

# CA Spectrum®

## 概念指南

版本 9.3



本文档包括内嵌帮助系统和以电子形式分发的材料（以下简称“文档”），其仅供参考，CA 随时可对其进行更改或撤销。

未经 CA 事先书面同意，不得擅自复制、转让、翻印、透露、修改或转录本文档的全部或部分内容。本文档属于 CA 的机密和专有信息，不得擅自透露，或除以下协议中所允许的用途，不得用于其他任何用途：(i) 您与 CA 之间关于使用与本文档相关的 CA 软件的单独协议；或者 (ii) 您与 CA 之间单独的保密协议。

尽管有上述规定，但如果您为本文档中所指的软件产品的授权用户，则您可打印或提供合理数量的本文档副本，供您及您的雇员内部用于与该软件相关的用途，前提是所有 CA 版权声明和标识必须附在每一份副本上。

打印或提供本文档副本的权利仅限于此类软件所适用的许可协议的有效期内。如果该许可因任何原因而终止，您应负责向 CA 书面证明已将本文档的所有副本和部分副本已退还给 CA 或被销毁。

在所适用的法律允许的范围内，CA 按照“现状”提供本文档，不附带任何保证，包括但不限于商品适销性、适用于特定目的或不侵权的默示保证。CA 在任何情况下对您或其他第三方由于使用本文档所造成的直接或间接的损失或损害都不负任何责任，包括但不限于利润损失、投资受损、业务中断、信誉损失或数据丢失，即使 CA 已经被提前明确告知这种损失或损害的可能性。

本文档中涉及的任何软件产品的使用均应遵照有关许可协议的规定且根据本声明中的条款不得以任何方式修改此许可协议。

本文档由 CA 制作。

仅提供“有限权利”。美国政府使用、复制或透露本系统受 FAR Sections 12.212、52.227-14 和 52.227-19(c)(1) - (2) 以及 DFARS Section 252.227-7014(b)(3) 的相关条款或其后续条款的限制。

版权所有 © 2013 CA。保留所有权利。此处涉及的所有商标、商品名称、服务标识和徽标均归其各自公司所有。

## CA Technologies 产品引用

本文档引用以下 CA Technologies 产品：

- CA Spectrum®
- CA Spectrum® Report Manager (Report Manager)
- CA Spectrum® Service Manager (Service Manager)
- CA Spectrum® Southbound Gateway Toolkit (Southbound Gateway)
- CA Spectrum® 建模网关工具包（建模网关）
- CA Spectrum® Alarm Notification Manager (SANM)
- CA eHealth® (eHealth)
- CA Business Intelligence (CABI)

## 联系技术支持

要获取在线技术帮助以及办公地址、主要服务时间和电话号码的完整列表，请联系技术支持：<http://www.ca.com/worldwide>。



# 目录

---

<b>第 1 章： CA Spectrum 概述</b>	<b>7</b>
关于 CA Spectrum .....	7
<b>第 2 章： SpectroSERVER 和 CA Spectrum 数据库概述</b>	<b>9</b>
关于 SpectroSERVER .....	9
CA Spectrum 数据库 .....	11
知识库 .....	11
Archive Manager .....	12
建模目录 .....	13
模型 .....	17
推理处理程序 .....	18
报告数据库 .....	20
SpectroSERVER 和线程 .....	20
托管元素 .....	21
设备发现 .....	21
设备通信管理器 .....	22
报警、事件和警报 .....	23
格局和分布式 SpectroSERVER .....	24
<b>第 3 章： 客户端应用程序</b>	<b>27</b>
客户端应用程序概述 .....	27
OneClick 控制台 .....	28
OneClick 控制台图标 .....	28
分层视图 .....	28
关于 AlarmNotifier .....	29
警报监控过程 .....	30
关于 SANM .....	31
CA Spectrum 如何监控警报 .....	31
使用 CA Business Intelligence (CABI) 报告 .....	32
InfoView 报告管理 .....	32
<b>附录 A： 属性和关系定义</b>	<b>35</b>
属性 .....	35
应用程序模型发现 .....	35
设备模型发现 .....	35
常规模型类型信息 .....	36

---

网络信息.....	36
轮询信息.....	37
端口标识.....	37
SNMP 信息.....	37
属性说明.....	38
关系说明.....	46

## 词汇表

49

# 第 1 章： CA Spectrum 概述

---

此部分包含以下主题：

[关于 CA Spectrum \(p. 7\)](#)

## 关于 CA Spectrum

CA Spectrum 是一个服务和基础架构管理系统，可监控托管元素（包括设备、应用程序、主机系统和连接）的状态。来自这些元素的故障和性能数据等状态信息会被收集和存储。CA Spectrum 会不断分析此信息，以跟踪计算基础架构中的状况。如果检测到异常状况，将进行隔离，并向您发出报警。CA Spectrum 还会为您提供该问题的可能原因和解决方案。

CA Spectrum 设计基于客户端/服务器模型。其主服务器 SpectroSERVER 负责收集、存储和处理数据。SpectroSERVER 使用感应建模技术 (IMT) 来执行这些功能。IMT 结合了面向对象的数据库和推理处理程序的智能。面向对象的数据库包含定义每个托管元素表示形式的模型类型以及表示特定托管元素的模型。面向对象的数据库还包含定义模型类型之间可能关联的关系。推理处理程序通过对 CA Spectrum 或托管元素生成的事件作出反应，向此系统提供更多的功能。

SpectroSERVER 在定义了模型类型、模型和关系的知识库中存储数据。SpectroSERVER 还轮询托管元素，并接收来自计算基础架构的报警信息。SpectroSERVER 分析此信息，将其存储在知识库中，并允许客户端应用程序访问此信息。

CA Spectrum 包括很多客户端应用程序。其主要的客户端应用程序 OneClick 提供的图形用户界面用于监控网络并启动其他客户端应用程序。OneClick 控制台中提供的视图包含表示网络不同元素的图标、表和图表。这些图形组件提供状态信息并提供特定于它们所表示托管元素的管理工具的访问权限。所有客户端应用程序数据都检索自 SpectroSERVER。



## 第 2 章： SpectroSERVER 和 CA Spectrum 数据库概述

---

此部分包含以下主题：

[关于 SpectroSERVER](#) (p. 9)

[CA Spectrum 数据库](#) (p. 11)

[知识库](#) (p. 11)

[报告数据库](#) (p. 20)

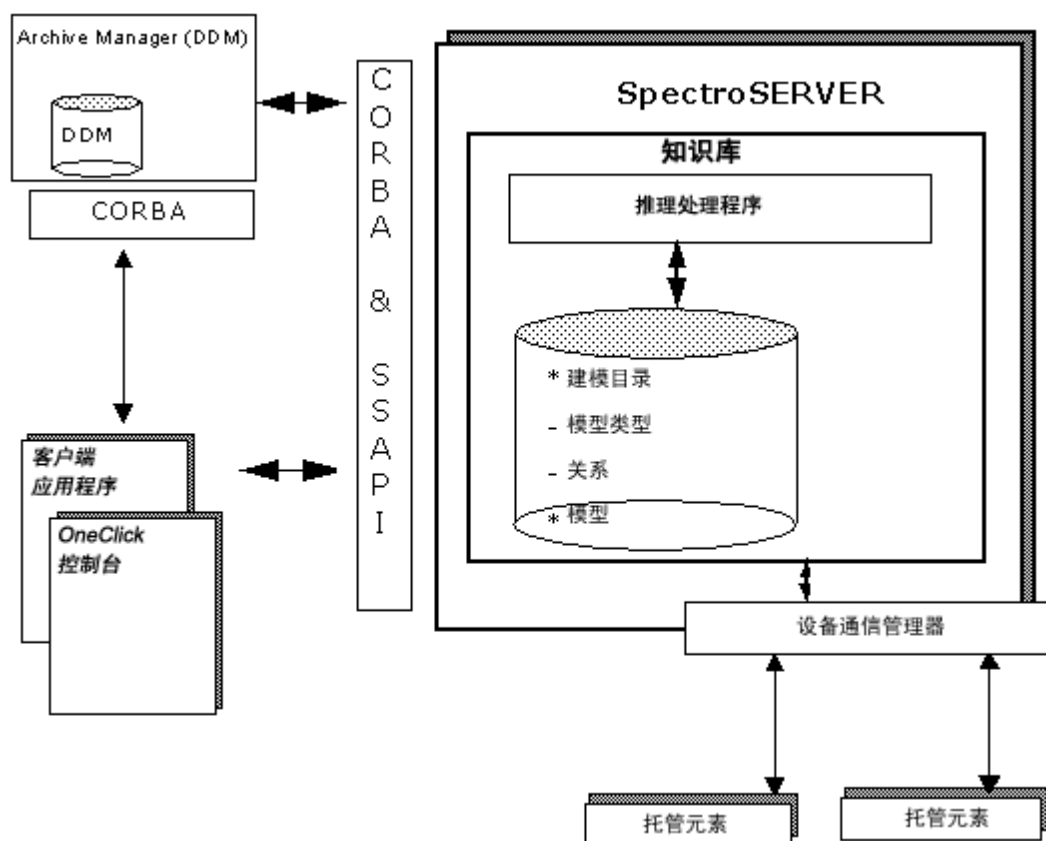
[SpectroSERVER 和线程](#) (p. 20)

[托管元素](#) (p. 21)

### 关于 SpectroSERVER

SpectroSERVER 是 CA Spectrum 的主服务器，可充当数据库服务器、建模引擎和设备管理器。SpectroSERVER 处理事件，生成警报，并跟踪托管元素的统计信息。此信息可供客户端应用程序使用，并可通过 SpectroSERVER 应用程序编程接口 (SSAPI) 和 CA Spectrum CORBA 接口请求。

下图在简化视图中说明了 SpectroSERVER 组件:



**注意:** SpectroSERVER 也称为 VNM (虚拟网络计算机)。具体而言, VNM 是指负责对托管元素进行建模的 SpectroSERVER 部分。

## CA Spectrum 数据库

CA Spectrum 包括以下数据库：

- SpectroSERVER 数据库。  
有关详细信息，请参阅《数据库管理指南》。
- 分布式数据管理器 (DDM) 数据库，可以存储 CA Spectrum 事件和统计信息以供多个格局使用。  
有关详细信息，请参阅《数据库管理指南》。
- MIB 工具数据库，支持 MIB 工具实用工具。  
有关详细信息，请参阅《认证用户指南》。
- [报告数据库](#) (p. 20)，支持 Report Manager 和 Service Manager。  
有关详细信息，请参阅《Report Manager 安装和管理指南》。
- eHealth 集成数据库，它是 CA Spectrum 与 eHealth 集成所必需的。  
有关详细信息，请参阅《CA eHealth 与 CA Spectrum 集成和用户指南》。

## 知识库

知识库是 SpectroSERVER 的主要组件。知识库包括管理计算基础架构所需的数据和程序信息。

知识库包括存储模型类型、模型、关系以及事件和统计信息的组件。模型的复杂系统和模型之间的关联关系允许知识库表示和存储网络元素的相关信息。如果将此模型系统及其关联关系看作单个逻辑实体，可描述计算基础架构的物理和逻辑拓扑。CA Spectrum 在此基础上构建其 *根本原因分析* 功能。

知识库中的所有模型都基于称为 *模型类型* 的模板。模型类型用于定义构成实例化模型的属性。所有模型类型都存储在知识库建模目录中。

知识库还包含向模型类型提供部分智能的进程。这些进程包括推理处理程序和操作。当 SpectroSERVER 正在运行时，进程数据会存储在内存中，同时也是知识库的一部分。

知识库还使用 Archive Manager 和分布式数据管理器 (DDM) 来存储有关特定模型的历史事件和统计信息。此信息会随着时间的推移而积累，使得 CA Spectrum 可以获得有关正被托管的计算基础架构的大量知识。

## Archive Manager

每个格局都有一个 Archive Manager 服务器，该服务器将从 SpectroSERVER 检索事件和统计信息，压缩这些信息，并将其存储在 DDM 数据库中。通过数据压缩可以存储更多的性能数据，并减少应用程序与 DDM 数据库之间的网络通信。

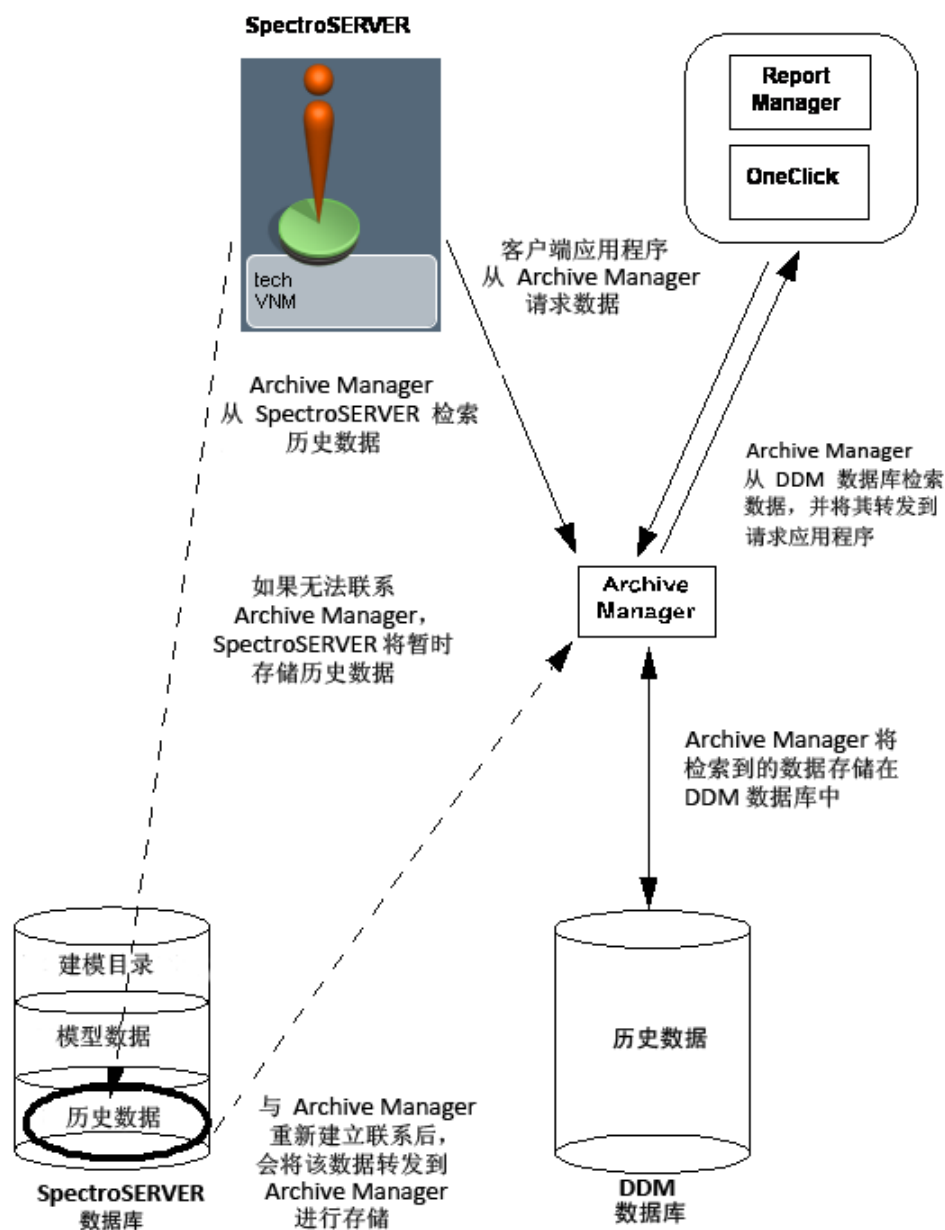
如图所示，如果 SpectroSERVER 无法联系 Archive Manager，SpectroSERVER 将一直存储事件和统计数据，直到重新建立联系。然后，SpectroSERVER 会将数据发送到 Archive Manager 以进行存储。.vnmrc 文件中的 Events 和 Statistics Archive 选项确定了 SpectroSERVER 存储的数据量。.configrc 文件中的选项确定了历史数据在 DDM 数据库中的存储时间长度。

**注意：**有关详细信息，请参阅《分布式 SpectroSERVER 管理员指南》。

Archive Manager 还可以提供以下信息以响应客户端应用程序的请求：

- 提供了其信息的格局的列表
- 针对每个格局，可用信息的时间范围以及提供了其信息的模型类型的列表
- 针对每个模型类型，提供了其信息的模型的列表
- 针对每个模型，提供了其信息的属性的列表
- 指定时间范围内的统计数据
- 指定时间范围内的事件数据

下图显示了 SpectroSERVER 与 DDM 数据库之间的交互。



## 建模目录

建模目录是知识库的元数据存储库。建模目录对象随 CA Spectrum 一起提供, 并且相对静态, 但是您可以出于调整目的处理某些目录方面。还可以自定义建模目录, 使 CA Spectrum 了解新网络技术或新类型的托管网络元素。以下部分介绍了包含在建模目录中的各种特定类型的对象。

## 模型类型

模型类型主要与托管元素系列相对应，是用于构建模型的模板。模型类型包含管理特定类型的托管元素所需的信息或属性。模型类型具有告诉 CA Spectrum 该模型类型表示的托管元素如何行为的情报。此情报还说明了托管元素对在网络中托管元素或其他位置上发生的事件如何作出反应。

例如，CA Spectrum 建模目录包含 NokiaFW 模型类型。此模型类型表示特定类型的诺基亚防火墙，如 IP330、IP440、IP650 和 IP740。CA Spectrum 使用它来创建表示网络中特定诺基亚防火墙的模型。

每个模型类型使用 *模型类型句柄* 编号（通常以十六进制格式表示）在建模目录中唯一标识。

## 模型类型属性

每种模型类型都具有定义模型类型所表示托管元素的特征和属性的属性。这些属性可以是内部的，也可以是外部的。内部属性反映特定于特定元素的 CA Spectrum 管理的消息。外部属性反映托管元素支持的 MIB 中的对象。所有属性都具有与模型类型关联的默认值。

在许多情况下，在特定模型类型的模型被实例化时，属性会具有新值。属性值特定于模型所表示的托管元素。但是，有些属性是共享属性。给定模型类型的所有模型都访问相同的共享属性及其值。这些属性和值在每个模型的内存或数据库中不重复。

每个属性使用称为 *属性 ID* 的编号（通常以十六进制格式表示）在知识库中唯一标识。许多属性跨多种模型类型使用。例如，建模目录中几乎每个模型类型都使用属性 `Modeltype_name` 或 `IPAddress`。这些属性的属性 ID 在所有模型类型中保持相同。通过使用模型类型继承，可实现这种属性规范化。

### 详细信息:

[模型类型层次结构](#) (p. 16)

## 关系

关系定义模型类型可相互关联的可能方法。关系在 CA Spectrum 知识库中定义。关系的示例有包含、管理和 `Connects_to`。每个关系都有唯一的编号（通常以十六进制格式表示）对其进行标识。此标识符称为 *关系句柄*。

详细信息:

[关系说明](#) (p. 46)

## 元规则

元规则通过定义使用关系的上下文为关系赋予含义。元规则标识可参与关系的模型类型。要理解元规则的概念，请将模型类型和关系分别看作名词和动词。名词和动词短语可以组合成句。要使句子有意义，必须满足三个条件：

- 句子必须采用以下格式：（主语）名词 + 动词 + （宾语）名词。
- 句子必须符合逻辑；不能使用任何动词来连接任意两个名词。
- 句子必须反映现实。

元规则的 **CA Spectrum** 概念强调的是第二个条件。元规则可以在动词上定义，以限制该动词可以连接的名词。仔细定义元规则，使得它们施加给动词的限制符合逻辑。通常，有多个元规则控制每个动词。

考虑一种将以下名词和动词定义为构成网络的对象的模型类型和关系的语言。此外，假定元规则定义为以可将名词和动词一起使用的方式施加逻辑。元规则针对某种关系而定义，它包括两种模型类型：左侧模型类型和右侧模型类型。元规则从左到右的顺序是构建逻辑句子的格式。左侧模型类型是主语，关系是动词，右侧模型类型是句子的宾语。

### 名词

建筑物

房间

### 网络

LAN

打印机

工作站

### 动词

包含

收集

### 元规则

包含 [建筑物, 房间]、[房间, 工作站]

收集 [LAN, 打印机]、[LAN, 工作站]、[网络, LAN]

要使用此语言创建逻辑语句，必须满足前两个条件。下列示例满足第一个和第二个要求，并且是计算基础架构的现实表示：

- 工程馆包含测试实验室
- 测试实验室包含工作站 ABC
- 工程 LAN 收集工作站 ABC
- 工程 LAN 收集激光打印机

下列句子无效，因为它们未使用“名词/动词/名词”格式，因此不满足第一个条件：

- 包含建筑物收集
- 房间 LAN 工作站

下列句子满足格式要求，但是不符合逻辑或者不遵循定义的元规则：

- 建筑物包含工作站
- LAN 收集房间
- 打印机收集 LAN

## 关系的基数

关系定义为具有一对多或多对多的基数。例如，包含关系具有一个一对多基数。元-规则已定义为允许在房间模型类型和工作站模型类型之间存在包含关系。因为包含是一个一对多关系，因此单个房间可以包含多个工作站，但是单个工作站只能位于一个房间内。

**Connects\_to** 关系是一个多对多关系的示例。元规则已定义为允许在交换机和路由器模型类型之间存在 **Connects\_to** 关系。单个交换机连接到多个其他设备，其中包括路由器。同样，路由器也连接到多个设备，其中包括交换机。

关系的基数允许 **CA Spectrum** 模型以可以真正表示实际计算基础架构的方式在逻辑上进行连接、关联或组合。

## 模型类型层次结构

模型类型采用层次结构方式构建。首先构建的常规模型类型越多，从常规类型中派生的特定模型类型也就越多。模型类型使用主要继承派生。多重继承用于从多个基础模型类型派生模型类型。

派生的模型类型会同时继承从其中进行派生的一个或多个模型类型的属性和智能。派生的模型类型还会参与与基础模型类型相同的元规则。此外，派生的模型类型还使用基础模型类型所使用的相同的推理处理程序。

派生自多个基础模型类型的模型类型会使用特定顺序从基础模型类型继承属性和推理处理程序。因此，派生的模型类型不能多次继承某个属性或推理处理程序。继承还会确定属性的初始值。可以向派生的模型类型添加新属性（包括内部和外部）和新推理处理程序。与基础类型相比较，其派生的模型类型更为具体。

## 模型

除存储模型类型之外，知识库还存储已实例化的所有模型，以表示计算基础架构的元素。模型通过实例化特定模型类型来创建。将生成模板（模型类型）的副本，且该副本随后可用于表示计算基础架构中的实际元素。

模型被实例化后，该模型类型的属性会具有值。知识库还存储每个模型属性的当前值。这些模型属性中，有些是“共享”属性，对相同类型的所有元素都通用，并描述模型类型的各个方面或行为。给定模型类型的每个模型对于这些共享属性都具有相同的值。根据基础架构中的当前工作状况，每个模型未共享的属性的值可能不同。属性的值可描述单个模型的独特方面、特征和行为。

## 关联

在 CA Spectrum 使用模型创建计算基础架构组件的表示形式时，这些模型不会作为隔离的元素存在。与计算基础架构中的元素之间相互关联一样，模型之间也相互关联。在 CA Spectrum 实例化某个模型时，该模型与其他模型之间的适用关系也会被实例化。实例化的关系称为关联。关联必须遵循用于定义关系的元规则。OneClick 和其他客户端应用程序实施元规则，而不是 SpectroSERVER。

例如，考虑房子的买方与卖方之间的关系。名词（模型类型）是“买方”、“卖方”和“房子”。动词“购买”和“销售”会捕获名词之间可存在的可能关联关系。这些动词就是关系。关系需要元规则来提供含义。两个元规则是“买方购买房子”和“卖方销售房子”。基于这些元规则，“购买”和“销售”关系开始具有一些对于建模系统来说有意义的值。

关联会将元规则应用到现有的模型。假定 BuyerSmith 是类型“买方”的一个模型，而 SellerJones 是类型“卖方”的一个模型。NiceNewHouse 是类型“房子”的一个模型。基于元规则，可以设置一个关联，指明 BuyerSmith 购买 NiceNewHouse。同样，可以设置一个关联，指明 SellerJones 销售 NiceNewHouse。但是，元规则不允许诸如 NiceNewHome 购买 BuyerSmith 这样的关联关系，因此此关联无效。

详细信息：

[托管元素 \(p. 21\)](#)

## 推理处理程序

推理处理程序可定义模型类型的行为和智能。每个推理处理程序都可以执行特定任务。任务可以像更改属性值一样简单。或者，任务也可以像发现网段上的所有托管元素一样复杂。推理处理程序还可以执行常规任务，例如计算平均值。或者，推理处理程序也可以执行特定于模型类型的详细任务，例如创建 LAN 交换机端口的模型。从根本上说，推理处理程序是属于 CA Spectrum 核心的许多智能片段。借助于推理处理程序，CA Spectrum 可以提供其很多基础架构管理功能。

推理处理程序是与模型类型关联的 C++ 代码段。推理处理程序通常是支持各种触发器的代码的休眠片段。推理处理程序一经触发，便会执行任务。任务的结果可以是新的数据片段、更改的建模方案，也可以是触发的其他 CA Spectrum 子系统（例如其他推理处理程序）。推理处理程序处理过程结束后，会进入空闲状态，等待其他触发。

推理处理程序根据模型类型以及模型类型对特定状况作出反应的方式指定模型的行为。它们可以定义：

- 模型在创建、销毁或激活时的行为。  
**注意：**模型在与正建模的托管元素建立必要的通信时会被激活。
- 模型在其属性值发生更改或者生成事件时的行为。
- 模型在形成与其他模型的新关联或者从现有关联中删除时的行为。
- 如何处理特定操作。

推理处理程序与知识库内的模型类型相关联，它们针对该模型类型的实例化模型执行。当某个模型类型的两个模型的外部条件以相同的方式发生更改时，这两个模型的反应也会相同。但是，反映该模型状态的特定模型的属性值会对推理处理程序产生影响。一个模型的属性值可以与其他模型的属性值不同。因此，即使外部条件相同，且推理处理程序对每个模型作出的反应相同，相同模型类型的两个模型的反应结果也可能是不同的。

例如，与表示路由器的模型类型相关联的推理处理程序旨在执行一项特定任务：每当新的路由器模型被实例化时，便会创建表示该路由器的接口的模型。此任务结束后，推理处理程序的作业即已完成。然后，推理处理程序会等待从此模型类型创建下一个路由器模型，使它可以再次执行此作业。

路由器模型类型具有记录路由器上所存在接口的数目和类型的属性。每个实例化的路由器模型都表示计算基础架构中的一个特定路由器。并且，很有可能每个路由器模型的这些属性都具有不同的值。推理处理程序创建的接口模型的数目和类型基于这些值。因此，如果在知识库中创建表示网络中不同类型的路由器的多个路由器模型，同一个推理处理程序会为每个新路由器模型创建不同（但适当）数目和类型的接口模型。

在路由器模型接收到实际路由器已被重新配置的通知后，之前提到的推理处理程序可以再次被触发。重新配置可导致路由器上接口的数目或类型发生更改。发生此更改后，推理处理程序会使用新信息重新创建这些接口模型。这种动态的自适应建模功能是推理处理程序在整个 CA Spectrum 中基本应用的一个示例。

#### 详细信息：

[报警、事件和警报 \(p. 23\)](#)

## 操作

CA Spectrum 定义了可以在模型上执行的一组操作，例如读取或写入属性。为了扩展可能进行的操作，CA Spectrum 允许操作。操作是不属于 CA Spectrum 定义用于模型的基本操作集的任意操作。向模型发送操作会导致模型类型以某些方式作出反应。例如，它可以将请求的数据返回给操作发送方，或者也可以导致模型类型执行特定任务。

## 报告数据库

CA Spectrum Report Manager 使用 MySQL 数据库来存储在其报告中使用的数据。请考虑关于报告数据库的下列几点：

- 如果在不安装 Report Manager 的情况下安装 OneClick，将会安装一个本地 MySQL 数据库服务器。此本地数据库提供分布式数据管理器 (DDM) 数据库、MIB 工具数据库、eHealth 集成数据库以及报告数据库。

对于此安装，报告数据库仅用于存储 Service Manager 数据。有关详细信息，请参阅 [CA Spectrum 数据库](#) (p. 11)。

- 如果在 OneClick 安装中包含 Report Manager，则报告数据库会存储 Service Manager 数据。出于报告目的，该数据库还会存储资产数据以及历史事件和警报数据。

**注意：**资产数据以及历史事件和警报数据必须存储在位于 OneClick Web 服务器上的 MySQL 实例中。数据不能存储在单独的计算机上。

**注意：**有关详细信息，请参阅《*Report Manager 安装和管理指南*》以及《*Report Manager 用户指南*》。

## SpectroSERVER 和线程

SpectroSERVER 在同时访问磁盘和网络时，还处理来自许多客户端应用程序的请求。为了提高效率，SpectroSERVER 使用多线程体系结构，该体系结构的开销比运行单独进程要少。

SpectroSERVER 会在启动时创建一些只有在 SpectroSERVER 终止时才会终止的线程。SpectroSERVER 会动态创建其他线程，并在不再需要它们时将其终止。例如，每次客户端连接到 SpectroSERVER 或通过 API 发出请求时，就会启动一个新线程。

通常不必关注这种内部线程机制。但是，如果需要进行高级调整来最大化您的系统吞吐量，此概念在负载繁重的系统上就会变得非常重要。

## 托管元素

SpectroSERVER 使用模型来表示托管元素，这些模型基于在建模目录中定义的模型类型。有些模型类型可以进行实例化，以表示在计算基础架构中运行的设备、应用程序或主机。如果适用，SpectroSERVER 可以使用 SNMP 直接与这些托管元素进行通信。有些模型类型会实例化为充当容器并用于分组其他模型的模型。例如，可以创建一个 LAN 模型，以便把特定的托管元素组合到一个网段中。或者，可以创建一个房间模型，以便把托管元素组合到一个房间内。

容器模型可以包含其他容器模型和/或表示托管元素的模型，具体视容器模型类型而定。例如，IPClassB 容器可以包含一个表示路由器的模型，它还可以包含多个表示一系列子网的 LAN 模型。但是，建筑物模型只能包含容器模型：楼层、分区或房间。

SpectroSERVER 使用 *管理模块* 来管理计算基础架构的特定元素。管理模块由模型类型、关系、推理处理程序和支持文件组成。管理模块使用一系列模型来表示特定类型托管元素的各个组件。托管元素可以使用设备模型表示。其他类型的模型组合（如应用程序模型）支持设备功能。托管元素的每个主要功能组件都可以建模为单独的应用程序或包含在设备模型中。应用程序通常与 MIB 或该 MIB 的某部分的功能相对应。

用于表示托管元素的所有模型都基于在建模目录中定义的模型类型。模型彼此之间的关联基于在建模目录中定义的关系和元规则。

SpectroSERVER 可以实施各种关联，如下例所示：

- 容器模型可以包含表示托管元素或容器模型的模型
- 可以在设备或端口模型之间建立连接，以表示物理或逻辑连接
- 支持设备模型的应用程序模型可表达的关系显示了设备提供哪些功能

**注意：**并非在 CA Spectrum 知识库中定义的所有模型类型都可用于在 OneClick 中创建模型。一些模型类型只能充当其他模型类型从中进行派生的基础模型类型。

## 设备发现

CA Spectrum 支持发现和建模计算基础架构中的托管元素和连接的自动方法。*发现*是一种 OneClick 功能，用于自动发现 IT 基础架构中的实体并为之建模。您可以创建和编辑发现及建模配置，以便自定义和简化过程。通过发现功能，还可以筛选和导出发现或建模会话的结果。

发现由两个组件组成。发现应用程序扫描 IP 地址范围或列表，并从遇到的每个已启用 SNMP 的管理元素读取所选的一组关键 MIB 对象。结果将提供给您，或者发送给 SpectroSERVER 以供建模。在服务器端或后端，发现进程使用每个托管元素的关键 MIB 对象来确定要使用的最佳模型类型。然后，发现进程会创建该模型类型的实例以表示该托管元素。

在为发现找到的托管元素创建并激活模型后，后端发现进程会确定其位置和连接。模型位置和连接部分基于用户指定的选项。基于相应 IP 地址，网桥和工作站的模型会放置在 LAN 容器内。随后，这些模型使用从托管元素的 MIB 读取的生成树和源地址表连接到其他网桥。路由器的模型会放置在网络容器或 Universe 中。这些模型使用 IP 地址和掩码、IP 路由表信息或专有发现协议 MIB 连接到 LAN 模型或其他路由器。

通过 CA Spectrum，您还可以从 OneClick 控制台创建特定模型。可以使用两种方法。第一种方法使用托管元素的 IP 地址或 DNS 名称。通过此信息，SpectroSERVER 会联系托管元素，并检索托管元素的名称、供应商、说明、位置和 sysOID。然后，CA Spectrum 使用最能表示此设备功能的模型类型创建模型。

您也可以通过先选择一个模型类型作为基础来创建模型。在这种情况下，您需提供 IP 地址或 DNS 名称，以便 SpectroSERVER 可以与托管元素进行通信。但是，无论 SpectroSERVER 对托管元素功能的评估如何，选择的模型类型都会被实例化。SpectroSERVER 会创建所有相应的支持模型类型，并与托管元素描述的 MIB 功能相匹配。

## 设备通信管理器

设备通信管理器 (DCM) 是 SpectroSERVER 与托管元素之间的接口。DCM 包括各种协议接口，这些接口使用特定的协议与托管元素通信。对于受支持的两个协议 SNMP 和 ICMP，每个协议都存在一个接口。当 SpectroSERVER 与托管元素进行通信时，会将请求发送到 DCM 中的相应协议接口。DCM 然后将请求传递给托管元素。

## 正在轮询

SpectroSERVER 使用轮询和日志记录服务不断更新其网络状况的知识。DCM 会处理与被轮询的托管元素进行的通信。模型类型属性定义为外部（从托管元素获得）或内部（存储在内存或数据库中）。有些外部属性定义为 *已轮询*，这意味着 SpectroSERVER 会定期轮询托管元素。轮询频率取决于为模型定义的 `polling_interval` 属性值。未设置轮询的外部属性的值基于每个客户端应用程序或推理处理程序请求从托管元素获得。

**注意：**轮询会影响 SpectroSERVER 和网络的性能。缩短轮询时间间隔可限制 SpectroSERVER 的响应性，并且会生成不可接受的网络流量。

## 日志记录

属性也可以定义为将被记录，意味着它们的值会通过 Archive Manager 写入 DDM 数据库中。记录值的频率基于为模型定义的 `polling_interval` 和 `Poll_Log_Ratio`。

例如，可以将 `Poll_Log_Ratio` 设置为 10，将 `polling_interval` 设置为 60。此设置会在每十次轮询或每 600 秒将属性值记录到统计文件中。

**注意：**记录会影响 SpectroSERVER 和网络的性能。降低日志记录率可限制 SpectroSERVER 的响应性，并且会生成不可接受的网络流量。

## 报警、事件和警报

CA Spectrum 是一个服务和基础架构管理系统，会在您的网络中的托管元素出现故障时向您发送通知。CA Spectrum 实现此功能的一种方法是接收来自计算基础架构中问题区域的报警（通常是 SNMP 陷阱）。然后，CA Spectrum 会将这些报警转换为要在 CA Spectrum 应用程序中显示的事件和警报。CA Spectrum 使用一系列命名为事件配置文件的支持文件来指示处理报警、事件和警报的方式。

**详细信息：**

[推理处理程序](#) (p. 18)

## 报警

报警是托管元素发送给 CA Spectrum 的未经请求的消息。CA Spectrum 用来与托管元素进行通信的主要管理协议是 SNMP。SNMP 兼容的托管元素可以发送报警，也称为陷阱。支持 SNMP 陷阱的托管元素可配置为将其陷阱定向到 SpectroSERVER。SpectroSERVER 使用陷阱的源 IP 地址来标识与该托管元素关联的模型。标识该模型后，将按照与该模型类型关联的 AlertMap 文件的说明来处理陷阱。CA Spectrum 中的大多数设备模型类型都存在 AlertMap 文件。AlertMap 是一个 ASCII 文件，用于将 SNMP 陷阱映射到 CA Spectrum 事件中。

## 事件

事件是表示 CA Spectrum 中瞬时状况的对象。事件通常表示在模型或其他组件上发生了重要的事情。大多数设备模型类型都具有关联的 EventDisp 事件配置文件。EventDisp 文件是一个指示事件处理方式的 ASCII 文件。在 AlertMap 文件将 SNMP 陷阱转换为事件之后，EventDisp 文件会告知 CA Spectrum 如何处理此模型的事件。事件的处理可以包括记录事件和生成警报。

## 警报

警报是一个对象，指示在托管环境中存在用户可操作的异常状况。通常在发生了事件且 `EventDisp` 文件指定警报生成后，会生成一个警报。配置的监视也可以生成警报，因为 `CA Spectrum` 可以检测到未基于事件的异常情况。在导致警报的异常状况结束后，其他事件会清除相应的警报，或者您可以将其清除。可以将警报通知发送给需要此信息的应用程序和推理处理程序。`CA Spectrum` 可以检查各种网络事件，分析它们，并生成一些重要警报。

事件格式和可能原因文件有助于显示与事件和警报有关的信息。

**注意：**有关事件格式和可能原因文件的详细信息，请参阅《*事件配置用户指南*》。

## 格局和分布式 SpectroSERVER

格局是 `CA Spectrum` 中的一个术语，表示单个 `SpectroSERVER` 管理的网络域。格局由属于特定 `SpectroSERVER` 的模型、关联、属性值、警报、事件和统计信息组成。网络中包含的每个格局都是唯一的，并且每个格局通过一个唯一的格局句柄 (ID) 来标识。格局图标可以在 `OneClick` 控制台中表示每个格局。格局图标提供 `SpectroSERVER` 知识库的图形表示。

分布式 `SpectroSERVER` (DSS) 是一个强大的建模功能，能够分布式管理大规模网络的各个部分。可按地理位置或者跨单个物理位置中的多个服务器进行分布。在通过分布管理通信引入的网络负载来管理计算基础架构时，DSS 可以提高 `CA Spectrum` 性能。DSS 还可以将管理功能委派给远程工作站。

通过使用 DSS，可以创建计算基础架构（包括多个格局，且每个格局都具有其自己的本地 `SpectroSERVER`）的统一表示。在 DSS 环境中，`SpectroSERVER` 客户端（如 `OneClick` 控制台）可以同时从多个 `SpectroSERVER` 访问信息。

**注意：**《*分布式 SpectroSERVER 管理员指南*》包含有关如何将网络分段为格局的提示。

在使用 DSS 对多个格局进行建模时，每个格局的数据库必须包含相同的建模目录。一个格局的建模目录中存在的所有模型类型还必须存在于每个其他格局的建模目录中。因此，如果在一个格局中安装附加应用程序，则必须在每个格局上安装相同的应用程序。例如，如果在一个格局中安装 `VPN Manager`，则需在所有格局中安装 `VPN Manager`。

可使用主目录的概念轻松管理分布式环境中的所有建模目录。主目录是您指定要更新格局映射中其他 SpectroSERVER 的 SpectroSERVER。需要更改时，会对主目录进行更改。整个主目录会手动复制到格局映射中的所有其他 SpectroSERVER 中，从而传播所有更改并使建模目录保持一致。

**注意：**有关详细信息，请参阅《*分布式 SpectroSERVER 管理员指南*》。



## 第 3 章： 客户端应用程序

---

此部分包含以下主题：

[客户端应用程序概述](#) (p. 27)

[OneClick 控制台](#) (p. 28)

[关于 AlarmNotifier](#) (p. 29)

[关于 SANM](#) (p. 31)

[使用 CA Business Intelligence \(CABI\) 报告](#) (p. 32)

### 客户端应用程序概述

主要的 CA Spectrum 客户端应用程序是 OneClick。通过其他几个 CA Spectrum 客户端应用程序，您还可以与在 SpectroSERVER 上存储和处理的信息进行交互。

当自定义 CA Spectrum 或与 CA Spectrum 进行集成时，可以使用以下客户端应用程序：

- **AlarmNotifier:** 此应用程序用于将警报数据转发到用户定义的脚本或第三方应用程序。有关详细信息，请参阅[关于 AlarmNotifier](#) (p. 29)。
- **SANM:** 此应用程序与 AlarmNotifier 一起使用，可指定用于筛选发送给用户定义的脚本或第三方应用程序的警报数据的策略。有关详细信息，请参阅[关于 SANM](#) (p. 31)。
- **Report Manager:** 此应用程序是一个功能全面的系统，可通过从 SpectroSERVER 上的知识库中提取的数据创建报告。Report Manager 依赖于两个其他客户端应用程序：[CA Business Intelligence](#) (p. 32) 和 [InfoView](#) (p. 32)。

这些应用程序中的每个应用程序还有其自己的《*用户指南*》，可在 CA Spectrum 文档总目录中找到。

下列应用程序不是 CA Spectrum 客户端，但是一些可选的 CA Spectrum 自定义和集成所必需的。

- **进程后台程序：**进程后台程序是一个启动和跟踪后台程序的进程，后台程序允许 CA Spectrum 控制工作站上运行的各种进程。当应用程序（如控制面板）发出请求时，进程后台程序会启动进程。进程后台程序还可以在系统启动时启动进程（如果已配置）。如果关键进程意外停止，进程后台程序会自动重新启动它们。CA Spectrum 控制面板是唯一一个由您（即 CA Spectrum 用户）主动启动的可执行程序。用户或其他应用程序请求后，进程后台程序会启动其他所有应用程序。进程后台程序在后台运行，对您透明的。在 CA Spectrum 安装期间，只要系统启动，进程后台程序就会自动启动。
- **模型类型编辑器：**此应用程序用于派生支持新管理模块开发的新模型类型。

## OneClick 控制台

OneClick 控制台使用图标和视图来显示 SpectroSERVER 中的信息。图标是模型的图形说明，定义为表示您的计算基础架构的托管元素。视图是用于组织 SpectroSERVER 中要显示的数据的各种方式。

### OneClick 控制台图标

图标是基于 CA Spectrum 建模目录中模型类型的实例化模型的图形表示。各种图标可以表示单个托管元素、托管元素组、地理位置、用户、格局、模型之间的连接等等。管道是一种特殊类型的图标，用于表示托管元素之间的连接。

有关模型的常规信息（如模型名称和模型类型名称）显示在图标上。有关模型的详细信息，可以通过双击图标访问各种图标子视图来查看。有些图标使用颜色来指示它们所表示的托管元素的状况。

### 分层视图

CA Spectrum 中的视图一种数据组织方式，便于数据的显示或处理。分层视图表示构建网络数据的方式。在 xml 文件中构建网络数据时，请从表示各分层视图的元素中进行选择。存在两种分层视图：拓扑和位置。

## 拓扑视图

拓扑视图实际上是网络组件的抽象化。使用此视图时，您表示的是您网络中的物理或逻辑组件，对这些组件进行分组时应考虑其逻辑连接。还可以使用显示端口级别或设备级别设备连接情况的管道，以图形方式来表示连接。在 OneClick 控制台中，此视图显示为 Universe 拓扑。

**注意：**有关 Universe 拓扑视图的详细信息，请参阅《IT 基础架构建模与管理 - 管理员指南》。

## 位置视图

位置视图可按照物理位置组织您的网络数据。使用此视图时，可以通过地理措词来描述您的网络。可以从全球办事处开始，一直到您办事处所在各个地区各建筑物各楼层的配线柜。在 OneClick 控制台中，此视图显示为 World 拓扑。

**注意：**有关 World 拓扑视图的详细信息，请参阅《IT 基础架构建模与管理 - 管理员指南》。

## 关于 AlarmNotifier

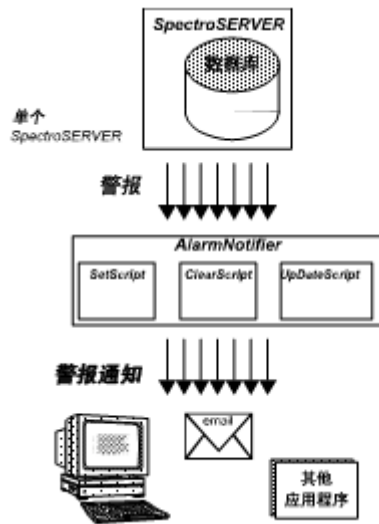
AlarmNotifier 是随核心 CA Spectrum 组件一起安装的 SpectroSERVER 客户端应用程序。AlarmNotifier 应用程序连接到单个 SpectroSERVER，并调用脚本来提供有关 CA Spectrum 警报状态的通知。

从终端 shell 命令提示符下启动 AlarmNotifier。一旦启动后，不管是设置、清除还是更新了警报，它便会持续显示被调用脚本的输出。AlarmNotifier 为 CA Spectrum 提供以下功能：

- 单个 SpectroSERVER 警报监控。
- 生成警报信息的三个脚本：SetScript、ClearScript 和 UpdateScript。  
这些脚本包含可以针对您的环境自定义的设置。
- 您可以通过配置资源文件参数来修改 AlarmNotifier 的运行功能。

## 警报监控过程

AlarmNotifier 增强了 CA Spectrum 警报监控和通知功能。下图说明了 AlarmNotifier 和 CA Spectrum 之间的关联关系：



CA Spectrum 执行一些警报功能，而 AlarmNotifier 执行其他功能。CA Spectrum 轮询已建模的网络元素，并更新 SpectroSERVER 数据库中存储的每个元素的状态信息。

在从网络接收陷阱或检测到网络元素模型中存在关键的状态更改时，CA Spectrum 会生成警报。在 OneClick 拓扑视图中，模型图标由绿色更改为其他颜色指示了警报重要级别。CA Spectrum 在“警报”选项卡中发布有关警报的信息。有关该警报的事件信息显示在 OneClick “内容”面板的“事件”选项卡中。

在 AlarmNotifier 启动时，它会在 CA Spectrum 中注册。然后，会创建一个名为 AlarmNotifier、ClientApp 类型的模型。该模型在任何 CA Spectrum 拓扑视图中都不可见。但是，您可以在“事件”选项卡中看到它。“事件”选项卡显示了与该模型相关的信息，如应用程序的启动和停止时间。

AlarmNotifier 会查询 SpectroSERVER 并请求有关现有警报的信息。AlarmNotifier 可运行脚本并生成有关现有警报的通知。

每次设置、清除或更新警报时，AlarmNotifier 都会从 SpectroSERVER 接收信息并调用相关脚本。AlarmNotifier 脚本可以启动电子邮件通知，以便将警报发送给网络中的人员。还可以将警报信息传送给第三方应用程序。

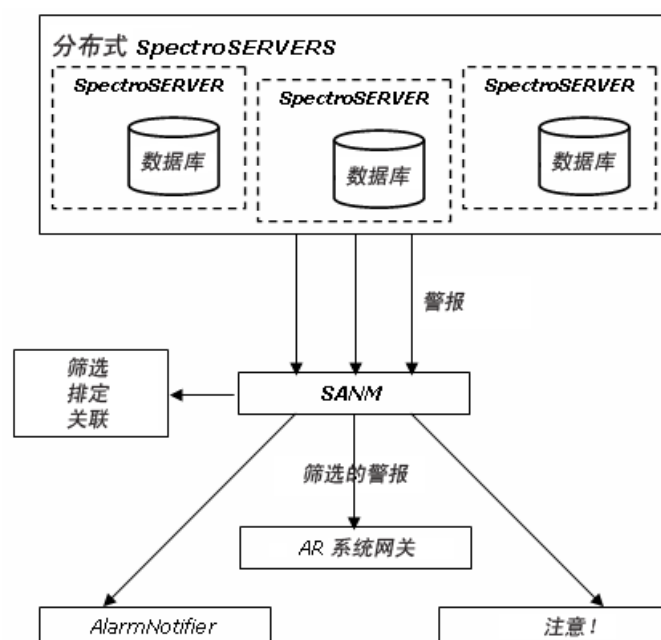
## 关于 SANM

CA Spectrum Alarm Notification Manager (SANM) 是 CA Spectrum 组件，该组件可以增强 CA Spectrum 警报处理应用程序的功能。CA Spectrum 可使用多个警报处理应用程序，包括 AlarmNotifier 和 Attention!。这些应用程序通过发送电子邮件通知、创建故障单等操作来响应 CA Spectrum 警报。通过 SANM，您可以创建警报通知策略，并将其与应用程序相关联。

### CA Spectrum 如何监控警报

CA Spectrum、警报处理应用程序和 SANM 在警报监控过程中协同工作。

下图显示了警报监控过程：



下面的工作流说明了 CA Spectrum 监控警报的方式：

1. CA Spectrum 轮询建模的网络元素，并更新 SpectroSERVER 数据库中每个元素的状态。
2. CA Spectrum 在从网络接收到陷阱或检测到网络模型中的关键状态更改时，会生成警报。在 OneClick 控制台中，模型图标从绿色更改为指示警报重要级别的其他颜色。
  - CA Spectrum 在“组件详细信息”窗格的“警报详细信息”选项卡上为每个警报发布具体信息。
  - CA Spectrum 在“组件详细信息”面板的“事件”选项卡上发布警报事件信息。

3. 有关 CA Spectrum 已生成的警报的数据将传递给 SANM。通过 SANM，您可以创建警报通知策略，并将其与警报处理应用程序相关联。此外，通过“SANM 排定”子视图，您可以排定应用程序与策略的关联，并自动完成关联过程。
4. 仅当出现策略中指定的警报类型时，SANM 才会将警报信息传递给警报处理应用程序。

## 使用 CA Business Intelligence (CABI) 报告

CA Spectrum 报告组件使用 CA Business Intelligence (CABI) 来显示报告。

CABI 是 CA Spectrum 和其他 CA 产品用于表示信息的报告和分析软件包。CA Spectrum 使用 CABI 来集成、分析和提供通过报告进行有效的企业 IT 管理所需的信息。

CABI 包括 SAP BusinessObjects Enterprise XI，即一个完整的信息管理、报告、查询和分析工具套件。

CABI 将 SAP BusinessObjects Enterprise XI 作为一个独立组件安装。该安装独立运行 CA Spectrum 和其他 CA 产品，允许各种 CA 产品共享 Business Intelligence 服务。CABI 安装是总体 CA 产品安装过程中一项独特的单独活动。

**注意：**有关详细信息，请参阅《CA Business Intelligence 实施指南》和《CA Business Intelligence 版本说明》。

## InfoView 报告管理

BusinessObjects Enterprise InfoView (InfoView) 是一个基于 Web 的界面，允许您通过以下功能管理报告：

- 浏览和搜索功能。
- 内容访问（创建、编辑和查看）。
- 内容排定和发布。

InfoView 的功能类似于 Windows 应用程序，而不是简单的 Web 应用程序。InfoView 工具栏会动态地发生更改，以通过与要执行的功能一致的上下文菜单来提供操作。InfoView 提供了上下文菜单，您只需单击鼠标右键即可。可以双击窗口中各项来执行默认操作。报告结构一致，并且提供强大的安全性和授权。

InfoView 可提供对 WebIntelligence (WEBI) 设计器的访问权限。通过此设计器，您可以使用简单的拖放界面创建自定义报告。具有有效筛选选项的自定义数据对象选择项能够为您的环境提供强大的报告功能。

可以从 OneClick 主页或直接从 Web 浏览器访问 InfoView。典型的 URL 格式如下所示：

`http://<hostname>/InfoViewApp`

**注意：**有关详细信息，请参阅《*Report Manager 用户指南*》。



# 附录 A： 属性和关系定义

---

此部分包含以下主题：

[属性](#) (p. 35)

[属性说明](#) (p. 38)

[关系说明](#) (p. 46)

## 属性

属性的概念及其在模型和模型类型方面的角色在[模型类型属性](#) (p. 14)中进行了概述。

知识库中的许多属性是 MIB 变量的对等项。外部属性直接对应于特定 MIB 变量。内部属性值可以从 MIB 变量的值派生。这些值通常经过一些数学计算，以达到其 CA Spectrum 值。命名约定通常会使属性和 MIB 变量之间的关联关系变明显。

下表显示了已定义的属性，可帮助您将其他应用程序与 CA Spectrum 相集成。在创建管理模块，使用 Southbound Gateway 工具包，或使用建模网关工具包时，可使用这些属性。每个表都是快速参考，其中属性按功能进行分组。

**注意：** 可以从模型类型编辑器访问有关属性和关系的特定信息。有关详细信息，请参阅 *《模型类型编辑器用户指南》*。

## 应用程序模型发现

属性	属性 ID	发现位置
default_attr	0230006	应用程序模型类型

## 设备模型发现

属性	属性 ID	发现位置
DeviceNameList	0x1293E	设备模型类型
DeviceType	0x23000e	设备模型类型

Disposable_Precedence	0x114e2	设备模型类型
Enable_IH_Spec_Dev_Name	0x3d0062	设备模型类型
Enable_IH_Device_Name	0x3d0008	设备模型类型
Image_Index	0x3d0001	设备模型类型
System_OID_Verify	0x110bb	设备模型类型
System_OID_Verify_List	0x12910	设备模型类型

## 常规模型类型信息

属性	属性 ID	发现位置
CompanyName	0x118b8	设备和应用程序模型类型
说明	0x230017 和 0x118bc	设备、应用程序和接口模型类型
DeviceType	0x23000e	设备模型类型
Manufacturer	0x10032	设备模型类型
MMName	0x1196a	设备和应用程序模型类型
MMPartNumber	0x1196b	设备、应用程序和接口模型类型
Model_Class	0x11ee8	设备模型类型
Model_Name	0x1006e	设备、应用程序和接口模型类型
Modeltype_Name	0x10000	设备、应用程序和接口模型类型
Vendor_Name	0x11570	设备、应用程序和接口模型类型

## 网络信息

属性	属性 ID	发现位置
Network_Address	0x12d7f	设备、应用程序和接口模型类型
Network_Mask	0x12dbc	设备、应用程序和接口模型类型

## 轮询信息

属性	属性 ID	发现位置
polling_interval	0x10071	设备、应用程序和接口模型类型
poll_log_ratio	0x10072	设备、应用程序和接口模型类型
pollingstatus	0x1154f	设备、应用程序和接口模型类型

## 端口标识

属性	属性 ID	发现位置
ifAlias	0x11f84	接口模型类型
if_Index	0x11348	接口模型类型
ifName	0x11f60	接口模型类型
if_Phys_Addr	0xd0399	接口模型类型
ip_address	0x12dbb	接口模型类型

## SNMP 信息

属性	属性 ID	发现位置
Community_Name	0x10024	设备、应用程序和接口模型类型
CommunityNameForSNMPsets	0x11a7f	设备、应用程序和接口模型类型
isManaged	0x1295d	设备模型类型
Security_String	0x10009	设备、应用程序和接口模型类型

## 属性说明

本节提供前面的表中所述属性的相关详细信息。

**注意：**您可通过使用模型类型编辑器来访问有关各属性的详细信息。提供的详细信息包括属性 ID、属性值的数据类型或为属性设置的属性标志等。请参阅《*模型类型编辑器用户指南*》。

可为每个属性设置多个属性标志，其中包括以下主要标志：

- **外部：**如果设置为 TRUE，则此标志指示此属性的值在 SpectroSERVER 的外部进行维护。此设置还指示属性值的更新可在用户请求时执行，也可以按轮询时间间隔执行。
- **可读：**如果设置为 TRUE，则此标志会通知 SpectroSERVER 有客户端或其他应用程序被允许从 SpectroSERVER 读取此属性值。如果设置了“外部”标志，则会根据此属性的可读变量的 MIB 定义设置此标志。如果未设置“外部”标志，则会根据需要设置此标志。
- **可写入：**如果设置为 TRUE，则此标志会通知 SpectroSERVER 有客户端或其他应用程序可以将此属性值写入 SpectroSERVER 数据库。如果设置了“外部”标志，则会根据此属性的变量的 MIB 定义设置此标志。如果未设置“外部”标志，则会根据需要设置此标志。
- **共享：**如果设置为 TRUE，则此标志会声明此属性存在一个值，并且当前模型类型的所有模型都共享该值。该值在内存或数据库中对每个模型都不重复。

以下列表包含一个或多个 CA Spectrum 集成点所使用或引用的属性的说明。

### Community\_Name

标识在 CA Spectrum 尝试使用 SNMP 与托管元素进行通信时所使用的 SNMP 团体字符串。此属性在执行 SNMP get 时进行评估。如果属性 CommunityNameForSNMPsets 为空，则还会在执行 SNMP set 时使用此属性。

### CommunityNameForSNMPsets

指定在执行 SNMP set 时使用的团体名称。如果保留为空，则 SNMP set 会使用 Community\_Name 属性值。

### CompanyName

指定设备和应用程序模型类型使用的公司名称。此值设置为等于开发模型类型的公司的名称。

**default\_attr**

标识特定托管元素支持的应用程序。将在应用程序发现进程中使用此属性。default\_attr 的值设置为等于表示唯一标识 MIB 的 MIB 对象的属性的属性 ID。

**说明**

提供模型类型的文本说明。

**DeviceNameList**

包含对应于 SysOIDVerifyList 属性中 OID 的设备名称。要使用此属性来设置设备名称，请将 Enable\_IH\_Device\_Name 属性和 Enable\_IH\_Spec\_Dev\_Name 属性设置为 TRUE。

**DeviceType**

当模型类型仅表示一种特定类型的设备时，保留该设备类型的名称。此属性的值显示在模型的图标底部的字段中。如果给定设备存在多个设备类型，请使用 DeviceNameList 属性。

**Disposable\_Precedence**

包含 CA Spectrum 设备发现机制用于解决设备模型类型选择进程中的冲突的值。当多个模型类型具有与托管元素的 SystemObjectID 相匹配的 System\_OID\_Verify 值时，就会发生冲突。为了解决该冲突，发现会使用具有最高 Disposable\_Precedence 值的模型。

**Enable\_IH\_Spec\_Dev\_Name**

如果 Enable\_IH\_Spec\_Dev\_Name 属性为 TRUE，CA Spectrum 会使用 Enable\_IH\_Device\_Name 推理处理程序，通过设备中的企业编号来确定供应商名称。

**Enable\_IH\_Device\_Name**

如果 Enable\_IH\_Device\_Name 属性为 TRUE，CA Spectrum 会使用 Enable\_IH\_Spec\_Dev\_Name 推理处理程序来读取设备系统对象 ID，以确定产品名称。

**ifAlias**

对应于 MIB II 接口表中的此值。

**ifIndex**

对应于 MIB II 接口表中的此值。

**ifName**

对应于 MIB II 接口表中的此值。

**if\_Phys\_Addr**

对应于 MIB II 接口表中的此值。

### **Image\_Index**

将 GnSNMPDev 模型与表示该模型的图标图像相链接。可能值及其相应的图像如下所示：

- 1**  
常规设备
- 2**  
网桥
- 3**  
路由器
- 4**  
集线器
- 5**  
PC
- 6**  
终端服务器
- 7**  
Workstation
- 8**  
交换机

### **ip\_address**

指定与接口模型关联的 IP 地址。

### **isManaged**

指示设备模型是否受管理。如果此属性设置为 TRUE，CA Spectrum 将使用 SNMP 通信管理此设备。

### **Manufacturer**

与设备模型一起使用，可显示负责该设备的制造商。

### **MMName**

保留设备和应用程序模型的管理模块名称。

### **MMPartNumber**

包含管理模块开发者已分配给管理模块的部件号。大多数设备、应用程序和接口模型都使用此属性。

**Model\_Class**

Model\_Class 定义模型所表示设备的类型。下表列出了 CA Spectrum 中可用的模型类及其各自的整数标识符。例如，模型类 SWITCH 使用标识符 2。

<b>0</b>	未知
<b>1</b>	其他
<b>2</b>	SWITCH
<b>3</b>	ROUTER
<b>4</b>	SWITCH_ROUTER
<b>5</b>	HUB
<b>7</b>	LINK
<b>8</b>	NETWORK
<b>9</b>	WORKSTATION_SERVER
<b>10</b>	CONTAINER
<b>11</b>	CHASSIS
<b>12</b>	PINGABLE
<b>13</b>	MAC
<b>14</b>	SNMP

- 15**  
PORT
- 16**  
USER
- 17**  
APPLICATION
- 18**  
COMPONENT
- 19**  
LANDSCAPE
- 20**  
ROUTER\_APP
- 21**  
SWITCH\_APP
- 22**  
BRIDGE\_APP
- 23**  
MIB\_APP
- 24**  
RMON\_APPL
- 25**  
UNIX
- 26**  
NT
- 27**  
FIREWALL
- 28**  
IDS
- 29**  
SECURITY\_SCANNERS
- 30**  
ANTI\_VIRUS\_APPLICATION

- 
- 31**  
PKI\_SYSTEMS
  - 32**  
PACKET\_SNIFFER
  - 33**  
SYSLOG
  - 34**  
RESPONSE\_TIME\_TEST
  - 35**  
RESPONSE\_TIME\_TEST\_HOST
  - 36**  
TRANSPORT\_SERVICE
  - 37**  
GENERIC\_TL1\_DEVICE
  - 38**  
VOIP
  - 39**  
CMTS
  - 40**  
WIRELESS
  - 41**  
CABLE\_MODEM\_MTA
  - 42**  
VPN
  - 43**  
DSL
  - 44**  
MULTIPLEXOR
  - 45**  
SAN
  - 46**  
PBX

<b>47</b>	USER_CUSTOMIZATION
<b>48</b>	PRINTER
<b>49</b>	TRANSPORT_DEVICE
<b>50</b>	SERVICE_MGT_COMPONENT
<b>51</b>	SLA_COMPONENT
<b>52</b>	CUSTOMER
<b>53</b>	PROCESS
<b>54</b>	DIAGNOSTIC_DATA
<b>55</b>	DIAGNOSTIC_COMPONENT
<b>56</b>	HOST_CONFIGURATION
<b>57</b>	DIAGNOSTIC_SCRIPT
<b>103</b>	POWER_SUPPLY
<b>104</b>	AMPLIFIER
<b>105</b>	LINE_MONITOR
<b>106</b>	TEST_POINT
<b>107</b>	FIBER_NODE

**108**

HEFIBER

**109**

IP\_PHONE

**110**

TELCO

**111**

LOAD\_BALANCER

**112**

LMT

**113**

FILE\_SERVER

**114**

ENVIRONMENTAL

**115**

AUTOMATIC\_TELLER\_MACHINE

**Model\_Name**

指定提供给模型的名称。模型名称将该模型与该类型的其他模型区分开来。

**Modeltype\_Name**

指定模型类型的文本名称。

**Network\_Address**

包含设备模型的网络地址。CA Spectrum 使用此值来标识模型并与其进行通信。此值通常是 IP 地址。

**Network\_Mask**

进一步标识设备模型在网络上的逻辑位置。

**polling\_interval**

由大多数设备、应用程序和接口模型使用。此属性的值标识在轮询请求之间必须传递的秒数。可以在“模型信息”视图中设置此属性。

**poll\_log\_ratio**

标识记录轮询的属性的轮询周期。例如，值为 10 表示每十个轮询周期记录一次轮询的属性。在大多数设备、应用程序和接口模型上都会找到此属性。

### **pollingstatus**

控制是否为模型执行轮询。大多数设备、应用程序和接口模型都使用此属性。TRUE 值表示已为模型启用轮询。FALSE 值表示已为模型禁用轮询。

### **Security\_String**

定义可以访问此模型的 SNMP 团体字符串。大多数应用程序、设备和接口模型都使用此属性。

### **System\_OID\_Verify**

包含发现用于确定表示设备的相应模型类型的系统对象 ID 值。CA Spectrum 会在实例化模型时分配模型类型，以表示托管元素。CA Spectrum 通过将托管元素的系统对象 ID 与该模型类型的 System\_OID\_Verify 值相匹配，来标识相应的模型类型。如果发现多个模型类型具有某个匹配值，则会使用具有最高 Disposable\_Precedence 值的模型。如果模型类型用于表示设备系列，则会使用 SysOIDVerifyList 属性，而不是 System\_OID\_Verify 属性。

### **SysOIDVerifyList**

包含发现用于确定表示设备的相应模型类型的系统对象 ID 值的列表。CA Spectrum 会在实例化模型时分配模型类型，以表示托管元素。CA Spectrum 通过将托管元素的系统对象 ID 与模型类型 SysOIDVerifyList 列表中的值相匹配，来标识相应的模型类型。如果发现多个模型类型具有某个匹配值，则会使用具有最高 Disposable\_Precedence 值的模型。

### **Vendor\_Name**

标识托管元素的供应商。大多数设备、应用程序和接口模型都使用此属性。

## 关系说明

下列关系属于 CA Spectrum 的最重要功能：

### **Connects\_to**

Connects\_to 关系是多对多关系类型。Connects\_to 形成两个模型之间的连接。此关系在端口模型和设备或拓扑模型之间建立连接。

### **包含**

包含关系是一对多关系类型。此关系允许模型包含其他模型。位置模型使用包含关系来确定位置模型中可以包含的位置或设备模型。当您尝试将模型添加或复制到位置模型中时，CA Spectrum 会测试此关系的规则。

### **HASPART**

HASPART 关系是多对多关系类型。HASPART 在设备模型和表示设备组件的模型之间建立关联。通常，这些组件是设备的接口模型。但是，此关系也会在组件模型及其组件之间建立关联。

### **Links\_with**

Links\_with 关系是一对多关系类型。此关系用于表示两个模型之间的解析的连接。通常会在双端口模型之间找到此关系。链接视图和活动管道都严重依赖于此关系。

### **管理**

管理关系是一对多关系类型。此关系在应用程序模型和正在运行该模型的设备模型之间形成关联。管理关系还可以在元素管理系统模型和此系统的元素模型之间形成关联。

### **提供**

提供关系是一对多关系类型。此关系标识哪些应用程序提供哪些子应用程序。提供关系还可用于确定哪个应用程序提供了特定的子应用程序。

### **详细信息:**

[关系](#) (p. 14)



# 词汇表

---

## AlertMap 文件

每个模型类型都存在一个 *AlertMap* 文件。此文件将传入 SNMP 陷阱数据映射到 CA Spectrum 事件。

## CA Spectrum 扩展集成 (SEI) 工具包

*SEI 工具包* 是用于打包和分发集成，使之可以在其他 CA Spectrum 主机上安装的一组工具。

## CORBA (公共对象请求代理体系结构)

*CORBA* 是允许各种不同的软件组件进行通信的软件组件体系结构。可使用不同的语言 (C、C++、Java、Smalltalk、COBOL、ADA、Lisp、Perl、Tcl、Eiffel、Python) 编写软件组件。软件组件也可以驻留在其他计算机上，并且可以为其他操作系统编写。

## EventAdmin

*EventAdmin* 是表示第三方管理系统的 Southbound Gateway 模型类型。

## EventDisp 文件

每种模型类型都存在 *EventDisp* 文件，该文件可确定 CA Spectrum 处理模型类型事件的方式。

## EventModel

*EventModel* 是 CA Spectrum 中的一种模型类型，用于表示 Southbound Gateway 集成中的唯一报警源。

## ICMP (Internet 控制消息协议)

*ICMP* 支持包含错误、控件和信息消息的数据包。

## MIB (管理信息库)

*MIB (管理信息库)* 是驻留在网络设备上的数据库，可将设备表示为对象的分层集合。MIB 对象表示单个信息元素，如设备的正常运行时间。MIB 本身是具有特殊语法的文本文件。

## mkcd

*mkcd* 是 CA Spectrum 扩展集成 (SEI) 工具包中的一个工具。此工具通过添加版本号并使 VCD 在 CA Spectrum 主机上可安装，来完成 VCD。

## mkmm

*mkmm* 是 CA Spectrum 扩展集成 (SEI) 工具包中的一个工具。此工具使用索引文件和适用的组件文件创建 VCD。

## OID

*OID* 是托管对象的标识符。

---

## OID 映射

*OID 映射*是用于将报警变量映射到事件变量的语法。

## SANM

CA Spectrum Alarm Notification Manager (*SANM*) 是 CA Spectrum 组件，该组件可以增强 CA Spectrum 警报处理应用程序的功能。

## SNMP 陷阱

*SNMP 陷阱*是广播或定向到网络管理应用程序，以通知应用程序发生了设备或网络活动的事件。已启用 *SNMP* 的设备或应用程序会生成陷阱。

## SNMP（简单网络管理协议）

*SNMP（简单网络管理协议）*是用于监控网络上设备和应用程序的网络协议。

## SpectroSERVER

*SpectroSERVER* 是控制与数据库和托管元素进行通信的 CA Spectrum 软件。*SpectroSERVER* 还充当应用程序（如用户界面）与数据库和设备提供的信息之间的接口。*SpectroSERVER* 中的主要组件包括设备通信管理器 (DCM) 和虚拟网络计算机 (VNM)。

## SSORB 客户端

*SSORB 客户端*是可访问 CA Spectrum CORBA 界面 *SSORB* 的应用程序。

## VCD（虚拟 CD）

*VCD（虚拟 CD）*是 CA Spectrum 扩展集成 (SEI) 工具包创建的一系列文件。*VCD* 允许开发者轻松打包和分发其集成。

## 元规则

*元规则*指定哪些模型类型可以参与关系。

## 元素管理制度

*元素管理制度*允许您开通、管理和监控网络基础架构中的元素。

## 分布式 SpectroSERVER (DSS)

*分布式 SpectroSERVER (DSS)* 是一个建模功能，使用格局的概念在管理大型计算基础架构时提高 CA Spectrum 性能。*DSS* 技术分发管理通信引入的负载，并允许您将管理功能委派给远程工作站。

## 开发 API

*开发 API* 是 *SpectroSERVER* 所使用的基于 *CORBA* 的应用程序接口 (API)。*开发 API* 也称为 *SpectroSERVER* 对象请求代理 (*SSORB*) 接口，因为它依赖于对象请求代理 (*ORB*)。

---

## 开发者 ID

*开发者 ID* 包括 14 个字符的开发者名称和四个字符的前缀。开发者名称必须是字母数字，并且可以包括下划线，但是不允许包括任何其他标点符号。前缀也必须是字母数字，但是不允许包括下划线或任何其他标点符号。CA Spectrum 使用 *开发者 ID* 来验证由用户或集成人员创建的对象（如模型类型、属性或关系）是否具有唯一标识符。因此，这些对象可以分发给其他用户，而不会产生冲突。  
要获得开发者 ID，请与 CA 支持部门联系。

## 发现

*发现* 是一种 OneClick 功能，用于自动发现 IT 基础架构中的实体并为之建模。您可以创建和编辑发现及建模配置，以便自定义和简化过程。通过发现功能，还可以筛选和导出发现或建模会话的结果。

## 可轮询属性

*可轮询属性* 是 VNM 定期查询托管元素以获得当前值的属性。属性可通过模型类型编辑器定义为可轮询或不可轮询。

## 可能原因文件

*可能原因文件* 是 ASCII 文本文件，可定义警报的症状、可能原因和建议的更正操作。事件生成警报后，关联的可能原因文件中的文本会显示在 OneClick 中的“警报详细信息”选项卡上。

## 关系

*关系* 是用于描述模型相互间连接的信息。

## 关联

*关联* 是关系在两个模型之间形成的链接。

## 协议

*协议* 是计算机用来相互进行通信的一组规则。

## 同步调用

*同步调用* 是对在程序可以继续之前执行整个请求操作的方法的一种调用。程序仅在调用的方法完成之后才会继续。

## 导航模式

*导航模式* 是从一个视图到 CA Spectrum 中下一个视图的移动。在导航模式下，菜单栏包含“文件”和“视图”选项。导航模式不允许编辑视图。

## 异步调用

*异步调用* 是对在允许程序继续之前开始执行（但不一定完成）请求操作的方法的一种调用。在某一时刻，请求的操作完成并通知程序。在此期间，程序和请求的操作都可以同时进行。

---

## 托管元素

*托管元素*是其状态正受监控和控制的项或设备。托管元素可以是网络设备、主机系统、应用程序、服务或其他计算基础架构组件。

## 托管对象

*托管对象*是包含一条有关节点信息的托管元素变量。每个节点可以具有多个对象。

## 网络管理代理

*网络管理代理*是管理协议的实施，可与管理站交换有关托管元素的信息。

## 网络管理协议

*网络管理协议*是管理站和托管元素交换信息的方式。

## 网络管理站

*网络管理站*是正在运行网络管理应用程序和协议的主机系统或工作站。

## 自定义安装脚本

*自定义安装脚本*是可以包含在 VCD 中的脚本。作为集成数据包执行的脚本正在 SpectroSERVER 上安装。

## 设备

*设备*是一个托管元素。

## 设备通信管理器 (DCM)

*设备通信管理器 (DCM)* 是 SpectroSERVER 中的多协议通信引擎，可处理与使用各种协议的所有托管元素进行的通信。DCM 将 SpectroSERVER 请求转换为各个设备理解的协议。

## 应用程序编程接口 (API)

*应用程序编程接口 (API)* 是用于调用其他软件包的一组例程。

## 报警

*报警*是从托管元素发送到 SpectroSERVER 的未经请求的消息。

## 报警代码

*报警代码*是标识报警的字符串。从 SNMP 源接收到的报警包含一个报警代码，该代码包含一个常规陷阱，然后紧跟一个小数点 (.)，其次是一个企业特定的陷阱。

## 事件

*事件*是来自 SpectroSERVER 的重要消息。

## 事件代码

*事件代码*是唯一标识事件的十六进制数。

---

## 事件变量 ID

*事件变量 ID* 是表示事件变量数据的 ID。使用此 ID 可返回事件变量值。

## 事件重要级别

*事件重要级别* 是 EventDisp 文件条目中存在的一个数值，用于指示事件的严重度。有效值从 0 到 100，其中 0 表示严重度最低。

## 事件格式文件

*事件格式文件* 包含在事件发生时 OneClick 控制台的“事件”选项卡中显示的事件的相关消息。

## 事件数据模板

*事件数据模板* 是 Southbound Gateway 用于格式化来自报警的变量数据的一系列整数。

## 图标

*图标* 是屏幕上显示内容的直观表示。

## 定时间隔

*定时间隔* 是当前视图更新显示的模型属性信息的频率。默认情况下，针对一定数量的属性每 5 秒钟对每个模型发出一次请求。

## 实例化

在面向对象的设计中，进行 *实例化* 可创建对象的特定实例。

## 实例变量

*实例变量* 存储 OID 的实例部分。如果变量绑定标识陷阱 MIB 内表变量中的特定对象，则它可能会包括实例 ID。

## 实例变量 ID

*实例变量 ID* 是用于标识 OID 映射中实例变量的整数。使用此 ID 可返回实例变量。

## 服务器

*服务器* 是提供服务以响应客户端-服务器体系结构中客户端请求的进程。

## 知识库

*知识库* 是 CA Spectrum 可以建模和管理的所有内容的集合，其中包括概念、关联关系、陈述性知识和程序性知识。

## 知识库管理系统

*知识库管理系统* 是用于在知识库中定义和管理信息的软件。

## 视图

*视图* 是网络格局的许多表示之一。

---

## 变量绑定

*变量绑定*是作为 SNMP 陷阱的一部分发送的变量数据。

## 客户端

*客户端*是客户端-服务器体系结构中的应用程序进程。

## 客户端-服务器体系结构

*客户端-服务器体系结构*是基于服务提供进程（*服务器*）和服务使用进程（*客户端*）之间关联关系的系统设计。

## 面向对象设计 (OOD)

*面向对象设计 (OOD)* 是包含将系统分解成多个部分的进程的设计。每个部分表示问题域中的某个类或对象。可通过将全局视为相互合作以实现某个所需功能的对象的集合来应用 OOD。OOD 通常包括一个符号，用于描绘设计中系统的逻辑或物理以及静态或动态模型。

## 值变量 ID

*值变量 ID* 是用于检索与 SNMP 陷阱一起发送的变量绑定的值的 ID。

## 格局

*格局*是特定于单个网络中的任一虚拟网络计算机 (VNM) 的所有数据。该术语也标识由单个 SpectroSERVER 管理的网络域。在 OneClick 中，格局是 SpectroSERVER 的网络视图。

## 索引文件

*索引文件*是使用 CA Spectrum 扩展集成 (SEI) 工具包创建的文件，可定义集成的组件。索引文件指示集成组件在主机上的存在位置及其在客户主机上的位置。

## 推理处理程序

*推理处理程序*是 CA Spectrum 背后的智能。推理处理程序监控环境的更改、相关模型之间的更改以及模型与属性之间关联关系的更改。

## 虚拟网络计算机 (VNM)

在 SpectroSERVER 中，虚拟网络计算机 (VNM) 是提供对存储在任意位置的数据的访问权限的软件层。数据可以存储在数据库、VNM 内存或网络上的任意托管元素中。VNM 还会具体化称为感应建模技术 (IMT) 的 CA Spectrum 智能。

## 属性

*属性*是 CA Spectrum 中用于定义模型类型的陈述性知识。可以使用模型类型编辑器来定义属性。

## 智能电路

*智能电路*是用于定义模型类型行为的推理处理程序的集合。

---

## 程序性知识

在 CA Spectrum 中, *程序性知识*是用于定义概念对环境更改的表现方式或反应方式。

## 编辑模式

*编辑模式*允许您在 CA Spectrum 中编辑当前的视图。选择“编辑模式”后, 将在菜单栏中显示“文件”和“编辑”选项。

## 感应建模技术 (IMT)

*感应建模技术 (IMT)* 是 CA 的一组人工智能技术。这些技术允许对任意复杂性的计算基础架构进行建模, 使每个元素都能获得智能。

## 数据库

*数据库*是进行组织以便于有效、准确的查询和更新的相关数据的集合。

## 数据库管理系统

*数据库管理系统*是用于组织和维护数据库的软件包。

## 模块化面向对象线程 (MOOT)

*模块化面向对象线程 (MOOT)* 是 SpectroSERVER 中使用的任务控制管理器。

## 模型

*模型*是形成某些基础定义类型的特定实例的信息集合。在 CA Spectrum 中, 模型是模型类型的实例。

## 模型类型

*模型类型*是用于说明与 CA Spectrum 中托管元素有关的属性、操作和关联的模板。

## 模型类型编辑器 (MTE)

*模型类型编辑器 (MTE)* 是用于定义 CA Spectrum 知识库中的概念、关联关系、元规则和陈述性知识的主要工具。

## 管理代理

*管理代理*是管理协议的实施, 可与管理站交换有关托管元素的信息。

## 管理站访问提供商 (MSAP)

*管理站访问提供商 (MSAP)* 是一个软件任务, 可为对象提供对管理站的访问权限。

## 管道

*管道*是表示两个设备或端口之间的连接的图标。

## 操作

*操作*是不属于 CA Spectrum 定义用于模型的基本操作集的任意操作。

---

## 警报

警报指示模型存在异常状况。

## 警报重要级别

警报重要级别是 EventDisp 文件中存在的值，用于指示模型的状况。状况表示警报的存在与否和严重度。有效值为 0 到 6，所有值都表示不同颜色的状况。

