Level II Toolkit
Overview
Summary of Changes

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Reason/Rational</th>
<th>Nature of Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Notice

Cabletron Systems reserves the right to make changes in specifications and other information contained in this document without prior notice. The reader should in all cases consult Cabletron Systems to determine whether any such changes have been made.

The hardware, firmware, or software described in this manual is subject to change without notice.

IN NO EVENT SHALL CABLETRON SYSTEMS BE LIABLE FOR ANY INCIDENTAL, INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING BUT NOT LIMITED TO LOST PROFITS) ARISING OUT OF OR RELATED TO THIS MANUAL OR THE INFORMATION CONTAINED IN IT, EVEN IF CABLETRON SYSTEMS HAS BEEN ADVISED OF, KNOWN, OR SHOULD HAVE KNOWN, THE POSSIBILITY OF SUCH DAMAGES.

Copyright © September 1998, by Cabletron Systems, Inc. All rights reserved.

Printed in the United States of America.

Order Number: 9030621 E8

Cabletron Systems, Inc.
P.O. Box 5005
Rochester, NH 03866-5005

SPECTRUM, the SPECTRUM IMT/VNM logo, DCM, IMT, and VNM are registered trademarks, and SpectroGRAPH, SpectroSERVER, Inductive Modeling Technology, Device Communications Manager, and Virtual Network Machine are trademarks of Cabletron Systems, Inc.

Adobe and Acrobat are trademarks of Adobe Systems, Inc.
C++ is a trademark of American Telephone and Telegraph, Inc.
UNIX, OSF/1 and Motif are registered trademarks of The Open Group.
X Window System is a trademark of the X Consortium.
Ethernet is a trademark of Xerox Corporation.
**Virus Disclaimer**

Cabletron Systems makes no representations or warranties to the effect that the Licensed Software is virus-free.

Cabletron has tested its software with current virus checking technologies. However, because no anti-virus system is 100% reliable, we strongly caution you to write protect and then verify that the Licensed Software, prior to installing it, is virus-free with an anti-virus system in which you have confidence.

**Restricted Rights Notice**

(Applicable to licenses to the United States Government only.)

1. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.


2. (a) This computer software is submitted with restricted rights. It may not be used, reproduced, or disclosed by the Government except as provided in paragraph (b) of this Notice or as otherwise expressly stated in the contract.

   (b) This computer software may be:

      (1) Used or copied for use in or with the computer or computers for which it was acquired, including use at any Government installation to which such computer or computers may be transferred;

      (2) Used or copied for use in a backup computer if any computer for which it was acquired is inoperative;

      (3) Reproduced for safekeeping (archives) or backup purposes;

      (4) Modified, adapted, or combined with other computer software, provided that the modified, combined, or adapted portions of the derivative software incorporating restricted computer software are made subject to the same restricted rights;

      (5) Disclosed to and reproduced for use by support service contractors in accordance with subparagraphs (b) (1) through (4) of this clause, provided the Government makes such disclosure or reproduction subject to these restricted rights; and

      (6) Used or copied for use in or transferred to a replacement computer.

   (c) Notwithstanding the foregoing, if this computer software is published copyrighted computer software, it is licensed to the Government, without disclosure prohibitions, with the minimum rights set forth in paragraph (b) of this clause.

   (d) Any other rights or limitations regarding the use, duplication, or disclosure of this computer software are to be expressly stated in, or incorporated in, the contract.

   (e) This Notice shall be marked on any reproduction of this computer software, in whole or in part.
Preface

Who Should Read This Guide ................................................................. v
What Is in This Guide ........................................................................ v
Questions about SPECTRUM Documentation ................................ vi

Chapter 1  SPECTRUM Overview

What Is in This Chapter ...................................................................... 1-1
SPECTRUM Client/Server Model .................................................... 1-1
SPECTRUM Products ...................................................................... 1-2

Chapter 2  SPECTRUM Tools

What Is in This Chapter ...................................................................... 2-1
SPECTRUM Toolkit Products .......................................................... 2-1
  SPECTRUM Level II Developer’s Tools ........................................... 2-2
  Development Environment .......................................................... 2-5
SPECTRUM Tools Documentation .................................................. 2-5
  SPECTRUM External Protocol Interface (EPI) .............................. 2-5
  SPECTRUM SpectroSERVER API & View API ............................. 2-5
  SPECTRUM Inference Handler API ............................................. 2-6
Level II Tools Documentation Reading Path .................................. 2-7

Chapter 3  SPECTRUM Toolkit Development Process

What Is in This Chapter ...................................................................... 3-1
Introduction ..................................................................................... 3-1
  Preparing the Development Environment .................................. 3-2
  Development Process Summary .................................................. 3-2
Management Module Certification .................................................. 3-3
Chapter 4 Using Developer’s Tools to Extend SPECTRUM

What Is in This Chapter ..................................................................................................... 4-1
What Can Be Accomplished ............................................................................................... 4-1
Basic Extensions ................................................................................................................. 4-2
Advanced Extensions ........................................................................................................ 4-3
  New Device Views ........................................................................................................... 4-3
  New Protocols ................................................................................................................... 4-3
  Application Integration ..................................................................................................... 4-4
  Gateways ........................................................................................................................... 4-4
  Intelligence Circuits ............................................................................................................ 4-4
Who Should Read This Guide

This guide provides an overview of SPECTRUM, the SPECTRUM Level II Developer's Toolkit, and the SPECTRUM Developer's Toolkit documentation. It also provides a brief discussion on how the various elements of the SPECTRUM Extensions Integration (SEI) Toolkit are used to extend the base functionality of SPECTRUM to suit the specific needs of our customers.

This guide is intended for developers who want to use the Level II Developer's Toolkit to create integrated and installable SPECTRUM extensions for customer use or resale.

What Is in This Guide

This guide is organized as follows:

• **Chapter 1** briefly describes SPECTRUM, its two major components (SpectroGRAPH and SpectroSERVER), and the categories of SPECTRUM products that are available.

• **Chapter 2** briefly describes each of the various Level II developer's tools that are currently available from Cabletron Systems.

• **Chapter 3** provides an overview of the SPECTRUM Software Developer's Toolkit development process. It also describes the process of certifying new management modules developed through the Toolkit.

• **Chapter 4** describes the various uses of the Software Developer’s Toolkits and the types of basic and advanced extensions that can be developed using the Toolkits.
Questions about SPECTRUM Documentation

Send your questions, comments or suggestions regarding SPECTRUM documentation to the Technical Communications Department directly via the following internet address:

spectrum-techdocs@ctron.com
Chapter 1

SPECTRUM Overview

What Is in This Chapter

SPECTRUM is an integrated management system that runs on UNIX and NT platforms and uses an X Window System/Motif user interface. The SPECTRUM design incorporates two key elements that support the integration of new functionality:

- The object-oriented paradigm of the C++ language provides the ability to model real world objects and develop extensions to these models.

- The Inductive Modeling Technology™ (IMT) is event-driven code that provides the artificial intelligence for the models to infer knowledge about events in other parts of SPECTRUM and respond to them.

Although it was developed primarily as a management station for different kinds of computer networks, SPECTRUM can be extended to manage other domains (such as facilities maintenance) where management of systems and resources are important. SPECTRUM extensions can take the form of device management modules, gateways to other network management systems, or integration with other applications related to network management, such as facilities management.

SPECTRUM Client/Server Model

SPECTRUM is based on a client/server model. The client is the SPECTRUM user interface, which provides a graphical representation of the network environment. The server includes the SPECTRUM knowledge base, which obtains and stores all network information. In SPECTRUM terminology, the client user interface is called SpectroGRAPH™ and the knowledge-base server is called SpectroSERVER™.
SpectroSERVER, which provides the intelligence of SPECTRUM, contains models of the actual network devices and their interactions. These elements combine to provide a comprehensive management perspective of a network. This network conceptualization is referred to as the Virtual Network Machine or VNM. These models are continuously collecting data about the objects they represent. As a result of the data collection process, the SPECTRUM database gains extensive knowledge about any network that it is managing. By analyzing this information when maintaining a network, an administrator can maximize system performance while minimizing cost.

SpectroGRAPH provides users with a multi-dimensional picture of their network. Users can look at the network via location views (e.g., region, city, building, or room), or they can look at the network via topological views (e.g., WAN, LAN, Device, or Board). With SpectroGRAPH, users can retrieve and view the information maintained in SpectroSERVER, or invoke SpectroSERVER to control objects on the network.

**SPECTRUM Products**

Three categories of SPECTRUM products are available:

- The SPECTRUM Network Management core package which consists of the SpectroSERVER management server and SpectroGRAPH graphical user interface. Also available as end products are the Cabletron-developed management modules that model vendor-specific devices, and application programs to enhance your network management system.

- Level I Developer’s Tools (See *Getting Started with Level 1 Tools.*)

- Level II Developer’s Tools
SPECTRUM Tools

What Is in This Chapter

This chapter describes the various SPECTRUM Toolkit products that are available and the documentation that is distributed with each toolkit.

SPECTRUM Toolkit Products

There are two categories of SPECTRUM Toolkit products:

• Level I Developer's Tools -- (See Getting Started with Level I Tools.)

• Level II Developer's Tools -- These tools offer programming interfaces for the purpose of developing C++ advanced extensions to SPECTRUM. These tools include header files, object libraries, example source code, complete documentation, training, and direct developer support from Cabletron.

Figure 2-1 shows how the Level II developer's tools fit into SPECTRUM, providing the capability to tailor SPECTRUM to a site's specific management needs.

Figure 2-1. Extending SPECTRUM using the Developer's Tools

Each of the developer's tools products (shown with thick borders in Figure 2-1) gives customers and VARs the ability to enhance SPECTRUM. The following sections outline the various developer's tools that make up the Level II Toolkit.
SPECTRUM Level II Developer’s Tools

The Level II Developer’s Tools offer advanced interfaces for extending SPECTRUM. These tools allow customers and developers to integrate new C++ objects and programs with SPECTRUM. The available Level II Developer’s Tools are:

- **SPECTRUM External Protocol Interface (EPI) Application Program Interface (API)** -- Provides an interface for attaching new or customized protocols to SPECTRUM. Protocols can be written in C and
Assembler, and integrated through a straight-forward interface layer, with no need to access or modify SPECTRUM internals.

- **SPECTRUM SpectroSERVER API** -- The SpectroSERVER API is the primary communications interface between the SpectroSERVER and client processes (such as SpectroGRAPH). This API provides both an asynchronous interface (for interactive and windowed applications) and a synchronous interface (for query and report requests to SPECTRUM). This toolkit also includes the Global and VnmParmBlock APIs and Class Libraries. These libraries are essential to the development of C++ objects and programs that integrate with the SpectroSERVER. The Global classes provide methods to create and manipulate buffers, Object IDs, messages, date and time settings, and other general data representations used throughout SPECTRUM. The VnmParmBlock classes are used to define objects that allow access to specific SPECTRUM model types, attributes, aspects, and processes.

- **View API** -- The View Toolkit helps developers integrate new or modified views into the SPECTRUM environment. Views from non-Cabletron products can be accessed, or external processes can be initiated, from SPECTRUM. This toolkit also includes the Global and VnmParmBlock APIs and Class Libraries. These libraries are essential to the development of C++ objects and programs that integrate with the SpectroSERVER. The Global classes provide methods to create and manipulate buffers, Object IDs, messages, date and time settings, and other general data representations used throughout SPECTRUM. The VnmParmBlock classes are used to define objects that allow access to specific SPECTRUM model types, attributes, aspects, and processes.

- **SPECTRUM Inference Handler API** -- Provides source code and documentation for the development of inference handlers, which are the basis of SPECTRUM's management intelligence. With this API, developers can create Inference Handlers that allow SPECTRUM to manage entirely new objects and devices through defining the possible interactions between SPECTRUM and these objects or devices. The IHAPI Toolkit includes C++ object libraries, header files, and tutorials based on sample source code to guide developers through the process of creating new Inference Handlers. Also included are built-in tracing and reporting, as well as a discussion of debugging techniques applicable to the SPECTRUM multi-threaded environment. This toolkit also includes the Global and VnmParmBlock APIs and Class Libraries. These libraries are essential to the development of C++ objects and programs that integrate with the SpectroSERVER. The Global classes provide methods to create and manipulate buffers, Object IDs, messages, date and time settings, and other general data representations used throughout SPECTRUM. The VnmParmBlock classes are used to define objects that allow access to specific SPECTRUM model types, attributes, aspects, and processes.

- **SPECTRUM Extensions Integration Toolkit** -- Intended for customers or VAR developers who wish to package and sell a SPECTRUM extension they have developed. This toolkit enables a developer to produce
a complete, ready-to-install SPECTRUM extension. The toolkit includes tools and information for integrating a customer-developed extension with the SPECTRUM tape formats and the SPECTRUM Installation Script. Tools are also provided to export models from the SPECTRUM database, icons from SpectroGRAPH, and protocols from SPECTRUM. Purchasers of the toolkit become Cabletron SPECTRUM Partners and receive a unique Developer Identification Code to ensure that their SPECTRUM extensions can be integrated into any SPECTRUM site in the world.

With the SPECTRUM Level II Developer’s Tools, customers and VAR developers can create advanced extensions, by integrating new objects and programs into the SPECTRUM knowledge base. These additions range from customer-written customization of the SPECTRUM user interface at a specific site to a VAR-developed management module that can be sold to multiple customer sites. Several of the most common types of Level II extensions are:

- Development of a new physical or logical device view, using the Asynchronous SpectroSERVER API and the View API. Although a device management module can be developed with just the Level I Developer’s Tools, it cannot contain either a physical or logical device view. The Asynchronous SpectroSERVER API allows the new view complete read and write access to the model of the device contained within the SpectroSERVER process, including all that is known about the current state of the model. The View API allows the new view to appear integrated with the SpectroGRAPH process. Full copy, cut, and paste functionality is supported, as well as the starting of other views within SPECTRUM. Of particular note, the View API provides a way for a new device view to bring up Generic Views detailing particular aspects of the model’s state.

- The addition of different protocols to SPECTRUM to manage devices that do not support SNMP using the MSAP-EPI API. Each new protocol leads to an additional EPI process, which translates SpectroSERVER EPI requests into the appropriate proprietary requests.

- Integration of new or existing applications onto the SPECTRUM Platform. Depending on the type of application being integrated and the level of integration needed, this may require the use of one or more of the Level II Developer’s Tools. For example, to allow the application to access SPECTRUM data, use either the Synchronous or Asynchronous SpectroSERVER APIs. To allow an application to appear integrated at the user interface, use the View API. If the application requires that intelligence be added to the SpectroSERVER, then the Inference Handler API is needed.

- Implementation of a Gateway between SPECTRUM and another Network Management System using the Asynchronous SpectroSERVER API. A Gateway allows the sharing of Alarm information among network management systems, and allows centralized management of a network whose subcomponents are managed by different network management systems.
Development Environment

The SPECTRUM Level II Developer's Tools are designed for use with a platform-specific version of C++ (or C, in the case of the EPI). Refer to the Software Release Notice (SRN) accompanying the SPECTRUM developer's tools for more information on release-specific requirements.

SPECTRUM Tools Documentation

This section lists the documentation that is included with each of the Level II Developer's Tools.

SPECTRUM External Protocol Interface (EPI)

• SPECTRUM MSAP/EPI Developer's Guide -- Provides the information needed to develop Management Station Access Provider (MSAP) software that can communicate with SPECTRUM using the External Protocol Interface (EPI). Also defines the requirements for building management modules for device and MSAP model types.

SPECTRUM SpectroSERVER API & View API

SPECTRUM SpectroSERVER API Developer's Guide -- Provides information about the design and structure of SSAPI applications. Also includes a discussion of the demo programs provided with the SSAPI Toolkit.

• SPECTRUM SpectroSERVER API Reference -- Describes the classes and methods of the Asynchronous and Synchronous interfaces provided by the SSAPI for client application communication with SpectroSERVER.

• View Application Program Interface Developer's Guide -- Explains how to develop or modify SPECTRUM Views.
• **Global Classes Reference** -- Describes the Global C++ classes and methods used with the Inference Handler API, the View API, and the asynchronous and synchronous SpectroSERVER APIs.

• **VnmParmBlock Reference** -- Describes the VnmParmBlock C++ classes and methods that are used with the Inference Handler API, the View API, and the asynchronous and synchronous SpectroSERVER APIs.

**SPECTRUM Inference Handler API**

• **Inference Handler API Developer's Guide** -- Describes the process of developing inference handlers to add specialized intelligence to user/VAR-developed model types within SPECTRUM.

• **Global Classes Reference** -- Describes the Global C++ classes and methods that are used with the Inference Handler API, the View API, and the Asynchronous and Synchronous SpectroSERVER API.

• **VnmParmBlock Reference** -- Describes the VnmParmBlock C++ classes and methods that are used with the Inference Handler API, the View API, and the Asynchronous and Synchronous SpectroSERVER API.
Level II Tools Documentation Reading Path

Figure 2-2 shows the recommended reading path for the Level II Developer's Tools Documentation.

Figure 2-2. Level II Developer's Tools Documentation Reading Path
What Is in This Chapter

This chapter provides an overview of the SPECTRUM Software Developer’s Toolkit development process. It also describes the process of certifying new management modules developed through the Toolkit.

Introduction

Throughout the SPECTRUM API documents, several examples are provided to highlight the use of the individual APIs. In addition, this chapter provides an overview of a sample development process to show how the toolkits interact during the development of a complete management module.

NOTE

It is not the intent of Cabletron to impose a particular development methodology or software development environment structure on customers. Rather, this information is presented as an example of a development process that works efficiently for SPECTRUM Extensions.
Preparing the Development Environment

Three important factors must be emphasized in outlining this development process:

1. Before attempting to develop extensions to SPECTRUM via the Toolkits, the customer/developer should be thoroughly familiar with the basic SPECTRUM product offered by Cabletron Systems.

2. The SPECTRUM core (SPECTRUM without device-dependent management modules installed) must be purchased and installed as part of the development environment.

3. To ensure software integrity, Cabletron strongly recommends that separate SPECTRUM core areas be maintained for each of the following uses:
   - Actual network management operation of SPECTRUM.
   - A clean copy of the SPECTRUM core in an area to be used for extension development and manufacturing (media release preparation), isolated from the online network management operation.
   - A clean copy of the SPECTRUM core for integration testing of customer-produced management modules, isolated from the online network management operation and development area.

Development Process Summary

This section presents an overview of how a customer can produce SPECTRUM-compatible management modules:

1. Purchase and install the SPECTRUM end-user products (for example, the core system). If SPECTRUM is to be used for live Network Management, make an additional copy of the area for development use, and an additional copy for management module integration testing.

2. Purchase the SPECTRUM Software Developer’s Toolkit package and install it into the development area.

3. Obtain a registered developer ID from Cabletron.

4. Set up a management module development on a local system. This should be an area separate from the installed SPECTRUM core.

5. Design management module models.

6. Develop the management module according to the guidelines in the SPECTRUM Software Developer’s Toolkit documents.
7. Perform integration testing of the management module. The **Install.quick** tool, found in the *SPECTRUM Extensions Integration (SEI) Toolkit Developer’s Guide*, can be used to perform integration testing of single or multiple management modules with the SPECTRUM core.

8. Produce a version of the management module for integration into the manufacturing area (refer to the *SPECTRUM Extension Integration (SEI) Toolkit Developer’s Guide* for more information).

9. Produce the management module media from the manufacturing tools.

10. Test the distribution media with the released SPECTRUM core.

11. Produce the final version of the management module with checking enabled.

12. Integrate the final version of the management module into the manufacturing area.

13. Produce management module distributions for customers.

The steps outlined above show the basic procedures to follow in the design, development, testing, and packaging of SPECTRUM-compatible management modules. Once a developer has become familiar with this basic process, the developer can choose to accelerate the process by collapsing steps or developing automated tools.

## Management Module Certification

The SPECTRUM Software Developer’s Toolkit is sold for two distinct reasons. The first reason is for end users to extend SPECTRUM strictly for their own use. The second reason is to develop SPECTRUM extensions intended for resale, i.e., new management modules.

A SPECTRUM management module consists of any combination of the following:

- New (derived) model types, relations, and rules - simple and/or complex
- New management protocols
- New presentation formats (simple and/or complex icons, views, information blocks, etc.)

To maintain usable standards for SPECTRUM extensions, Cabletron distinguishes between extensions developed by customers and extensions developed by commercial Original Equipment Manufacturers (OEM). Customers who would like to add management modules for internal use only
are considered common vendors in this context. Those who would like to add management modules for resale to third parties do so as registered vendors.

Both common and registered vendors can add management modules that involve basic and/or advanced extensions. The process of building extensions to SPECTRUM (basic or advanced) is the same for common and registered vendors. However, the method for assigning a developer (or Vendor) ID is significantly different for common vendors than it is for registered vendors.

A model type developed as part of a management module by a common vendor is stamped with a common developer’s ID. This is not a unique ID, and if used outside the vendor’s environment, may clash with other model types. If model types developed by common vendors are exported or sold to third parties, they are not guaranteed to work in combination with other extensions.

To develop model types for resale to third parties, a customer has to be accepted by Cabletron into the Developer’s Program. The customer is then issued a registered developer’s ID, and all model types developed as part of a management module by such a customer are stamped with this ID. It uniquely identifies the model types developed by each registered vendor. Thus, model types developed by registered vendors can be resold or exported to other environments. The registered vendor developer ID ensures that changes imported for one management module do not adversely interact with changes imported for a different management module. As part of the process of applying for and receiving a registered vendor developer ID, the vendor agrees to Cabletron certification of all management modules that are developed for resale. The certification process ensures that defects in the management module do not corrupt the integrity of the SPECTRUM core. The SPECTRUM Extensions Integration (SEI) Toolkit Developer’s Guide provides information about the tools required to package an extension (a) for installation at a SPECTRUM customer site and (b) for resale.
Chapter 4

Using Developer’s Tools to Extend SPECTRUM

What Is in This Chapter

This chapter describes the various uses of the Software Developer's Toolkits and the types of basic and advanced extensions that can be developed using the Toolkits.

What Can Be Accomplished

SPECTRUM was designed to be a highly extendable network management platform. Customers can add new model types and relations to the SPECTRUM knowledge base. They can also add new management protocols, icons, views, etc. (referred to as SPECTRUM extensions). Depending on the complexity of the additions, they are categorized as basic or advanced extensions. Simply stated, if the additions involve writing C++ code, they are considered advanced extensions. The SPECTRUM Software Developer's Toolkit is required in order to build advanced extensions.

If the additions do not involve writing C++ code, they are considered basic extensions. The advanced Software Developer's Toolkit is not required for a customer to build basic extensions for use at a SPECTRUM site. Many of the SPECTRUM extensions needed by customers and developers can be accomplished using only Basic Extensions. See “Basic Extensions” in this chapter for more information.
However, if an Original Equipment Manufacturer (OEM) developer wishes to develop a basic extension to SPECTRUM and sell or distribute it, the advanced Software Developer’s Toolkit is needed in order to develop software to install and integrate the OEM SPECTRUM extension with an installed SPECTRUM platform at the customer site. Refer to Chapter 3 for more details.

The SPECTRUM Command Line Interface also provides a mechanism to extend SPECTRUM through the development of shell scripts that read and write to the models in the SpectroSERVER. Refer to the Command Line Interface User’s Guide for more information on this aspect of basic extensions.

The SPECTRUM Software Developer’s Toolkit was developed to handle the small percentage of extensions that require new C++ code to be generated. The extensions fall into one of several categories according to the Software Developer’s tools used. These categories are discussed in “Advanced Extensions” in this chapter.

Basic Extensions

There are several categories of extensions that can be developed without using the advanced Software Developer’s Toolkit. Simple model types and relations (those that involve only the addition of attributes) can be derived from existing ones by using tools such as the Model Type Editor (MTE). This does not involve writing C++ code and is considered a basic extension.

The process of adding a new alert, mapping an alert to an event, or mapping an event to an alarm does not involve writing C++ code, and is considered a basic extension, whether it is for a simple or complex model type.

New base icons can be created by simply creating a new information block. Views can be made to display new icons by modifying the information blocks associated with them. Also, simple views can be created by modifying an information block.

Information about developing basic extensions to SPECTRUM is contained in the SPECTRUM Knowledge Base Guide. This guide provides information on utilizing the concepts described in the SPECTRUM Concepts Guide.
Advanced Extensions

Advanced extensions go beyond the basic extensions to SPECTRUM by integrating new objects and programs into the SPECTRUM knowledge base. These additions range from customer-written customization of the SPECTRUM user interface at a specific site to an OEM-developed management module that can be sold to multiple customer sites. The most common types of advanced extensions are described in the following sections.

New Device Views

New device views are probably the most frequent reason for advanced Toolkit use. In order to develop a device view, both the Asynchronous SpectroSERVER API and the View API (the physical and logical device views) are required.

The Asynchronous SpectroSERVER API allows the new view complete access to the model of the device contained within the SpectroSERVER process. This interface allows full access, both read and write, to all that is known about the current state of the model. The new device view may need storage for local information required only for that view. This is accomplished by adding new attributes to the model type with the MTE.

The View API allows the new view to appear integrated with the SpectroGRAPH process. Full cut and paste functionality is supported, as well as the starting of other views within SPECTRUM. Of particular note, the View API is the way for a new device view to bring up Generic Views detailing particular aspects of the model's state.

New Protocols

Another frequent use of the Toolkit is the addition of different proprietary protocols to SPECTRUM to manage devices that do not support SNMP. The protocol is added using the EPI SDK, and is implemented in a separate process from the SpectroSERVER process. Each new protocol leads to an additional satellite EPI process that translates SpectroSERVER EPI requests into the appropriate proprietary requests.

No other Toolkit is required to add a new protocol into SPECTRUM. Of course, a protocol is of no use until a device management module is created that uses the new protocol.
Application Integration

A third use of the Toolkit is integration of new or existing applications onto the SPECTRUM platform. Depending on the type of application being integrated and the level of integration required, this may use all aspects of the Toolkit or only a small subset.

Use either the Synchronous or Asynchronous SpectroSERVER API to allow the application to access SPECTRUM data.

Use the View Toolkit to allow the application to appear integrated at the User Interface.

Use the Inference Handler Toolkit if the application requires the addition of intelligence to the SpectroSERVER.

Gateways

A fourth use of the Toolkit is the implementation of a gateway between SPECTRUM and some other Network Management System. The Asynchronous SpectroSERVER API is most commonly used for this type of solution. A gateway allows the sharing of Alarm information between Network Management Systems and allows for centralized management of a network whose subcomponents are managed by different Network Management Systems.

Intelligence Circuits

Perhaps the least common use of the Toolkit is for the development of new intelligence circuits. Intelligence circuits are implemented using the Inference Handler API. For a complete description of what an intelligence circuit is and how they may be developed, refer to the SPECTRUM Inference Handler API Developer’s Guide.
Index

A
advanced extensions 4-3
API
  Asynchronous SpectroSERVER 2-4
  External Protocol Interface 2-2
  Global 2-3
  Inference Handler 2-3
  SpectroGRAPH 2-3
  View 2-3
API, VnmParmBlock 2-3
application integration 4-4
Application Program Interface (See API)
Asynchronous SpectroSERVER API 2-4

B
basic extensions 4-1, 4-2

C
Cabletron Developer’s Program 3-4
Cabletron SPECTRUM Partners 2-4
certifying management modules 3-3
class libraries 2-3
classes
  Global 2-3
  VnmParmBlock 2-3
Command Line Interface 4-2

device views
  development of 4-3

E
environment
  for SPECTRUM development 2-5
EPI, Application Program Interface 2-2
Extensions
  forms of 1-1
Extensions Integration Toolkit 2-3
External Protocol Interface (See EPI)

G
GateWay implementation 4-4
Global API 2-3
Global classes 2-3

I
implementing GateWays 4-4
Inductive Modeling Technology 1-1
Inference Handler API 2-3
Install.quick 3-3
integrating applications 4-4
intelligence circuits
  development of 4-4

L
Level II Developer’s Tools 2-2 to 2-5
Level II extensions, common types 2-4

M
management module certification 3-3
management modules 3-3
models
  in SpectroSERVER 1-2
N
Notice i

P
product categories 1-2
protocols
  adding 4-3

R
Restricted Rights Notice ii

S
SDK 4-2
SpectroGRAPH 1-1
SpectroSERVER
  overview 1-1
SpectroSERVER API 2-3
SPECTRUM
  Cabletron Partners 2-4
  client/server model 1-1
  development environment 2-5
  end user products 1-2
  integration support 1-1
  product categories 1-2
SPECTRUM Command Line Interface 4-2
SPECTRUM extensions
  forms of 1-1
SPECTRUM overview 1-1
SPECTRUM SDK 4-2

T
Trademarks i

V
vendor ID 3-4
View API 2-3
Virtual Network Machine
  definition 1-2
Virus Disclaimer ii
VnmParmBlock API 2-3
VnmParmBlock classes 2-3