

Openness in AXE

Einar Wennmyr

In recent years, the term open system has become a catch phrase in the data and telecommunications industries. Open systems are expected to improve time to market and make it easier for operators to introduce new features that will help them to attract and retain their subscribers. As competition in the market stiffens, it matters more and more who is first to market with new applications.

In this article, the author describes various attributes of open systems, and to what extent Ericsson has introduced them into AXE. He also describes the added value that is being integrated into AXE, extending its competitive edge even further.

What is openness?

A distinction can be made between external openness to a node (network openness) and openness within a node (system openness).

Network openness

Network openness signifies the ability to interoperate with other nodes in different networks. In this context, the AXE system has always been open—it supports numerous protocol standards and market variants, and can interoperate with any node implemented on another system platform, provided that node supports the same protocols. Examples of protocols currently supported by AXE are:

- channel-associated signaling (CAS);
- common channel signaling (CCS) and applications on top of it, such as ISDN signaling user part (ISUP), the mobile application protocol (MAP), and so on;

- network element management (NEM) protocols such as Telnet, file transfer protocol (FTP), file transfer, access and management (FTAM), and so on;
- asynchronous transfer mode (ATM) protocols; and
- Internet protocols (IP).

Obviously, AXE is also continually being updated to accommodate new standards. Some examples include the implementation of the bearer independent call control (BICC) protocol, the media gateway control protocol (H.248), and the common object request broker architecture (CORBA), which can be used for implementing Ericsson's integration reference points (IRP) for operation and maintenance (O&M).

System openness

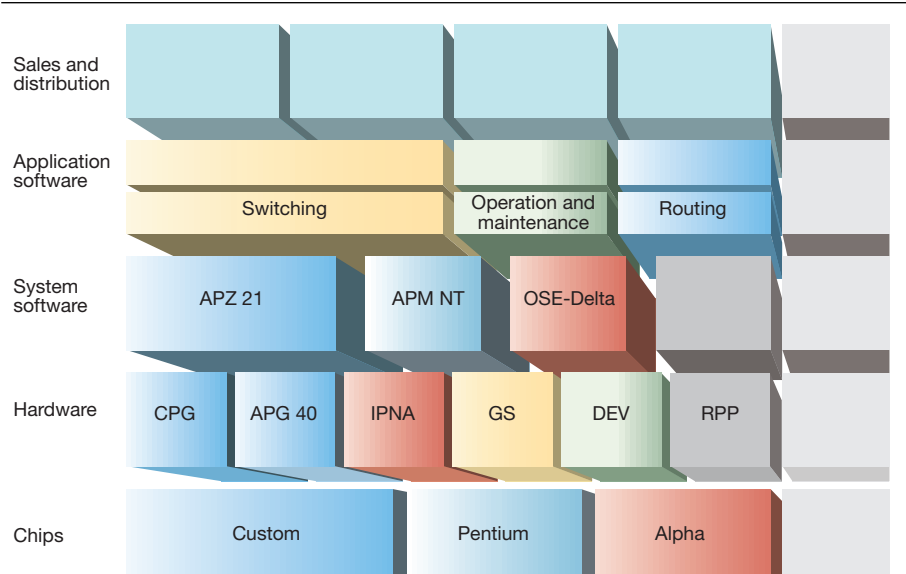
System openness signifies the use of standard, commercially available components to build the AXE system platform. More specifically, system openness is attained through the use of

- commercial hardware components (sub-racks, boards, chipsets);
- standard hardware building practices and buses;
- standard programming languages using standard software-development tools—software can thus be ported to different hardware and operating systems; and
- commercially available software components and interfaces.

In the past, AXE was considered a proprietary system with few standard components. Today, however, this description of AXE no longer applies, since standard components are being introduced at an increasingly accelerated pace. By using standard components to build their systems, companies like Ericsson can concentrate on their core business and still benefit from technological advances in other segments. What is more, they can achieve shorter time to market (TTM). The sourcing of components should be managed with care, however. And in some cases, sourcing might not be the appropriate or viable alternative:

- Satisfactory commercial components might not be available on the market.
- It might be more profitable to develop and produce components in-house—even when similar components are available on the market.
- Proprietary, in-house designs might be needed to remain competitive.
- Some "open" components lack the interfaces that are needed for integration.

Figure 1
System unbundling in AXE.



Components in AXE

Figure 1 gives a component view of AXE. The goal is to unbundle the system into a layered set of cooperating (interoperating) components—the trend we are seeing in the market is for specialization in horizontal product layers. By pursuing this approach, Ericsson can

- take full advantage of commercially available technology;
- reuse existing components and reduce the time it takes to introduce new products; and
- separate software functions from the hardware implementation.

Figure 2 gives an overview of the evolving architecture of the AXE system. Several of the products shown are described in this article.

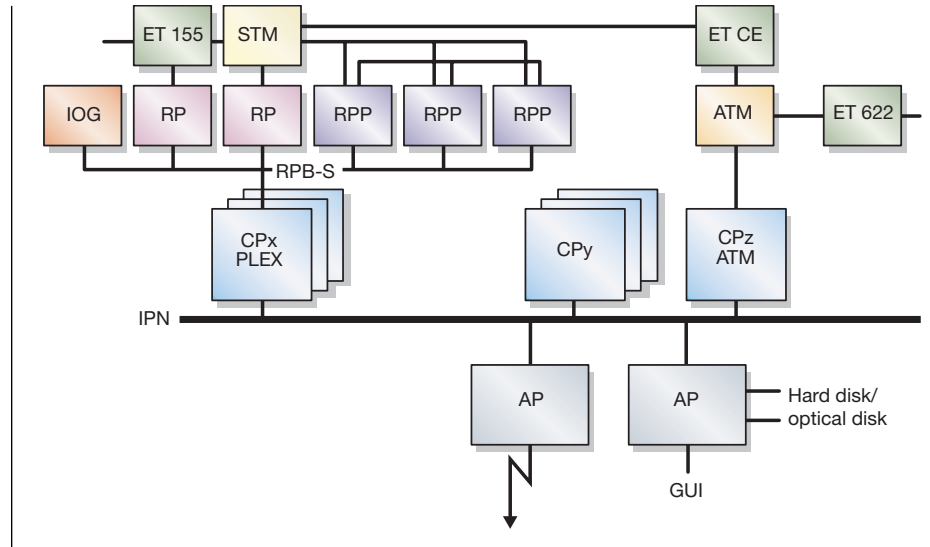


Figure 2
AXE system architecture evolution.

The adjunct processor platform

The adjunct processor (AP) is one of the first applications in AXE to use large building blocks from a commercial vendor.¹ The AP complements the central and regional

processors with the following type of functionality:

- input-output (I/O) functions (a file system, storage, formatting and output of call data records for charging, and storage

BOX A, ABBREVIATIONS

AM	Application module	GDDM-H	Generic device and datacom magazine, half-height	OMG	Object Management Group
AOT	Ahead of time	GPRS	General packet radio service	ORB	Object request broker
AP	Adjunct processor	GSM	Global system for mobile communication	OSS	Operations support system
APIO	Adjunct processor input and output	HDLC	High-level data link control	PAL	Privileged architecture library
APSI	Application platform service interface	IDL	Interface description language	PCI	Peripheral component interconnect
ASA	Assembly statements	IIOP	Internet inter-ORB protocol	PCU	Packet control unit
ASIC	Application-specific integrated circuit	IN	Intelligent network	PMC	PCI mezzanine card
ATM	Asynchronous transfer mode	I/O	Input-output	RMP	Resource module platform
BICC	Bearer independent call control	IP	Internet protocol	RNC	Radio network control
BMC	Base management controller	IPC	Interprocessor communication	RNS	Radio network server
BSC	Base station controller	IPN	Interplatform network	RP	Regional processor
CAS	Channel associated signaling	IPNA	IPN adapter	RPC	Remote procedure call
CCS	Common channel signaling	IPU	Instruction processing unit	RPH	RP handler
CORBA	Common object request broker architecture	IRP	Integration reference point	RPHM	RPH magazine
CP	Central processor	ISDN	Integrated services digital network	RPHMI	RPHM interface
CSH	Connection service handler	ISUP	ISDN signaling user part	RPP	Regional processor platform
DAT	Digital audio tape	JIT	Just in time	SMP	Symmetric multiprocessor
DDS	Digital data storage	LAN	Local area network	SPU	Signal processing unit
DMA	Direct memory access	MAP	Mobile application part	STOC	Signaling terminal open communication
DSP	Digital signal processor	MAU	Maintenance unit	STS	Statistics service
ENGINE	Next-generation switch	MML	Man-machine language	TCP	Transmission control protocol
EPSP	Ethernet packet switch board	MSCS	Microsoft cluster server	TDM	Time-division multiplexing
ET	Exchange terminal	MTBF	Mean time between failures	TRH	Transaction record handler
ETC	Exchange terminal circuit	NEM	Network element management	TTM	Time to market
ETCE	Exchange terminal circuit emulation	NSP	Next-generation switch platform	UDP	User datagram protocol
FOS	Formatting and output service	NT	Network termination	UMTS	Universal mobile telecommunications system
FTP	File transfer protocol	O&M	Operation and maintenance	UPB	Update board
FTAM	File transfer, access and management	OCITS	Open communication Internet transport service	VM	Virtual machine
				XSS	Existing source system

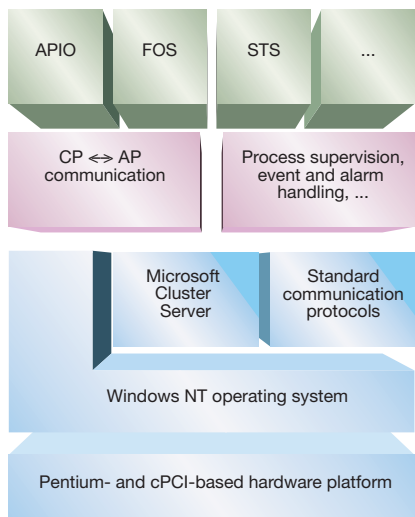


Figure 3
Component view of the adjunct processor (AP).

of counters for statistics and traffic measurements);

- boot server for the AXE central processor;
- support for man-machine communication; and
- connectivity to operations support systems (OSS) including protocols for file transfer, message transfer and operator access.

The next-generation computer platform for the adjunct processor functions is called APG 40. Like its predecessor (the APG 30), the APG 40 will be based on open, commercial products.

Focus of development

Ericsson has focused its development efforts on the interface to the central processor and on improving operator support for handling the adjunct processor (Figure 3). In particular, developers have worked on improving or adding the following functionality:

- CP-AP heartbeat;
- system calendar synchronization between the AP and CP;
- system monitoring and diagnostics—hardware events, system log analysis, Microsoft Cluster Server (MSCS) process supervision;
- event handling—every reported event is supplemented with a date and time stamp

and stored persistently. Events are also forwarded for alarm processing;

- alarm handling;
- support for upgrading software in the AP—high-level commands enable operators to initiate a software upgrade, complete it, and if necessary, to fall back on old software;
- authorization; and
- audit log—all commands issued to AXE and all man-machine-language (MML) printouts are logged, as are all state changes in the network termination (NT) part.

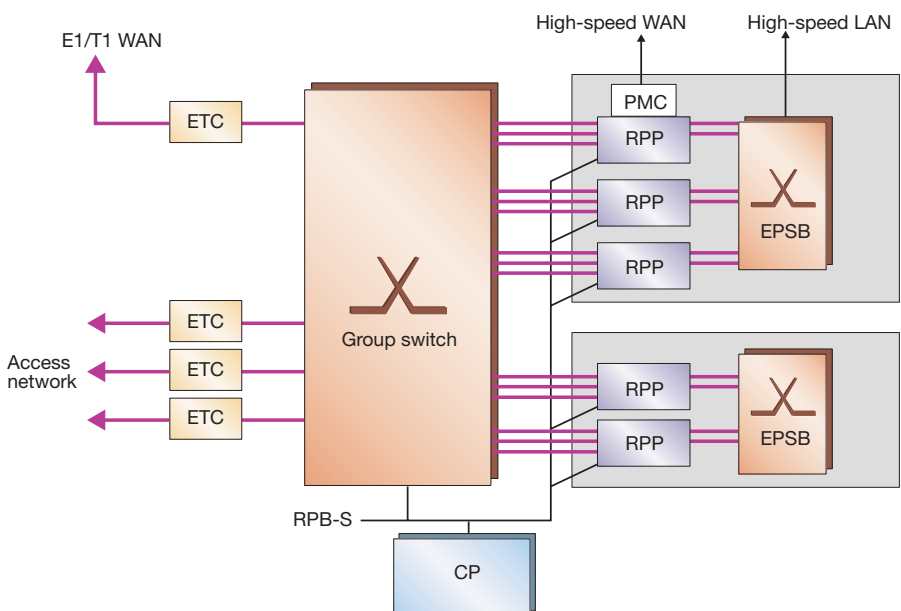
The CP-AP communication service, statistics service (STS), formatting and output service (FOS) for charging data, and the AP input-output (APIO) for backup are each proprietary solutions.

Sourced products

To a large extent, the platform requirements are fulfilled by the following commercial products:

- A commercial processor platform with sufficient capacity and I/O devices. This large building block, which has been sourced from an external vendor, is based on Pentium II processors. It contains one or three 18 Gbyte hard disks (depending on the configuration), digital audio tape (DAT) or digital data storage (DDS) streaming tape drivers, 100 Mbit/s Ethernet ports, PCI mezzanine card (PMC) modules for communication support, and power supply.
- Microsoft Windows NT operating system. Many third-party software vendors provide products for this operating system.
- Microsoft Cluster Server (MSCS) software for high-availability servers. The MSCS currently supports a two-node cluster, defining the interdependency between, and handling the restarting of, services, thus making the entire system fail-safe.
- Disk management and data handling—Volume Manager supports online management, reconfiguration, and fast-failure recovery; Diskkeeper handles defragmentation; WinZip handles the compression of data; backup software handles backup and disk cloning.
- Connectivity to OSS—several commercial protocols are used, including the remote procedure call (RPC), the transmission control protocol/Internet protocol (TCP/IP), and the Internet inter-ORB protocol (IIOP).

Figure 4
AXE architecture with the regional processor with PCI bus (RPP).



Data communication interfaces in AXE

Other open products already included in the AXE system are the regional processor with PCI bus (RPP) and the Ethernet packet switch board (EPSB). The RPP, which supports data communication-related telecommunication applications, offers a range of open hardware interfaces, software applications, and a complete development environment. One of the first applications to use the RPP and EPSB is the packet control unit (PCU) in the base station controller (BSC). The PCU provides support for general packet radio service (GPRS) in GSM networks.²

RPP

The RPP, which is situated as an ordinary regional processor (RP) in the AXE architecture (Figure 4), extends the functionality of a traditional regional processor by supporting several protocols and physical links and by providing support via duplicated Ethernet for distributing a datacom application over multiple RPPs. Both sourced and proprietary components have been used in the RPP (Figure 5).

The RPP CPU is based on commercially available processors (currently a 333 MHz PowerPC). Greater capacity is achieved by upgrading the RPP CPU as new processor generations are introduced. The basic processor operating system is OSE Delta. Some additions have been introduced on top of the OS, yielding a product called RTOS (real-time OS). Standard programming languages, such as C and C++, can be used to build datacom applications on top of RTOS. This makes it easy to source applications and to find development resources in the future.

The I/O board contains standard Ethernet interfaces and custom interfaces to AXE (to the group switch and the RP bus). Standard networking interfaces can also be included on externally sourced PMC modules on the PMC carrier board. Various commercially available digital signal processors (DSP) can be included for bit-stream-oriented protocols. Most protocol support for modem, fax, V.110, voice coding, echo cancellation, ATM, IP (TCP/IP) and high-level data link control (HDLC) are provided through sourced products.

O&M capabilities are primarily based on extensions of traditional AXE functionality in the central processor. However, some maintenance can be performed via TCP/IP.

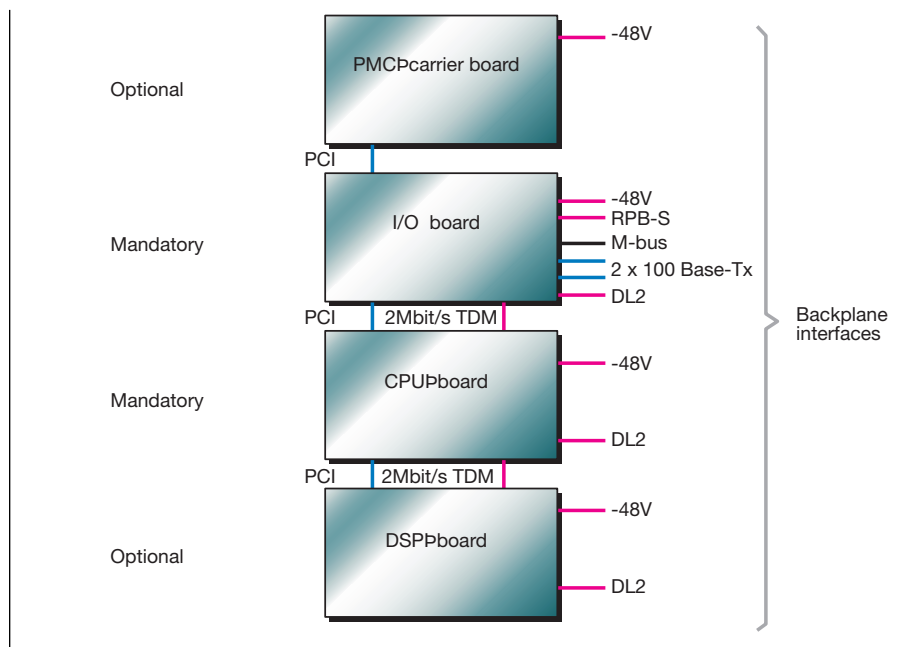


Figure 5
Structure of the regional processor with PCI bus (RPP).

The different components within the RPP are connected internally via a PCI bus.

Ethernet packet switch board

The Ethernet packet switch board and Ethernet connectors in the backplane of the housing subrack (GDDM-H) have been introduced to support applications that are distributed over multiple RPPs. Ethernet was chosen because it is the industry standard for local area network (LAN) communication.

The Ethernet switch, which is built on a single board using sourced switch circuits, gives interworking functionality to separate RPPs in the same subrack, in different sub-racks, or to external equipment. The Ethernet switch contains thirteen 10 Mbit/s ports and three 100 Mbit/s ports. The 10 Mbit/s ports and one of the 100 Mbit/s ports are available in the backplane. The remaining two 100 Mbit/s ports are available at the front of the board. The switch is duplicated for redundancy.

Interplatform network

The interplatform network (IPN) is a product that will introduce a 100 Mbit/s Ether-

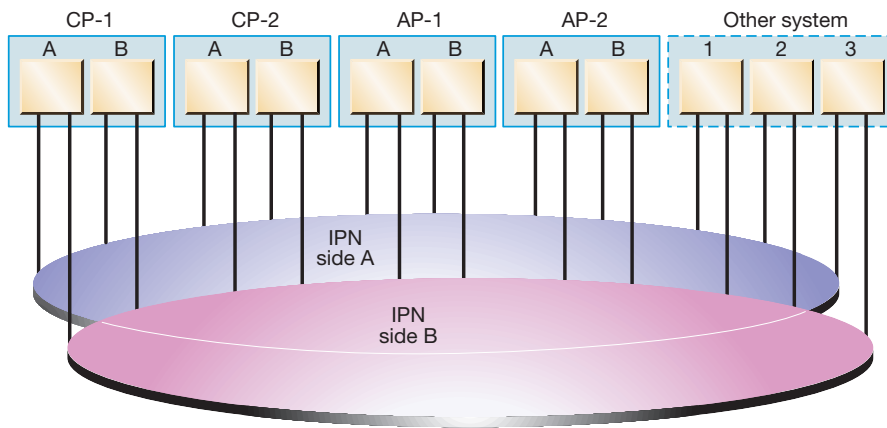


Figure 6
The interplatform network (IPN).

net interface into the AXE CP. An important reason for choosing the 100 Mbit/s Ethernet interface is that it is an industry standard supported by many commercial products, and as such, simplifies interconnecting AXE with other commercial products. As shown in Figure 6, the IPN has many applications:

- CP-AP communication—improved bandwidth can decrease the time required for backup and reloading software;
- communication between the AXE CP and other platforms (such as the next-generation switch platform, NSP, or AXD 301) for hybrid nodes or for moving functionality from AXE, in order to gain an increase in capacity^{3,4} and
- CP-CP communication within a cluster—this alternative is currently being investigated as an option for providing greater capacity in large nodes.

The IPN will first be deployed in AXE-based nodes of the universal mobile telecommunications system (UMTS).

Benefits of the IPN

IPN functionally replaces the signaling terminal for open communication (STOC), which currently terminates Ethernet in AXE. Note: the STOC solely terminates 10 Mbit/s Ethernet. Other benefits of the IPN are increased bandwidth for data transfer out of the CP, improved performance for sending large volumes of data to and from the CP, and improved latency and throughput of messages.

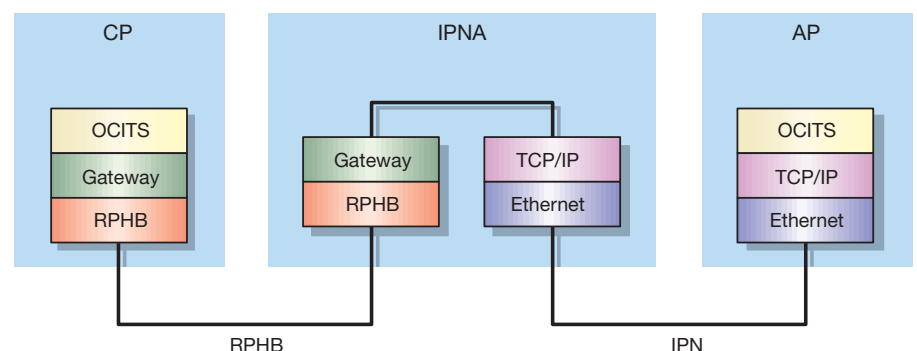
Increased bandwidth

Greater bandwidth for transferring data out of the CP is obtained by connecting the IPN to the RPH bus instead of to the RP bus. The IPN can potentially use the entire RPH bus bandwidth of 160 Mbit/s instead of being limited to the RP bus bandwidth of 10 Mbit/s—notwithstanding, this bandwidth must be shared with the RP buses needed for ordinary RPs. The distribution of bandwidth can be configured and set by operator commands. In the next-generation processor, Ethernet will be terminated directly on a CP board instead of via the RPH bus.

Improved performance

Support for soft, direct-memory access (DMA) in the CP microprogram improves performance when large volumes of data are sent to and from the central processor (64 Kbytes of data can be transferred at one time). This makes for the efficient transfer of communication buffers and dynamic buffers in AXE to the IPN adapter (IPNA).

Figure 7
Protocol stacks in the IPNA.



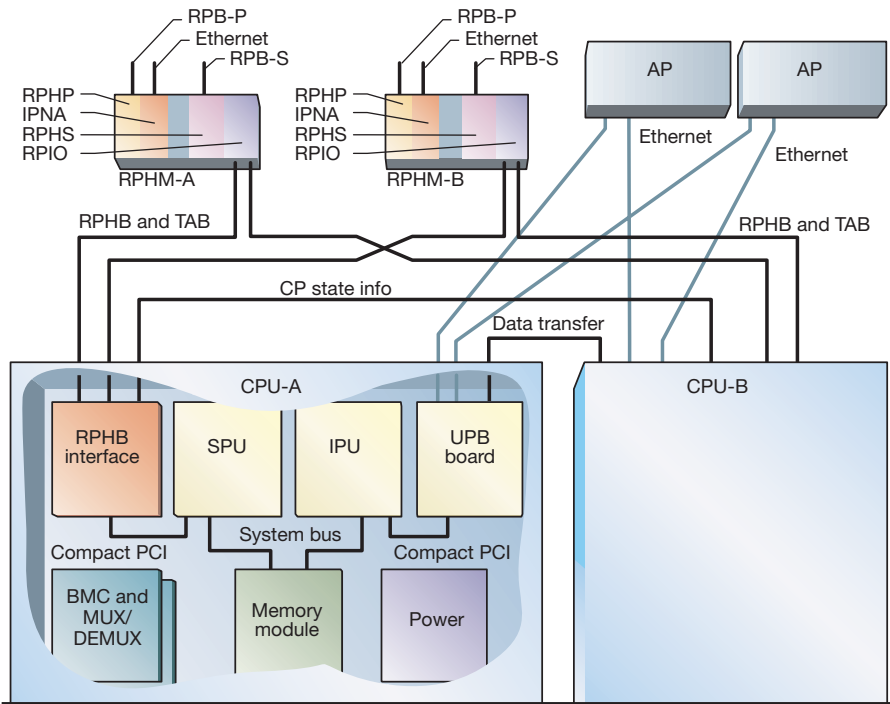


Figure 8
The components and data flow in the APZ 212 40 CP.

Improved latency and throughput

Latency and throughput of messages are especially important when the IPN is used for traffic applications. These two characteristics will be improved using a commercial, high-capacity processor chip that can be upgraded as new generations become available.

IPNA design

The IPN connects to the AXE CP through the IPN adapter, which will be based on

- a standard, commercial processor with the OSE Delta operating system; and
- a commercial Ethernet chip.

Proprietary ASICs will be developed to support the physical interface to the RPH bus (Figure 7).

The IPNA terminates the TCP/IP protocol and includes gateway functionality for converting TCP/IP into internal AXE protocols. The file transfer protocol will be used for transferring data to the AP. Other standard protocols can be provided if necessary. The open communication Internet transport service (OCITS) layer provides additional support for the application level protocol, such as addressing, message fragmentation and recombination.

APZ 212 40

Ericsson's APZ 212 40, which is the next-generation central processor in AXE, will be based on a commercial processor and not on in-house processor chips. This decision was based on a benchmark test of several commercial processors. A compiler prototype has been developed for the most promising candidate, in order to test the execution of various telecommunications applications. The results compare favorably to those of a traditional CP and indicate that the commercial processor delivers roughly the same capacity as can be derived from a next-generation, in-house solution.

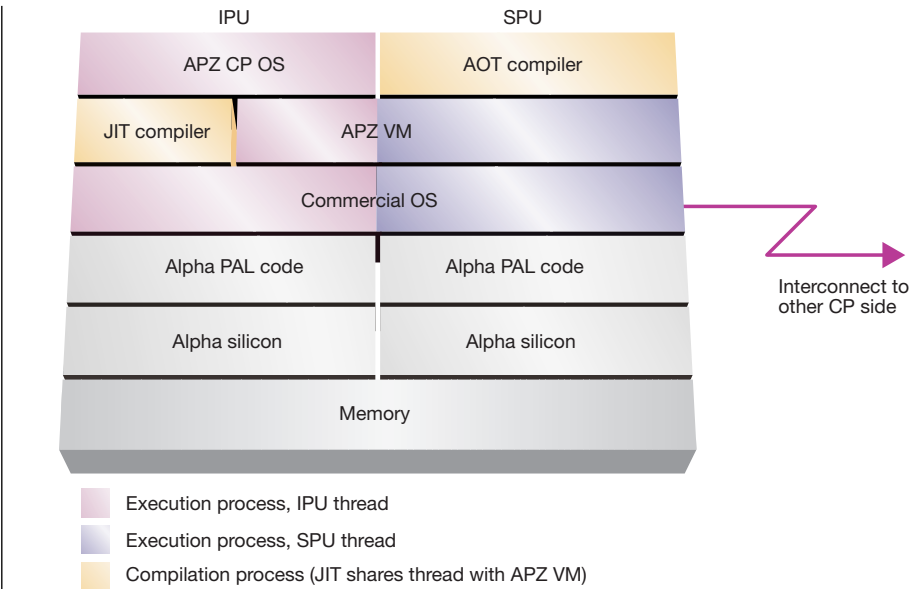
Interfacing the processor to the AXE architecture

Ericsson will develop several hardware and software components (Figure 8)

- to guarantee fault tolerance (hardware and software);
- to provide support for upgrading software while traffic is being handled in the processor; and
- to provide support for existing interfaces to other parts of the AXE system.

The CPU subrack will contain two CPU

Figure 9
CP layering.



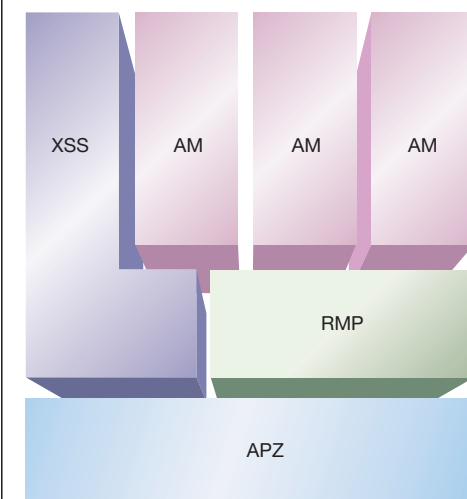
boards (the instruction processing unit, IPU, and the signal processing unit, SPU) and up to 16 Gbytes of memory. It will also contain maintenance unit (MAU) functionality and an interface board to the RP bus handlers for interfacing existing RPs and APT hardware.

Fault tolerance will be supported through a duplicated CPU. However, in contrast to previous generations, the execution side and the standby side of the APZ 212 40 will not operate in parallel-synchronous mode; in-

stead, a cold-standby processor will be used. This means that during ordinary traffic, the standby CPU will be loaded with the latest version of the software and configuration data, but will not be updated with traffic data. If a non-recoverable hardware failure occurs, the standby processor will take over execution by reloading and restarting. Because the mean time between failures (MTBF) is very long for modern processors, the MTBF goals for AXE can still be met.

The update board (UPB) will implement the physical link between the two CP sides through 1 Gbit/s Ethernet. In addition, 100 Mbit/s Ethernet will be supported for AP communication (backup, reload, charging data, and so on).

Figure 10
System modules.



Equipment practice

The APZ 212 40 will fit into the standard BYB 501 equipment practice. Two CP sides fit into a standard-width subrack; each CP side will be built using the standard, compact PCI 6U equipment practice. Maintenance buses will be based on the I2C standard and kept independent of the cPCI bus.

CP structure

The memory chip, processor chip, privileged architecture library (PAL) code, and operating system (Compaq Tru64 UNIX, a commercial 64-bit version of UNIX) will be sourced from external vendors.

Ericsson will provide the components and

functionality indicated in the shaded areas of Figure 9. The APZ virtual machine (VM), which is implemented in C++, provides the PLEX execution environment as well as middleware support for function changes and switching between CP sides (in case of failure). The APZ VM executes the PLEX software in a process with two threads: the IPU thread and the SPU thread. The process provides a scheduler, job buffer handling, communication support (for sending and receiving signals), and support for error handling, recovery, and a run-time log.

The traditional way of compiling code is to generate machine code directly for the target machine. However, the APZ 212 40 will not follow this approach, since it is not compatible with assembler additions to legacy applications. Instead, the PLEX code will be compiled to assembler instructions. This makes the incorporation of existing assembler additions transparent. The gains in handling will far outweigh the minor loss in performance.

Communication services

The application modularity architecture was introduced to define the AXE 106 system (Figure 10). The basic principle was to divide the AXE system into different types of system module: application modules (AM), resource module platform (RMP), existing source system (XSS) and the APZ computing platform.

An important feature of this architecture is that all inter-application-module communication takes place indirectly via communication services within the RMP (APC, CLMT, and OBMAN function blocks). The logical names of protocols and services are used at the application level, which the RMP resolves to physical addresses. This is similar in concept to an object request broker (ORB).

New kind of openness

With this architecture, one or more of the application modules can, in principle, execute on a physically separate platform (AXE or other), since the physical distribution is hidden by the RMP. This possibility has already been exploited for charging information and statistics; that is, the FOS and STS have been moved to the AP platform. This points to a new kind of openness in AXE that can be used for

- boosting capacity—some application modules require huge amounts of capac-

ity. Therefore, more capacity will become available in AXE if the AMs can be executed on separate platforms. Some examples where this approach has been pursued are: moving IN functionality and ISUP protocol termination to separate AXE for ENGINE, moving the VLR functionality from AXE to NSP for TDMA, and finally investigating whether IN functionality can be ported to the NSP in a prototype.

- step-by-step migration—if an application module (or part of one) needs to be extensively redesigned for other purposes, it might be beneficial to redesign it to run on a new platform using a commercial development environment and commercial implementation languages. An example where this approach has been pursued is from TDMA, where existing MSC functionality in AXE, which handles radio network control (RNC), is being redesigned on the NSP.
- accessing new functionality—instead of developing new functionality in AXE, it might be more cost-effective to combine it with functionality already available in another platform. For example, the next-generation switch (ENGINE) will provide ATM capabilities to a transit node by combining AXE with AXD, thus using ATM functionality supported in the AXD 301.
- reuse—instead of redeveloping all existing AXE functionality on a new platform it might be more cost-effective to combine some of it with new functionality on another platform. An example from TDMA is where existing maintenance functionality of the old radio network is kept in AXE while control functions are moved to the NSP.

To support cases that involve distributed execution of traffic AMs, new communication services will have to be introduced, since present-day communication services indirectly assume that the AMs execute on the same platform. For instance, no error cases are handled if the other platform is not available or if the communication link between platforms fails.

New communication services

TRH

The transaction record handler (TRH) protocol—which was developed to support the communication needs of the control interface between AXE and AXD 301 in

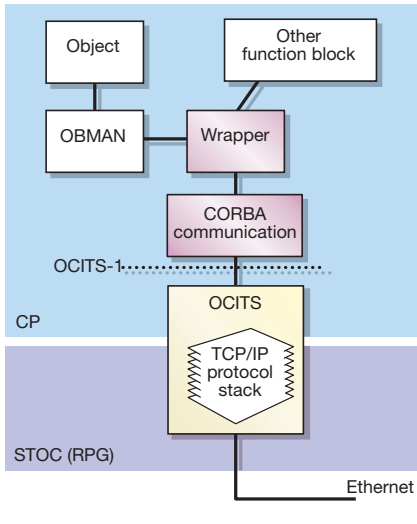


Figure 11
Implementation structure of the CORBA prototype in AXE.

ENGINE—adds a proprietary layer on top of TCP/IP and runs over the STOC hardware.

CORBA

The common object request broker architecture (a standard promoted by the Object Management Group, OMG) is frequently used for O&M. Several prototypes have been developed to test its usability. One prototype implements a CORBA communication service (a PLEX-ORB). Wrappers that provide service interfaces within AXE to applications executing on other platforms are also implemented (Figure 11). The wrappers use the PLEX-ORB for communication. The interface between platforms has been specified using the interface description language (IDL) defined by OMG.

This implementation proves that O&M functionality can be moved to another platform, using IIOP to communicate with the traffic applications that remain in AXE. As an added advantage, it is easy to provide a Web-based O&M interface to applications that use an object request broker.

CSH

A main drawback of distributing a traffic application over more than one platform is that a lot of CP capacity is required to send messages and data between the platforms. A third alternative being developed, called the connection service handler (CSH), aims to minimize CP load during communication from AXE.

The CSH communication service will be based on the proprietary inter-processor communication (IPC) protocol used within an NSP cluster. The IPC can run on raw Ethernet (and on top of other standard protocols, such as the user datagram protocol, UDP, or ATM) with low overhead; the IPC header size is only 7 Words compared to 16 Words in TCP/IP. Moreover, IPC is robust and supports efficient addressing. In its first release, the IPC protocol will be terminat-

ed on the IPNA, in order to offload the central processor.

Summary of communication services

The PLEX interfaces to different communication services (Figure 12) are similar and typically provide support for

- defining the messages in a protocol (used to support byte ordering and transport format of data);
- setting up communication connections;
- sending and receiving messages (simple data, communication buffers and message buffers);
- tearing down connections;
- reporting the failure of communication peers and handling diverse AXE recovery scenarios (Forlopp, minor and major restart); and
- handling duplicated links and, if a failure occurs, switching from the failing link to the working link.

Conclusion

The adjunct processor platform is chiefly made up of sourced components. Ericsson provides additional value to the adjunct processor platform in AXE by adding support for supervision, alarm handling, event handling, upgrading software, audit log, and access authorization, as well as support for interfacing the CP. Since the availability and robustness requirements for the AP are less severe than for the traffic platform, Ericsson can source components comparatively high in the layered component model.

The RPP and Ethernet packet switch board are composed of several standard components. Ericsson has also, among other things, added interfaces to AXE, and O&M functions according to the AXE standard. Because the RPP and EPSB support industry-standard interfaces, other commercial products can easily be introduced as they become needed.

TRADEMARKS

Diskkeeper is a registered trademark of Executive Software.

Pentium is a registered trademark of Intel Corporation.

Sun Enterprise Volume Manager is a registered trademark owned by Sun Microsystems Inc. in the United States and other countries.

Windows NT is a registered trademark of Microsoft Corporation.

WinZip is a registered trademark of Nico Mak Computing, Inc.

The interplatform network adapter will add the industry-standard 100 Mbit/s Ethernet interface to AXE CP, thereby improving reload and backup times and providing support for interworking with traffic applications on other platforms. The IPN adapter is composed of several commercial components, such as processor chips, operating system, and protocol stacks.

Ericsson will introduce several commercial products, interfaces, and key software components into the next generation APZ. In-house hardware and software components will also be introduced to enable the next-generation CPU to execute legacy applications, to adhere to ISP requirements, and to interface legacy parts of the AXE system. Future technology that can be introduced into AXE includes larger symmetric multiprocessor (SMP) clusters and super-scalability.

The introduction of new communication services in AXE will make it possible to combine the functionality in AXE with that of other platforms, in order to support step-by-step migration and reuse. For example, the transaction record handler has been shown to work for ENGINE traffic applications, the IIOP works with O&M functions, and the connection service handler is being developed to work with traffic applications. The communication services described will either run on raw Ethernet or add functionality on top of TCP/IP or IIOP.

During the past several years, Ericsson's hardware and software developers have been using commercial components to build the AXE system. This process has been accelerated and, as this article shows, commercial products and interfaces will soon make up the core components of AXE. This means that Ericsson can fully focus its development efforts on providing telecommunications characteristics, such as high availability, robustness, fault tolerance, and capacity.

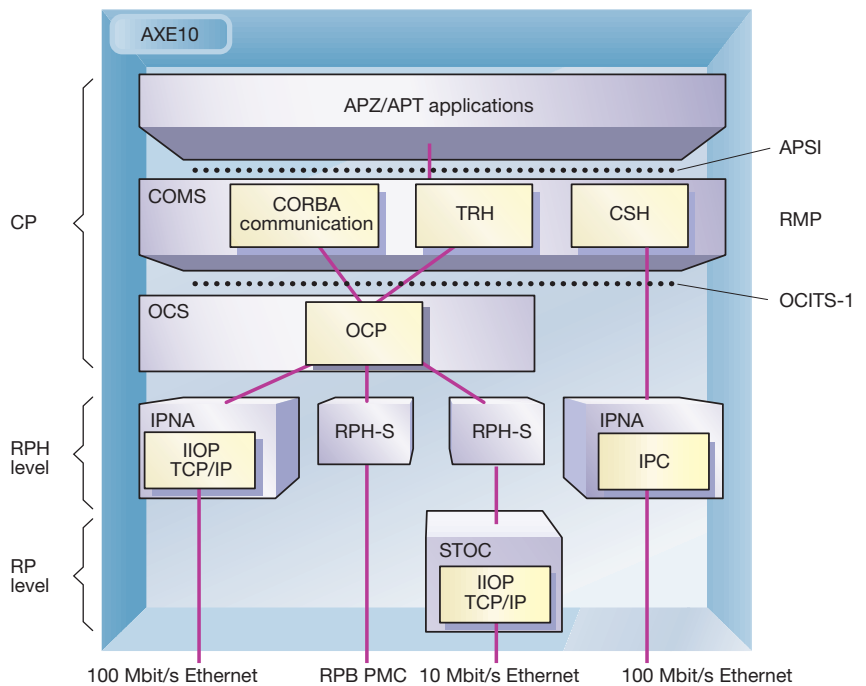


Figure 12
Overview of the communication services in AXE.

REFERENCES

- 1 Ericsson, M. and Korla, N.: Adjunct processor—A new AXE-integrated open-standard computer system for call-related data processor. Ericsson Review Vol.74 (1997): 2, pp. 82-90.
- 2 Granbohm, H. and Wiklund J.: GPRS—General packet radio service. Ericsson Review Vol. 76 (1999): 2, pp 82-88.
- 3 Blau, S. and Rooth, J.: AXD 301—A new generation ATM switching system. Ericsson Review Vol. 75 (1998):1, pp.10-18.
- 4 Hennert, L. and Larruy, A.: TelORB—The distributed communications operating system. Ericsson Review Vol. 76 (1999): 3, pp.156-167.