EARTHQUAKE MANAGEMENT

Under the guidance of

Prof. Chaiyaporn Mutsalklisana

TA Yunwei Jiang

Team Ventures

Sameer Goel
Hari Panjwani
Tarun Keswani
Sankalp Jadon
Cuican Wang

Contents

Problem Statement

During a disaster, there is no electricity, no wi-fi and no telephone working for over a week, countless number of lives are at stake. The problem lies in ineffective communication between the government authorities, volunteers, NGOs and public etc. Because a strong earthquake or hurricane can severely damage the communication infrastructure. Once the regular communication lines are down, it's very difficult to get an overview of the disaster and get the accurate statistics pertaining to the loss at the site, and requirements like relief resources. This is primarily due to ineffective and fragile communication channel between concerned authorities and volunteers. The data gathered is inaccurate due to data duplication from various different sources, which in our case are the volunteers.

This incorrect figures of data are sent to the government organization for relief planning and action. As data received is not actual the relief planning and actions such as resource allocation and mobilization go wrong, thus the supplies sent to the earthquake site is either less or in excess of the real demand.

There is also lack of coordination among Ngo's volunteers due to wrongly made analysis thus most of the volunteer's time get consumed in managing the excess resources and real relief action gets fail to execute efficiently.

Introduction

During earthquake we needs better communication infrastructure to respond to disaster, taking the example of Nepal earthquake when the 7.8 Richter scale earthquake hit Kathmandu, the only thing everybody wanted to do was communicate.

Everyone who survived the first jolt first wanted to make sure their family members and friends were safe. Others wanted to know the scale of damage to their property, loss of lives and the situation in their community (the toll exceeded 8,000, injured over 17,000, house damaged over 400,000). The government apparatus had to be mobilized and that required communication.

State-owned Radio Nepal was the only mass media functioning after the earthquake as it had a sturdy building. After few hours, a few stations that had their buildings intact continued to broadcast. Others shut down and many others collapsed during the quake. Television was not accessible to many, as it required power supply and people had to stay inside to watch. Aftershocks kept the people out in the open for almost a week and during that time radio was the main medium.

Radio Nepal, followed by few some other radio stations, soon started broadcasting news which not only included the death toll, injuries and damages but also crucial announcements, notices from the Government's disaster response authority. Radio became the major means of mass communication, although it was mostly one way. The radio stations faced the challenge of continuing their broadcasts amid fear of aftershocks in order to dispel rumours and help rescue and relief operations.

Communication for rescue work:

In the first week after the quake, the priority was to rescue people who were trapped and alive. However, as there was no emergency means of communication, the rescue operations faced challenges. Many lives would have been saved had there been any emergency means of contract for the people to call the rescue team and a system of prompt response.

People in remote areas were completely cut off. It was impossible to travel, which had come to a standstill because of landslides and fear of more to come. Even if some managed to communicate to the rescue team, it was too big a task for a small team to handle. It would take months if other international rescue teams had not arrived.

Communication for relief:

Local people and organizations, including the international organizations, started distributing immediate relief, mostly food and tents to the people. However, relief mainly went to people who were close by and were accessible by road. Those in remote areas and in places where there was no road or it was damaged could not receive anything.

While it was realized by the Nepal Risk Reduction Consortium (NRRC) that "communication on disaster risk reduction should be harmonized to avoid mixed messages to the public, avoid confusion, and maximize impact", it was far from being implemented due to a lack of any such agreed upon communication strategy or awareness of same.

Secondly, due to lack of communication, those who wanted to distribute relief could not receive information on what kind of relief the people needed. That resulted in distribution of excess or unnecessary materials in some areas and nothing in other areas in the first week: causing angry protests, looting and violence.

Proper communications and dispelling rumors:

"Media shape our perception of risk and that the risks that kills us are not necessary the risks that scare us the most," Dr. Orlando Mercado, an international expert on communicating disaster risks, once said while in Kathmandu in 2013. Indeed, dispelling rumors and making the mass media communicate the right message was another challenging task after the earthquake.

There was false news about "scientists forecasting 9 or 11 Richter scale quake" and "rotten human corpses under the debris", of food and water shortages in the Kathmandu valley, and of cholera that terrified already scared people of spreading infectious disease and more serious disaster. Similar sensational news helped worsen the situation and forced over 1 million terrified people out of Kathmandu in very few days.

Another outstanding phenomenon after the earthquake was the twitter campaign called "GoHomeIndianMedia" with thousands of supporters, that began after the public found coverage by the Indian media too "intense" and "insensitive". The Indian media fraternity also realized that the Indian media too needed to mature and should learn how to better report during disaster.

Communication to handle public fury during disaster:

The scale of people's frustration was huge after the disaster as the public did not feel the presence of their government. There was no one to talk to from government in villages or in cities. There was no place to lodge complaints. In some places it has been many years since local elections have happened and this means there are no people's representatives in the local bodies, like villages and wards.

It was very challenging for the government to reach out to the people in a systematic way. A simple emergency inquiry system would have made things easier for people. The only public line some people were aware of was 100 – the police. Not many were aware of emergency contacts. In every corner, you would find peoplecriticizing the government and the political party representatives for failing to do anything to help them during the disaster.

The government, however, took some immediate damage control measures to quell public anger. Prime Minster Sushil Koirala decided to address the nation three days after the first quake. With the announcement of relief packages, the address would certainly have reassured earthquake survivors.

Communication and resource mobilization:

Resource mobilization was probably the sector most affected by lack of proper and systematic communication. As foreign donations started pouring in after the quake, local NGOs rushed to the villages with relief. Many organizations and groups started their own fundraising campaigns nationally and internationally. An unaccounted amount of money was wired through money transfers.

The Prime Minster's office soon jumped in, set up a PM Relief Fund and declared that it would pool all the money collected at all the accounts set up by fundraisers. After news that the NGOs relief work was not transparent and not monitored, the government made another announcement that NGOs could distribute relief only through government channels. This triggered huge uproar and confusion and the government adjusted its decision after incurring criticism.[1]

Between these two decisions, there was a lot of confusion and lack of systematic communication. It was all ad-hoc -- an official of the National Planning Commission (which was coordinating resource mobilization) twitted and posted in Facebook that, "I have been swamped with enquiries on what the recent Central Bank directive transfer of funds after the April 25 earthquake means." He took pains to explain the confusion with a long explanatory note.

After the NGOs criticism, the government rolled back its restriction on direct distribution relief. This is clearly a challenge for Nepal to find out what kind of communication system caused these decisions and confusions.

The Internet and Facebook:

Perhaps there is no debate that Facebook and other social media helped people connect and communicate on an unprecedented scale. If not immediately after the earthquake, during the very first week itself, Facebook had become the only major hub for communication among local people. It allowed people to share their thoughts and feelings and other information. Words of prayer, love and support poured in from all across the social circle, including from friends and families living outside the country.

The warm words helped people heal their trauma and keep them controlled in the face of the tragedy. Facebook Inc. and Mark Zuckerberg himself took the initiative to connect people and started the "Mark Safe" project for all Facebook account holders in Nepal, which was an innovation and helped millions of people to communicate their needs.

Local radio:

Radio stations turned out to be very helpful in making communication possible during the disaster. This experience made everyone realize the importance of taking extra care in setting up public buildings, including communication hubs like radio stations. It also prompts us to think about disaster preparedness plans for mass media.

The humanitarian aid coordination:

International humanitarian organizations played an exemplary role in addressing some of the major communications challenges during Nepal's earthquake. The UN OCHA took charge of coordinating all international organizations and their responses with local authorities and the people by setting up different coordination hubs at different levels. Regular situation updates were helpful in coordinating relief work.

PROPOSED SOLUTION

To solve this problem, we have a conceptual model of a device, which is an enhanced version of ham radio operating on radio frequency, sending text messages. The device works on broadcast mode and the minimum effective radius of the device is 30km. Having a large range possible on the device, the volunteers can be closely connected. The device works on Wireless Mesh Network (WMN), enabling the users to be connected and share data. A heuristic algorithm is used to reduce traffic congestion. A concurrent addition algorithm designed by us is the main principle of the model. Using this algorithm, every device used by the volunteer will show an updated data number each second on the device screen in coherence with the data entered by the user. We think that by having an accurate data, on the channel, and the knowledge of updated data about the disaster, can help everyone involved to contribute in rescue and relief operations. Users can communicate with each other and the government agencies, giving early planning options for effective response measures. We believe that it has potential to support effective management and expedite rescue work and save lives in such unfortunate occurrences around the world.

PROPOSED DEVICE DESCRIPTION

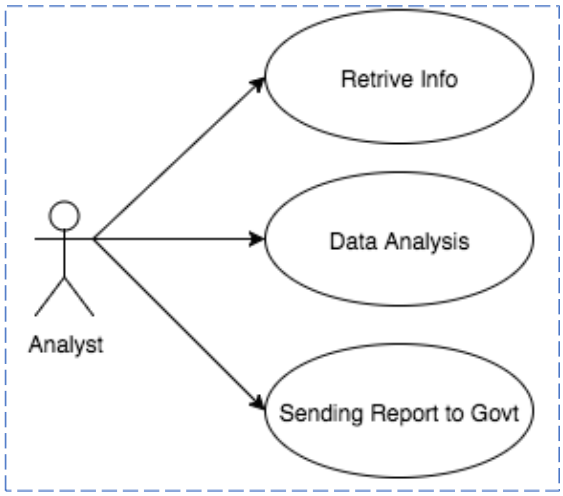
Proposed model is based upon ham radio to pass messages through radio frequency. The existing ham radio setup allows for two-way communication between two devices. Our device will be redesigned to be used by the volunteers present at the disaster location, on a multi-emission mode which allows multiple to multiple message sharing. The user interface will involve a set of options like:

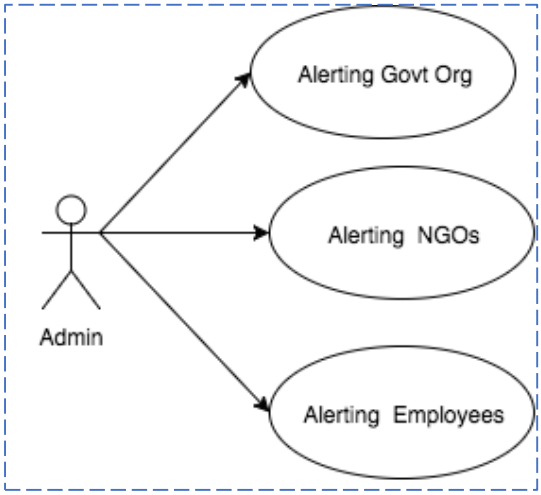
- Number of Casualties?
- Number of People Injured?
- Number of houses destroyed?

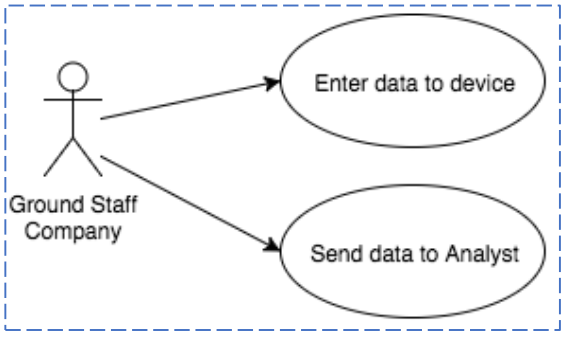
The device will use the range of ham frequency using Mash Network Routing Algorithm. On approval of government, the use of other distress frequencies can also be permitted. Such channels are known as distress, safety and calling frequencies, typically ranging from 30 MHz to 500 MHz

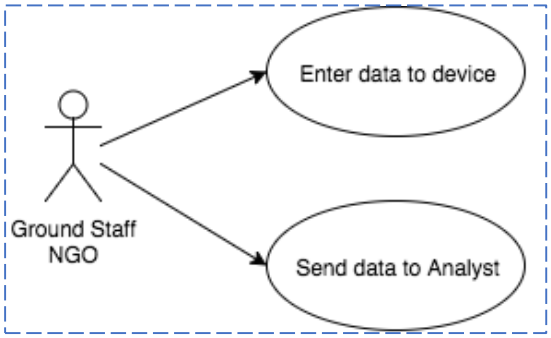
A clear operational instruction for response in emergency is a required attribute of any device. In a study by the Rehabilitation Engineering Research Center for Wireless Technologies, it was revealed that regardless of the initial form of notification, a secondary form was necessary before action would be taken. This supports the important observation that providing clear and concise instructions may reduce dependency on such secondary verification; and thus, providing instructions may save lives in an urgent emergency situation. Keeping this in mind, a simple user interface is required on the device.

Use case Diagram

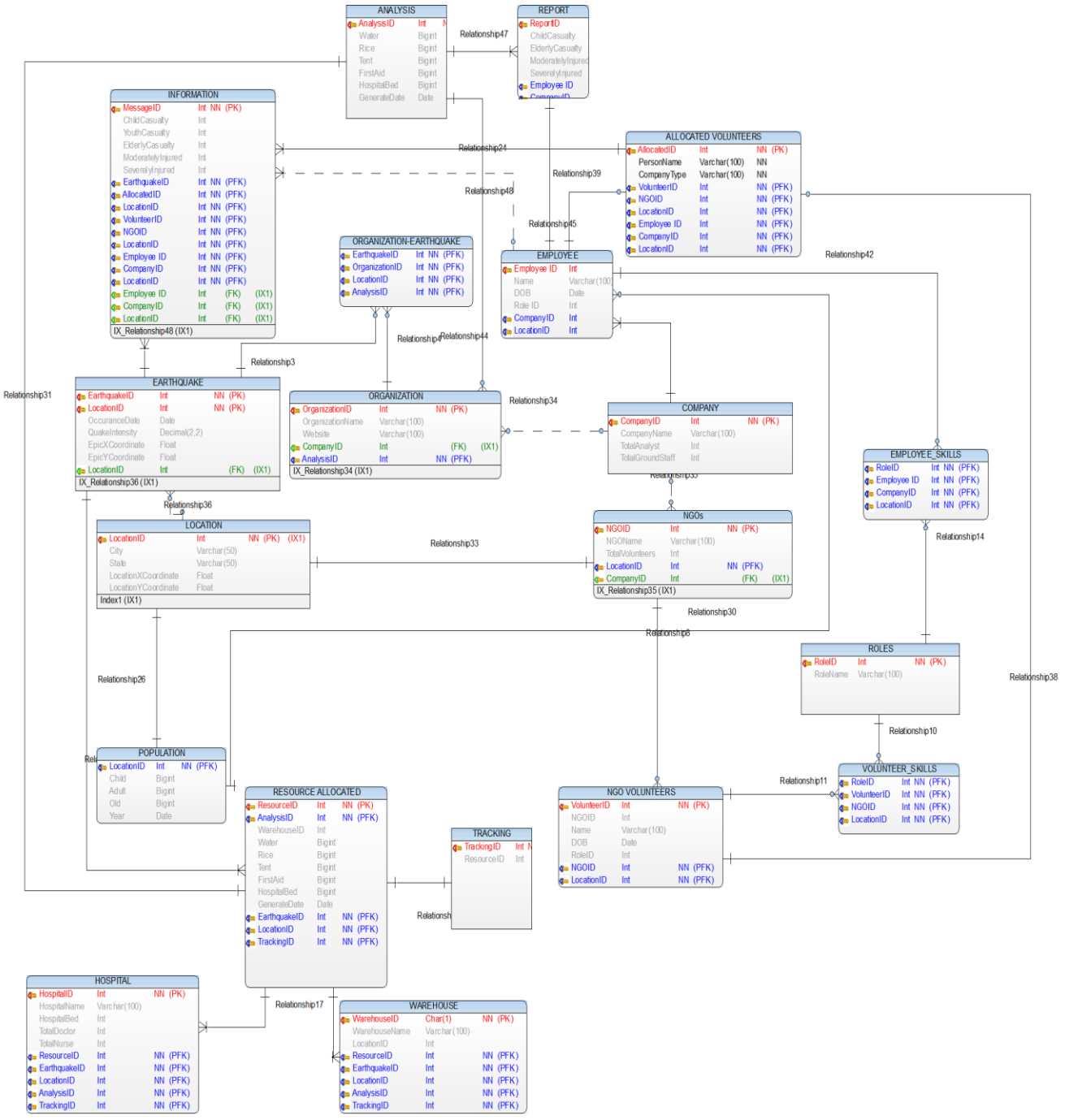








EER Diagram


ANALYSIS
AnalysisID Int
Water Bigint
Rice Bigint
Tent Bigint
FirstAid Bigint
HospitalBed Bigint
GenerateDate Date
Relationship47
REPORT
ReportID
ChildCasualty
ElderlyCasualty
ModeratelyInjured
SeverelyInjured
Employee ID
INFORMATION
MessageID Int NN (PK)
ChildCasualty Int
YouthCasualty Int
ElderlyCasualty Int
ModeratelyInjured Int
SeverelyInjured Int
EarthquakeID Int NN (PFK)
AllocatedID Int NN (PFK)
LocationID Int NN (PFK)
VolunteerID Int NN (PFK)
NGOID Int NN (PFK)
LocationID Int NN (PFK)
Employee ID Int NN (PFK)
CompanyID Int NN (PFK)
LocationID Int NN (PFK)
Employee ID Int (FK) (IX1)
CompanyID Int (FK) (IX1)
LocationID Int (FK) (IX1)
IX_Relationship48 (IX1)
Relationship24
Relationship39
Relationship48
Relationship45
ALLOCATED VOLUNTEERS
AllocatedID Int NN (PK)
PersonName Varchar(100) NN
CompanyType Varchar(100) NN
VolunteerID Int NN (PFK)
NGOID Int NN (PFK)
LocationID Int NN (PFK)
Employee ID Int NN (PFK)
CompanyID Int NN (PFK)
LocationID Int NN (PFK)
Relationship42
ORGANIZATION-EARTHQUAKE
EarthquakeID Int NN (PFK)
OrganizationID Int NN (PFK)
LocationID Int NN (PFK)
AnalysisID Int NN (PFK)
EMPLOYEE
Employee ID Int
Name Varchar(100)
DOB Date
Role ID Int
CompanyID Int
LocationID Int
Relationship4
Relationship44
Relationship3
Relationship31
EARTHQUAKE
EarthquakeID Int NN (PK)
LocationID Int NN (PK)
OccurranceDate Date
QuakeIntensity Decimal(2,2)
EpicXCoordinate Float
EpicYCoordinate Float
LocationID Int (FK) (IX1)
IX_Relationship36 (IX1)
ORGANIZATION
OrganizationID Int NN (PK)
OrganizationName Varchar(100)
Website Varchar(100)
CompanyID Int (FK) (IX1)
AnalysisID Int NN (PFK)
IX_Relationship34 (IX1)
Relationship34
COMPANY
CompanyID Int NN (PK)
CompanyName Varchar(100)
TotalAnalyst Int
TotalGroundStaff Int
EMPLOYEE_SKILLS
RoleID Int NN (PFK)
Employee ID Int NN (PFK)
CompanyID Int NN (PFK)
LocationID Int NN (PFK)
Relationship14
Relationship36
LOCATION
LocationID Int NN (PK) (IX1)
City Varchar(50)
State Varchar(50)
LocationXCoordinate Float
LocationYCoordinate Float
Index1 (IX1)
Relationship33
NGOs
NGOID Int NN (PK)
NGOName Varchar(100)
TotalVolunteers Int
LocationID Int NN (PFK)
CompanyID Int (FK) (IX1)
IX_Relationship35 (IX1)
Relationship30
Relationship8
Relationship26
ROLES
RoleID Int NN (PK)
RoleName Varchar(100)
Relationship38
Relationship10
POPULATION
LocationID Int NN (PFK)
Child Bigint
Adult Bigint
Old Bigint
Year Date
VOLUNTEER_SKILLS
RoleID Int NN (PFK)
VolunteerID Int NN (PFK)
NGOID Int NN (PFK)
LocationID Int NN (PFK)
Relationship11
RESOURCE ALLOCATED
ResourceID Int NN (PK)
AnalysisID Int NN (PFK)
WarehouseID Int
Water Bigint
Rice Bigint
Tent Bigint
FirstAid Bigint
HospitalBed Bigint
GenerateDate Date
EarthquakeID Int NN (PFK)
LocationID Int NN (PFK)
TrackingID Int NN (PFK)
TRACKING
TrackingID Int N
ResourceID Int
NGO VOLUNTEERS
VolunteerID Int NN (PK)
NGOID Int
Name Varchar(100)
DOB Date
RoleID Int
NGOID Int NN (PFK)
LocationID Int NN (PFK)
HOSPITAL
HospitalID Int NN (PK)
HospitalName Varchar(100)
HospitalBed Int
TotalDoctor Int
TotalNurse Int
ResourceID Int NN (PFK)
EarthquakeID Int NN (PFK)
LocationID Int NN (PFK)
AnalysisID Int NN (PFK)
TrackingID Int NN (PFK)
Relationship17
WAREHOUSE
WarehouseID Char(1) NN (PK)
WarehouseName Varchar(100)
LocationID Int
ResourceID Int NN (PFK)
EarthquakeID Int NN (PFK)
LocationID Int NN (PFK)
AnalysisID Int NN (PFK)
TrackingID Int NN (PFK)

NORMALIZATION

The main purpose of normalization is to remove data redundancy and anomalies created while performing INSERT, UPDATE and DELETE operations.

Therefore, we have normalized all the tables to 3NF.

Example – In Population Table which have population data of various cities, we have normalized the attribute Location by making another table called “Location” and referencing it to Population Table. This reduces data redundancy.

POPULATION:

LocationID(PK)	City	State	LocationXCoordinate	LocationYCoordinate

LOCATION:

LocationID(PK)	City	State	LocationXCoordinate	LocationYCoordinate

Similarly, all other tables have been normalized and data redundancy has been reduced.

VIEWS

1. **Volunteer List** – This view contains the list of people who will be going to the affected area to provide the real time data. We have provided each person with the ID to identify them as they send the messages. Also the view contains whether they belong to NGO or they are our company employees.

```
create view VolunteerList as
(
    select row_number() over ( order by PERSON_NAME) as AllocatedID, PERSON_NAME, Type
    from  (
        select Name as PERSON_NAME, 'NGO' as Type
        from NGOVOLUNTEER n
        where NGOId IN (select NGOID from NGO where LocationID = 10)
        UNION
        select  Name, 'COMPANY' as Type
        from employee
        where companyID IN (select companyId from company where locationID = 10)
    ) a
)

select * from VolunteerList
```

%

Results | Messages

AllocatedID	PERSON_NAME	Type
1	Aimee	COMPANY
2	Alisa	NGO
3	Audrey	COMPANY
4	Brenna	NGO
5	Cameron	NGO
6	Idona	COMPANY
7	Jaden	NGO
8	Joan	COMPANY
9	Thomas	NGO

2. **Assigned Message** – This view provides the result of which message was assigned to which employee. Messages are assigned to individual employee which they will be analyzing and providing the report of those analyzed data.

```
create procedure RetriveMessage
@allocatedID varchar,
@empID varchar(200)
as
DECLARE @SQL nvarchar(max)
begin

  SET @SQL = 'create view AssignedMessage AS
           select '+ @empID +' as EmployeeID,MessageID, LocationID, EarthQuakeID, ChildCasualty, YouthCasualty,
           ElderlyCasualty, ModeratelyInjured, SeverelyInjured from information where AllocatedID = ' + @allocatedID

   EXEC (@SQL)

   INSERT INTO Report (EmployeeID, LocationID, EarthquakeID, ChildCasualty, YouthCasualty, ElderlyCasualty, ModeratelyInjured, SeverelyInjured)
    (
        select EmployeeID, LocationID, EarthquakeID, avg(ChildCasualty), avg(YouthCasualty), avg(ElderlyCasualty), avg(ModeratelyInjured), avg(SeverelyInjured)
        from AssignedMessage m
        where m.LocationID = 1
        group by m.EmployeeID, m.LocationID, m.EarthQuakeID
    )
end
```

Results | Messages

EmployeeID	MessageID	LocationID	EarthQuakeID	ChildCasualty	YouthCasualty	ElderlyCasualty	ModeratelyInjured	SeverelyInjured
40	5	1	80	1167	2432	1760	13637	7325
40	6	1	80	1376	2412	1677	11164	7853
40	7	1	80	1746	2810	1679	11107	7637
40	8	1	80	1839	2753	1716	11973	6712

ReportID	EmployeeID	LocationID	EarthquakeID	ChildCasualty	YouthCasualty	ElderlyCasualty	ModeratelyInjured	SeverelyInjured
4	40	1	80	1532	2601	1708	11970	7381

STORED PROCEDURE

Procedures improve the performance of the database significantly. It's basically set of queries which are precompiled which needs to be executed frequently.

1. **Minimum Distance** – This procedure calculates the distance of the cities from the affected area and provides the location which is nearer to that earthquake affected area. This helps in giving us the information regarding place which can be used to allocate the resources like hospital, warehouse etc. Also CURSOR was used to calculate the distance for every location in a loop. Using the cursor, procedure was called for every location sending the latitude and longitude as an input to procedure.

```
--Cursor Creation
DECLARE @loc int
DECLARE @lat float
DECLARE @long float
DECLARE @epix float
DECLARE @epiy float
DECLARE @locationID int
DECLARE @Result int

DECLARE curs CURSOR LOCAL FAST_FORWARD FOR
   select l.LocationID, LocationXCoordinate, LocationYCoordinate, EpicXcoordinate, EpicYcoordinate
   from earthquake e, Location l
   where e.OccurenceDate =(CAST(GETDATE() as DATE))
   and l.LocationID NOT IN
(select LocationID from Earthquake where OccurenceDate =cast(GETDATE() as date))

OPEN curs

FETCH NEXT FROM curs INTO @loc, @lat, @long, @epix, @epiy

WHILE @@FETCH_STATUS = 0 BEGIN
```

esults | Messages

locationID	distance
37	9500

1. **Person Message** – This procedure provides the information regarding the message sent by individual employee. It lists all the messages send by an individual with all the data associated with that particular information.

```
--Procedure to get the message from a specific person
create procedure person_message
@personId int
AS
DECLARE @message int
BEGIN
select * from INFORMATION where AllocatedID = @personId
END

--DECLARE @Message
exec person_message @personID = 2

--Cursor Creation
DECLARE @personID int
DECLARE @Result int
DECLARE curs CURSOR LOCAL FAST_FORWARD FOR
```

Results | Messages

MessageID	AllocatedID	ChildCasualty	YouthCasualty	ElderlyCasualty	ModeratelyInjured	Severelyinjured	LocationID	EarthquakeID
1	1	1940	2740	1672	14585	6612	1	80
2	1	1732	2936	2314	10406	7801	1	80
3	1	1443	2766	2226	10904	6536	1	80
4	1	1163	2399	1605	10789	9306	1	80

MessageID	AllocatedID	ChildCasualty	YouthCasualty	ElderlyCasualty	ModeratelyInjured	Severelyinjured	LocationID	EarthquakeID
5	2	1167	2432	1760	13637	7325	1	80
6	2	1376	2412	1677	11164	7853	1	80
7	2	1746	2810	1679	11107	7637	1	80
8	2	1839	2753	1716	11973	6712	1	80

2. **Retrieve Message** – This view provides information of messages assigned to the Analyst of the company. The Analysts analyzes the data of the messages assigned to them and submit the report which is later consolidated into one which provides final information regarding the resource allocated.

```
create procedure RetriveMessage
@allocatedID varchar,
@empID varchar(200)
as
DECLARE @SQL nvarchar(max)
begin

  SET @SQL = 'create view AssignedMessage AS
           select '+ @empID +' as EmployeeID,MessageID, LocationID, EarthQuakeID, ChildCasualty, YouthCasualty,
           ElderlyCasualty, ModeratelyInjured, SeverelyInjured from information where AllocatedID = ' + @allocatedID

   EXEC (@SQL)

   INSERT INTO Report (EmployeeID, LocationID, EarthquakeID, ChildCasualty, YouthCasualty, ElderlyCasualty, ModeratelyInjured, SeverelyInjured)
    (
        select EmployeeID, LocationID, EarthquakeID, avg(ChildCasualty), avg(YouthCasualty), avg(ElderlyCasualty), avg(ModeratelyInjured), avg(SeverelyInjured)
        from AssignedMessage m
        where m.LocationID = 1
        group by m.EmployeeID, m.LocationID, m.EarthQuakeID
    )
end
```

Results | Messages

EmployeeID	MessageID	LocationID	EarthQuakeID	ChildCasualty	YouthCasualty	ElderlyCasualty	ModeratelyInjured	SeverelyInjured
40	5	1	80	1167	2432	1760	13637	7325
40	6	1	80	1376	2412	1677	11164	7853
40	7	1	80	1746	2810	1679	11107	7637
40	8	1	80	1839	2753	1716	11973	6712

ReportID	EmployeeID	LocationID	EarthquakeID	ChildCasualty	YouthCasualty	ElderlyCasualty	ModeratelyInjured	SeverelyInjured
4	40	1	80	1532	2601	1708	11970	7381

3. **Allocated Resource** – This stored procedure will provide the information about the resource that should be allocated after the analysis on the data provided by the volunteers on the site.

```
--Analysis Procedure
create procedure AllocateResource
@location int
AS
BEGIN

insert into Analysis (GenerateDate, LocationID) values (cast(getdate() as Date), @location)

Update Analysis set

--LocationID = @location,

Rice = (select (CEILING(((p.Child - r.ChildCasualty) /1000) * 250) + CEILING(((p.Adult - r.YouthCasualty) /1000) *400) + CEILING(((p.Old - r.ElderlyCasualty) / 1000
from Report r
INNER JOIN Population p
ON r.LocationID = p.LocationID
where p.LocationID = @location),

Water = (select (CEILING(((p.Child - r.ChildCasualty) * 1.5)) + CEILING((p.Adult - r.YouthCasualty) * 3) + CEILING((p.Old  - r.ElderlyCasualty) *2.5))
from Report r
INNER JOIN Population p
ON r.LocationID = p.LocationID
where p.LocationID = @location),

Tent = (select (CEILING(((p.Child - r.ChildCasualty) + CEILING(p.Adult - r.YouthCasualty) + CEILING(p.Old - r.ElderlyCasualty)) / 1000))
```

Results | Messages

AnalysisID	LocationID	Water	Rice	Tent	FirstAid	HospitalBed	GenerateDate
6	10	1803604	250600	706	8977	7381	2015-12-08
7	2	1803604	250600	706	8977	7381	2015-12-08
8	1	1803604	250600	706	8977	7381	2015-12-08
9	1	1803604	250600	706	8977	7381	2015-12-08
10	1	1803604	250600	706	8977	7381	2015-12-08
11	1	1803604	250600	706	8977	7381	2015-12-08

TRIGGERS

Triggers are nothing but a specific reaction when we perform a certain action of any of the tables like INSERT, UPDATE and DELETE.

We have three triggers which does the following work –

1. **Affected Area Population** – As soon as an Earthquake occurs, a trigger is initiated which calculates the population of that are. The population has been segregated into CHILD, ADULT and OLD. The population in the population table is updated quarterly and therefore we get the accurate data when an Earthquake occurs.

```
CREATE TRIGGER AFFECTED_AREA_POPULATION ON Earthquake
FOR INSERT
AS
IF EXISTS (SELECT * FROM Earthquake
where OccurenceDate = cast(GETDATE() as date))
BEGIN
select * from population
where locationID IN
(select LocationID from Earthquake where OccurenceDate = cast(GETDATE() as date))
END

-------------------------------------------------------------------------------

insert into earthquake values (3, GETDATE(), 7, 120, 37);
```

%

Results | Messages

LocationID	Child	Adult	Old	Year_date
3	103145	325864	241795	2016-06-03

2. **Resource Management** – As soon as the Analysis table is updated with the final allocation of resources, an entry is made into Resource Allocated table. This table gives the final details of the allocated resources.

```
create Trigger resourceManagement on Analysis
after update
as
if exists (Select * from Analysis where GenerateDate = (CAST(GETDATE() as Date)))
BEGIN

DECLARE @warehouseID int
SET @warehouseID = (select top 1 WarehouseID from Warehouse
where locationID IN
(select e.locationID from distance e
where e.distance = (select min(d.distance) from distance d)));

Insert into ResourceAllocated(AnalysisID, WarehouseID, Water, Rice, Tent, FirstAid, HospitalBed, GenerateDate)
Select AnalysisID, @warehouseID, Water, Rice, Tent, FirstAid, HospitalBed, (CAST(GETDATE() as Date))
from Analysis where GenerateDate = (CAST(GETDATE() as Date))
END

select * from ResourceAllocated
```

%

Results | Messages

ResourceID	AnalysisID	WarehouseID	Water	Rice	Tent	FirstAid	HospitalBed	GenerateDate
3	6	7	1803604	250600	706	8977	7381	2015-12-08
4	7	7	1803604	250600	706	8977	7381	2015-12-08
5	8	7	1803604	250600	706	8977	7381	2015-12-08
6	9	7	1803604	250600	706	8977	7381	2015-12-08
7	10	7	1803604	250600	706	8977	7381	2015-12-08
8	11	7	1803604	250600	706	8977	7381	2015-12-08
9	12	7	1803604	250600	706	8977	7381	2015-12-08

3. **Generate Tracking** – When the resource has been allocated, a trigger gets executed which tracks the usage of the resources that has been allocated. It keeps track how much quantity has been used and how much is still left. Based on this the next analysis result will be generated with second and subsequent round of resource allocation.

```
create Trigger generateTracking on ResourceAllocated
after Insert
as
IF Exists (Select * from ResourceAllocated where GenerateDate = (CAST(GETDATE() as Date)))
BEGIN
Insert into TRACKING(ResourceID) select resourceID from ResourceAllocated where GenerateDate = (CAST(GETDATE() as Date))
END

select * from Tracking
```

%

esults | Messages

TrackingID	ResourceID
1	3
2	4
3	5
4	6
5	7
6	8
7	9

INDEXING

Indexes are special lookup tables that the database engine can be used to speed up data retrieval. Thus to improve the data retrieval speed we have used indexing in our project.

1. Index on LocationID of Location Table to retrieve the location faster during an earthquake so that further queries can be executed quickly. Any delay would result in more casualties during a calamity.

```
create index LocationIndex on Location(LocationID)
```

2. Index on HospitalID of Hospital Table to get the list of all hospitals near to the affected area quickly so that medical help can be activated as soon as possible.

```
create index HospitalIndex on Hospital(HospitalID)
```

```
%
Messages
Command(s) completed successfully.
```

3. Index on NGO_ID to get the list of all nearby NGO's to the affected area.

```
create index NGOIndex on NGO(NGO_ID)
```

```
%
Messages
Command(s) completed successfully.
```

TRANSACTIONS

Transactions is a set of one or more SQL statements that perform a set of related actions. The statements are grouped together and treated as a single unit whose success or failure depends on the successful execution of each statement in the transactions. Thus we have used transaction to insert initial data into the database.

```
BEGIN TRANSACTION
SAVE TRAN ANALYSIS1
insert into Analysis (GenerateDate, LocationID) values (cast(getdate() as Date), @location)
Update Analysis set
Rice = (select (CEILING(((p.Child - r.ChildCasualty) /1000) * 250) + CEILING(((p.Adult - r.YouthCasualty) /1000) *400) + CEILING(((p.Old - r.ElderlyCasualty) / 1000) *
from Report r
INNER JOIN Population p
ON r.LocationID = p.LocationID
where p.LocationID = @location),

Water = (select (CEILING(((p.Child - r.ChildCasualty) * 1.5)) + CEILING((p.Adult - r.YouthCasualty) * 3) + CEILING((p.Old  - r.ElderlyCasualty) *2.5))
from Report r
INNER JOIN Population p
ON r.LocationID = p.LocationID
where p.LocationID = @location),

Tent = (select (CEILING(((p.Child - r.ChildCasualty) + CEILING(p.Adult - r.YouthCasualty) + CEILING(p.Old - r.ElderlyCasualty)) / 1000))
from Report r
INNER JOIN Population p
ON r.LocationID = p.LocationID
where p.LocationID = @location),

HospitalBed = (select (CEILING(r.severelyinjured))
from Report r
INNER JOIN Population p
ON r.LocationID = p.LocationID
where p.LocationID = @location),
FirstAid = (select (CEILING((r.ModeratelyInjured * 3/ 4)))
from Report r
INNER JOIN Population p
ON r.LocationID = p.LocationID
where p.LocationID = @location)
where GenerateDate = (CAST(GETDATE() as date)) and locationID = @location
END

COMMIT
```

USER PRIVILEGES

There are various kind of users who can access different tables and/or attributes.

1. **SYSADMIN** – This user has the admin privileges who have full control on the database. This user will be responsible for inserting the data into Earthquake table which will trigger the whole system and the actions will be taken to supply the resources to the affected area as soon as possible.

```
create LOGIN sysadmin with PASSWORD = 'sysadmin'

create user sysadmin for login sysadmin

GRANT ALL PRIVILEGE TO sysamdin
```

2. **NGO VOLUNTEER** – This is the user for NGO Volunteers who will be sending the real time data which would be analyzed. They will have the privilege to INSERT and UPDATE the messages they have sent.

```
create LOGIN ngoVol with PASSWORD = 'ngoVol'

create user ngoVol for login ngoVol

Grant INSERT, UPDATE TO ngoVol
```

3. **EMPOYEE VOLUNTEER** – This is the company on ground staff which will also be present in the affected area to send the real time data. That data will be used for analysis and reaching to the conclusion of resource allocation.

```
create LOGIN empVol with PASSWORD = 'empVol'

create user empVol for login empVol

Grant INSERT, UPDATE TO empVol
```

4. **ANALYST** – These are the company staff which will stay in office and do the analysis of the data sent by the members on ground.

```
create LOGIN empAnalyst with PASSWORD = 'empAnalyst'
                    .
create user empAnalyst for login empAnalyst


Grant SELECT, INSERT, UPDATE, DELETE TO empAnalyst
```

REFERENCES

1. Zhikui Chen, Liao Chen, Yang Liu, Yong Piao, “Application Research of Wireless Mesh Network on Earthquake”- 2009 International Conference on Industrial and Information Systems.
2. https://en.wikipedia.org/wiki/Emergency_communication_system.
3. http://www.globalmarinenet.com/hf-radio-vs- satellite-phones-what-is-really-free
4. Soumen Kanrar, “Efficient Packet Forwarding in Mesh Network”- ijim.v6i2.1991
5. http://www.w3schools.com/sql/default.asp
6. https://msdn.microsoft.com/en-us/library/ms188929.aspx
7. https://stackoverflow.com/questions/10911249/returning-multiple-values-from-a-stored-procedure