. Integrated Ocean Observing System, "Manual for real-time quality control of water level data," http://www.ioos.noaa.gov/qartod/welcome.html, 2014.
48. U.S. Geological Survey, "Data management: Manage quality," http://www.usgs.gov/datamanagement/qaqc.php.
49. K. Knight, "UI design guidelines for responsive design," Codrops, http://tympanus.net/codrops/2013/01/21/ui-design-guidelines-for-responsive-design/, 2013.
50. J. Tidwell, *Designing Interfaces*, 2nd ed., O'Reilly, December 2010.
51. Y. J. Hu, "Design rationales for disaster record capture tool," MS Thesis, Department of Computer Science, National Tsing-Hua University, 2015.
52. CKAN, https://ckan.org/.

Han Wang (王涵) received the double M.S. degrees in Information Technology from Uppsala University, Sweden, and Computer Science and Information Engineering from National Taiwan Normal University, Taiwan in 2017. She is currently working as a Research Assistant in Institute of Information Science, Academia Sinica. Her research interests include disaster management, Internet of Things and machine learning.

Cheng-Yen Tang (唐振嚴) received the B.S degree in 2018 in Computer Science and Information Engineering from National Yunlin University of Science and Technology, Yunlin, Taiwan. He has three year experience in Network Management. His research interests include disaster systems, machine learning, and blockchain.

Chin-Jung Tao (陶璟榕) is currently a master degree student of Computer Science at National Tsinghua University in Taiwan. His research interests include authoring system, disaster management and embedded system technology.

Heng-Chih Hsieh (謝恆至) received the master degree in 2016 from Computer Science and Information Engineering in National Yunlin University of Science and Technology, Yunlin, Taiwan. He has one year experience in academia on disaster prevention system. His research interests include disaster systems, image processing, and embedded system technology.