SRI VENKATESHWARAA COLLEGE OF ENGINEERING & TECHNOLOGY
(Approved by AICTE, New Delhi & Affiliated to Pondicherry University, Puducherry.)
13-A, Villupuram – Pondy Main road, Ariyur, Puducherry – 605 102.
Phone: 0413-2644426, Fax: 2644424 / Website: www.svcetpondy.com

Department of Computer Science and Engineering

Subject Name: **MOBILE COMPUTING**

Subject Code: **CS E84**

UNIT II

Emerging Wireless Network Standards: 3 G Wireless Networks – State of Industry – Mobility support Software – End User Client Application – Mobility Middleware –Middleware for Application Development - Adaptation and Agents - Service Discovery Middleware - Finding Needed Services - Interoperability and Standardization.

2 Marks

1. **Write some of the examples for emerging wireless networks.** Wearable Computing (MIT), Wearable Computer Systems (CMU), IBM Wearable PC BodyLAN: A Wearable RF Communications System

2. **Explain End-user client**

A flurry of activity appeared in the trade press in late 1995 describing the rush by vendors, both large and small, to market mobile client software packages. Some of those products are discussed in this section. Recent literature search suggests that many of these products never materialized, were re-targeted to wired networks, or in some cases, are still struggling with weak sales. However, there are some big players with deep enough pockets to continue to pursue this marketplace. The discussions here are restricted to those products and services that still appear to have a current or promised market presence.

3. What are the two key players in mobility middleware?

Two key players in the wired-network middleware market that provide sup-port for distributed are Novell's Netware and Microsoft's Remote Access. Neither of these products will be discussed further since neither has yet announced plans (that we have seen) for moving into the wireless middleware domain.

4. What are the limitations of mobility middleware?

Mobility-support software and products that are commercially available today leaves much to be desired in terms of functionality, performance, portability, and interoperability.

Translucent Overlays
Multi-Database Access
Multi-Database Access
Workflows
Location Dependent Services

5. What is meant by SDMA?

SDMA is a technology which enhances the quality and coverage of wireless communication systems. It uses a technique wherein the subscriber's access is via a narrow focused radio beam and the location of the subscriber is tracked adaptively by an intelligent antenna array system.

6. Define 3G network.

3G is the third generation of wireless technologies. It comes with enhancements over previous wireless technologies, like high-speed transmission, advanced multimedia access and global roaming. 3G is mostly used with mobile phones and handsets as a means to connect the phone to the Internet or other IP networks in order to make voice and video calls, to download and upload data and to surf the net.

7. How is 3G better?

3G has the following enhancements over 2.5G and previous networks:

- Several times higher data speed;
- Enhanced audio and video streaming;
- Video-conferencing support;
- Web and WAP browsing at higher speeds;
- IPTV (TV through the Internet) support.

8. Explain state of industry.

The suite of products that provide some support for mobile computing spans the technology space from end-user client applications, such as spreadsheets, Web browsers, through middleware, down to products implemented in hardware that provides cellular or other radio transmission-based communications services.

9. What is interoperability?

Interoperability is the ability of diverse systems and organizations to work together (inter-operate). The term is often used in a technical systems engineering sense, or alternatively in a broad sense, taking into account social, political, and organizational factors that impact system to system performance.

10. What is standardization?

Standardization or **standardization** is the process of developing and implementing standards. The goals of right standardization can be to help with independence of single suppliers (commoditization), compatibility, interoperability, safety, repeatability,
or quality.

11. What are the types of interoperability?

1) Syntactic interoperability
2) Semantic interoperability

12. What is the basic strategy of mobile ware office server?

The basic strategy that underlies Mobile Ware is to minimize mobile platform connect time by executing data transfers in a burst mode. The intent of this software is to make the mobile platform appear to the user as though it were actually a node connected into the wired network. The initial customer target focused on large sales staffs that were primarily mobile and who needed access on demand to sales support information that was too bulky and or volatile to carry on extended trips.

13. What is meant by WAP?

The wireless application protocol (WAP) standard currently being developed by the WAP forum group offers an OSI-like protocol stack for interoperability of different wireless

networks. The WAP stack allows applications to register interest in quality of service events and thresholds (QoS). This, in turn, allows the application to be mobility-aware and adaptable to changes in the environment.

14. What is Shiva PPP (UQ: April'13)

Shiva's remote access client (known as PPP for Point-to-Point Protocol) enables mobile users to access servers embedded in either wireline or mobile servers almost seamlessly. For example, a client application that uses transaction processing services from BEA's Tuxedo can now access those services from a mobile platform using PPP. This software suite provides some limited security features such as limiting the number of login tries, or disconnecting a session and call- ing the user back at a pre-established number. However, it does not provide the rich collection of services available from Mobile Ware's Intelligent Transport Engine.

15. What is mobile ware office server?

Mobile Ware Office Server is an agent-based middleware for wireless or wire line access to application data. A service supported by Mobile Ware Office Server includes Lotus Notes, Web browsing, e-mail, and file transfer. A core component of the Mobile Ware Office Server is the Intelligent Transport Engine.

16. Explain Sybase SQL remote.

Unlike the Oracle Replication Manager, the Sybase product called SQL Remote has adopted a centralized model for managing replication. This product is a member of the Sybase SQL Any Where suite of tools (formerly called Watcom SQL).

17. What is Oracle replication manager?

Oracle has announced a version of its Replication Manager which will eventually support bi-directional replication among a collection of distributed and centralized server databases. The Oracle approach is based on a peer-to-peer model, much like Lotus Notes, in which a collection of distributed processes manage replication collectively.

18. What is Oracle lite?

This product is a cut-down version of the Oracle server that can run in a small portable system (or a desktop workstation). It can be used as a companion technology for the Oracle Agent Software to store local copies of subsets of corporate databases and can accumulate updates to the data that are generated locally at the mobile client.

19. What is UMTS (UQ: April'13)

UMTS, which stands for Universal Mobile Telecommunications System, is currently a project under the SMG (Special Mobile Group), a committee in ETSI. Decisions made in early 1998 by ETSI has given Europe a clear direction to- wards the realization of its third generation wireless communication system. ETSI agreed in January 1998 on two different UTRA methods: W-CDMA in the paired portion of the radio spectrum, and TD-CDMA in the unpaired portion.

11 Marks

1. **Write in detail about emerging wireless network standards (11)**

ITU (International Telecommunication Union) is a United Nation affiliated organization that oversees global telecommunication systems and standards. ETSI (European Telecommunications Standards Institute) is Europe's premier telecom standards organization well known for its development of the GSM standards. Both organizations are currently leading efforts to promote cooperation in the definition and development of future wireless networks. One goal common to both organizations is achieving seamless communication for the global consumer through cooperation on technical developments. Decisions made within these two organizations will have a dramatic effect on the future directions of wireless networks and services.

IMT-2000

The ITU (International Telecommunication Union), headquartered in Geneva, Switzerland, is an international organization within which governments and the private sector coordinate global telecom networks and services. IMT 2000 (International Mobile Telecommunications by the year 2000 project) is a project under the ITU that plans to facilitate cooperation in deciding global wireless access for the 21st century. Until recently, IMT 2000 was known as FPLMTS or Future Public Land Mobile Telephony System.

IMT 2000's vision is to "provide direction to the many related technological developments in the wireless industry to assist the convergence of these essentially competing wireless access technologies." IMT 2000 is expected to unify many different wireless systems, leading to the global offering of a wide range of portable services. It is expected that the IMT 2000 project will enable the merging of wireless services and Internet services, leading to the creation of a mobile multimedia technology and new modes of communication.

IMT 2000's vision of future wireless tele services is specified in terms of information rate, user delay sensitivity, and bit error rate requirements. Eight classes of services are specified as listed in Table 5.2.

Initial assignments of the IMT-2000 spectrum for Europe, US and Japan are shown in Figure 5.2. It shows how the dream of global roaming might be achieved in the future.

UMTS

UMTS, which stands for Universal Mobile Telecommunications System, is currently a project under the SMG (Special Mobile Group), a committee in ETSI.

Decisions made in early 1998 by ETSI has given Europe a clear direction to- wards the realization of its third generation wireless communication system. ETSI agreed in January 1998 on two different UTRA methods: W-CDMA in the paired portion of the radio spectrum, and TD-CDMA in the unpaired portion.

The main goals of the UMTS system can be summarized as follows:

Table 5.2 Third-generation requirements for wireless teleservices

Service Classification	Teleservices	Information Rate (kbps)	BER	Delay Sensitivity (ms)
Voice	Speech	8–32	10^{-3}	40
	Emergency Call	8–32	10^{-3}	40
	Teleconference	32–128	10^{-3}	40
Voice Band Audio	Facsimile	32–64	10^{-6}	100
	Telefax	64	10^{-6}	100
	Modems	32–64	10^{-6}	200
	Data Terminals	2.4–64	10^{-6}	200
Sound	Program Sound	128	10^{-6}	200
	High Quality Audio	940	10^{-5}	200
Video	Conferencing	384–768	10^{-7}	90
	Surveillance	64–768	10^{-7}	90
	Telephony	64–384	10^{-7}	40–90
Messaging	SMS & Paging	1.2–9.6	10^{-6}	100
	Voice Mail	8–32	10^{-4}	90
	Facsimile Mail	32–64	10^{-6}	90
	Video Mail	64	10^{-7}	90
	E-Mail	1.2–64	10^{-6}	100
Broadcast	Message	1.2–9.6	10^{-6}	100
	Multicast	1.2–9.6	10^{-6}	100
	SMS Cell	1.2–9.6	10^{-6}	100
	Public/Emergency Announcement (Voice)	8–32	10^{-4}	90
	Public/Emergency Announcement (Data)	1.2–9.6	10^{-6}	100
Data	Database Access	2.4–768	10^{-6}	200+
	Teleshopping	2.4–768	10^{-7}	90
	Newspapers	2.4–2000	10^{-6}	200
	GPS	64	10^{-6}	100
Teleaction	Remote Control	1.2–9.6	10^{-6}	100
	Remote Terminal	1.2–64	10^{-6}	100
	User Profile Editing	1.2–9.6	10^{-6}	200

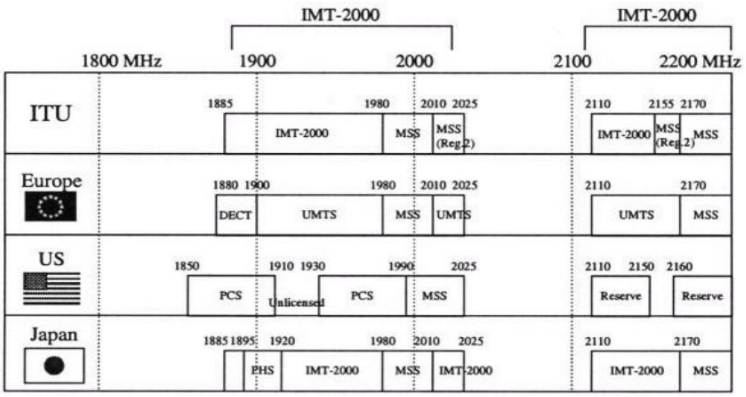


Figure 5.2 IMT-2000 spectrum assignment for Europe, US, and Japan

The accommodation of high speed, multimedia interfaces to support Internet applications at speeds of up to 2 Mbps, through a quantum leap in technology

- At least a 3-fold increase in spectral efficiency
- Support from an evolved GSM core network
- Compliance in meeting or exceeding ITU's Family Concept IMT 2000 System

The UMTS project schedule and milestones are shown in Figure 5.3. The first UMTS deployment is shown to be planned for the year 2002.

ACTS

Another organization that is greatly influencing the direction of wireless communications, particularly W-ATM, are the projects funded out of ACTS (the Advanced Communications Technologies and Services). ACTS is a group of European research projects with budget 50% funded by the European Economic Commission (EEC). The remaining 50% of the research funding is provided by those industry organizations involved in the research. ACTS broad objective is to develop advanced communications systems and services for economic

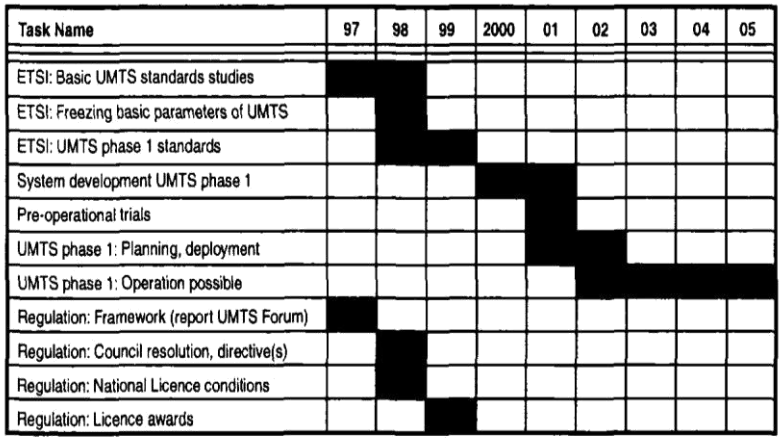


Figure 5.3 UMTS schedule and milestones

development and social cohesion in Europe. Research projects by ACTS include: multimedia, photonics, high speed networking, and mobile and portable communications.

2. Explain in detail about the Third generation wireless networks (11) (UQ April'13) APRIL/MAY2014

IMT 2000's original vision for third generation wireless networks was to create a single global communication system common to all countries and regions. This vision was too revolutionary to second generation wireless network providers who have invested heavily in current technology. To protect their investments, carriers requested ITU to consider a more evolutionary approach to third generation network standards. ITU, in turn, modified its vision into creating a "family of systems" that would converge and comply with a common set of requirements for third generation networks.

Following IMT 2000's vision, current research, development, and global standardization efforts are focused on upgrading second generation systems including GSM, CDMA, and TDMA. A major goal of this conversion is to upgrade these system evolutionary over time while maintaining the operation and profitability of the existing second generation network infrastructure.

TD-CDMA (Time Division, Code Division Multiple Access) and W-CDMA (Wideband Code Division Multiple Access) are the two major evolutionary network schemes currently under consideration by ITU. SDMA (Space Division Multiple Access) is also receiving attention as a network scheme with similar evolutionary nature.

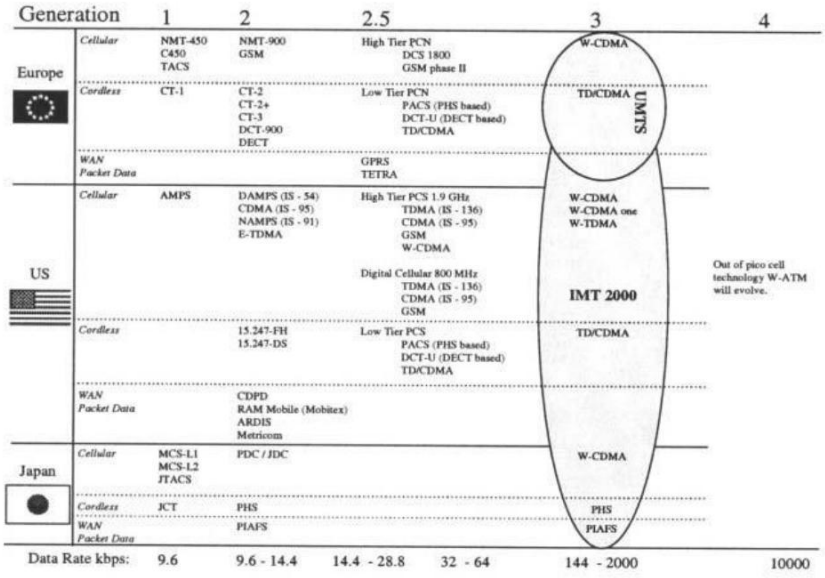


Figure 5.4 Evolution of wireless network technologies in Europe, US and Japan

Before summarizing the details of TD-CDMA, W-CDMA, and SDMA, we first describe the evolution of the wireless network technology in Japan, Europe, and the USA. Figure 5.4 helps clarify the alphabet soup in which the wireless industry swims. The figure defines the technology and generation in which a particular wireless system operates (or once operated). The data rates at the bottom of the chart apply only to the cellular technologies, not the cordless or WAN packet data rates. It is also important to note that at the time of publication, the third generation wireless technologies for the US had not been selected. Europe (presented in ETSI) just selected a combination of W-CDMA and TD-CDMA. Japan had chosen W-CDMA for their third generation wireless technology.

Time Division/Code Division Multiple Access

TD-CDMA is a proposed radio interface standard that uses CDMA signal spreading techniques to enhance the capacity offered by conventional TDMA system. Digitized voice and data would be transmitted on a 1.6 MHz wide channel using time-segmented TDMA technology. Each time slot of the TDMA channel would be individually coded using CDMA technology, thus supporting multiple users per time slot.

One design goal of TD-CDMA is to allow the CDMA technology to be smoothly integrated into the existing, second generation GSM TDMA structure world- wide. This will allow GSM operators to compete for wideband multimedia services while protecting their current and future investments. An important feature of TD-CDMA is its ability to adjust the ratio of spectrum allocated for the uplink and the downlink. The air interface can therefore be tuned to enhance the performance of certain applications such as Internet access and voice applications.

TD-CDMA uses the same frame structure as GSM 5.5. It has eight time slots with a burst duration of 577 micro seconds and a frame length of 4.616 m/sec as shown in Table 5.3. The TD-CDMA carrier bandwidth is eight times that of the 200 kHz GSM carrier equaling 1.6 MHz The compatibility between the TD-CDMA and GSM time bursts and frame structure permits the evolutionary step to third-generation systems. As many as 8 simultaneous CDMA codes are allowed in one time slot in TD- CDMA. This permits 8 users per time slot, or a larger combination of voice and data users to communicate without interference. For example, TD-

5 data users and still maintains the appropriate BER

[83] (10^{-3} for voice and 10^{-6} for data). Eight users per time slot appears to be selected because it offers a happy medium between the number of voice and data calls that the system can accommodate.

Because TD-CDMA has the ability to assign multiple codes to one user, it permits broadband (or bandwidth on demand) transmission capabilities. Assuming a bandwidth of 1.6 MHz, a time slot with an information rate of 16 kbps using QPSK data modulation, eight possible users per time slot (eight CDMA codes per time slot) gives you an information rate of 128 kbps. If all eight time slots were allocated to a single subscriber in a pico cell environment or where mobility is restricted, 1024 kbps can be achieved. Changing to a different data

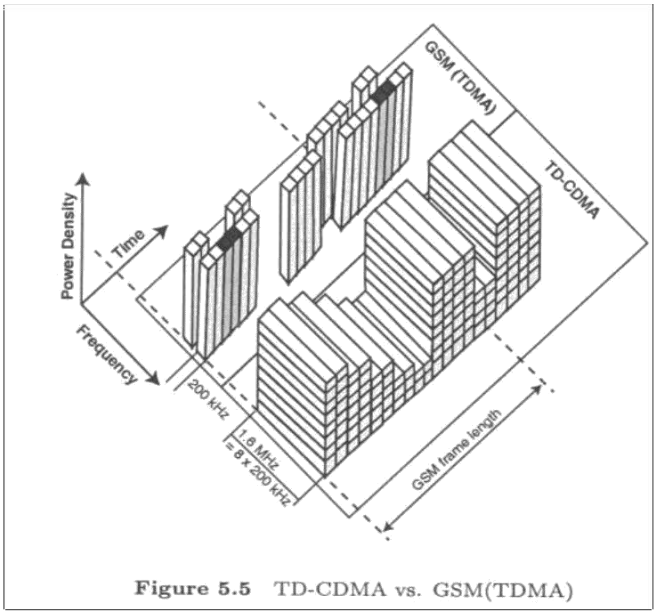


Figure 5.5 TD-CDMA vs. GSM(TDMA)

Table 5.3 TD-CDMA systems parameters

Bandwidth	1.6 or 3.2 MHz
Frequency band	2 GHz
Time slots per frame	8
Burst duration	577 micro sec
Frame length	4.616 msec
Full rate speech channels per carrier	64
Code slots per time slot	8
Chip rate	2.167 Mcps (million chips per sec)
Data modulation QPSK	(quadrature phase shift keying) or 16QAM (16 points quadrature amplitude modulation)
Spreading modulation	Linearized GMSK (gaussian minimum shift keying)

Modulation scheme like 16QAM instead of QPSK, an information rate of 2048 kbps is conceivable. Requirements for third-generation cellular systems are met by the TD-CDMA system.

Another advantage of TD-CDMA is the fact that intra-cell interference is orthogonal by time. This enables multiple subscriber signals to be received at differing power levels thereby eliminating the near-far effect and the need for a soft hand-off. The hand-off is conducted through a separate TD-CDMA or GSM carrier simplifying dual mode, dual band handsets. This

is a divergence from GSM that conducts the voice and control channels on the same 200 kHz radio band. The sources studied on the TD-CDMA were not clear if mo- bile assisted hand-offs (MAHO), where the subscriber unit returns radio signal strength information back to the base station, is a feature in the TD-CDMA system.

The strategic importance of TD-CDMA can be summarized as follows:

1. The networks that TD-CDMA is catered towards are significantly deployed infrastructures, including:

- GSM: deployed in 74 countries, 200+ networks, and 20+ million subscribers and
- AMPS: deployed in 110 countries, 40+ million AMPS subscribers, 1.5+ million D-AMPS subscribers.

2. The cost of changing the air interface in a cellular system is significant.

System design and setup with regards to the MSC (mobile switching center), the BSC (base station controller), the cell location, and frequency reuse are all based upon the characteristics of the access scheme. Selecting a revolutionary different access scheme is therefore more than just changing the air interface; it is a costly operation. The TD-CDMA system is de- signed to be an evolutionary –not a revolutionary– step from GSM second generation infrastructure to third generation infrastructure.

The benefits to taking a revolutionary step include the absence of a legacy system and a quantum leap in abilities. The risk, however, is shortening the return on investment on second generation infrastructure. But regardless of the philosophical underpinnings, keeping deployed infrastructure profitable is a concept well-embedded in the telecommunication industry.

3. TD-CDMA promises to be future proof:

- Spectral efficiency twice that of GSM
- Reuse of existing GSM network structure and principles: cell sites, planning, hierarchical cell structures
- Efficient interworking with GSM
- Inherent TDD (time division duplex) support for cordless operation
- Data rate up to 2 Mbps indoor, 1 Mbit in all environments
- No soft hand-off and fast power control

TD-CDMA has recently been agreed upon by ETSI as a third-generation solution for GSM service providers. Support from most major telecommunications equipment providers in Europe has played a role in ETSI's decision to adapt TD-CDMA.

Wideband Code Division Multiple Access

W-CDMA is a spread spectrum technology in which the entire band- width is shared by multiple subscribers for transmission. A subscriber's data is modulated with PN codes; the

signal is then spread and transmitted across a wideband. The receiver is responsible for dispreading the desired signal from the wideband transmission and contending with interference. The dispreading process at the receiver shrinks the spread signal back down to the original signal and at the same time decreases the power spectral density of the interference . This inherent ability to manage interference is at the heart of W-CDMA.

In W-CDMA, there are four control channels: the pilot, synchronization, paging and access channels. The channels are identified in the transmission by using a specific PN code, either a Walsh or Hadamard function (see [48] for further explanation). In the 5 MHz W-CDMA forward link, there are two designated codes for possible assignment to two possible pilot channels, two codes for two possible synchronization channels, a maximum of seven inclusive sequential codes for paging channels, and the remainder codes are not assigned and are used for forward traffic channels (see Table 5.4). The traffic channels are assigned *n* channel code numbers based upon desired the data rate, *n* = 0 … 64 for 64 kbps, *n* = 0…127 for 32 kbps, and *n* = 0… 255 for 16 kbps. The re- verse link channels are the access and the reverse traffic channels. Codes remain unassigned on the reverse channels so that channel assignment can be done dynamically and in response to paging channels and to interference. The forward link and the reverse link are FDD (frequency division duplexed). Frequency separation depends on the countries frequency allocation scheme. The channels in both the forward and reverse links are frequency division multiplexed.

Table 5.4 W-CDMA channel usage [48]

Bandwidth (MHz)	Pilot Channel Number	Sync Channel Number	Paging Channel Number
1.25	0	32	1-7 (sequential)
5.0	0 and 64	32 and/or 96	1-7 (sequential)
10.0	0 and 128	64 and/or 192	1-7 (sequential)
15.0	0	384	1-7 (sequential)

W-CDMA's channel responsibilities can be described as follows:

- Pilot channel (Forward link)

The BTS (base transceiver station) transmits one or two pilot channels carrying a reference clock necessary for demodulation and the hand-off process. The pilot channel also carries information used in estimating BTS signal strength therein indicating the best communication link for the subscriber terminal. After deciding on the best pilot signal, the subscriber terminal demodulates the synchronization channel.

- Synchronization channel (Forward link)

The synchronization channel contains system parameters, offset time, access parameters, channel lists and neighboring radio channel lists, all necessary in synchronizing with the paging, access and voice channels. The synchronization channel always operates at 1200 bps.

- Paging channel (Forward link)

System parameters and paging information to groups or a single subscriber are continually sent on the paging channel. Pages are combined into groups permitting a sleep mode to be built into the subscriber terminal, extending battery life. Subscriber terminals can monitor

multiple paging channels. Thus, when another cell's paging channel has a better signal, a hand-off is requested. The paging channel has a data rate of 9600 bps or 4800 bps.

- Access channel (Reverse link)

When a page is detected the terminal attempts to access the system through the access channel. The terminal increases signal strength sent to the BTS until the system responds, a random time limit has expired, or maximum power levels have been exceeded.

- Traffic channel (Both forward and reverse links)

Within the traffic channel, there are two types of in-band signaling used. Blank-and-burst in-band signaling where an entire 20 m/sec frame is re- placed with control information. Dim-and-burst is also in-band signaling but the control information is distributed throughout a variable number of 20 m/sec frames.

The forward and reverse channels are modulated differently, QPSK for the forward channel, and O-QPSK for the reverse channel. There are five steps to the modulation process:

1. A PN multiplier multiplies the user data by the Walsh or Hadamard function, uniquely identifying that information to a specific subscriber terminal. The functions are time-shifted so that the set of functions are orthogonal

2. The output of the multiplier is a code rate of 4.096 Mcps using 5 MHz bandwidth (see Table 5.5) that is split into two signals: in phase (I) signal and quadrature (Q) signal

3. The pulse shapes for the I and Q signals are smoothed, minimizing rapid signal transition that results in radio frequency emissions outside the allocated bandwidth

4. The balanced modulator multiplies the I and Q signals by two signals that are 90 degree phase-shifted. The number of bits per chip depends on the data rate supplied to the balanced modulator5. The output of the balanced modulator is then fed to a RF (radio frequency) amplifier

W-CDMA's 5 MHz bandwidth provides robust frequency diversity. Selective frequency fading usually affects only a 200 - 300 kHz range of the signal. Time diversity happens because of multipath fading channels. W-CDMA solves the problem in a couple of different ways. One way is the selection of only the strongest signal, a process similar to antenna diversity. Rake reception is another technique where weak signals are added together to build a strong signal. Inherent time diversity receiver provides robustness against fading.

Table 5.5 W-CDMA system characteristics

Bandwidth (MHz)	PN Rate (Mcps)	Code Rate (Mcps)	Data Rate (kbps)	Channel Selection
1.25	1.228	1.228	1.2-9.6	Walsh 64
5.0	4.096	4.096	2-64	Walsh 64X4
10.0	8.192	8.192	2-64	Walsh 64X8
15.0	12.288	12.288	2-64	Hadamard 48X4 & 96X8

Table 5.6 Maximum number of channels in W-CDMA

Bandwidth (MHz)	Total Number of Channels			
	64 kbps	32 kbps	16 kbps	Total
1.25	16	32	64	64
5.0	64	128	256	256
10.0	128	256	512	512
15.0	192	384	768	768

Exact time alignment in W-CDMA is not necessary. Time offset or variable offset is why W-CDMA is considered to be quasi-orthogonal by time. This characteristic is used to decrease the interference by shifting the time alignment by 1.25 msec.

W-CDMA employs ADPCM (adaptive differential pulse code modulation) as a speech coding method at a coding rate of 32 kbps. Voice is sampled and digitized at 64 kbps then supplied to the speech coder that characterizes and compresses the data at a rate of 16 - 32 kbps, depending on speech activity. The acceptable BER and the speech encoding are variable, which improves the capacity in a W-CDMA environment. Data is sent when speech is detected; there is no reservation of time slots or frequencies, thus you are achieving maximum utilization of the available bandwidth. To achieve the same effect in a TDMA or FDMA environment, statistical multiplexing must be employed with speech detection. Statistical multiplexing requires frequencies and time slots to be reassigned, thus complicating the system and raising the cost.

Power control in a W-CDMA environment is an open and closed loops. An open loop is a coarse adjustment of the signal strength. This means the sub scriber terminal continually receives from the BTS Radio Frequency amplifier adjustments measuring signal strength loss and the terminal reacts accordingly. A closed loop is fine adjustment of the signal strength, meaning in every 1.25 msec time slot from the BTS there is a power control bit indicating to the sub- scriber unit to increase or decrease transmission power. The end result is the signal received at the BTS is always at approximately the same power level.

There are two fundamental types of W-CDMA systems–synchronous and asynchronous. In synchronous operations, all symbol/chip transmissions of all subscribers are orthogonal by time eliminating co-subscriber interference. This decreases interference and increases channel capacity, but increases system complexity. Asynchronous operation, on the other hand, permits co-subscriber interference and allows more flexibility in system design, but lowers channel capacity. Synchronization on a system level is coordinated through the use of the synchronization channel. Synchronization on the subscriber level is coordinated via the pilot channel reference clock and is used in demodulating the received signal.

Hand-offs in a W-CDMA system are soft hand-offs. They are initiated by the subscriber terminal finding a better paging channel from a different BTS. The subscriber terminal communicates with both BTSs while the MSC coordinates the simultaneous communication. The call is then handed off from one BTS to the other, completing what is known as a soft hand-off. The hand-off is referred to as soft because the terminal is always in communication with a BTS, a condition that results in fewer dropped calls.

Japan's jump into W-CDMA is encouraged by a lack of capacity in the presently deployed system. The population density is such that a third-generation system is needed immediately. NTT predicts that wireless subscribers will equal wireline subscribers in the year 2000 at 60 million.

The Japanese W-CDMA system will be connected to an advanced broadband digital wireline network. The wire line connections are to be as follows: ATM adaptation layer 2 (AAL2) to be used between the BTS and the MSC via the BSC, PSTN and ISDN from MSC to the central office, and TCP/IP for Internet connections. The experimental prototype includes three cell sites, seven mobile stations, and up to 2 Mbps transmission rate. Other W-CDMA system parameters for the NTT DoCoMo/Ericsson testbed are outlined in Table 5.7.The following are the features in the NTT DoCoMo/Ericsson W-CDMA experimental system:

1. Subscriber unit can receive multiple channels resulting in multimedia bandwidth. The NTT DoCoMo W-CDMA system can accommodate up to six

64 kbps channels simultaneously for a total bandwidth of 384 kbps per subscriber, enabling six different tele services at the one time. This bit rate achieves the NTT DoCoMo phase one testbed goal of 384 kbps per sub- scriber; phase two has the goal of achieving 2 Mbps per subscriber. For more details about W-CDMA channel information see Tables 5.5 and 5.6.

2. The system allows for future expansion with the aid of adaptive antennas.

Adaptive antennas use SDMA techniques. As explained in section (5.3.3), SDMA manages interference and thus increases the network capacity, im- proves link quality, increases signal range, reduces transmission power, and extends the life and profitability of the deployed infrastructure.

3. New random access procedure with fast synchronization that provides flexibility in user data rates

4. Protocol structure that is similar to the GSM protocol structure

5. Inter-Frequency Hand-off (IFHO)

6. Hierarchical Cell Structure (HCS), permitting hand-offs between different wireless systems, (i.e. a hand-off between PHS infrastructure to the *W*- CDMA infrastructure)

7. VOX - Voice activation silence suppression, does not send data when the audio level is below a threshold. VOX is also noted in the PHS ARIB standard as a low power consumption operation for the private system.

8. Speech coding Orthogonal Variable Spreading Factor codes (OVSF). Utilization of a speech detection tool and orthogonal speech codes provides maximum bandwidth utilization in the W-CDMA environment. The speech detection tool, as explained earlier, assists in transmitting only the necessary data by transmitting less when speech activity is low. The orthogonal speech codes prevent interference with other channels decreasing interference and increasing capacity.

Table 5.7 NIT DoCoMo/Ericsson W-CDMA experimental system specs

Frequency band	
Carrier bandwidth	5 MHz
Carrier frequency forward link	2175 MHz
Carrier frequency reverse link	1990.5 MHz
Number of carriers	2 per sector
Number of sectors	6 per base transceiver station
Chip rate	4.096 Mchips per second
Frame length	10 ms
Time slots per frame	16
Services	
Voice	8 kbps voice operated relay
Packet data	2.4 - 384 kbps
Video, UDI data	64 - 384 kbps
Exchange Terminals	
1.5, 2 and 6.3 Mbps	G.703/G.704/I.4331
Protocols	AAL2 & AAL5
Modulation/Demodulation	
Data (Downlink)	QPSK
Data (Uplink)	O-QPSK
Spread (Downlink)	BPSK
Spread (Uplink)	O-QPSK
Demodulation	RAKE
Coding	
Short code	256 to 16 chip layered orthogonal code
Long Code (forward link)	10ms $2^{18} - 1$ chip gold code cut into 10 ms lengths
Long Code (reverse link)	2^{18} *10ms $2^{41} - 1$ chip gold code cut into 2^{18} *10ms lengths
FEC Coding	
Inner coding for traffic channels	Convolutional (R=1/3, K=9) Soft decision Viterbi decoding
Inner coding for control channels	Convolutional (R=1/2, K=9) Soft decision Viterbi decoding
Outer coding for UDI data	Reed Solomon coding RS(36,32)
BTS diversity	RAKE + antenna, 1 to 8 chip time window, 2 branch
BTS synchronization	Asynchronous
Power control	Closed loop + Open loop

Space Division Multiple Access

SDMA is a technology which enhances the quality and coverage of wireless communication systems. It uses a technique wherein the subscriber's access is via a narrow focused radio beam and the location of the subscriber is tracked adaptively by an intelligent antenna array system (see Figure 5.6). The name

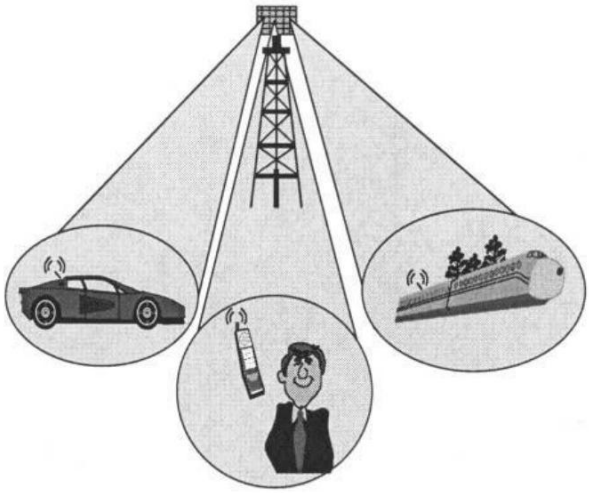


Figure 5.6 Space Division Multiple Access (SDMA)

SDMA is derived from the physical spatial characteristics between the focused radio beams. Spatial processing is not a new concept; it is used in presently deployed cellular infrastructures. For example, some cell sites are sectored at 120 degree. Also, most base station sites use two antennas for diversity reception regardless of whether they are sectored or not.

The most distinguishing aspect of SDMA is its management of interference. Reducing interference increases the effective network capacity, link quality and signal range. It also reduces the transmission power. Collectively, all the benefits brought by SDMA are expected to extend the life and profitability of second-generation network infrastructure.

SDMA is applied to the TDMA and CDMA systems differently because of the systems' basic differences. TDMA co-cell subscribers are orthogonal by time. Increasing the capacity in a TDMA environment by employing SDMA techniques requires multiple users on different radio beams to be assigned to the same carrier frequency and time slot. If the spatial component becomes insufficient between subscribers then an intra-sector hand-off is required to be initiated. The TDMA protocol needs to be expanded to permit these intra-sector handoffs.

CDMA subscribers use the same frequency and are quasi-orthogonal by time. Subscriber signals are distinguished by code filtering, not by time slots. Because of these CDMA characteristics, no intra-sector hand-offs are needed, keeping protocol overhead to a minimum.

Four types of interference concern cellular systems:

1. Background noise
2. External interference
3. Other cell interference, and
4. Other user noise

All systems deal with interference types 1 and 2. Interference types 3 and 4 are dealt with differently depending on the type of access method. In a TDMA system, interference types 3 and 4 are orthogonal either by frequency or time and do improve

with frequency reuse planning. An interference signal from a neighboring cell base station is orthogonal by frequency to the desired signal. Also an interference signal from a co-subscriber within the same cell is orthogonal by time to the desired signal.

In a CDMA system, interference types 3 and 4 are spread across the same frequency and not necessarily orthogonal by time. In CDMA, all subscribers use the same frequency. The access method distinguishes between the desired signal and interference types 1 through 4 (which include co-subscriber interference) by using a Pseudo Noise (PN) code. The PN code is known by both the base station and the subscriber unit for spreading and dispreading the desired signal in the bandwidth. In a SD/CDMA environment, the spreading code acts like a direction estimator. The spreading code has the responsibility of locating the signal within the interference so the antenna array just has to establish an antenna beam in the direction of the user. In the TDMA environment, the antenna array has to distinguish between the interferer and the user whose signal structures are the same. The interferer signals have to be "nulled" before establishing a radio beam; the consequence of which might be the cancellation of all but one subscriber accessing that antenna array.

The employment of SDMA in a CDMA environment provides an easy increase in capacity. There are no additional protocols or controls that need to be implemented; only the deployment of an intelligent antenna array is required. Employment of SDMA in a TDMA environment, however, requires new frequency planning and alteration of protocols.

3. Explain fourth generation wireless research (6)

Beyond third generation wireless infrastructure is W-ATM (Wireless Asynchronous Transfer Mode). Wireless infrastructure is greatly influenced by the wire line infrastructure. Future directions in wire line infrastructure are towards ATM, and the wireless infrastructure will align itself appropriately. W-ATM is being researched because of the possibilities of providing high speed data transmissions with a low BER and high QoS in densely populated areas.

ATM is a protocol designed to accommodate multiple network services. The intent is to have one type of network for all types of data thereby increasing efficiency, services, and throughput, while decreasing costs and network complexity. ATM is an end-to-end communication system accommodating network services requirements for lossy or lossless data, bursty traffic with real-time requirements, or data with no time requirements.

W-ATM is a communication system for hybrid tethered/tetherless environments. It has challenges and obstacles at layers one, two, and three of the OSI (Open Systems Interconnection) model. Comments in this section will be focused on layer one issues and solutions. Many W-ATM layer one issues are mainly comprised of characteristics found in a mobile radio environment. This includes:

- Fading
- Multi-path propagation
- Signal attenuation, and
- Interference types, including inter-cell subscriber, intra-cell subscriber, background noise, and external or other noise.

Some significant research toward resolving W-ATM layer one issues is in the area of diversity reception like antenna arrays and SDMA. Diversity reception techniques solve some of the issues that W-ATM is facing, like fading and multi- path propagation. Multiple received signals caused by multipath propagation, received at antennas spaced at a distance of a fraction of the wavelength, allows the received signals to be treated as independent rays. Statistically, one of the received signals at a given point will not have faded, then by using diversity combining, the strongest portion of the two independent signals are used to create a third signal, partially eliminating the effects of multipath propagation and fading. Diversity combining also allows the mobile terminal to reduce the transmit power something battery researchers like to hear.

Higher frequencies and wide bands are capable of delivering the needed through- put rates expected of an ATM network. Interference and other layer one issues at higher frequencies are minimized by decreasing the transmission distance between the mobile terminal and the base station, thereby improving BER and QoS. Given the bandwidth, BER, and QoS requirements of an ATM network, W-ATM will be deployed first in the private and public pico-cell infrastructure or in highly populated areas, i.e. indoor wireless LANs or systems like Japan's PHS infrastructure.

Japan has the advantages of already having a publicly-deployed pico cell network and the population density to make a pico cell network financially possible. The PHS network is deployed on the island of Japan, where 125 million people are packed into an area slightly smaller than the size of California. With a publicly-deployed pico-cell network, the evolution to W-ATM will be swift and decisive. In the year 2002, Japan will be fulfilling the mobile multimedia dream with 10 Mbps wireless ATM links.

4. Explain in detail about state of industry and Mobility support software.(6)

The mobile client-application architectures which are emerging in commercial products can be roughly divided into three overlapping classes: Remote-Node,Client Proxy, and Replication. The basis for this subdivision is the need to address the problems associated with wireless bandwidth and battery limitations and the alternatives, that are commercially available today, for managing those problems. This classification is, therefore, different from the "research" classification. A brief description of each class follows.

■ *Remote-Node* This approach attempts to create a facsimile of a fixed network client node by hiding all artifacts introduced by wireless communications. Under this model, all client software which run on a wired network platform would function without change on a mobile platform that includes a compatible OS and other library services. Accordingly, it places the most stringent demands on the middleware and other software (which supports the client application) to mediate the problems that arise as wire- less artifacts. As a result, this approach is most susceptible to failures in the wireless infrastructure. Software packages which adopt this approach may recognize some of the wireless limitations and adapt their behavior accordingly. For example, when response time is of concern, the limited bandwidth of wireless communications encourages the system to deliver records one at a time as they are retrieved from a database server rather than sending all record hits for some query. However, the ultimate goal is to provide an opaque overlay for the underlying ensemble of networks that shields the user from any concern for their interoperability. Remote-node applications can be realized by porting full clients (as used in

the wireline network) to a mobile computer with compatible communication middle- ware.
Shiva PPP is a famous middleware that supports most TCP/IP clients.

- *Client Proxy* This approach, characterized by products like Oracle Mo- bile Agents, attempts to minimize transmission costs and the impact of disconnects by buffering a client's requests, and/or the servers responses, and by resorting to batch transmissions. In this way, a user may select a variety of record types from several different tables, and then save battery power by disconnecting while the server processes the request. At some later time, the client can reconnect and receive a batch of records that satisfies all of the requests. The underlying assumption is that the end-user recognizes that periods of disconnect will occur, and that these periods will not impact the user's ability to perform useful work.

- *Replication* Clients which will be disconnected for extended periods of time, but which require immediate access to important data can satisfy those requests from locally cached replicas of key subsets of the database which are stored at some server site. Changes to the data that occur either at the client or the server must be reconciled through periodic client connects which may be initiated manually by the user, or automatically by the replication software. Some update conflicts may occur when multiple disconnected clients alter the same records. These collisions must be reconciled in some way.

5. Explain the End-user Applications (11) (UQ April'13)

A flurry of activity appeared in the trade press in late 1995 describing the rush by vendors, both large and small, to market mobile client software packages. Some of those products are discussed in this section. Recent literature search suggests that many of these products never materialized, were re-targeted to wired networks, or in some cases, are still struggling with weak sales. However, there are some big players with deep enough pockets to continue to pursue this marketplace. The discussions here are restricted to those products and services that still appear to have a current or promised market presence.

Oracle Mobile Agents

This product is a buffering and communications package for wireless platforms. A software agent that runs on the mobile client platform intercepts requests made by the client to the Oracle server and buffers them for a later transmission to the server. A companion Oracle agent runs on the Oracle server platform. That agent receives the buffered requests, submits them to the Oracle server, and buffers the responses for later transmission to the client. The server agent is capable of serving any number of mobile agents simultaneously. Conversely, a client agent can access any server agent that it knows about and for which it holds the appropriate DBA access privileges. Oracle agents can run on mobile platforms equipped with NT, Unix, or Windows and can communicate over TCP/IP using Shiva's
PPP communications middleware. This product does not automatically support transactions or queries that span multiple Oracle servers.

Oracle Lite

This product is a cut-down version of the Oracle server that can run in a small portable system (or a desktop workstation). It can be used as a companion technology for the Oracle Agent Software to store local copies of subsets of corporate databases and can accumulate updates to the data that are generated locally at the mobile client. Oracle may provide the Oracle Lite server with "two way" replication which could automatically propagate updates either to the client from the central server site, or vice versa. Recently, Oracle and Palm Computing (a 3Com company) announced an alliance to integrate the Oracle Lite client database and the 3Com Palm III and Palm Pilot organizers, allowing new and existing Palm Computing platform applications and data to be replicated, synchronized, and shared with an Oracle 8 database server.

Oracle Software Manager

This product is intended for a database administrator who needs to propagate software updates to remote copies of the Oracle server. It is capable of performing the distribution via hardwired networks or through wireless connections. It is not clear whether this package is versatile enough to accomplish distributed software update to a collection of mobile devices as though the entire operation were a distributed transaction. For example, if the DBA needs to update the mobile Oracle-Lite server software for entire sales staff, the updates may have to be performed individually by the DBA.

Oracle Replication Manager

Oracle has announced a version of its Replication Manager which will eventually support bi-directional replication among a collection of distributed and centralized server databases. The Oracle approach is based on a peer-to-peer model, much like Lotus Notes, in which a collection of distributed processes manage replication collectively.

Sybase SQL Remote

Unlike the Oracle Replication Manager, the Sybase product called SQL Remote has adopted a centralized model for managing replication. This product is a member of the Sybase SQL Any Where suite of tools (formerly called Watcom SQL). Also, Sybase has optimized its replication server to accommodate users that are only occasionally connected. So while this product has been developed with wired network users as a primary target, the software does include a component that recognizes the frequent disconnects that typify mobile users.

6. Explain mobility middleware (6) APRIL/MAY2014

The majority of products targeted for the middleware market rely on TCP/IP and socket-like connections for the client server interface whether they are in- tended to be deployed in the wireline network arena or the wireless domain. Variants of TCP have been proposed to circumvent the problems that plague TCP for some wireless applications. By choosing to adopt this defacto standard transport protocol, vendors are positioning their products for deployment in a large existing infrastructure. As a result, it is already possible to surf the Internet using a Netscape interface on many wireless platforms and a simple cellular phone connection.

Two key players in the wired-network middleware market that provide sup- port for distributed users are Novell's Netware and Microsoft's Remote Access. Neither of these

products will be discussed further since neither has yet announced plans (that we have seen) for moving into the wireless middleware domain. However, Microsoft Exchange has been integrated with Shiva's PPP software that allows communication of clients to servers through the cellular phone network.

MobileWare Office Server

This suite of products was introduced in 1995 as a solution to managing mobile access to corporate data. The basic strategy that underlies MobileWare is to minimize mobile platform connect time by executing data transfers in a burst mode. The intent of this software is to make the mobile platform appear to the user as though it were actually a node connected into the wired network. The initial customer target focused on large sales staffs that were primarily mobile and who needed access on demand to sales support information that was too bulky and/or volatile to carry on extended trips.

The current flagship product, MobileWare Office Server, includes a native Lotus Notes mail and database replication support. MobileWare Office Server is an agent-based middleware for wireless or wireline access to application data services supported by Mobile Ware Office Server includes Lotus Notes, Web browsing, e-mail, and file transfer. A core component of the MobileWare Office Server is the Intelligent Transport Engine. The transport engine provides several features including:

- Connection Profiles: The user chooses from a collection of profiles based on current working environment (LAN, Dial-up, or wireless connections, and TCP/IP, NetBios, etc.). Each profile contains a set of tuned parameters that optimize the communication between the clients and the servers.

- Data check pointing: This ensures efficient recovery and fast reconnection after failures and involuntary disconnections.

- Automatic reconnection in response to involuntary lost connections Data compression.

-

- Dynamic Packet-Scaling: Based on the current connection quality and capacity, data packets are dynamically re-sized to minimize connection time.

- Encryption and authentication. Uses DES encryption for per-connection authentication.

- Security. Forces re-authentication from the client upon receipt of any unregistered packet.

- Queuing. Application data is stored on both the client and the server in a client assigned outbox until a connection is made to transfer the data.

- Follow-Me Server. Uses a notification and delivery mechanism for events such as arrival of data to the client's outboxing on the server. The user's mobile computer is notified if connected, and alternative notification procedures (such as paging) are allowed. MobileWare Corporation was founded in 1991, and is now a private subsidiary of Itochu Japan.

Shiva PPP

Shiva's remote access client (known as PPP for Point-to-Point Protocol) enables mobile users to access servers embedded in either wireline or mobile servers almost seamlessly. For example, a client application that uses transaction processing services from BEA's Tuxedo can now access those services from a mobile platform using PPP. This software suite provides some limited security features such as limiting the number of login tries, or disconnecting a session and calling the user back at a pre-established number. However, it does not provide the rich collection of services available from Mobile Ware's Intelligent Transport Engine described above

7. Explain Adaptation (11)

Mobile computers must execute user- and system-level applications subject to a variety of resource constraints that generally can be ignored in modern desktop environments. The most important of these constraints are power, volatile and nonvolatile memory, and network band-width, although other physical limitations such as screen resolution are also important. In order to provide users with a reasonable computing environment, which approaches the best that currently available resources will allow, applications and/or system software must adapt to limited or fluctuating resource levels. For example, given a sudden severe constraint on available bandwidth, a mobile audio application might stop delivering a high-bit-rate audio stream and substitute a lower-quality stream. The user is likely to object less to the lower- quality delivery than to the significant dropouts and stuttering if the application attempted to continue delivering the high-quality stream. Similarly, a video application might adjust dynamically to fluctuations in bandwidth, switching from high-quality, high-frame-rate color video to black-and-white video to color still images to black-and-white still images as appropriate. A third example is a mobile videogame application adjusting to decreased battery levels by modifying resolution or disabling three-dimensional (3D) features to conserve power.

The spectrum of adaptation

At one end of the spectrum, adaptation may be entirely the responsibility of the mobile computer's operating system (OS); that is, the software for handling adaptation essentially is tucked under the OS hood, invisible to applications. At the other end, adaptation may be entirely the responsibility of individual applications; that is, each application must address all the issues of detecting and dealing with varying resource levels. Between these extremes, a number of application-aware strategies are possible, where the OS and individual application each share some of the burden of adaptation. While applications are involved in adaptation decisions, the middleware and/or OS provides support for resource monitoring and other low-level adaptation functions. The spectrum of adaptation is depicted in Fig. 6.1. we are concerned primarily with middleware for adaptation, that is, software interfaces that allow applications to take part in the adaptation process. Pure system-level adaptation strategies, those which take place in a mobile-aware file system such as Coda (e.g., caching and hoarding), are covered elsewhere in this book.

Resource monitoring

All adaptation strategies must measure available resources so that adaptation policies can be carried out. For some types of resources— cash, for example—monitoring is not so difficult. The user simply sets limits and appropriate accounts. For others, more elaborate

approaches are required. The Advanced Configuration and Power Interface (ACPI) provides developers with a standardized interface to power-level information on modern devices equipped with "smart" batteries. Accurately

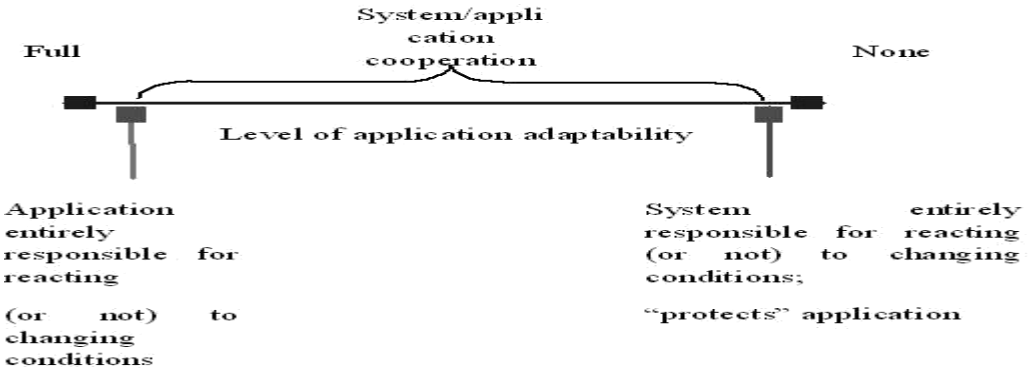


Figure 6.1 At one end of the spectrum of adaptation, applications are entirely responsible for reacting to changing resource levels. At the other end of the spectrum, the operating system reacts to changing resource levels without the interaction of individual applications. measuring network bandwidth over multi hop networks is more difficult. Some approaches are described in Lai and Baker (1999) for the interested reader. Whatever methods are used to measure resource levels have a direct impact on the effectiveness of the entire adaptation process because accurate measurement of resource levels is critical to making proper adaptation decisions.

Characterizing adaptation strategies

The Odyssey project (Noble et al., 1997; Noble, 2000) at Carnegie Mellon University was one of the first application-aware middleware systems, and it serves as a good model for understanding application-aware adaptation. In describing the Odyssey system, Satyanarayanan proposed several measures that are useful for classifying the goodness of an adaptation strategy. We describe these—fidelity, agility, and concurrency below.

Fidelity measures the degree to which a data item available to an application matches a reference copy. The reference copy for a data item is considered the exemplar, the ideal for that data item—essentially, the version of the data that a mobile computer would prefer given no resource constraints. Fidelity spans many dimensions, including perceived quality and consistency. For example, a server might store a 30- frame-per-second (fps), 24-bit color depth video at 1600 × 1200 resolution in its original form as shot by a digital video camera. This reference copy of the video is considered to have 100 percent fidelity. Owing to resource constraints such as limited network bandwidth, a mobile host may have to settle for a version of this video that is substantially reduced in quality (assigned a lower fidelity measure, perhaps 50 percent) or even for a sequence of individual black-and-white still frames (with a fidelity measure of 1 percent). If the video file on the server is replaced periodically with a newer version and a mobile host experiences complete disconnection, then an older, cached version of the video may be supplied to an application by adaptation middleware. Even if this cached version is of the same visual quality as the current, up-to-date copy, its fidelity may be considered lower because it is not the most recent copy (i.e., it is stale).

While some data-dependent dimensions of fidelity, such as the frame rate of a video or the recording quality of audio, are easily characterized, others, such as the extent to which a database table is out of date or a video is not the most current version available, do not map easily to a 0 to 100 percent fidelity scale. In cases where there is no obvious map- ping, a user's needs must be

taken into account carefully when assigning fidelity levels. More problematic is the fact that fidelity levels are in general type-dependent—there are as many different types of fidelity-related adaptations as there are types of data streams; for example, image compression schemes are quite different from audio compression schemes. Generally, an adaptation strategy should provide the highest fidelity possible given current and projected resource levels. Current adaptation middleware tends to concentrate on the present. Factoring projected resource levels into the equation is an area for future research.

Agility measures an adaptation middleware's responsiveness to changes in resource levels. For example, a highly agile system will deter- mine quickly and accurately that network bandwidth has increased substantially or that a fresh battery has been inserted. An adaptation middleware's agility directly limits the range of fidelity levels that can be accommodated. This is best illustrated with several examples, which show the importance of both speed and accuracy. For example, if the middleware is very slow to respond to a large increase in network bandwidth over a moderate time frame (perhaps induced by a user resting in an area with 802.11 WLAN connectivity), then chances to perform opportunistic caching, where a large amount of data are transferred and hoarded in response to high bandwidth, may be lost. Similarly, an adaptation middleware should notice that power levels have dropped substantially before critical levels are reached. Otherwise, a user enjoying a high-quality (and power-expensive) audio stream may be left with nothing, rather than a lower-quality audio stream that is sustainable

Agility, however, is not simply a measure of the speed with which resource levels are measured; accuracy is also extremely important. For example, consider an 802.11a wireless network, which is much more sensitive to line-of-sight issues than 802.11b or 802.11g networks. A momentary upward spike in available bandwidth, caused by a mobile host connected to an 802.11a network momentarily having perfect line of sight with an access point, should not necessarily result in adjustments to fidelity level. If such highly transient bandwidth increases result in a substantial increase in fidelity level of a streaming video, for example, many frames may be dropped when bandwidth suddenly returns to a lower level.

The last measure for adaptation middleware that we will discuss is con- currency. Although the last generation of PDAs (such as the original Pilot by Palm, Inc.) used single-threaded operating systems, capable of executing only one application at a time; newer PDAs, running newer versions of Palm OS, variants of Microsoft Windows, and Linux, run full- featured multitasking OSs. Thus it is reasonable to expect that even the least powerful of mobile devices, not to mention laptops that run desk- top operating systems, will execute many concurrent applications, all of which compete for limited resources such as power and network band- width. This expectation has a very important implication for adaptation: Handling adaptation at the left end of the spectrum (as depicted in Fig. 6.1), where individual applications assume full responsibility for adapting to resource levels, is probably not a good idea. To make intelligent decisions, each application would need to monitor available resources, be aware of the resource requirements of all other applications, and know about the adaptation decisions being made by the other applications. Thus some system-level support for resource monitoring, where the OS can maintain the "big picture" about available resources needs and resource levels, is important.

An application-aware adaptation architecture: Odyssey

The Odyssey architecture in greater detail. In the spectrum of adaptation, Odyssey sits in the middle—applications are assisted by the Odyssey middleware in making decisions concerning fidelity levels. Odyssey provides a good model for understanding the issues in application-aware

adaptation because the high-level architecture is clean, and the components for supporting adaptation are clearly delineated. The Odyssey architecture consists of several high-level components: the interceptor, which resides in the OS kernel, the viceroy, and one or more wardens. These are depicted in Fig. 6.2. The version of Odyssey described in Nobel and colleagues (1997) runs under NetBSD; more recent versions also support Linux and FreeBSD. To minimize changes to the OS kernel, Odyssey is implemented using the Virtual File System (VFS) interface, which is described in great detail for kernel hacker types in Bovet and Cesati (2002). Applications interact with Odyssey

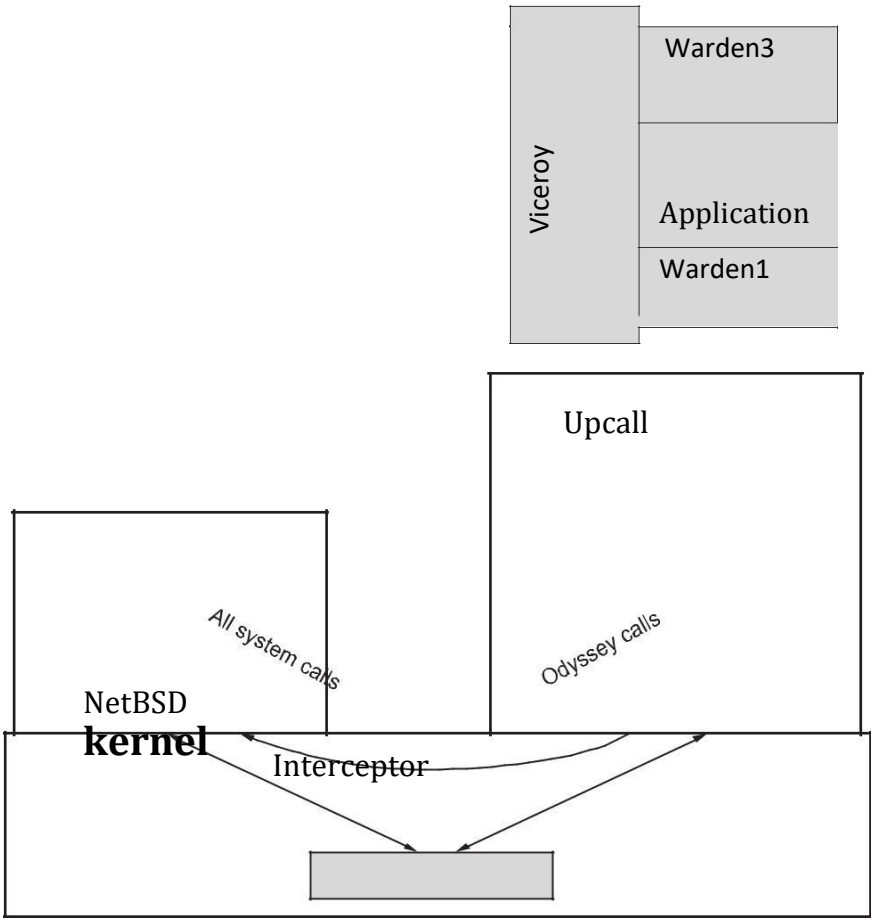


Figure 6.2 The Odyssey architecture consists of a type-independent viceroy and a number of type-specific wardens. Applications register windows of acceptable resource levels for particular types of data streams and receive notifications is when current resource levels fall outside the windows. Odyssey using (mostly) file system calls, and the interceptor, which resides in the kernel, performs redirection of Odyssey-specific system calls to the other Odyssey components.

The basic Odyssey model is for an application to choose a fidelity level for each data type that will be delivered—e.g., 320 × 240 color video at 15 fps. The application then computes resource needs for delivery of each stream and registers these needs with Odyssey in the form of a "window" specifying minimum and maximum need. The viceroy monitors avail- able resources and generates a callback to the application when available resources fall outside registered resource-level window. The application then chooses a new fidelity level, computes resource needs,

and registers these needs, as before. Thus applications are responsible for deciding fidelity levels and computing resource requirements—the primary contribution that Odyssey makes is to monitor resources and to notify applications when available resources fall outside constraints set by the application. Before describing a sample Odyssey application, the war- dens and viceroy are discussed in detail below.

Wardens. A warden is a type-specific component responsible for handling all adaptation-related operations for a particular sort of data stream (e.g., a source of digital images, audio, or video). Wardens sit between an application and a data source, handling caching and arranging for delivery of data of appropriate fidelity levels to the application. A warden must be written for each type of data source. An application typically must be partially rewritten (or an appropriate proxy installed) to accept data through a warden rather than through a direct connec- tion to a data source, such as a streaming video server.

Viceroy. In Odyssey, the viceroy is a type-independent component that is responsible for global resource control. All the wardens are statically compiled with the viceroy. The viceroy monitors resource levels (e.g., available network bandwidth) and initiates callbacks to an application when current resource levels fall outside a range registered by the application. The types of resources to be monitored by the viceroy in Odyssey include network bandwidth, cache, battery power, and CPU, although the initial implementations of the Odyssey architecture did not support all these resource types.

A sample Odyssey application

We now turn to one of the sample applications discussed in Nobel and colleagues (1997): the xanim video player. The xanim video player was modified to use Odyssey to adapt to varying network conditions, with three fidelity levels available—two levels of JPEG compression and black-and-white frames. The JPEG compression frames are labeled 99 and 50 percent fidelity, whereas the black-and-white content is labeled

1 percent fidelity. Integration of xanim with Odyssey is illustrated in Fig. 6.3. A “video warden” prefetches frames from a video server with the appropriate fidelity and supplies the application with metadata for the video being played and with individual frames of the video.

The performance of the modified xanim application was tested using simulated bandwidths of 140 kB/s for “high” bandwidth and 40 kB/s for “low” bandwidth. A number of strategies were used to vary bandwidth: step up, which holds bandwidth at the low level for 30 seconds, followed by an abrupt increase to high bandwidth for 30 seconds; step down, which reverses the bandwidth levels of step up but maintains the same time periods; impulse up, which maintains a low bandwidth over a 60-second period with a single 2-second spike of high bandwidth in the middle; and impulse down, which maintains high bandwidth for 60 seconds with a single 2-second spike of low bandwidth in the middle. Both high and low bandwidth levels are able to support black-and-white video and the lower-quality (50 percent fidelity) JPEG video. Only the high bandwidth level is sufficient for the 99 percent fidelity JPEG frames to be delivered without substantial numbers of dropped frames. In the tests, Odyssey maintained average fidelities of 73, 76, 50, and 98 percent for step up, step down, impulse up, and impulse down, respectively, all with less than 5 percent dropped frames. In contrast, trying to maintain the 99 percent fidelity rate by transferring high-quality video at all times, ignoring available network bandwidth, resulted in losses of 28 percent of the frames for step up and step down and 58 per- cent of the frames for impulse up. Several other adapted applications are discussed in the Odyssey publications.

MORE ADAPTATION MIDDLEWARE (APRIL 2013)

Puppeteer. For applications with well-defined, published interfaces, it is possible to provide adaptation support without modifying the applications directly. The Puppeteer architecture allows component-based applications with published interfaces to be adapted to environments with poor network bandwidth (a typical situation for mobile hosts) with- out modifying the application . This is accomplished by outfitting applications and data servers with custom proxies that support the adaptation process. A typical application adaptation under Puppeteer is a retrofit of Microsoft PowerPoint to support incremental loading of slides from a large presentation or sup- port for progressive JPEG format to speed image loading. Both these adaptations presumably would enhance a user's experience when handling a large PowerPoint presentation over a slow network link.

The Puppeteer architecture is depicted in Fig. 6.4. The Puppeteer provides a kernel that executes on both the client and server side proxies, supporting a document type called the Puppeteer Intermediate Format (PIF), a hierarchical, format-neutral format. The kernel also handles all communication between client and server sides. To adapt a document, the server and client side proxies communicate to establish a high-level PIF skeleton of the document. Adaptation policies control which portions of the document will be transferred and which fidelities will be chosen for the transmitted portions. For example, for a Microsoft PowerPoint document, selected slides may transferred, with images rendered at a lower fidelity than in the original presentation. The import driver and export driver parse native document format to PIF and PIF to native document format, respectively. Transcoders in Puppeteer per- form transformations on data items to support the adaptation policies. For example, a Puppeteer transcoder may reduce the quality of JPEG images or support downloading only a subset of a document's data. A typical Puppeteer-adapted application operates as follows:

■ When the user opens a document, the Puppeteer kernel instantiates an appropriate import driver on the server side.

■ The import driver parses the native document format and creates a PIF format document. The skeleton of the PIF is transmitted by the kernel to the client-side proxy.

■ On the client side, policies available to the client-side proxy result in requests to transfer selected portions of the PIF (at selected fidelities) from the server side. These items are rendered by the export driver into native format and supplied to the application through its well- known interface.

■ At this point, the user regains control of the application. If specified by the policy, additional portions of the requested document can be transferred by Puppeteer in the background and supplied to the application as they arrive.

Coordinating adaptation for multiple mobile applications The adaptation middleware architectures is coordination among adaptive applications. Odyssey and Puppeteer, for example, support sets of independently adapting applications but do not currently assist multiple applications in coordinating their adaptation strategies. When multiple applications are competing for shared resources, individual applications may

make decisions that are suboptimal. At least three issues are introduced when multiple applications attempt to adapt to limited resources—conflicting adaptation, suboptimal system operation, and suboptimal user experience.

Several sample scenarios illustrate these concerns. First, consider a situation where a number of applications executing on a mobile host with limited power periodically write data to disk. This would occur, for example, if two or more applications with automatic backup features were executing. Imagine that the mobile host maintains a powered-down state for its hard drives to conserve energy. Then, each time one of the automatic backup facilities executes, a hard disk on the system must be spun up. If the various applications perform automatic backups at uncoordinated times, then the disk likely will spin up quite frequently, wasting a significant amount of energy. If the applications coordinated to perform automatic backups, on the other hand, then disk writes could be performed "in bulk," maximizing the amount of time that the disk could remain powered down. This example illustrates suboptimal system operation despite adaptation.

Another issue when multiple applications adapt independently is conflicting adaptation. Imagine that one application is adapting to varying power, whereas another application is adapting to varying network bandwidth. When the battery level in the mobile device becomes a concern, then the power-conscious application might throttle its use of the network interface. This, in turn, makes more bandwidth available, which might trigger the bandwidth-conscious application to raise fidelity levels for a data stream, defeating the other application's attempt to save energy.

A third issue is that in the face of limited resources, a user's needs can be exceedingly difficult to predict. Thus some user participation in the adaptation process probably is necessary. To see this, imagine that a user is enjoying a high-bandwidth audio stream (Miles Davis, Kind of Blue?) while downloading a presentation she needs to review in 1 hour. With abundant bandwidth, both applications can be well served. However, if available bandwidth decreases sharply (because an 802.11 access point has gone down, for example, and the mobile host has fallen back to a 3G connection), should a lower-quality stream be chosen and the presentation download delayed because Miles is chewing up a few tens of kilobits per second? Or should the fun stop completely and the work take precedence? Efstatiou and colleagues propose using an adaptation policy language based on the event calculus (Kowalsky, 1986) to specify global adaptation policies. The requirements for their architecture are that a set of extensible adaptation attributes be sharable among applications, that the architecture be able to centrally control adaptation behavior, and that flexible, system-wide adaptation policies, depending on a variety of issues, be expressible in a policy language. Their architecture also allows human interaction in the adaptation process both to provide feedback to the user and to engage the user in resolving conflicts (e.g., Miles Davis meets downloading PowerPoint). Applications are required to register with the system, providing
a set of adaptation policies and modes of adaptation supported by the application. In addition, the application must expose a set of state variables that define the current state of the application. Each application generates events when its state variables change in meaningful ways so that the adaptation architecture can determine if adaptive actions need to be taken; for example, when a certain application is minimized, a global adaptation policy may cause that application to minimize its use of system resources. A registry in the architecture stores information about each application, and an adaptation controller monitors the state of the system, determining when adaptation is necessary and which applications should adapt. Another policy language–driven architecture advocating user involvement is described in Keeney and Cahill .

8. Write about Mobile Agents (11)

We now turn to another type of mobile middleware, mobile agent systems. Almost all computer users have used mobile code, whether they realize it or not—modern browsers support Javascript, Java applets, and other executable content, and simply viewing Web pages results in execution of the associated mobile code. Applets and their brethren are mostly static, in that code travels from one or more servers to a client and is executed on the client. For security reasons, the mobile code often is prevented from touching nonlocal resources. Mobile agents are a significant step for- ward in sophistication, supporting the migration of not only code but also state. Unlike applets, whose code typically travels (at an abstract level at least) one "hop" from server to client, mobile agents move freely about a network, making autonomous decisions on where to travel next. Mobile agents have a mission and move about the network extracting data and communicating with other agents in order to meet the mission goals.

Like adaptation middleware, mobile agent systems (e.g., Cabri, Leonardi, and Zambonelli, 2000; Gray, 1996, 1997; Gray et al., 1998, 2000; Bradshaw et al., 1999; Lange and Oshima, 1998; Peine and Stoplmann,1997; Wong, Paciorek, and Moore, 1999; Wong et al., 1997) support execution of mobile applications in resource-limited environments, but mobile agent systems go far beyond allowing local applications to respond to fluctuating resource levels. A mobile agent system is a dynamic client-server (CS) architecture that supports the migration of mobile applications (agents) to and from remote servers. An agent can migrate whenever it chooses either because it has accomplished its task completely or because it needs to travel to another location to obtain additional data. An alter- native to migration that an agent might exercise is to create one or more new agents dynamically and allow these to migrate. The main idea behind mobile agents is to get mobile code as close to the action as possible—mobile agents migrate to remote machines to perform computations and then return home with the goods.

For example, if a mobile user needs to search a set of databases, a traditional CS approach may perform remote procedure calls against the database servers. On the other hand, a mobile agents approach would dispatch one or more applications (agents) either directly to the data- base servers or to machines close to the servers. The agents then per- form queries against the database servers, sifting the results to formulate a suitable solution to the mobile user's problem. Finally, the mobile agents return home and deliver the results.

The advantages of this approach are obvious. First, if bandwidth available to the mobile user is limited and the database queries are complicated, then performing a series of remote queries against the servers might be prohibitively expensive. Since the agents can execute a number of queries much closer to the database servers in order to extract the desired information, a substantial amount of bandwidth might be saved (of course, transmission of the agent code must be taken into account). Second, continuous network connectivity is not required.

The mobile user might connect to the network, dispatch the agent, and then disconnect. When the mobile user connects to the network again later, the agent is able to return home and present its results. Finally, the agents are not only closer to the action, but they also can be executed on much more powerful computers, potentially speeding up the mining of the desired information. Of course, there are substantial difficulties in designing and implementing mobile agent systems.

Why mobile agents? And why not?

We first discuss the advantages of mobile agents at a conversational level, and then we look at the technical advantages and disadvantages in detail. First, a wide variety of applications can

be supported by mobile agent systems, covering electronic commerce (sending an agent shop- ping), network resource management (an agent might traverse the net- work, checking versions of installed applications and initiating upgrades where necessary), and information retrieval (an agent might be dis- patched to learn everything it can about Thelonious Monk).

An interesting observation made by Gray and colleagues (2000) is worth keeping in mind when thinking about agent-based applications: While particular applications may not make a strong case for deployment of mobile agent technology, sets of applications may make such a case. To see this point, consider the database query example discussed in the preceding section. Rather than using mobile agents, a custom application could be deployed (statically) on the database servers. This application accepts jobs (expressing the type of information required) from a mobile user, performs a sequence of appropriate queries, and then returns the results. Since most of the processing is done off the mobile host, the resource savings would be comparable to a mobile agents solution.

Similarly, little computational power on the mobile host is required because much of the processing can be offloaded onto the machine hosting the custom application. However, what if a slightly different application is desired by a mobile user? Then the server configuration must be changed. Like service discovery protocols, covered in mobile agent systems foster creation of powerful, personalized mobile applications based on common frameworks. While individual mobile applications can be written entirely without the use of agent technologies, the amount of effort to support a changing set of customized applications may be substantially higher than if mobile agents were used.

Mobile agent systems provide the following set of technical advantages

■ The limitations of a single client computer are reduced. Rather than being constrained by resource limitations such as local processor power, storage space, and particularly network bandwidth, applications can send agents “into the field” to gather data and perform computations using the resources of larger, well-connected servers.

■ The ability to customize applications easily is greatly improved. Unlike traditional CS applications, servers in an agent system merely pro- vide an execution environment for agents rather than running customized server applications. Agents can be freely customized (within the bounds of security restrictions imposed by servers) as the user’s needs evolve.

■ Flexible, disconnected operation is supported. Once dispatched, a mobile agent is largely independent of its “home” computer. It can per- form tasks in the field and return home when connectivity to the home computer is restored. Survivability is enhanced in this way, especially when the home computer is a resource-constrained device such as a PDA. With a traditional CS architecture, loss of power on a PDA might result in an abnormal termination of a user’s application.

Despite these advantages, mobile agent architectures have several significant disadvantages or, if that is too strong a word, disincentives. One is that neither a killer application nor a pressing need to deploy mobile agent technology has been identified. Despite their sexiness, mobile agents do not provide solutions to problems that are otherwise unsolvable; rather, they simply seem to provide a good framework in which to solve certain problems. In reflections on the Tacoma project (Milojicic, Douglis, and Wheel, 1998), Johansen, Schneider, and van Renesse note that while agents potentially reduce bandwidth and tolerate intermittent communication

well, bandwidth is becoming ever more plentiful, and communication is becoming more reliable. As wire- less networking improves and mobile devices become more powerful and more prevalent, will mobile agents technologies become less relevant? Further, while a number of systems exist, they are largely living in research laboratories. For mobile agent systems to meet even some of their potential, widespread deployment of agent environments is required so that agents may travel freely about the Internet.

A related problem is a lack of standardization. Most mobile agent systems are not interoperable. Some effort has gone into interoperability for agent systems, but currently, there seem to be no substantial market pressures forcing the formation of a single (or even several) standards for mobile agent systems. The Mobile Agents System Interoperability Facility (MASIF; Milojicic et al., 1999) is one early attempt at fostering agent interoperability for Java-based agent systems.

All the disadvantages just discussed are surmountable with a little technical effort—apply a good dose of marketing, and most disappear. There is a killer disadvantage, however, and that is security. Even applets and client-side scripting languages (such as Javascript), which make only a single hop, scare security-conscious users to death, and many users turn off Java, Javascript, and related technologies in their Web browsers. Such users maintain this security-conscious stance even when interacting with Web sites in which they place significant trust because the potential for serious damage is high should the sandbox leak. Security for mobile agent systems is far more problematic than simple mobile code systems such as Java applets because agents move autonomously.

There are at least two broad areas of concern. First, agents must be prevented from performing either unintentional or malicious damage as they travel about the network. Could an agent have been tampered with at its previous stop? Is it carrying a malicious logic payload? Does it contain contraband that might be deposited on a machine? Will the agent use local resources to launch a denial-of-service attack against another machine? Essentially, if agents are to be allowed to get “close to the action,” then the “action” must be convinced (and not just with some marketing) that the agents will not destroy important data or abuse resources. Second, the agents themselves must be protected from tampering by malicious servers. For example, an agent carrying credit card information to make purchases on behalf of its owner should be able to control access to the credit card number. Similarly, an agent equipped with a proprietary data-mining algorithm should be able to resist reverse engineering attacks as it traverses the network.

Agent architectures

To illustrate the basic components of mobile agent architectures, a high- level view of Telescript (White in Milojicic, Douglic, and Wheel, 1998) works well. Telescript was one of the first mobile agent systems, and while it is no longer under development, many subsequent systems bor- rowed ideas from Telescript. There are a number of important components in the Telescript architecture: agents, places, travel, meetings, connections, authorities, and permits. These are depicted in Fig. 6.5. Each of these components is described in detail below.

Places. In a mobile agent system, a network is composed of a set of places—each place is a location in the network where agents may visit. Each place is hosted by a server (or perhaps a user’s personal computer) and provides appropriate infrastructure to support a mobile agent migrating to and from that location. Servers in a network that do not

Figure 6.5 The major Telescript components are illustrated above. Tom has just dispatched an agent which has not yet arrived at the theater server. When Tom's agent arrives, it will interact with the static agent in the box office place to arrange for theater tickets. Daryl previously dispatched an agent to purchase tickets and has a connection with her agent in the box office place, so she can actively negotiate prices. Daryl's agent and the box office agent have identified each other through their respective authorities and permits associated with Daryl's agent have been evaluated to see what actions are permitted. The static agents in the drugstore and music store places, which both reside on a shopping center server, are currently idle. To interact with the drugstore or music store agents, Daryl or Tom's agents will have to travel to the drugstore and music store places .offer a "place" generally will not be visitable by agents. Places offer agents a resting spot in which they can access resources local to that place through a stationary agent that "lives" there, interacting with other agents currently visiting that place.

Travel. Travel allows agents to move closer to or to colocate with needed resources. For example, an agent dispatched by a user to obtain tickets to a jazz concert and reservations at one of several restaurants (depending on availability) might travel from its home place to the place hosted by the jazz club's box office before traveling to the places hosted by the restaurants. The primary difference between mobile code strategies such as Java applets and agents is that agents travel with at least part of their state intact—after travel, an agent can continue the computation it was engaged in at the instant that travel was initiated. Migration is studied in further detail below in the section entitled, "Migration Strategies."

Meetings. Meetings are local interactions between two or more agents in the same place. In Telescript, this means that the agents can invoke each other's procedures. The agent in search of jazz tickets and a restaurant reservation (discussed under "Travel" above) would engage in meetings with appropriate agents at the ticket office and at the restaurant's reservation office to perform its duty.

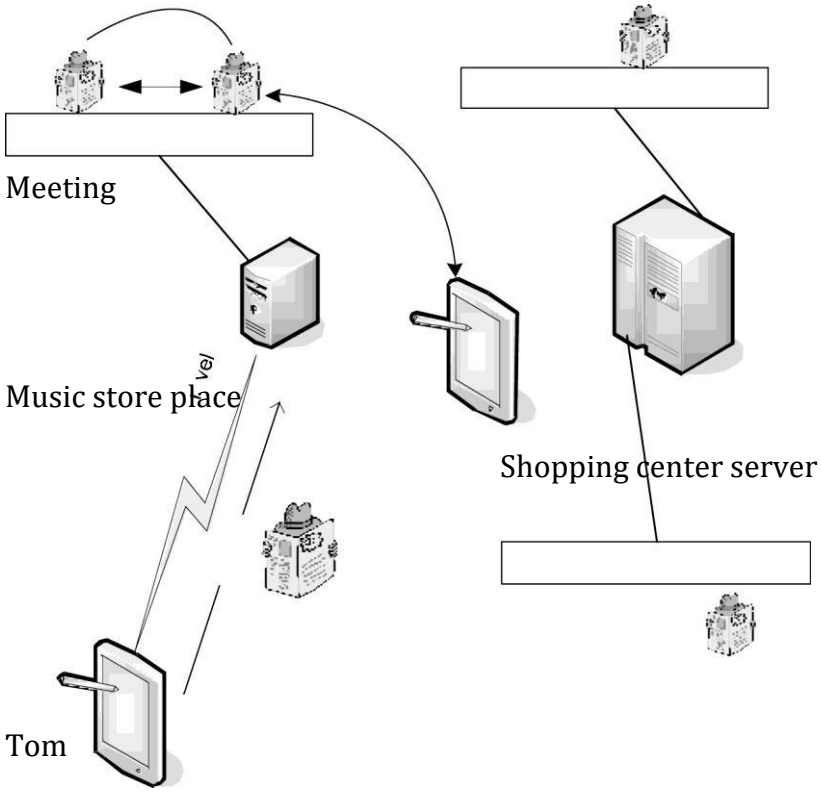
Connections. Connections allow agents at different places to communicate and allow agents to communicate with human users or other applications over a network. An agent in search of jazz tickets, for example, might contact the human who dispatched it to indicate that an additional show has been added, although the desired show was sold out (e.g., "Is the 11 P.M. show OK?"). Connections in Telescript require an agent to identify the name and location of the remote agent, along with some other information, such as required quality of service. This remote communication method, which tightly binds two communicating agents (since both name and location are required for communication), is the most restrictive of the mechanisms discussed in further detail below in the section entitled, "Communication Strategies."

Authorities. An agent's or place's authority is the person or organization (in the real world) that it represents. In Telescript, agents may not with- hold their authority; that is, anonymous operation is not allowed—the primary justification for this limitation is to deter malicious agent activity. When an agent wishes to migrate to another location, the destination can check the authority to determine if migration will be permitted. Similarly, an agent may examine the authority of a potential destination t determine if it wishes to migrate there. Implementation of authorities in an un trusted network is nontrivial and requires strong cryptographic methods because an agent's authority must be un forgeable.

Permits. Permits determine what agents and places can do—they are sets of capabilities. In general, these capabilities may have virtually any form, but in Telescript they come in two flavors. The first type of capability determines whether an agent or place may execute certain types of instructions, such as instructions that create new agents. The second type of capability places resource limits on agents, such as a maximum number of bytes of network traffic that may be generated or a maximum lifetime in seconds. If an agent attempts to exceed the limitations imposed by its permits, it is destroyed. The actions permitted an agent are those which are allowed by both its internal permits and the place(s) it visits.

Other issues. A number of details must be taken into account when designing an architecture to support mobile agents, but one of the fundamental issues is the choice of language for implementation of the agents (which might differ from the language used to implement the agent architecture). To support migration of agents, all computers to which an agent may migrate must share a common execution language. While it is possible to restrict agents to a particular computer architecture and OS (e.g., Intel 80 × 86 running Linux 2.4), clearly, agent sys- tems that can operate in heterogeneous computer environments are the most powerful. Compiled languages such as C and C++are problematic because agent executables must be available for every binary architecture on which agents will execute. Currently, interpreted languages such as Java, TCL, and Scheme are the most popular choices because many problems with code mobility are alleviated by interpreted languages. In cases where traditionally compiled languages such as C++ are used for implementation of agents, a portable, interpreted byte code typically is emitted by a custom compiler to enable portability. Java is particularly popular for mobile agents because Java has native support for multithreading, object serialization (which allows the state of arbitrary objects to be captured and transmitted), and remote procedure calls. Other factors, aside from the implementation language for agents, include migration strategies, communication, and security. Migration and communication strategies are discussed in detail below.

Authority/permit evaluation



Migration strategies

To support the migration of agents, it must be possible to either capture the state of an agent or to spawn an additional process that captures the state of the agent. This process state must then be transmitted to the remote machine to which the agent (or its child, in the case of spawn- ing an additional process) will migrate. This is equivalent to process check pointing, where the state of a process, including the stack, heap, program code, static variables, etc., is captured and stored for a later resuscitation of the process. Process check pointing is a very difficult problem that has been studied in the operating systems and distributed systems communities for a number of years, primarily to support fault tolerance and load balancing (Jul et al., 1988; Douglis and Ousterhout,1991; Plank, 1995). In general, commodity operating systems do not provide adequate support for check pointing of processes, and add-on solutions [e.g., in the form of libraries such as libckpt (Plant, 1995)] are nonportable and impose significant restrictions, such as an inability to reconstitute network connections transparently. A number of research operating systems have been designed that better support process migration, but since none of these is viable commercially (even in the slightest sense), they are not currently appropriate platforms. Check pointing processes executing inside a virtual machine, such as Java processes, are a bit easier, but currently most of these solutions (Richard and Tu, 1998; Sakamoto, Sekiguchi, and Yonezawa, 2000; Truyen et al., 2000) also impose limitations, such as restrictions on the use of call- backs, network connections, and file activity. The virtual machine itself can be check pointed, but then the issues of portability discussed earlier reemerge, and network connections and file access will still pose problems. So where is this going? The punch line is that if commodity operating systems are to be targeted by agent systems—and for wide-scale deployment, this must be the case—then completely capturing the state of general processes to support migration is rife with problems.

One solution is to impose strong restrictions on the programming model used for mobile agents. Essentially, this entails capturing only the essential internal state of an agent, i.e., sufficient information about its current execution state to continue the computation on reconstitution, combined with a local cleanup policy. This means that an agent might perform a local cleanup, including tearing down communication connections and closing local files, before requesting that the agent middleware perform a migration operation. For example, in Aglets (Lange and Oshima, 1998), which is a Java-based mobile agents system, agents are notified at the beginning of a migration operation. It is the responsibility of an individual agent, on receiving such a notification, to save any significant state in local variables so that the agent can be properly “reconstituted” at the new location. Such a state may include the names of communication peers, loop indices, etc. Agent migration in Aglets begins with an agent initiating a migration (its own or that of another agent) by invoking dispatch(). A callback, onDispatch(), will be triggered subsequently, notifying the agent that it must save its state. After the migration, the agent’s onArrival()callback will be invoked so that the agent can complete its state restoration.

Communication strategies

Communication among agents in a mobile agent system can take many forms, including the use of traditional CS techniques, remote procedure call, remote method invocation (e.g., using Java’s RMI), mailboxes, meeting places, and coordination languages. Each of these communication strategies has advantages and disadvantages, some of which are exacerbated in mobile agent systems. One consideration is the degree of temporal and spatial locality

exhibited by a communication scheme (Cabri, Leonardi, and Zambonelli, 2000).Temporal locality means that communication among two or more agents must take place at the same physical time, like a traditional telephone conversation. Interagent communication mechanisms exhibiting temporal locality are limiting in a mobile agent's architecture because all agents participating in a communication must have network connectivity at the time the communication occurs. If an agent is in transit, then attempts to communicate with that agent typically will fail.

Spatial locality means that the participants must be able to name each other for communication to be possible—in other words, unique names must be associated with agents, and their names must be sufficient for determining their current location. Some of the possible communication mechanisms for interagent communication are discussed below.

Traditional CS communication. Advantages of traditional CS mechanisms such as sockets-based communication, Remote Method Invocation (RMI) in Java, and CORBA include a familiar programming model for soft- ware developers and compatibility with existing applications. Significant drawbacks include strong temporal and spatial locality—for communication to be possible, agents must be able to name their communication peers and initiate communication when their peers are also connected. RMI and other communication mechanisms built on the Transmission Control Protocol/Internet Protocol (TCP/IP) also require stable network connectivity; otherwise, timeouts and subsequent connection reestablishments will diminish performance significantly. Examples of agents systems that use traditional CS mechanisms are D'Agents (Gray et al.,

1998) and Aglets (Lange and Oshima, 1998). In Aglets, an agent first must obtain another agent's proxy object (of type AgletProxy) before communication can take place. This proxy allows the holder to transmit arbitrary messages to the target and to request that the target agent per- form operations such as migration and cloning (which creates an identical agent). To obtain a proxy object for a target agent, an agent typically must provide both the name of the target agent and its current location. If either agent moves, then the proxy must be reacquired.

Meeting places. Meeting places are specific places where agents can congregate in order to exchange messages and typically are defined statically, avoiding problems with spatial locality but not temporal locality. In Ara (Peine and Stolpmann, 1997), meeting places are called service points and provide a mechanism for agents to perform local communication. Messages are directed to a service point rather than to a specific agent, eliminating the need to know the names of colocated agents.

Tuple spaces. Linda-like tuple spaces are also appropriate for interagent communication. Linda provides global repositories for tuples (essentially lists of values), and processes communicate and coordinate by inserting tuples into the tuple space, reading tuples that have been placed into the tuple space, and removing tuples from the tuple space. Tuple spaces eliminate temporal and spatial bindings between communicating processes because communication is anonymous and asynchronous.

9. Explain interoperability and standadization (11)

The wired infrastructure has been designed and deployed around a rich set of international standards. For example, the legacy local area network (LAN) consists of such technologies as Ethernet, Token-Ring, and Token-Bus which were defined in precise details by the IEEE 802 committees. Moreover, newer network technologies like FDDI, HIPPI, and Fibre Channel have been defined by the ANSI X3T working groups with a mature set of approved specifications. Both the IEEE and the ANSI bodies add further credibility to their work by helping international organizations like ISO and ITU to easily migrate the specifications into international standards bodies for worldwide acceptance. Similar efforts are underway in the ATM Forum to create a set of implementation agreements which should permit interoperability between different vendor implementations and products.

The wireless industry currently embraces a small number of standards. The closest effort is within the IEEE 802 working group which recently completed the IEEE 802.11 Wireless MAC (media access control) standard. The primary objective of the IEEE 802.11 effort is to permit wireless LANs from different vendors to interoperate.

IEEE 802.11 does not, however, address the needs of the wide area wireless networking industry which currently deploys various packetized protocols (e.g. CDPD, GPRS) across unused cellular channels. Each network type is based on its own set of assumptions about the kinds of service the customers are willing to purchase. Service providers for each of these types of networks have different goals and strategies and do not seem likely to provide interoperability among the other classes of service.

Other mobile infrastructures are also lacking in internationally recognized standards. This is evident in the cellular telephone industry: a PHS telephone will not function in a cell serviced by a GSM or PCS infrastructure. The same is true for any combination of the aforementioned technologies. Moreover, cordless telephones, infrared transmission, satellite channels, and most mobile communication systems are either based on proprietary data interfaces, or have implemented selected parts of existing and/or emerging deployment agreements. Such key attributes as Quality of Service, Location Register con- tents, Database formats, update policies, and data exchange rates are left to the equipment providers and service providers which may be based more on deployment schedules than on availability of standards and interoperability guarantees. The emerging UMTS system standard which is expected to be deployed by the year 2002, will provide a golden opportunity for interoperability of data links interfaces, digital voice, and wireless data and services.

Many of the client-application products, or the communications substrate that they rely on, are recognizing that several competing wireless transmission pro- tocols exist with each network type. They also recognize that the number of such protocols may grow or shrink. As a result, these client-level packages are adapted to use the popular underlying protocols. This limited form of inter- operability appears to meet the needs for developers of client software. As an example, the Oracle Mobile Agents product discussed previously supports both CDPD and Shiva PPP. However, no client software we have seen claims to migrate seamlessly among the different wireless network classes.

At some level, interoperability among the various network classes can be pro- vided by adopting popular communications standards. For example, those client applications developed to exploit TCP/IP in wired networks can inter- operate without change in the wireless domain if some variant of TCP/IP is offered as a service, e.g. lETF's Mobile IP. However, the quality of service that is provided by this approach may not be transparent, or even acceptable. In addition, it remains the client's responsibility to transfer among the various competing network services.

The heterogeneity of the existing and emerging wireless network protocols poses not only a need for interoperability, but also a stringent quality of service requirements. This is because the inherent unreliability and bandwidth limitation largely varies from one network to the other, leading to rapid fluctuations in the quality of the provided services. Recent research efforts proposed extensions to formal open systems standards.

The wireless application protocol (WAP) standard currently being developed by the WAP forum group offers an OSI-like protocol stack for inter operability of different wireless networks. The WAP stack allows applications to register interest in quality of service events and thresholds (QoS). This, in turn, allows the application to be mobility-aware and adaptable to changes in the environment. The WAP stack also provides negotiation protocols between producers and consumers of data to optimize the necessary level of data presentation based on the nature of data, the current wireless network between the source and the destination, and the capabilities of the destination device. Content negotiation should play a major role in maintaining QoS across heterogeneous networks.

Another proposal in extends the ISO Reference Model for Open Distributed Processing (RM-ODP) so that clients are able to adapt to the variation of the network service they encounter. Under this proposal, applications will have to be mobility-aware, but nonetheless will be able to maintain the required QoS.

10. Explain shortcomings and limitations (6)

Mobility-support software and products that are commercially available today leaves much to be desired in terms of functionality, performance, portability, and interoperability. The continuous decline in wireless communication cost and the recent convergence towards a truly global and standard communication system will help reduce the business risk associated with investments in this software market. This will be true for giant software vendors as well as small startup companies. Several limitations and shortcomings of existing products are summarized below.

- **Translucent Overlays.** None of the product offerings or announced plans for products that we have seen have included the vision of a translucent client context for exploiting nomadic applications. Each client or middle- ware offering is tailored to a specific kind of network service and assumes the client will manage its transition from one network class to another as the mobile platform roams about.

- **Multi-Database Access.** DBMS vendors such as Oracle and Sybase which are announcing mobile client products are providing connection ser- vices only to their proprietary DBMS product. This trend mirrors the path that these vendors have adopted for the fixed

network environment. Products which have emerged for the wired environment which make it possible or a client to interact with a variety of vendor DBMS, e.g. ODBC inter-faces, will not automatically extend service into the wireless domain. This failure occurs because current solutions that couple mobile clients with server DBMS include insertion of vendor-specific agents on both ends of the wireless connection to mediate wireless artifacts. Thus, a Mobile Oracle client can only talk to those servers that are serviced by the Mobile Oracle server agents .

- **Mobile Transactions.** We have seen no product that addresses issues of transaction management in the mobile environment. The concept of a mo- bile transaction, where the locus of control of the transaction is maintained by the mobile user, remains as a research idea needy of commercialization. BEA Systems markets a variety of products that couple wired clients with a variety of TP monitors, and implicitly to any of the DBMS products those TP monitors serve. If mobile transaction products will ever be made available, they will be offered by companies such as BEA Systems and Transarc.
- **Workflows.** Workflow products lag database access products in their migration to wireless and mobile environments. We have seen no workflow products targeted for the mobile domain in our literature searches.
- **Location Dependent Services.** None of the server or client software packages targeted for the wireless domain claim to offer location dependent services. This important class of services will be essential at both ends of the wireless client-server communications link.