

CreativEdge Developer's Guide

Please read this entire guide

Important

Please read this entire guide before you install or operate this equipment. Give particular attention to all safety statements.



Scientific
Atlanta

CreativEdge Developer's Guide

Notices

Trademarks

Scientific-Atlanta, Scientific-Atlanta ARCS logo, and EXPLORER are registered trademarks of Scientific-Atlanta, Inc. PowerKEY, VCR Commander, and Tape 'N View are trademarks of Scientific-Atlanta, Inc.

All other trademarks listed herein are the property of their respective owners.

Disclaimer

This interface definition/system overview is published by Scientific-Atlanta, Inc. to inform the industry of the requirements and operational characteristics for interactions with the Scientific-Atlanta Digital Broadband Delivery System for the delivery of broadcast and/or interactive video services.

Scientific-Atlanta assumes no responsibility for errors or omissions that may appear in this guide. Scientific-Atlanta reserves the right to revise this document without notice at any time, for any reason, including but not limited to, conformity with standards promulgated by various agencies or industry associations, utilization of advances in the state of the technical arts, or the reflection of changes in the design of any equipment, techniques, or procedures described or referred to herein.

Scientific-Atlanta makes no representation or warranty, express or implied, of any kind with respect to this document or any of the information contained herein. YOUR USE OR RELIANCE UPON THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN IS AT YOUR OWN RISK. Scientific-Atlanta shall not be liable to you for any damage or injury incurred by any person arising out of your use of this material.

This document is not to be construed as conferring by implication, estoppel, or otherwise any license or right under any copyright or patent, whether or not the use of any information in this document necessarily employs an invention claimed in any existing or later issued patent.

Continued on next page

Notices, Continued

Documentation Copyright Notice

© 1999 Scientific-Atlanta, Inc. All rights reserved.

Information in this document is subject to change without notice. No part of this document may be reproduced in any form without the express written permission of Scientific-Atlanta, Inc.

Printed in the United States of America.

Software Use Notice

The software described in this document is copyright Scientific-Atlanta, Inc. and is furnished to you under a license agreement or nondisclosure agreement, and may only be used or copied in accordance with the terms of your agreement.

Firmware Use Notice

This equipment contains firmware that is copyright Scientific-Atlanta, Inc., and may be used only in the equipment on which it is provided. Any reproduction or distribution of this firmware, or any portion of it, is prohibited without the express written consent of Scientific-Atlanta, Inc.

Contents

	Notices	i
	Warranty and Disclaimer	vii
Chapter 1	Introduction	1-1
	Overview	1-1
	Introduction.....	1-1
	Audience and Scope	1-1
	In this Chapter.....	1-1
	Related Documents.....	1-2
	Acronyms	1-3
Chapter 2	EXPLORER DBDS Overview	2-1
	Overview	2-1
	Introduction.....	2-1
	In this Chapter.....	2-1
	Network Overview	2-2
	Understanding the DBDS Network.....	2-2
	Core Technologies.....	2-2
	Supported Media	2-3
	Hardware Overview.....	2-4
	Headend Components.....	2-4
	EXPLORER DHCT.....	2-4
	Software Overview	2-5
	Application Interface System.....	2-6
	DNCS	2-7
	PowerTV OS.....	2-7
	SARA	2-7
	Applications	2-8
	Development Environments.....	2-9
	Network Development Kits (NDK)	2-9
	Software Development Kits (SDK)	2-9

Continued on next page

Contents, Continued

Chapter 3	EXPLORER DBDS Applications	3-1
	Overview	3-1
	Introduction.....	3-1
	In this Chapter.....	3-1
	Types of Applications.....	3-2
	Native Applications.....	3-2
	Middleware-Based Applications	3-2
	Resident Applications.....	3-2
	Downloadable Applications.....	3-3
	Application Lifecycle.....	3-4
	Development.....	3-5
	System Integration and Testing	3-5
	Certification.....	3-6
	Installation.....	3-6
	Provisioning	3-7
	Authorization	3-8
	Request	3-9
	Download	3-9
	Launch	3-10
	Service Activation	3-10
	Execution	3-11
	Service Suspension.....	3-11
	Termination.....	3-12
	Un-Installation	3-13
	Application Architecture.....	3-14
	Application Server	3-14
	Application Client.....	3-14
	Network Interface	3-14
	Broadcast File System (BFS)	3-15
	MPEG Private Data	3-15
	IP Data.....	3-15

Continued on next page

Contents, Continued

	DSM-CC PassThru Messaging.....	3-16
	VBI / Closed-Captioning Data.....	3-17
	DSM-CC Session Protocol.....	3-17
Chapter 4	Application Interface System (AIS).....	4-1
	Overview.....	4-1
	Introduction.....	4-1
	In this Chapter.....	4-1
	Overall Architecture.....	4-2
	AIS Components.....	4-2
	Middleware Support.....	4-3
	HTML/JavaScript Engine.....	4-3
	HTTP Proxy.....	4-3
	Personal Java (pJava).....	4-4
	Application and Service Support.....	4-5
	Application Server Manager.....	4-5
	Billing Interface.....	4-5
	Emergency Alert System (EAS).....	4-5
	Flash Packager.....	4-5
	Navigator API.....	4-6
	Service Application Manager (SAM).....	4-6
	Resource Access and Management.....	4-9
	Broadcast File System (BFS).....	4-9
	Client View of Broadcast File System.....	4-9
	Server View of Broadcast File System.....	4-10
	DSM-CC Messaging Manager.....	4-10
	IP Gateway.....	4-10
	Server.....	4-11
	Client.....	4-11
	Window Manager.....	4-12

Continued on next page

Contents, Continued

- Chapter 5 Digital Network Control System (DNCS).....5-1**
 - Overview 5-1
 - Introduction..... 5-1
 - In this Chapter..... 5-1
 - DNCS Components 5-2
 - Broadcast Control Suite (BCS) 5-2
 - Conditional Access Manager (CAM)..... 5-2
 - DSM-CC Session Manager 5-3

- Chapter 6 Customer Information..... 6-1**
 - Overview 6-1
 - Introduction..... 6-1
 - In this Chapter..... 6-1
 - Customer Support 6-2
 - List of Telephone Numbers..... 6-2
 - Returning Products..... 6-3
 - Procedure..... 6-3

Warranty and Disclaimer

Statement

Scientific-Atlanta, Inc. (“S-A”) warrants that software licensed by it (“Licensed Software”), as provided, shall conform in all material respects to its published specifications current at the time the Licensed Software was shipped, downloaded or otherwise delivered to the Customer. During the first ninety (90) days after the date of delivery of Licensed Software, S-A shall use reasonable commercial efforts to correct errors detected in Licensed Software after receiving notification of such errors from Customer.

Disclaimer

S-A MAKES NO OTHER WARRANTIES, WHETHER EXPRESS OR IMPLIED, WITH RESPECT TO ANY LICENSED SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. S-A DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN LICENSED SOFTWARE WILL MEET THE CUSTOMER'S REQUIREMENTS OR THAT THE OPERATION OF THE LICENSED SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE. S-A MAKES NO WARRANTY OF NON-INFRINGEMENT, EXPRESS OR IMPLIED. ANY THIRD PARTY SOFTWARE SUPPLIED WITH OR INCORPORATED IN LICENSED SOFTWARE IS PROVIDED "AS IS," WITHOUT WARRANTIES OF ANY KIND. IF ANY ADDITIONAL WARRANTIES ARE SUPPLIED BY A THIRD PARTY, SUCH WARRANTIES WILL BE OFFERED DIRECTLY BY SUCH THIRD PARTY TO CUSTOMER.

Customer Responsibility

Customer acknowledges its responsibility to use all reasonable methods to prove out and thoroughly test the operation of and output from Licensed Software prior to its use in Customer's operations.

Unless otherwise provided in a separate writing, and subject only to the warranty of this Section, S-A is under no obligation to provide Customer with any modifications, updates, additions or revisions to Licensed Software, nor to maintain Licensed Software in any manner.

In the event that any modifications are made to Licensed Software which have not been authorized by S-A, any and all warranty and other obligations of S-A shall immediately cease with respect to such software.

Continued on next page

Warranty and Disclaimer, Continued

Year 2000 Statement

Licensed Software may require upgrades to accept and correctly process dates before, on and after January 1, 2000 ("Year 2000 Ready"). In addition, S-A cannot warrant that any Licensed Software, even though itself Year 2000 Ready, will operate correctly when used in conjunction with hardware, software, data, applications or services of third parties or products previously purchased from S-A. S-A cannot be responsible for the interoperability of the Licensed Software with hardware, software, data, applications or other services of third parties and advises the Customer to contact all other third party suppliers and to perform its own testing adequately in advance.

Limitation of Liability

EXCEPT FOR CLAIMS FOR PERSONAL INJURY CAUSED BY LICENSED SOFTWARE FURNISHED BY SCIENTIFIC-ATLANTA, SCIENTIFIC-ATLANTA SHALL NOT BE LIABLE TO THE CUSTOMER OR ANY OTHER PERSON OR ENTITY FOR INDIRECT, SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE TRANSACTION IN WHICH THE LICENSED SOFTWARE WAS FURNISHED OR ANY ACTS OR OMISSIONS ASSOCIATED THEREWITH OR RELATING TO THE SALE OR USE OF ANY LICENSED SOFTWARE FURNISHED, WHETHER SUCH CLAIM IS BASED ON BREACH OF WARRANTY, CONTRACT, TORT OR OTHER LEGAL THEORY AND REGARDLESS OF THE CAUSES OF SUCH LOSS OR DAMAGES OR WHETHER ANY OTHER REMEDY PROVIDED HEREIN FAILS. IN NO EVENT SHALL S-A BE LIABLE TO CUSTOMER FOR ANY DAMAGES, HOWEVER BASED, IN EXCESS OF THE LESSER OF TEN THOUSAND UNITED STATES DOLLARS (U.S. \$10,000) OR THE LICENSE FEE PAID BY CUSTOMER TO S-A FOR THE LICENSED SOFTWARE.

Chapter 1

Introduction

Overview

Introduction

This guide provides information for developers of software applications for the Scientific-Atlanta Digital Broadband Delivery System (DBDS) and the EXPLORER® Digital Home Communications Terminal (DHCT)—hereafter referred to as the EXPLORER DBDS. Included is a discussion of the network, software, and hardware platforms provided by the EXPLORER DBDS from an application development perspective. This document should be used as a starting point for application development; it is part of the CreativEdge suite of developer documents regarding the EXPLORER DBDS. Specific component documentation and development tools are available separately.

Audience and Scope

This guide is designed as an architectural and functionality overview for application developers and system operators. Details of network performance, hardware specifications, and software APIs are found in the *CreativEdge Developer's Reference, Phase <N>*, for each phase of the DBDS system as it evolves. Note that components and functionality discussed in this document are not necessarily available on a particular release of the DBDS. The reference document for each phase details the exact functionality available in a specific DBDS.

Access www.sciatl.com for updates and modifications to this reference. Updates are made freely available to application developers.

In this Chapter

This chapter contains the following topics.

Topic	See Page
Related Documents	1-2
Acronyms	1-3

Related Documents

The following are documents also available as part of the CreativEdge developer documentation:

- *CreativEdge Developer's Reference, Phase <N>*
- *Scientific-Atlanta Resident Application (SARA) Application Developer's Guide*
- *Scientific-Atlanta Resident Application (SARA) Application Developer's Reference, Phase <N>*

The following lists references useful for developing EXPLORER DBDS applications.

- *Generic Coding of Moving Pictures and Associated Audio Information, Part 6: Extension for DSM-CC, ISO/IEC JTC 1/SC 29N, December 1995*
 - *Business Operations Support System Interface Specification, Scientific-Atlanta[®], Inc.*
 - *Business Applications Support System, Scientific-Atlanta[®], Inc.*
 - *Generic Coding of Moving Pictures and Associated Audio Information, Part 1: Systems (MPEG-2 Systems), ISO/IEC 13818-1, March 1994*
 - *System Information for Digital Television, Advanced Television Systems Committee Standard A/56, January 1996*
 - *RPC: Remote Procedure Call Protocol Specification Version 2, RFC 1057, Sun Microsystems, Inc., June 1, 1998*
 - *Digital Network Control System (DNCS) Product Family Product Specifications, Revision 1.5, September 1996*
 - *Digital Broadband Delivery System (DBDS) Phase 1.0 System Description, Revision 1.6, September 1996*
 - *PowerTV Application Development API Reference, Version 1.5, October 1998*
 - *PowerTV Operating System, Version 1.5, October 1998*
 - Bloomer, John. *Power Programming with RPC*. O'Reilly & Associates, Inc. Sebastopol, CA. 1992
-

Acronyms

The following table contains a list of acronyms used in this guide.

Acronym	Definition
AC-3	Audio Compression Level 3
AG	Administrative Gateway
AIS	Application Interface System
API	Application Programming Interface
ASI	Application Specific Integration
ATSC	Advanced Television System Committee
BFS	Broadcast File System
BIG	Broadband Integrated Gateway
BOSS	Business and Operations Support System
CA	Conditional Access
CAT	Conditional Access Table
DAVIC	Digital Audio Visual Council
DBDS	Digital Broadband Delivery System
DCN	Display Channel Number
DCT	Display Channel Table
DHCT	Digital Home Communications Terminal
DNCS	Digital Network Control System
DSM-CC	Digital Storage Media - Command and Control
DVB	Digital Video Broadcasting
ECM	Entitlement Control Message
EDT	Exception Dispatch Table
EMM	Entitlement Control Message
GUI	Graphical User Interface
HCT	Home Communications Terminal

Continued on next page

Acronyms, Continued

Acronym	Definition
HFC	Hybrid Fiber Coax
HITS	Headend in the Sky
ICM	Inverse Channel Map
IP	Internet Protocol
IPG	Interactive Program Guide
IPPV	Impulse Pay-Per-View
IRD	Integrated Receiver Decoder
IRT	Integrated Receiver Transcoder
ISC/IEC	Integrated Service Commission/International Electrotechnical Commission
LID	LAN Interconnect Device
MAC	Media Access Control
MIB	Management Information Broker
MPEG	Moving Pictures Experts Group
MSK	Multisession Key
NIT	Network Information Table
NTSC	National Television System Committee
NSAP	Network Service Access Point
NVM	Non-Volatile Memory
OC-3	Optical Carrier Level 3
OOB	Out-of-Band
OS	Operating System
OTF	On-The-Fly Settings
PAL	Phase Alternating Line
PAT	Program Association Table

Continued on next page

Acronyms, Continued

Acronym	Definition
PES	Packetized Elementary Stream
PIN	Personal Identification Number
PMT	Program Map Table
PPV	Pay-Per-View
PSI	Program Specific Information
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase-Shift Keyed
RPC	Remote Procedure Call
RTE	Real Time Encoder
SAM	Service Application Manager
SARA	Scientific-Atlanta Resident Application
SCTE	Society of Telecommunications Engineers
SCH	Schematic
SDT	Service Definition Table
SG	Service Gateway
SNMP	Simple Network Management Protocol
SONET	Synchronous Optical Network
STM-1	Synchronous Transfer Mode Level 1
TCP/IP	Transport Control Protocol/Internet Protocol
UDP/IP	User Datagram Protocol/Internet Protocol
UI	User Interface
URL	Universal Resource Locator
VASP	Value Added Service Provider
VCT	Virtual Channel Table

Chapter 2

EXPLORER DBDS Overview

Overview

Introduction

Scientific-Atlanta developed the Digital Broadband Delivery System (DBDS) to provide an economically sound approach to deploying analog and digital services. It supports the many different existing broadband service providers, as well as the digital service providers of the future. An important characteristic of the DBDS is its ability to provide a real-time reverse path. This allows many types of interactive applications to be supported, in addition to more traditional broadcast services.

The DBDS receives analog and digital services from various sources (for example, analog and digital broadcast distribution systems, Wide Area Networks, and audio and video program servers). The DBDS then securely transports the services to Digital Home Communications Terminals (DHCTs) in the area where the system resides. Services supported by the DBDS are controlled by the Digital Network Control System (DNCS).

The remainder of this chapter provides an overview of the DBDS network, hardware, and software to help application developers understand the system architecture.

In this Chapter

This chapter contains the following topics.

Topic	See Page
Network Overview	2-2
Hardware Overview	2-4
Software Overview	2-5
Development Environments	2-9

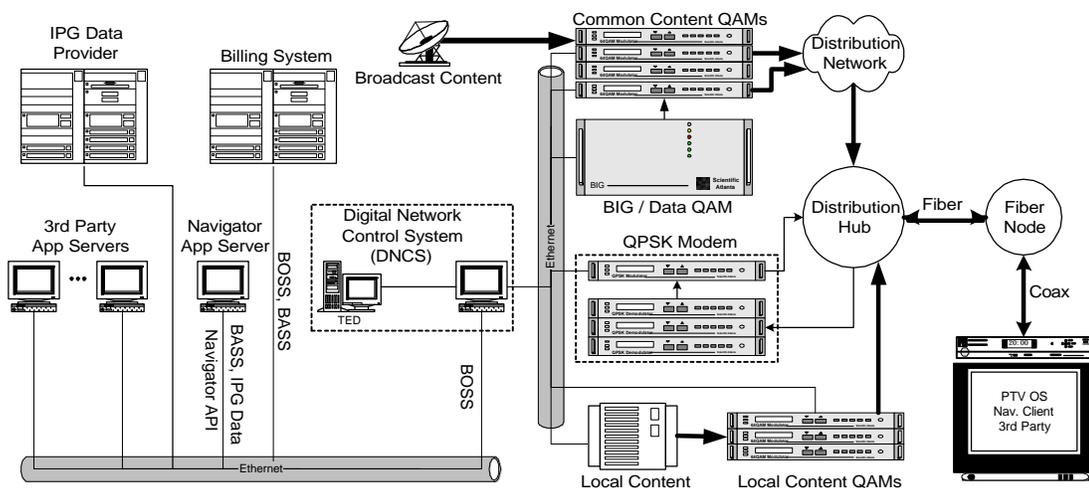
Network Overview

Understanding the DBDS Network

The flexibility of the EXPLORER DHCT enables the DBDS network components to be configured based on particular headend needs. The two most widely implemented configurations are the following:

- AM Fiber Transport Network - offers cost effective content delivery within smaller metropolitan areas
- SONET/SDH Transport Network - offers tailored content delivery over greater distances

The following diagram illustrates a simplified DBDS.



Core Technologies

The core technologies used in the DBDS are based on *open standards*. Some of the technologies were developed by sources external to Scientific-Atlanta.

- Scientific-Atlanta's Digital Broadband Delivery System (DBDS) is compliant with the following:
 - Society of Cable Television Engineers (SCTE) and DVB standards for Motion Picture Experts Group (MPEG) video
 - ATSC Audio (digital audio), System and Signaling
 - Digital Audio Video Council (DAVIC)

Continued on next page

Network Overview, Continued

- PowerTV, Inc.'s PowerTV Operating System supports complete session and signaling management on a hybrid fiber/coax network
- Standard Application Programming Interfaces (APIs) allow software developers to write applications without proprietary information
- Harmony specifications allow for dual conditional access for DigiCipher and PowerKEY systems

Note: Harmony is a cable TV industry technology that allows multiple digital home communications terminals, from different manufacturers, to co-exist within the same cable system, and enables decryption between terminals.

Supported Media

Via the EXPLORER DHCT, the DBDS delivers interactive digital services as well as analog services in real time, and supports the following media types and sources:

- Digital satellite delivered content (HITS, HBO, PowerVu/Athena, etc.)
- Terrestrial delivered content (AM Fiber, SONET/SDH)
- MPEG Real Time Encoders (RTE) encoding live video or other analog video feeds
- Internet and Intranet (Internet Protocol—IP data)
- Media servers (World Wide Web, Interactive Program Guide, Pay-Per-View, Near Video-On-Demand, and Video-On-Demand)
- Digital data or Internet Protocol (IP) data
- MPEG-2, DAVIC, DVB, ATSC

Note: The EXPLORER DHCT also supports AM Fiber, SONET and SDH interconnects, DAVIC 1.1 two-way real-time signaling, and SNMP network management support.

Hardware Overview

Headend Components

The following table provides brief descriptions of the headend components within a DBDS.

Component	Function
DNCS Hardware Platform	The hardware platform upon which the DNCS software executes, usually a Sun workstation.
BIG (Broadband Integrated Gateway)	The BIG is an MPEG-2 transport re-multiplexor. It is capable of receiving input from multiple sources and performing drop/add operations. The BIG outputs MPEG-2 transport streams that are made up of programs from the various input transport streams.
QAM Modulator	The QAM Modulator receives MPEG-2 Transport Streams and modulates the digital signal onto an RF carrier (may be 6 or 8 MHz).
QPSK Modulator	The QPSK Modulator transmits packets containing IP or MPEG private sections over the forward data channel to the EXPLORER DHCT. Receives Asynchronous Transfer Mode (ATM) cells that contain data or Media Access Control (MAC) messages from QPSK Demodulators and routes them to the LAN Interconnect Devices.
QPSK Demodulator	The QPSK Demodulator receives and routes the reverse data channel signal from the EXPLORER DHCT. Provides the QPSK Modulator with power level and timing performance information.

A typical network configuration includes QAM Modulators at the headend and possibly at the network hubs. A QPSK Modulator is present in each hub, with up to eight QPSK Demodulators.

EXPLORER DHCT

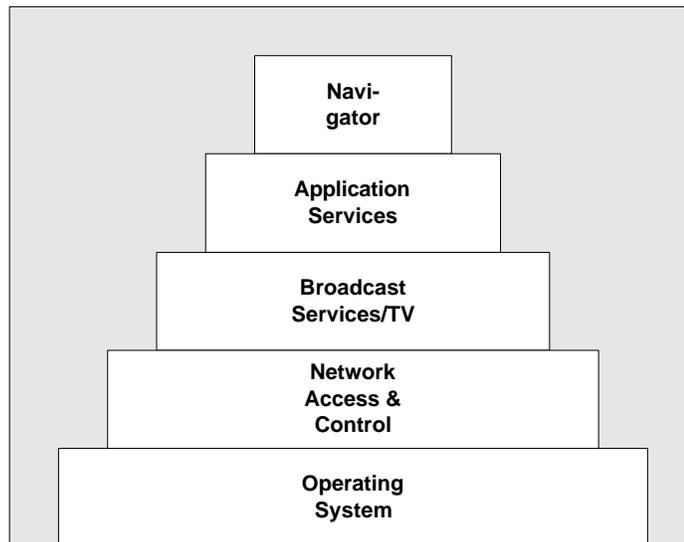
Scientific-Atlanta's EXPLORER DHCT is the hardware device that delivers digital and analog interactive television content in the subscriber's home. The EXPLORER DHCT is a client device that supports real-time, session-based interactivity in addition to broadcast digital and analog services.

There are several versions of the EXPLORER, beginning with the EXPLORER 2000. The EXPLORER 2000 DHCT uses the SUN UltraSPARC 32-bit RISC microprocessor core and the PowerTV[®] Operating System and Application Programming Interface (API). The Eagle[®] graphics/audio accelerator allows for 256 or 65,000 graphics colors optimized for NTSC and PAL displays. The EXPLORER 2000 DHCT also includes the PowerKEY[®] Conditional Access System with hybrid Public Key/Secret Key technology.

Software Overview

The figure below shows the layering of software functionality that supports applications in the EXPLORER DBDS. The most basic functionality is that provided by an Operating System. A network layer provides downstream and upstream paths for IP data, as well as control of network elements. Upon this is built functionality for broadcast television services, including delivery of MPEG video, audio, and data. The next layer allows different types of interactive applications to provide services and share resources in the network. Finally, a Navigator provides the DHCT user with access to the basic cable services and a paradigm to access the different applications available on the network.

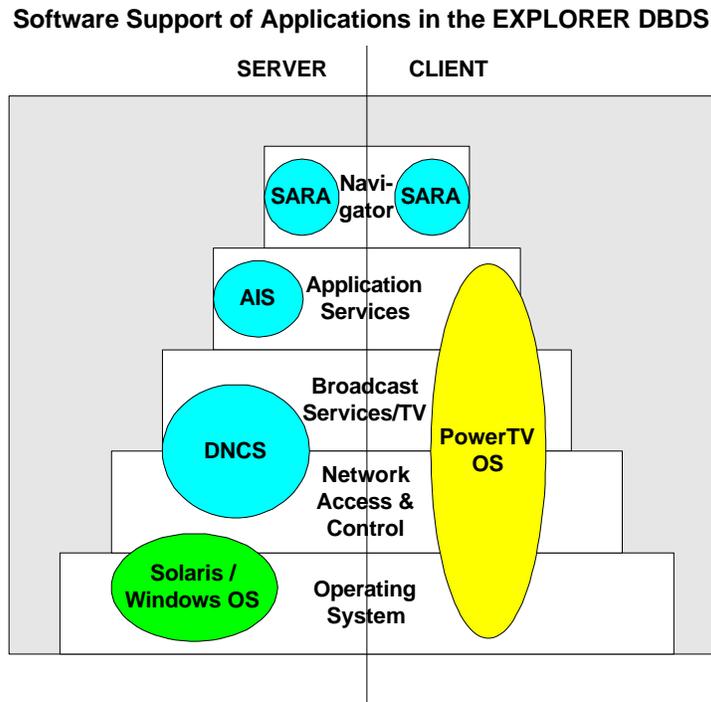
Layers of Software Functionality in the EXPLORER DBDS



Continued on next page

Software Overview, Continued

The next figure shows the EXPLORER DBDS software products superimposed on the layers of functionality that the products provide. These products are described in the following sections.



Application Interface System

The Application Interface System (AIS) is the collection of end-to-end software interfaces enabling applications on the DBDS. The AIS provides an interface and mechanism through which applications can be developed, introduced on the network, defined as a service, downloaded to the DHCT, executed on the DHCT, and removed from the network.

The AIS is an end-to-end layer (client and server) of application and service management software components that an application must utilize to exist in the DBDS. It also consists of resource access and management components generally required by applications in the DBDS. None of the AIS components have any client-side user-interface, although GUIs can exist to access server-side components.

The AIS at a high level represents functionality that is available as part of the EXPLORER DBDS. Currently, the interface to client-side components is provided by the PowerTV Operating System. Some AIS client components are part of SARA in early phases and will migrate into the PowerTV OS in the future.

Continued on next page

Software Overview, Continued

DNCS

The software components of the Digital Network Control System (DNCS) provide complete management, monitoring, and control of DBDS network elements and broadcast services provided to subscribers. This includes the definitions of sources and DSM-CC sessions, UNconfiguration of DHCTs in the network, and conditional access management. Additionally, the DNCS interfaces with and manages the various hardware elements of the DBDS. An Administrative Console GUI allows system operators to access the DNCS.

PowerTV OS

The PowerTV Operating System is the resident operating system on the EXPLORER 2000 DHCT. Designed specifically for developing and delivering multimedia over broadcast and interactive cable networks, the PowerTV OS uses an open and modular platform to allow flexibility and customization. The features of the operating system include:

- Real-time multimedia delivery
- Multitasking capability
- Open platform

See www.PowerTV.com for additional information.

SARA

The Scientific-Atlanta Resident Application (SARA) software, also known as the EXPLORER Navigator, provides basic cable services and navigation framework to the subscriber. Services include watching television and Pay-Per-View events, listening to digital music, and an Interactive Program Guide. SARA also allows subscribers to access the settings of the EXPLORER DHCT, including volume, parental control, VCR Commander, etc. The SARA Server resides at the cable headend, while the SARA Client is resident in the DHCT flash memory.

Continued on next page

Software Overview, Continued

The SARA API end-to-end layer provides additional functionality not required by or of applications. This API allows applications to access information about the current channel and other user settings, such as parental control status and PINs. SARA also provides a user-interface style guide to help developers of applications that co-exist with SARA present a unified experience to subscribers. Because of the way the DBDS is evolving, some components of the AIS are provided at the SARA API level initially and then migrate into the PowerTV Operating System.

See the *Scientific-Atlanta Resident Application (SARA) Application Developer's Guide* and the *Scientific-Atlanta Resident Application (SARA) Application Developer's Reference, Phase <N>* for more information.

Applications

As detailed in the remainder of this document, the EXPLORER DBDS is a rich platform for interactive applications. In general, every application has both a server and client component. The application server component usually exists at the system headend, interacts with AIS Server components and the Billing System, and can provide a GUI for the system operator. The application client can be resident or downloaded into the EXPLORER DHCT. It interfaces with the AIS Client components, the PowerTV Operating System, and provides a GUI for the subscriber. Applications can be classified as native or middleware-based, the latter making use of middleware components such as HTML and Java.

The next section details the types of applications that can exist on the DBDS, the application lifecycle, and the typical architecture of applications on the DBDS.

Development Environments

Application developers can use client and server development kits to aid them in their efforts, so that a complete DBDS headend is not required. Members can purchase Network Development Kits (NDK) through CreativEdge, which simulates the network environment but communicates over Ethernet to the set-top. Some developers may want a complete headend, which can be provided at a discount through a Developer License. More detailed descriptions of developer toolkits follow.

Network Development Kits (NDK)

Scientific-Atlanta's Network Development Kit (NDK) can be used for application server development, and to become familiar with the Digital Network Control System environment. The NDK simulates the Scientific-Atlanta network environment using the Ethernet interface in the EXPLORER DHCT and an inexpensive server platform running NDK software. The application server can then be developed, which will interface with web servers, customer databases, and other external subsystems, without the need for a headend. The enhanced NDK adds QAM and QPSK functionality to the development environment. Finally, developers can make arrangements to perform system integration in the CreativEdge headend labs.

Software Development Kits (SDK)

Client Software Development Kits (SDK) can be obtained directly from PowerTV, Inc. The Client SDK allows the developer to develop the client portion of an application without requiring a network. The SDK includes an Explorer Development Settop with a SCSI port, a SCSI hard drive, cables, and a PowerTV Developer Software CD which contains documentation and software tools. The SARA client can also be executed in the SDK environment to facilitate client integration. PowerTV can be reached at (408) 777-7577.

Chapter 3

EXPLORER DBDS Applications

Overview

Introduction

This chapter details the types of applications that can be developed for the EXPLORER DBDS, describes the application lifecycle of DBDS applications, and discusses the typical architecture of applications. Based on the contents of this chapter, application developers can design a specific application for the EXPLORER DBDS.

As mentioned previously, applications on the DBDS usually have server and client components: the **application server** and the **application client**. Each provides a user-interface, the server to the system operator, and the client to the subscriber. Another important aspect of the application architecture is the network interface between the application server and client.

In this Chapter

This chapter contains the following topics.

Topic	See Page
Types of Applications	3-2
Application Lifecycle	3-4
Application Architecture	3-13

Types of Applications

The software platform on the EXPLORER DBDS is designed to support several different types of applications. More importantly, these applications must be able to co-exist in the network, sharing resources in a cooperative manner. There are two types of applications: native or middleware-based, plus two classifications for where the application client code resides: resident or downloaded.

Native Applications

Native applications are usually written in C or C++ using the interfaces available from the operating systems on the server and client hardware platforms. A native application server can be a C application compiled for UNIX or Windows NT, while a native application client is written for the PowerTV Operating System.

Middleware-Based Applications

Middleware-based applications are written in the language of the middleware engine, such as HTML, JavaScript, pJava, etc. The middleware engine itself must first be ported to the EXPLORER DBDS, usually as a native application that co-exists with other native applications. Middleware-based applications can then access the application and broadcast services of the DBDS via the middleware engine API, as native function calls or a thin adapter layer.

HTML, JavaScript, and pJava support are currently available on the EXPLORER DBDS.

Resident Applications

Resident applications are applications whose client resides in the persistent flash memory on the DHCT. The AIS allows the set of applications resident in the DHCT to be configured by the system operator via the Flash Packager (Phase III or later).

The term Resident Application is often used to describe the application client software that performs the basic services such as watching TV, music, Pay-Per-View, and a program guide. However, this document refers to the application providing this functionality as the resident **Navigator**, which is in fact *one* of the resident applications. An example of such an application is the Scientific-Atlanta Resident Application (SARA), sometimes referred to as the EXPLORER Navigator.

Note: The SAM provides a load-on-boot capability that allows applications to be downloaded into dynamic RAM when a DHCT powers-on and connects to the network. Such an application thus behaves as if it is resident in flash—it executes immediately at boot and is available to provide services to the user.

Continued on next page

Types of Applications, Continued

Downloadable Applications

A non-resident application's client component is downloaded as needed and then executed on the EXPLORER DHCT. The AIS allows multiple application clients to be downloaded into the DHCT at a particular time, based on the amount of dynamic RAM available. The fundamental difference between a resident application and a downloadable application is simply where the client executable resides.

Application Lifecycle

This section describes and identifies the phases of an application's lifecycle supported by the Application Interface System.

The following lists the application's lifecycle phases:

- Development
- System Integration and Testing
- Certification
- Installation
- Provisioning
- Authorization
- Request
- Download
- Launch
- Service Activation
- Execution
- Service Suspension
- Termination
- Un-installation

Example: To help developers understand the application lifecycle and the use of AIS APIs, an example CyberDeli application is introduced here. This application is an interactive application that lets subscribers order deli sandwiches using the EXPLORER DHCT. The lifecycle discussion here is supplemented with a generic example from the CyberDeli application. Because the specifics of the CyberDeli lifecycle are particular to the different phases or system releases, the details and the source code are described separately as part of the *CreativEdge Developer's Reference, Phase <N>* for different releases of the DBDS.

Continued on next page

Application Lifecycle, Continued

Development

Application development is the first phase of the application lifecycle. The AIS supports application development in several ways. This document provides application designers with an understanding of the capabilities of the system. The *CreativEdge Developer's Reference, Phase <N>* documents include the actual AIS interface specification, for both the server and client.

Note: The CreativEdge developer reference documents cite other documents such as API references published by PowerTV.

Files, compilers, and packagers are also included as part of the AIS development package, some provided by PowerTV or other vendors. These components allow the application developer to build an installable executable that operates with the AIS on the EXPLORER DBDS. Separate application executables are created for the server and client. The last part of the development phase is the creation of an installable application package.

Application servers can utilize files for access to various AIS Server components or for SARA Server APIs. These interfaces can be BOSS transactions, Remote Procedure Calls, or CORBA interfaces. Application clients usually include only PowerTV OS header files, but can also include HTML or pJava header files, or SARA API client headers. Most of these interfaces are C, C++, or Java.

Example: The CyberDeli application is a native application. The client is written in C++ using the PowerTV interface and the server in C++ using the DNCS API. The CyberDeli application client is packaged as a PowerTV application module.

System Integration and Testing

The entire application, including client and server components, are tested on an end-to-end digital network, including all RF components. This phase is completed by the application developer in a lab environment. The application is tested for functionality to insure that it meets product specifications and user requirements. In addition, marketing may undertake focus group testing and usability testing to further refine the product prior to certification.

Example: The CyberDeli application is provided as an example and has been tested and integrated on the DBDS.

Continued on next page

Application Lifecycle, Continued

Certification

CreativEdge Certification is designed to help insure that application runs successfully on the network, without negatively impacting the performance of the cable operator's core services. It does not test core functionality of the application, but only the application's interaction with the system while providing its services. Certification is a testing and verification procedure that must be completed by Scientific-Atlanta or an authorized third-party. Issues such as memory leakage, resource management, network traffic impact, and latency will be covered to help insure that the application runs properly under loaded conditions. Certification guideline documentation assists the developer in understanding network requirements in addition to operational issues such as installation and maintenance. Certification is completed for a specific release of the application against System Release of the S-A DBDS. In this way, both network operators and application developers can be confident in delivering these new services to subscribers in a reliable manner.

Important: An application must abide by the requirements and guidelines set forth in this and other CreativEdge developer documents to be certified on the DBDS.

Example: The CyberDeli application is provided as an example and has not been certified on the DBDS.

Installation

During installation, the application is added to the set of applications currently existing on the DBDS at a particular cable system. Depending on how the application server is configured, the software can be installed on the DBDS Application Server Hardware Platform, or an application-specific machine can be connected to the DNCS network. In future phases, the Flash Packager will allow installation of particular application clients into the flash memory of the DHCT, so that an application client could be installed directly, rather than downloaded at run-time.

The last step of installation is execution of the application server software.

Example: The CyberDeli server application is installed on the Application Server Hardware Platform from a CD.

Continued on next page

Application Lifecycle, Continued

Provisioning

Provisioning is the process that defines the application's services, including the reservation and configuration of system resources needed to provide those services, and the capability to bill for such services.

The first step in provisioning is adding the application client to the BFS and registering it with SAM. An application is identified by a URL and is characterized by several attributes, including:

- Whether or not it stays resident after all of its services are suspended
- Whether or not it receives SAM events (Phase I and II only)
- Whether or not it has a PowerKEY conditional access package to be used for application and service authorization
- Whether or not the application client module is to be loaded into DHCT memory on boot, and optionally launched (Phase II)
- Application-specific configuration information (Phase III)
- An application logo (Phase III)

Ideally the application server can do this provisioning automatically; however, it can be performed manually by a system operator via the BFS and SAM GUI on the Admin Console GUI.

Next, an application server provisions its services, as directed by the system operator via the application server's GUI. Application servers utilize AIS components such as the Session Manager, BCS, Messaging Server, CAM, BFS, and IP Gateway during this process. Once the application's underlying service data is provisioned, the application server uses the SAM to register each service, providing the service parameter(s)—short and long description, optional logo, and alternative services to activate if the service is not authorized. Each service is assigned a service ID. Once the services are defined, they are assigned to channels via the SAM interface. If necessary, service provisioning can be done manually via SAM Server GUIs available as part of the Admin Console GUI.

Continued on next page

Example: The CyberDeli application does not provide a GUI to do provisioning. A conditional access package (Entitlement Unit) is created using the Admin Console Package GUI to control access to the CyberDeli service. Its application client must be added manually to the BFS using the Admin Console BFS GUI. The application server is then registered with Messaging Server as a Value-Added Service Provider (VASP) using the Admin Console VASP GUI, so that it can send and receive DSM-CC PassThru messages. Configuration and pricing files are generated by the CyberDeli server and placed on the BFS using the BFS Server API. The CyberDeli application provides a single service of buying deli sandwiches to the user. Thus, a SAM service is created using the Admin Console SAM Service GUI, and given the description DELI. This service is then added to a channel using the Admin Console Channel Map GUI.

Authorization

Application authorization is controlled by the PowerKEY conditional access system. Service authorization is in general an application-dependent mechanism, but typically relies on the PowerKEY conditional access facilities provided by the DBDS. Both application and service authorization can be performed by designating an Entitlement Unit (PowerKEY package) for conditional access during provisioning. Then each DHCT must be authorized for the package(s) to access the application/service.

The SAM Client provides an interface to query if a service is authorized. If the application providing the service specifies conditional access information when the service is provisioned, the SAM Client utilizes this information to determine service authorization via the CAM Client. Otherwise, the SAM Client must ask the application client if its services are authorized, because only the application has the facilities to answer that question, based on the service parameter. The application in turn can use the CAM interface or another mechanism for authorization. If the application client is not resident in DHCT memory and cannot be asked if its services are authorized, the SAM Client assumes they are not.

Example: During the provisioning stage, a package was created for access to the CyberDeli service. Only DHCTs authorized for this package are able to download the CyberDeli application client; others get a *not authorized* message from the resident Navigator when tuning to the channel where the CyberDeli service is assigned.

Continued on next page

Application Lifecycle, Continued

Request

The request for an application can originate from either an application server or an application client, either as a SAM Signaling DSM-CC PassThru message or as an API call. The most common originator of a request is the resident Navigator application on the DHCT, for a service provided by the application in response to a channel change by the user. The service requested can be defined in the set of existing services (identified by service ID), or it can be a new, dynamic service where the definition and activation request occur simultaneously (identified by an application URL and parameter).

Another request for an application can come as a result of that application being designated as load-on-boot or one of the application's services being designated as activate-on-boot.

Example: The request for the CyberDeli service is originated by the resident Navigator when the user tunes to the CyberDeli channel.

Download

When an application client is requested, it must first be loaded into memory on the DHCT. If not already resident, the SAM Client arranges for the application to be downloaded using the OS Stream Manager API. The application client code is identified by a URL (available on BFS), with resident application clients permanently cached in the flash memory of the DHCT. If an application has been downloaded previously, it may be resident in the DHCT DRAM.

It is possible that to download the requested application, other application client(s) must be terminated and unloaded from memory. The SAM Client tries to remove only applications that are not providing services. In early phases, there are restrictions on the number of dynamically loaded applications.

Conditional Access can be provided for application download if specified when the application is registered with the SAM Server. If so, the SAM Client on the DHCT asks the CAM Client if the application's Entitlement Unit (PowerKEY package) is authorized before the application download proceeds.

Continued on next page

Application Lifecycle, Continued

The download operation is asynchronous so that a suspension request can cancel the activation. If appropriate, the resident Navigator presents a “download barker” to the users to inform them that a service is being loaded. Changing the channel cancels the download.

Once an application client module is loaded into DHCT memory, its initialization function is called by the PowerTV Operating System.

Example: The CyberDeli application client module is downloaded from the BFS.

Launch

After the application client is downloaded, the next step is to launch or run the application executable. When launched, a thread is created for the application client and its *main* function is called by the PowerTV Operating System.

Applications that are designated as auto-launch in their Module Dispatch Table are launched automatically once they are downloaded.

Example: The CyberDeli application is an auto-launch application. It is launched and its *main()* function is called after it is downloaded.

Service Activation

To activate a service requires that the application client is first downloaded and launched, as described in the previous sections. If the application is a PowerTV application module, the SAM Client brings the application into focus. See the *CreativEdge Developer's Reference, Phase <N>* and the *Scientific-Atlanta Resident Application (SARA) Application Developer's Guide* for details. If the application is registered to receive SAM events, a SAM Activate event is sent to the application client, specifying the service ID and the service parameter to be activated. At this point, the application begins providing the service.

Note: Application clients can be designed to be *service reentrant*. Such applications can provide multiple services simultaneously. The most basic applications only provide a single service at a time, so that a subsequent activation automatically causes an existing service to be suspended.

Example: The CyberDeli application client receives a ResumeReq event from the PowerTV OS when it is brought into focus. It can now draw to the screen and present its GUI to the user.

Continued on next page

Execution

While an application client executes, it uses AIS Client components such as the Window Manager, BCS, Session Manager, and Stream Manager to gain access to DHCT and network resources. There are several data paths from server to client through the DBDS network, including the BFS, MPEG private streams, in-band IP, out-of-band IP, and Ethernet (for development purposes). Upstream paths include out-of-band IP and Ethernet (for development purposes). DSM-CC PassThru messaging provides a signaling mechanism from server to client.

Since the DHCT is resource constrained, the AIS Client is designed to support resource sharing in the most efficient manner supported by the hardware, while being able to be expanded as hardware capabilities evolve. Beyond Phase I and II, the AIS Client defines a service resource environment for each service that specifies its current access status for each important resource. Application clients must abide by the service resource environment when activating a service, executing with or without active services. Application clients must be designed to handle all combinations of resource availability gracefully, and should not rely on the availability of a particular resource for proper operation.

See **Chapter 8 - Further Considerations for Application Development** in the *CreativEdge Developer's Reference, Phase <N>* for more details on client resource management.

Example: When first activated, the CyberDeli application client loads menu and configuration information from files on the BFS. It then presents a GUI that allows a user to place an order. Once the order is complete, a socket is opened with the CyberDeli server to process the order (the IP address of the server is in the configuration file). Feedback to the user is given if the order is successful.

Service Suspension

During suspension the SAM Client first receives a request to end a service. The request can originate from either an application server or an application client; the most common originator is the resident Navigator on the DHCT. The particular service is identified by a service ID, and the SAM Client keeps track of application clients currently providing services.

Continued on next page

Application Lifecycle, Continued

The SAM Client checks if the application is currently being downloaded, in which case it terminates the download. If the application is registered for SAM events, the SAM Client sends a Suspend message to the appropriate application client, passing the service ID to suspend. The application client must suspend that service and minimize any resource usage particular to that service. If absolutely necessary, the application client can keep certain resources in anticipation of the service being re-activated. However, the application client can receive resource environment minimization requests ordering the application to free any of the specified resources.

If the application is a PowerTV application, it is brought out of focus when another application is brought into focus as part of a service activation. If the application is not a stay-resident application or was not loaded on DHCT boot, and it is no longer providing any services, and can be terminated as described in the next section.

Example: When the user tunes off of the CyberDeli channel, the application client receives a SuspendReq event. If the CyberDeli application is in the middle of processing an order, it must cancel the order with the server. All window and graphic resources are then freed, as is any memory associated with the service.

Termination

Termination refers to ending the execution of an application client. This is usually initiated by the SAM Client to make room for another application client to be executed in the DHCT. It begins with a request to the application client from the DHCT operating system to free all resources, then quits. Once the application client acknowledges the request, its shutdown method is called and its code is unloaded from DHCT DRAM, unless the application code resides in flash memory.

Note: If an application needs time to clean up resources after receiving the request to quit, it must notify the PowerTV OS by sending a “quit-proceeding” indication. See the *PowerTV Operating System Overview* for details.

Example: When the CyberDeli application client receives a QuitReq event, it frees all resources. It then acknowledges the QuitReq and is removed from the DHCT memory.

Continued on next page

Application Lifecycle, Continued

Un-Installation

All applications are removable from the DBDS; this is referred to as un-installation. During un-installation, all provisioning is undone, the application client code and data removed from the BFS, the application un-registered, then the application server code and hardware platform, if provided, removed from the system.

Example: The CyberDeli service is deleted and removed from the channel map using the Admin Console SAM Service GUI. Its package is deleted using the Admin Console Package GUI. Its VASP ID is deleted using the Admin Console VASP GUI. All of its files, including the application client, are deleted from the BFS using the Admin Console BFS GUI. Finally, the CyberDeli application server quits.

Application Architecture

The basic application architecture includes an application server component, an application client component, and the network interface between the two.

Application Server

The application server is responsible for provisioning the services provided by the application and for providing the content or data needed by the application client. The application server uses the AIS Server component interfaces, and optionally can interface with the resident Navigator Server. It can also provide a GUI for configuration by the system operator.

The application server can execute on the DBDS Application Server Hardware Platform or on a dedicated hardware platform provided with the application. The choices of server hardware are application-specific and outside the scope of this guide. In order to use the AIS Server interfaces, the hardware platform must support ONC/RPC. The server hardware platform can interface with the following headend components:

- QAM
- QPSK
- BIG
- DNCS Hardware Platform

Application Client

The application client is the portion of the application that executes on the DHCT. It provides the application's services to the subscriber, usually through a graphical user interface. It interfaces with the AIS Client components and optionally with the resident Navigator Client. Application clients must abide by the client resource management protocol prescribed by the AIS—see **Chapter 8 - Further Considerations for Application Development** in the *CreativEdge Developer's Reference, Phase <N>* for more details.

Network Interface

Data needed by an application client is usually stored on the server and transmitted continuously or on demand to a specific EXPLORER DHCT. The AIS provides access to several different mechanisms and protocols for delivering data between the application server and client.

Continued on next page

Broadcast File System (BFS)

The BFS provides a network file system for delivering broadcast data from server to client. Data placed in the BFS is repeatedly transmitted over the network, providing virtual storage for the DHCT. This allows an application client to quickly access the data at any time without having to request the data from the server. This mechanism is useful where all DHCTs running the application use the same data. An example is a stock quote and statistics application that makes the same stock information available to any DHCT that has access to the application. Rather than sending the information to each DHCT or having each DHCT request the information, the application server updates the data once in the BFS and it is then broadcast over the network to the DHCTs.

BFS files can be transmitted in-band over QAM or out-of-band over QPSK. Application clients access files over the BFS using the PowerTV Stream Manager interface.

MPEG Private Data

An application client can retrieve raw data sent in MPEG-2 private sections. This is appropriate when the data is transmitted along with video over the same transport stream or when the application server does not want to use BFS. It is up to the application server and client to format and interpret the data transmitted in such private sections.

A client application uses the PowerTV Stream Manager to access MPEG-2 private stream data. Before an application can open and retrieve such data, the application must ensure that the DHCT is tuned to the appropriate frequency and receiving the correct MPEG PSI data.

IP Data

There are several means for delivering IP data to and from a DHCT, all usually accessed via the Socket API. These interfaces include:

- Out-of-band RF over QPSK (forward/reverse)
- In-band RF over QAM (forward)
- Ethernet (development only)

The IP address RF interface is assigned when the EXPLORER DHCT boots and receives a DSM-CC UNConfig indication from the DNCS. This IP address is private to the network between DNCS and the EXPLORER DHCT and is not a valid Internet IP address. The development version of PowerTV Operating System allows an IP address to be assigned to the Ethernet interface for development purposes. Presently, in a production environment, an IP address is not assigned to the Ethernet interface.

Continued on next page

Internet protocol data over in-band is a feature provided by the IP Gateway component of the AIS. This feature allows sending downstream IP packets over a QAM channel, embedded in MPEG packets as DSM-CC Addressable Sections. The upstream data is delivered out-of-band. Application servers must configure IP Gateways during provisioning or choose from the default set provided by the Gateway Server. The application client sets up a connection with an IP Gateway before opening a socket. Once the socket is opened, the transport mechanism is transparent to the application.

Unlike in-band data, which is only available when tuned to the appropriate QAM frequency, out-of-band data is always available to applications. Because downstream out-of-band bandwidth is often scarce in typical cable networks, it is recommended that applications take advantage of the IP Gateway for downstream IP connectivity. However, upstream or reverse path data is always transmitted out-of-band.

DSM-CC PassThru Messaging

DSM-CC PassThru messages are used for sending messages from an application server to an application client and vice versa. PassThru messages are sent via UDP, so reception is not guaranteed by the protocol. In a one-way network, this is the only way to send messages from an application server to a client. Even in a two-way network, this method is preferred over IP datagrams because it does not require the overhead of a socket connection, and the DNCS can provide rate control to improve the performance and reliability of the network.

Before an application can use PassThru messages, the application developer must reserve a PassThru Message Type by contacting PowerTV, Inc. This is a statically assigned number between 0x8000 and 0xFFFF, used to identify messages belonging to an application.

Continued on next page

Application Architecture, Continued

VBI / Closed-Captioning Data

VBI is a one-way path for data broadcast along with the analog video signal. The API for application to access to the VBI data is part of the PowerTV Stream Manager. Closed-Captioned Data is the digital counterpart to VBI data, broadcast as part of an MPEG stream including digital video. The Stream Manager can access it as well.

DSM-CC Session Protocol

The DSM-CC Session Signaling protocol establishes and manages interactive sessions for on-demand data or video sessions. A session is used to manage bandwidth and network utilization, and is not a data delivery protocol itself. For example, the IP Gateway establishes a session as part of an application's usage of in-band IP delivery.

Applications use the Session Manager component of AIS for DSM-CC session signaling. The Admin Console Session GUI allows sessions to be set up by the system operator manually.

Chapter 4

Application Interface System (AIS)

Overview

Introduction

The Application Interface System (AIS) is the collection of end-to-end software components enabling applications on the EXPLORER DBDS. The AIS provides an interface and mechanism for applications to be developed, introduced on the network, defined as a service, downloaded to the DHCT, executed on the DHCT, and removed from the network. AIS includes documentation, APIs, and development and installation tools, in addition to the actual AIS module code.

In this Chapter

This chapter contains the following topics.

Topic	See Page
Overall Architecture	4-2
Application and Service Support	4-5
Resource Access and Management	4-8

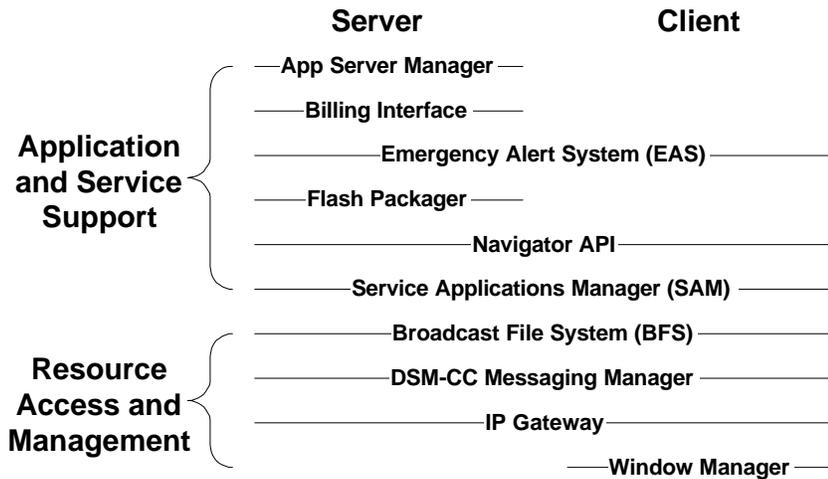
Overall Architecture

The AIS is an end-to-end layer (client and server) of application and service management software components that an application *must* utilize to exist in the EXPLORER DBDS. It also consists of resource access and management components usually required by applications in the DBDS. None of the AIS components have any client-side user-interface, although GUIs may exist to access server-side components.

Although the AIS does not include HTML or Java components directly, its interface is available to middleware applications via each middleware engine's API—sometimes through a thin adapter layer or even as a native function call.

AIS Components

The following figure lists the software components of the AIS. These components can be grouped into two categories: Application & Service Support and Resource Access & Management. The next sections give a high-level description of component functionality and interfaces.



Continued on next page

Overall Architecture, Continued

Middleware Support

The following sections describe middleware support that has been built on top of AIS and DNCS functionality.

HTML/JavaScript Engine

An HTML/JavaScript Engine allows services to be specified entirely in HTML, with the HTML interpreter as the application and the HTML content as the parameter. Native application clients can also utilize the HTML Engine interface as a parser, and display the result themselves. In this way, application clients can support modified HTML to meet their own needs. An HTML Engine also provides application clients direct access to content decoders for common MIME-types, such as JPEG images or AIFF audio.

The W3C HTML specification is being extended to support program synchronous data in the broadcast environment, through standards efforts in ATVEF and DASE.

An HTML Engine utilizes core AIS components to access DHCT resources, such as BFS and Window Manager. By invoking HTML-based services using the SAM, HTML applications can co-exist with native applications and Java applications in the DHCT.

HTTP Proxy

The HTTP Proxy is the HTTP link between the DBDS and outside networks, such as the Internet. It provides access to objects outside the DBDS using HTTP. These objects are usually referenced in HTML documents being displayed on the DHCT client. The objects can be other HTML documents, Java applets, or the like. The HTTP Proxy also provides any filtering, translation, and authentication necessary for the objects to be understood by the applications on the DBDS.

The HTTP Proxy is a server component that usually utilizes the IP Gateway to deliver content to clients on the DHCT, such as the HTML Engine or other applications.

Continued on next page

Overall Architecture, Continued

Personal Java (pJava)

The Personal Java (pJava) libraries and Java Virtual Machine (JVM) are a long-term platform for application clients. This allows application clients to be written in Java downloaded, then executed on the DHCT. Part of the pJava interface includes *JavaTV* libraries to access the DHCT cable-related functionality, as provided by components of the AIS.

The pJava virtual machine and libraries utilize core AIS components to access DHCT resources, such as BCS and Window Manager. By invoking Java services using the SAM, Java applications can co-exist with native applications and HTML-based applications in the DHCT. A Java-based service can be defined as the URL for the Java virtual machine as the application, and the URL for the Java application byte-code as the parameter, or a native application can execute the JVM itself as an API call.

Application and Service Support

This section describes components of AIS that provide application and service support. Some components described here are not available in early phases of the DBDS.

Application Server Manager

The Application Server Manager is the registry for application servers residing on the DBDS. Through the App Server Manager GUI, the GUI for all application servers can be accessed. In early phases, only application servers that reside on the Application Server Hardware Platform need to use the Application Server Manager. In the future, there may be a collection of HTML links to application server GUIs. Application servers may be required to implement an interface specified by the Application Server Manager, for determining information such as available services. This interface may be specified using CORBA or Java.

Billing Interface

The Billing Interface allows an Administrative Gateway or Billing System to communicate with application servers for billing purposes. Through the interface, the Billing System is notified of billable services offered by application servers, and the Billing System can communicate with an application server to identify billable services. Currently, the Billing interface primarily supports a Pay-Per-View model.

Emergency Alert System (EAS)

The Generic Emergency Alert System Receiver server publishes an interface through which EAS hardware (EAS Controllers) can be connected to the DBDS to transmit EAS messages. The Generic EAS Receiver broadcasts any alert messages it receives over the DBDS network as an out-of-band DSM-CC Pass-Thru message, including the alert type, alert text, and alert audio. The audio content can be placed on BFS at a specified URL. This message can be received by a Navigator application for presentation to the DHCT user. Application clients other than the Navigator do not need to handle EAS messages; the use of the Window Manager allows them to be presented without application knowledge.

As the protocol for EAS transmission in a cable network is standardized, the EAS implementation will be changed to meet new requirements.

Flash Packager

The Flash Packager allows several application clients to be packaged together and downloaded into the DHCT flash memory, thus making them resident applications.

Continued on next page

Navigator API

The Navigator API includes access to information controlled by the Navigator application resident in the DHCT. The Navigator application is the default application that presents the service selection paradigm to the user. Additionally, the Navigator allows a user to access and set core DHCT information; this information can be read-only to applications other than the existing Navigator. The Navigator API provides access to the following information or functionality:

- Current channel, service, and status information (blocked, favorite)
- Blocked and favorite service lists
- Blocked ratings and times
- Parental control status
- Language selection (UI and audio)
- NVM timers (VCR, Reminder)
- Purchased PPV events
- Miscellaneous user settings: VCR type, audio level, blocking PIN, purchase PIN, AC outlet mode, clock display mode, etc.

The Navigator API on the client provides access to these settings, most being stored in persistent storage on the DHCT. The Navigator API on the server gives the ability to access and change these settings from the server. It is possible that Navigator settings can be stored on the server in two-way networks.

Part of the Navigator API is the mechanism for an application client to register interest in events notifying it of a change in state of the user settings. These events are available for the following settings:

- Channel change
- Parental control status changed
- Language selection changed

Service Application Manager (SAM)

The traditional TV definition of a channel is an analog broadcast video source. In the digital world, in which the set-top box is actually a fairly powerful computer, the situation is not as simple. In addition to supporting analog broadcast video, we also must support things that are *not* analog (digital), *not* broadcast (two-way communication), and *not* video (such as a web browser). In order to provide these more powerful and complex features, the simple channel abstraction needs to be extended.

Continued on next page

Application and Service Support, Continued

The Service Manager (SAM) provides a model in which the user can access *services*, which consist of an application to run and a parameter, such as data content, specific to that service. Many services can be defined using the same application component, with different parameters. For example, an application that can tune video programming could be executed with one set of parameters to view HBO and a separate set of parameters to view CNN. Each association of the application component (tune video) and one of the parameter components (HBO and CNN) represent a particular service and has a unique service ID. The following table gives some examples of services.

DCN	Application	Parameter	Short Description
2	Video	NBC source ID	NBC
3	Video	CBS source ID	CBS
4	Video	HBO source ID	HBO
51	Video	CNN source ID	CNN
59	Video	E! source ID	E!
71	PPV	movie of week	PPV1
72	PPV	Spice source ID	PPV2
80	VOD		VOD
101	Music	Country source ID	CNTRY
102	Music	Classical source ID	CLASS
103	Music	Rap source ID	RAP
104	Music	Metal source ID	METAL
200	Ticker	Stocks	STOCK
201	Ticker	Sports	SPORT
250	Weather		WTHR
301	Catalog 1		CAT1
302	BBTV		BBTV
303	USPS		USPS
350	SPRE	store 1	SPRE1
351	SPRE	store 2	SPRE2
352	SPRE	store 3	SPRE3
400	Browser	Internet	WEB1
401	Browser	Intranet	WEB2
501	Language	Spanish	SPAN
502	Language	Japanese	JAPAN

With the capabilities of a two-way digital network, other applications such as text channels, pay-per-view, video-on-demand, and web browsing become possible. In addition to an application and a parameter, each service also has an identity, which includes a short textual description (call letters), a long textual description, and a logo.

Because consumers are comfortable with the current cable TV channel paradigm, the SAM provides the notion of a *display channel* as a way to access services. A display channel is the channel number a person sees on the TV set, and is used to specify the current service. To support existing analog cable systems, the SAM also provides *split channel* capability, meaning that two different services can be displayed on a single channel at different times of the day.

Continued on next page

The SAM service paradigm also represents the unit by which resources are shared in the DHCT. In other words, each active service in the client has a particular resource environment specifying that service's degree of access to all controlled resources. See **Chapter 8 - Further Considerations for Application Development** in the *CreativEdge Developer's Reference* for a more detailed description of the client resource management model.

The SAM provides several important features:

- A mapping from the display channel number (presented to the user) to the service, and vice-versa, including split channel capability
- An interface on the server to specify service related data, and on the client to access this information efficiently
- A mechanism whereby applications and services on the client can be activated/suspended remotely from the server, via a signaling message
- A mechanism to determine whether an application's client software is loaded on the DHCT and, if not, arrange for it to be acquired from the server and loaded
- The means to apply Conditional Access to application client download and service authorization, so that only particular subscribers can access certain applications or services
- The means to designate an application client module to be loaded on boot of the DHCT, and optionally launched
- Application versioning such that new versions of the application be seamlessly added to the system
- For each service, an alternative service to activate if the original service is not authorized
- An interface on the server to store application configuration data for access by the application client

All applications are required to use the AIS SAM to provide and access services. It is the only way that application clients can be executed on the DHCT within the AIS platform.

Resource Access and Management

This section describes components of AIS that provide network and resource management. Some components described here are not available in early phases of the DBDS.

Broadcast File System (BFS)

The Broadcast File System (BFS) provides a mechanism for delivering various types of data from a group of servers to a client that is attached to a network. This data can contain any type of information. Both the server and client view the data as a file system similar to those found on disk operating systems.

A client doesn't have enough resources to store all the data that is being broadcast over the BFS. Even if the client could store all of the data, there is no guarantee that the client can receive an error-free copy of the data in a single transmission. In a broadcast environment, the client has no way to request that a server re-send any data that was missed or received in error. Also, since the data is being sent to many clients, it might be prohibitive to require that the server re-send missed data to each client that requests it. To ensure that all clients are able to receive an error free copy of the data, the BFS server repeatedly sends the data over a period of time so that a client that is interested in the data may receive it only when it is required. If the client misses some of the data, or the data has an error, it waits for the next broadcast of the data to receive any data that it may need.

The BFS is implemented to appear to the servers and clients as a standard hierarchical file system that is common in computer operating systems. The underlying mechanism for transporting files from the servers to a client relies on a broadcast data carousel mechanism which is defined in the MPEG-2 DSM-CC specification (ISC/IEC 13818-6). Uniform Resource Locators (URL) specify bfs as the protocol identify files on the carousel.

Client View of Broadcast File System

An application client accesses the BFS as if it were a local file system. The actual client interface to the BFS is through the Stream Manager, described in a later section. When the BFS is being delivered over the broadcast access network, the servers and files appear as a hierarchical file system. As servers log into the BFS, server entries are added under the root BFS directory. As additional directories and files are added by servers, they appear in that server's directory tree. When operating in one-way mode, a client can access a file in read-only mode only.

Continued on next page

Resource Access and Management, Continued

Server View of Broadcast File System

An application server accesses the BFS server using commands similar to those used to access a disk file system. Before operating as part of the BFS, the server must register with the BFS server. After a server has registered with the BFS server, it can add, modify, and delete directory entries and files as if the BFS is a local file system.

The server can read the directory and file system from the BFS server, however, the server's view of the BFS will contain only the files which appear under that server's directory tree.

If a server has chosen to deliver the files directly, rather than the BFS server, it is not be able to read those files from the BFS server. It can, however, read the directory and file tree structure.

DSM-CC Messaging Manager

The Messaging Manager provides an interface to send DSM-CC User-to-Network Pass-Thru messages. A pass-thru message can be addressed to a single DHCT or broadcast to a group of DHCTs. On the client side, an application client can register for particular Pass-Thru messages that its application server is known to send. For example, a PPV application server can broadcast a message to PPV application clients when the PPV event definition file has changed. A registry of reserved Pass-Thru message IDs is maintained by PowerTV, Inc.

Note: An application server must be registered as a Value-Added Service Provider (VASP) with the DNCS to use the Messaging Manager.

IP Gateway

An IP Gateway is a collection of DSM-CC continuous feed sessions that provide a particular downstream quality of service (bandwidth) for in-band IP transmission and the paths that provide an upstream quality of service using a particular reverse path protocol.

Applications can use a gateway to provide the actual TCP-UDP/IP socket connectivity between the application server and client, downstream over QAM and upstream over QPSK.

Continued on next page

Server

The definition of IP gateways can happen in two ways, depending on the system operator's preference and network. The IP Gateway Server interface allows the system operator to set up default public gateways providing specified downstream bandwidth over DSM-CC continuous feed sessions to the different hubs in the network. For the initial implementation, the upstream portion of a gateway uses the slotted aloha mechanism. Depending on the applications the system operator wants to support, different QoS gateways can be created. The default set of gateways is made available to applications as a public set of gateway IDs.

A private gateway can also be set up and reserved for use by a particular application. This can be done through the IP Gateway Server GUI, or through the Application Interface System (AIS) API into the IP Gateway Server.

An application server is responsible for communicating to its application client the gateway ID(s) to use and any other application specific information required for the application client to choose which gateway to connect to using the IP Gateway Client.

When the IP Gateway Server is asked by the IP Gateway Client to open a gateway connection, the IP Gateway Server asks the Session Manager to choose the appropriate downstream path (continuous feed session) out of the set of sessions already set up for that gateway.

Client

The IP Gateway Client interface allows an application client to open a gateway connection given a gateway ID. The IP Gateway Client opens an exclusive session with the IP Gateway Server, specifying the gateway ID and the DHCT's MAC QAM address. Via this session, the IP Gateway Client receives the IP address and subnet mask of the client end of the gateway, and the QAM frequency, modulation mode, and program number of the MPEG stream containing the downstream IP traffic.

The IP Gateway uses a predetermined stream type with which the IP Gateway Client can parse the PMT to determine the PID of the IP data. The IP Gateway Client then opens the MPEG transport stream and sets up the DHCT filters to look for only MPEG packets destined for the DHCT's QAM MAC address. The IP Gateway Client takes care of extracting the DSM-CC addressable section from the MPEG transport, extracting the IP datagrams from the DSM-CC section, and delivering the datagrams to the TCP/IP stack. The IP data then gets delivered to the application client's socket.

Continued on next page

Resource Access and Management, Continued

The IP Gateway Client can also allow a cable modem application to configure it as a simple router for basic packet forwarding through the DHCT's Ethernet port. The assigned IP address for the DHCT's Ethernet port is entered into the DNCS as part of the DHCT configuration. This IP address can be on the same subnet as the subscriber's PC or home network. This IP address is sent to the DHCT in UN-Config. This IP address would also be communicated to the IP Gateway server so that it is associated with the DHCT's QAM MAC address. This allows the system to route to that subnet over an IP Gateway. If the DHCT receives a packet through the QAM interface that is not destined for an internal interface, it routes it out of the Ethernet port. If the DHCT receives a packet from its Ethernet port that is not destined for the DHCT, it sends it through the upstream data channel.

Window Manager

The Window Manager provides the mechanism for sharing screen real estate and user input. The Window Manager on the client DHCT is responsible for the creation, display, and de-allocation of the limited DHCT screen resources. It allows multiple application clients to share the screen by assigning ownership of screen regions, or windows. The Window Manager client allows windows to overlap and supports windows with special attributes such as backdrop, private, child, or video. Window Manager client also allows application clients to register interest in user-input events and routes input to the appropriate applications using a filtering and prioritization scheme. All application clients that want to draw to the screen or receive user input are required to use the AIS Client Window Manager.

Note: In Phase I and II, application clients can use PowerTV OS Screen Manager APIs to draw to the screen.

Chapter 5

Digital Network Control System (DNCS)

Overview

Introduction

The software components of the Digital Network Control System (DNCS) provide complete management, monitoring, and control of DBDS network elements and broadcast services provided to subscribers. This includes the definitions of sources and DSM-CC sessions, UNconfiguration of DHCTs in the network, and conditional access management. An Administrative Console GUI allows system operators to access the DNCS. Client-side interfaces to DNCS functionality are available via the PowerTV Operating System.

In this Chapter

This chapter contains the following topics.

Topic	See Page
DNCS Components	5-2
Broadcast Control Suite (BCS)	5-2
Conditional Access Manager (CAM)	5-2
DSM-CC Session Manager	5-3

DNCS Components

The following sections describe the components of the DNCS from an application perspective. Details of element management are not included.

Broadcast Control Suite (BCS)

The Broadcast Control Suite (BCS) provides an interface for applications to setup and access broadcast-oriented network resources such as sources and segments. A source is identifiable digital content that on a DSM-CC session, such as a broadcast MPEG program. A segment is then a time-bounded portion of a source. BCS manages the defined resources and provides the necessary information to access these sources to its corresponding AIS Client component.

On the client side, the following broadcast resources are available:

- Analog video & audio tuning API
- Digital (MPEG) video & audio decoding API
- Direct access to MPEG transport streams
- Access to VBI data

The PowerTV Operating System TV Manager, Stream Manager, and MPEG Transport APIs provide access to the BCS functionality.

Conditional Access Manager (CAM)

The CAM manages subscriber access to services supplied via the DBDS through the use of purchasable units called packages. The package is the unit of authorization on which the CAM grants permission to use the defined segments; a given package may contain any number of segments. When a DHCT is granted authorization (permission) for the package, that user has the right to use any of the segments contained in the package.

Packages and segments are organized as a tree structure in which segments represent leaves and packages represent all other nodes. Packages may contain other packages. This has the effect of adding segments from the contained package to the membership of the containing package. When a user attempts to use a segment, the CA attributes for the package through which the user gains permission to use the segment apply.

Conditional access information is specified for each package, and this CA information is used in the construction of Entitlement Control Messages (ECMs) distributed to the network within the bandwidth of the defined segments. Unlimited duration packages are used to provide CA for download or access to a particular application or service.

Continued on next page

DNCS Components, Continued

The CAM is considered the authority on the existence and definition of packages and on the existence and definition of package authorizations. The CAM maintains authorizations on a per-DHCT basis. For each DHCT, a list of packages for which the DHCT is authorized is maintained.

DSM-CC Session Manager

The Session Manager works with other components of the DNCS in order to support delivery of services in the Scientific-Atlanta Digital Broadband Delivery System (DBDS). The Session Manager processes DSM-CC User to Network Session signaling messages, manages allocation of session related network resources and supports network management operations.

The services supported may be broken down into two categories - Broadcast Services and Exclusive Services. Broadcast services are supported by network resources that are relatively static over time and that generally are visible to all or large subsets of clients of the network. This includes the following services:

- Digital Broadcast
- Pay Per View (PPV) Digital Broadcast
- Impulse Pay Per View (IPPV) Digital Broadcast
- Near Video On Demand (NVOD) Digital Broadcast
- Electronic Program Guide
- Broadcast File system
- In Band Internet Protocol Gateway (IBIP)

The Session Manager supports broadcast services by providing the signaling interface to establish, maintain, and release Continuous Feed Sessions (CFS). The user semantics that distinguish these services from one another are not identifiable from the Session Manager perspective. Utilizing bandwidth reservations, providing conditional access and distributing system information for these services are not within the scope of the Session Manager. The Session Manager acts as a point of contact to the network for servers to establish sessions.

Exclusive services are supported by associating a single DHCT client to the network resources that carry a particular service. These network resources can be devoted to a single client or can be shared among many clients, where each client is actively bound over to a shared network resource. This includes the following services:

- Video On Demand (VOD)
- In Band Internet Protocol Gateway (IBIP)

Continued on next page

DNCS Components, Continued

The Session Manager supports exclusive services by providing the signaling interface to establish, maintain and release Client Initiated Exclusive Sessions. The user semantics that distinguish these services from one another are not identifiable from the Session Manager perspective. Utilizing bandwidth reservations, providing security and distributing system information for these services are not within the scope of the Session Manager. The Session Manager acts a point of contact to the network for clients to establish sessions.

Chapter 6

Customer Information

Overview

Introduction

This chapter contains information on obtaining product support and returning damaged products to Scientific-Atlanta.

In this Chapter

This chapter contains the following topics.

Topic	See Page
Customer Support	6-2
Returning Products	6-3

Customer Support

List of Telephone Numbers

If you have questions about this product, contact your distributor or sales agent for information. If further assistance is required, telephone your nearest Scientific-Atlanta office at one of the following telephone numbers.

The Americas		
United States	Scientific-Atlanta Technical Assistance Center, Atlanta, Georgia	From <i>within North America</i> <ul style="list-style-type: none">• All Other Products 1-800-722-2009 (toll-free)• Digital Products 1-800-283-2636 (toll-free) From <i>outside North America</i> +1-770-903-5400 (direct)
United Kingdom and Europe		
United Kingdom	Kings Langley	<ul style="list-style-type: none">• +44-1-923-266133• +44-1-923-271420 (Technical Assistance Centre for Europe)

Returning Products

Procedure

To return any Scientific-Atlanta product for repair or replacement, follow the steps in the following table. Products **must** have an RMA number to receive credit.

1. Telephone or fax Scientific-Atlanta and request a **return material authorization (RMA) number**.
 - **From within the U.S.**
 - Tel: 1-800-722-2009
 - Fax: 770-903-5888
 - **From outside the U.S.**
 - To the United States
Tel: +1-770-903-5300
Fax: +1-770-903-5888
 - To the United Kingdom
Tel: +44-1-923-271460
2. Tag or identify the defective product and write a detailed description of the circumstances.

Include the following information on the tag:

 - RMA number
 - Sales order
 - Purchase order (if available)
 - Date the product was received
3. Pack the product in its original container and protective packing material.

Note: If the original container and packing material are no longer available, pack the product in a sturdy, corrugated box and cushion it with packing material.

Continued on next page

Returning Products, Continued

4. Write the following information on the **outside** of the container:

- Your name
- Complete address
- Telephone number
- RMA number
- Problem description

Note: Absence of the RMA number may delay processing your product for repair. Include the RMA number in all correspondence.

5. Ship the product, prepaid and insured, via your *postal service* to the following address:

Scientific-Atlanta, Inc.
RMA Number _____
Factory Services
4245 International Boulevard
P. O. Box 6850
Norcross, GA 30091-6850
USA

Ship the product, prepaid and insured, via United Parcel Service (UPS) or other *freight carrier* to the following address:

Scientific-Atlanta, Inc.
RMA Number _____
Factory Services
4245 International Boulevard
Norcross, GA 30093
USA

Note: Scientific-Atlanta, Inc. does not accept freight collect. Be sure to prepay all shipments.

© 1999 Scientific-Atlanta, Inc. All rights reserved.

Index

Acronyms, 1-3

AC-3, 1-3
Administrative Gateway. *See* AG
Advanced Television System Committee.
 See ATSC
AG, 1-3
AIS, 1-3
AIS Components, 4-2
API, 1-3
Application and Service Support, 4-5
Application Architecture, 3-14
Application client, 3-1
Application Client. *See* Application
 Architecture
Application development, 3-5
Application Interface System. *See* Software
 Overview. *See* AIS
Application Interface System (AIS), 4-1
Application Lifecycle, 3-4
Application server, 3-1
Application Server. *See* Application
 Architecture
Application Server Manager, 4-5
Applications, 2-8
ASI, 1-3
ATSC, 1-3
Audio Compression Level 3. *See* AC-3
Authorization. *See* Application Lifecycle

BFS, 1-3

BIG, 1-3
Billing Interface, 4-5
BOSS, 1-3
Broadband Integrated Gateway. *See* BIG
Broadcast Control Suite (BCS). *See* DNCS
 Components
Broadcast File System. *See* BFS

Broadcast File System (BFS, 4-9. *See*
 Network Interface
Business and Operations Support System.
 See BOSS

CAT, 1-3

CA, 1-3
Certification. *See* Application Lifecycle
Client, 4-11
Client View of Broadcast File System, 4-9
Conditional Access. *See* CA
Conditional Access Manager (CAM). *See*
 DNCS Components
Conditional Access Table. *See* CAT
Core Technologies, 2-2
Customer Support, 6-2

DAVIC, 1-3

DBDS, 1-3
DCN, 1-3
DCT, 1-3
Development. *See* Application
 Development
Development Environments, 2-9
DHCT, 1-3
Digital Audio Visual Council. *See* DAVIC
Digital Broadband Delivery System. *See*
 DBDS. *See* DBDS
Digital Home Communications Terminal.
 See DHCT
Digital Home Communications Terminals.
 See DHCT
Digital Network Control System. *See* DNCS
Digital Storage Media - Command and
 Control. *See* DSM-CC
Digital Video Broadcasting. *See* DVB
Display Channel Number. *See* DCN

Index, Continued

DNCS, 1-3, 5-1. *See* Digital Network Control System
DNCS Components, 5-2
Download. *See* Application Lifecycle
Downloadable Applications, 3-3
DSM-CC, 1-3
DSM-CC Messaging Manager, 4-10
DSM-CC PassThru Messaging. *See* Network Interface
DSM-CC Session Manager. *See* DNCS Components
DSM-CC Session Protocol. *See* Network Interface
DVB, 1-3

ECM, 1-3
EDT, 1-3
Emergency Alert System (EAS), 4-5
EMM, 1-3
Entitlement Control Message. *See* ECM. *See* ECM
Exception Dispatch Table. *See* EDT
Execution. *See* Application Lifecycle
EXPLORER DBDS, 3-1, 4-1
EXPLORER DHCT, 2-4

Flash Packager, 4-5

Graphical User Interface. *See* GUI
GUI, 1-3

Hardware Overview, 2-4
HCT, 1-3
Headend Components. *See* Hardware Overview
Headend in the Sky. *See* HITS

HFC, 1-4
HITS, 1-4
Home Communications Terminal. *See* HCT or DHCT
HTML/JavaScript Engine, 4-3
HTTP Proxy, 4-3
Hybrid Fiber Coax. *See* HFC

ICM, 1-4
Impulse Pay-Per-View. *See* IPPV
Installation. *See* Application Lifecycle
Integrated Receiver Decoder. *See* IRD
Integrated Receiver Transcoder. *See* IRT
Integrated Service
 Commission/International
 Electrotechnical Commission. *See* ISC/IEC
Integration and Testing. *See* Application Lifecycle
Interactive Program Guide. *See* IPG
Internet Protocol. *See* IP
Inverse Channel Map. *See* ICM
IP, 1-4
IP Data. *See* Network Interface
IP Gateway, 4-10
IPG, 1-4
IPPV, 1-4
IRD, 1-4
IRT. *See*
ISC/IEC, 1-4

LAN Interconnect Device. *See* LID
Launch. *See* Application Lifecycle
LID, 1-4
List of Telephone Numbers, 6-2

MAC, 1-4

Management Information Broker. *See* MIB
Media Access Control. *See* MAC
MIB, 1-4
Middleware Support, 4-3
Middleware-Based Applications, 3-2
Moving Pictures Experts Group. *See* MPEG
MPEG, 1-4
MPEG Private Data. *See* Network Interface
MSK, 1-4
Multisession Key. *See* MSK

National Television System Committee. *See*
NTSC
Native Applications, 3-2
Navigator, 3-2
Navigator API, 4-6
Network Development Kits (NDK. *See*
Development Environments
Network Information Table. *See* NIT
Network Interface. *See* Application
Architecture
Network Overview, 2-2
Network Service Access Point. *See* NSAP
NIT, 1-4
Non-Volatile Memory. *See* NVM
NSAP, 1-4
NTSC, 1-4
NVM, 1-4

OC-3, 1-4
On-The-Fly Settings. *See* OTF
OOB, 1-4
Operating System. *See* OS
Optical Carrier Level 3. *See* OC-3
OS, 1-4
OTF, 1-4
Out-of-Band. *See* OOB

Overall Architecture, 4-2
Packetized Elementary Stream. *See* PES
PAL, 1-4
PAT, 1-4
Pay-Per-View. *See* PPV
Personal Identification Number. *See* PIN
Personal Java (pJava), 4-4
PES, 1-5
Phase Alternating Line. *See* PAL
PIN, 1-5
PMT, 1-5
PowerTV OS, 2-7
PPV, 1-5
Procedure for returning products, 6-3
Program Association Table. *See* PAT
Program Map Table. *See* PMT
Program Specific Information. *See* PSI
Provisioning. *See* Application Lifecycle
PSI, 1-5

QAM, 1-5
QPSK, 1-5
Quadrature Amplitude Modulation. *See*
QAM
Quadrature Phase-Shift Keyed. *See* QPSK

Real Time Encoder. *See* RTE
Related Documents, 1-2
Remote Procedure Call. *See* RPC
Request. *See* Application Lifecycle
Resident Applications, 3-2
Resource Access and Management, 4-9
Returning Products, 6-3
RPC, 1-5
RTE, 1-5

Index, Continued

SAM, 1-5
SARA, 1-5. *See* Scientific-Atlanta Resident Application
SCH, 1-5
Schematic. *See* SCH
Scientific-Atlanta Resident Application. *See* SARA
SCTE, 1-5
SDT, 1-5
Server, 4-11
Server View of Broadcast File System, 4-10
Service Activation. *See* *Application Lifecycle*
Service Application Manager. *See* SAM
Service Application Manager (SAM), 4-6
Service Definition Table. *See* SDT
Service Gateway. *See* SG
Service Suspension. *See* *Application Lifecycle*
SG, 1-5
Simple Network Management Protocol. *See* SNMP
SNMP, 1-5
Society of Telecommunications Engineers. *See* SCTE
Software Development Kits (SDK). *See* *Development Environments*
Software Overview, 2-5
SONET, 1-5
STM-1, 1-5
Supported Media, 2-3
Synchronous Optical Network. *See* SONET
Synchronous Transfer Mode Level 1. *See* STM-1

TCP/IP, 1-5

Termination. *See* *Application Lifecycle*
Transport Control Protocol/Internet Protocol. *See* TCP/IP
Types of Applications, 3-2

UDP/IP, 1-5

UI, 1-5
Understanding the DBDS Network, 2-2
Un-Installation. *See* *Application Lifecycle*
Universal Resource Locator. *See* URL
URL, 1-5
User Datagram Protocol/Internet Protocol. *See* UDP/IP
User Interface. *See* UI

Value Added Service Provider. *See* VASP

VASP, 1-5
VBI / Closed-Captioning Data. *See* *Network Interface*
VCT, 1-5
Virtual Channel Table. *See* VCT

Window Manager, 4-12



United States: Scientific-Atlanta, Inc., 4261 Communications Drive, P. O. Box 6850, Norcross, GA 30091-6850; Tel: 770-903-5000; TWX: 810-799-4912; Telex: 0542898

Europe: Scientific-Atlanta Europe, Ltd., Home Park Estate, Kings Langley, Herts WD4 8LZ, England; Tel: +44-1-923-266133; Fax: +44-1-923-269018

Asia-Pacific: Scientific-Atlanta (HK), Ltd., Suite 56-57, 5/F New Henry House, 10 Ice House Street, Central, Hong Kong; Tel: 852-2522-5059; Fax: 852-2522-5624

© 1999 Scientific-Atlanta, Inc.

All rights reserved.

Printed in USA

Part Number 714580 Rev 003

August 1999