

Key Lessons in Mobile Applications

David Dack, Colin I'Anson, Graeme Proudler Networks & Communications Laboratory HP Laboratories Bristol HPL-93-108 December, 1993

mobile data, cellular data, infra-red cellphone interface, locator This paper describes some of our investigations into mobile applications and proposes a standard interface to mobile phones for computing devices. This interface is based on experience from a number of experimental systems we have built, and outlines the consequent thinking that lead to our proposal for a simple computer interface for mobile communications. The interface is based on the 115kb/s infra-red interface that is being promoted by the Infra Red Data Association [1]. We describe the interface in two ways; one is based on a structural block model, the second uses the OSI model of communications.

Internal Accession Date Only

Presented at the IEEE COMSOC Conference, Tokyo, Japan, December 8-10, 1993 © Copyright Hewlett-Packard Company 1993

1. Introduction

It is not a simple task to find applications of mobile communications that will generate significant profits. The revenue aspects of the basic voice service are strongly driven by regulation and competition and will hinder the introduction of useful mobile data services. When cellular phones are sold at below their manufacturing cost there is a significant distortion of the market. With this impact from voice service, the potential income from innovative applications is reduced and they may never be developed. In addition to these financial issues in the market there are major difficulties in locating the mobile data applications that will be appreciated. Comments like "I leave the office to get away from my e-mail, I don't want it to follow me", are often heard. Others expect services on the road that they have not got in their offices and therefore applications are difficult to create since access to the sources of information are limited.

In the late 1980's the Locator system provided valuable insights to mobile applications, figure 1 [2,3]:

- Built on analogue cellular it used error correcting modems to support an X.400 service.
- The system demonstrated the worlds first X.413 message store, and X.400 secure messaging service.
- A full security profile was made available [4, 5].
- The system was evaluated during a 3 month trial with 15 mobiles passing many messages.
- The main user application was a database look-up which was complemented by a more traditional mail service.
- The mail service was also used to transport logs of the trial for analysis of the system performance.



Figure 1: Locator Architecture

Although the trial prototyped and tested the security features there were practical difficulties with other aspects of the trail. The reliability of data modems over cellular, despite the extensive error correction (CDLC) and the revised timer values, was poor and most users experienced multiple re-trys every day. Since the system was fully automated this did not produce a negative reaction from the users of the trial, indeed one user wanted the existing authentication system used to be replaced by the Locator mechanism since it was easier to use! Codes of practice to reduce the number of re-trys could be introduced but these would have defeated the key principle of the trial - to use non-productive time between customer calls to collect information for the next call. As a result of these practical difficulties the principle conclusion of the trial was that analogue cellular would not offer sufficient quality of service for pervasive applications and that we should wait for digital cellular data services.

More recent studies on the quality of service of voice band modems over analogue cellular have produced mixed results[6]. In many cases since the session is interactive the mobile data user is stationary. This reduces many effects that may degrade the service and a mobile user who cannot get a good voice service does not attempt to use data. This filters the results making the overall quality of service look better. Other applications have a higher order error control beyond that simply provided by the modems to improve the quality of service. Error correction in the fax service is a good example especially when line repeats and deletes are also used to make the received image look better to the recipient.

The promise of better quality of service with digital cellular data services has been slow to appear since the network operators must first get a voice service working to ensure a good revenue flow. As a provider of mobile appliances we would like to see these services introduced and have tried to promote the introduction of these services by writing simple applications including a diary reconciliation package to allow a manager and secretary to keep a common diary. One of the most difficult issues in performing a demonstration and one that has not been fully resolved is the interface to the mobile. Although we could make our own proposal there is no motivation for a cellular handset manufacturer to include it in their product. Therefore pre-competitive industrial collaboration is essential to agree an interface that can be implemented by all. Before describing our contribution in more detail we would like to consider other trials that have been performed in Hewlett Packard Laboratories, Bristol since they provide additional insight into our thinking.

One key data service is fax. The provision of this service in digital cellular systems is more complex than with analogue cellular systems with an inter working function required. The requirement for this extra function network will delay the introduction of any data services and leads to an extra source of service quality problems. In the Locator system there was also an inter working function between the modem used on the air interface and the modem used on the PSTN, which contributed strongly to the quality problems we experienced.

In GSM there are 2 forms of the fax service, the transparent [7] and the non-transparent [8] forms. In the transparent form there is a guaranteed latency but the error rate is not guaranteed. The non-transparent form guarantees the reliable transmission of the service but the latency will change. ETSI has commissioned work to investigate the performance of the two fax services - during these studies we performed our own evaluation of the transparent service. The resulting real-time simulator was demonstrated during Asia Telecom 93 at the HP executive forum (figure 2). This showed the service working in a satisfactory manner at

2.4kb/s, 4.8kb/s and 9.6kb/s at a variety of C/I ratios on the air interface down to 7dB. Therefore we were able to verify the GSM transparent fax service, notably the operation of the air interface protocol, terminal adapter, inter working function.



Figure 2: Real-Time GSM Fax Service Evaluation

The trial showed that digital cellular systems like GSM can provide quality data services. Therefore we expect that applications provided by the stream data services, fax services and the short message service will have good potential to provide a high quality of service. Indeed many simple applications, for example cc mail, can be readily implemented without the need to code a dedicated application. We wait with interest for experiments with a live short message service.

In addition to experiments with cellular systems we have also prototyped systems using packet networks. We have built an e-mail gateway to a Mobitex network and implemented a simple mail client on an HP95LX (Figure 3). Using the standard mail functions of the operating system of a workstation, the system was operational in a few months. Some unexpected reliability issues were encountered with the mobile bearer, notably the loss of packet ordering in the network. Since no sequencing guarantee is provided we were forced to implement a simple session layer to ensure a reliable service. With a gateway to a fixed e-mail network we were able to access a variety of standard resources, notably an HP OpenMail gateway. This permitted the mobile to send e-mail to a variety of recipients including X.400, Telex and fax.



Figure 3: Mobitex Experiment

These various trials have provided useful insights into the requirements of mobile data communications. Notably,

- 1. The batteries in the phone/modem and sometimes the computer run out of charge very quickly.
- 2. A simple interface to mobile communications is required.
- 3. The service on this interface must be simply represented to application programmers
- 4. The programming interface must be able to cope with a variety of transport mechanisms.
- 5. The mobile data service must be reliable.
- 6. A variety of data services must be supported including packet data, stream data and fax.
- 7. Connection between the mobile computer and the mobile phone must use a simple cable or Infra Red.
- 8. Since the equipment is intended to be portable, the resources (hardware and software) needed to implement the interface should be lightweight.

2. The Battery Problem

One day, we are sure, we shall all use a "Star Trek" communicator and we must hope that by then we will not have to carry a briefcase full of spare batteries to power the radio. However, we do not have to wait for such a highly integrated solution. Already we have very small powerful computers and cellular telephones but what is the best way of bringing them together? To answer this question we need to consider the technological characteristics of computers and phones, and also the human factors involved.

2.1 Personal Computer Trends

Palmtop computers exist which can run any of the programmes of their desk top equivalents even the number of characters on the screen is the same. Indeed, with "Lotus to go" in ROM one can worry about budgets while on holiday! These machines and the larger portable PCs provide insight into the requirements of a terminal for mobile data applications.

- From the human factors point of view, the screen size on a palm top is the minimum desirable size.
- For some simple applications a pen will replace the keyboard, but for compatibility with most current software and for clarity of entry a key board is still preferred by many. The palm top size is about as small as is practicable.
- The most significant feature of the palmtop computer is that two alkaline batteries will last for many weeks. There is no need to worry about recharging batteries after a few hours of use. Any additional feature which detracts from this enormous advantage will cause intense frustration.
- For the class of users for whom word processing is of primary importance, the palm top keyboard is not a good idea. Thus, as in the HP Omnibook Super Portable, the size of the computer is determined by the size of a normal QWERTY keyboard. Since now there is room for hard disks, bigger displays etc. then battery drain is increased but with careful design long battery life can be provided. For a truly portable PC this is a major differentiating feature.

2.2 Communications Devices

It takes little imagination to see the advantages of adding communications capabilities to the palm tops and super portables.

The most obvious first step is to incorporate a modem for use wherever a telephone jack is available. Even so, the most desirable option appears to be to use cellular or packet radio systems to ensure immediate access to the network. Today, all of these solutions are available with varying degrees of simplicity but all are power hungry.

Modems are heavy users of digital signal processing and even PCMCIA modem cards can consume up to 1 watt which will rapidly destroy the battery life advantage of any palm top or super portable. Cellular phones and packet radio systems suffer from a fundamental constraint in that they have to radiate a certain amount of RF. energy in order to achieve adequate range with a good quality of service. The worst offenders are analogue cellular phones which of course need modems as well in order to handle data.

It appears then that although computers are still on an improving performance to power consumption curve, we are approaching a lower limit to the power consumption of the phone, at least until pico cellular systems take over. So in the short to medium term, we have an interesting situation in which modem and radio manufacturers are falling over themselves to produce PCMCIA card modems and radios whose main function seems to be to destroy the primary advantage of the palm tops and super portables into which they are inserted.

2.3 Human Factors

Technology, to be successful, must be subservient to human needs. As we have seen the super portable with its full size keyboard is preferred for word processing to the palm top. Because of its size, it will not be carried everywhere in the way that a much smaller phone will. This is an important point. It is not essential to incorporate a phone into a super portable yet it is a very desirable feature to be able to pick up the appropriate device for the task in hand. This also leads to another important maxim:

The batteries should be associated with the functionality they are powering

The computer should have its own batteries which can be relied upon to last for weeks even when communicating. If the functionality of a cell phone is required then it should come with its own batteries. Until cellular technology is improved these will need to be charged overnight. Fortunately this paradigm is understood by the cellphone user.

How can the two different functionalities, the computer and the cellphone be connected? At first sight the simple answer appears to be a cable but this idea can be dismissed because of the inconvenience and because of the difficulty of establishing standard connectors pinouts and cables between different manufacturers. (A simple RS232 cable is rarely simple.) A cable would also lead radio manufacturers into the temptation of taking power from the computer. Fortunately, the industry is giving its backing to an interface which enables communication without cables. The Infra Red Data Association has voted the adoption of the Hewlett Packard Serial Infra Red Link, HP SIR, as an interconnect standard.

The adoption of this standard means that manufacturers of computers and communications devices can design and improve their products independently of each other but can inter work them simply by bringing them close together.

3. Introduction to SIR

Hewlett Packard has addressed the need for portable device inter working by creating low power optical transceivers for portable computing devices. This patented technology has been implemented as the Serial Infrared Communications Interface (SIR). In order to make SIR an industry standard, HP is working with Connexus to license the technology and enable the industry to design SIR into the broadest range of products [9].

Interoperability among devices from multiple vendors is very important. The objective of SIR is to create an open industry standard. Experience shows that when this is done, the market has expended rapidly and become larger than when proprietary or non standard implementations have been used.

The Serial Infrared Communications Interface addresses the critical needs of the mobile computing environment. By their very nature, portable devices are not continuously interconnected, but are designed to be connected only when the need exists. Requiring no cables to connect devices, SIR makes the electronic transfer of information very easy and effortless - 'unconscious portability'.

SIR is a wireless serial communications interface that adheres to a subset of the RS-232 interface functional characteristics. It currently operates at speeds up to 115,200 bits per second and is intended for 'point and shoot' operation; whereby the devices to communicate are positioned within line-of-sight and in relative close proximity. Implementation is very low cost, and the power requirements are very low, extending battery life. Operating distance is very dependent upon the design implementation, but reliable operation over more than a meter is possible.

3.1 Protocol

SIR employs the standard asynchronous serial protocol used in most personal computers. It consists of three signal connections: receive data, transmit data, and signal ground where the data stream consists of 5, 6, 7, or 8 bit words with optional parity. Each data word is preceded by a start bit and followed by at least one stop bit. When the transmitter is idle, the transmit line is held at a marking condition.

The encoding format is structured to require a very low average power per bit transmitted (figure 4):

- A zero (0) is signalled by sending a single pulse at the beginning of the serial bit time.
- A one (1) is signalled by sending nothing.
- The width of each pulse is 3/16th of a single bit time.
- The serial bit time at 115k bits per second is 8.6µsec and the pulse is 1.5µsec, while at 2400 bits per second they are 416µsec and 78µsec respectively.



Figure 4: SIR Frames

3.2 Hardware

The hardware implementation is low cost and can be incorporated into most devices with a minimal amount of effort. Indeed devices that already have an RS-232 or other serial interface may be able to share some components with the SIR interface.

A 16550 compatible UART is used just as it would be in an RS-232 port (figure 5). The FIFO in the 16550 will help with interrupt latency caused by operating systems such as Microsoft Windows, allowing 115k bits per second to be achieved. A 16450 or 8250 UART can also be used..



Figure 5: HP SIR Hardware

The IR Transmitter encodes the bit stream to transmit a pulse for every zero from the UART The IR LED has a 950nm wavelength and is chosen to avoid interference from the visible light spectrum. To achieve 115kb/s, the switching speed of the transmitting LED must be less than 500ns. In addition, multiple LEDs can be used to both widen the signal cone and increase the distance between communicating devices.

The IR pulses are detected by an IR photo diode and transformed back to CMOS levels by the IR Receiver. The zero pulses are stretched by the IR Format Decoder to get back to a serial data stream that can be processed by the UART. Given this structure of transmitter and receiver it is obvious that SIR can only operate in a half-duplex mode since all transmissions are received by the photo diode (due to reflection).

Because of local reflections the IR receiver will always 'see' the transmitter output. Thus half duplex operation is essential.

The sensitivity of the receiver also means that error free reception cannot be guaranteed but half duplex protocols such as certain versions of kermit and xmodem have been sucessfully used for error correction.

3.3 Software

Software control of the SIR port can be accomplished through BIOS interrupts routines. Control can also be accomplished by writing directly to the hardware registers that control the communication parameters such as baud rate, parity, and number of data bits.

3.4 Commitment to a Standard

Hewlett Packard is committed to SIR becoming an open industry standard. The current embodiment of SIR does not address all of the requirements that every vendor may desire. It is intended to be a viable starting point for evolution of the open standard. Due to the near term needs of many vendors, every effort will be made to evolve SIR as rapidly as possible. Meeting marketplace needs is of paramount importance. Connexus, a consulting firm, is co-ordinating this standards process for SIR [9].

4. Structural Block Model for a Standard Interface to Mobile Communications

An interface in a communication system is just a link between processes. It must allow the processes on one side of the interface to co-operate with processes on the other side of the interface in order to achieve a communications objective. This interface must therefore support transport of the "user data" and any additional data which is used to co-ordinate those processes.

This section presents an argument that the interface should communicate blocks of data, and that there are four basic *types* of block. (Note that *type* is used here in the same sense as in computer programs, in that it defines the operations that may be performed on or with the data.)

With such a simple model, the elements of a standard interface may be developed and selected as required for particular applications. Observe that:

- the interface is driven by processes which have finite resources. Hence there are limits on the quantity of data which can be accepted from the interface, and even a stream of data would have to be halted on occasion. This implies that blocks of data are the natural structure for transmission over the interface.
- various types of data are passed over the interface. Some data (user-data) must be transferred to a remote system and may be altered by processes in the communications system during transfer but must be restored to the original form at the destination. Other data (control-data) will be used to change the state of processes in the communications system. Control-data can be further subdivided into communications-medium-control-data (that is independent of the type of information) and service-specific-control-data (that varies with the type of information). For example:
 - Communications-medium-control-data might be the destination of the communication path, a telephone number.
 - In the case of a fax service, service-specific-control-data will be associated with certain service specific control parameters that are different to those required by e-mail.



1



Figure 6: Block Data Types at the Interface

It follows that an interface that defines the link between adjacent processes should deal with various types of data block (figure 6). These can be described by four simple types that were introduced above:

- User_data. User-data blocks must be passed over the data link. Although the user-data may be altered by the processes within the communication system it must be restored to the original form at the corresponding interface in the destination system.
- Flow_control. This control is used to regulate the flow of data over the interface. They consist of a request_to_send signal and a ready indication. Observe that this set can potentially overcome latency in the link, by defining each ready indication to mean that space exists for a single packet, and therefore by sending multiple ready indications in response to a single request_to_send signal a better throughput can be achieved. These blocks require passing over the interface, and are interpreted by the processes controlling the interface.
- Communications_medium_control. This is used to control the state of the processes that transport the user-data and is used to configure the physical and logical parameters of the communication network.
- Service_specific_control. These controls are used to modify the state of processes that transport the user-data and will directly affect the processing of the user-data.

There are several ways of delimiting and labelling blocks of data for passing between processes. But most result in data structures which are not easily human-readable and are consequently less user-friendly. It is therefore suggested that the blocks of data that are presented at the interface are prefixed with a few alphanumeric characters that identify the type of block. This leads to a format which is similar to the popular Hayes "AT" command set¹.

The proposed interface therefore passes blocks of data which are prefixed by alphanumeric characters, and (to an extent) directly readable by humans.

Hayes is a registered trademark of Hayes Microcomputer Products Inc

- User-data, request_to_send, and ready blocks which are always present whatever the communication system.
- Communications_medium_control blocks which are always the same for a given communication system, irrespective of the sort of data being communicated.
- Service_specific_control blocks which will vary according to the sort of data being communicated. The number and types of controls that an application must provide will depend on the nature of the processes within the communication system.

It follows that a 100% resource-efficient yet completely standard interface for all applications for all communication systems is unlikely, since some elements of the interface must be specific to certain communication systems and to certain information types. This indicates a tool-kit approach to building an interface:

- User_data, request_to_send, ready signals and indicators will always be required.
- The communications_medium_control block, signals and indicators will be specific to a certain communication system and will always be required in that communication system.
- Service_specific_control signals and indicators will be specific to a particular sort of information, and will always be required when that sort of information is sent.

Each new instance of particular computing platform and radio system could inherit existing communications_medium_control signals and indicators, and may require development of new communications_medium_control signals and indicators. Each new information type could inherit existing service_specific_control signals and indicators, and may require development of new service_specific_control signals and indicators.

It may be that the complete interfaces for the most common systems and information types will be sufficiently similar to enable a "standard" interface of sufficient resource-efficiency. Even if this is not the case, the tool-kit approach nevertheless encourages the most uniform interface for a wide range of platforms and information types, and hence encourages the development of applications for those platforms.

5. Modelling of a Standard Interface to Mobile Communications

The interface introduced in the previous section can be described with a layered model. Although we cannot fit them to the 7 layers of the OSI model, the generic layer description can be profitably used [10, 11]. This leads to the concept of an abstract service, a protocol and, if required, an API.

The abstract service is a key element in ensuring that the processes in the various implementations carry out the same task. The service relies on the concepts of end-to-end user data, end-to-end control information and local control information. These are modelled as the (N)-service-data-unit, (N)-protocol-control-information and the (N)-interface-control-information respectively (figure 7). Indeed in a layered model the

(N)-protocol-data-unit itself becomes the (N-1)-service-data-unit. This helps model the increasing levels of encapsulation experienced by the data as the various levels of control are added. In the case of the interface we propose only one level of encapsulation is currently required but in the future the inclusion of new features like security may require further further functional layering.



Figure 7: OSI Layer Interface Data Units

In terms of the more abstract OSI model, the elements of the structural block model can be described as below:

- The (N)-service-data-unit or (N+1)-protocol-data-unit models the user-data blocks in a conventional manner.
- The (N)-protocol-control-information provides the mechanism that the layer entities use to communicate with the peer entity in the destination. It is therefore hidden from the interface and has no corresponding data block from the structural block model.
- The (N)-interface-control-information models the communication_medium_control, flow_control and service_specific_control signals.

In addition to the use of the general OSI model the service can also be described using service primitives with ladder diagrams. For example, figure 8a shows the simple attention command (AT) that results in a positive response (OK). Figure 8b shows a more complex interaction where a stored phone number is selected and dialled.



Figure 8: Simple Examples of Ladder Diagrams for Service Primitives

The protocol needs to be very simple and should be based on commands that are well accepted in the industry. As shown earlier we propose to use the well known modem control commands that are commonly known as the Hayes AT command set as a starting point. These commands can be displayed on a conventional terminal and once the cryptic form is mastered are quite obvious. The choice of this protocol on the Infra Red interface is obvious since it ensures the maximum backwards compatibility with existing systems. The addition of an abstract service means that some of the more informal, intuitive aspects of the interface can be recognised. This paves the way for a variety of API implementations.

The API is the interesting and also contentious area since a variety of operating systems will be used. Each implementor will have their own ideas but it is hoped that only a limited number of interfaces will used in practice. OSI cannot provide clear guidance in this area since the major contribution of the modelling is in the service description and the protocol. The protocol ensures that the messages that are transferred over the infra-red link are understood by every implementation and the service ensures that the operation of the messages over the link provides the desired service. Within a phone or computer the implementation of the service as a set of procedure calls, objects, a library or whatever will be dictated by the local environment. Certainly the operating system in the phone is unlikely to be a DOS environment! This diversity leads us to conclude that a toolset approach to the API is desirable, hopefully restricting the system dependent elements in each implementation.

6. Conclusion

This paper has outlined the need for an open interface to mobile communications. A number of proposals have been made on the requirements and on the modelling of the interface. It is hoped that this paper will stimulate further discussion that can lead to an industry wide agreed interface based on a wire free Infra Red interface.

Acknowledgements

.

.

The authors would like to thank their many colleagues in the Networks and Communications Laboratory in Hewlett Packard Laboratories, Bristol for allowing us to describe and draw conclusions from their work - Mark Eve, Roberto Morassi, Adrian Pell, Duncan Smith, Nick Wainwright, John Waters and Ascan Woermann.

-

References

[1] For further information on IRDA contact:

PCCA-IRDA PO Box 495 Brookdale CA 95007-0495

- [2] R Cole, et al., Mobile Access to Secure Interpersonal Messaging, On-Line International Open Systems Conference, April 1988
- [3] R Cole, C I'Anson, and A Pell, Architectural aspects of a Mobile X.400 Service, IFIP TC6/WG 6.5 Conference on Message Handling Systems and Distributed Applications, Costa Mesa, California, October 1988. (Proceedings published by North-Holland)
- [4] C Mitchell, D Rush, and M Walker, A Secure Messaging Architecture Implementing the X.400-1988 Security Features, The Computer Journal, Vol 33, No. 4, 1990.
- [5] Chris Mitchell, D Rush, and M Walker, LThe Security Architecture of the Secure Mobile Messaging Demonstrator, UK IT88 Conference Publication, University College Swansea, 4-7 July; DTI, 1988
- [6] Dr. J. Dasilva; DGXII/B-RACE Programme; commission of the European Communities.
- [7] Draft pr ETS 300 538; European Digital Cellular Telecommunications System (phase 2); Technical Realization of Facsi