

Apple, SAA, and OfficeVision

Product Marketing

March, 1990



This paper outlines Apple Computer's position on IBM's Systems Application Architecture (SAA) and OfficeVision products. It is not a competitive analysis nor does it engage in gratuitous SAA bashing. Rather, in combination with other materials such as Macintosh Advantage, Market Intelligence Bulletins, and industry publications it forms the basis for sales strategy and marketing program development aimed at penetrating large IBM-oriented accounts.

This paper is intended for Apple eyes only and assumes a basic knowledge of IBM products and technologies.

Contents:

- Apple's position on IBM's SAA and OfficeVision
 - Recommended reading & education
 - Glossary
-

The Apple Strategy for IBM Integration

A key element of Apple's networking strategy has been Macintosh integration into the IBM environment. Our goal is to provide users with access to IBM applications and services and developers with a standard set of protocols, interfaces, and tools to enable the integration of the Macintosh with the IBM environment.

The introduction of Systems Application Architecture (SAA) by IBM in 1987 clearly defined our target environments as System/370 mainframes running VM and MVS, the AS/400 mid-range running OS/400, and the PS/2 running OS/2 Extended Edition. SAA specifies a common set of user interfaces, communications, and application programming interface standards across these environments, primarily to improve application consistency and portability across the product line. By including specific technologies under the SAA umbrella, IBM intends to support these technologies under each of the four operating system environments, and over the long term. This means that Apple products implementing certain SAA technologies not only extend the reach of the Macintosh user, but also protect investments by Apple and our customers over a wider range of systems and for a longer period time.

Our approach is to implement those SAA functions that enable Macintosh users to connect to IBM systems for data, service, and application access. This allows us to focus our development efforts on a limited set of products that are well integrated with the Macintosh while providing key networking functions to developers for the creation of Macintosh applications. The customer benefits from a standard set of integrated networking functions through greater consistency and

interoperability among networked Macintosh applications and services. The commercial developer builds better, network friendly software in a shorter time with Apple-supplied networking functions.

To achieve these goals, our development has so far been focused on the lower level networking functions - mainly physical connections and their associated protocols -- the "plumbing". Ultimately, these must be hidden from the user to enable the transparent access to remote information that has been a major Apple goal for several years. With the new N&C products introduced last year, these low level networking functions are now available.

Our next phase of IBM connectivity development will be focused in two major areas: on-going enhancements to the existing "plumbing" and a set of high level services, tools, and interfaces that will allow programmers to create integrated applications using IBM's own SNA and SAA standards. In both cases, Apple will implement key IBM networking functions under the Macintosh OS and A/UX™ and these networking functions will be based on standards that IBM has deemed "strategic" within Systems Application Architecture.

SAA is, to an extent, a "road map" for Apple and other vendors to achieve connectivity to IBM systems using IBM's own accepted standards, with the opportunity to provide users with unique and substantial Macintosh value-added in those environments.

Why SAA ?

Why did IBM create SAA ? While many forces were at work including internal IBM issues, customer requests, and technical considerations, the driving forces can be summarized as follows:

1. Resolve the multiple platform dilemma
 - IBM's product divisions developed products for their favorite design points, not according to customer needs.
 - Pressure from Digital increased dramatically due to their success in the mid-range with a single operating system.
2. Extend and leverage existing customer investments
 - Pressure from customers for non-disruptive growth by leveraging existing applications, programmers, and systems
 - Pressure from customers for greater program product synergy - less overlap, reduced learning curves, greater consistency.
3. Incorporate personal computers into mainstream business systems
 - The success of the IBM PC exceeded expectations, but internal power remained in mainframe hardware.

- A means to move more quickly to distributed processing

IBM's solution to these problems is SAA. IBM developed the four functional areas of SAA based on the above requirements. Common Communications Support to ensure consistent, peer-oriented protocol availability across platforms. Common Programming Interface to extend programming skills across systems and improve application portability. Common User Access to address the wildly variant user interfaces across, and within, IBM systems. Common Applications to provide a framework for a single user view of all systems and applications.

The reality, of course, does not match these lofty goals, at least for the near term and probably beyond. First and foremost, SAA solves a set of problems that are unique to IBM. An Apple response need only address networking and programming interface integration requirements - not how we will comply with IBM's application and user interface rules. We have not done so to date and there's no reason to start now.

IBM's SAA marketing strategy to customers is centered around:

- Protection of customer investments in people, applications, and systems
- Easy migration and horizontal growth
- Targeting large I/S organizations' managers and application programmers
- The transfer of technologies from the mainframe to AS/400 and PS/2.

The corresponding implicit messages to competitors are:

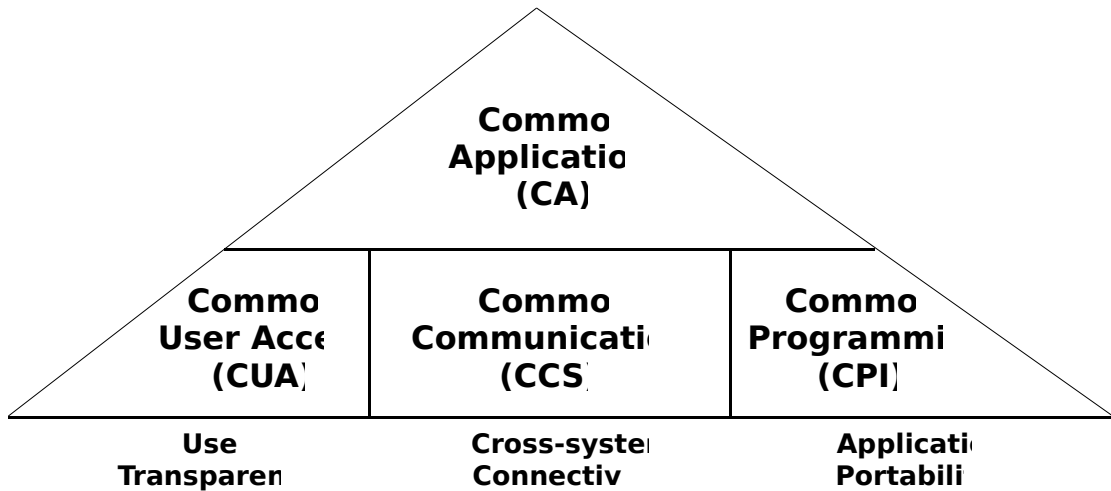
- IBM to control and keep corporate information
- Customer FUD (Fear, Uncertainty, Doubt) of SAA compliance checklist
 - competitors need a more cohesive IBM product line
- Published interfaces, but developing proprietary products
- Frequent SAA revisions = moving target
- Attractive third party platforms - a developer "road map"

SAA Compliance Strategy

A dissection of SAA reveals four distinct elements - communications, programming tools, user interfaces, and applications. IBM intends to provide products implementing these technologies across the four SAA operating environments to enable greater user and programmer consistency. Our goal is to comply with SAA in all areas that benefit our customers and complement the Macintosh; i.e. in the areas specifying the connection to and and transfer of **data** through communications links and programming interface standards. In general, we will support specific SAA technologies and standards that meet each of the following criteria:

- 1) complement an existing or emerging Macintosh product or technology
- 2) are strategic to existing and new Apple customers

SAA Components



Common Communications Support - CCS

Function: Aspires to provide *consistent communications* support by providing protocols for interconnection and data interchange among SAA systems. Consists of both SNA architectures and international standards.

Reality: This is the SAA element which offers the most complete support across the four software platforms and where IBM will achieve the broadest market acceptance. However, IBM's implementations still vary widely and not all of the CCS technologies are implemented consistently across the platforms.

Apple's Position: This is the most critical element of SAA that must be supported on the Macintosh. It specifies open technologies that will enable Macintosh users to connect to IBM systems to send and receive data, run applications, and access services. Our intention is to continue to support all data streams, application services, session services, network services, and data link controls of CCS as the market need for each evolves.

Status: Extensive support now available from Apple and third parties. See Appendix A for specific information.

Common Programming Interface (CPI)

Function: A set of *languages* and *programming interfaces* intended to improve application portability, leverage existing programming skills, and establish a standard application development platform.

Reality: While IBM's goal is to eventually implement these standards across the SAA platforms, their implementations still vary widely. This is the area where IBM faces the biggest challenge given that consistent support is far from complete and IBM's customers have large investments in existing application enabling and development products. However, after CCS, it is the aspect of SAA that customers believe holds the most value over the long term.

Apple's Position: The languages supported under CPI are, for the most part, host development oriented and are not applicable to the Macintosh. The most popular language for microcomputer development is "C", which is supported extensively by Apple and third parties.

The programming interfaces will be supported to the extent that they complement the Macintosh and provided they have achieved market acceptance. Of the five CPI standards - Database, Communications, Query, Dialog, and Presentation, only the Database and Communications are relevant for Macintosh developers. The database interface is SAA SQL and the communications interface is CPI-C (the SAA version of APPC). Direct alternatives for the Query, Dialog, and Presentation interfaces are provided as integral components of the Macintosh architecture in ROM (dialog and presentation) or as separate products (CL/1 for database query) (see Appendix A for details).

The position of Macintosh with regard to existing or new CPI technologies and products can be classified as follows:

1. part of CPI is designed for a mainframe environment

Not broadly applicable to the Macintosh (*e.g., REXX, CSP, Cobol*)

2. part of CPI specifies functions provided by the Macintosh

Products will not be supported by Apple (*e.g., Query, Presentation, Dialog interfaces*).

3. part of CPI specifies functions that complement the Macintosh in IBM environments

May be implemented by Apple or third parties as market requirements appear (*e.g., Database, Communications, and Repository interfaces*)

We will be tracking the implementation and market acceptance of the CPI communications interface – CPI-C, and may implement it in the future as an

enhancement to our APPC products. For at least the next year though, customers will still be struggling to understand how to implement cooperative processing and which applications are best suited to it. Its still a 3270 world, but having a strong APPC product is a strong statement to our customers and positions us well for the coming wave of cooperative applications built on emerging peer-oriented SNA products such as APPC, Low Entry Networking (LEN/APPN), and VTAM 3.3.

CL/1 is Apple's strategic data base access product that provides the functionality of the CPI database interface - SQL, and more. CL/1 is designed to provide unified access to multiple data base environments from within Macintosh applications. In addition to supporting CL/1, the Database Access Manager (DAM) of System 7.0 will offer third party developers a means of implementing their own SQL dialect to communicate directly to target databases.

Status: CPI-C support will be investigated as the market need increases. CL/1 offers the Macintosh user or developer a platform for multivendor data base access corresponding to the SAA CPI Database and Query programming interfaces. Refer to OASIS and other Apple and industry materials regarding Macintosh differentiation and advantages over the CPI Dialog and Presentation interfaces as implemented under IBM's OS/2 Extended Edition operating system.

Common User Access (CUA)

Function: Aims to establish a set of rules, guidelines, and technologies for use during *user interface design*. The primary goal is to standardize the semantics, syntax, and physical aspect of systems and application interfaces across the SAA platforms. IBM intends to enforce/encourage application developers to conform by implementing key user interface functions as part of the CPI.

Reality: While IBM has set for itself an ambitious goal, constraints based on the diverse SAA operating systems will limit significant progress in this area outside the OS/2 environment. Interfaces on the System/370 hosts are still predominantly character oriented with limited graphic support via GDDM. The PS/2 uses OS/2 Presentation Manager, which is IBM's real target for advances in graphical user interfaces. Additional confusion abounds in the PS/2 and PC worlds with a variety of operating environments and user interface tools to choose from: e.g., OS/2 Presentation Manager, OS/2 Standard Edition vs. OS/2 Extended Edition, MS-DOS/Microsoft Windows (several versions of each), and Metaphor.

Apple's Position: This is one of the key competitive advantages of the Macintosh. There is only one user interface standard and its use is enforced by being part of the hardware and software fabric of the Macintosh. Applications are thus able to provide a consistent interface that is easy to learn and use and is increasingly being applied to various host environments - ranging from host front ends to cooperative processing peers. Hence, the user interface problems that IBM will attempt to solve

over the long term across their systems can be solved today by applying the Macintosh interface to access applications, services, and data across IBM's "strategic" platforms.

Status: Apple currently provides a range of application development tools from the built-in Macintosh user interface functions to host integration tools such as the 3270 API, MacAPPC, MacWorkstation, and emerging third party tools such as Mitemview and Connectivité.

Common Applications (CA)

Function: SAA-compliant applications are user applications that conform to one or more of the three underlying SAA components (CCS, CPI, CUA). They should interoperate with each other, have the same user interface, and offer the same functionality.

Reality: Operating system differences will continue to dictate a multilevel, fragmented approach to user functions and interfaces across the SAA platforms. CA is only relevant in the context of a particular organization's existing and planned computer/communications solutions to their business needs.

According to IBM an SAA application has the following characteristics:
(SAA...A Guide For Evaluating Applications, IBM G320-9803-00)

- Runs in an SAA environment
- Conforms with Common User Access
- Uses Common Programming Interface for all functions addressed by the CPI
- Uses Common Communications Support for all functions addressed by CCS
- Is structured for cooperative processing... or to allow migration

While Macintosh applications can not technically be SAA Common Applications due to lack of CUA support and the obvious fact that Macintosh applications do not run in one of the SAA environments, Macintosh can participate in SAA environments through targeted support of key SAA elements. IBM further states that:

“Applications, therefore, should be evaluated on their use of SAA in relation to the needs of the organization and availability of SAA elements ... not on a binary basis.”

As an example, IBM's OfficeVision family of products, billed as a set of SAA common applications, are products that the Macintosh can for the most part communicate with today, or will be able to in the future based on existing Apple protocols, interfaces, and tools (see Appendix B).

Apple's Position: We will continue to support key CCS networking protocols, and integrated them with Macintosh functions to enable applications that run in SAA environments.

Status: Key CCS products for the Macintosh are available from Apple and third parties, CPI tools under investigation (see Appendix A for details).

OfficeVision

OfficeVision has been put forth by IBM as the first set of SAA compliant applications. While these applications do not offer the customer the same functions or even the same user interface across the three SAA platforms, they will, over time, provide common functionality - primarily mail, document management, and calendaring as well as a means to integrate other business applications. Also supported is the interoperability of each product with other family members - e.g., OfficeVision/2 LAN can act as a enterprise-wide mail gateway to OfficeVision/VM or OfficeVision/MVS.

There is very little new or visionary about the OfficeVision products. They are a collection of programs which, with the exception of OfficeVision/2 LAN Series, have been available for years and have been renamed and nominally enhanced. What's more important is that IBM is positioning these office products as the integration point and single user view for a wide range of office and line of business applications.

OfficeVision/VM Series

Function: A set of office-oriented applications based on the existing PASF (DOS, OS/2) and PROFS (VM/CMS) products. Provides mail, scheduling, and decision support.

Reality: PROFS is the most widely installed of IBM's office products. While IBM has changed the name of PROFS to OfficeVision/VM, it is still PROFS. IBM has committed to providing future versions of OfficeVision/VM that are closer to the user interface and interoperability (e.g., APPC, LU6.2) goal for SAA Common Applications.

Apple's Position: With the large installed base of PROFS in many of Apple large accounts, we acknowledge the importance of PROFS and OfficeVision/VM for organization-wide mail service and are committed to Macintosh participation beyond terminal emulation from a development tool and third party perspective.

Status: Macintosh access is currently supported with MacDFT and third party 3270 emulators. In the very near future, currently available and emerging front-ending tools will result in a variety of PROFS and OfficeVision/VM interface and gateway alternatives. The key categories of support are:

- 1) 3270 terminal emulation (available now - see Appendix B)
- 2) Macintosh fronts end running over Apple and third party 3270 SNA products (beginning to appear from third parties based on HyperCard, MacWorkstation, and vendor proprietary tools)
- 3) Transparent AppleTalk-based mail gateways (future third party).

OfficeVision/MVS Series

Function: A set of office oriented products for users in an MVS/CICS/DISOSS environment. Host-based functions include mail, document processing, and library store, search, and retrieve. For the PC user, the Personal Services/CICS software will be replaced by the OfficeVision/2 LAN functions.

Reality: DISOSS and its associated products have never achieved more than nominal market acceptance - primarily due to high cost, host resource consumption, and it only runs in a CICS environment.

Apple's Position: As we implement the CCS technologies such as DIA and SNA/DS to complement MacAPPC, Macintosh client access can be provided by Apple or third parties.

Status: Currently supported access from the Macintosh through MacDFT and third party 3270 emulators. The Macintosh user cannot, however, move document to or from the host component until DIA is supported. We will monitor whether OfficeVision/MVS becomes more successful than DISOSS and either make product plans or evangelize developers accordingly.

OfficeVision/400 Series

Function: A superset of the office functions provided by the AS/400 Office application software. The AS/400 Office software currently provides mail, correspondence processing, scheduling, database access, and business graphics. Enhancements (OS/400 Office, Release 1, June, 1990) will include an enhanced "SAA CUA" user interface along with calendar and correspondence processing extensions.

Reality: High memory and disk space requirements on the AS/400. While OfficeVision/400 will have a new SAA CUA user interface, it will remain text-based.

Apple's Position: Apple currently provide network access to the AS/400 (TokenTalk NB Card, Coax/Twinax Card, and MacAPPC), enabling Macintosh integration with the various AS/400 services. We will continue to monitor market requirements and IBM interface products, possibly resulting in Apple or third party Macintosh to AS/400 applications.

Status: 5250 terminal emulation access to AS/400 Office (and OfficeVision/400 in June, 1990) is available from third parties, but without access to the Shared Folders feature. Several of the SAA CCS architectures under investigation and development such as SNA/DS and DIA will provide the basis for future third party application access to these services.

OfficeVision/2 LAN Series

Function: A set of LAN-based office products incorporating mail, scheduling, document library, telephony, and address book. Support is provided for OS/2 Office Servers and client software and MS-DOS client access to the OS/2 Office Server. The user interface for the OS/2 version is built on Presentation Manager with the DOS client version remaining text-based. Release 1 (mail, address book, telephony) is now available and Release 2 (adds calendar, file cabinet, library store/search) is scheduled to ship in March, 1990.

Reality: This the only new product in the OfficeVision family. While its features are basic to the office environment, there is nothing at all “visionary” or trend-setting about it. Additionally, the product carries the weight of the OS/2 environment: high cost and complexity and lacking a rich base of consistent, mature complementary software.

Apple’s Position: While the same functionality is available from third party vendors for the Macintosh, certain customers may choose OS/2 Office as their organizational standard. This means that Macintosh client access to the services offered may be required as OfficeVision products become available, evolve functionally, and are accepted in the marketplace.

Apple will monitor the vendor interface specifications as they are revealed by IBM (preliminary due in June, 1990) and market acceptance of OS/2 Office before making a commitment to provide access to the services.

Status: Macintosh users cannot access an OS/2 Office Server. Alternatives under investigation.

Summary

SAA is an attempt to solve IBM’s unique problems of multiple hardware architectures, operating systems and development tools. Eventually, SAA will result in applications that “look” and function in a similar fashion across all of the SAA designated platforms – just as applications on the Macintosh do today. But it will take IBM quite a few years to reach that goal. SAA was announced in 1987, yet no SAA compliant applications exist today and the definition of SAA compliance continues to evolve.

What SAA does provide today is a road map; a set of guide lines for developing products that work with IBM’s strategic architecture. Since Apple is committed to integrating the Macintosh with these environments, SAA helps to define the appropriate development projects. To achieve its integration goal, Apple is focusing on the elements of SAA that are important in this respect. This includes all of the CCS data links and implementations of the key programming interfaces of CPI. We are not addressing any of the elements of CUA, since these are modeled on the Macintosh user interface. CUA is an attempt by IBM to get to where the Macintosh has been since 1984.

Support for these elements of SAA will allow us to integrate the Macintosh with SAA compliant applications such as OfficeVision while preserving our traditional added value.

Recommended Reading and Education

Inside SAA, 2nd Edition, Architecture Technology Corporation, Minneapolis, Mn. 55424 • (612) 935-2035

Miscellaneous seminars and reports, Gen2 Ventures, Saratoga, Ca. 95070 • (408) 446-2277

SAA: An Overview, IBM manual GC26-4341-03

SAA: A Guide For Evaluating Applications, IBM manual G320-9803-00

IBM Systems Journal, IBM Corporation, Vol. 27, No. 3, 1988

Apple SNA Environments and Products Course, Apple Training Support

Apple Guide to Multivendor Communications, Apple N&C Publications

OASIS and Macintosh Advantage marketing materials

Apple Market Intelligence (AppleLink: COMPETITION to request documents)

- OfficeVision: IBM Comes to Terms With the PC
- Winning the Graphical User Interface War
- Macintosh vs. OS/2: Sales Implications

Glossary

Source: Inside SAA (ref. above)

ACF/NCP	Resides in the IBM 3720 or IBM 3725 Communication Controller and provides the physical management of the communication network. Its main function is to control attached lines and workstations, perform error recovery, and route data through the SNA network.
ACF/VTAM	The base for the IBM SNA network, which may be thought of as an “operating system” for the network. Its functions are analogous to the functions of an IBM host operating system in terms of resource sharing and logical handling of user requests.
Advanced Program-to-Program Communication (APPC)	An architecture for peer-to-peer, application-to-application program. Also called LU6.2, which is the technical name for the marketing name APPC.
Application Programming	A protocol boundary which can be used by arbitrary user-written programs.

Interface(API)

Advanced Peer-to-Peer Networking (APPN) common communications support(CCS)

An extension of LU6.2 and PU2.1, which allows peripheral nodes to perform intermediate and dynamic routing functions.

Establish SNA and international communications standards chosen for SAA to provide enterprise-wide systems solutions.

common programming interface(CPI)

Building blocks for architecture development under SAA that provide a consistent programmer view for interfaces, languages, and program services.

cooperative processing

A loosely defined term that includes extensions of resources (virtual disks and printers), access to distributed data, and offloading of processing in distributed applications.

Database2 (DB2)

IBM's relational database management system program product for the MVS/XA and MVS/370 environments.

datastream

A series of bits appearing on a communications link structured according to some agreed rules. These rules may consist of character codes, control characters, header and trailer information, and field lengths.

Distributed Data Management (DDM)

An architecture consisting of formats and protocols for supporting distributed database access allowing a single database to span multiple machines. DDM operates over APPC.

Document Content Architecture (DCA)

Defines the form and meaning of the content of a document. Both revisable form and final (printable) forms are defined.

Document Interchange Architecture (DIA)

Defines how document distribution and processing functions are to be communicated through an office system network

Enhanced Connectivity Facilities (ECF)

A uniform architecture for PCs/Personal Systems to exchange data with System/370 host processors and to provide access to resources on these processors. Users can access host files, disk space, and printer facilities.

Graphical Display Data Manager (GDDM)

IBM's key product for graphics and related advanced function device support. It reduces apparent complexity for the user who is creating screen and printer layouts.

Intelligent Printer

Protocols and formats for intelligent output devices for inter-

Data Stream (IPDS)	pretation of different kinds of data streams; i.e. data, text, graphics, charts, image, and voice.
Logical Unit (LU)	Code residing in a machine on a network, either in firmware or in software, which allows the machine to communicate on the network. The LU provides services to the end-user, and may be thought of as the end user's "port" into the network. Nodes may contain multiple LUs.
Low Entry Networking (LEN)	Direct peer-to-peer connections between PU2.1 nodes and attachment to System/36 APPN. LEN is the marketing name for PU2.1 which is the technical name.
peer	Generally, being equal. Used for distribution systems and intelligent work stations to denote a balanced relationship in their communication (no master/slave or primary/secondary relation-ship).
peer-to-peer communication	Communication between two LUs without an intermediate host.
Physical Unit (PU)	A component of an SNA node that manages network resources, such as communication lines. There is one PU per node.
Query Management Facility (QMF)	A program product that provides interactive data base facilities to novice end-users and is also useful as a programming tool in application development and prototyping of database queries and reports.
relational data base	A data structure perceived as a collection of tables.
Server-Requester Programming Interface (SRPI)	A programming interface for developing applications that require coordination between System/370 hosts (servers) and personal computers (requesters) allowing users to query and extract data from hosts, transfer files between hosts and PCs and issue host commands from a PC.
session	A logical connection between two NAUs.
Structures Query Language (SQL)	A high level non-navigational language that allows programmers to access a relational database such as DB2.
Synchronous Data Link Control (SDLC)	A synchronous low-level communications protocol which can support APPC.

Systems Application Architecture (SAA)	A collection of selected software interfaces, conventions, and protocols that is the framework for development of consistent applications across the future offerings of the major IBM computing environments--System/370, System/3X (AS/400), and Personal Computer (OS/2).
Systems Network Architecture (SNA)	IBM's master plan for allowing its products to communicate. SNA defines logical structures, formats, protocols, and procedures for exchanging information on a data communications network.
Systems Network Architecture Distribution Services (SNA/DS)	An architecture incorporating DCA, DIA, and APPC that supports store-and-forward document distribution between multiple mainframes running DISOSS.
Systems Network Architecture File Services (SNA/FS)	An application layer function that uses SNA/DS for identifying and moving files among systems in an SNA network.
Systems Network Architecture Management Services (SNA/MS)	A range of SNA network management architectures and products. Includes Netview and Netview/DM
token-ring	IBM's strategic local area network (LAN) based on star shaped ring wiring topology using a controlled token for access. Also an IEEE 802.5 standard.
transaction	A logical unit of work, from the point of view of an application program on a network.
transaction program	An application program which performs transaction in cooperation with one or more other application programs on a network.
X.25	A CCITT standard consisting of a collection of physical, datalink, and network protocols for public long-haul networks.

Appendix A

Macintosh Products Mapped to SAA

SAA Element	Macintosh Products	Notes/Issues
<hr/>		
Common Communications Support - CCS		
Data Streams		
3270	Apple 3270 API	Available through APDA. 3270 API subject to interface changes in 1991 as the Apple SNA product line evolves.
DCA	Apple, 3rd party	RFT supported today. Mixed Object DCA (MODCA) is the new version - graphic, sound, image, font objects. Investigation planned.
IPDS	ncs	Intelligent Printer Datastream Protocol. Investigation planned. Postscript™ is Apple's description technology.
corresponding page		
Application Services		
SNA/DS	ncs	SNA Distribution Services under development.
DIA	ncs	Document Interchange Architecture under development.
DDM	ncs	Distributed Data Management. Investigation planned.
SNA/MS	partial	SNA Management Services encompasses a variety of network management and control functions, including NetView alerts, which are supported in MacDFT 1.1. Further investigation planned.
SNA/FS	ncs	SNA File Services is an emerging IBM technology that is likely to be incorporated into SAA CCS Investigation planned.

key: na = *not applicable* to the Macintosh.

ncs = *not currently supported* on the Macintosh, but candidate for future Apple or 3rd party support.

partial= subset implemented, additional support under investigation.

Session Services

LU6.2	MacAPPC	SNA direct or AppleTalk-distributed APPC verb interface for cooperative application development.
-------	---------	--

Network

NT2.1	MacAPPC	Does not support the independent LU capability in VTAM release 3.2.
-------	---------	---

Data Link

Token-Ring	TokenTalk NB	4 MB/sec. today. 16/4 MB/sec. under development.
------------	--------------	--

SDLC	Serial NB	Supports MacDFT 1.1 and MacAPPC 1.1.
------	-----------	--------------------------------------

X.25	MacX25	SNA over MacX25/Serial NB support planned.
------	--------	--

Common Programming Interface - CPI

Languages

C	MPW, 3rd party	The most popular language for Macintosh applications.
---	----------------	---

PL/1	na	Limited or no demand for Macintosh.
------	----	-------------------------------------

Cobol	ncs	May be developed by 3rd party tool vendors.
-------	-----	---

RPG	na	Specific to mid-range System 3/x and AS/400.
-----	----	--

Fortran	3rd party	Limited demand for Macintosh.
---------	-----------	-------------------------------

IBM CSP	na	IBM's host-based application generator.
---------	----	---

REXX and	na	High level procedural language. Corresponding functionality is available for the Macintosh from Apple 3rd parties.
-------------	----	--

QMF	na	Query Management Facility. Corresponding functionality available for the Macintosh within CL/1 and third party applications and desk accessories that use CL/1.
-----	----	---

GDDM advanced	ncs	Graphical Data Display Manager - high level mainframe oriented graphic facilities for programmers and users.
------------------	-----	--

EasyView na Host-based tool for programmers to build character-based screens and menus.

Programming Interfaces

Communications

CPI-C ncs The SAA version of APPC/LU6.2. Will be supported in a future Apple APPC/LU6.2 product.

Database

SQL ncs Structured Query Language for application access to databases. CL/1 is Apple's corresponding SQL-based language for access to IBMDB2 and SQL/DS. databases. System 7.0 DAM enables implement specific SQL host third parties to dialects.

Query na A QMF-based tool for users to perform SQL database queries. Corresponding applications and desk accessories based on CL/1 available from 3rd parties for the Macintosh.

Dialog na Programming tools for control of user interaction with applications. Implemented in ISPF, EZ-VU, and SGML (ISO std) on the host and OS/2 Dialog Manager on the PS/2.

Presentation na Based on the mainframe GDDM, which is also the basis for the OS/2 Presentation Manager. Corresponding, superior functionality is built into the Macintosh.

Common User Access - CUA

1987 na IBM's first cut at a single system and application view for the user. But oops! Forgot that mainframes don't get along well with windows, icons, and mice.

1989 na That's better. Now there are TWO kinds of common user access: one for mainframes (characters and brain-dead windows) and another for "Programmable Workstations" (text and graphics) like the PS/2. The Apple Human Interface Guidelines continues

to be our corresponding, but
bible.

superior, user interface

Appendix B

Macintosh Products Mapped to the OfficeVision Family

OfficeVision IBM Product

Product	Replaced	Macintosh Access/Notes
---------	----------	------------------------

OfficeVision/VM PROFS (host)		MacDFT, 3rd party 3270, and fronts ends.
	PASF (PC)	

OfficeVision/MVS DISOSS (host)		Accessible via MacDFT and 3rd party 3270, but file
		Personal Services(PC) transfer (library access) not supported.

OfficeVision/400 AS/400 Office (host)		Third party 5250 available, but Shared Folder feature
	PC Support (PC)	not supported.

OfficeVision/2 LAN New		Macintosh client access not available. Alternatives for
under		access to mail, library, and calendar services
vendor interface		investigation pending the release of
(due in June, 1990).		specifications by IBM