Programação de modelo multidimensional

O Analysis Services oferece várias APIs que você pode usar para programar em uma instância do Analysis Services e os bancos de dados multidimensionais que ele disponibiliza. Os projetos de desenvolvimento do Analysis Services podem se basear em código gerenciado ou não gerenciado, executado em uma plataforma Windows ou em outras plataformas que ofereçam suporte ao acesso HTTP.

# Arquitetura Microsoft OLAP

# Objetos do servidor (Analysis Services – Dados Multidimensionais)

O objeto [Server](https://technet.microsoft.com/pt-br/library/microsoft.analysisservices.server(v=sql.110).aspx) representa o servidor e a instância do MicrosoftSQL ServerAnalysis Services com a qual você deseja trabalhar.

* Todos os bancos de dados que você pode acessar, como uma coleção.
* Todas as propriedades de servidor definidas, como uma coleção.
* A cadeia de caracteres da conexão, as informações da conexão e o ID da sessão.
* O nome, edição e versão do produto.
* As coleções de funções.
* A coleção de rastreamentos.
* A coleção de assemblies.

# Visão geral da arquitetura lógica (Analysis Services – Dados Multidimensionais)

# O Analysis Services funciona em um modo de implantação de servidor que determina a arquitetura de memória e o ambiente de tempo de execução usados pelos diferentes tipos de modelos do Analysis Services. O modo de servidor é determinado durante a instalação. O modo Multidimensional e de Mineração de Dados oferece suporte ao OLAP e à mineração de dados tradicionais. O Modo de tabela oferece suporte a modelos de tabela. O modo integrado do SharePoint refere-se a uma instância do Analysis Services que foi instalada como PowerPivot para SharePoint, usado para carregar e consultar modelos de dados PowerPivot ou do Excel em uma pasta de trabalho.

# [Arquitetura básica](javascript:void(0))

# Uma instância do Analysis Services pode conter vários bancos de dados, e um banco de dados pode ter objetos OLAP e objetos de mineração de dados simultaneamente. Os aplicativos se conectam a uma instância especifica do Analysis Services e a um banco de dados especifico. Um computador servidor pode servir de host de várias instâncias do Analysis Services. Instâncias do Analysis Services são nomeadas como "<Nome\_do\_servidor>\<Nome\_da\_instância>". A ilustração a seguir mostra todas as relações mencionadas entre objetos Analysis Services.

# Relações de objetos de execução AMO

As classes básicas são o conjunto mínimo de objetos necessários para criar um cubo. Esse conjunto mínimo de objetos é uma dimensão, um grupo de medidas e uma partição. Uma agregação é opcional.

As dimensões são criadas a partir de atributos e hierarquias. As hierarquias são formadas por um conjunto ordenado de atributos, sendo que cada atributo do conjunto corresponde a um nível na hierarquia.

Os cubos são criados a partir de dimensões e grupos de medidas. As dimensões da coleção de dimensões de um cubo pertencem à coleção de dimensões do banco de dados.

Os grupos de medidas são coleções de medidas que têm o mesmo ponto de vista da fonte de dados e o mesmo subconjunto de dimensões do cubo.

Um grupo de medidas tem uma ou mais partições para gerenciar os dados físicos. Um grupo de medidas pode ter um projeto de agregação padrão. O projeto de agregação padrão pode ser usado por todas as partições no grupo de medidas, além disso, cada partição pode ter seu próprio projeto de agregação.

**Objetos do servidor**

Cada instância do Analysis Services é vista como um objeto de servidor diferente no AMO; cada instância diferente está conectada a um objeto [Server](https://technet.microsoft.com/pt-br/library/microsoft.analysisservices.server(v=sql.110).aspx) por uma conexão diferente. Cada objeto de servidor contém uma ou mais fonte de dados, exibições de fonte de dados e objetos de banco de dados, bem como assemblies e funções de segurança.

**Objetos de dimensão**

Cada objeto de banco de dados contém vários objetos de dimensão. Cada objeto de dimensão contém um ou mais atributos que são organizados em hierarquias.

**Objetos Cubo**

Cada objeto de banco de dados contém um ou mais objetos de cubo. Um cubo é definido por suas medidas e dimensões. As medidas e dimensões em um cubo são derivadas de tabelas e visões existentes na fonte de dados na qual o cubo é baseado ou para a qual é gerado a partir das definições de medida e dimensão.

[Herança de objetos](javascript:void(0))

O modelo de objeto ASSL contém muitos grupos de elementos repetidos. Por exemplo, o grupo de elementos, “**Dimensions** contêm **Hierarchies**”, define a hierarquia da dimensão de um elemento. Ambos **Cubes** e **MeasureGroups** contêm o grupo de elementos, “**Dimensions** contêm **Hierarchies**."

A menos que explicitamente substituído, um elemento herda os detalhes desses grupos de elementos repetidos do nível mais alto. Por exemplo, o **Translations** para um **CubeDimension** é igual ao **Translations** de seu elemento ancestral, **Cube**.

Para substituir explicitamente as propriedades herdadas de um objeto de nível mais alto, um objeto não precisa repetir toda a estrutura e propriedades do objeto de nível mais alto. As únicas propriedades que um objeto precisa declarar explicitamente são aquelas que ele deseja substituir. Por exemplo, um **CubeDimension** pode listar apenas aquelas **Hierarchies** que precisam ser desabilitadas no **Cube** ou para o qual a visibilidade precisa ser alterada ou para a qual alguns detalhes do **Level** precisam ser fornecidos no nível da **Dimension**.

Algumas propriedades especificadas em um objeto fornecem valores padrão para a mesma propriedade em um objeto filho ou descendente. Por exemplo, **Cube.StorageMode** fornece o valor padrão de **Partition.StorageMode**. No caso dos valores padrão herdados, o ASSL aplica essas regras aos valores padrão herdados:

* Quando a propriedade do objeto filho for nula no XML, o valor da propriedade usará o valor herdado como padrão. No entanto, se você consultar o valor a partir do servidor, este retornará o valor nulo do elemento XML.
* Não é possível determinar programaticamente se a propriedade ou objeto filho foi definida corretamente no objeto filho ou herdado.

# [Exemplo](javascript:void(0))

# O cubo Importações contém duas medidas, Pacotes e Último, e três dimensões relacionadas, Rota, Origem e Horário.

# Exemplo de Cubo 1

# Os valores alfanuméricos menores ao redor do cubo são os membros das dimensões. Exemplos de membros são: terra (membro da dimensão Rota), África (membro da dimensão Origem) e 1º trimestre (membro da dimensão Horário).

### **Medidas**

# Os valores nas células do cubo representam as duas medidas, Pacotes e Último. A medida Pacotes representa o número de pacotes importados e a função Sum é usada para agregar os fatos. A medida Último representa a data de recebimento e a função Max é usada para agregar os fatos.

### **Dimensions**

# A dimensão Rota representa os meios pelos quais as importações alcançam seu destino. Os membros dessa dimensão incluem terra, não-terra, aérea, marítima, rodoviária ou ferroviária. A dimensão Origem representa os locais onde as importações são produzidas, como África ou Ásia. A dimensão Horário representa os trimestres e semestres de um mesmo ano.

### **Agregações**

# Os usuários empresariais de um cubo podem determinar o valor de qualquer medida de cada membro de qualquer dimensão, independentemente do nível do membro na dimensão, pois o Analysis Services agrega valores em níveis superiores, conforme necessário. Por exemplo, os valores de medida na ilustração anterior podem ser agregados de acordo com uma hierarquia de calendário padrão usando a hierarquia Tempo de Calendário na dimensão Horário, conforme ilustrado no diagrama a seguir.

# Diagrama de medidas organizadas ao longo da dimensão de hora

# Além de agregar medidas usando uma única dimensão, você pode agregar medidas usando combinações de membros de diferentes dimensões. Isso permite que os usuários empresariais avaliem, simultaneamente, as medidas em várias dimensões. Por exemplo, se um usuário empresarial quiser analisar trimestralmente as importações que são recebidas por via aérea dos hemisférios ocidental e oriental, ele poderá emitir uma consulta ao cubo para recuperar esse conjunto de dados.

# Programando o cliente no ADOMD.NET

# Os componentes de cliente do ADOMD.NET residem no namespace Microsoft.AnalysisServices.AdomdClient (em microsoft.analysisservices.adomdclient.dll). Esses componentes de cliente oferecem a aplicativos cliente e da camada intermediária a funcionalidade de consultar com facilidade dados e metadados de um repositório de dados analíticos, como o Microsoft SQL Server Analysis Services.

# Estabelecendo conexões no ADOMD.NET

# Para abrir uma conexão no ADOMD.NET, primeiro especifique uma cadeia de conexão para uma fonte de dados analíticos e um banco de dados válidos. Em seguida, abra a conexão explicitamente para a fonte de dados.

### Especificando uma fonte de dados multidimensional

# Para especificar uma fonte de dados analíticos e um banco de dados, defina a propriedade [ConnectionString](https://technet.microsoft.com/pt-br/library/microsoft.analysisservices.adomdclient.adomdconnection.connectionstring(v=sql.110).aspx) propriedade do objeto [AdomdConnection](https://technet.microsoft.com/pt-br/library/microsoft.analysisservices.adomdclient.adomdconnection(v=sql.110).aspx). A cadeia de conexão especificada para a propriedade [ConnectionString](https://technet.microsoft.com/pt-br/library/microsoft.analysisservices.adomdclient.adomdconnection.connectionstring(v=sql.110).aspx) é uma cadeia de caracteres compatível com OLE DB. O ADOMD.NET usa a cadeia de conexão especificada para determinar como será feita a conexão ao servidor.

# A propriedade [ConnectionString](https://technet.microsoft.com/pt-br/library/microsoft.analysisservices.adomdclient.adomdconnection.connectionstring(v=sql.110).aspx) pode ser definida em um objeto [AdomdConnection](https://technet.microsoft.com/pt-br/library/microsoft.analysisservices.adomdclient.adomdconnection(v=sql.110).aspx) existente ou durante a criação de uma instância de um objeto [AdomdConnection](https://technet.microsoft.com/pt-br/library/microsoft.analysisservices.adomdclient.adomdconnection(v=sql.110).aspx). O código a seguir demonstra como definir a propriedade [ConnectionString](https://technet.microsoft.com/pt-br/library/microsoft.analysisservices.adomdclient.adomdconnection.connectionstring(v=sql.110).aspx) durante a criação de um[:Microsoft.AnalysisServices.AdomdClient.AdomdConnection]:

# C#

AdomdConnection advwrksConnection = new AdomdConnection("Data Source=localhost;Catalog=AdventureWorksAS");

System.Diagnostics.Debug.Writeline(advwrksConnection.ConnectionString);

### Abrindo uma conexão HTTP segura

O exemplo a seguir demonstra como usar o ADOMD.NET para abrir uma conexão HTTP para o banco de dados de exemplo **AdventureWorksAS** do Analysis Services:

VB

Public Function GetAWEncryptedConnection( \_

Optional ByVal serverName As String = "https:\\localhost\isapy\msmdpump.dll") \_

As AdomdConnection

Dim strConnectionString As String = ""

Dim objConnection As New AdomdConnection

Try

' To establish an encrypted connection, set the

' ProtectionLevel setting to PktPrivacy.

strConnectionString = "DataSource=" & serverName & ";" & \_

"Catalog=AdventureWorksAS;" & \_

"ProtectionLevel=PktPrivacy;"

' Note that username and password are not supplied here.

' The current security context is used for authentication

' purposes.

objConnection.ConnectionString = strConnectionString

objConnection.Open()

Catch ex As Exception

objConnection = Nothing

Throw ex

Finally

' Return the encrypted connection.

GetAWEncryptedConnection = objConnection

End Try

End Function

# Componentes do servidor de mecanismo OLAP

# O componente de servidor do Microsoft SQL Server Analysis Services é o aplicativo msmdsrv.exe, que é executado como um serviço do Windows. Esse aplicativo consiste em componentes de segurança, um componente de ouvinte do XML for Analysis (XMLA), um componente de processador de consulta e vários outros componentes internos que executam as seguintes funções:

# [Diagrama de arquitetura](javascript:void(0))

# Uma instância do Analysis Services executada como serviço e comunicação autônomos com o serviço acontece por meio do XML for Analysis (XMLA), usando HTTP ou TCP. AMO é uma camada entre o aplicativo de usuário e a instância do Analysis Services. Essa camada fornece acesso a objetos administrativos do Analysis Services. AMO é uma biblioteca de classe que recebe comandos de um aplicativo cliente e os converte em mensagens de XMLA para a instância do Analysis Services. AMO apresenta objetos de instância do Analysis Services como classes para o aplicativo de usuário final, com membros de método que executam comandos e membros de propriedade que mantêm os dados para os objetos Analysis Services.

# A ilustração a seguir mostra a arquitetura de componentes do Analysis Services, inclusive todos os elementos principais executados dentro da instância do Analysis Services e todos os componentes de usuário que interagem com a instância. A ilustração também mostra que o único modo de acessar a instância é usando o ouvinte do XML for Analysis (XMLA) ou usando HTTP ou TCP.

# Diagrama da arquitetura de sistema do Analysis Services

# [Ouvinte XMLA](javascript:void(0))

# O componente ouvinte XMLA processa todas as comunicações de XMLA entre o Analysis Services e seus clientes. O parâmetro de configuração do Analysis ServicesPort no arquivo msmdsrv.ini pode ser usado para especificar uma porta na qual uma instância do Analysis Services ouve. Um valor 0 nesse arquivo indica que o Analysis Services ouve na porta padrão. A menos que especificado de outro modo, o Analysis Services usa as seguintes portas TCP padrão:

|  |  |
| --- | --- |
| Porta | Descrição |
| 2383 | Instância padrão do SQL Server Analysis Services |
| 2382 | Redirecionador para outras instâncias do SQL Server Analysis Services. |
| Atribuído dinamicamente na inicialização do servidor | Instância nomeada do SQL Server Analysis Services. |

# Cubos locais (Analysis Services – Dados Multidimensionais)

# Os cubos locais e os modelos de mineração locais permitem a análise em uma estação de trabalho cliente enquanto ela estiver desconectada da rede. Por exemplo, um aplicativo cliente pode chamar o OLE DB for OLAP 9.0 Provider (MSOLAP.3), que carrega o mecanismo do cubo local para criar e consultar cubos locais, conforme mostra a seguinte ilustração:

# Arquitetura cliente para cubos e modelos locais

### **Exemplo de recuperação de dados em um estado conectado**

O exemplo a seguir cria uma conexão ao servidor local e executa um comando nessa conexão. O exemplo analisa os resultados usando o modelo de objeto **CellSet**: as legendas (metadados) das colunas são recuperadas do primeiro eixo e as legendas (metadados) de cada linha são recuperadas do segundo eixo, e os dados de interseção são recuperados por meio da coleção de [P:Microsoft.AnalysisServices.AdomdClient.CellSet.Cells](https://msdn.microsoft.com/pt-br/library/microsoft.analysisservices.adomdclient.cellset.cells(v=sql.120).aspx).

C#

string ReturnCommandUsingCellSet()

{

//string builder para aramazenar os resultadoss

StringBuilder result = new StringBuilder();

//Conectar ao servidor local

using (AdomdConnection conn = new AdomdConnection("Data Source=localhost;"))

{

conn.Open();

//Cria um comando, usand a conexão

AdomdCommand cmd = conn.CreateCommand();

cmd.CommandText = @"

WITH MEMBER [Measures].[FreightCostPerOrder] AS

[Measures].[Reseller Freight Cost]/[Measures].[Reseller Order Quantity],

FORMAT\_STRING = 'Currency'

SELECT

[Geography].[Geography].[Country].&[United States].Children ON ROWS,

[Date].[Calendar].[Calendar Year] ON COLUMNS

FROM [Adventure Works]

WHERE [Measures].[FreightCostPerOrder]";

//Execute the query, returning a cellset

CellSet cs = cmd.ExecuteCellSet();

//Output the column captions from the first axis

//Note that this procedure assumes a single member exists per column.

result.Append("\t");

TupleCollection tuplesOnColumns = cs.Axes[0].Set.Tuples;

foreach (Tuple column in tuplesOnColumns)

{

result.Append(column.Members[0].Caption + "\t");

}

result.AppendLine();

//Output the row captions from the second axis and cell data

//Note that this procedure assumes a two-dimensional cellset

TupleCollection tuplesOnRows = cs.Axes[1].Set.Tuples;

for (int row = 0; row < tuplesOnRows.Count; row++)

{

result.Append(tuplesOnRows[row].Members[0].Caption + "\t");

for (int col = 0; col < tuplesOnColumns.Count; col++)

{

result.Append(cs.Cells[col, row].FormattedValue + "\t");

}

result.AppendLine();

}

conn.Close();

return result.ToString();

} // using connection

}