Department of Computer Science and Engineering
Netaji Subhas Institute of Technology
University of Delhi
Delhi,INDIA.

INFOSTATION:
Fostering growth the DigitALL way.

Team Name: Chip de Carpediem

Team Members:

Dr. M.P.S Bhatia (Faculty Mentor)

Gursharan Singh
Manas Mittal
Arvind Batra
Kanav Abrol

Windows Embedded Student ChallengE
Final Report
April 23, 2005

1. Abstract

From time immemorial, there has been a quest for development of newer technologies to ease our lives and make this world a better place to live where each one has the desired and the required information at the right time. However, in a country like India with over 44 % of the population illiterate, it is especially challenging to bring the benefits of the new 'Information Technology' to the people. Yet it is these very people that the technology is likely to benefit the most, especially since most currently live in information-deprived state. We identified the challenge as trying to develop a system that would bring to the people information that they need, without preconditions like literacy.

Traditional information providers like Television or Public Radio involve large production costs and hence tend to cater to a numerically large audience base. A Radio or a Television station covers the state, or at minimum a zone. Ergo, the information they delivered is not localized to a fine granularity level of a small community. Hence, a system that could provide customized information to small communities, perhaps rural, was missing. Our project *InfoStation* attempts to provide a solution.

The "*InfoStation*" is a comprehensive information acquisition, processing and delivery system. It uses novel approach at each step and can be deployed in small communities like villages and provide locally customized information. The information may be in the form of raw information, like local weather reports, selling prices of agricultural produce or health alerts. Addiotionally, the system is capable of using locally deployed sensors as well to acquire data. The device is 'smart' and can process data and come up with suggestions based on known relationships. For example, if it is dry season and the maize crop is in its fourth week, the system will generate a suggestion that it must be watered twice a day.

The system is ideally suited for community deployment and is both autonomous and education agnostic. Further, it does not require the individual information consumer to make any purchases or skill additions. We envision the *InfoStation* as a way to bring the benefits of IT to the lowest common denominator of the world, and to help them rise from that status.

2. System Overview

2.1 High Level Design Specifications

Our design objective was to create an Information system that would acquire, process and then provide customized information in a manner which adhered to the following design requirements:

1. **Customized**: The information must be customized / localized to a fine granularity level that is to provide localized information

2. **Education agnostic**: Lack of Education should not prevent the individual from obtaining information.

3. **Autonomous:** The system itself should be autonomous or should require minimum supervision and upkeep.

4. **Cost effective:** The system should finally deliver information in a cost effective manner, i.e. – it should be within the reach of its intended target audience.

5. **User Friendly**: The system should deliver information in a form familiar to the end user and should not entail the learning of a new skill to become an information consumer. For example, the user should not be expected to learn how to type. Also the language of delivery of information must be the local language.

Our final product, a small device we call the "*InfoStation*" is a small footprint device built on ICOP's e-Box reference board running a custom build Windows CE version. The software running on it uses interactive web services and external sensors to acquire customized information. Some of this information may be of general interest and will be stored locally to be disseminated. Other information will be fed into a locally deployed expert system (AgriExpert) to produce tangible, customized advice for the user. The information is finally disseminated by first running a text to speech engine on it and converting it to local language speech and then broadcast over FM airwaves.

2.2 Performance Requirement

The system must adhere to the following performance requirements:

1. **Quality of output of Text to Speech (TTS) Engine**: Even though text to speech technology has matured, we were concerned about the software available for the small footprint device platform. The speech rendition must be easily understood by a listener.

2. **Independent**: The system must perform its work independent of human supervision and should require no upkeep.

3. **Low Power Requirement**: Since the system may be deployed in rural developing areas, power consumption and the possibility of powering it using alternative energy sources (solar) feasible.

4. **Continuous Transmission**: The FM transmission must be uninterrupted. The setup must be able to run multiple processes such as downloading web service information at the same time when the TTS is doing the conversion.
.
5. **FM Transmission quality**: The FM transmitter deployed must have adequate coverage and frequency response.

6. **Remotely updatable:** The application must be remotely updatable. This complements the ability of the process to run unsupervised.

2.3 Design Methodology

We designed our applications as separate components and amalgamated them into our project as and when they became available. We chalked out a black box approach in the beginning of the project. Output and Input for each component was clearly defined and design changes had to be negotiated between module owners. One of us worked on getting data from the relevant web services and local sensors. Another person worked on the expert system. Another person worked on getting the Text to speech engine to work. Finally one of us build the operating system image and ensured that people's work was compatible. But there were certain issues that demanded whole team's efforts and were addressed accordingly.

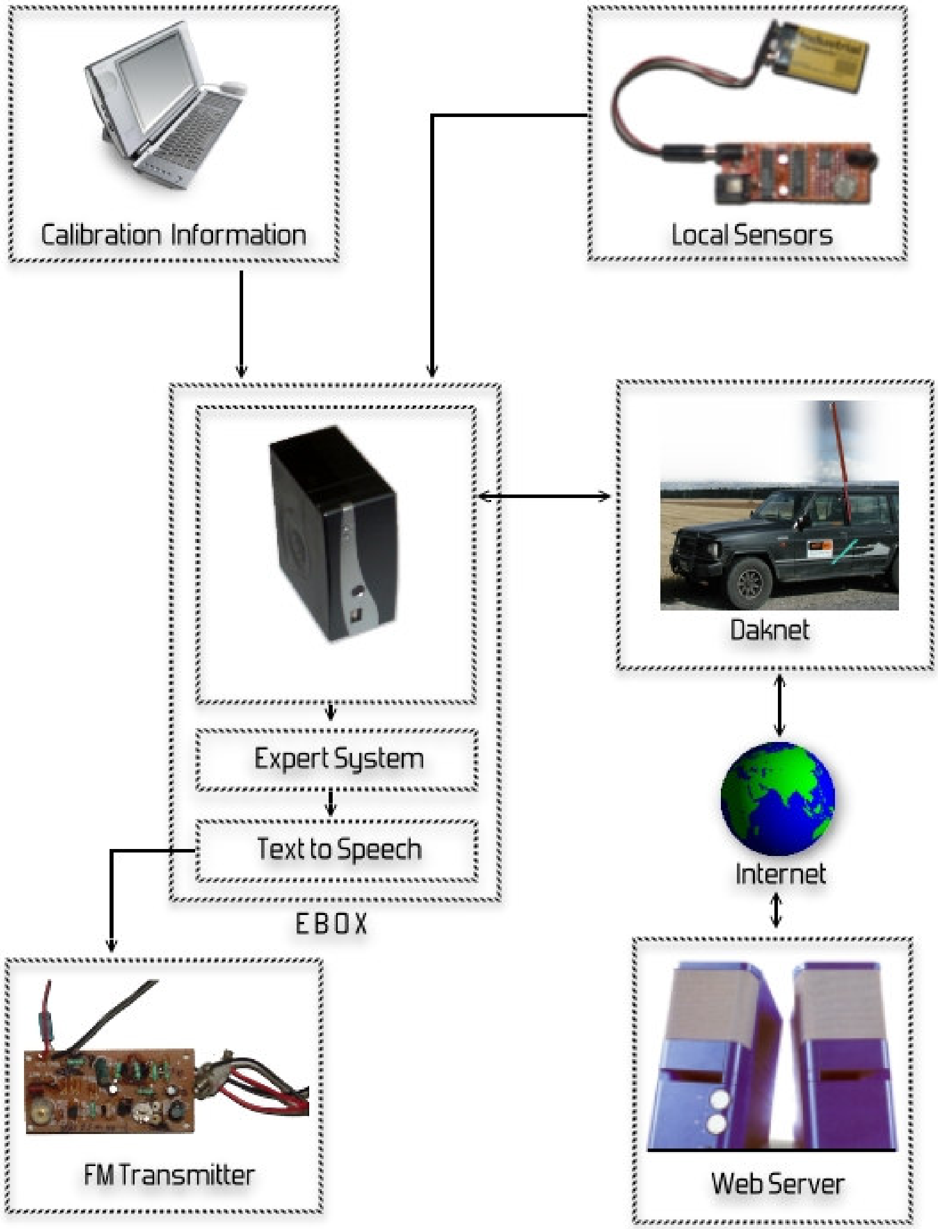
The modular approach allowed us to work in parallel. Further we swapped modules to test them before accepting them for amalgamation into the main project. This project entailed for us learning several new technologies, and this division of labor enabled each of us to concentrate of particular new skill.

2.4 Innovation:

Our approach of using web services to acquire and broadcast information in this manner over FM using Text to speech is a new application with tremendous implications. This application can become the killer app driving community centric web services. Further, this approach removes the disadvantage faced due to lack of education. It epitomizes the spirit of going beyond boundaries, for it opens new avenues of knowledge for those who have been deprived of it….for way too long.

Also, using multiple web services and correlating the required information, like weather and water to finally come up with a suggestion using an expert system represents a new way of looking at things. Finally, the nice thing about all this is that it works marvelously well on the Windows CE / e-Box platform which was readily available to us.

Most of the technologies we used were already available. Text to speech has been there for years, so have been FM transmitters. Yet we believe that we were able to go that last mile and come up with a new solution that directly benefits people.



1

Fig. 1: A pictorial representation of the working of the system with actual components representing the flow of control among various components.

3. Detailed System Discussion:

To begin with, we were given an e-Box to work with and quickly realized that it was well suited to build our application on. That's because firstly it runs Windows CE, a mature Operating system with a good platform of development tools (the .NET platform) as well as availability of Text to Speech Engines (TTS) from commercial providers. Secondly, the e-Box has extremely frugal power requirements, a factor that was important to us since we may have to rely on alternative energy sources to power it in remote corners of the country.

We have developed a sample application in which our system will give advice regarding agricultural techniques and current requirements. This application is not, in itself the only use of our device. Instead, we are using it as a proof of concept of the usefulness and practicality of our system.

3.1 The Algorithm:

1. Acquire calibration information that identify the region / customize the Information available or can be used to feed the Expert System.

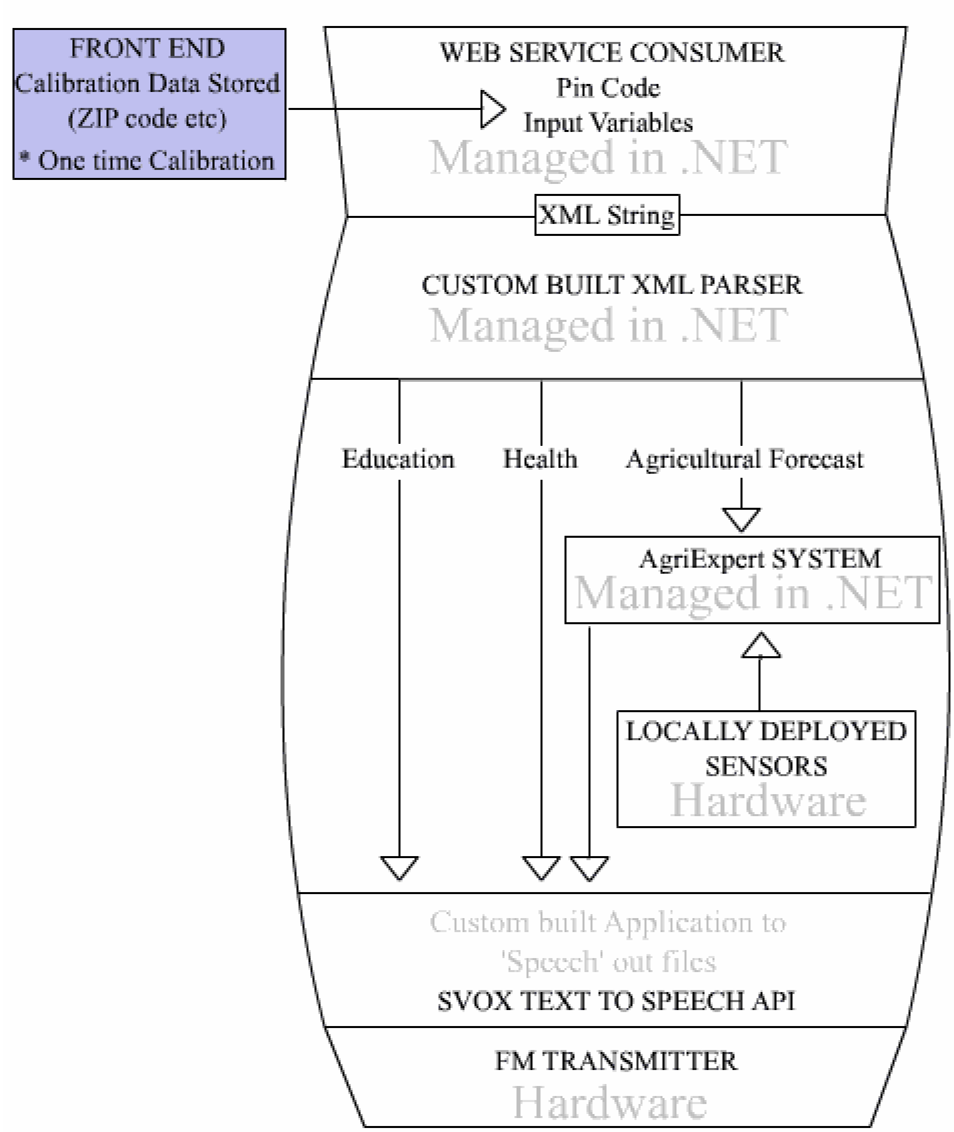
2. Use a web service to acquire information that would be of interest, as identified using information gained through step 1.

3. If possible, gain additional information using locally deployed sensors.

4. If possible, use expert system(s) to process information obtained using steps 1, 2 & 3. This processed information would be extremely customized in response to the demands and interest of the consumers.

5. Use a Text to Speech engine to convert selected information from steps 2,3, & 4 to spoken form.

6. Use a FM Transmitter to broadcast this information locally.



3.2 Start up Step:

The system initially asks for calibration information containing region identifiers: ZIP / PIN Code or the City Name, Country Name, the type of crops grown where the device will be deployed, when was it first sown etc (for information through AgriExpert) and this information is stored locally in the form of XML files. The software will use this calibration information to pull out customized information from the internet using web services. This information needs to be entered only once or when it changes. This can be done on site using a simple GUI interface, or alternatively, it can be done remotely from a central location. Given the tendency of villagers to take collective decisions with regard to what they want to grow, the *InfoStation* is likely to serve a homogeneous group of people and can hence serve them well.

3.3 Web Services:

The sample web service that we are using is a freely available weather service called Global Weather from www.webservicesX.com [2]. While the current web service only gives weather information for only a few selected locations, we envision rapid proliferation in services providing customized Zip / Pin code level forecasts. Already such forecasts are available for the United States and countries like India, with extensive remote sensing and weather satellites are also capable of providing it.

Internet facility will be typically provided using schemes like 'DakNet'[1]. DakNet is a system in which a vehicle with an internet connection through satellite link drives up to a village and provides internet access to the region temporarily using Wi-Fi access. The project is currently operational in parts of India and Cambodia. DakNet[1] is just one simple way to provide internet access to the e-Box. As one demonstration of the practicality of this new technology for rural connectivity, researchers from the Indian Institute of Technology at Kanpur, working with Media Lab Asia, have "unwired" a 100-sq km area of the Gangetic Plain in central India. This project provides broadband connectivity along a corridor with almost one million residents. Where available, we can, of course, use dial up connections etc to get continual internet access to the e-Box.

The web services have been implemented asynchronously thereby allowing us to poll multiple web services at the same time and also ensure uninterrupted audio broadcast at the same time. Multiple threads have been used for implementing XML parsing concurrently with other processes.

3.4 Additional Data acquisition system:

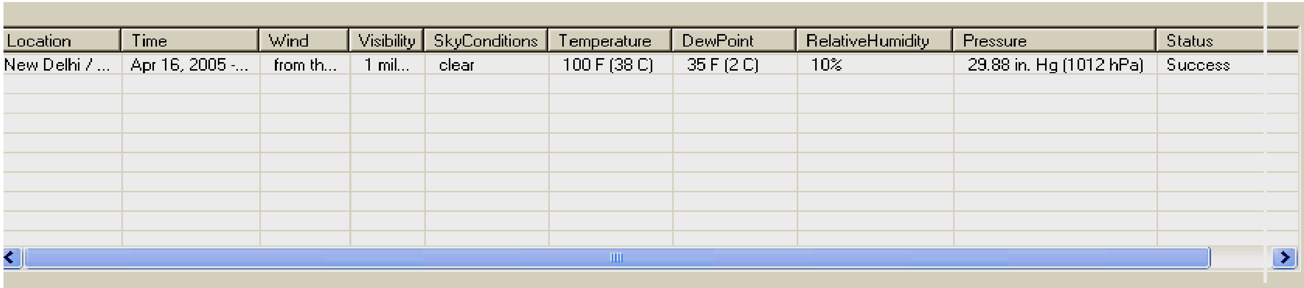
A simple weather probe [4] is used to sample current weather conditions. This device, built using a PIC microcontroller uses Temperature, Humidity and Pressure sensor to acquire this information locally. This information can be utilized by the Expert System. We further envision usage of additional acquisition systems like soil analyzers, water analyzers etc. They can be easily incorporated in our system and would add to the information available to the end user, both as raw information and better expert system deductions.

Some standard text files which could be public messages or educational files that could be of public interest were included to playback in routine.

3.5 Locally deployed Expert System:

The Expert System: AgriExpert system is used to derive suggestions about farming. As an input, the names of locally cultivated crops have been stored along with calibration data. This rule based system incorporates information about the climatic conditions suitable for the crop(temperature range, humidity range, pressure, dew-point required) and its fertilizer and water requirements depending on the time of sown of the crop seeds for each crop variety. Although the expert system is a rather simplistic one in terms of the current database, yet it effectively illustrates the power and effectiveness of such a system.

The schema of the expert system used is shown below. The min and max parameters represent the range of values of the respective variables. The parsed weather information obtained from web service [2] (see fig. 3) is compared with this database and the matched results are retrieved. The matching is done by using a sequential search algorithm.



Location	Time	Wind	Visibility	SkyConditions	Temperature	DewPoint	RelativeHumidity	Pressure	Status
New Delhi / ...	Apr 16, 2005 -...	from th...	1 mil...	clear	100 F (38 C)	35 F (2 C)	10%	29.88 in. Hg (1012 hPa)	Success

Fig 3: Parsed information from Weather web service [2]

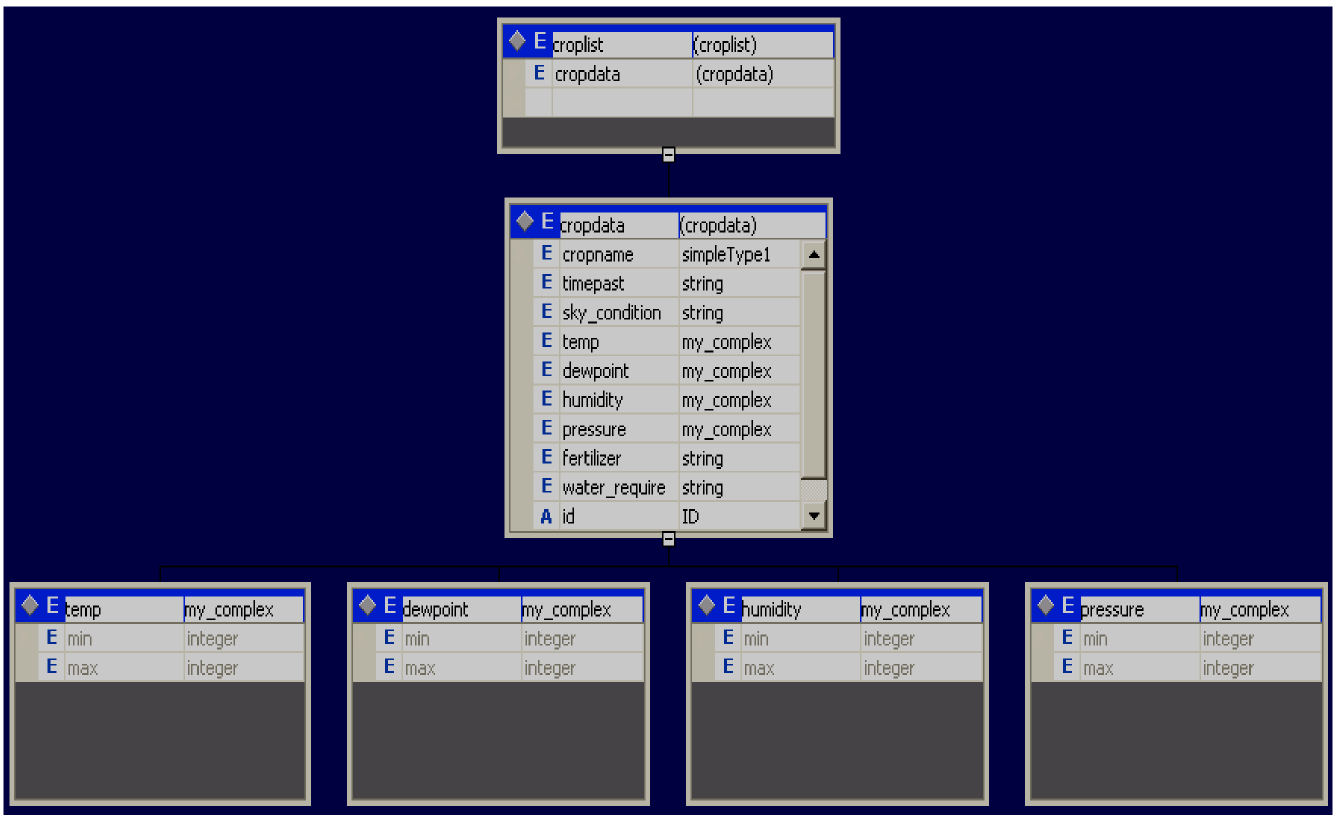


Fig 4: Schema of the AgriExpert system deployed locally

3.6 Text To Speech:

We are using a SVOX [5] Text to speech engine converter sponsored by SVOX [5]. Although we are using an English language TTS Engine, the language of deployment will finally depend on the regional language of the target community. Regional language text to speech engines are increasingly becoming available from both private and Govt. organizations. For example, we have been intimated by the Ministry of Information Technology, Govt. of India that they have been working on regional language text to speech engines.

The Text to Speech interface uses the SVOX TTS API defined by the header svoxttsapi.h. The system is based on the concept of **synthesis channels**. The input to a synthesis channel is the text to be synthesized, and the output of a channel is the resulting speech signal, represented by sequences of signal samples. The file channel takes the input text file and produces the synthesized speech to an output wav file which is then played by the computer audio. We have our application that takes the input as the file stored in a directory in the root of the device and when launched takes the files one by one to speech out. As we were provided with the evaluation version of the TTS engine, we were unable to explore the enhanced features of the engine and merely managed to do what was required. It was not easy for us to go further even after requiring the TTS engine as now an application on their API was required to be built.

The speech application that fitted into the system required MFC 3.0 which was not available with window CE version 5.0 and hence it did not run in the windows CE environment. With changes in the application (native code) and after requesting for the compatible DLLs from SVOX, the application will be up and running. At present, the test application we have built is working fine but still some speed and pitch corrections are to be done which we will correct in due course of time with some help from SVOX technical support. It must be noted that we have been only given the TTS API and the engine and not the SDK, so even to correct things like pitch and speed, technical assistance was required which we were able to get from SVOX. Technical support was required because we do not recognize some of the variables that are used in the API.

3.7 FM transmitter

The FM circuit we have used has been made by us completely with design and testing at our Communication systems laboratory, including the fabrication of the PCB. It costs around less than $4 to make this circuit which provided a fairly clear voice unto a range of 2 meters. The circuit is a simple prototype of the commercially available FM transmitters and we intend to make use of one off the shelf FM transmitter providing better clarity and range at a later stage.

3.8 Programming Environment

We have used the .NET programming environment since it made it easy for us to consume web services, and we have written it largely in .NET managed code based on Compact Framework. We benefited extensively from [3] and [6].

Native code written in eMbedded Visual C++ has also been used to write test speech application.

3.9 Choice of Windows CE

The choice of an Operating System for such type of an application is a critical issue. Although we were required to use the Windows CE OS, we could not proceed before we convinced ourselves that the choice of Windows CE is apt.

3.9.1 Why Windows CE?

Choosing windows CE as an Operating System was critical for the type of application we have built. A Hard Real Time system was required because the Network Access was provided for a limited period of time and all the data upload/download was to be completed in a stipulated period of time. Moreover, the small device had limited hardware resources(memory, storage and processor) and hence was required an OS for small footprint devices.

Windows CE also gave superb functionalities that enabled us to build, debug and deploy the customized OS image. Came with Windows CE, the Platform Builder, an IDE that made the job of building a custom Operating system image, a game. The emulation environment was also comprehensive and played a major role in the development and testing of the applications. Windows CE error reporting provided the reliability and the features such as Indic Support and multi language script support ensured extensibility.

3.9.2 Development of the OS image

We were having a course on Operating Systems. When we registered for WESC we were thrilled to think that we will get to know the nuances of building and making a modern Real Time OS-Windows CE. After submitting the project description, we started waiting for the Development Kit to arrive. The Platform Builder was a heavy bit on the network (it was more than a GB of download) and all we had was an internet Connection supporting maximum download speed of 3-4 kbps. After around 2 days, the installation was complete.

The OS image development started with identifying the frameworks, drivers, classes and libraries that were to be included in the Windows CE image to make our applications run. Also, as our application contained a TTS engine, the dynamic link libraries that came with it were included in /windows to make the associated applications link to them during run time. We take an object oriented approach to include the catalog components i.e. a component was included in the image when our application that demanded or required that component was ready to be tested. So, we describe in brief the need for various components that were included in the image as below-

- .NET Compact Framework- The counterpart of the .NET framework (Desktop) on handheld devices, this component was included to provide a running environment to our applications that were built using C# in the .NET IDE.
- C Libraries and Runtimes for to include STDIO and STDIOA used in a launch program.
- COM and DCOM to enable communication between processes and communication to the network such as for Web Services.
- Microsoft Foundation Classes (MFC) to support our TTS engine and the related applications that invoked some of the MFC DLLs.
- OBEX - incorporated for future support like enabling weather probes to communicate with the expert system via InfraRed.
- SOAP toolkit for Web Service functions.
- Speech Interface (SAPI 5.0) for some Text to Speech specific invocations and also for future support to incorporate Speech to Text support.
- XML Core Services and DOM model for the XML string operations to provide formatted input to the Expert System.

- Audio Compression Manager (ACM) and Waveform Audio, Mp3 audio, WMA and Mp3 local playback and WMA and Mp3 streaming (future support) for the Text to Speech Engine.
- Networking – WLAN (802.3 and 802.5) Wireless LAN, TCP/IP support , Network Utilities and domain discovery.
- Device drivers – RealTek Ethernet card support , native WiFi drivers, NE2000 compatible PCI card, Serial port and USB function drivers.
- Internet Client Services – IE 6.0 with HTML/DHTML API, IE Multiple Language Base API and Windows Internet Services.
- Windows CE Error reporting component.
- Platform Manager.
- Third Party Components – ICOP Vortex x86 BSP and our application CEC files that by default come under this category.

The developed applications were added to the OS image (some of them were deployed on the fly using the remote file viewer and process viewer tools) as third party catalog components using the CEFileWiz tool available from Mike Hall's WebBlog [6].

3.9.3 Deploying the OS image

The image was deployed on the Ethernet with a cross over cable connecting the e-Box directly with the development station and both the download and transport mechanisms were set to Ethernet with KdStub providing the debug support. The tools such as the Remote file viewer Remote Zoom- in , Remote Registry Editor, Remote Process viewer and the Remote Call profiler proved instrumental in optimization and design procedures with all of them working on the Kernel Independent Transport Layer (KITL) and using the CESH server for start up.

3.10 Tradeoffs

When we decided to use a text to speech application, one of the prime questions facing us was, would we have the engine run on a remote server and sends the data in an audio format, or should we instead send the text and run the speech engine locally on the device.

We were concerned about the quality of internet access that is going to be available to us. We believe that novel solution like the DakNet[1] may be the only internet access available in rural areas in the developing world. Hence we worked on the assumptions of continual (typically half an hour a day) internet access and poor internet bandwidth. Therefore we chose the second option i.e. to run the TTS on the e-box. Further, the system continues to remain useful even without any internet access, broadcasting locally acquired information (from sensors) and expert system outputs generated.

A similar choice existed with respect to deploying an expert system. A localized expert system is used instead of a centralized one because:-

a. It contains information highly specific to the area; depending upon area's soil composition, irrigation facilities available, financial condition of its inhabitants: i.e. can they afford high quality genetically developed seeds etc. A centralized database is bound to be bulky and would entail matching the localized parameters with the database elements and hence consume a lot of time.
b. This database information is not dynamic. Hence there is no need to download it every time from internet. Hence low bandwidth requirements
c. The updation if required is also supported. The central main agency which is the owner of this project can ftp new updated xml file to this e-Box. This ftp file will override the previous xml file.
d. Moreover, since the information is stored in Xml based text files, it requires very low storage requirements suitable to our requirements.

One key decision was about the level of user interactivity to be provided. Since we wanted the system to be inherently autonomous, we have eliminated those requirements by use physical sensors and easily updateable (locally/remotely) expert system calibration information (like the crop being grown). Our intention was to ensure that the system remains autonomous and hence can be community deployed, rather than handed down to one 'supervisor' who would exercise total control on it. We want the device to be a quiet box mounted on a lamp post steadily running yet making the loudest noise on the airwaves.

3.11 Testing

We did component level testing (Black box approach) to ensure that each of the individual components were robust and performing optimally. The XML parser we developed was able to successfully parse data following any XML schema. The web services used did, of course, follow a standardized schema known to our system a priori. For example, variables like Temperature returned by the web services needed to have the standard 'Temperature' XML tag attached to it so that retrieved information is correlated with the respective parameters of expert system. The AgriExpert system was found to generate text files of the matched results.

Acquiring the TTS engine has been quite a difficult task; however we have taken the initiative and have been the first Windows Challenge team to have acquired a text to speech engine for Windows CE (sponsored by SVOX, Switzerland). At present, the test application we have built is working fine but still some speed and pitch corrections are to be done which we will correct in due course of time with some help from SVOX technical support as has been mentioned before.

The FM output on the self made test FM transmitter was of decent quality with a range of about 2 meters. However we recommend a well developed Crystal oscillator based (commercially available) FM transmitter which has high quality sound with a range (radial) of 2-4 kilometers.

The *InfoStation* was tested on extreme cases and was subjected to conditions faced that might occur in actual use. We intentionally disabled the internet service to check the system's response. We also manually switched off the power supply. On switching off the power supply, the whole system rebooted. The text to speech conversion engine started to read from the start of the text file instead of ideally continuing from the point of conversion at which the power was turned off. The whole procedure started off automatically since we included the application in the OS start up folder.

Finally, we would typically expect the system to work with wireless internet connectivity. We have included this capability in the Windows CE image that we have built. However, for testing purpose, we have simulated this connection using a wired Ethernet connection. We plan to use a wireless router as a receiving end and another wireless router as an access point. However, we may also acquire a wireless Ethernet card . We will go for one of the two options mentioned above to simulate the DakNet and e-Box connectivity procedure.

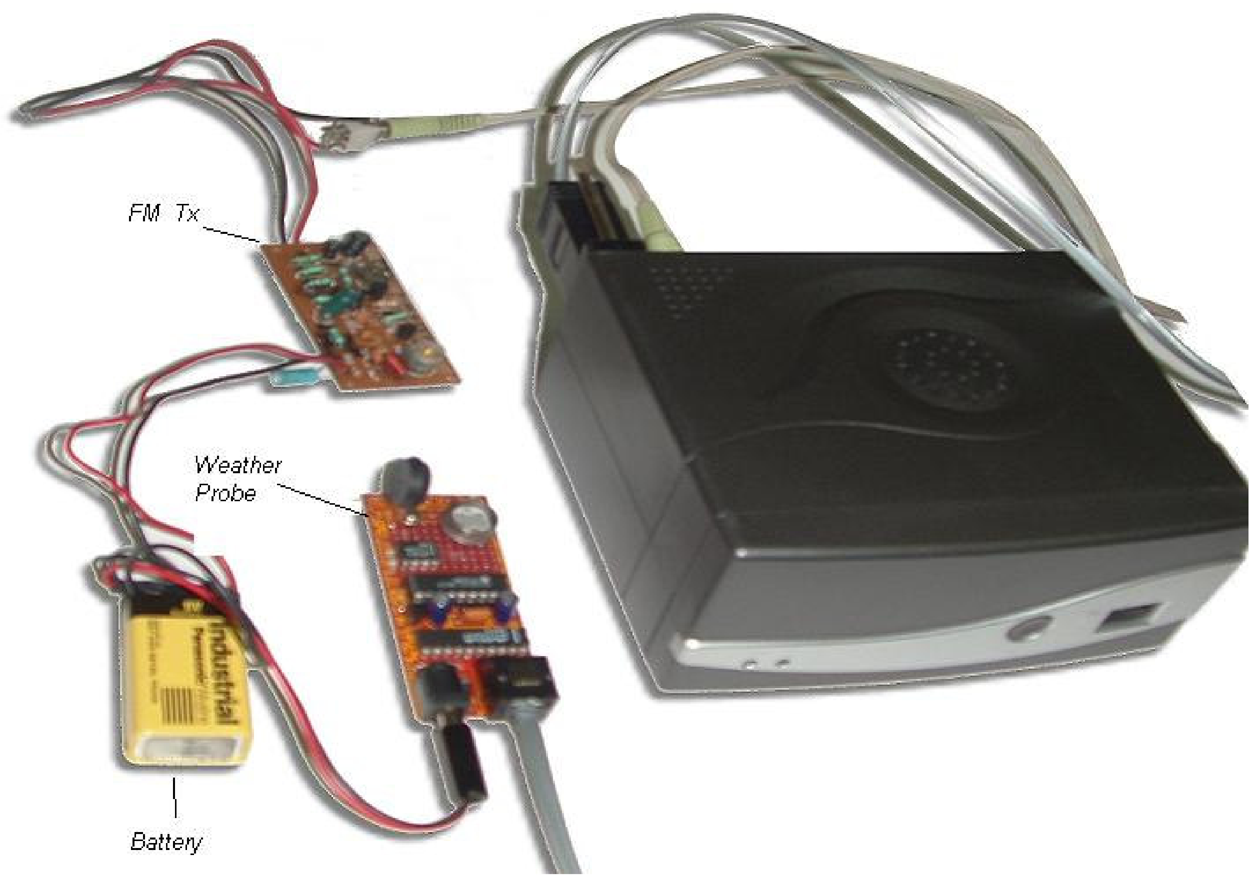


Fig. 3 : The Complete Unit while testing

1.Weather Probe is connected to the serial port of the e-Box(Connection is not visible in the figure)

3.12 Extensibility

Extensibility was one of our most important design goals and every component in our system has been designed keeping it in mind. The usage of web services allows us to retrieve and capture information from a variety of sources. Further, we hope to incorporate a facility to remotely configure the device to access, extract and broadcast information from new web services as and when they become available. Further,

- New local sensors can be added as and when technology makes it economically feasible to do so. As a result, we can foresee new sensors like, soil sensors which can collect and feed information to the AgriExpert expert system.

- New expert systems can be easily added or the existing ones can be easily improved. Further, since the back-end of the expert system stores conditional behavior in XML, it can be easily updated remotely. The expert system, then, is dynamically updatable.

- The systems functionality is also enhanced as and when newer web services pertinent to the issues taken become available.

4. Marketability

We expected this true, rural level system to be deployed through either Govt. Initiative or the system can also be deployed by a private enterprise and a 'targeted' advertisement model can be used for revenue generation. For example, a region dominated by Maize producers may get frequent advertisements for fertilizers relevant to the crop. Further given the limited technical support, the system is autonomous and without maintenance and hence there will be no middleman. Thus the product is a benefactor in making the target user independent and self confident.

However, the idea in its concept is generic and has its implications in various places and modes. For example, one fine application would be to automate Railway announcement systems. Since the information about Train delays and cancellations is almost always pulled from a computer network, we propose that a web service be made that will give the status corresponding to a particular railway station. An e-Box system that we have deployed on the railway station will automatically poll this web service for train information corresponding to this particular railway station. This information will then be converted to speech using a text to speech engine and broadcasted over the Station announcement system. Also, e-Box can also be used in institutions like schools, call centers, shopping malls for their own dedicated localized purports. Hence such mechanisms make our product an excellent product for manufacturing for commercial purposes. Huge revenues can be generated by marketing the product appropriately.

5. Costs Incurred:

Service Provider Costs:

S.No.	Device Name	Cost
1.	e-Box	$200
2.	FM Transmitter	$200commercial-to be used actually)
	Wireless Router	$20

Total $400

Consumer Cost FM receiver module $2

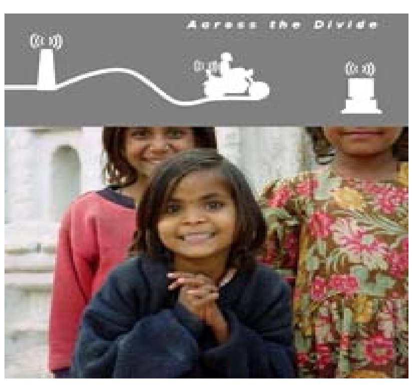
6. Summary

The moment we heard of the theme for the competition- Going beyond boundaries, we instanty recognized the need to develop a system that would make the life of the disabled, better and prosperous. Numerous ideas came related to removing disability. Then one fine day , after hearing a talk at a conference that addressed rural problems, we realized that lack of information is in a form, disability.

Initiatives like internet kiosks, Media Lab Asia's DakNet , postman carrying cell phone, video conferencing through holes in the walls of hut and numerous other efforts have been taken by the Govt. of India in the past. While so much has been achieved, there is still such a distance to go, globally in general and India in particular. And so we decided to take up an agenda that offers the poor rural guy greater opportunity and empowerment. An agenda that uses Information Technology tools to help people break out of the vicious cycle of illiteracy, an agenda for not only escaping illiteracy but for achieving prosperity, an agenda that is rooted in the vision and noble ideals of the past.

We have managed to implement a simple prototype that demonstrates the power of this idea. We have implemented one of the possible applications of this system in the form of a system that uses a Global weather web service and locally deployed weather sensor to feed this information into the expert system. The expert system in turn will come up with valuable deductions about the kind of inputs to be given to the crops. Finally this information is getting converted to audio and broadcast through the FM transmitter we are using. The Text to speech system still needs to be tweaked and we would be using commercial, off the shelf FM transmitter with better quality. Both these steps should be accomplished in the near future. It must be remembered that our sample application is intended to illustrate the proof of concept of this idea. Opportunities are there to implement several other useful information services for community users using this mode. We can envision applications that will broadcast information about on-going health initiatives in the region and others which will inform the producers the best price of their services i.e. - the best price to sell a Wheat crop. However, it is important for more and more regional information be available on the internet. We believe our application can be the driver for this to happen.

The inspiration for the idea came from the following couplet written by Tagore - ***"Soon the world will see an India where the mind is without fear and the head is held high and where the knowledge is free."***



References:

1. DakNet: www.medialabasia.org.
2. MSDN library.
3. www.webservicesx.com- Free Global Weather webservice.
4. Weather probe, based on circuit designed by Rick Fletcher, fletcher@media.mit.edu.
5. SVOX Switzerland – Brona Collins<collins@svox.com> and Juergen Keller < keller@svox.com>. Official Web site : www.svox.com.
6. Mike Hall : Technical Product Manager, Mobile and Embedded device group. mikehall@microsoft.com Milke Hall's Blog.: http://blogs.microsoft.com/mikehall .
7. Microsoft help : <helpp@microsoft.com>
8. Simon Haykins, *Communication Systems,* Wiley Eastern Inc.
9. John Kennedy, *Principles of Communication Systems*, Tata Mc Graw Hill.
10. Google Images.