

Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

As informações abaixo foram obtidas do fornecedor do dispositivo em questão sem nenhuma verificação independente por parte da Dell e estão sujeitas às [restrições e isenções de responsabilidade](#) descritas a seguir.

- [Introdução](#)
 - [WMI](#)
 - [Principais recursos](#)
 - [Arquivos instalados](#)
 - [Segurança](#)
 - [Espaço de nomes e contexto](#)
 - [Localidades e localização](#)
 - [Relatório de erros](#)
 - [O esquema do núcleo](#)
 - [Esquema do adaptador Ethernet](#)
 - [Esquema de configuração](#)
 - [Esquema de grupos](#)
 - [Esquema de VLAN](#)
 - [Obtenção da configuração atual:](#)
 - [Atualização da configuração](#)
 - [Notificações de eventos](#)
 - [Consultas WQL otimizadas](#)
 - [Diagnósticos](#)
 - [Execução de métodos em IANet_DiagTest](#)
 - [Resumo das classes CIM](#)
 - [Licença de software](#)
 - [Suporte ao cliente](#)
-

As informações contidas neste documento estão sujeitas a alterações sem aviso prévio.

© 2003 Intel Corporation. Todos os direitos reservados.

Marcas comerciais usadas neste texto: *Dell* e o logotipo *DELL* são marcas comerciais da Dell Computer Corporation; *Intel* é marca comercial ou marca comercial registrada da Intel Corporation ou suas subsidiárias nos Estados Unidos e em outros países.

* Outras marcas e nomes comerciais poderão ser usados neste documento para fazer referência a entidades que se declaram proprietárias dessas marcas e nomes ou a seus produtos. A Intel Corporation declara que não tem nenhum interesse de propriedade sobre nenhuma marca registrada ou nome comercial que não sejam os seus próprios.

Restrições e isenções de responsabilidade

As informações contidas neste documento, inclusive todas as instruções, precauções e aprovações de normalização, foram obtidas com o fornecedor e não foram verificadas nem testadas isoladamente pela Dell. A Dell não se responsabiliza por danos resultantes do cumprimento ou descumprimento dessas instruções.

Todas as declarações ou reivindicações relativas às propriedades, capacidades, velocidades ou qualificações da entidade citada neste documento foram feitas pelo fornecedor e não pela Dell. A Dell afirma especificamente que não tem conhecimento da exatidão, integralidade ou comprovação de nenhuma destas declarações. Todas as questões e comentários relativos a tais declarações ou reivindicações devem ser encaminhados ao fornecedor.

Versão inicial: outubro de 2003

Introdução: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

Visão geral

Bem-vindo ao *Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO*. Este documento descreve a visualização externa dos provedores de WMI e CDM dos adaptadores de rede Intel PRO. O provedor WMI (Windows Management Interface) é um bloco de configuração de NCS (Network Configuration Services), que é um meio de implementar e gerenciar todas as tecnologias de rede de estação final Intel utilizando métodos padrão do setor. O provedor Common Diagnostic Model (CDM) do Intel PRO é uma API de interface superior, compatível com os padrões CIM 2.5 e WMI. Na interface inferior, o provedor CDM implementa uma interface cliente nas camadas inferiores da pilha de software do PROSet. Em decorrência disso, são mantidos todos os mecanismos do PROSet para integridade de dados.

Os Provedores WMI e CDM são conjuntos de componentes de software que implementam as classes de redes WMI Intel. Essas classes estão baseadas no esquema CIM versão 2.5 da DMTF (Desktop Management Task Force).

Este documento não repete as informações contidas nos arquivos MOF (Managed Object Format) fornecidos com este produto (por exemplo, detalhes dos significados de atributos individuais podem ser encontrados nas descrições dos atributos MOF).

Este documento descreve como um aplicativo WMI, como o Intel PROSet, utiliza classes para configurar uma rede do sistema e como um aplicativo WMI usa classes para testar uma placa de interface de rede da Intel. Os leitores já devem conhecer as APIs WMI e o SDK WMI (disponíveis em <http://www.microsoft.com/>).

[Voltar ao início](#)

Documentos relacionados

Os documentos a seguir podem ser utilizados com o objetivo de compreender melhor a tecnologia WMI.

- Esquema CIM versão 2.0, 2.2, publicado pela DMTF (Desktop Management Task Force). Disponível em <http://www.dmtf.org>.
- Microsoft Windows Management Interface (e outras informações sobre gerenciabilidade). Disponível em http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/wmi/wmi_start_page.asp.
- Iniciativa do WBEM (Web-Based Enterprise Management) pela DMTF. Disponível em <http://www.dmtf.org/standards/wbem>.
- WMI (Microsoft CIM implementation) SDK. Disponível em <http://msdn.microsoft.com/downloads/>.
- Documento sobre o System Diagnostic Model gerado pela DFTM. Disponível em <http://www.dmtf.org/standards/documents/CIM/DSP0138.pdf>.

 **AVISO:** Este produto contém informações que podem ser utilizadas para realizar ataques e/ou desativar sistema(s) ou rede(s) de computador(res). Deverá ser um pré-requisito para qualquer implementação deste produto um conhecimento completo dos recursos de segurança do sistema operacional da Microsoft e os desenvolvedores e usuários são encorajados a contatarem a Microsoft com relação a quaisquer questões relativas à segurança que possam ter antes de utilizar qualquer implementação deste produto num ambiente de produção.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do índice](#) [Voltar ao início](#)

WMI: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Visão geral](#)

[Esquema CIM \(Common Information Model\)](#)

Visão geral

O WBEM (Web-Based Enterprise Management) é uma iniciativa da DMTF (Desktop Management Task Force) destinada a fornecer gerenciadores de sistemas empresariais com um método padronizado e barato para gerenciamento de estações finais. A iniciativa do WBEM inclui uma variedade de tarefas, que vão desde a simples configuração de uma estação de trabalho a um gerenciamento em larga escala da empresa através de várias plataformas. O fundamental para a iniciativa é o CIM (Common Information Model), um modelo amplo de dados que representa os objetos existentes em ambientes de gerenciamento típicos e a linguagem MOF (Managed Object Format) para definir e armazenar os dados modelados.

O WMI (Windows Management Instrumentation) é uma implementação da iniciativa do WBEM para plataformas Microsoft* Windows*.

O WMI é composto por três componentes principais:

- Core — Esses componentes fazem parte do sistema operacional. Eles são necessários para que um aplicativo habilitado com WMI funcione e devem estar instalados para ser possível a utilização do SDK.
- SDK — O SDK contém ferramentas para navegar pelo esquema WMI, estender o esquema, criar provedores, registrar e utilizar os eventos WMI. Ele também fornece documentação útil para o desenvolvimento de aplicativos que utilizarão o WMI. O SDK é instalado como parte do processo de instalação do Microsoft Platform SDK e é suportado no Windows NT4 SP4 ou SP5, Windows 2000, Windows Me, Windows XP e Windows Server 2003.
- Tools — O Microsoft WMI Tools fornece aos desenvolvedores as ferramentas necessárias para construir toda uma nova geração de aplicativos e soluções de gerenciamento.. Ele está repleto de documentos e ferramentas para guiá-lo através do processo de acessar os dados de gerenciamento a partir do WMI.

A arquitetura do WMI consiste nos seguintes componentes:

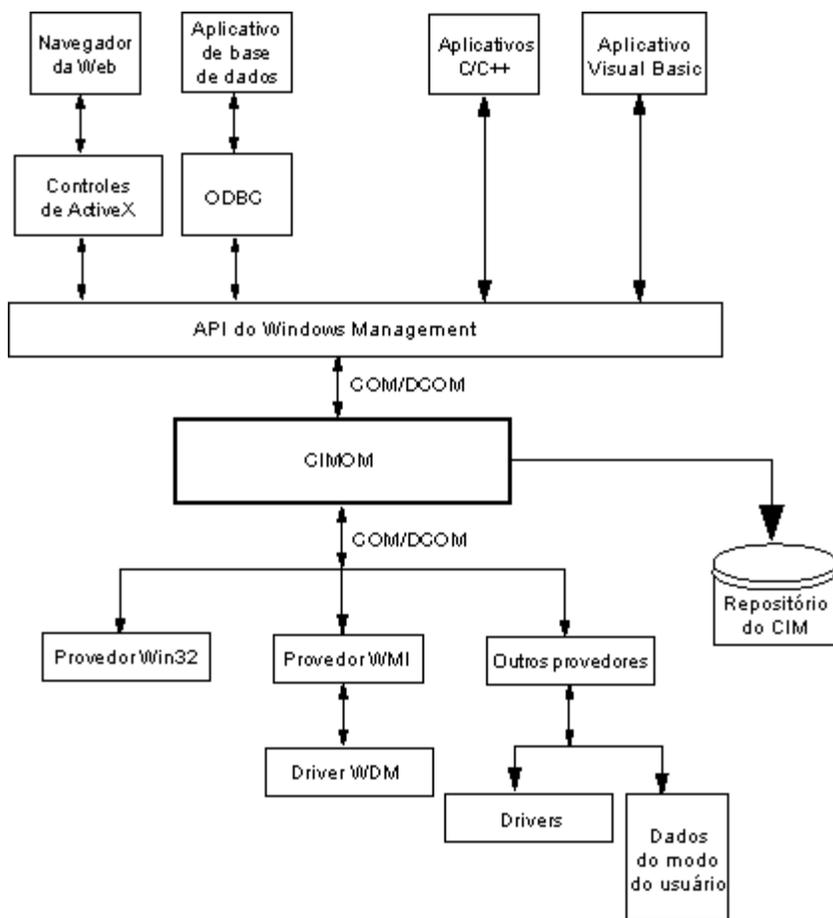
- Aplicativos de gerenciamento
- Objetos gerenciados
- Provedores
- Infra-estrutura de gerenciamento (consiste no Windows Management e no repositório do Windows Management)
- API do Windows Management (que utiliza o COM/DCOM para ativar provedores e aplicativos de gerenciamento para que se comuniquem com a infra-estrutura do Windows Management).

Os aplicativos gerenciados processam ou exibem os dados a partir de objetos gerenciados, os quais componentes lógicos ou físicos da empresa. Esses componentes são modelados por meio do CIM e são acessados pelos aplicativos através do Windows Management. Os provedores utilizam o API do Windows Management para suprirem o Windows Management com dados dos objetos gerenciados e tratarem as solicitações a partir dos aplicativos e gerarem a notificação de eventos.

A infra-estrutura gerenciada consiste no Windows Management (para tratar as comunicações entre os aplicativos gerenciados e os provedores) e no repositório do Windows Management (para armazenar dados). O repositório do Windows Management retém os dados estáticos de gerenciamento. Os dados dinâmicos são gerados apenas mediante solicitação dos provedores. Os dados são colocados no repositório usando o compilador da linguagem MOF ou o API do Windows Management.

Os aplicativos e os provedores comunicam-se através do Windows Management ao utilizarem o API do Windows Management, que fornece serviços como o de notificação de eventos e o de processamento de consultas.

O diagrama a seguir mostra o inter-relacionamento dos componentes da arquitetura WMI:



[Voltar ao início](#)

Esquema CIM (Common Information Model)

O CIM (Common Information Model) apresenta uma visão consistente e unificada de todos os tipos de objetos lógicos e físicos em um ambiente gerenciado. Os objetos gerenciados são representados usando-se construções orientadas para objetos, tais como as classes. As classes incluem propriedades que descrevem os dados e os métodos que descrevem o comportamento. O CIM é desenvolvido pelo DMTF para ser independente de sistema operacional e de plataforma. A tecnologia do WBEM inclui uma extensão do CIM para as plataformas do sistema operacional Microsoft Windows. Para obter mais informações, consulte o esquema CIM da DMTF no site da Web da DMTF.

O CIM define três níveis de classes:

- Classes representando objetos gerenciados que se aplicam a todas as áreas de gerenciamento. Essas classes fornecem um vocabulário básico para analisar e descrever os sistemas gerenciados e fazem parte do que é conhecido como o modelo de núcleo.
- Classes representando os objetos gerenciados que se aplicam a uma área específica de gerenciamento, mas que são independentes de uma implementação ou tecnologia específica. Essas classes fazem parte do que é conhecido como o modelo comum.
- Classes representando os objetos gerenciados que são adições ao modelo comum específicas da tecnologia. Essas classes aplicam-se a plataformas específicas, tais como ambiente UNIX ou Microsoft Win32, conhecidas como modelo estendido.

Todas as classes podem estar relacionadas pela sua descendência, onde uma classe filho inclui dados e métodos de sua classe pai. Os relacionamentos de descendência não são tipicamente visíveis para o aplicativo de gerenciamento utilizando-os e os aplicativos não necessitam conhecer a hierarquia de descendência. As hierarquias de classes podem ser obtidas ao utilizar aplicativos que estão incluídos no WMI Tools (para obter mais informações consulte WMI Tools em <http://www.microsoft.com>).

O Windows Management também suporta a associação de classes. As classes de associação vinculam duas classes diferentes para modelar um relacionamento definido pelo usuário e são visíveis para os aplicativos de gerenciamento. O Windows

Management define as classes de associação para suportar classes de sistemas. Desenvolvedores de terceiros podem também definir classes de associação para seu ambiente de gerenciamento.

O WBEM suporta o conceito de esquemas para agrupar as classes e .instâncias que são utilizadas dentro de um ambiente de gerenciamento específico. O Platform SDK inclui dois esquemas: o esquema CIM e o esquema Microsoft Win32. O esquema CIM contém as definições de classes para os primeiros dois níveis do CIM. Essas classes representam objetos gerenciados que fazem parte de cada ambiente gerenciado, independente da plataforma. O esquema Win32 contém as definições de classes para objetos gerenciados que fazem parte de um ambiente dWin32 típico.

Para informações adicionais sobre o CIM, visite <http://www.dmtf.org>.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Principais recursos: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Recursos do provedor WMI NCS](#)

[Recursos do provedor CDM](#)

Recursos do provedor WMI NCS

Os principais recursos do provedor WMI são os seguintes:

Recursos do adaptador

- Enumerar todos os adaptadores físicos suportados pelo Intel® PROSet.
- Enumerar as configurações de um adaptador instalado.
- Adicionar/remover/atualizar as configurações para um adaptador instalado.
- Obter as informações do dispositivo físico de um adaptador.
- Obter as informações do dispositivo do slot do sistema de um adaptador.
- Obter as configurações do protocolo IPv4 para um adaptador.
- Atualizar e alterar o boot agent e as configurações associadas de um adaptador.
- Desinstalar um adaptador.

Recursos de grupo

- Enumerar os grupos suportados pelo Intel PROSet.
- Criar/remover um grupo de adaptadores.
- Adicionar/remover/atualizar as configurações do grupo.
- Adicionar/remover adaptadores membros de um grupo.
- Obter as configurações do protocolo IPv4 para um grupo.

Recursos de VLAN

- Enumerar as LANs virtuais em um adaptador ou grupo.
- Criar/remover LANs virtuais em um adaptador físico ou em um grupo de adaptadores.
- Adicionar/remover/atualizar as configurações da VLAN.
- Obter as configurações do protocolo IPv4 para um grupo.

Recursos de notificação de eventos

- Permitir ao cliente fazer o registro.
 - Eventos de status do adaptador.
 - Eventos de configuração do adaptador.
 - Eventos de sessões.
 - Eventos de status do grupo.
 - Eventos de configuração do grupo.
 - Eventos de configuração da VLAN.

[Voltar ao início](#)

Recursos do provedor CDM

Os principais recursos do provedor CDM são os seguintes:

- Executar testes, parar testes e eliminar resultados de testes sem dependência com relação ao tipo de teste de diagnóstico.
- A utilização de classe de configuração genérica deverá permitir o controle de testes de modos não antecipados pelo próprio

software CDM.

- O provedor CDM é utilizado apenas para adaptadores.
 - A utilização da classe de resultado genérica libera mensagens específicas de resultados do código do provedor CDM.
 - Entradas de registro controlam a execução do provedor.
 - Os resultados dos testes são gravados no arquivo de registro de resultados
-

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Arquivos instalados: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Arquivos WMI](#)

[Arquivos do provedor CDM](#)

Arquivos WMI

Executáveis

Os executáveis do provedor WMI são os seguintes:

- **NcsWmiCo.exe** — O provedor do núcleo. Implementa o IANet_NetService e as classes de eventos do núcleo.
- **NcsWmiCo.exe** — O provedor de instâncias e métodos. Implementa o esquema do adaptador Ethernet, o esquema de agrupamento, o esquema de configuração e o esquema de VLAN.
- **NcsWmiEv.exe** — O provedor de eventos. Implementa os eventos do adaptador, do grupo e de VLAN.

Arquivos MOF

Há arquivos MOF separados para dados neutros ao idioma e específicos ao idioma. Além disso, há arquivos MOF separados para o espaço de nomes do IntelNCS e CIMV2. Para obter mais detalhes, consulte [Localidades e localizações](#) e [Relatório de erros](#).

Os arquivos MOF para o espaço de nomes do IntelNCS são os seguintes:

- **NcsCmLn.mof** — Classes básicas do CIM nas quais dependem as classes do NCS.
- **NcsCmEnu.mfl** — Versão em inglês dos EUA das classes básicas do CIM.
- **NcsCoLn.mof** — Classes do núcleo implementadas pelo provedor do núcleo.
- **NcsCmEnu.mfl** — Modificações de texto em inglês dos EUA das classes do núcleo.
- **NcslaLn.mof** — Classes para adaptadores, grupos e VLANs IEEE 802.3.
- **NcslaEnu.mfl** — Modificações de texto em inglês dos EUA das classes do núcleo 802.3.

Os arquivos MOF para o espaço de nomes do CIMV2 são os seguintes:

- **C2CmLn.mof** — Classes básicas do CIM nas quais dependem as classes do NCS.
- **C2CmEnu.mfl** — Versão em inglês dos EUA das classes básicas do CIM.
- **C2CoLn.mof** — Classes do núcleo implementadas pelo provedor do núcleo.
- **C2CoEnu.mfl** — Modificações de texto em inglês dos EUA das classes do núcleo.
- **C2laLn.mof** — Classes para adaptadores, grupo e VLANs IEEE 802.3.
- **C2laEnu.mfl** — Modificações de texto em inglês dos EUA das classes do núcleo 802.3.

Arquivos de recursos

Os arquivos de recursos do provedor WMI são os seguintes:

- **ENU_8023.dll** — Recurso inglês EUA 8023.
- **ENU_NWRC.dll** — Recursos do WMI em inglês EUA para provedores do núcleo.
- **ENU_NWR.dll** — Recursos do WMI em inglês EUA para provedores de 8023.

Os outros arquivos de recursos localizados podem ser carregados conforme a necessidade. O padrão geral para os nomes de DLLs de recursos localizados é "_mwr.dll" que é o código do idioma para a localização (e.g. FRA para o francês padrão).

[Voltar ao início](#)

Arquivos do provedor CDM

Executáveis

Os executáveis do provedor CDM são os seguintes:

- **Ncsdiag.exe** é o arquivo executável para os diagnósticos do CDM. Ele segue a especificação da interface WMI Microsoft* e é acessado como um servidor COM fora do processo.
- Outros executáveis a partir da pilha de software do Intel® PROSet

Arquivos MOF

Os arquivos **.mof** master não são enviados com o produto, mas são compilados em componentes dependentes do idioma e neutros ao idioma, de acordo com o modelo de globalização do Microsoft* Windows* Management Instrumentation. Para mais informações, consulte o Microsoft* WMI SDK (componente do Platform SDK) capítulo relativo à localização do WMI. Preste atenção à seção *Compilação de arquivos MOF localizados*.

Excluir um arquivo **.mof** (DNcsCdmN.mof) exclui as definições de classes derivadas da Intel, mas não exclui as classes definidas da DMTF, uma vez que isso poderá prejudicar outros aplicativos existentes.

A utilização típica dessa implementação CDM está baseada no espaço de nomes do CIMV2. Os arquivos MOF para um espaço de nomes do IntelNCS são os seguintes:

Nome do arquivo	Tipo de idioma	Descrição
Cdla.mof	Master	Definições de classes para implementação do CDM Intel
CdlaLn.mof	Neutro ao idioma	Definições de classes para implementação do CDM Intel
CdlaEnu.mfl	Dependente do idioma inglês	Definições de classes para implementação do CDM Intel
CdCm.mof	Master	Definições de classes CDM do superconjunto do núcleo
CdCmLn.mof	Neutro ao idioma	Definições de classes CDM do superconjunto do núcleo
CdCmEnu.mfl	Dependente do idioma inglês	Extensões do idioma para definições de classes CDM do superconjunto do núcleo
DNcsCdmN.mof	Não se aplica	Exclui as classes do CDM Intel

Os arquivos MOF para um espaço de nomes do CIMV2 são os seguintes:

Nome do arquivo	Tipo de idioma	Descrição
C2lcd.mof	Master	Definições de classes para implementação do CDM Intel
C2lcdLn.mof	Neutro ao idioma	Definições de classes para implementação do CDM Intel
C2lcdEnu.mfl	Dependente do idioma inglês	Definições de classes para implementação do CDM Intel
C2Cd.mof	Master	Definições de classes CDM do superconjunto do núcleo
C2CdLn.mof	Neutro ao idioma	Definições de classes CDM do superconjunto do núcleo
C2CdEnu.mfl	Dependente do idioma	Extensões do idioma para definições de classes CDM do superconjunto do

	inglês	núcleo
DNcsCdm2.mof	Não se aplica	Exclui as classes do CDM Intel

Nota: A localização necessita da adição do arquivo **.mof** correto dependente do idioma.

Arquivos de recursos

Os arquivos de recursos do provedor CDM são os seguintes:

- **ENU_Diag.dll** - Recursos do WMI em inglês EUA para provedor de diagnóstico.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

[Voltar à página do índice](#)

Segurança: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

Os provedores WMI e CDM utilizam a personificação do cliente para gerenciar a segurança. Cada chamada aos provedores será feita no contexto de segurança do próprio cliente, o qual é enviado para as camadas inferiores. Uma ou todas as operações podem falhar caso você não tenha direitos administrativos sobre a máquina de destino.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Espaço de nomes e contexto: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

As classes CIM residem em um espaço de nomes. O espaço de nomes padrão da Microsoft* é chamado **root/cimv2** e está baseado no CIM v2.2 ou **root/default**. As classes de provedores WMI e CDM podem ser adicionadas a esse espaço de nomes. Os provedores são baseados no CIM v2.5. Devido a isso e às diferenças utilizadas nas chaves dos objetos, as classes dos provedores estão localizadas em um espaço de nomes separado, **root/IntelINCS**.

Contexto WBEM

Objetos de contexto fornecem informações adicionais para os provedores que não podem ser passados como um parâmetro para um método de API WMI. Para registrar qualificadores de contexto, utilize o **IWbemContext** para registrar os qualificadores de contexto. O ponteiro da interface para o objeto de contexto é passado como o último parâmetro de um método **IWbemServices**.

A tabela a seguir contém os qualificadores de contexto (valores com nomes) utilizado pelos provedores. A maioria dos qualificadores, como o SessionHandle, é utilizada apenas em conjunto com áreas funcionais específicas dos provedores, ao passo que LocaleID, MachineName e ApplicationName podem ser definidos para todas as chamadas de **IWbemServices**.

Se nenhum contexto for passado para os provedores, eles utilizarão o LocaleID passado na chamada de **Initialize** para os provedores. Qualquer leitura feita com um contexto lerá a configuração atual até ser realizada uma operação de gravação. Leituras subsequentes mostrarão o sistema com ele seria após a gravação ocorrer com sucesso. Um contexto nulo pode ser utilizado para leituras.

Qualificador de contexto	Tipo de variável	Descrição
SessionHandle	VT_BSTR	Identifica a cópia das classes de rede IANet do aplicativo. O aplicativo não pode fazer quaisquer alterações às classes ou a seus atributos sem primeiro estabelecer um identificador de sessão. Consulte a seção na classe IANet_NetService para ver como estabelecer e utilizar um identificador de sessão. Este qualificador não é necessário se o aplicativo irá apenas ler os dados das classes. O identificador da sessão permite ao software NCS gerenciar simultaneamente vários acessos para a configuração sem um usuário bloquear todos os outros. Cada sessão possui um cache separado para armazenar quaisquer mudanças que tenham sido feitas. Se houver vários usuários realizando mudanças simultaneamente, então, o primeiro usuário a aplicar suas mudanças terá sucesso. Todos os outros caches de usuários serão invalidados.
LocaleID	VT_BSTR	ID da Microsoft para uma localidade. Isso é necessário se o aplicativo necessitar seqüências de texto localizado a partir dos provedores. Todas as mensagens e avisos de erro serão em inglês, a menos que o LocaleID solicitado seja utilizado.
ApplicationName	VT_BSTR	O nome do aplicativo que efetuou a chamada. Esse procedimento é obrigatório para registros.
MachineName	VT_BSTR	O nome da máquina que está se conectando aos provedores. Esse procedimento é obrigatório para registros.
PreCheck	VT_BOOL	Esse valor booleano é utilizado para informar aos provedores que o cliente está tentando verificar se uma operação é permitida antes de efetuar realmente a operação. Por exemplo, adicionar um adaptador a um grupo. Valores: <ul style="list-style-type: none">• TRUE = O provedor não executará a operação, mas retornará um código de erro e um status estendido se a operação não for permitida.• FALSE = O provedor executará a operação.

		Caso esse qualificador esteja faltando, ele terá o mesmo efeito como se o atributo fosse FALSE.
WarningErrorCode	VT_I4	Algumas operações podem necessitar que sejam enviados avisos para o usuário (por exemplo, adicionar um adaptador ao grupo pode exigir que o grupo seja recarregado em algumas circunstâncias). O WMI não fornece um mecanismo para isso. Se esse qualificador estiver presente e for diferente de zero, o provedor retornará E_FAIL se a operação for bem sucedida, mas houve um aviso associado. O cliente deverá utilizar o status estendido para obter o texto do aviso.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Localidades e localização: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Arquivos MOF localizados](#)

[Dados de atributos localizados](#)

Há dois aspectos para a localização de provedores WMI e CDM — arquivos MOF localizados e dados de atributos localizados.

Arquivos MOF localizados

Todos os arquivos MOF utilizados pelos provedores são localizados de acordo com o modelo de globalização do Microsoft Windows* Management Instrumentation (WMI). Para isso, cada definição de classe é separada conforme a seguir:

- Uma versão neutra ao idioma que contenha apenas a definição de classes básicas no arquivo **.mof**.
- Uma versão específica ao idioma que contém informações localizadas, como, por exemplo, descrições de propriedade que são específicas a uma localidade no arquivo **.mfl** correspondente.

Idiomas suportados

Chinês (Taiwan)
Chinês (RPC)
Dinamarquês
Holandês (Países baixos)
Inglês (Estados Unidos)
Finlandês
Francês (França)
Alemão (Alemanha)
Italiano (Itália)
Japonês
Norueguês (Bokmal)
Português (Brasil)
Espanhol (Espanha - Moderno)
Sueco

Armazenamento de classes

As definições de classes específicas ao idioma são armazenadas em um sub-espço de nomes filho dentro do espaço de nomes que contém a definição da classe básica neutra ao idioma. Por exemplo, para o provedor WMI e CDM, existirá um espaço de nomes filho **ms_409** dentro do espaço de nomes **root/Intelncs** para a localidade do inglês. Similarmente, lá existe um sub-espço de nomes filho para cada idioma suportado dentro do espaço de nomes **root/Intelncs**.

Suporte a MOV localizado no espaço de nomes cimv2

Para os espaços de nomes **root/cimv2**, as classes dos provedores (isto é, classes IANet_) são derivadas das classes básicas acrescentadas a esse espaço de nomes pelo WMI. Dentro do espaço de nomes **root/cimv2** já existe um sub-espço de nomes com definições de classes específicas ao idioma para classes básicas. As definições de classes específicas ao idioma IA_Net serão adicionadas a esse espaço de nomes filho. Devido a essa discrepância nas classes básicas, a localização do MOF é efetuada apenas nas localidades padrão do sistema.

Suporte de tempo de execução

Um aplicativo WMI, para receber os dados localizados, pode especificar a localidade utilizando um parâmetro strLocale em chamadas **SWbemLocator::ConnectServer** e **IWbemLocator::ConnectServer**. Se a localidade não for especificada, será utilizada a localidade padrão para esse sistema. (por exemplo, MS_409 para inglês dos EUA). Essa localidade é utilizada para selecionar o espaço de nomes correto quando for acrescentar strings em inglês.

Além disso, o **IWbemServices::GetObject**, **SwbemServices.GetObject**, **IWbemServices:: ExecQuery** e **SWbemServices.ExecQuery** devem especificar a marcação WBEM_FLAG_USE_AMENDED_QUALIFIERS para solicitar dados localizados, juntamente com a definição básica. Isso é necessário em todas as funções que geram valores que podem ser exibidos por meio de mapas de valores, exibir descrições ou outros qualificadores de correção a partir dos arquivos MOF.

[Voltar ao início](#)

Dados de atributos localizados

Para obterem os dados de atributos localizados (como, por exemplo, mensagens de erro), os provedores necessitam saber a localidade do chamado para cada chamada. Para que isso funcione corretamente, o cliente deve adicionar a localidade ao objeto de contexto, que é passada para cada chamada (consulte [Espaço de nomes e contexto](#) em contexto do WBEM). Se os provedores necessitarem retornar a uma string localizável, em seguida, tentarão carregar uma DLL de recurso que seja adequada à localidade do cliente. Se não houver uma DLL de recurso adequada, os provedores retornarão strings em inglês dos EUA.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Relatório de erros: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Visão geral](#)

[Códigos de erro](#)

Visão geral

Esta seção sobre o IANet_ExtendedStatus descreve como tratar os erros gerados pelos provedores WMI e CDM. Como e quando um objeto de erro é retornado depende se uma chamada é síncrona, semi-síncrona ou assíncrona. Quando ocorrer um erro, na maioria dos casos, o HRESULT é definido como WBEM_E_FAILED. Nesse ponto, contudo, não é conhecido se o WMI ou os provedores geraram o erro.

Para obter o objeto de erro para chamadas síncronas, utilize o GetErrorInfo() para obter o objeto IErrorInfo. Utilize o QueryInterface() para obter o IWbemClassObject, que contém as informações de erro.

Para obter um objeto de erro para chamadas assíncronas, o IWbemClassObject é retornado como o último item na última chamada do SetStatus(). Após obter a instância do objeto de erro, é possível verificar a propriedade de __Class para determinar a origem do erro. O WMI cria uma instância de __ExtendedStatus e os provedores criam uma instância de of IANet_ExtendedStatus para erros relatando as classes de IANet_. O IANet_ExtendedStatus é derivado de __ExtendedStatus e contém os qualificadores de objetos de erro a seguir:

- Description – Descrição do erro personalizado para a localidade atual.
- File – Arquivo de códigos onde os erros foram gerados.
- Line – Número da linha do arquivo de códigos com o erro.
- ParameterInfo – Classe ou atributo que está sendo utilizado quando ocorreu o erro.
- Operation – Operação sendo tentada quando ocorreu o erro.
- ProviderName – Nome do provedor que causou o erro..
- StatusCode – Código retornado a partir da chamada interna que falhou.
- SessionHandle – Identificador da sessão utilizado para a operação.
- RuleFailureReasons – Motivo para a falha da operação. Uma operação pode falhar devido à falha de uma regra técnica. (por exemplo, você deve ter um adaptador de gerenciamento em certos grupos).

[Voltar ao início](#)

Códigos de erro

Os provedores fornecem uma descrição personalizada para a localidade para todos os códigos de erro. Os códigos de erro estão no formato de HRESULT com a severidade definida para 1(um) e o recurso definido para ITF. Um aplicativo pode utilizar os códigos a seguir como base para uma ação de recuperação:

- 0x80040901 – "WMI: falha de colocação de propriedade"
- 0x80040902 – "WMI: sem objeto de classe"
- 0x80040903 – "WMI: falha ao criar classe"
- 0x80040904 – "WMI: falha ao gerar instância de classe"
- 0x80040905 – "WMI: falha ao criar matriz de segurança"
- 0x80040906 – "WMI: falha ao posicionar segurança"
- 0x80040907 – "WMI: falha ao retornar objeto para a WMI"
- 0x80040908 – "WMI: falha de obtenção de propriedade"
- 0x80040909 – "WMI: tipo inesperado ao obter propriedade"
- 0x8004090A – "WMI: classe não implementada por este provedor"
- 0x8004090B – "WMI: impossível analisar instrução WQL"
- 0x8004090C – "WMI: os provedores suportam apenas a WQL"

- 0x8004090D – "WMI: o parâmetro do contexto é de um tipo incorreto"
- 0x8004090E – "WMI: erro ao formatar o registro de depuração"
- 0x8004090F – "WMI: caminho incorreto do objeto"
- 0x80040910 – "WMI: falha ao atualizar configuração"
- 0x80040911 – "WMI: foi passado um parâmetro nulo para o método"
- 0x80040912 – "Valor muito baixo da configuração."
- 0x80040913 – "Valor muito baixo da configuração."
- 0x80040914 – "Configuração não definida na etapa"
- 0x80040915 – "A configuração da string é longa demais"
- 0x80040916 – "A configuração não é um dos valores permitidos"
- 0x80040917 – "WMI: qualificador não encontrado"
- 0x80040918 – "WMI: qualificador definido não encontrado"
- 0x80040919 – "WMI: falha de acesso da matriz de segurança"
- 0x8004091A – "WMI: exceção não resolvida"
- 0x8004091B – "WMI: operação não suportada para esta classe"
- 0x8004091C – "WMI: classe de evento inesperada"
- 0x8004091D – "WMI: dados de evento incorretos"
- 0x8004091E – "WMI: operação bem-sucedida com avisos"
- 0x8004081F – "WMI: serviço NCS interrompido."

- 0x80040801 – "EAL: exceção interna"
- 0x80040802 – "EAL: falha geral"
- 0x80040803 – "EAL: não inicializado"
- 0x80040804 – "EAL: falha ao inicializar."
- 0x80040805 – "EAL: limites da sessão excedidos"
- 0x80040806 – "EAL: memória esgotada"
- 0x80040807 – "EAL: erro de sintaxe da regra"
- 0x80040808 – "EAL: fim inesperado da lista"
- 0x80040809 – "EAL: erro de link da regra"
- 0x8004080A – "EAL: falha na criação do dispositivo"
- 0x8004080B – "EAL: serviços de mídia não encontrado"
- 0x8004080C – "EAL: serviço de dispositivo não encontrado"
- 0x8004080D – "EAL: módulo de barramento PCI não encontrado"
- 0x8004080E – "EAL: o adaptador é membro de um grupo"
- 0x8004080F – "EAL: erro na criação do ponto de acesso da regra"
- 0x80040810 – "EAL: erro de chave do Registro"
- 0x80040811 – "EAL: erro de caminho do arquivo XML do Registro"
- 0x80040812 – "EAL: classe de evento desconhecida"
- 0x80040813 – "EAL: id do módulo desconhecida"
- 0x80040814 – "EAL: serviço de regra não encontrado"
- 0x80040815 – "EAL: ponteiro de entrada NULO"
- 0x80040816 – "EAL: erro gramatical da regra"
- 0x80040817 – "EAL: falha da regra"
- 0x80040818 – "EAL: configuração já agrupada"

- 0x80040220 – "Camada de sincronização: falha na remoção da equipe."
- 0x80040221 – "Camada de sincronização: falha na criação de Vlan."
- 0x80040222 – "Camada de sincronização: falha na remoção de Vlan."
- 0x80040223 – "Camada de sincronização: falha na remoção do adaptador."
- 0x80040224 – "Camada de sincronização: falha na modificação/criação/remoção"
- 0x80040225 – "Camada de sincronização: falha na modificação/remoção de parâmetros."
- 0x80040226 – "Camada de sincronização: subsistema NetConfig bloqueado." "
- 0x80040227 – "Camada de sincronização: atualização do sistema em andamento. Favor tentar mais tarde."
- 0x80040228 – "Camada de sincronização: adaptador bloqueado."
- 0x80040229 – "Camada de sincronização: falha de leitura da flash."
- 0x8004022A – "Camada de sincronização:

- 0x80040210 – "Camada de sincronização: deslocamento inválido de evento."
- 0x80040211 – "Camada de sincronização: entrada inválida."
- 0x80040212 – "Camada de sincronização: chave inválida."

- 0x80040213 – "Camada de sincronização: o adaptador não é membro de grupo."
- 0x80040214 – "Camada de sincronização: driver não carregado."
- 0x80040215 – "Camada de sincronização: falha de representação do cliente"
- 0x80040216 – "Camada de sincronização: exceção detectada"
- 0x80040217 – "Camada de sincronização: sessão não bloqueada."
- 0x80040218 – "Camada de sincronização: camada de acesso a hardware não disponível."
- 0x80040219 – "Camada de sincronização: flash não disponível."
- 0x8004021A – "Camada de sincronização: diagnósticos não suportados."
- 0x8004021B – "Camada de sincronização: o teste de diagnóstico não está em execução."
- 0x8004021C – "Camada de sincronização: atualização do Boot Agent não disponível."
- 0x8004021D – "Camada de sincronização: Boot Agent corrompido."
- 0x8004021E – "Camada de sincronização: falha de leitura da flash."
- 0x8004021F – "Camada de sincronização: falha na criação do grupo."
- 0x80040201 – "Camada de sincronização: falha de inicialização."
- 0x80040202 – "Camada de sincronização: referência temporária inválida da inicialização."
- 0x80040203 – "Camada de sincronização: a referência temporária da sessão já existe."
- 0x80040204 – "Camada de sincronização: referência temporária inválida da sessão."
- 0x80040205 – "Camada de sincronização: o número máximo de sessões foi atingido."
- 0x80040206 – "Camada de sincronização: a referência temporária de bloqueio da sessão já existe."
- 0x80040207 – "Camada de sincronização: referência temporária inválida de bloqueio da sessão."
- 0x80040208 – "Camada de sincronização: sessão já bloqueada."
- 0x80040209 – "Camada de sincronização: id do módulo de serviço da mídia inválida."
- 0x8004020A – "Camada de sincronização: id do módulo de serviço avançado inválida."
- 0x8004020B – "Camada de sincronização: id do módulo de serviço do dispositivo inválida."
- 0x8004020C – "Camada de sincronização: id do tipo de componente inválida."
- 0x8004020D – "Camada de sincronização: id do módulo de interface do barramento inválida."
- 0x8004020E – "Camada de sincronização: referência temporária de janela de recepção inválida."
- 0x8004020F – "Camada de sincronização: Id de evento inválida"

- 0x80040401 – "PCI HAM: endereço de mapa de memória inválido."
- 0x80040402 – "PCI HAM: falha no carregamento do driver de configuração."
- 0x80040403 – "PCI HAM: incompatibilidade de versões do driver de configuração."
- 0x80040404 – "PCI HAM: slot de dispositivo não encontrado."
- 0x80040405 – "PCI HAM: falha no carregamento do driver de diagnóstico."
- 0x80040406 – "PCI HAM: incompatibilidade de versões do driver de diagnóstico."
- 0x80040407 – "PCI HAM: falha de inicialização do driver de diagnóstico."
- 0x80040408 – "PCI HAM: diagnóstico não inicializado."
- 0x80040409 – "PCI HAM: diagnóstico já inicializado."
- 0x8004040A – "PCI HAM: teste de diagnóstico em execução."
- 0x8004040B – "PCI HAM: o teste de diagnóstico não está em execução."
- 0x8004040C – "PCI HAM: teste de diagnóstico encerrado."
- 0x8004040D – "PCI HAM: número do teste de diagnóstico inválido."
- 0x8004040E – "PCI HAM: ausência de hardware de diagnóstico."
- 0x8004040F – "PCI HAM: falha de inicialização de envio/recebimento de diagnóstico."

- 0x80040511 – "Serviços de mídia: falha na chamada de E/S do NDIS."
- 0x80040512 – "Serviços de mídia: miniporta não carregada."
- 0x8004051B – "Serviços de mídia: referência temporária inválida do dispositivo."
- 0x8004051C – "Serviços de mídia: referência temporária inválida do adaptador."
- 0x8004051D – "Serviços de mídia: referência temporária inválida do grupo."
- 0x8004051E – "Serviços de mídia: referência temporária inválida de VLAN."
- 0x8004051F – "Serviços de mídia: dispositivo ausente"
- 0x80040520 – "Serviços de mídia: tipo de configuração inválida."
- 0x80040521 – "Serviços de mídia: objeto inválido desconhecido"
- 0x80040522 – "Serviços de mídia: referência temporária inválida da configuração."
- 0x80040523 – "Serviços de mídia: modo de grupo inválido."
- 0x80040525 – "Serviços de mídia: configuração já existente."

- 0x80042001 – "RAP: já inicializado."
- 0x80042002 – "RAP: arquivo XML inválido"

- 0x80042003 – "RAP: erro no carregamento do XML."
 - 0x80042004 – "RAP: não inicializado."
 - 0x80042005 – "RAP: regra não extraída anteriormente."
 - 0x80042006 – "RAP: erro na contagem de condições."
 - 0x80042007 – "RAP: erro de aplicação de resultados."
 - 0x80042008 – "RAP: regra inválida."
 - 0x80042009 – "RAP: nó não encontrado"
 - 0x8004200A – "RAP: erro de nó único."
 - 0x8004200B – "RAP: nenhuma regra de ação."
 - 0x8004200C – "RAP: condição zero."
 - 0x8004200D – "RAP: ação zero."
 - 0x8004200E – "RAP: erro na decodificação do XML."
-

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

O esquema do núcleo: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Visão geral](#)

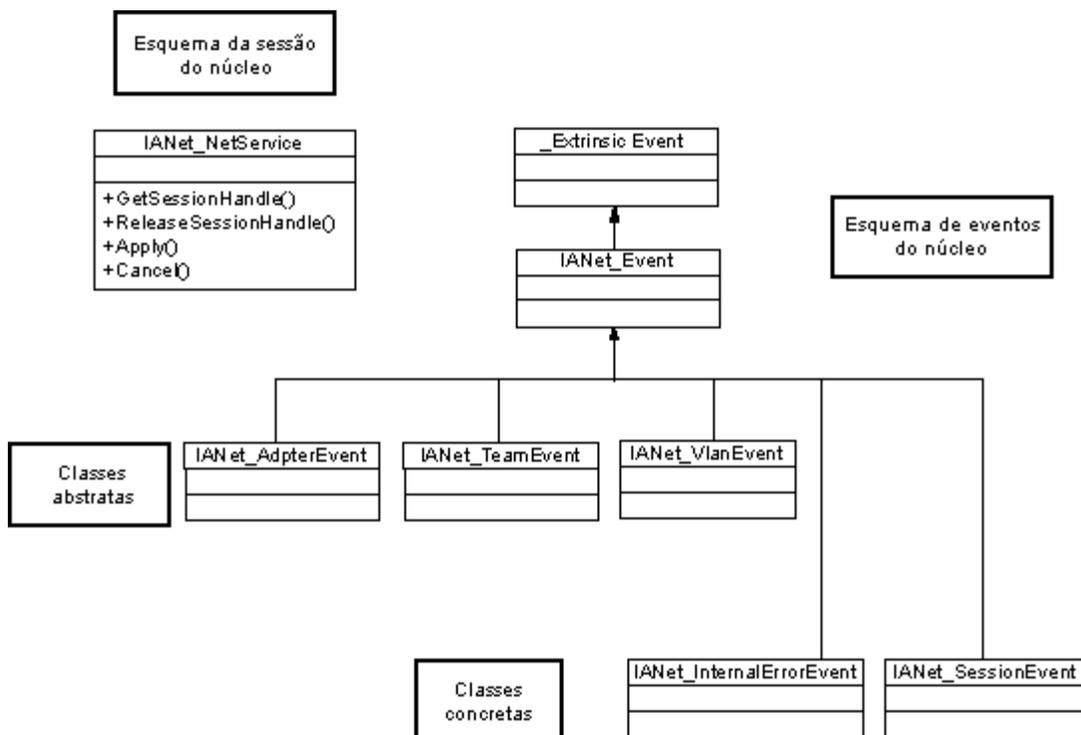
[IANet_NetService](#)

[Eventos do núcleo](#)

[Casos de utilização](#)

Visão geral

O esquema do núcleo consiste na classe IANet_NetService e nas classes de eventos do núcleo.



[Voltar ao início](#)

IANet_NetService

Objetivo

A classe IANet_NetService é o objeto root do esquema IANet_. Essa classe permite ao cliente acessar as sessões que são necessárias para executar as configurações.

Instâncias

Há uma instância desse objeto. O cliente não deverá se basear na chave utilizada para esta classe. Ao invés, o cliente deverá obter a instância da classe ao enumerar todas as instâncias de IANet_NetService.

Criação de instâncias

Não é possível criar instâncias de IANet_NetService.

Remoção de instâncias

Não é possível excluir a instância de `INet_NetService`.

Modificação de propriedades

Não há propriedades modificáveis pelo usuário dessa classe.

Atributos suportados

Essa classe implementa dois atributos:

- `Version` — Contém a versão atual do provedor de núcleo.
- `InstallDate` — Contém a data que os provedores foram instalados.

Métodos

Os métodos a seguir podem ser utilizados para gerenciar sessões:

- **`void GetSessionHandle([OUT] string SessionHandle, [out] uint32 ActiveSessions)`** — Utilizado para definir um identificador da sessão que deverá ser colocado no objeto do contexto no qualificador `SessionHandle`. `ActiveSessions` retorna o número de sessões ativas para este sistema. Isso permite aos clientes avisarem que outros podem estar modificando as configurações da rede.
- **`void Apply([IN] string sSessionHandle, [OUT] uint32 FollowupAction)`**; — Aplica-se a mudanças feitas com um identificador da sessão específico. O argumento `uint32` retornado é utilizado pelos provedores WMI e CDM para informar ao aplicativo que o servidor deve ser reinicializado antes que as mudanças entrem em vigor. Isso pode ser realizado ao chamar o método **`Reboot`** no `Win32_OperatingSystem` da classe.

Valores:

1 = É necessário reiniciar o sistema

0 = Não é necessário reiniciar

- **`void ReleaseSessionHandle ([IN] string SessionHandle)`** — Libera um identificador da sessão após ele ter sido utilizado. Quaisquer mudanças feitas com essa sessão serão perdidas. O identificador da sessão não será mais válido após essa chamada e não pode mais ser utilizado.
- **`void Cancel([IN] string SessionHandle)`**; — Cancela uma sessão. O cache interno será liberado e quaisquer dados lidos após essa chamada mostrarão a configuração atual.

[Voltar ao início](#)

Eventos do núcleo

INet_SessionEvent

Objetivo

Esse evento é utilizado para notificar o cliente sobre a utilização do API da sessão do NCS. Os clientes podem utilizar esse evento para serem informados se outros clientes estão criando ou utilizando sessões.

Disparadores

Esse evento é disparado quando um cliente cria uma sessão, exclui uma sessão ou chama **`Apply`** para uma sessão.

Dados do evento

O **`EventType`** pode ter um dos seguintes valores:

- "New session" indica que uma nova sessão foi criada pelo cliente ou por um outro cliente..
- "End session" indica que um cliente finalizou uma sessão. A sessão pode ter sido terminada pelo cliente ou por um outro cliente.
- "Cache invalidated" indica que um outro cliente chamou **`Apply`** em uma sessão. Todas as outras sessões são invalidadas e o cache associado a suas sessões são excluídos.
- "Configuration changed" indica que a configuração da sessão mudou.

O **`SessionHandle`** contém o identificador da sessão que disparou o evento.

`OpenSessions` contém o número de sessões abertas. Esse item de dados é `NULL` para os eventos "Cache invalidated" e "Configuration changed".

IANet_InternalErrorEvent

Objetivo

Esse evento é utilizado para notificar o cliente que ocorreu um erro interno nos provedores de eventos. Em alguns casos isso significa que o provedor de eventos não é capaz de relatar mais eventos.

Disparadores

Esse evento ocorrerá:

- Após o provedor de eventos obter um evento desconhecido a partir de uma origem de eventos
- Após o software que fornece os eventos ter sido desligado
- Após o provedor de eventos obter um evento, mas não foi possível à origem de eventos obter mais dados sobre o evento

Dados do evento

O **EventType** pode ser um dos a seguir:

- "Could not get event data". Ocorreu um evento, mas não foi possível à origem de eventos obter mais dados sobre o evento.
- "Event source has shut down" A origem de dados para o evento foi desligada. Nesse caso o provedor de eventos também será desligado e não serão gerados mais eventos até que a origem seja reiniciada e sejam feitas novas consultas de notificação.
- "Unexpected message" O provedor de eventos recebeu um evento inesperado.

[Voltar ao início](#)

Casos de utilização

É necessário um identificador da sessão para alterar a configuração. O identificador da sessão permite ao software NCS gerenciar simultaneamente vários acessos para a configuração, impedindo, dessa forma, que uma sessão bloqueie todas as outras. Cada sessão possui um cache separado para armazenar quaisquer mudanças que tenham sido feitas. Se houver várias sessões realizando mudanças simultaneamente, o primeiro a aplicar suas mudanças terá sucesso. Todos os outros caches de sessão serão invalidados.

Obtenção de um identificador da sessão

O cliente deve obter o caminho do objeto da única instância de **IANet_NetService** antes de acessar o identificador da sessão. Chama **IWbemServices::CreateInstanceEnum** e passa o nome da classe: **IANet_NetService**. Isso é equivalente a chamar **IWbemServices::ExecQuery** com a consulta **SELECT * FROM IANet_NetService**. O cliente deve obter um identificador da sessão antes de efetuar quaisquer mudanças na configuração. Utilize o método **GetSessionHandle** para iniciar uma nova sessão.

É possível ao cliente utilizar **IWbemServices::ExecMethod** para executar um método em um objeto CIM e ele necessitará do caminho do objeto, a partir do atributo **__PATH** da instância de **IANet_NetService**. Esse método também retorna o número de sessões ativas no momento. O cliente pode querer emitir um aviso se não tiver acesso exclusivo ao NCS (Network Configuration Service).

Utilização de um identificador da sessão no objeto IWbemContext

Após o cliente obter um identificador da sessão ele deve criar um objeto **IWbemContext**. Armazene o identificador da sessão no qualificador **SessionHandle** desse objeto. Um ponteiro para esse objeto COM deverá ser passado para cada chamada no **IWbemServices**. Esse identificador da sessão não é necessário quando efetuar chamadas para acessar o objeto **IANet_NetService**, uma vez que isso considera o identificador como um argumento explícito.

Leitura de alterações pendentes ao utilizar um identificador da sessão

Ao ler a configuração, se passar o identificador da sessão no contexto, os provedores retornarão a configuração como se as atualizações pendentes fossem aplicadas (por exemplo, os adaptadores não instalados estarão faltando e as configurações alteradas retornarão seus valores originais). Contudo, alguns objetos não aparecerão até **Apply** ter sido chamado (por exemplo,

IANet_IPProtocolEndpoints não será criado até o protocolo ter sido vinculado à miniporta apropriada).

Finalização de um identificador da sessão

Após alterar a configuração, chame o método **Apply** para aceitar as mudanças. Isso pode retornar um código de ação de acompanhamento (por exemplo, reiniciar o sistema antes que as mudanças surtam efeito).

Chame sempre o **ReleaseSessionHandle** após a finalização de uma seção, do contrário, quaisquer alterações feitas serão descartadas. Chamar o método **Cancel** também descartará quaisquer alterações feitas, mas o cliente pode continuar a utilizar os identificadores da sessão como se ele tivesse sido recém criado.

Registro de eventos do núcleo

Os aplicativos devem usar o **IWbemServices::ExecNotificationQuery** ou **IWbemServices:: ExecNotificationQueryAsync** para solicitar notificação de evento. As consultas a seguir são exemplos de consultas de notificação de eventos (essa lista não é completa, uma vez que é possível muitas consultas):

- **SELECT * FROM IANet_Event** — Solicita todos os eventos.
- **SELECT * FROM IANet_SessionEvent** — Solicita todos os eventos da sessão.
- **SELECT * FROM IANet_InternalErrorEvent** — Solicita todos os eventos internos.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

O esquema do adaptador Ethernet: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Visão geral](#)

[IANet_EthernetAdapter](#)

[IANet_IPProtocolEndpoint](#)

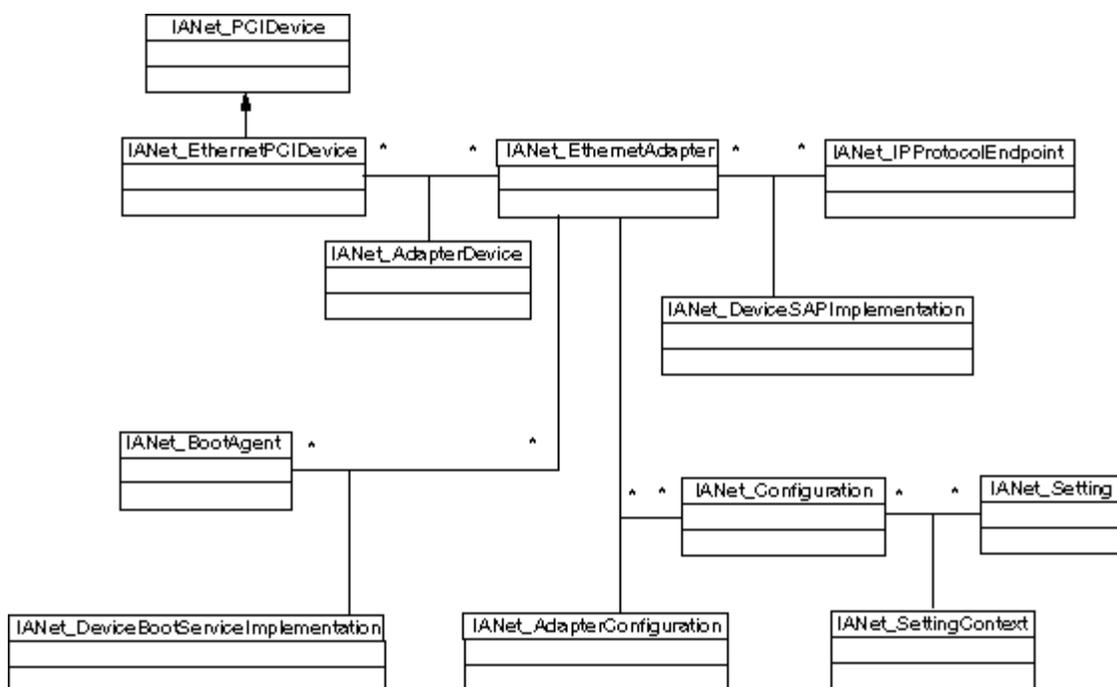
[IANet_BootAgent](#)

[IANet_PCIDevice](#)

[IANet_EthernetPCIDevice](#)

Visão geral

O esquema do adaptador é utilizado para modelar os vários adaptadores Ethernet Intel® PROSet configuráveis. Esse esquema está baseado no esquema CIM v2.5.



[Voltar ao início](#)

IANet_EthernetAdapter

Objetivo

O IANet_EthernetAdapter define os recursos e o status de todos os adaptadores de rede Intel PRO instalados, bem como quaisquer outros adaptadores que possam ser agrupados utilizando os drivers intermediários da Intel. A classe é derivada da superclasse CIM_EthernetAdapter, definida no CIMv2.5. O CIM_EthernetAdapter é derivado do CIM_NetworkAdapter, uma classe abstrata que define os conceitos gerais de hardware de rede, como PermanentAddress, CurrentAddress, Speed of operation etc.

Instâncias

Existirão instâncias dessa classe para cada um dos itens a seguir:

- NICs Intel suportados e instalados

- NICs não-Intel que podem participar em um grupo de vários fornecedores Intel
- Grupo criado de adaptadores Intel

Criação de instâncias

Não é possível criar instâncias de IANet_EthernetAdapter.

Remoção de instâncias

Excluir uma instância de IANet_EthernetAdapter desinstalará os adaptadores físicos. Apenas os adaptadores Intel não virtuais podem ser desinstalados dessa forma. É necessário um identificador da sessão para essa operação.

Modificação de propriedades

Não há propriedades modificáveis pelo usuário para essa classe.

Atributos não suportados

Os atributos a seguir não são requeridos pelo Intel PROSet e, portanto, não são suportados:

- AutoSense (este é exposto como uma configuração)
- ErrorCleared
- OtherIdentifyingInfo
- IdentifyingDescriptions
- InstallDate
- LastErrorCode
- MaxDataSize
- MaxQuiesceTime
- PowerManagementCapabilities (este é exposto como um método)
- PowerManagementSupported (este é exposto como um método)
- PowerOnHours
- ShortFramesReceived
- SymbolErrors
- TotalPowerOnHours

Métodos

Essa instância de classe suporta os métodos a seguir:

- **IdentifyAdapter** — Identifica o adaptador ao piscar a luz no adaptador por alguns segundos. Esse método funcionará apenas para adaptadores físicos.
- **HasVLANs** — Retorna o número de VLANs nesse adaptador.
- **IsPowerMgmtSupported** — Indica se o gerenciamento de energia é suportado pelo adaptador.
- **GetPowerUsage** — Detecta o uso geral de energia do adaptador.
0 = Energia normal
1 = Energia baixa
- **SetPowerUsage** — Reduz o uso geral de energia do adaptador. A configuração de uso geral de energia não é preservada entre reinicializações do sistema ou recargas do driver. Quando o sistema reinicializar ou o driver for recarregado, o adaptador retorna automaticamente ao consumo normal de energia.
- **GetPowerUsageOptions** — Detecta quaisquer configurações opcionais de uso de energia (por exemplo, uso de energia para espera, operação por bateria, etc.).
- **SetPowerUsageOptions** — Mudanças das opções de uso de energia (por exemplo, o método pode ser utilizado para reduzir o uso de energia para espera, operação por bateria, etc.)
Nota: As configurações de uso de energia são armazenadas e utilizadas para reinicializações subseqüentes.
- **TestCable** — Executa teste de diagnóstico em um adaptador específico. Em uma falha esse método retorna possíveis problemas, causas e soluções.
- **AdvancedTestCable** — Executa testes avançados de cabo em um adaptador específico. Esse conjunto de testes está disponível com os adaptadores de 1000 Mbps. O método retorna nomes de testes e os respectivos resultados.
Nota: Pode ocorrer a falha do link quando **SpeedDuplex** não estiver configurado para **Auto Negotiate**. Nessa instância o parâmetro de saída de **SpeedAndDuplexNotAutomatic** é TRUE.
- **TestLinkSpeed** — Determina se o adaptador está em execução na velocidade máxima. Se o adaptador estiver divulgando menos do que 1 Gbit, o método indica os motivos possíveis (por exemplo, "O parceiro do link não é capaz de executar em 1000 Mbps").

[Voltar ao início](#)

IANet_IPProtocolEndpoint

Objetivo

Essa classe é utilizada para descrever as configurações de IP para um ponto extremo do protocolo no sistema. O provedor WMI não fornece informações para quaisquer outros tipos de protocolos de rede.. A classe é derivado do CIM_IPProtocolEndpoint da classe de resumo. O provedor WMI fornece apenas informações do protocolo quando ele envolver uma entidade gerenciada pelo Intel PROSet.

Instâncias

Existirá uma instância do IANet_IPProtocolEndpoint para cada vinculação da pilha do protocolo IP com um ponto extremo suportado pela Intel (ou seja, adaptadores Intel, adaptadores com capacidade de agrupamento Intel e VLANs). Alguns adaptadores agrupados não possuem seu próprio endereço IP e, dessa forma, não possuem IANet_IPProtocolEndpoint associado diretamente à sua instância do adaptador. O IANet_IPProtocolEndpoint existe apenas após o sistema operacional ter vinculado o protocolo ao adaptador ou VLAN. Embora alguns adaptadores possam ter mais do que um endereço IP, ele será associado com apenas uma instância do ponto de extremidade do protocolo IP. O provedor não suporta esta utilização avançada, uma vez que ela não é necessária ou é utilizada pelo Intel PROSet.

Criação de instâncias

Não é possível criar instâncias de IANet_IPProtocolEndpoint. A instância existe apenas se o sistema operacional tiver vinculado o protocolo ao ponto de extremidade.

Remoção de instâncias

Não é possível remover instâncias de IANet_IPProtocolEndpoint.

Modificação de propriedades

Não há propriedades modificáveis pelo usuário para essa classe.

Associações

Uma instância de ANet_AdapterProtocolImplementation é utilizada para associar um IANet_EthernetAdapter com um IANet_IPProtocolEndpoint. Uma instância de IANet_VLANProtocolDependency é utilizada para associar uma VLAN com um IANet_IPProtocolEndpoint.

Nota: Grupos são associados ao ponto de extremidade através do adaptador que representa o adaptador virtual do grupo.

Atributos suportados

Os seguintes atributos somente leitura são exigidos pelo Intel PROSet:

- Address
- AddressType
- DefaultGateway
- DHCPServerAddress
- DHCPAutoAssign
- IPVersionSupport
- SubnetMask

Atributos não suportados

Os atributos a seguir não são exigidos pelo Intel PROSet e, portanto, não são suportados:

- Caption
- Description
- InstallDate
- NameFormat
- OtherTypeInfo
- ProtocolType
- Status

Métodos

Nenhum.

[Voltar ao início](#)

IANet_BootAgent

Purpose

Essa classe é utilizada para capturar informações sobre os recursos de inicialização de rede de um adaptador (por exemplo, configurações para o PXE Boot Agent suportadas por alguns adaptadores Intel). Essa classe é derivada do CIM_BootService.

Instâncias

Existe uma instância do IANet_BootAgent para cada adaptador que suporta o recurso boot agent, mesmo se o boot agent não estiver instalado no momento.

Criação de instâncias

Não é possível criar instâncias de IANet_BootAgent. Uma instância existe apenas se o adaptador suportar a funcionalidade do boot agent.

Remoção de instâncias

Não é possível remover instâncias de IANet_BootAgent.

Modificação de propriedades

Não há propriedades modificáveis pelo usuário dessa classe.

Associações

É utilizada uma instância de IANet_DeviceBootServiceImplementation para associar um IANet_EthernetAdapter a um IANet_BootAgent, se o adaptador suportá-lo.

Atributos suportados

Os seguintes atributos somente leitura são exigidos pelo Intel PROSet:

- InvalidImageSignature
- Versão
- UpdateAvailable
- FlashImageType

Atributos não suportados

Os atributos a seguir não são requeridos pelo Intel PROSet e, portanto, não são suportados:

- Caption
- Description
- InstallDate
- Started
- StartMode
- Status

Métodos

Os métodos a seguir nessa classe podem ser utilizados para atualizar a Flash ROM no NIC:

<pre>uint32 ProgramFlash([IN, ValueMap {"0","1"} , Values {"Check Version", "Write Flash"}: Amended] uint32 Action, [IN] uint8 NewFlashData[], [OUT] string strErrorMessage);</pre>	<p>Esse método é usado para atualizar a ROM FLASH no NIC. Isso fará com que o NIC pare de se comunicar com a rede enquanto o flash é atualizado.</p>
<pre>uint32 ReadFlash([OUT] uint8</pre>	<p>Esse método lê a Flash ROM no NIC.</p>

[Voltar ao início](#)

IANet_PCIDevice

Objetivo

Essa classe é utilizada para descrever as propriedades de um dispositivo PCI para um dispositivo de rede no sistema. Essa classe é derivada do CIM_PCIDevice.

Instâncias

Existe uma instância para essa classe para cada placa PCI que é um dispositivo de rede no sistema. Para IA64, terão instâncias apenas os dispositivos PCI que são adaptadores suportados pelo Intel PROSet.

Criação de instâncias

Não é possível criar instâncias de IANet_PCIDevice.

Remoção de instâncias

Não é possível remover instâncias de IANet_PCIDevice.

Modificação de propriedades

Não há propriedades modificáveis pelo usuário dessa classe.

Associações

Consulte o IANet_EthernetPCIDevice para obter as associações de classes.

Métodos

Não há métodos suportados nessa classe.

Atributos não suportados

Os atributos a seguir não são suportados pelo provedor WMI:

- AdditionalAvailability
- Capabilities
- CapabilityDescriptions
- Caption
- DeviceSelectTiming
- ErrorCleared
- ErrorDescription
- IdentifyingDescription
- InstallDate
- LastErrorCode
- MaxNumberController
- MaxQuiesceTime
- Name
- OtherIdentifyingInfo
- PowerManagementCapabilities
- PowerManagementSupported
- PowerOnHours
- ProtocolDescription
- ProtocolSupported
- SelfTestEnabled
- TimeOfLastReset
- TotalPowerOnHours

[Voltar ao início](#)

IANet_EthernetPCIDevice

Objetivo

Essa classe é utilizada para descrever as propriedades de um dispositivo PCI para um adaptador de rede Ethernet suportado pelo Intel PROSet. Essa é uma subclasse de IANet_PCIDevice. A classe contém alguns atributos extras que são conhecidos apenas por dispositivos PCI suportados pelo PROSet.

Instâncias

Existe uma instância para essa classe para cada placa PCI que é um adaptador de rede Ethernet suportado pelo Intel PROSet.

Criação de instâncias

Não é possível criar instâncias de IANet_EthernetPCIDevice.

Remoção de instâncias

Não é possível remover instâncias de IANet_EthernetPCIDevice.

Modificação de propriedades

Não há propriedades modificáveis pelo usuário para essa classe.

Associações

Uma instância de IANet_AdapterDevice é utilizada para associar um IANet_PCIDevice com o IANet_EthernetAdapter. Adaptadores virtuais (isto é, adaptadores criados para representar grupos) não possuem um IANet_PCIDevice associado.

Atributos não suportados

Os atributos a seguir não são suportados pelo provedor WMI:

- AdditionalAvailability
- Capabilities
- CapabilityDescriptions
- Caption
- DeviceSelectTiming
- ErrorCleared
- ErrorDescription
- IdentifyingDescription
- InstallDate
- LastErrorCode
- MaxNumberController
- MaxQuiesceTime
- Name
- OtherIdentifyingInfo
- PowerManagementCapabilities
- PowerManagementSupported
- PowerOnHours
- ProtocolDescription
- ProtocolSupported
- SelfTestEnabled
- Status
- StatusInfo
- TimeOfLastReset
- TotalPowerOnHours

Métodos

Não há métodos suportados nessa classe.

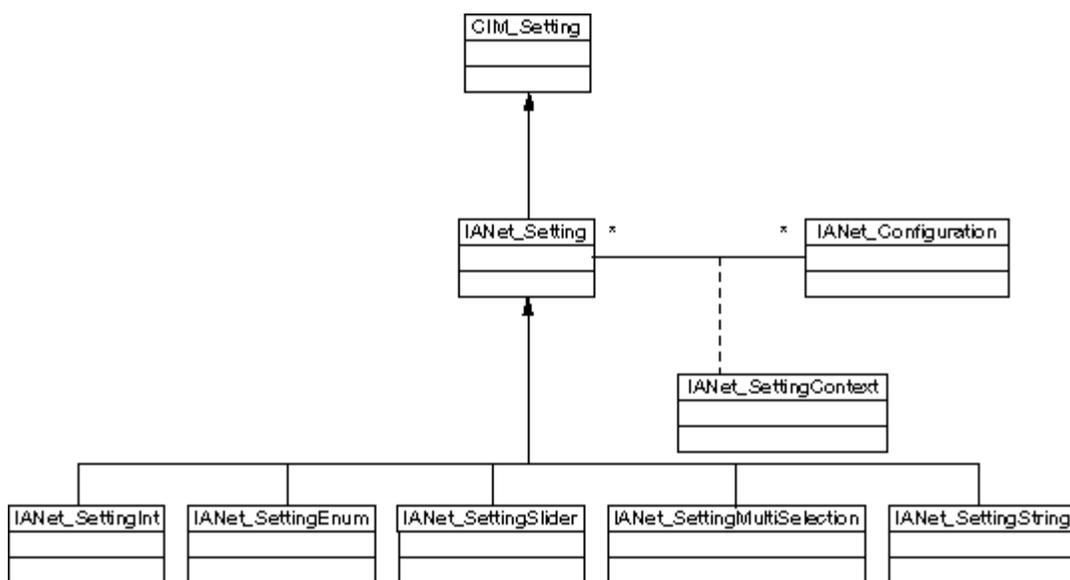
Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

O esquema de configuração: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

- [Visão geral](#)
- [IANet_Configuration](#)
- [IANet_Setting](#)
- [IANet_SettingInt](#)
- [IANet_SettingEnum](#)
- [IANet_SettingSlider](#)
- [IANet_SettingMultiSelection](#)
- [IANet_SettingString](#)

Visão geral



IANet_Configuration

Objetivo
Esta classe é utilizada para agrupar uma coleção de instâncias de `IANet_Setting`. Essa classe é derivada do `CIM_Configuration`.

Instâncias
Cada adaptador, VLAN ou grupo pode ter várias instâncias de `IANet_Configuration` associadas (cada configuração corresponde a um cenário de uso diferente para o adaptador).

Para essa versão de provedores WMI e CDM há apenas uma instância de `IANet_Configuration` para cada adaptador, VLAN ou grupo.

Criação de instâncias
Não é possível criar instâncias de `IANet_Configuration`.

Remoção de instâncias

Não é possível remover instâncias de IANet_Configuration.

Modificação de propriedades

Não há propriedades modificáveis pelo usuário para essa classe.

Associações

Existirá uma instância de IANet_AdapterConfiguration para associar cada adaptador (IANet_EthernetAdapter) à respectiva configuração. Existirá uma instância de IANet_VLANConfiguration para associar cada VLAN (IANet_VLAN) à respectiva configuração. Existirá uma instância de IANet_BootAgentConfiguration para associar cada boot agent (IANet_BootAgent) à respectiva configuração.

Métodos

Não há métodos suportados nessa classe.

Atributos não suportados

Nenhum.

[Voltar ao início](#)

IANet_Setting

Objetivo

Esta classe abstrata é utilizada para descrever uma propriedade configurável em uma configuração. Essa classe é derivada do CIM_Setting.

Instâncias

Existirá uma instância separada dessa classe para cada configuração em cada adaptador, VLAN ou grupo. As definições não são compartilhadas entre as configurações.

Há várias subclasses para IANet_Setting. As subclasses correspondem a tipos e faixas de valores diferentes que as configurações podem assumir. Cada subclasse corresponde a um estilo diferente de GUI que pode ser utilizada para exibir ou alterar as configurações.

Criação de instâncias

Não é possível criar instâncias de IANet_Setting.

Remoção de instâncias

Não é possível remover instâncias de IANet_Setting.

Modificação de propriedades

Esta classe abstrata não possui propriedades modificáveis, mas as classes filho possuem tais propriedades (veja abaixo).

Associações

Cada instância de IANet_Setting está associada a uma instância de IANet_Configuration utilizando uma instância de IANet_SettingContext.

Métodos

Não há métodos suportados nessa classe. Para efetuar mudanças a uma configuração, modifique a propriedade requerida e chame **PutInstance**.

Atributos não suportados

O SettingID não é utilizado.

[Voltar ao início](#)

IANet_SettingInt

Objetivo

Esta classe modela uma configuração que assume um valor inteiro. Há várias classes de configuração de IANet utilizadas para modelar valores inteiros. As diferenças entre essas classes estão relacionadas ao modo como a GUI exibe e modifica o valor

inteiro e ao modo como os provedores validam. Para `IANet_SettingInt` a GUI exibirá uma caixa de edição com um controle de incremento/decremento de valores.

Instâncias

Existe uma instância dessa classe para cada configuração que deverá ser exibida como uma caixa de edição de valores inteiros.

Criação de instâncias

Não é possível criar instâncias dessa classe.

Remoção de instâncias

Não é possível remover instâncias dessa classe.

Modificação de propriedades

O atributo "CurrentValue" é a única propriedade modificável dessa classe. É possível modificar essa propriedade ao utilizar `IWbemClassObject::Put()` para alterar o valor e, em seguida, chame `IWbemServices::PutInstance()` para atualizar a configuração. Os provedores verificarão se:

CurrentValue <= **max**

CurrentValue >= **min**

(**CurrentValue** - **min**) é um múltiplo de **Step**

Onde **max**, **min**, **CurrentValue** e **Step** são atributos de `IANet_SettingInt`.

Associações

Cada instância de `IANet_SettingInt` está associada a uma instância de `IANet_Configuration` utilizando uma instância de `IANet_SettingContext`.

Atributos não suportados

O `SettingID` não é utilizado.

Métodos

Não há métodos suportados nessa classe. Para efetuar mudanças a uma configuração, modifique a propriedade requerida e chame `PutInstance`.

[Voltar ao início](#)

IANet_SettingEnum

Objetivo

Esta classe modela uma configuração que assume um valor inteiro. Há várias classes de configuração de `IANet` utilizadas para modelar valores inteiros. As diferenças entre essas classes estão relacionadas ao modo como a GUI exibe e modifica o valor inteiro e ao modo como os provedores validam. Para `IANet_SettingEnum`, a GUI exibirá uma lista de strings que mapeia um pequeno número de valores enumerados (por exemplo, uma caixa de lista suspensa).

Instâncias

Existe uma instância dessa classe para cada configuração que será exibida como lista.

Criação de instâncias

Não é possível criar instâncias dessa classe.

Remoção de instâncias

Não é possível remover instâncias dessa classe.

Modificação de propriedades

O atributo **CurrentValue** é a única propriedade modificável dessa classe. Modifique essa propriedade ao utilizar `Put()` para alterar o valor e chame `PutInstance()` para atualizar a configuração. Os provedores verificarão se **CurrentValue** € **PossibleValues[]**

Associações

Cada instância de `IANet_SettingEnum` está associada a uma instância de `IANet_Configuration` utilizando uma instância de `IANet_SettingContext`.

Atributos não suportados

O SettingID não é utilizado.

Métodos

Não há métodos suportados nessa classe. Para efetuar mudanças a uma configuração, modifique a propriedade requerida e chame **PutInstance**.

[Voltar ao início](#)

IANet_SettingSlider

Objetivo

Esta classe modela uma configuração que assume um valor inteiro. Há várias classes de configuração de IANet utilizadas para modelar valores inteiros. As diferenças entre essas classes estão relacionadas ao modo como a GUI exibe e modifica o valor inteiro e ao modo como os provedores validam. Para o IANet_SettingSlider a GUI exibirá uma barra deslizante que permitirá a você escolher o valor de um modo gráfico - o valor real escolhido não necessita ser exibido.

Instâncias

Existe uma instância dessa classe para cada configuração que será exibida como uma barra deslizante.

Criação de instâncias

Não é possível criar instâncias dessa classe.

Remoção de instâncias

Não é possível remover instâncias dessa classe.

Modificação de propriedades

O atributo **CurrentValue** é a única propriedade modificável dessa classe. Modifique essa propriedade ao utilizar **Put()** para alterar o valor e chame **PutInstance()** para atualizar a configuração. Os provedores verificarão se **CurrentValue** € **PossibleValues[]**

Associações

Cada instância de IANet_SettingSlider está associada a uma instância de IANet_Configuration utilizando uma instância de IANet_SettingContext.

Atributos não suportados

O SettingID não é utilizado.

Métodos

Não há métodos suportados nessa classe. Para efetuar mudanças a uma configuração, modifique a propriedade requerida e chame **PutInstance**.

[Voltar ao início](#)

IANet_SettingMultiSelection

Objetivo

Esta classe modela uma configuração por meio da qual é possível selecionar várias opções em uma lista de opções. Para IANet_SettingMultiSelection a GUI exibirá uma caixa de lista com várias seleções, que permitirá escolher qualquer opção ou nenhuma.

Instâncias

Existe uma instância dessa classe para cada configuração que será exibida como uma caixa de lista com várias seleções.

Criação de instâncias

Não é possível criar instâncias dessa classe.

Remoção de instâncias

Não é possível remover instâncias dessa classe.

Modificação de propriedades

O atributo **CurrentValue** é a única propriedade modificável dessa classe. Modifique essa propriedade ao utilizar **Put()** para alterar

o valor e utilize **PutInstance()** para atualizar a configuração. Os provedores verificarão se **CurrentValue € PossibleValues[]**

Associações

Cada instância de IANet_SettingMultipleSelection está associada a uma instância de IANet_Configuration utilizando uma instância de IANet_SettingContext.

Atributos não suportados

O SettingID não é utilizado.

Métodos

Não há métodos suportados nessa classe. Para efetuar mudanças a uma configuração, modifique a propriedade requerida e chame **PutInstance**.

[Voltar ao início](#)

IANet_SettingString

Objetivo

Esta classe modela uma configuração por meio da qual é possível digitar um valor de string de formato livre. Para IANet_SettingMultiSelection, a GUI exibirá uma caixa de edição.

Instâncias

Existe uma instância dessa classe para cada configuração que será exibida como caixa de edição.

Criação de instâncias

Não é possível criar instâncias dessa classe.

Remoção de instâncias

Não é possível remover instâncias dessa classe.

Modificação de propriedades

O atributo **CurrentValue** é a única propriedade modificável dessa classe. Modifique essa propriedade ao utilizar **Put()** para alterar o valor e chame **PutInstance()** para atualizar a configuração.

Associações

Cada instância de IANet_SettingMultipleSelection está associada a uma instância de IANet_ElementConfiguration utilizando uma instância de IANet_SettingString.

Métodos

Não há métodos suportados nessa classe.

Atributos não suportados

O SettingID não é utilizado.

Métodos

Não há métodos suportados nessa classe. Para efetuar mudanças a uma configuração, modifique a propriedade requerida e chame **PutInstance**.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

O esquema de grupos: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Visão geral](#)

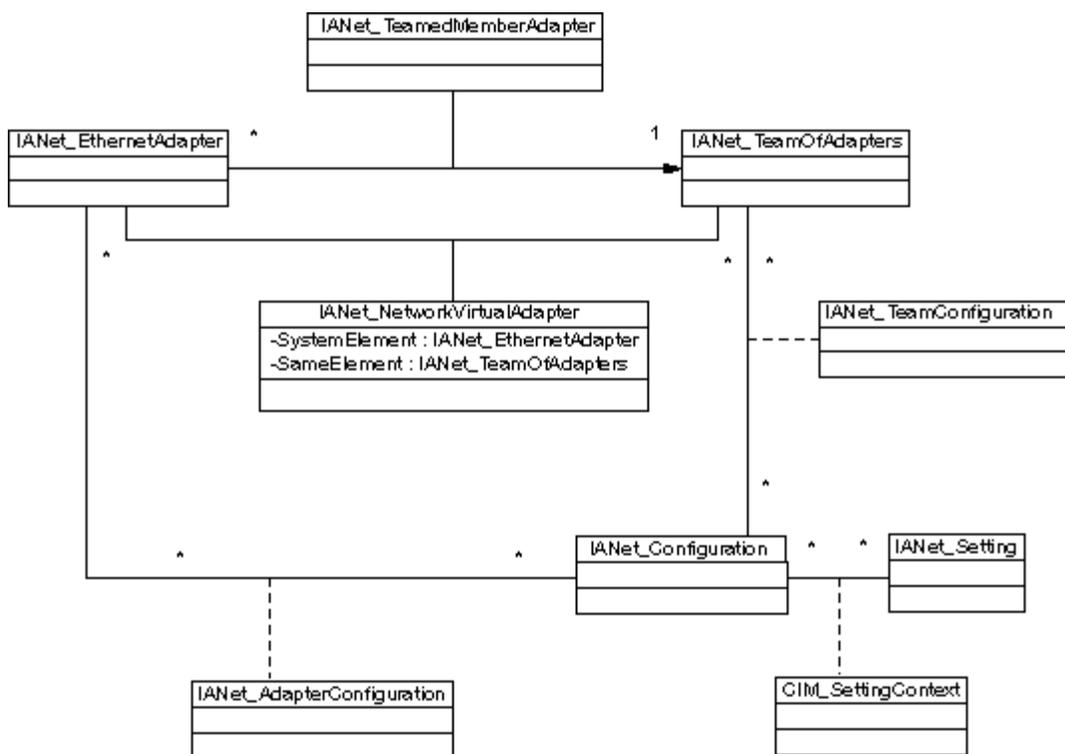
[IANet_TeamOfAdapters](#)

[IANet_TeamedMemberAdapter](#)

[IANet_NetworkVirtualAdapter](#)

Visão geral

O esquema de grupos descreve como os adaptadores Ethernet são reunidos em grupos.



[Voltar ao início](#)

IANet_TeamOfAdapters

Objetivo

Essa classe implementa a classe CIM_RedundancyGroup. Essa classe tem membros que descrevem o tipo de grupo, a quantidade de adaptadores no grupo e o número máximo de adaptadores que o grupo pode ter.

Instâncias

Há uma instância dessa classe para cada grupo Intel.

Criação de instâncias

Para criar um grupo vazio, crie uma instância de IANet_TeamOfAdapters. É necessário configurar o **TeamingMode** correto antes de chamar **IWbemServices::PutInstance()** para criar o objeto nos provedores. Os provedores retornam uma string contendo o caminho do novo objeto.

Remoção de instâncias

Da mesma forma, para remover um grupo, exclua a instância de `IANet_TeamOfAdapters`. Os provedores excluem as associações aos membros do grupo e o adaptador virtual e as configurações do grupo.

Modificação de propriedades

Use `Put()` para alterar o valor da propriedade `TeamingMode` e, em seguida, chame `PutInstance()` para atualizar o grupo.

Associações

Cada adaptador do grupo é associado a uma instância do grupo de `IANet_TeamOfAdapters` usando uma instância de `IANet_TeamMemberAdapter`. O adaptador virtual do grupo é associado a essa classe usando uma instância de `IA_NetNetworkVirtualAdapter`.

Métodos

Essa instância de classe suporta o seguinte método:

TestSwitchConfiguration — Testa a configuração do comutador para garantir que o grupo funcione corretamente com ele. Esse teste pode ser usado para verificar se os parceiros de link (ou seja, um dispositivo ao qual o adaptador se conecta, como outro adaptador, hub, comutador, etc.) suporta o modo de agrupamento de adaptadores escolhido. Por exemplo, se o adaptador é membro de um grupo de agregação de link, esse teste pode verificar quais parceiros conectados ao adaptador suportam a agregação de link.

[Voltar ao início](#)

IANet_TeamedMemberAdapter

Objetivo

Essa classe é utilizada para associar o adaptador ao grupo e determina a função do adaptador dentro do grupo, além de estabelecer que o adaptador está ativo no grupo naquele momento. Essa classe implementa a classe CIM `CIM_NetworkAdapterRedundancyComponent`.

Instâncias

Existe uma instância dessa classe para cada adaptador que faz parte do grupo.

Criação de instâncias

Para adicionar um adaptador a um grupo, crie uma instância de `IANet_TeamedMemberAdapter` para associar o adaptador ao grupo.

Remoção de instâncias

Para remover um adaptador do grupo, remova a instância de `IANet_TeamedMemberAdapter`. O adaptador não fará mais parte do grupo e pode ser vinculado a um ponto de extremidade do protocolo IP depois que a função **Apply()** é chamada.

Modificação de propriedades

A propriedade `AdapterFunction` dessa classe pode ser modificada a fim de descrever como o adaptador é usado dentro do grupo.

Associações

É uma classe de associação.

Métodos

Não há métodos suportados nessa classe.

[Voltar ao início](#)

IANet_NetworkVirtualAdapter

Objetivo

Essa classe é utilizada para associar o `IANet_TeamOfAdapters` do grupo ao `IANet_EthernetAdapter` que representa o adaptador virtual do grupo. Essa classe implementa a classe CIM `CIM_CIM_NetworkVirtualAdapter`.

Instâncias

Existe uma instância dessa classe para cada grupo Intel que foi vinculado a um adaptador virtual.

Criação de instâncias

Não é possível criar instâncias dessa classe. Para criar um grupo, crie uma instância de `IANet_TeamOfAdapters`. A classe só vai passar a existir depois que você chamar `IANet_NetService.Apply()` dentro do contexto de uma sessão válida e a instância `IANet_EthernetAdapter` for criada.

Remoção de instâncias

Não é possível excluir instâncias dessa classe.

Associações

É uma classe de associação.

Métodos

Não há métodos suportados nessa classe.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

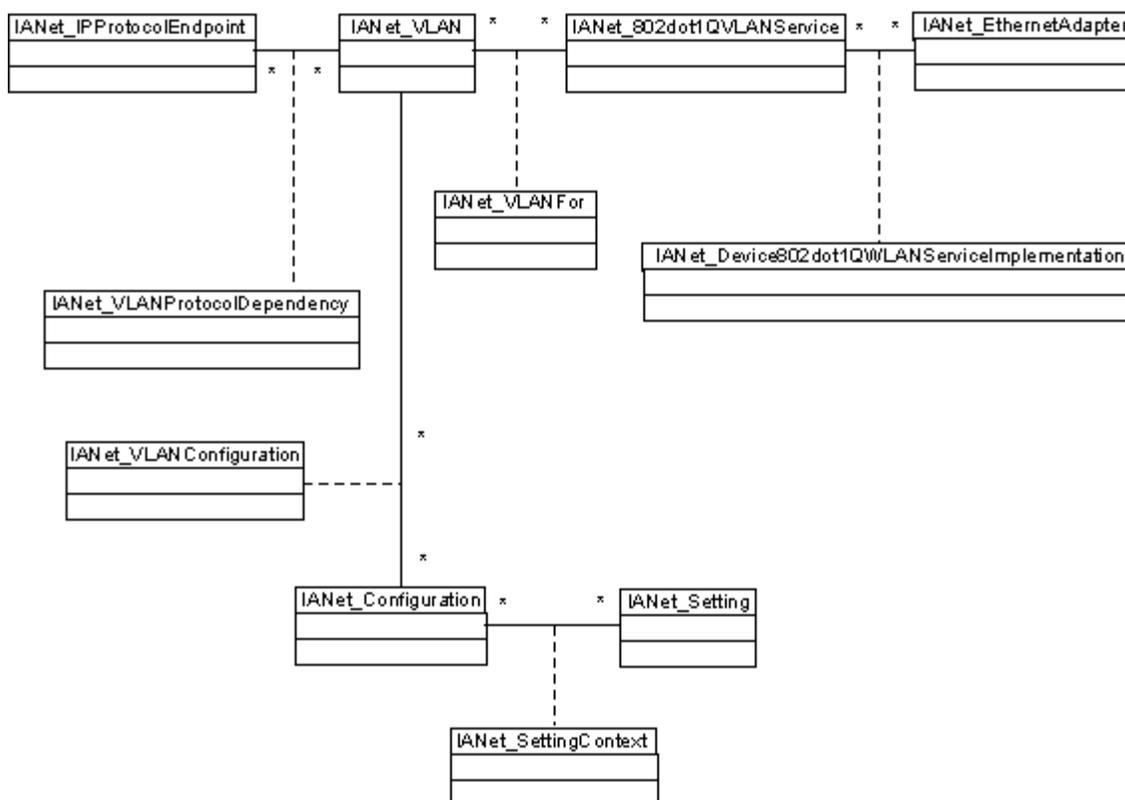
O esquema de VLAN: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Visão geral](#)

[IAnet_802dot1QVLANService](#)

[IAnet_VLAN](#)

Visão geral



[Voltar ao início](#)

IAnet_802dot1QVLANService

Objetivo

Essa classe é usada para conter as propriedades IEEE 802.1Q de um adaptador de rede. Essa classe implementa a classe CIM CIM_802dot1QVLANService.

Instâncias

Existe uma instância dessa classe para cada adaptador ou grupo que suporta IEEE 802.1Q. Cada adaptador pode ter apenas um IAnet_802dot1QVLANService. Alguns grupos, com os MVT tolerantes a falhas, não suportam esse serviço.

Anomalia

Um grupo sem VLANs não tem serviço de VLAN se você não enumerá-lo dentro do contexto de uma sessão válida. Para os grupos, a instância 802.3QvlanService aparece apenas nas situações a seguir:

- Se o grupo já tem VLANs
- Se o grupo não tem VLANs e você usa um identificador de sessão no contexto ao enumerar essa classe.

Criação de instâncias

Não é possível criar instâncias dessa classe. Se o adaptador não tiver uma instância associada a ele, o adaptador não suporta esse serviço.

Remoção de instâncias

Não é possível excluir instâncias dessa classe.

Modificação de propriedades

Essa classe não tem propriedades modificáveis.

Associações

Cada instância dessa classe será associada a um IANet_EthernetAdapter usando uma instância de IANet_DeviceServiceImplementation.

Cada instância de IANet_802dot1QVLANService pode suportar várias VLANs; cada VLAN será associada à instância usando a associação IANet_VLANFor.

Métodos

uint16 CreateVLAN([in] uint32 VLANNumber, [in] string Name, [out] IANet_VLAN REF VLANpath); — Usado para criar uma VLAN no adaptador ou grupo. O cliente deve fornecer o número e o nome da VLAN, e obterá o caminho de objeto da VLAN recém-criada.

[Voltar ao início](#)

IANet_VLAN

Objetivo

Essa classe contém as informações sobre cada VLAN Intel. Essa classe implementa CIM_VLAN.

Instâncias

Existe uma instância dessa classe para cada VLAN Intel.

Criação de instâncias

Para criar uma VLAN, chame **CreateVLAN** na instância adequada de IANet_802dot1QVLANService.

Remoção de instâncias

É possível remover uma instância dessa classe a fim de remover a VLAN correspondente.

Modificação de propriedades

É possível modificar o VLANNumber e o atributo Caption.

Associações

Cada instância é associada a uma instância de IANet_802dot1QVLANService e, portanto, a uma instância de IANet_EthernetAdapter, usando a classe IANet_VLANFor.

Cada instância pode ser associada a várias instâncias de IANet_Configuration para agrupar um conjunto de definições da VLAN. Nessa versão do provedor, há apenas um objeto IANet_Configuration por VLAN.

Cada instância pode ser associada a um IANet_IPProtocolEndpoint para fornecer as configurações de IP referentes à VLAN usando a classe IANet_VLANProtocolDependency.

Métodos

Nenhum

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Obtenção da configuração atual: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

- [Obtenção dos adaptadores físicos](#)
- [Obtenção dos dispositivos PCI](#)
- [Obtenção das configurações do adaptador](#)
- [Obtenção das configurações do grupo](#)
- [Obtenção das definições do grupo](#)
- [Obtenção da configuração da VLAN](#)
- [Obtenção das definições da VLAN](#)
- [Obtenção das informações sobre o protocolo IP](#)
- [Obtenção das informações sobre o boot agent](#)
- [Obtenção das definições do boot agent](#)

O cliente não precisa obter um identificador da sessão para ler a configuração atual. Os clientes podem usar um contexto NULL; entretanto, as mensagens de erro serão retornadas na linguagem padrão da máquina gerenciada. Na tabela a seguir, os itens entre { } são caminhos de objeto. Pressupõe-se que esses caminhos foram obtidos de consultas WQL anteriores. O cliente nunca deve ter necessidade de criar um caminho de objeto sem fazer uma consulta. O atributo **__PATH** de todo objeto contém o caminho do mesmo.

Em todos os casos de uso a seguir, os métodos **IWbemServices::ExecQuery** e **IWbemServices::ExecQueryAsync** são usados para executar consultas WQL .

Obtenção dos adaptadores físicos

Tarefa	Consulta WQL	Classe do resultado	Comentário
Enumere todos os adaptadores	SELECT * FROM IANet_EthernetAdapter	IANet_EthernetAdapter	Retorna todos os IANet_EthernetAdapters. Equivala a IWbemServices::CreateInstanceEnumAsync.
Determine se o adaptador é virtual.	ASSOCIATORS OF {caminho do adaptador} WHERE AssocClass = IANet_NetworkVirtualAdapter	IANet_TeamOfAdapters	Se o resultado da consulta não for nenhuma classe, o adaptador é real.
Determine se o adaptador é fantasma	ASSOCIATORS OF {caminho do adaptador} WHERE ResultClass = IANet_EthernetPCIDevice	IANet_EthernetPCIDevice	Se o adaptador não for virtual e a consulta não retornar nenhum objeto, o adaptador é fantasma.

A classe principal dos adaptadores é IANet_EthernetAdapter. Essa classe é usada para adaptadores físicos e virtuais; o cliente deve saber fazer a distinção entre eles.

Obtenção dos dispositivos PCI

As classes principais são IANet_EthernetPCIDevice, IANet_PCIDevice e IANet_AdapterDevice (uma classe de associação para associar um adaptador a esse dispositivo).

Nesse caso, a classe de associação não contém dados, ou seja, não tem valor por si mesma. IANet_EthernetPCIDevice herda de IANet_PCIDevice e contém atributos extra que são específicos para um dispositivo PCI que é um adaptador Ethernet.

Tarefa	Consulta WQL	Classe do resultado	Comentário
Enumere os dispositivos PCI de rede	SELECT * FROM IANet_PCIDevice	IANet_PCIDevice	A consulta pode retornar modems e outros dispositivos que o Intel® PROSet não configura. Equivale a IWbemServices::CreateInstanceEnumAsync.
Enumere todos os dispositivos PCI usados pelos adaptadores configurados pelo Intel® PROSet.	SELECT * FROM IANet_EthernetPCIDevice	IANet_EthernetPCIDevice	A consulta retorna somente os dispositivos PCI que o Intel® PROSet gerencia. Equivale a IWbemServices::CreateInstanceEnumAsync.
Determine se o adaptador está instalado.	N/D	IANet_EthernetPCIDevice	Verifique o atributo "Availability" do IANet_EthernetPCIDevice. Se for igual a 10 - "Not Installed", o dispositivo não está instalado. Nota: Nesse estágio, não se tem muita informação sobre o dispositivo.
Obtenha o dispositivo PCI associado ao adaptador	ASSOCIATORS OF {caminho de IANet_EthernetAdapter} WHERE ResultClass = IANet_EthernetPCIDevice	IANet_EthernetPCIDevice	Se o comando não retornar nenhum objeto, o adaptador é virtual ou fantasma.
Obtenha o adaptador associado a um dispositivo PCI	ASSOCIATORS OF {caminho do dispositivo Ethernet PCI} WHERE ResultClass = IANet_EthernetAdapter	IANet_EthernetAdapter	Essa consulta não é ideal para o provedor — é melhor que o cliente inicie com o adaptador.

[Voltar ao início](#)

Obtenção das configurações do adaptador

Os objetos de definição não estão associados diretamente ao adaptador. Em conformidade com o padrão CIM, eles estão associados a um objeto de configuração, associado ao adaptador.

As classes envolvidas nessa parte do esquema são IANet_EthernetAdapter, IANet_Configuration, IANet_SettingInt, IANet_SettingString, IANet_SettingEnum, IANet_SettingMultiSelection e IANetSettingSlider.

As classes de associação IANet_AdapterConfiguration e IANet_SettingContext não contêm dados reais — elas atuam como uma "cola" entre as definições e o objeto pai.

Tarefa	Consultas WQL	Classe do resultado	Comentários
Obtenha o objeto de configuração para o adaptador.	ASSOCIATORS OF {caminho de IANet_EthernetAdapter} WHERE ResultClass = IANet_Configuration	IANet_Configuration	Retorna exatamente um objeto; mesmo se não houver definições, sempre haverá um objeto de configuração. O caminho desse objeto é usado na próxima consulta.
Obtenha as definições associadas ao	ASSOCIATORS OF {caminho de IANet_Configuration}	Uma mescla de IANet_SettingInt, IANet_SettingString,	Retorna todas as classes de definição associadas ao adaptador. O cliente deve usar o atributo __CLASS para determinar o tipo de

adaptador	WHERE AssocClass = IANet_SettingContext	IANet_SettingEnum, IANet_SettingMultiSelection, IANet_SettingSlider	cada definição.
-----------	--	---	-----------------

[Voltar ao início](#)

Obtenção da configuração do grupo

As classes principais do esquema de agrupamento são IANet_EthernetAdapter, IANet_TeamOfAdapters, IANet_NetworkVirtualAdapter e IANet_TeamedMemberAdapter.

A dificuldade desse esquema é que existe uma instância de IANet_EthernetAdapter para cada adaptador físico e virtual. O cliente deve ter a capacidade de fazer a distinção entre o adaptador virtual de um grupo e os adaptadores que são membros do grupo.

A classe de associação IANet_NetworkVirtualAdapter não contém dados úteis — na verdade, os clientes só estão interessados nos pontos de extremidade dessa associação. IANet_TeamedMemberAdapter contém dados úteis sobre o uso do adaptador membro dentro do grupo.

Tarefa	Consultas WQL	Classe do resultado	Comentários
Enumere todos os grupos	SELECT * FROM IANet_TeamOfAdapters	IANet_TeamOfAdapters	Há uma instância de IANet_TeamOfAdapters para cada grupo. Equivale a IWbemServices::CreateInstanceEnumAsync.
Obtenha o adaptador virtual de um grupo	ASSOCIATORS OF {caminho de IANet_TeamOfAdapters} WHERE AssocClass = IANet_NetworkVirtualAdapter	IANet_EthernetAdapter	Retorna somente o objeto de adaptador para o adaptador virtual do grupo. Esse adaptador não existe quando o grupo foi criado mas Apply não foi chamado. (veja abaixo como atualizar a configuração).
Enumere os adaptadores dos membros do grupo	ASSOCIATORS OF {caminho de IANet_TeamOfAdapters} WHERE AssocClass = IANet_TeamedMemberAdapter	IANet_EthernetAdapter	Retorna os adaptadores que estão no grupo, mas não descreve a função desempenhada pelo adaptador.
Determine a função do adaptador em um grupo	REFERENCES OF {caminho de IANet_EthernetAdapter} WHERE ResultClass = IANet_TeamedMemberAdapter	IANet_TeamedMemberAdapter	A classe contém informações sobre a relação entre o adaptador membro e o grupo e sobre o status atual dentro do grupo.

[Voltar ao início](#)

Obtenção das definições do grupo

Os objetos de definição não estão associados diretamente ao grupo. Em conformidade com o padrão CIM, eles estão associados a um objeto de configuração, associado ao IANet_EthernetAdapter virtual do grupo. O mesmo objeto de configuração também está associado ao objeto IANet_TeamOfAdapters do grupo.

As classes envolvidas nessa parte do esquema são IANet_EthernetAdapter, IANet_TeamOfAdapters, IANet_Configuration, IANet_SettingInt, IANet_SettingString, IANet_SettingEnum, IANet_SettingMultiSelection e IANetSettingSlider.

As classes de associação IANet_AdapterConfiguration e IANet_SettingContext não contém dados reais — elas atuam como uma "cola" entre as definições e o objeto pai. É exatamente igual ao caso das definições de adaptador.

Tarefa	Consultas WQL	Classe do resultado	Comentários
Obtenha o objeto de configuração do grupo, começando pelo adaptador virtual	ASSOCIATORS OF {caminho de IANet_EthernetAdapter} WHERE ResultClass = IANet_Configuration	IANet_Configuration	Retorna exatamente um objeto. Mesmo se não houver definições, sempre haverá um objeto de configuração. O caminho desse objeto é usado na próxima consulta.
Obtenha o objeto de configuração do grupo, começando pelo grupo de adaptadores	ASSOCIATORS OF {caminho de IANet_TeamOfAdapters} WHERE ResultClass = IANet_Configuration		
Obtenha as definições associadas ao adaptador	ASSOCIATORS OF {caminho de IANet_Configuration} WHERE AssocClass = IANet_SettingContext	Uma mescla de IANet_SettingInt, IANet_SettingString, IANet_SettingEnum, IANet_SettingMultiSelection, IANet_SettingSlider	Retorna todas as classes de definição associadas ao adaptador. O cliente deve usar o atributo __CLASS para determinar o tipo de cada definição.

[Voltar ao início](#)

Obtenção da configuração da VLAN

Cada adaptador que suporta VLANs tem um IANet_802dot1QVLANService associado a ele, usando a classe de associação IANet_Device802do1QVLANServiceImplementation. Se o adaptador não tiver uma instância dessa classe associada a ele, o adaptador não suporta VLANs.

Cada VLAN é representada por uma instância de IANet_VLAN. A VLAN não é associada diretamente ao adaptador — é associada ao IANet_802dot1QVLANService do adaptador.

A classe de associação IANet_VLANFor é usada para associar cada instância da VLAN ao IANet_802dot1QVLANService correto. Essa classe não contém dados úteis para o usuário.

Tarefa	Consultas WQL	Classe do resultado	Comentários
Obtenha o objeto de serviço 802.1q VLAN associado ao adaptador	ASSOCIATORS OF {caminho de IANet_EthernetAdapter} WHERE ResultClass = IANet_802dot1QVLANService	IANet_802dot1QVLANService	Retorna um objeto ou nenhum.
Obtenha as VLANs em um adaptador.	ASSOCIATORS OF {caminho de IANet_802dot1QVLANService} WHERE ResultClass = IANet_VLAN	IANet_VLAN	Pode não retornar nenhum objeto se não houver VLANs instaladas.

[Voltar ao início](#)

Obtenção das definições de VLAN

Os objetos de definição não estão associados diretamente à VLAN. Em conformidade com o padrão CIM, eles estão associados a um objeto de configuração, associado ao OBJETO IANet_VLAN da VLAN.

As classes envolvidas nessa parte do esquema são are IANet_VLAN, IANet_Configuration, IANet_SettingInt, IANet_SettingString, IANet_SettingEnum, IANet_SettingMultiSelection e IANetSettingSlider.

As classes de associação IANet_VLANConfiguration e IANet_SettingContext não contêm dados reais — elas atuam como uma "cola" entre as definições e o objeto pai. É exatamente igual ao caso das definições de adaptador.

Tarefa	Consultas WQL	Classe do resultado	Comentários
Obtenha o objeto de configuração para a VLAN.	ASSOCIATORS OF {caminho da IANet_VLAN} WHERE ResultClass = IANet_Configuration	IANet_Configuration	Retorna exatamente um objeto; mesmo se não houver definições, sempre haverá um objeto de configuração. O caminho desse objeto é usado na próxima consulta.
Obtenha as definições associadas à VLAN	ASSOCIATORS OF {caminho de IANet_Configuration} WHERE AssocClass = IANet_SettingContext	Uma mescla de IANet_SettingInt, IANet_SettingString, IANet_SettingEnum, IANet_SettingMultiSelection, IANet_SettingSlider	Retorna todas as classes de definição associadas à VLAN. O cliente deve usar o atributo __CLASS para determinar o tipo de cada definição.

[Voltar ao início](#)

Obtenção das informações sobre o protocolo IP

O provedor fornece algumas informações limitadas sobre os pontos de extremidade do protocolo IP que estão associadas a adaptadores, VLANs e grupos. Não há suporte para outros protocolos.

A classe principal que contém as informações sobre protocolo é IANet_IPProtocolEndpoint. Há duas classes de associação: IANet_VLANProtocolDependency e IANet_AdapterProtocolImplementation. Para obter o ponto de extremidade IP de um grupo, primeiro obtenha o IANet_EthernetAdapter virtual do grupo, ou seja, o ponto de extremidade IP é associado a essa instância.

Tarefa	Consultas WQL	Classe do resultado	Comentários
Obtenha os pontos de extremidade de protocolo IP associados ao adaptador	ASSOCIATORS OF {caminho de IANet_EthernetAdapter} WHERE ResultClass = IANet_IPProtocolEndpoint	IANet_IPProtocolEndpoint	Embora alguns adaptadores tenham mais de um endereço IP, eles estarão associados a apenas uma instância de ponto de extremidade de protocolo IP.
Obtenha os pontos de extremidade de protocolo IP associados à VLAN.	ASSOCIATORS OF {caminho de IANet_VLAN} WHERE ResultClass = IANet_IPProtocolEndpoint	IANet_IPProtocolEndpoint	Haverá um ponto de extremidade de protocolo IP associado à VLAN.

[Voltar ao início](#)

Obtenção de informações sobre o boot agent

Cada adaptador que suporta um boot agent em ROM Flash terá uma instância de IANet_BootAgent associada a ele usando a classe de associação IANet_DeviceBootServiceImplementation.

Tarefa	Consultas WQL	Classe do resultado	Comentários
Obtenha o boot	ASSOCIATORS OF {caminho de	IANet_BootAgent	Os atributos "somente leitura" a seguir

agent associado ao adaptador	IANet_EthernetAdapter} WHERE ResultClass = IANet_BootAgent	forneem informações sobre a imagem ROM de inicialização para o adaptador. InvalidImageSignature, Version, UpdateAvailable, FlashImageType
------------------------------	--	---

[Voltar ao início](#)

Obtenção das definições do boot agent

Os objetos de definição não estão associados diretamente ao boot agent. Em conformidade com o padrão CIM, eles estão associados a um objeto de configuração, associado ao boot agent.

As classes envolvidas nessa parte do esquema são IANet_BootAgent, IANet_Configuration, IANet_SettingInt, IANet_SettingString, IANet_SettingEnum, IANet_SettingMultiSelection e IANetSettingSlider.

As classes de associação IANet_BootAgentConfiguration e IANet_SettingContext não contêm dados reais — elas atuam como "cola" entre as definições e o objeto pai.

Tarefa	Consultas WQL	Classe do resultado	Comentários
Obtenha o objeto de configuração para o boot agent.	ASSOCIATORS OF {caminho de IANet_BootAgent} WHERE ResultClass = IANet_Configuration	IANet_Configuration	Retorna exatamente um objeto. Mesmo se não houver definições, sempre haverá um objeto de configuração. O caminho desse objeto é usado na consulta a seguir.
Obtenha as definições associadas ao boot agent	ASSOCIATORS OF {caminho de IANet_Configuration} WHERE AssocClass = IANet_SettingContext	Uma mescla de IANet_SettingInt, IANet_SettingString, IANet_SettingEnum, IANet_SettingMultiSelection, IANet_SettingSlider	Retorna todas as classes de definição associadas ao adaptador. O cliente deve usar o atributo __CLASS para determinar o tipo de cada definição.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Atualização da configuração: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Visão geral](#)

[Alteração das configurações de adaptador, grupo ou VLAN](#)

[Criação de um novo grupo \(Vazio\)](#)

[Acréscimo de um adaptador a um grupo](#)

[Remoção do adaptador de um grupo](#)

[Exclusão de um grupo](#)

[Alteração do modo de um grupo](#)

[Alteração da prioridade de um adaptador dentro de um grupo](#)

[Desinstalação de um adaptador](#)

[Criação de uma VLAN](#)

[Alteração dos atributos de uma VLAN](#)

[Exclusão de uma VLAN](#)

[Atualização do boot agent](#)

Visão geral

Na maioria dos casos, para atualizar a configuração, o aplicativo cliente precisa obter um identificador da sessão da classe IANet_NetService e armazenar a identificação em um objeto de contexto IWbemContext. As alterações na configuração só ocorrem quando o método **Apply** no IANet_NetService é chamado. Existem algumas exceções para esse requisito:

- As alterações na classe do agente de inicialização ocorrem no momento em que são feitas e não requerem identificador da sessão.
- Certas chamadas de método (por exemplo: identificar o adaptador) fazem com que uma operação seja realizada antes que **Apply** seja chamado.

Em algumas operações, você pode usar o qualificador PreCheck no contexto para verificar se a operação é permitida. Isso acontece para permitir que a interface desabilite determinados controles ou itens de menu se for necessário.

[Voltar ao início](#)

Alteração das configurações do adaptador, grupo ou VLAN

Alterar o adaptador, grupo ou VLAN

- Requer identificador da sessão.
- PreCheck está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para alterar a configuração de um adaptador, VLAN ou grupo, o cliente deve obter primeiro o caminho de objeto da configuração que irá alterar. A melhor forma de fazer isso é enumerar as configurações do objeto e armazenar o atributo **__PATH** da configuração (consulte os parágrafos acima).

A forma mais fácil de o cliente atualizar uma configuração é a seguinte:

1. Obter a instância do objeto de configuração do WMI.
2. Modificar o atributo CurrentValue (usando **IWbemClassObject::Put()**).
3. Chamar **IWbemServices::PutInstance()** para passar a instância modificada ao provedor WMI. **PutInstance** deve ser chamado com o indicador WBEM_FLAG_UPDATE_ONLY.

O provedor WMI valida CurrentValue e retorna WBEM_E_FAIL se a validação falhar. O motivo exato da falha será retornado no atributo Description do objeto IANet_ExtendedStatus.

As descrições específicas da configuração envolvem:

- O número inteiro correspondente ao valor da configuração era menor que o mínimo permitido.
- O número inteiro correspondente ao valor da configuração era maior que o máximo permitido.
- O número inteiro correspondente ao valor da configuração não é permitido.
- O comprimento da string de configuração é maior que o máximo permitido.
- O valor da string não é um dos valores permitidos.

A última descrição é retornada quando o valor atual de `IANet_SettingEnum`, `IANet_SettingSlider` ou `IANet_SettingMultiSelection` não for um dos valores permitidos.

O único atributo de configuração que o cliente pode alterar é `CurrentValue`. O provedor WMI ignora as alterações feitas a qualquer um dos outros valores.

Não há métodos suportados na classe de configuração. Para fazer alterações em uma configuração, modifique a propriedade `CurrentValue`, chame **PutInstance**.

[Voltar ao início](#)

Criação de um novo grupo (Vazio)

Criação de um novo grupo

- Requer identificador da sessão.
- `PreCheck` está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para criar um novo grupo, crie uma instância de `IANet_TeamOfAdapters` (ou seja., use **`IWbemServices::GetObject()`** para obter um objeto de classe para `IANet_TeamOfAdapters` e, em seguida, utilize **`IWbemServices::SpawnInstance()`** para criar uma instância para esse objeto).

Em seguida, use **`IWbemClassObject::Put`** para configurar o atributo `TeamMode` na instância para que seja do tipo que você deseja para o grupo (exemplo: `AFT`). Por fim, chame **`IWbemServices::PutInstance()`** para criar o grupo, passando o indicador `WBEM_FLAG_CREATE_ONLY`.

O caminho de objeto do novo grupo é armazenado em **`IWbemCallResultObject`** que é passado novamente a você quando a chamada termina. O método **`IWbemCallResult::GetResultString`** obtém o novo caminho de objeto.

Se essa ação falhar, o cliente deve verificar o status de `IANet_ExtendedStatus` para ver os motivos da falha.

As classes `IANet_EthernetAdapter` virtual e `IANet_IPProtocolEndpoint` do grupo não ficam disponíveis até que **Apply** seja chamado. É possível acessar as configurações do grupo por meio do objeto `IANet_Configuration` associado ao novo `IANet_TeamOfAdapters`.

[Voltar ao início](#)

Acréscimo de um adaptador a um grupo

Acréscimo de um Adaptador a um Grupo:

- Requer identificador da sessão.
- `PreCheck` está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para adicionar um adaptador a um grupo, crie uma instância de `IANet_TeamedMemberAdapter` (ou seja., use **`IWbemServices::GetObject()`** para obter um objeto de classe para `IANet_TeamedMemberAdapter` e, em seguida, utilize **`IWbemServices::SpawnInstance()`** para criar uma instância para esse objeto).

É necessário configurar os seguintes atributos do objeto por meio de `IWbemClassObject::Put()`:

- É necessário configurar GroupComponent para que seja o caminho de objeto completo para o IANet_TeamOfAdapters ao qual o adaptador deve ser adicionado.
- É necessário configurar PartComponent para que seja o caminho de objeto completo do IANet_EthernetAdapter que deve ser adicionado ao grupo.

Também é possível configurar a prioridade para o adaptador no grupo. Por fim, chame **IWbemServices::PutInstance()** para adicionar o adaptador ao grupo, passando o indicador WBEM_FLAG_CREATE_ONLY. Se essa ação falhar, verifique IANet_ExtendedStatus para ver o código de erro.

[Voltar ao início](#)

Remoção de um adaptador de um grupo

Remoção de um adaptador de um grupo:

- Requer identificador da sessão.
- PreCheck está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para remover um adaptador de um grupo, exclua a instância IANet_TeamedMemberAdapter que associa o adaptador ao grupo usando **IWbemServices::DeleteInstance()**. Se essa ação falhar, verifique IANet_ExtendedStatus para ver o código de erro.

[Voltar ao início](#)

Exclusão de um grupo

Exclusão de um grupo:

- Requer identificador da sessão.
- PreCheck está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para excluir um grupo, exclua a instância IANet_TeamOfAdapters usando **IWbemServices::DeleteInstance()**. Se essa ação falhar, verifique IANet_ExtendedStatus para ver o código de erro.

[Voltar ao início](#)

Alteração do modo de um grupo

Alteração do modo de um grupo:

- Requer identificador da sessão.
- PreCheck está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para alterar o modo de um grupo, obtenha a instância de IANet_TeamOfAdapters para o grupo (por exemplo: use **IWbemServices::GetObject** utilizando o caminho de objeto do grupo). Em seguida, use **IWbemClassObject::Put** para alterar o atributo TeamMode do grupo. Por fim, chame **IWbemClassObject::PutInstance** para instruir o provedor WMI a atualizar o modo do grupo, passando o indicador WBEM_FLAG_UPDATE_ONLY. Se essa ação falhar, verifique IANet_ExtendedStatus para ver o código de erro.

[Voltar ao início](#)

Alteração da prioridade de um adaptador dentro de um grupo

Alteração da prioridade de um adaptador dentro de um grupo:

- Requer identificador da sessão.
- PreCheck está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para alterar a prioridade de um adaptador, o cliente deve primeiro obter a instância de `IANet_TeamedMemberAdapter` para o adaptador. (por exemplo: use `IWbemServices::GetObject` usando o caminho do objeto). Em seguida, use `IWbemClassObject::Put` para alterar o atributo `AdapterFunction` do adaptador. Por fim, o cliente precisa chamar `IWbemClassObject::PutInstance` para instruir o provedor WMI a atualizar a prioridade do adaptador. Se essa ação falhar, o cliente deve verificar `IANet_ExtendedStatus` para ver o código de erro.

[Voltar ao início](#)

Desinstalação de um adaptador

Desinstalação de um adaptador:

- Requer identificador da sessão.
- PreCheck está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para desinstalar um adaptador, chame `IWbemServices::DeleteInstance`, passando o caminho de objeto do adaptador a ser desinstalado.

[Voltar ao início](#)

Criação de uma VLAN

Criação de uma VLAN:

- Requer identificador da sessão.
- PreCheck está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para criar uma VLAN, chame o método `CreateVLAN` no `IANet_802dot1QVLANService` referente ao adaptador ao qual a VLAN deve ser adicionada. É necessário passar os seguintes argumentos ao método:

- `VLANNumber`, que é o número da VLAN. (Faixa 1- 4094)
- `Name`, que é o nome (definido pelo usuário) para identificar a VLAN.

A função retorna o caminho de objeto da VLAN recém-criada no parâmetro de saída (Out) `VLANpath`. Se essa ação falhar, verifique `IANet_ExtendedStatus` para ver o código de erro.

[Voltar ao início](#)

Alteração dos atributos de uma VLAN

Alteração dos atributos de uma VLAN:

- Requer identificador da sessão.
- PreCheck está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

O cliente pode alterar os atributos `VLANNumber` e `VLANName` de uma VLAN. Para alterar a prioridade de um adaptador, primeiro obtenha a instância de `IANet_VLAN` referente ao adaptador (por exemplo: use `IWbemServices::GetObject` utilizando o caminho de objeto).

Em seguida, altere `VLANNumber` ou `VLANName` de acordo com os valores desejados. Por fim, chame `IWbemClassObject::`

PutInstance para instruir o provedor WMI a atualizar os atributos, passando o indicador `WBEM_FLAG_UPDATE_ONLY`. Se essa ação falhar, verifique `IANet_ExtendedStatus` para ver o código de erro.

[Voltar ao início](#)

Exclusão de uma VLAN

Exclusão de uma VLAN:

- Requer identificador da sessão.
- PreCheck está disponível.
- Requer uma chamada de **Apply** antes que a operação seja executada.

Para excluir uma VLAN, chame **IWbemServices::DeleteInstance**, passando o caminho de objeto da VLAN a ser excluída.

[Voltar ao início](#)

Atualização do boot agent

Atualização do boot agent

- Não requer identificador da sessão
- PreCheck não está disponível.
- Não requer uma chamada de **Apply** antes que a operação seja executada.

O cliente pode atualizar a Boot Agent Image usando chamadas de métodos. Para ler/gravar uma imagem flash, primeiro obtenha a instância de `IANet_BootAgent` referente ao adaptador (por exemplo: use **IWbemServices::GetObject** utilizando o caminho de objeto).

Em seguida, execute `ReadFlash()` para ler a imagem da ROM de inicialização flash ou `ProgramFlash()` para atualizar essa imagem. Se essa ação falhar, verifique `IANet_ExtendedStatus` para ver o código de erro.

Tarefa	Métodos WMI	Resultado	Comentários
Atualize ou insira uma imagem da ROM de inicialização para o adaptador.	<pre>uint32 ProgramFlash([IN, ValueMap {"0","1"} , Values {"Check Version", "Write Flash"}: Amended] uint32 Action, [IN] uint8 NewFlashData[], [OUT] string</pre>	<p>Se a ação "Check Version" for especificada, esse método retornará uma mensagem de advertência, caso a imagem da ROM de inicialização que está sendo atualizada como em <code>NewFlashData[]</code> for mais antiga que a já existente no NIC.</p> <p>Se a ação "Write" for especificada, esse comando atualiza a ROM FLASH no NIC com <code>NewFlashData[]</code>.</p>	<p>Esse método é usado para atualizar a ROM FLASH no NIC. Isso fará com que o NIC pare de se comunicar com a rede enquanto o flash é atualizado.</p>

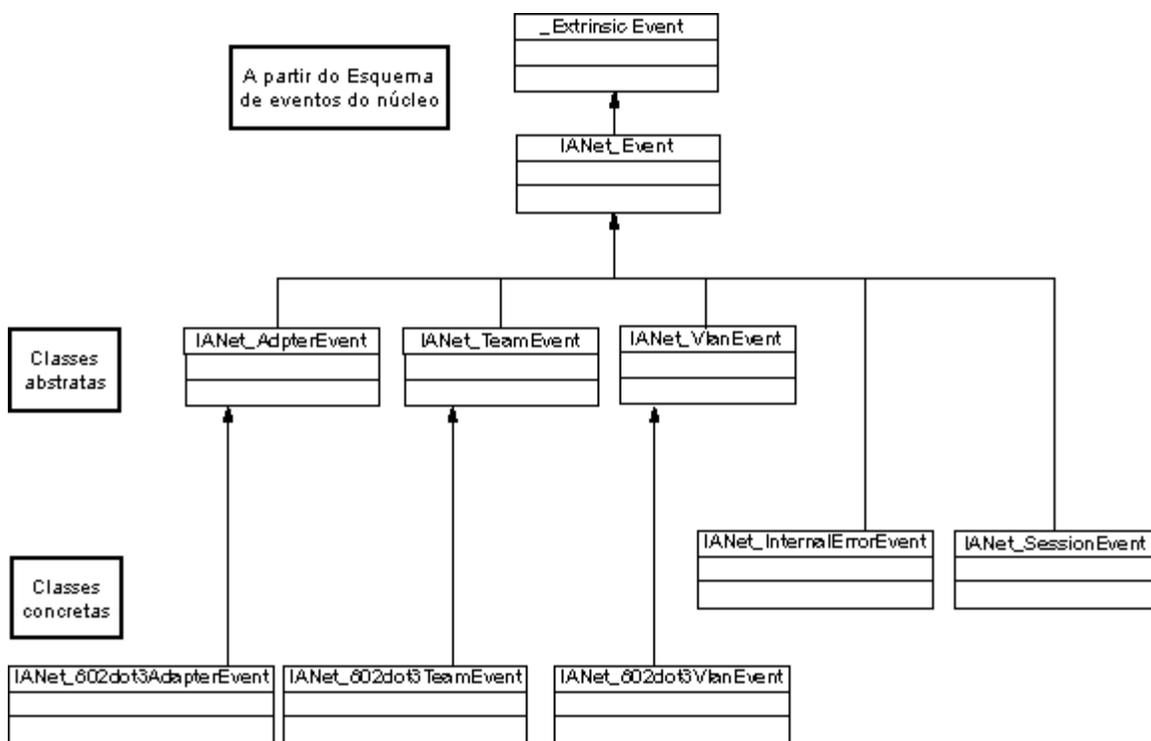
	strErrorMessage);		
Ler a imagem da ROM de inicialização	uint32 ReadFlash([OUT] uint8 FlashData[]);	FlashData[] contém a imagem de ROM Flash no NIC.	Esse método lê a ROM Flash no NIC que pode ser salva em arquivo.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Notificações de eventos: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Classes de eventos concretos](#)
[Registro para eventos](#)



Classes de eventos concretos

IANet_802dot3AdapterEvent

Objetivos

Esse evento informa o cliente sobre uma alteração no status ou na configuração de um adaptador.

Acionadores

O evento ocorre depois que o status de um adaptador muda ou depois que você altera a configuração de um adaptador e chama **Apply**.

Dados do evento

AdapterPath contém o caminho do objeto do adaptador que causou o evento.

IANet_802dot3TeamEvent

Objetivos

Esse evento informa o cliente sobre uma alteração no status ou na configuração de um grupo.

Acionadores

O evento ocorre:

- Depois que o status de um adaptador muda.
- Depois que você altera a configuração de um grupo e chama **Apply**.
- Depois que a configuração do grupo foi alterada e você chamou **Apply**.

Dados do evento

Team contém o caminho de objeto do grupo que causou o evento.

IANet_802dot3VlanEvent

Objetivos

Esse evento informa o cliente sobre uma alteração no status ou na configuração de uma VLAN.

Acionadores

O evento ocorre:

- Depois que o status de uma VLAN muda.
- Depois que você altera a configuração de uma VLAN e chama **Apply**.
- Depois que a configuração da VLAN foi alterada e você chamou **Apply**.

Dados do evento

VlanPath contém o caminho do objeto da VLAN que causou o evento.

[Voltar ao início](#)

Registro para eventos

Os aplicativos devem usar **IWbemServices:: ExecNotificationQuery** ou **IWbemServices:: ExecNotificationQueryAsync** para solicitar notificações para eventos. As consultas a seguir são exemplos de consultas para notificação de eventos. A lista não é completa, já que várias consultas são possíveis.

- **SELECT * FROM IANet_Event** — Utilizada para solicitar todos os eventos.
 - **SELECT * FROM IANet_AdapterEvent** — Utilizada para solicitar todos os eventos de adaptador.
 - **SELECT * FROM IANet_TeamEvent** — Utilizada para solicitar todos os eventos de grupo.
 - **SELECT * FROM IANet_SessionEvent** — Utilizada para solicitar todos os eventos de sessão.
 - **SELECT * FROM IANet_VlanEvent** — Utilizada para solicitar todos os eventos de VLAN.
 - **SELECT * FROM IANet_InternalErrorEvent** — Utilizada para solicitar todos os eventos internos
 - **SELECT * FROM IANet_AdapterEvent WHERE AdapterPath={caminho do objeto IANet_EthernetAdapter}** — Utilizada para solicitar os eventos de um determinado adaptador.
-

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Consultas WQL otimizadas: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Visão geral](#)

[Obtenção das configurações de um adaptador, VLAN ou grupo](#)

[Obtenção de uma configuração](#)

Visão geral

O provedor WMI é otimizado para permitir que os aplicativos obtenham configurações por meio de consultas. O provedor WMI reconhece as consultas apresentadas a seguir e só retorna os objetos que correspondem. Em todas as outras consultas, o provedor WMI obtém todas as configurações de todos os objetos e o CIMOM os filtra antes que cheguem ao aplicativo. Quando há vários adaptadores, grupos e VLANs, esse fato provoca um atraso de vários segundos ao recuperar os dados necessários.

[Voltar ao início](#)

Obtenção das configurações de um adaptador, VLAN ou grupo

A consulta a seguir obtém apenas as configurações de um determinado adaptador, VLAN ou grupo. O WQL não permite cláusulas adicionais na cláusula WHERE.

```
ASSOCIATORS OF {caminho de IANet_Configuration} WHERE AssocClass = IANet_SettingContext
```

[Voltar ao início](#)

Obtenção de uma configuração

A consulta a seguir pode ser usada para obter uma única configuração para um objeto sem fazer consultas para obter todos eles:

```
SELECT * FROM [SETTING CLASS] WHERE ParentId="[Device ID]" AND ParentType="[type]" AND Caption="[SETTING NAME]"
```

Notas:

- A classe deve ser exatamente a classe de configuração, não uma classe básica (por exemplo: IANet_SettingInt).
- Os ParentTypes aceitáveis são "NIC", "Team", "VLAN" ou "BootAgent".
- ParentId é o GUID que define de modo exclusivo o objeto que tem a configuração (no caso do adaptador, é a DeviceId).
- Esse método de obter as configurações associadas a um objeto não é recomendável: o método preferencial é o uso de associações. Entretanto, o WQL não suporta a consulta completa que é necessária (ou seja, não suporta ASSOCIATORS OF {caminho de IANet_Configuration} WHERE AssocClass = IANet_SettingContext AND Caption="[SETTING NAME]").

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Diagnósticos: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

[Classes de diagnósticos](#)

[Entradas de registro](#)

[Registro](#)

[Classes associativas](#)

[Testes](#)

Classes de diagnóstico

IANet_DiagTest

Objetivo

IANet_DiagTest é uma subclasse de CIM_DiagnosticTest. Essa classe proporciona um veículo genérico para executar e controlar testes de diagnóstico para um adaptador Ethernet com suporte para Intel® PROSet. A superclasse, CIM_DiagnosticTest, é projetada para suportar genericamente o teste de qualquer tipo de hardware em um sistema habilitado para CIM. As propriedades da classe são de natureza descritiva e a mecânica do teste é fornecida pelos métodos expostos.

Instâncias

A chave é Name (Nome) e, nesse provedor, é a concatenação de um índice numérico do teste no GUID do adaptador referenciado (exemplo: 1@{12345678-9ABC-DEF0-1234-123456789012}). Esse valor de chave é, de certa forma, uma informação redundante, já que todas as informações para referenciar um adaptador e testar são passadas como parâmetros de objeto para RunTest e outros métodos. Mesmo assim, a instância deve ser consistente com os parâmetros do método; caso contrário, o provedor não aceita o comando. A propriedade de legenda (caption) dá o nome do teste que a instância irá executar. Outras propriedades fornecem outras descrições e informações de tempo de execução.

Criação de instâncias

Não é possível criar instâncias de IANet_DiagTest.

Exclusão de instâncias

Não é possível excluir instâncias de IANet_DiagTest.

Modificação de propriedades

Não há propriedades modificáveis pelo usuário para essa classe.

Associação

- Uma instância de IANet_DiagTestForMSE associa uma IANet_DiagTest com uma IANet_ManagedSystemElement. A IANet_ManagedSystemElement será uma instância IANet_EthernetAdapter.
- Uma instância de IANet_DiagResultForTest associa uma IANet_DiagTest com uma instância de IANet_DiagnosticResult.
- Uma instância de IANetDiagSettingForTest associa uma IANet_DiagTest com uma IANet_DiagSetting.

Propriedades sem suporte

Install Date, OtherCharacteristicDescription

Métodos

Essa classe suporta os seguintes métodos:

- RunTest — Executa um teste definido com por três parâmetros fazendo referência a:
 - SystemElement — define o adaptador no qual o teste será executado, referenciando uma instância de SystemElement, que sempre será a subclasse IANet_EthernetAdapter.
 - Setting — define o teste a ser executado e o modo como será executado, referindo-se a uma instância de CIM_DiagnosticSetting, que sempre será a subclasse IANet_DiagSetting.
 - DiagnosticResult — define uma instância da classe CIM_DiagnosticResult, que sempre será a classe

IANet_DiagResult.

- DiscontinueTest — Tenta parar um teste de diagnóstico em andamento, segundo a definição de dois parâmetros referenciando CIM_ManagedSystemElement e CIM_DiagnosticResult. Esses parâmetros funcionam como o RunTest. Um terceiro parâmetro, TestingStopped, retorna um valor BOOLEANO, que indica se o comando conseguiu interromper o teste.
- ClearResults — Limpa os resultados do teste usando os seguintes parâmetros:
 - SystemElement
 - ResultsNotCleared

O parâmetro referenciado ManagedSystemElement, combinado com o caminho desse objeto, combina-se para referenciar instâncias de DiagnosticResultForMSE, que serão excluídas. Além disso, todas as referências a objetos de DiagnosticResult referenciados por DiagnosticResultForMSE serão excluídas. Também serão excluídas todas as instâncias de DiagnosticResultForTest, que se referem aos objetos de DiagnosticResult que foram excluídos. Por fim, o parâmetro de Output (Saída) de vetor de string, ResultsNotCleared, lista as chaves de DiagnosticResults que não foi possível limpar.

Hierarquia de classe

Para CimV2. Propriedades que não foram usadas e métodos não exibidos.

- CIM_ManagedElement:
 - Caption
 - Description
- CIM_ManagedSystemElement:
 - Install Date
 - Name
 - Status
- CIM_LogicalElement
- CIM_Service:
 - Chave
 - Name (string)
 - Propriedades
 - Caption (string)
 - CreationClassName (string)
 - Description (string)
 - Started (booleano)
 - StartMode (string)
 - Status (string)
 - SystemCreationClass (string)
 - SystemName (string)
- CIM_DiagnosticTest:
 - Propriedades
 - Características (matriz uint16)
 - IsInUse (Booleano)
 - ResourcesUsed (matriz uint16)
 - Métodos
 - RunTest
 - ClearResults
 - DiscontinueTest

Execução de RunTest e outros métodos em WbemTest

O método RunTest do arquivo MOF é o seguinte:

```
uint32
RunTest([IN] CIM_ManagedSystemElement ref SystemElement,
[IN] CIM_DiagnosticSetting ref Setting,
[OUT] CIM_DiagnosticResult ref Result);
```

Os primeiros dois parâmetros são parâmetros In. É necessário obter o caminho de objeto de ambos os objetos referenciados. Também é necessário obter o caminho de objeto de IANet_DiagTest, que está exportando o objeto RunTest.

1. Na caixa de diálogo principal de teste do WBEM, clique em **Connect**.
2. Insira o Server (Servidor)/Namespace (Espaço de Nomes) correspondentes. Há suporte para os espaços de nome IntelNCS e CimV2.

3. Clique no botão **Enum Instances** do teste do WBEM e digite `IANet_DiagTest`.
4. Clique duas vezes na instância do `IANet_DiagTest`. O nome terá a forma `X@[AdapterGUID]`, onde X é o nome do teste e AdapterGUID será o nome do adaptador, igual ao da chave Name (Nome) do `IANet_EthernetAdapter`.
5. Veja a seguir um exemplo do caminho do objeto recuperado:
`\\MYCOMPUTER\root\Cimv2:IANet_DiagTest.Name="1@{4A0CDABE-F6C3-45D0-B60D-F6E7BAFA2C2C}"`
6. Salve o caminho do objeto.
7. Clique no botão **Enum Instances** do teste do WBEM e digite `IANet_EthernetAdapter`.
8. Clique duas vezes no adaptador a ser testado.
9. Veja a seguir um exemplo do caminho do objeto recuperado:
`\\MYCOMPUTER\root\Cimv2:IANet_EthernetAdapter.DeviceID="{4A0CDABE-F6C3-45D0-B60D-F6E7BAFA2C2C}"`
10. Salve o caminho do objeto.
11. Clique no botão **Enum Instances** do teste do WBEM e digite `IANet_DiagSetting`.
12. Clique duas vezes na configuração que representa a combinação adaptador/teste.
13. Veja a seguir um exemplo do caminho do objeto recuperado:
`\\MYCOMPUTER\root\Cimv2:IANet_DiagSetting.SettingID="1@{4A0CDABE-F6C3-45D0-B60D-F6E7BAFA2C2C}"`
14. Salve o caminho do objeto.
15. Na caixa de diálogo principal de teste do WBEM, clique em **Execute Method**.
16. Cole o caminho do objeto `IANet_DiagTest` na caixa de diálogo. Clique em **OK**.
17. Selecione o teste na caixa suspensa que está sob o método.
18. Clique no botão **Edit In Parameters**.
19. Para `RunTest`, os parâmetros In são `Setting` e `SystemElement`. Cole os caminhos de objeto de `Setting` (Configuração) e `Adapter` (Adaptador), que você salvou anteriormente e feche.
20. Clique no botão **Execute**.
21. Enumere a classe `IANet_DiagResult` da mesma forma que enumerou os parâmetros In.
22. Analise o objeto de resultado selecionado conforme o necessário.

IANet_DiagSetting

Objetivo

Instâncias de `IANet_DiagSetting` fornecem diretivas específicas de teste de diagnóstico em tempo de execução. Todos os testes usam as mesmas diretivas, e elas estão vinculadas à superclasse `CIM_DiagnosticSetting`. Envolvem propriedades, como `ReportSoftErrors` e `HaltOnError`. Não há propriedades adicionais acrescentadas à subclasse `IANet_DiagSetting`.

Criação de instâncias

Não é possível criar instâncias dessa classe.

Exclusão de instâncias

Não é possível excluir instâncias dessa classe.

Modificação de propriedades

`UpdateInstanceAsync` é implementado e pode ser usado para configurar os parâmetros de teste como "Halt On Error", "Report Soft Errors", "Report Status Messages", "Quick Mode", "Test Warning Level" e "Percent Of Test Coverage".

Associações

Uma instância de `IANetDiagSettingForTest` associa uma `IANet_DiagTest` com um `IANet_DiagSetting`.

Propriedades sem suporte

O NCS não presta suporte às seguintes propriedades:

- Caption
- Descrição

Métodos

Nenhum

Hierarquia de classe

Para CimV2. Propriedades que não foram usadas e métodos não exibidos.

- `CIM_ManagedElement`
- `CIM_Setting`:

- o Propriedades
- o SettingID
- o Métodos (não há suporte para métodos)
- o VerifyOKToApplyToMSE
- o ApplyToMSE
- o VerifyOKToApplyToCollection
- o ApplyToCollection
- o VerifyOKToApplyIncrementalChangeToMSE
- o ApplyIncrementalChangesToMSE
- o ApplyIncrementalChangeToCollection
- CIM_DiagnosticSetting:
 - o Chave
 - SettingID (string)
 - o Propriedades
 - TestWarningLevel (uint16)
 - ReportSoftErrors (Booleano)
 - ReportStatusMessages (Booleano)
 - HaltOnError (Booleano)
 - QuickMode (Booleano)
 - PercentOfTestCoverage (uint8)

IANet_DiagResult

Objetivo

Instâncias de IANet_DiagResult exibem os dados de resultado de um determinado teste, executado em um adaptador determinado. As instâncias dessa classe são idênticas às instâncias de IANet_DiagTest e IANet_DiagSetting.

Instâncias

Instâncias de IANet_DiagResult correspondem aos resultados de um determinado teste, executado em um adaptador específico. O formato da chave é o mesmo de IANet_DiagTest e IANet_DiagSetting. A instância pode armazenar qualquer resultado de teste arbitrário como qualquer tipo de dados. Os que não se enquadram nas propriedades definidas, podem ser colocados na propriedade TestResults Array. A qualquer momento que um novo teste é executado no adaptador, a nova instância sobrescreve a instância de resultados de teste já existente que corresponde à combinação de adaptador e teste.

Criação de instâncias

Não é possível criar instâncias dessa classe.

Exclusão de instâncias

Não é possível excluir instâncias dessa classe.

Modificação de propriedades

Não é possível modificar instâncias dessa classe.

Associações

Uma instância de IANet_DiagResultForTest associa uma IANet_DiagTest com uma instância de IANet_DiagnosticResult.

Propriedades sem suporte

O NCS não presta suporte às seguintes propriedades:

- EstimatedTimeOfPerforming
- HaltOnError
- OtherStateDescription
- ReportSoftErrors
- TestWarningLevel

Métodos

Nenhum

Hierarquia de classe

Para CimV2. Propriedades que não foram usadas e métodos não exibidos.

- CIM_DiagnosticResult:
 - Chaves
 - DiagnosticCreationClassName (string)
 - DiagnosticName (string)
 - ExecutionID (string)
 - DiagSystemCreationClassName (string)
 - DiagSystemName (string)
 - Propriedades
 - TimeStamp (string)
 - IsPackage (Booleano)
 - TestStartTime (data/hora)
 - TestCompletionTime (data/hora)
 - TestState (uint16)
 - TestResults (string)
 - PercentComplete (uint8)
- IANet_DiagResult

[Voltar ao início](#)

Entradas de registro

As entradas a seguir são inseridas do registro (registry) durante a instalação, em:

HKLM\Software\Intel\NETWORK_SERVICES\NCS\NcsDiag. Essas chaves e valores controlam a execução dos testes de diagnóstico e são definidos abaixo.

A tabela a seguir contém os valores da chave, o tipo e uma breve explicação sobre o uso:

Valor	Tipo	Padrão	Uso
Check Time	REG_DWORD	2 seconds	Duração do período entre as verificações, para fazer o teste do emissor ou respondedor.
Enable	REG_DWORD	0	Habilita (1) ou Desabilita (0) o uso de um arquivo de registro de resultados.
FileAppend	REG_DWORD	1	Anexa (1) o arquivo de registro de resultados arquivo de registro de resultados já existente. Se 0, exclui o arquivo existente.
LogFileName	REG_SZ	NcsDiag.log	O nome do arquivo de registro de resultados.
MaxFileSize	REG_DWORD	0x10000	O tamanho máximo do arquivo de registro de resultados.
MaxPktsRcvd	REG_DWORD	200	No Quick Mode (de IANet_DiagSetting), o teste de emissão/recepção termina quando o número de pacotes recebidos é maior que esse valor.
TimeoutSndRsp	REG_DWORD	100	O teste termina quando a duração dele (em segundos) exceder esse valor.

[Voltar ao início](#)

Registro

Registro de resultados

O Registro de Resultados mostra principalmente as informações, que também podem ser obtidas de objetos IANet_DiagResult. A diferença é que as informações obtidas de um navegador CIM só mostram os últimos resultados imediatos de determinados testes em um determinado adaptador. Os testes seguintes sobrescrevem os resultados de testes anteriores. O Log de resultados é mais conveniente para configurar e visualizar várias execuções desse teste.

Habilitação do log de resultados

Para habilitar o log de resultados:

1. Na chave de registro **HKLM\Software\Intel\NETWORK_SERVICES\NCS\NCS Diag**, configure o valor **Enable** como 1.
2. Configure o valor **LogFileName** como o nome de arquivo preferencial para o registro ou use o padrão **NcsDiag.log**.
3. O arquivo de registro ficará no diretório indicado pelo valor **InstalledDir**.

[Voltar ao início](#)

Classes associativas

Classe associativa	Propriedade de referência/valor	Propriedade de referência/valor
IANet_DiagTestForMSE	Antecedent = IANet_DiagTest	Dependent = IANet_EthernetAdapter
IANet_DiagResultForTest	DiagnosticResult = IANet_DiagResult	DiagnosticTest = IANet_DiagTest
IANet_DiagSettingForTest	Element = IANet_DiagTest	Setting = IANet_DiagSetting
IANet_DiagResultForTest	DiagnosticResult = IANetDiagResult	DiagnosticTest = IANet_DiagTest

[Voltar ao início](#)

Testes

Os testes implementados podem ser executados em uma ou em duas máquinas. Este documento não se destina a dar explicações detalhadas sobre os testes, já que o CDM Provider se destina a ser um veículo genérico de teste, independente das particularidades dos testes. Entretanto, o código gera uma certa dependência e isso será explicado nesta seção.

Testes em adaptadores únicos

Os testes a seguir são executados em um único adaptador e não precisam interagir com nenhum outro adaptador:

- EEPROM
- Registros de controle
- Loopback de MAC
- Loopback de PHY
- Link

Todas as mensagens de erro desses testes são resultado de códigos de erro HRESULT retornados de chamadas a camadas mais baixas da pilha. Os códigos de erro são armazenados internamente como códigos de erro e não traduzidos em mensagens de erro até que os objetos IANet_DiagResult sejam "desreferenciados" por enumeração ou sejam recebidas chamadas de objetos de uma aplicação de gerenciamento.

Testes que requerem dois adaptadores

Os testes de emissor e respondedor são dependentes entre si. Um adaptador (o emissor) envia pacotes a outro adaptador (o respondedor) e este envia pacotes ao emissor, fechando o circuito. Esses testes são os mesmos que podem ser executados no Intel® PROSet. Porém, o Intel PROSet não usa CDM e não permite executar dois testes na mesma máquina ao mesmo tempo. O CDM permite que testes diferentes sejam executados simultaneamente na mesma máquina.

Testes de emissor/respondedor

Emissor/respondedor exige dois adaptadores Intel: uma para ser o emissor e um para ser o respondedor. Esse teste é o único a ser executado baseando-se em um segundo thread que continua executando até que o teste termine de acordo com os critérios de término ou seja interrompido pelo thread primário. Os critérios de término são tempos de espera baseados no período de duração do teste ou no número de pacotes recebidos. Esses dois valores são obtidos no registro. O teste só poderá ser terminado com base no número de pacotes recebidos se o modo Quick for ativado. Quick Mode é uma propriedade da classe

IANet_DiagSetting e, portanto, pode ser configurado adaptador por adaptador. Os respondedores CDM responderão a dispositivos PROSet e vice-versa.

Os testes de emissor/respondedor retornam dois tipos de valores de erro. Primeiramente, as camadas inferiores podem retornar um código de erro (HRESULT). Em segundo lugar, enquanto o teste está sendo executado (e se não for terminado prematuramente pelo retorno de um código de erro), o thread do teste retornará as estatísticas parciais e, em seguida, finais do teste, que conterão o seguinte:

- Link Status (Status do Link)
- Using Auto-Negotiation (Uso da Negociação Automática)
- Collisions (Colisões)
- Packets Received (Pacotes recebidos)
- Packets Received Total (Total de Pacotes Recebidos)
- Packets Sent (Pacotes enviados)
- Transmit Oks (Transmissão Ok)
- Receive Oks (Recepção Ok)
- Transmit Errors (Erros de Transmissão)
- Receive Errors (Erros de recepção)
- Collisions (Colisões)
- Diagnostic Phase (Fase do diagnóstico)

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

Execução de métodos em IANet_DiagTest: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

Execução de RunTest e outros métodos em WbemTest

O método RunTest do arquivo MOF é o seguinte:

```
uint32
RunTest([IN] CIM_ManagedSystemElement ref SystemElement,
[IN] CIM_DiagnosticSetting ref Setting,
[OUT] CIM_DiagnosticResult ref Result);
```

Os primeiros dois parâmetros são parâmetros In. É necessário obter o caminho de objeto de ambos os objetos referenciados. Também é necessário obter o caminho de objeto de IANet_DiagTest, que está exportando o objeto RunTest.

1. Na caixa de diálogo principal de teste do WBEM, clique em **Connect**.
2. Insira o Server (Servidor)/Namespace (Espaço de Nomes) correspondentes. Há suporte para os espaços de nome IntelNCS e CimV2.
3. Clique no botão **Enum Instances** do teste do WBEM e digite IANet_DiagTest.
4. Clique duas vezes na instância do IANet_DiagTest. O nome terá a forma X@[AdapterGUID], onde X é o nome do teste e AdapterGUID será o nome do adaptador, igual ao da chave Name (Nome) do IANet_EthernetAdapter.
5. Veja a seguir um exe4mplo do caminho do objeto recuperado:
\\MYCOMPUTER\root\Cimv2:IANet_DiagTest.Name="1@{4A0CDABE-F6C3-45D0-B60D-F6E7BAFA2C2C}"
6. Salve o caminho do objeto.
7. Clique no botão **Enum Instances** do teste do WBEM e digite IANet_EthernetAdapter.
8. Clique duas vezes no adaptador a ser testado.
9. Veja a seguir um exe4mplo do caminho do objeto recuperado:
\\MYCOMPUTER\root\cimv2:IANet_EthernetAdapter.DeviceID="{4A0CDABE-F6C3-45D0-B60D-F6E7BAFA2C2C}"
10. Salve o caminho do objeto.
11. Clique no botão **Enum Instances** do teste do WBEM e digite IANet_DiagSetting.
12. Clique duas vezes na configuração que representa a combinação adaptador/teste.
13. Veja a seguir um exe4mplo do caminho do objeto recuperado:
\\MYCOMPUTER\root\cimv2:IANet_DiagSetting.SettingID="1@{4A0CDABE-F6C3-45D0-B60D-F6E7BAFA2C2C}"
14. Salve o caminho do objeto.
15. Na caixa de diálogo principal de teste do WBEM, clique em **Execute Method**.
16. Cole o caminho do objeto IANet_DiagTest na caixa de diálogo. Clique em **OK**.
17. Selecione o teste na caixa suspensa que está sob o método.
18. Clique no botão **Edit In Parameters**.
19. Para RunTest, os parâmetros In são Setting e SystemElement. Cole os caminhos de objeto de Setting (Configuração) e Adapter (Adaptador), que você salvou anteriormente e feche.
20. Clique no botão **Execute**.
21. Enumere a classe IANet_DiagResult da mesma forma que enumerou os parâmetros In.
22. Analise o objeto de resultado selecionado conforme o necessário.

Leia todas as [restrições e isenções de responsabilidade](#).

Resumo das classes CIM: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

Classe	Pode criar?	Pode excluir?	Métodos implementados	Propriedades configuráveis	Propriedades sem suporte	Contagem de instâncias	Classes de associação relacionadas
IANet_802dot1QVLANService	N	N	CreateVLAN	GRVPEEnabled, JoinTime	Description, Install Date, Started, StartMode, Status	Uma instância para cada grupo ou adaptador que presta suporte a VLANs.	IANet_Device802dot1QVLANServiceImplementation, IANet_VLANFou
IANet_AdapterConfiguration	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada adaptador	Essa classe associa IANet_EthernetAdapter com IANet_Configuration.
IANet_AdapterDevice	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada adaptador que não seja fantasma	Essa classe associa IANet_EthernetAdapter com IANet_EthernetPCIDevice.
IANet_AdapterProtocolImplementation	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada ponto de extremidade de protocolo IP vinculado a um adaptador	Essa classe associa IANet_EthernetAdapter com IANet_IPProtocolEndpoint.
IANet_BootAgent	N	N	ProgramFlash ReadFlash	Nenhum	Caption, Description, InstallDate, Started, StartMode, Status	Uma instância para cada adaptador que suporta o recurso de boot agent	IANet_DeviceBootServiceImplementation, IANet_BootAgentConfiguration.
IANet_BootAgentConfiguration	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada boot agent	Essa classe associa IANet_BootAgent com IANet_Configuration.
IANet_Configuration	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada adaptador, VLAN e grupo	IANet_AdapterConfiguration, IANet_VLANConfiguration, IANet_SettingContext.
IANet_Device802dot1QVLANServiceImplementation	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada adaptador ou grupo que presta suporte a VLANs	Essa classe associa IANet_EthernetAdapter com IANet_802dot1QVLANService.
IANet_DiagTest	N	N	RunTest, DiscontinueTest, ClearResults	Nenhum	InstallDate, OtherCharacteristicsDescription	Um para cada combinação de adaptador/teste	IANet_DiagTestForMSE, IANet_DiagResultForTest, IANet_DiagSettingForTest
IANet_DiagSetting	N	N	Nenhum	HaltOnError, ReportSoftErrors, ReportStatusMessages, QuickMode, PercentOfTestCoverage, TestWarningLevel	Caption, Description	Um para cada combinação de adaptador/teste	IANet_DiagSettingForTest
IANet_DiagResult	N	N	Nenhum	Nenhum	EstimatedTimeOfPerforming,	Um para cada	IANet_DiagResultForTest,

					HaltOnError, OtherStateDescription, ReportSoftErrors, TestWarningLevel	combinação de adaptador/teste	IANet_DiagResultForMSEM
IANet_EthernetAdapter	N	S	IdentifyAdapter HasVLANs IsPowerMgmtSupported GetPowerUsage SetPowerUsage GetPowerUsageOptions SetPowerUsageOptions TestCable AdvancedTestCable TestLinkSpeed	Nenhum	AutoSense – (é exibido como uma configuração), ErrorCleared, OtherIdentifyingInfo, IdentifyingDescriptions, InstallDate, LastErrorCode, MaxDataSize, MaxQuiesceTime, PowerManagementCapabilities– (é exibido como um método), PowerManagementSupported– (é exibido como um método), PowerOnHours , ShortFramesReceived, SymbolErrors, TotalPowerOnHours	Um para cada adaptador Intel® PROSet instalado e suportado, adaptador e grupo fantasma.	IANet_AdapterProtocolImplementation, IANet_AdapterDevice, IANet_AdapterConfiguration, IANet_TeamedMemberAdapter, IANet_NetworkVirtualAdapter, IANet_Device802dot1QVLANServiceImplementation, IANet_DeviceBootServiceImplementation
IANet_EthernetPCIDevice	N	N	Nenhum	Nenhum	AdditionalAvailability, Capabilities, CapabilityDescriptions, Caption, Description, DeviceSelectTiming, ErrorCleared, ErrorDescription, IdentifyingDescription, InstallDate, LastErrorCode, MaxNumberController, MaxQuiesceTime, Name, OtherIdentifyingInfo, PowerManagementCapabilities, PowerManagementSupported, PowerOnHours, ProtocolDescription, ProtocolSupported, SelfTestEnabled, Status, StatusInfo, TimeOfLastReset, TotalPowerOnHours	Uma instância para cada placa PCI que seja um adaptador Ethernet com suporte do Intel PROSet	IANet_AdapterDevice
IANet_IPProtocolEndpoint	N	N	Nenhum	Nenhum	Caption, Description, InstallDate, NameFormat, OtherTypeInfoInformation, Status	Uma instância para cada vinculação da pilha de protocolo IP a um ponto de extremidade com suporte da Intel	IANet_AdapterProtocolImplementation, IANet_VLANProtocolDependency
IANet_NetService	N	N	GetSessionHandle, Apply, ReleaseSessionHandle, Cancel	Nenhum	Caption, Description, Install Date, Started, Start Mode, Status	Exatamente um.	Nenhum
IANet_NetworkVirtualAdapter	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada grupo.	Essa classe associa IANet_TeamOfAdapters com an IANet_EthernetAdapter.
IANet_PCIDevice	N	N	Nenhum	Nenhum	AdditionalAvailability, Capabilities, CapabilityDescriptions, Caption, DeviceSelectTiming, ErrorCleared, ErrorDescription, IdentifyingDescription, InstallDate, LastErrorCode, MaxNumberController,	Uma instância para cada placa PCI que é um dispositivo de rede no sistema	

					MaxQuiesceTime, Name, OtherIdentifyingInfo, PowerManagementCapabilities, PowerManagementSupported, PowerOnHours, ProtocolDescription, ProtocolSupported, SelfTestEnabled, TimeOfLastReset, TotalPowerOnHours		
IANet_Setting	N	N	Nenhum	Nenhum	SettingID	Classe abstrata	IANet_SettingContext
IANet_SettingContext	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada configuração	Essa classe associa IANet_Setting com an IANet_Configuration.
IANet_SettingInt	N	N	Nenhum	CurrentValue	SettingID	Uma instância para cada configuração de número inteiro	IANet_SettingContext
IANet_SettingMultiSelection	N	N	Nenhum	CurrentValue	SettingID	Uma instância para cada configuração de seleções múltiplas	IANet_SettingContext
IANet_SettingSlider	N	N	Nenhum	CurrentValue	SettingID	Uma instância para cada configuração da barra deslizante	IANet_SettingContext
IANet_SettingString	N	N	Nenhum	CurrentValue	SettingID	Uma instância para cada configuração de string	IANet_SettingContext
IANet_TeamedMemberAdapter	S	S	Nenhum	AdapterFunction	PrimaryAdapter, ScopeOfBalancing	Uma instância para cada adaptador que faz parte de um grupo	Essa classe associa IANet_TeamOfAdapters com an IANet_EthernetAdapter.
IANet_TeamOfAdapters	S	S	TestSwitchConfiguration	TeamingMode	Install Date, Status	Uma instância para cada grupo	IANet_NetworkVirtualAdapter, IANet_TeamedMemberAdapter
IANet_VLAN	N	S	Nenhum	VLANNumber, Caption	Description, Install Date, StartMode, Status	Uma instância para cada VLAN	IANet_VLANFou
IANet_VLANConfiguration	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada VLAN	Essa classe associa IANet_VLAN com IANet_Configuration
IANet_VLANFou	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada VLAN	Essa classe associa IANet_VLAN com IANet_802dot1QVLANSservice.
IANet_VLANProtocolDependency	N	N	Nenhum	Nenhum	Nenhum	Uma instância para cada VLAN.	Essa classe associa IANet_VLAN com IANet_IPProtocolEndpoint.

Contrato de licença de software Intel (Final, Licença): Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

IMPORTANTE - LEIA ANTES DE COPIAR, INSTALAR OU USAR

Não use ou carregue este software e qualquer outro material associado (coletivamente, o “Software”) antes de ler cuidadosamente os termos e condições a seguir. Ao carregar ou usar o Software, você automaticamente concorda com os termos deste Contrato. Se não concordar, não instale nem use o Software.

LICENÇAS:

- A “Licença de site” a seguir se aplica aos administradores de rede.
- A “Licença para usuário único” se aplica aos usuários finais.
- A “Licença para fabricantes de equipamentos originais (OEM)” se aplica a tais fabricantes.

Licença de Site. Você poderá copiar o Software nos computadores da sua empresa, para uso na sua empresa, e poderá fazer cópias de backup do Software, de acordo com as seguintes condições:

1. Este Software é licenciado somente para uso com produtos que tenham componentes Intel. O uso do Software com produtos que não sejam da Intel não é aqui licenciado.
2. Você não pode copiar, modificar, alugar, vender, distribuir ou transferir qualquer parte do Software, exceto na forma expressa neste Contrato, e você concorda em impedir a cópia não autorizada do Software.
3. Você não pode fazer engenharia reversa, descompilação ou desmontagem do Software.
4. Você não pode sublicenciar ou permitir o uso simultâneo do Software por mais de um usuário.
5. O Software pode ter componentes oferecidos sob termos adicionais aos aqui contidos, descritos em um contrato de licença que acompanha estes componentes.

Licença para usuário único. Você poderá fazer uma cópia do Software em um único computador para o seu uso pessoal, e não comercial, e poderá criar uma cópia de backup do Software, sujeita às seguintes condições:

1. Este Software é licenciado somente para uso com produtos que tenham componentes Intel. O uso do Software com produtos que não sejam da Intel não é aqui licenciado.
2. Você não pode copiar, modificar, alugar, vender, distribuir ou transferir qualquer parte do Software, exceto na forma expressa neste Contrato, e você concorda em impedir a cópia não autorizada do Software.
3. Você não pode fazer engenharia reversa, descompilação ou desmontagem do Software.
4. Você não pode sublicenciar ou permitir o uso simultâneo do Software por mais de um usuário.
5. O Software pode ter componentes oferecidos sob termos adicionais aos aqui contidos, descritos em um contrato de licença que acompanha estes componentes.

Licença para OEM. Você pode reproduzir e distribuir o software somente como parte integrante do seu Produto ou integrado a ele, ou como uma atualização independente de software para os usuários finais do seu Produto, excluindo-se quaisquer outros produtos independentes, sujeito às seguintes condições:

1. Este Software é licenciado somente para uso com produtos que tenham componentes Intel. O uso do Software com produtos que não sejam da Intel não é aqui licenciado.
2. Você não pode copiar, modificar, alugar, vender, distribuir ou transferir qualquer parte do Software, exceto na forma expressa neste Contrato, e você concorda em impedir a cópia não autorizada do Software.
3. Você não pode fazer engenharia reversa, descompilação ou desmontagem do Software.
4. A distribuição do Software aos seus clientes só pode ser feita mediante um contrato de licença por escrito. Esse contrato pode ser baseado no contrato de licença da “violação do lacre”. No mínimo, deve proteger os direitos de propriedade da Intel em relação ao Software.
5. O Software pode ter componentes oferecidos sob termos adicionais aos aqui contidos, descritos em um contrato de licença que acompanha estes componentes.

NENHUM OUTRO DIREITO. A Intel não lhe outorga direitos ou licenças, implícitos ou explícitos, em relação a informações de

propriedade ou patentes, direitos autorais, trabalho de máscara, segredo comercial ou outro direito de propriedade intelectual que a Intel controle ou possua, excetuando-se o que se encontra expresso neste Contrato.

PROPRIEDADE DO SOFTWARE E COPYRIGHTS. O título de todas as cópias do Software permanece com a Intel ou seus fornecedores. O Software contém direitos autorais e está protegido pelas leis dos Estados Unidos e de outros países e por provisões de tratados internacionais. Você não pode remover quaisquer avisos de direitos autorais do Software. A Intel pode fazer alterações no Software ou nos itens aqui citados a qualquer momento sem aviso prévio, mas não está obrigada a oferecer suporte ou a atualizar o Software. Exceto quando expressamente declarado, a Intel não concede qualquer direito explícito ou implícito sobre patentes, direitos autorais, marcas comerciais ou outros direitos sobre a propriedade intelectual. O Software pode ser transferido somente se o destinatário concordar totalmente com esses termos e se você não reter nenhuma cópia do Software.

GARANTIA LIMITADA DA MÍDIA. Se o Software foi entregue pela Intel em forma de mídia física, a Intel oferece garantia desta mídia contra defeitos físicos do material, por um período de 90 dias após a entrega. No caso de tais defeitos serem encontrados, devolva a mídia à Intel para que seja feita a substituição ou a entrega do Software por meio alternativo, a critério da Intel.

EXCLUSÃO DE OUTRAS GARANTIAS. EXCETO PELAS GARANTIAS FORNECIDAS ACIMA, O SOFTWARE É FORNECIDO “NO ESTADO EM QUE SE ENCONTRA”, SEM QUALQUER GARANTIA EXPRESSA OU IMPLÍCITA DE QUALQUER TIPO, INCLUINDO AS GARANTIAS DE ADEQUAÇÃO AO MERCADO, AUSÊNCIA DE INFRAÇÃO OU ADEQUAÇÃO A UM PROPÓSITO EM PARTICULAR. A Intel não oferece garantia ou assume responsabilidade pela exatidão ou pela integridade de qualquer informação, texto, gráficos, links ou outros itens contidos no Software.

LIMITAÇÃO DE RESPONSABILIDADE. SOB NENHUMA CIRCUNSTÂNCIA A INTEL OU SEUS FORNECEDORES SE RESPONSABILIZARÁ POR QUALQUER DANO (INCLUINDO, MAS NÃO SE LIMITANDO A, PERDA DE LUCROS, INTERRUPTÃO DE NEGÓCIOS OU PERDA DE INFORMAÇÕES) RESULTANTE DA UTILIZAÇÃO OU INCAPACIDADE DE UTILIZAÇÃO DO SOFTWARE, MESMO QUANDO A INTEL TIVER SIDO AVISADA DA POSSIBILIDADE DE OCORRÊNCIA DE TAIS DANOS. ALGUMAS JURISDIÇÕES PROIBEM A EXCLUSÃO OU LIMITAÇÃO DA RESPONSABILIDADE PARA GARANTIAS IMPLÍCITAS OU PARA DANOS CONSEQÜENCIAIS OU INCIDENTAIS; PORTANTO, A LIMITAÇÃO ACIMA PODE NÃO SE APLICAR A VOCÊ. VOCÊ PODE TAMBÉM TER OUTROS DIREITOS LEGAIS QUE VARIAM DE JURISDIÇÃO PARA JURISDIÇÃO.

RESCISÃO DESSE CONTRATO. A Intel pode rescindir este Contrato a qualquer momento no caso de violação dos seus termos. Com a rescisão, você destruirá imediatamente o Software ou devolverá todas as suas cópias à Intel.

LEGISLAÇÃO APLICÁVEL. As reclamações originadas desse Contrato estarão subordinadas às leis da Califórnia, excluindo seus princípios de conflito com as leis e a Convenção sobre Contratos das Nações Unidas para a Venda de Produtos. Você não poderá exportar o Software em violação às leis e regulamentos aplicáveis à exportação. A Intel não está obrigada ao cumprimento de quaisquer outros contratos, a menos que se apresentem por escrito e estejam assinados por um representante autorizado da Intel.

DIREITOS RESTRITOS DO GOVERNO. O Software é fornecido com “DIREITOS RESTRITOS”. O uso, duplicação ou divulgação pelo Governo está sujeito às restrições estabelecidas em FAR52.227-14 e DFAR252.227-7013 et seq. ou às suas cláusulas substitutas. O uso do software pelo Governo constitui o reconhecimento dos direitos de propriedade da Intel aqui descritos. A Intel é o fabricante ou contratante.

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do Índice](#) [Voltar ao início](#)

[Voltar à página do índice](#)

Suporte: Guia do usuário de provedores WMI e CDM dos adaptadores de rede Intel® PRO

Páginas da Web e sites da Internet

<http://www.dell.com>

Técnicos do Suporte ao cliente

Se os procedimentos para a solução de problemas não forem eficazes, entre em contato com a Dell Computer Corporation para obter assistência técnica (consulte a seção “Ajuda” na documentação do seu sistema).

Antes de ligar...

Você precisa estar no computador com o software em execução e ter a documentação à mão.

O técnico poderá pedir o seguinte:

- Seu endereço e telefone
 - O nome e modelo do produto sobre o qual você está fazendo a consulta
 - O número de série e a etiqueta de serviço do produto
 - Os nomes e números das versões dos softwares que você está usando para operar o produto
 - O nome e o número da versão do seu sistema operacional
 - O tipo do computador (fabricante e modelo)
 - As placas de expansão ou de extensão do computador
 - A quantidade de memória do computador
-

Leia todas as [restrições e isenções de responsabilidade](#).

[Voltar à página do índice](#)