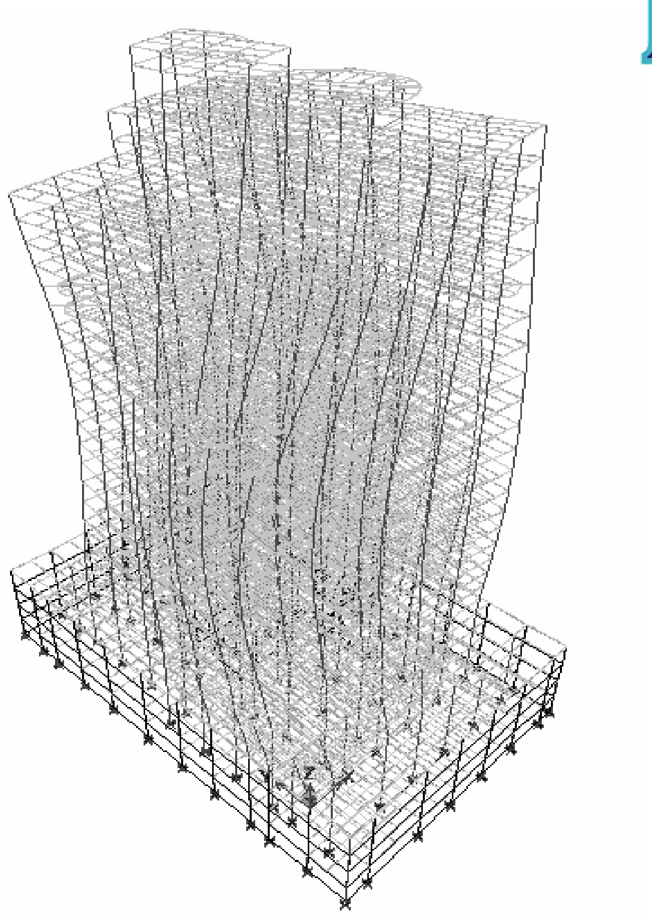
National Taiwan University

ADaptable VIsionary System for Earthquake Resolution



Mentor: Feipei Lai flai@ntu.edu.tw

Pei-Chin Wang b89091@csie.ntu.edu.tw

Keng-Hao Chang b89029@csie.ntu.edu.tw

Yuan-Chung Shen b89040@csie.ntu.edu.tw

Che-Hsuan Shu b89058@csie.ntu.edu.tw

1 Abstract

Serious earthquakes have caused severe and extensive damages all over the world. However, until now there is not any effective way to predict when the earthquake will happen. Here ***ADVISER*****, ADaptable VIsionary System for Earthquake Resolution**, is a system designed to minimize the damage caused by earthquakes.

Rather than trying to predict the earthquakes themselves, ***ADVISER*** predicts where and how the damage can be caused by the earthquakes, and takes corresponding action. Seismologists classify the seismic waves into two types, "P-waves" and "S-waves." P-waves travel two times faster than S-waves, which introduces more damage than the former. If we can utilize the fast traveling characteristic of P-waves to quickly compute the seismic parameters, prediction for the magnitude of the coming earthquake can be made and warnings can be broadcast, before the main damage power sources, S-waves, arrive.

ADVISER is aimed to make the world a safer place by taking advantage of combining the technique of seismology and computer science. We want to provide customized specific information about the incoming earthquake for an individual as soon as possible. To achieve early warning with adaptive messages, we implement the system by a hierarchical structure with five features described as follows.

- Earthquake information analysis
- Fast and large amount of message dissemination
- Wireless location tracking capability
- Adaptive message generator
- Warning application and target system control

In this report, we choose a hospital as the performing stage of our system. Imagine when an earthquake occurs, a hospital may suffer a serious loss. However, ***ADVISER***, when an earthquake occurs, can immediately take action to prevent disastrous damage before the earthquake arrives. ***ADVISER*** will provide the users with useful information, such as an emergency exit map, the standard operation procedure (SOP) for each person, and even hold some important systems not to be crashed, such as elevators, power supply system, hospital equipments. Although this project takes hospital as an example, it is proper to expect our system to work in other occasions, such as office and home. It is not necessary to change the nowadays network structure for building up the system. Besides, it is of hierarchical architecture, such that it can disperse the computational and network load to other computers, and it could add more computers to serve more users in the system if necessary. Therefore, ***ADVISER*** is scalable and low-cost, and it can be easily adapted in other situations and provide useful help without extra cost.

2 System Overview

2.1 System hardware and software

In order to alleviate the impact of the earthquake and make the world safer, ***ADVISER*** notices an earthquake is coming, warns people in advance adaptively, and activates target system controls automatically.

ADVISER focuses on a scenario that a large quantity of people and a wide range of area will be covered. Therefore ***ADVISER*** adopts a **hierarchical structure** such that the load can be evenly distributed and the adaptable vision can be fulfilled. In all, one central **main server** connects with multiple **sub-servers** distributed in the building via wired network. **Clients**, such as **users** and **target systems**, communicate indirectly with the main server through sub-servers: users interact with the local sub-servers using the existing wireless LAN with the prevailing personal devices, such as PDAs, cell phones, and notebooks; target systems such as elevators are controlled by control signals. In addition, there is a database storing adaptive SOP (Standard Operation Procedure), earthquake history and structural analysis records.

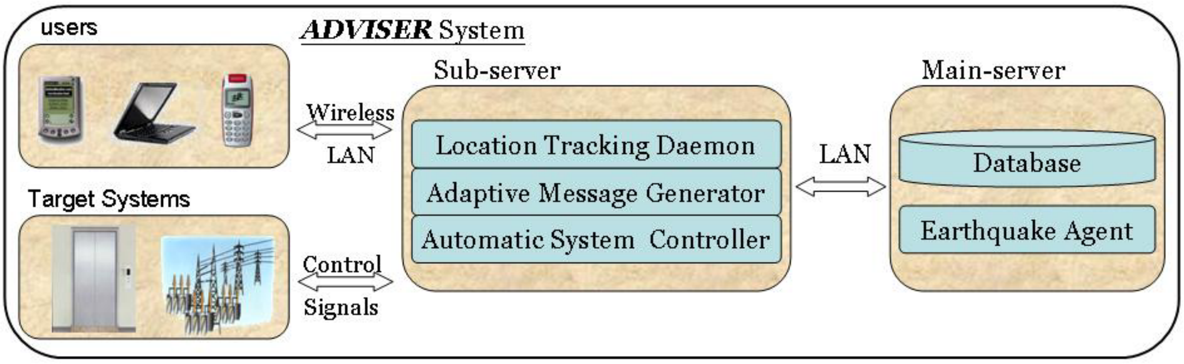


Figure 1. ***ADVISER*** Architecture

ADVISER consists of many components to alleviate the impact of the earthquake (Figure 1):

- **Earthquake Agent** receives and analyzes earthquake information: it gathers the seismic sensor data from the seismology center, and calculates the magnitude and the EAT (estimated arrival time) of the coming earthquake.
- **Adaptive Message Generator** gives adaptive messages to people in various circumstances, according to different magnitudes of the earthquake from *Earthquake Agent***.** For example, based on the structural analysis records, ***ADVISER*** can identify the structurally weak areas such that people located in those areas will be given severe warning.
- **Location Tracking Daemon** (using the 802.11 a/b/g WLAN infrastructure which will not affect the operation of the medical equipments.) monitors people's locations. The location information is an important context for *Adaptive Message Generator*.
- **Automatic System Controller** is responsible for the automatic target systems control.

When *Automatic System Controller* is notified by *Earthquake Agent* that an earthquake is coming, for example, it would *automatically* make the elevator stop and open the door on the first floor for evacuation and *immediately* open the automatic door of the surgery room.

2.2 Performance requirement

ADVISER delivers all adaptive messages and controls target systems efficiently before the earthquake comes. Therefore, real-time operations are required for every component in this system. ***ADVISER*** has to transmit messages fast and lost-free. Besides, ***ADVISER*** should be scalable as the quantity of users grows. In addition, to prevent ill-will disabuse of the system, secure and authenticated messages are required. We have to minimize the probability of false alarm as well.

2.3 Design Methodology

We choose the spiral model [1][2] when developing ***ADVISER***. Since ***ADVISER*** combines the technology of seismology and the application on medical system, we first propose our idea to both seismologists and doctors, and ask them about the feasibility of project and specific requirement. Afterwards, we set the system requirement and break the system into several modules, such as earthquake agent, adaptive message generator, location tracking on clients, UI on clients, etc. The interface between modules is well defined, but could be modified when needed. Each team member can work on different modules in a parallel way. Our design principal is to implement the most basic modules first, then add additional modules one by one to elaborate our system utilities. We try to accomplish a minimized usable system first, and release it to our users for feedback. After getting feedback, we set new requirements for next stage, including fixing bugs, increase reliability, and adding new modules into the existing system.

2.4 Innovation

ADVISER is quite innovative in three parts. First, we adopt the idea of earthquake alarming, which has not been used to alert people in distant before an earthquake actually strikes. With ***ADVISER***, we could actively prevent damage brought by earthquakes rather than passively handle them. Second, ***ADVISER*** is an adaptable system which can handle different situations. Based on locating technology, ***ADVISER*** could give different directions to different users according to their location, roles, and the severity of the coming earthquake, without users configuring their location manually. Last, it is quite inexpensive to apply ***ADVISER*** to existing institutes. Since the architecture of ***ADVISER*** complies with most nowadays network infrastructures, the entry barrier for usage would be very low.

3 Implementation and Engineering Consideration

In the following sections, we describe how we implement ***ADVISER***. Section 3.1 presents each part of ***ADVISER***, including main server, sub-server, and client. Section 3.2 is a scenario which shows how ***ADVISER*** will work if a serious earthquake happens. Section 3.3 discusses the tradeoff issues, and section 3.4 describes the tools developed. At last, we present the testing results in section 3.5.

3.1 The ADVISER System

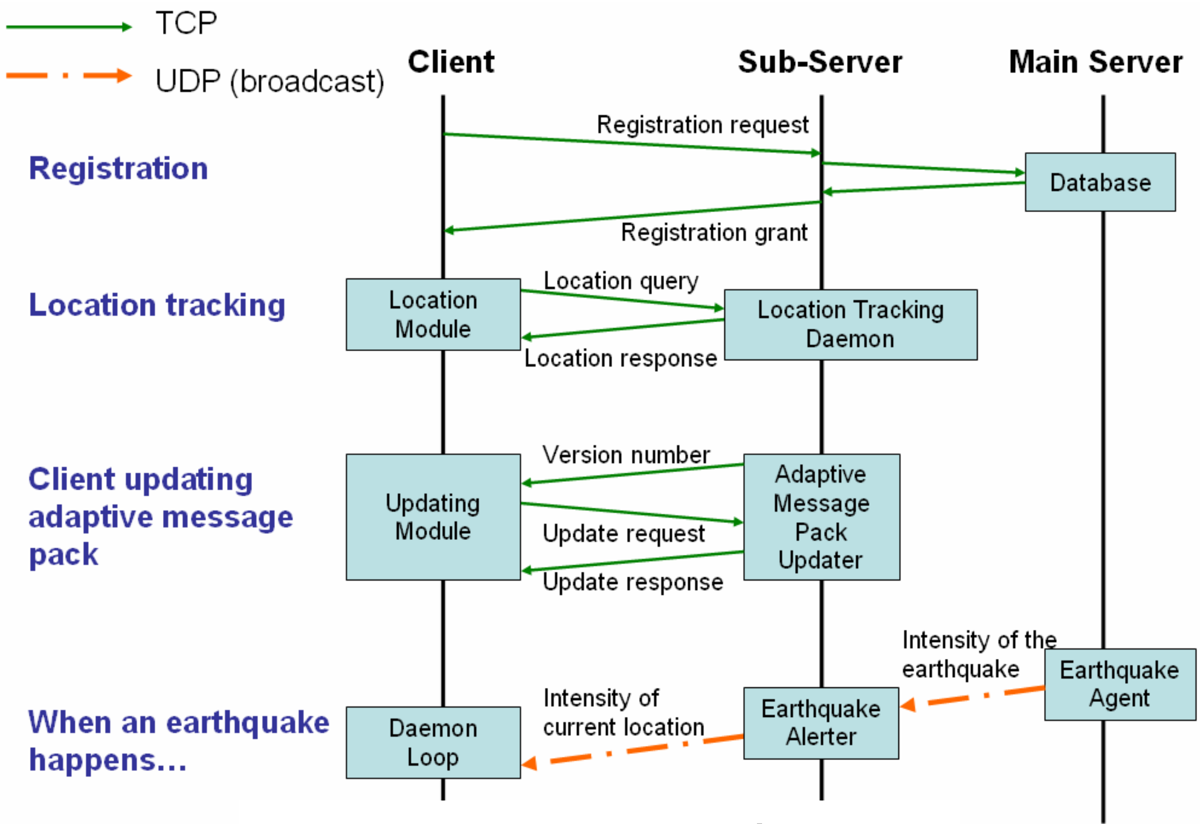


Figure 2. ***ADVISER*** components coordination

3.1.1 Main Server

The main server of ***ADVISER*** is composed of two components, database and earthquake agent. The earthquake agent is divided into two modules, earthquake detecting module and seismic intensity predicting module.

3.1.1.1 Database

There are two tables in the database of main server. One contains information of each user, including name, password, role, etc, which serves for authentication. The other one contains the SOP for different situations and different intensities of earthquake. We use Microsoft SQL server to manage the database and provide a Web interface, based on ASP .NET, for system administrators to modify the content of SOP and user information if necessary.

3.1.1.2 Earthquake Detecting Module

ADVISER receives the information of earthquakes from National Seismology Center (NSC) for further processing via Internet. Once an earthquake happens, the server in NSC will then compute the seismic magnitude of this earthquake according to the information reported from each observation station, using the theory based on seismology [3]. The calculation of the magnitude would be completed in about 20 seconds, and for most earthquakes happened in Taiwan, the time for damaging S-waves traveling from epicenter to densely populated cities is about 30~35 seconds. Thus, we could have about 10 seconds to allow ***ADVISER*** to send adaptive warning messages to alarm users. Once the calculation is completed, the server will generate a P-File, which includes time, magnitude, number of observation stations and the quality. The server in NSC will then send this P-File to ***ADVISER*** server via FTP. The size of a P-File is very small, mostly less than 0.5KB, therefore the transmitting time from the NSC server to ***ADVISER*** server would be very short and would not delay the time for warning the users.

3.1.1.3 Seismic Intensity Predicting Module

Once ***ADVISER*** server receives the P-file, the seismic intensity predicting module will calculate the estimated arrival time, and predict the seismic intensity of our target building, National Taiwan University Children's Hospital. Prediction of intensity is done using the following formula [3],

$$PeakGroundAcceleration = 12.44 \times e^{1.31 \times Magnitude} \times Dis^{-1.837} \times R \qquad (1)$$

Peak Ground Acceleration (PGA) is a real-valued number and has a mapping with the seismic intensity. In this formula the parameter R, which is called site effect, varies from different geological structures, and Dis is the distance from the epicenter. For different earthquakes and different transmitting paths, the value of R differs. As we can see from Formula (1), R substantially affects our final prediction for seismic intensity. Therefore, choosing a correct value of R becomes highly important. Here the predicting module adopts an extensively applied data mining technique, called Support Vector Machines (SVM) [4][5], to choose the value of R for each incoming earthquake. The SVM can do non-linear regression with minimized mean squared error. Figure 3 demonstrates an example of non-linear regression for data with two dimensions. We can see that the resulting curve presents the trend of the data points, and does not fit the noise data points. To apply this technique, we first collect the history data of past earthquakes, including the magnitude, the position, the distance from observation centers, and the measured seismic intensity, to train a model for predicting the value of R. Through the analysis of past data, we found that earthquakes with similar frequencies tend to have a similar value of R. After predicting the value of R for an incoming earthquake, we apply Formula (1) to compute the seismic intensity of the NTUCH.

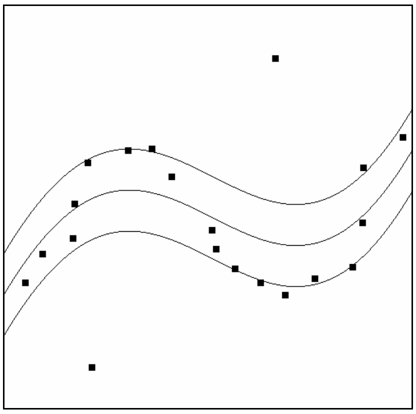


Figure 3. A 2-dimensioinal example of non-linear regression by SVM. The *x* and *y* axis means the coordinate of each point. Each *(x,y)* point is generated by $y = f(x) + \varepsilon$, where ε is a random number which stands for noise. The curve is predicted by SVM.

3.1.2 *ADVISER* Sub-server

The major functionality of the sub-server is to alert people fast with adaptive messages, via the **adaptive message generator** when an earthquake is coming.

A sub-server takes charge of several geographical regions at a time. For example, sub-server$_A$ is responsible for the ICU (Intensive Care Unit) and the surgery room on the 3rd floor (Figure 4). Doctors in the surgery room will be asked to stop operation immediately while nurses in the ICU will be asked to check the breathing machines when an earthquake is coming. Another example is: the sub-servers on the 2nd floor or the 3rd floor can calculate the PGA of the floor respectively and warns people if the coming earthquake is violent.

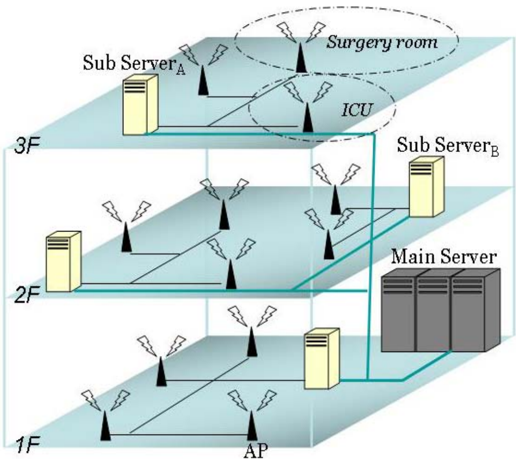


Figure 4. Hierarchical structure and monitored regions

In all, we can categorize the sub-server's functionality as follows:

- **Adaptive Message Generator**
  - **Earthquake Alert**
  - **Adaptive Message Pack Updater**
- **Location Tracking Daemon**

In order to reduce the size of the alert message, we divide **Adaptive Message Generator** into **Earthquake Alerter** and **Adaptive Message Pack Updater** such that adaptive messages could be updated in time of peace while only PGA value of each location is sent in emergency and adaptive messages can be looked up and shown on the client.

3.1.2.1 Location Tracking Daemon

The Location Tracking Daemon provides the important context, location of the clients. Since the 802.11 wireless LAN is a very common infrastructure and GPS cannot support indoor location tracking, we decide to utilize the wireless LAN technology to keep track of people's locations in order to estimate the PGA of the places where they stay. Briefly speaking, there are two phases in the Location Tracking Daemon:

- **Data collection phase**. (preprocessing stage)
- **Execution phase**. (Main stage)

In Figure 5, the scenario of the execution phase is given. $Client_1$ can detect signals of AP_1 and AP_2 whereas $Client_2$ can detect signals of AP_1, AP_2, and AP_3. Both clients send the detected signal strength of each AP and its identity to the Location Tracking Daemon. The Location Tracking Daemon evaluates the received data and estimates the location of the client. Finally, the Location Tracking Daemon sends a message back to tell where the client is.

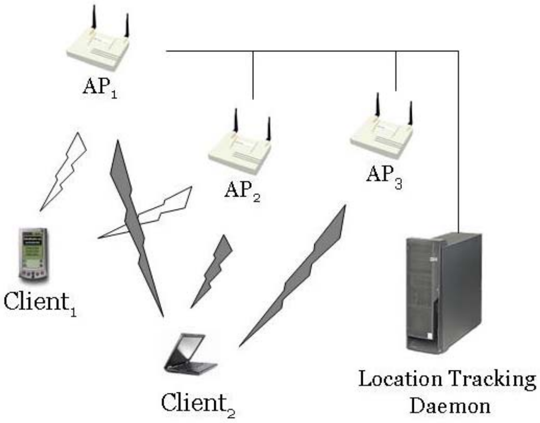


Figure 5. Location Tracking Scenario

To reach the above scenario (Execution Phase), the Data Collection Phase is required in advance. Here we describe the operation briefly. In this phase, a signal map of the whole building is constructed. At some predetermined sample points, people send back the signal strengths from APs. This operation is repeated for *n* times within a period of *t* milliseconds intervals in order to avoid the temporary fluctuation of the signal strength. *n* is chosen to be 100 and *t* is chosen as 10 which makes each data collection step about 1 second long and then the average signal strength is stored, the unit is dBM.

Table 1. The signal map in the Database where *n* is the number of APs in the hospital.

Floor	X	Y	AP_1	AP_2	AP_3	AP_4	AP_5	…	AP_n
3	10	8	0.03	3.74	6.82	0	4.23	…	0

Specification and Algorithm of Execution Phase

1 Receive **query Q** from the client, say $(AP_i, Signal\_Strength_i)\ 1 \le i \le k$

For every **sample point S** stored in the database, say *(AP_j, $Signal\_Strength_j$), $1 \leq j \leq m$*

do

//Find the sample point having the minimum distance with the query

$$\text{Distance(Q, S)} = \sqrt{\sum_{AP_i = AP_j} (Signal\_Strength_i - Signal\_Strength_j)^2}$$

Report the sample point having minimum distance with **Q**.

After finding out the nearest sample point related to the requesting client, a response message is sent to notify the client where it is.

3.1.2.2 Adaptive Message Pack Updater

The Adaptive Message Pack Updater makes sure that clients have the latest adaptive messages pack related to where they currently stay such that clients can look up adaptive messages in emergency. In ***ADVISER***, there is a set of SOP (adaptive messages) related to different kinds of roles, locations, and magnitudes of earthquakes in the hospital and clients may not carry all adaptive messages, so we distribute the adaptive messages related to certain areas in the corresponding sub-server to users when they enter to an area monitored by another sub-server. Adaptive Message Pack Updater uses version numbers to ensure all the clients get the required and latest adaptive message pack. Any modification of the adaptive message pack from the main server would be updated to all clients and clients can request pack after entering a new region.

3.1.2.3 Earthquake Alerter

The Earthquake Alerter broadcasts the PGA of the monitored region of the sub-server if an earthquake is coming. This operation has to be simple and fast. To be adaptive, the sub-server should have the ability to *calculate the PGA of different locations.* Our strategy is to compute the **amplification factor**[1] of every location such that given the PGA ***V*** generated by the *Earthquake Agent* for the coming earthquake, ***ADVISER*** can get the absolute PGA of every location simply by multiplying the **amplification factor**[1] with ***V***. When an earthquake is coming, the main server will broadcast warning messages to all sub-servers with the PGA and EAT (estimated arrival time) of the coming earthquake. After the sub-server receives the

---

[1]**Amplification factor – Structural Analysis**

We use the ground motion history and the building design data NTUCH to perform time history analysis, using the prevailing structural analysis software *ETABS* [12]. The structural responses of the reference point on every floor to the ground motion are gathered. Since we normalized the PGA of the ground motion to 1 in advance, the PGA of every reference point can be regarded as the amplification factor. And the PGA of the location rather than the reference point can be calculated in constant time [13].

message, *for every geographical region it monitors*, the sub-server would broadcast the following message immediately, shown in Table 2.

Table 2. Earthquake warning message

Item	Format	Description
PGA	Float	PGA of RegionID = PGA * amplification factor of RegionID
EAT	Integer	estimated arrival time of the coming earthquake
RegionID	Integer	The *id* of the region.

3.1.3 *ADVISER* client

A client provides users an interface to interact with ***ADVISER*** server. In this report, we choose Pocket-PC 2003 as the development environment. A client can be divided into two parts: the client database and the warning application. After launching the program on the client, it will be registered at our main server. When a user enters a different region, the client he brings with will automatically locate where it is and connect with a sub-server to get the latest adaptive message pack. And then, the client will enter into a daemon loop to wait for the warning message to inform the user or to check whether to locate itself again. This scenario is described as shown in Figure 6, and the application of the client is shown in Figure 7. Besides, if the user closes the window of the program, the client still stays resident on the RAM instead of terminating immediately.

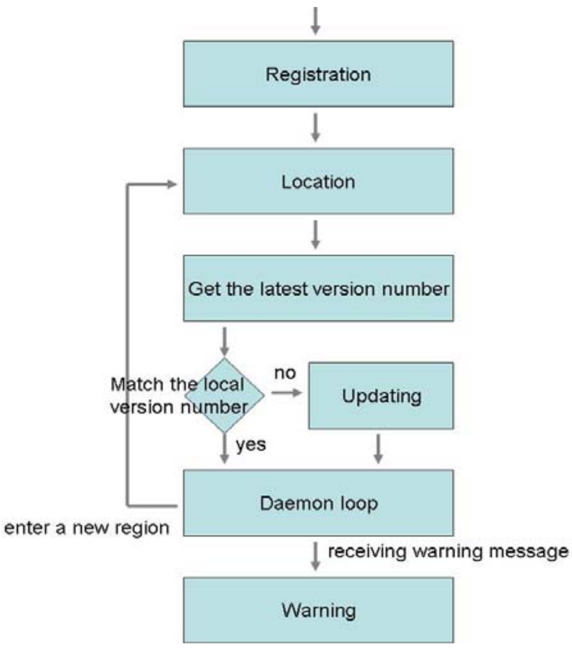


Figure 6. The flowchart of the client.

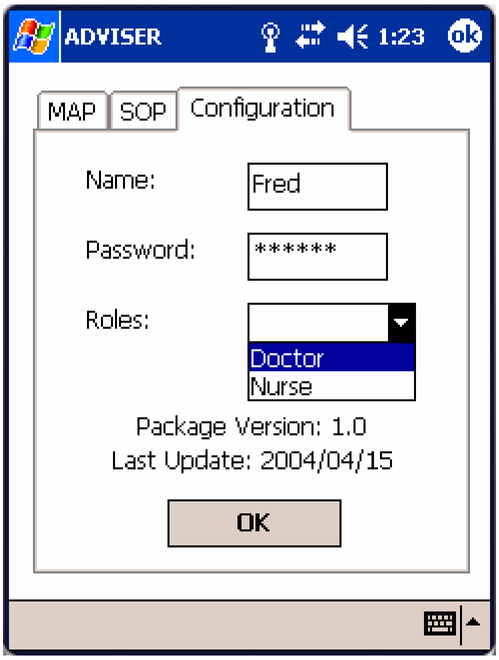


Figure 7. The application of the client.

3.1.3.1 Client database

In the client database, it stores much information to support the warning application. We use SQL server CE [14] as our database server. Items stored in the database are listed as follows.

- The associated adaptive messages.
- The maps of relative regions.
- The list of sub-servers' IP addresses.
- The list of access points' (APs') locations and MAC addresses.

3.1.3.2 Client application

The main goal of the client application is to generate the adaptive message and to warn the user of the incoming earthquake. It is composed of several modules discussed in the following subsections.

3.1.3.2.1 Location module

We implement this part by recognizing the AP which the user uses to connect to the Internet. Because the user may receive several APs' signals at the same time and only choose one of them as the bridge to set up connection, our method is to locate where the user is by analyzing these APs' signals. We use WRAPI (Wireless Research API) [15] to implement the idea, and our algorithm is as follows.

Location algorithm:

1 Search all APs' signals that we can receive.
2 Sort each signal by its magnitude.
3 Choose at most top three of these signals and send a location query message to a sub-server.
4 The sub-server sends back the location response message.

3.1.3.2.2 Updating module

After locating itself, the client will interact with a sub-server to update its local database. If the version number does not match with the current one stored in the client database or even there is not any message stored in the database, the client will send an update request message to the sub-server to ask for the latest adaptive message and update its local database. Then, the sub-server will send back an update response message (see Table 3) that the client needs.

Table 3. The update response message

Item	Format	Description
Region ID	Integer	The region associated with the message.
Role ID	Integer	What kind of people will need the message?
Version	Integer	The version number of the message.
Content Size	Integer	The size of the message content.
Content	Variable	Real data.

3.1.3.2.3 Daemon loop module

While finishing locating and registration, the ***ADVISER*** client will enter a daemon loop that works at background waiting for two different signals to trigger specified operations. In this loop, if the ***ADVISER*** client detects that the region has been changed, it will locate itself again before continuing the daemon loop. Our method is to check the signals we receive from APs and query a sub-server every 30 seconds.

At first, it will look up the local database and retrieve the relative adaptive messages based on the intensity of the earthquake and the location where the user is. We provide three kind of different information:

- Earthquake information: (Figure 8)
  The first thing that people want to know about is the strength of the incoming earthquake. We provide some basic information, including its intensity and the countdown before it arrives.

- SOP: (Figure 8)
  Because people may be nervous or have no idea about what to do after being informed about an earthquake, we provide the SOP which recommends what the user should do.

- MAP: (Figure 9)
  We provide the user a map about the region where he is. This map will be helpful for displaying an emergency path.

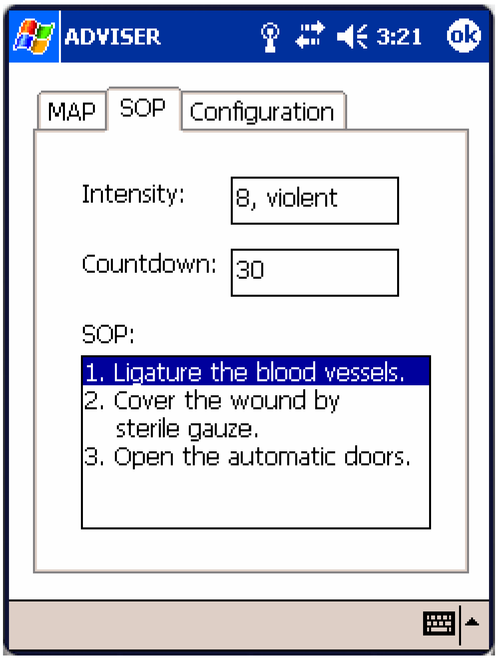


Figure 8. The SOP application.

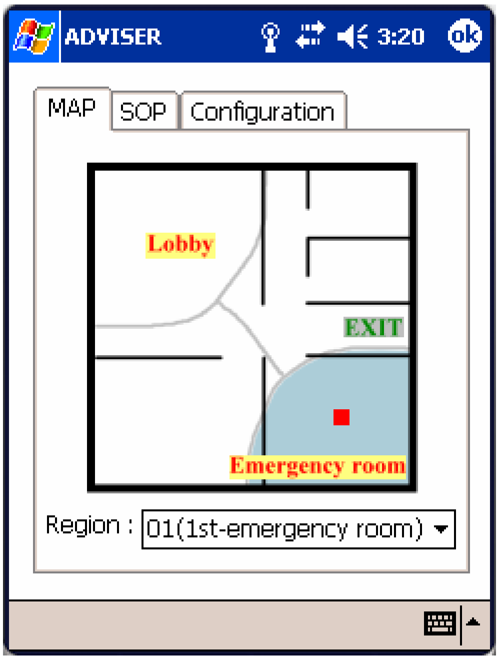


Figure 9. The MAP application.

3.1.4 Target system control

Once earthquake occurs, ***ADVISER*** system will automatically control the target system. The target system connected with the sub-server will take corresponding action to warn people or to save people right after receiving the warning message. We implement two modules described as follows.

- Warning sign module:
  We use FPGA (Field Programmable Gate Array) to implement a warning sign module. It will display the intensity of the incoming earthquake and the rough countdown (see Figure 10), and it is connected with a sub-server through RS-232.

- Door control module:
  The automatic door controlled by our system will be opened automatically after receiving the warning message. We implement the representative prototype of the module (see Figure 11) which could be applied to the real automatic door system in the future. We use motor stepper as the door controller which can drag the door to open. The prototype can receive trigger signals from the LPT port of the sub-server to activate the motor stepper.

Figure 10. The warning sign is composed by four numbers. The first two numbers are the intensity, and the other two are the countdown.

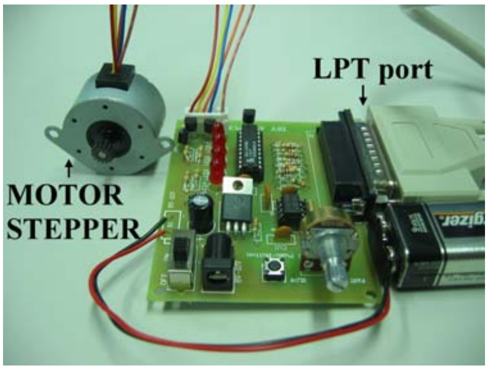


Figure 11. The prototype of the door control module.

3.1.5 Security

In order to save time, ***ADVISER*** uses broadcast to spread warning messages. Because everybody in the same LAN can also uses broadcast, if someone spreads fake messages contrarily, numerous hazards will arise. For example, on receiving messages of a serious earthquake's coming, ***ADVISER*** will automatically open the doors of operation rooms to prevent doors' distortion, but this approach will exposes the patients to virus and disease germ.

ADVISER uses the Digital Signature to ensure the messages' credibility. On the start of the main server, it generates two keys. Messages encrypted by one can only decrypted by the other. The main server holds one for its private key and gives the other, the public key, to all its clients. When a message has to be spread, the main server will encrypt it with the private key. Thus the clients can check the credibility of any message it receives by decrypting it with the corresponding public key. If a message can be decrypted, then it is reliable.

3.2 Scenario

Following we will depict a scenario to demonstrate how ***ADVISER*** could help the hospital to save more lives when a damaging earthquake happens.

Dr. Martin is a cardio vascular doctor in NTU Children's hospital. Everyday when he steps into the hospital, the PDA on his pocket detects the available network, and after being connected to the network the client program will login to ***ADVISER*** with certification of his role and password. On the way to his office, the ***ADVISER*** will update his current position and the life-saving kit. After a short rest, Dr. Martin is going to the surgery room for a cardiac surgery. Catherine is the chief nurse, and every morning she would stay in the nurse station. Unfortunately, a serious earthquake happens in the southern part of Taiwan, which is about 350km from this hospital, and in most cases the S-wave is going to arrive in Taipei in about

35 seconds. After about 25 seconds, Dr. Martin, who is now doing a cardiac surgery, will see the alarm device installed in the surgery shining red light, which means a very serious earthquake. His PDA shows him the emergent SOP and the safest region on this floor depicted by a map. So he would stop his surgery, cover his patient with a clean blanket, and then try to move his patient to the safest region if necessary. Catherine hears the alarm from both her PDA and the broadcast from the hospital at the same time. She would conduct the nurses in the department to evacuate patients to the safest region on that floor and soothe the discomposure of patients.

Moreover, the backup power system would be started up automatically and prepared to take over the original power system if necessary, so that the power supply of the hospital will not be interrupted even if the earthquake damages the national electricity system. The patients using life-sustaining system in ICU will not be affected. The transmission of hot water and gas inside the hospital would be stopped lest the earthquake should make those pipes broken and cause even more serious damage. Every elevator in the hospital would go to the first floor and open the door immediately so that people inside will not get stuck in the elevator.

3.3 Tradeoffs

In developing ***ADVISER***, we considered several tradeoff issues.

3.3.1 TCP or UDP

Although TCP is more reliable than UDP, it takes more time for initialization (three-way handshaking). When an earthquake happens, it is quite critical to send the warning messages to users as soon as possible. Therefore, we choose UDP as message transmission protocol for earthquake warning messages, and TCP for routine updating work.

3.3.2 Adaptive Message Pack Update

The adaptive message pack contains SOP for each user. If we send the adaptive message pack when the earthquake comes, the packet size may be large and it would take a long time to transmit, which is unbearable for earthquake alarming. Therefore, we choose to send adaptive message pack as a routine, and do update if the user has move to another floor or the content of SOP has been changed by administrator.

3.3.3 Locating Policy

The locating module of client will first use the intensity of detected AP signals to locate a region, and then send the intensity of the three highest AP signals along with the code of the located region to sub-server to do finer location. We could have sent the intensity of all detected AP signals directly to sub-server, but this would cost too much computing time to calculate the similarity between all sample points and the detected signals. Hence, we choose to locate a broader region first, and then send the code of the detected region to the sub-server. Then sub-server only needs to calculate the similarity for those sample points inside the detected region. Lots of computation could be saved.

3.4 Tools Developed

We have developed a map displaying toolkit on WinCE. With the input of map image, map coordinate system and people's location, the location-tracking-like application can be easily visualized on the PDAs. Besides, we have also developed a location tracking system based on the 802.11 wireless LAN.

3.5 Verification and Testing

ADVISER analyzes the earthquake information, supports adaptive vision with location tracking information, and fast message alert. So the testing will be focused on these issues.

- The accuracy of predicting Peak Ground Acceleration (PGA) of an earthquake

As mentioned in Section 3.1, ADVISER uses Support Vector Machines (SVM) to help predicting the intensity of coming earthquake. Figure 12 demonstrates the predicted PGA of each earthquake. The pink line is the predicted PGA by SVM, and the green line is a default parameter for site effect parameter, which is a conventional way used in seismology. As we can see from Figure 12, the predicted result by SVM is quite close to measured PGA in blue, which shows high accuracy. The green line performs well only when the PGA is very low. When PGA is much higher, namely a serious earthquake is coming, the green line will predict a much higher PGA which yields false alarm.

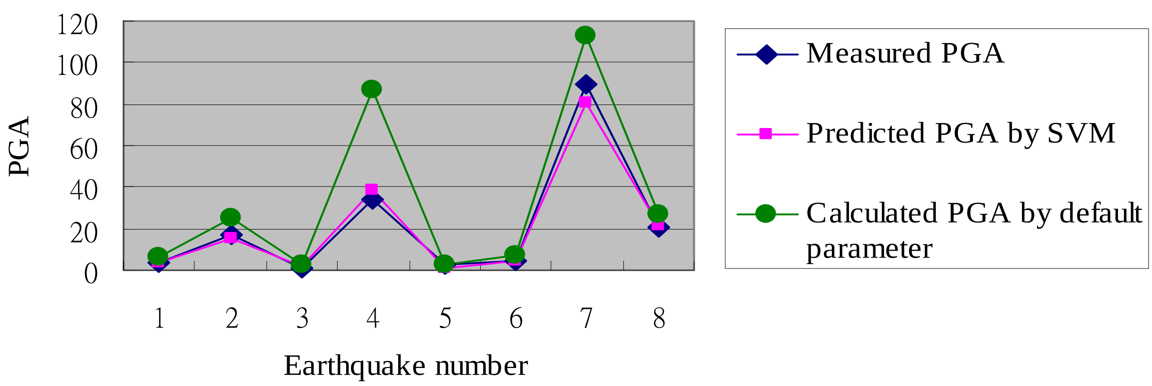


Figure 12. PGA of each earthquake

- The latency from the main server to the client:
  Using the 25 earthquakes history, the main server broadcasts the alert messages to 2 sub-servers with 2 clients in the monitored area respectively (4 clients total). We have recorded the latency from the main server to the client and the average latency is 44.79 ms, see Figure 13.

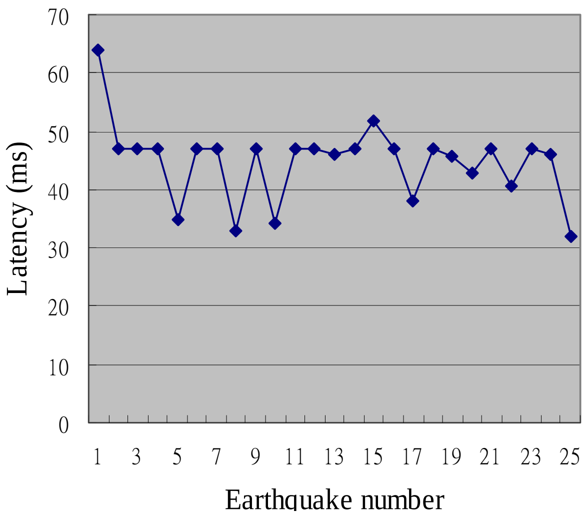


Figure 13. Latency of each earthquake

- The time for ***ADVISER*** users to prepare for the coming earthquake.

Here we present the latency that the main server receives an alert (P-file) from NSC versus the latency the earthquake waves arrive at the hospital, after an earthquake is detected far away from the hospital. As we can see from the previous experiment that the latency from the main server to clients is very small, the difference can be regarded as the time for ***ADVISER*** to prepare for the coming earthquake, see Figure 14.

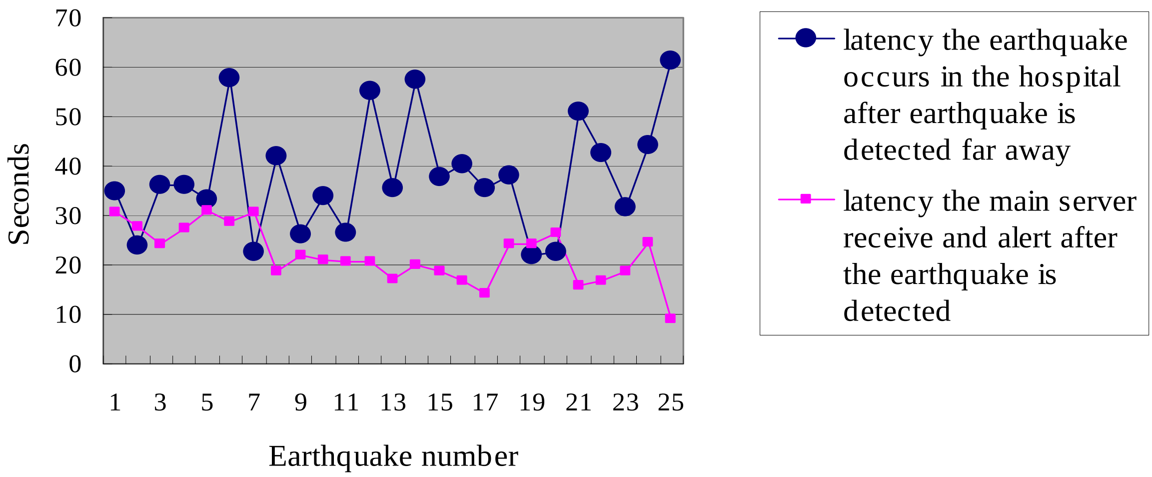


Figure 14. The difference of the two curves is the time for ADVISER clients to prepare and evacuate for the coming earthquake.

Though in Figure 14 there are several earthquake cases that ***ADVISER*** warns people after the earthquake has already happened (in the 2nd earthquake for example). However, the reason is the distance between the epicenter and the hospital is quite short, see Figure 15. (In Figure 15 the earthquake number in x axis matches that in Figure 14.)

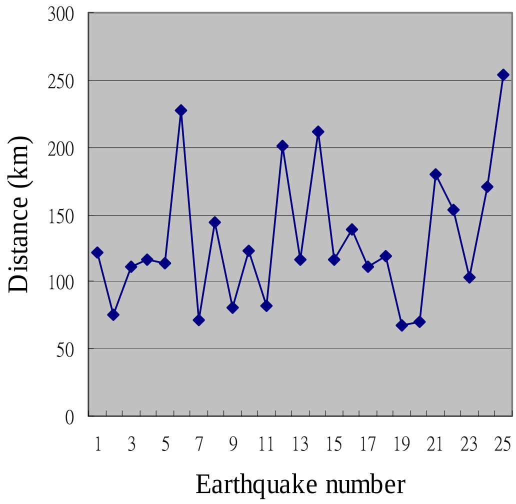


Figure 15. The distance between the epicenter and the hospital.

- The accuracy of the location tracking system.

Upon the consideration of cost, we want to know how many APs are required to locate a client. This factor is measured through a set of experiments. We locate the testbed on the third floor of a 4-story building which has dimension of 35m by 18m. Keeping all other factors constant, we only change the number of APs on this floor. Signals from APs on the other floors will be automatically filtered out because of their weak strength, so our test will not be influenced by them. We randomly select 100 different spots in the building and for each one we measure the error distance between the estimated position and the real one.

Figure 16 shows the cumulative frequency of the error distance for various number of APs. The average error distance is 8.1m for one AP, 4.6m for two APs and 2.2m for three APs. Since the smallest region defined in ADVISER is about 5m by 5m, three APs are sufficient to locate a client precisely.

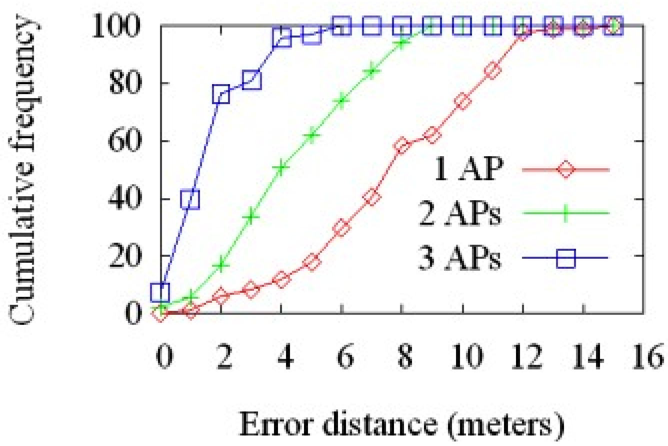


Figure 16. Cumulative frequency of the error distance vs number of APs

4 Summary

4.1 Status of the project

ADVISER is fast, scalable, robust, and inexpensive. According to the design methodology, we now achieve the basic requirement and prepare to go to the next stage. We proposed our idea on December 2003, and asked the seismologists and doctors for great help on the next month. The core system was developed on March 2004. We will continue developing advanced modules from now on. At the next step, it is expected to take about two months to upgrade the work and improve the performance. We set several targets as follows:

- Improve authentication and security
- Create a more user-friendly interface
- Enhance the accuracy of locating
- Increase the efficiency of storage utility

4.2 Further work

Our initial focus was on the basic requirement of the system. After accomplishing the above targets, we are interested in improving the system. We will add more target system modules, such as elevators and power supply system, to make the place safer automatically. On the other hand, currently the client is implemented on Pocket-PC, smart phone is also considered to be a good mobile device as the client environment. Besides, time is a very important topic in the project; we will also decrease the time we spent on message transmitting and computing. We can save more people if we can save more time.

4.3 Conclusion

ADVISER provides useful information before a lethal earthquake causes a catastrophe. To achieve the goal, we implemented the system with several features, which are earthquake analysis, message dissemination, wireless location capability, adaptive message generator, warning application, and database updating. Besides, ***ADVISER*** is designed using a hierarchical architecture which can support a fast and reliable service. Users who use PDA as the client device may get better quality because of the mobility and better performance.

In summary, ***ADVISER*** is convincing for several reasons. First, the research on seismology has been well developed, and the performance of our system will improve much further with the advance of this research area. Second, PDA is an amazing device with unlimited mobility and enough quality. It is a trend to have such a device within our wireless life styles. Finally, with the well-defined network protocol, such us IEEE 802.11b, we can provide a fast message transmitting and real-time information. Therefore, ***ADVISER***, ADaptable VIsionary System for Earthquake Resolution, will undoubtedly make the world a safer place.

5 Reference

1. Ian Sommerville, Software Engineering, 6th Edition, Addison Wesley, 2001.
2. Clemens Szyperski, Component Software, Beyond object oriented programming, Addison Weseley, 1999.
3. Y. M. Wu, T. C. Shin, and Y. B. Tsai, "Quick and reliable determination of magnitude for Seismic Early Warning", 1997.
4. Vapnik, V. The Nature of Statistical Learning Theory. New York, NY:Springer-Verlag. 1995.
5. C.-W. Hsu, C.-C. Chang, C.-J. Lin, A practical guide to support vector classification July, 2003.
6. J. Hightower, G. Borriello, "Location Systems for Ubiquitous Computing", IEEE Computer, August 2001.
7. N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. "The Cricket Location Support System", MobiCom'2000, August, 2000, pp. 32-43.
8. Ekahau, Inc., Ekahau Positioning Engine™ 2.0 Specification, http://www.ekahau.com/pdf/Ekahau_Positioning_Engine.pdf, 2003.
9. Can Komar and Cem Ersoy, Location Tracking and Location Based Service Using IEEE 802.11 WLAN Infrastructure.
10. Robert J. Orr and Gregory D. Abowd, "The Smart Floor: A Mechanism for Natural User Identification and Tracking", CHI'2000, April 1-6, 2000.
11. Changhao Jiang, Peter Steenkiste, "A Hybrid Location Model with Computable Location Identifier for Ubiquitous Computing", Ubicomp 2002.
12. Computers & Structures: ETABS http://www.csiberkeley.com/products_ETABS.html
13. McGuire, W. and Gallagher, R. H., Matrix Structural Analysis, John Wiley, USA, 1979.
14. Microsoft SQL Server: SQL Server CE Home http://www.microsoft.com/sql/ce/
15. Wireless Research API http://ramp.ucsd.edu/pawn/wrapi/