
WMI – Windows Management Instrumentation

Instrumentação e gerenciamento em
ambientes distribuídos

Motivação

- Custo muito alto para dar manutenção em uma rede distribuída de computadores sem uma maneira automática
 - Monitoração dos estados dos dispositivos
 - Medição das informações desses estados
 - Configuração dos dispositivos
-

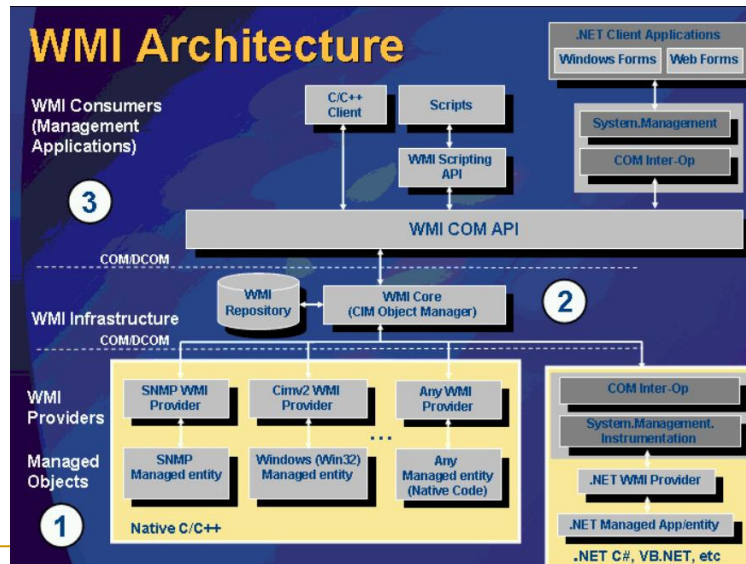
Início

- Criação de um padrão de monitoramento e instrumentação de dispositivos de forma distribuída
 - Criado pela Distributed Management Task Force (DMTF)
 - Padrão Web-Based Enterprise Management (WBEM)
 - Implementação para Windows pela Microsoft: Windows Management Instrumentation (WMI)

Características

- WMI não é uma plataforma genérica para desenvolver sistemas distribuídos
- Framework para gerenciamento de dispositivos distribuídos
 - Baseado em Monitoração e Instrumentação
 - Permite a integração de aplicações feitas por diferentes empresas
 - Implementação mais comum em COM / DCOM e .NET

Arquitetura WMI



Expondo dados via WMI

- Muitas aplicações e componentes (HW/SW) já possuem provedores (providers).
- É possível escrever o seu próprio provedor!
 - Diferentes técnicas nas tecnologias: C++ (COM) e **.NET (managed code providers)**.
- **.NET FCL: System.Management.Instrumentation** namespace
- Criar provedores de dados e de eventos.
- “Marcação” utilizando atributos.
- Instalar o assembly para publicar as classes no WMI.

Expondo Instâncias

- Usando o atributo **InstrumentationClassAttribute (InstrumentationType.Instance)**
 - Derivando de **Instance Class**
 - Gera um WMI schema da classe.
 - Desenvolvedores de código gerenciado não precisam de conhecimentos avançados para formatar dados no Common Information Model (CIM) schema.
-

Expondo Eventos

- Usando o atributo **InstrumentationClassAttribute(InstrumentationType.Event)**
 - **Derivando de BaseEvent (mais performático)**
 - Herança de eventos WMI
 - Forma de “gerenciamento passivo”
-

Consumindo dados

- Utilizando VBScript (ActiveX), C++ (COM), **.NET**.
- Conexão remota
- Subscrevendo eventos e coletando instâncias.
- Informações de componentes do sistema e das aplicações instrumentadas.
- O CIM Object Repository é dividido logicamente em *namespaces*.
- **System.Management** namespace contém classes para acessar e manipular informações gerenciáveis do WMI.
- **WQL** – WMI Query Language

Microsoft Windows

- Nativo:
 - Vista, 2K3, XP, ME e 2K
- NT 4.0:
 - SP4, necessita somente instalação.
- W95/98:
 - IE5, download no site: WMI CORE 1.5

Serviço do Windows

- WMI Performance Adapter
- Windows Management Instrumentation Driver Extensions
- Windows Management Instrumentation

Implementação do Provider

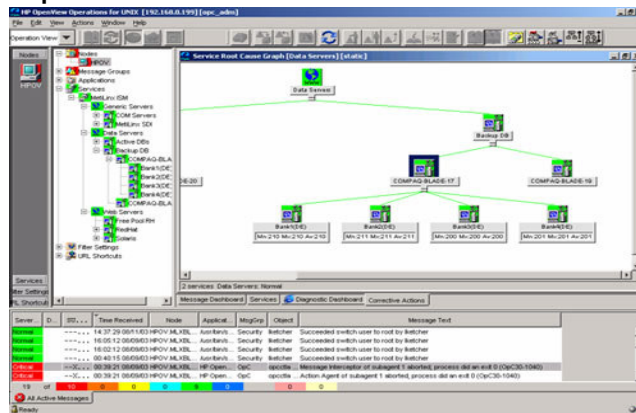
- Definição de Provider
- ATL Wizard do Visual Studio
- Definindo um COM contendo um provide
- Namespace
System.Management.Instrumentation

Aplicação

- Monitoramento
- Gerência
- Automação

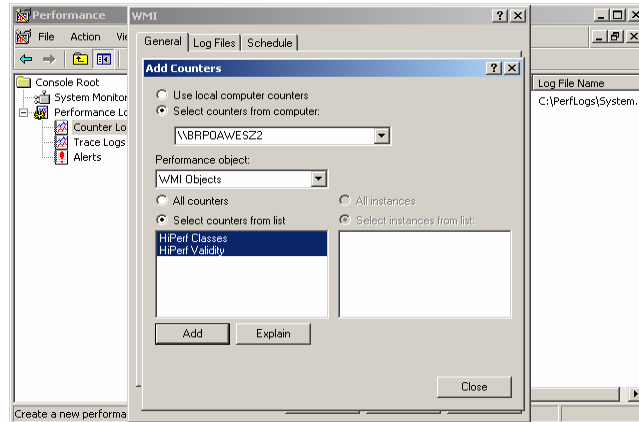
Ferramentas externas p/ WMI

- HP OpenView



Ferramentas externas p/ WMI

■ Perfmon



Conclusões

- Poderosa ferramenta de monitoração e instrumentação - sistemas distribuídos.
- Capacidade de expor informação para outras ferramentas.
- Baseado em um modelo padronizado e independente de linguagem utilizando paradigma de orientação a objetos.
- .NET WMI
 - Facilidades para instrumentação e monitoração.
 - Limitações .NET – Nova versão.
- **MSDN Architecture Webcast: .NET Application Instrumentation Using WMI.NET 2.0.**

Dúvidas?

Obrigado!