**SSAS Storage Modes:**

There are three standard storage modes (MOLAP, ROLAP and HOLAP) in OLAP applications which affect the performance of OLAP queries and cube processing, storage requirements and also determine storage locations.

A big advantage of a BI solution is the existence of a cube. Data and aggregations are stored in a optimized format to offer very fast query performance.

Sometimes, a big disadvantage of storing data and aggregations in a cube is the latency that it implies. SSAS processes data from the underlying relational database into the cube. After this is done the cube is no longer connected to the relational database so changes to this database will not be reflected in the cube. Only when the cube is processed again, the data in the cube will be refreshed.

**Storage Modes in SSAS cube**

* MOLAP (Multi-dimensional Online Analytical Processing) MOLAP is the most used **storage type**.
* ROLAP (Relational Online Analytical Processing) ROLAP does not have the high latency disadvantage of MOLAP.
* HOLAP (Hybrid Online Analytical Processing) HOLAP is a **storage type** between MOLAP and ROLAP.

**MOLAP (Multi-dimensional Online Analytical Processing)**

MOLAP is the most used storage type. It’s designed to offer maximum query performance to the users. Data AND aggregations are stored in optimized format in the cube. The data inside the cube will refresh only when the cube is processed, so latency is high.

**ROLAP (Relational Online Analytical Processing)**

ROLAP does not have the high latency disadvantage of MOLAP. With ROLAP, the data and aggregations are stored in relational format. This means that there will be zero latency between the relational source database and the cube.

Disadvantage of this mode is the performance, this type gives the poorest query performance because no objects benefit from multi-dimensional storage.

**HOLAP (Hybrid Online Analytical Processing)**

HOLAP is a storage type between MOLAP and ROLAP. Data will be stored in relational format (ROLAP), so there will also be zero latency with this storage type.

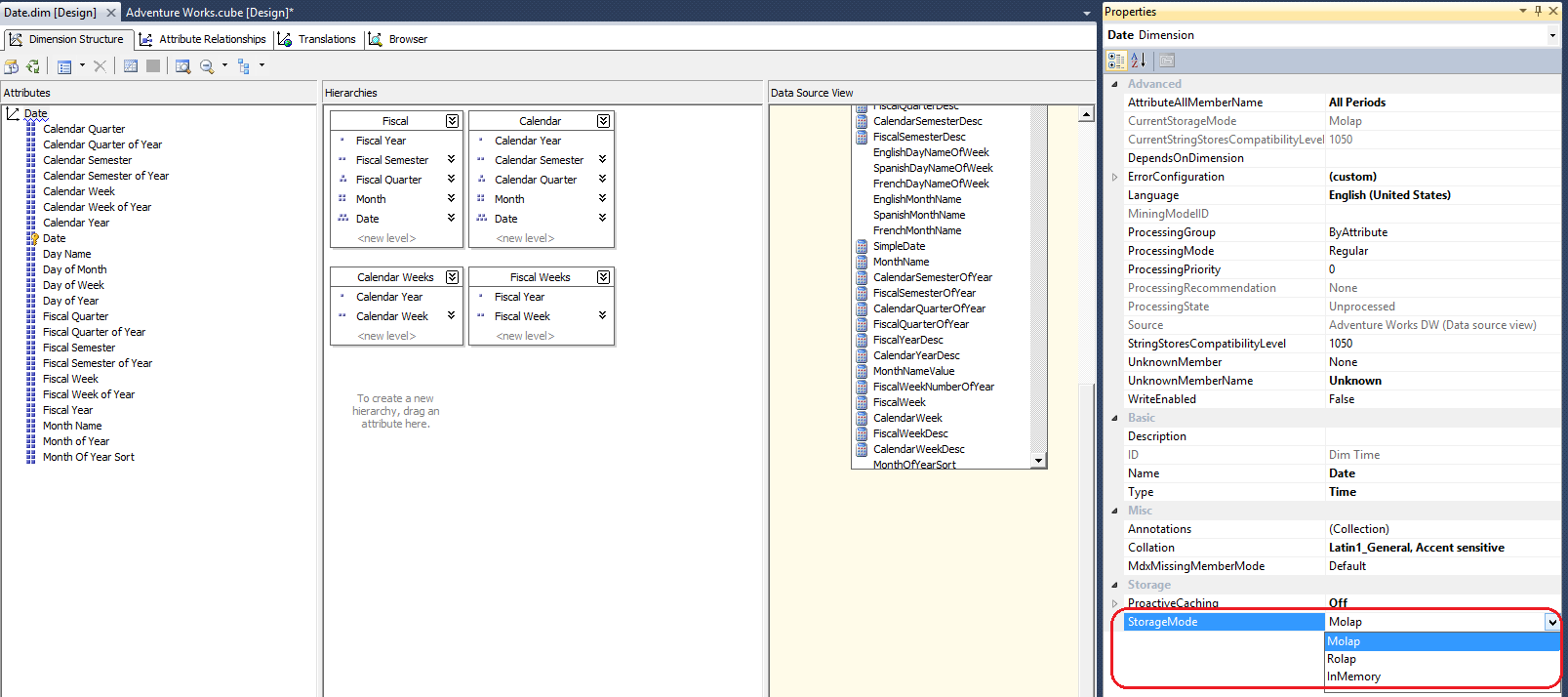
Aggregations, on the other hand, are stored in multi-dimensional format (MOLAP) in the cube to give better query performance. SSAS will listen to notifications from the source relational database, when changes are made, SSAS will get a notification and will process the aggregations again.

With this mode it’s possible to offer zero latency to the users but with medium query performance compared to MOLAP and ROLAP.

Please find below the difference between these three –

**Steps to change storage type:**

Open the SSAS database in BIDS --> Open the Dimension to change the storage mode --> Properties --> Storage Mode. You would see the current storage mode of your dimension, you can change the desired storage mode and save.



**Hierarchies**

SQL Server Analysis Services lets you create user-defined hierarchies. A hierarchy is a collection of levels based on attributes. For example, a time hierarchy might contain the Year, Quarter, Month, Week, and Day levels. In some hierarchies, each member attribute uniquely implies the member attribute above it. This is sometimes referred to as a natural hierarchy. A hierarchy can be used by end users to browse cube data. Define hierarchies by using the Hierarchies pane of Dimension Designer in SQL Server Data Tools (SSDT).

**The types of hierarchies in SSAS are:**

* User-Defined **Hierarchy**. A user-defined **hierarchy** is created to organize **hierarchical** structures and provide the end user with navigation paths in the cube. ...
* Parent-Child **Hierarchy**. ...
* Attribute **Hierarchy**

Although the possibilities are endless, some very intuitive uses for hierarchies include:

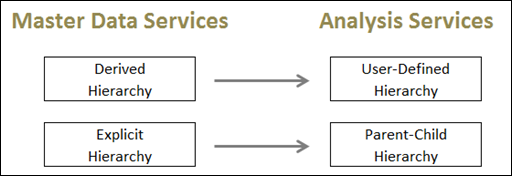
     Year  >  Quarter  >  Month  >  Day

     Country  >  State  >  City  >  Zip Code

     Company  >  Division  >  Department

**Summary of How MDS Hierarchies Translate to SSAS Hierarchies**

The MDS hierarchies relate to SSAS hierarchies as follows:



The remainder of this entry discusses the hierarchy types in a bit more detail.

**Hierarchies in Master Data Services**

There types of hierarchies in MDS include:

1. **Derived Hierarchy.**  A derived hierarchy is formed automatically by relationships in the data, with no maintenance of the level assignments required by the data steward.  (A recursive hierarchy, such as Manager –> Employee type of recursion, is a specific type of derived hierarchy.)
2. **Explicit Hierarchy.**  An explicit hierarchy uses consolidated members to group data (as opposed to levels).  The task of maintaining how data is assigned throughout the hierarchy is done manually by the data steward.
3. **Derived Hierarchy with Explicit Caps.**  This is a hybrid approach.  Level 1 underneath the Root would be from a derived hierarchy, and everything else underneath is an explicit hierarchy.  For simplicity, this entry focuses on items 1 and 2 only:  Derived and Explicit Hierarchies.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **MDS Hierarchy Type** | **Definition** | **How Maintained** | **Ragged Levels Permitted** | **Control Over Sorting** | **Usage of Consolidated Attributes** | **Support For Recursion** |
| Derived | Natural hierarchy formed from relationships in the data | Automatically (domain-based attributes) | No (requires fixed # of levels) | No  (by code only) | No | Yes |
| Explicit | A hierarchy which is created explicitly for grouping members | Manually (drag & drop) | Yes | Yes (drag & drop order of members) | Yes | No |

A derived hierarchy would be useful in a situation such as: Country > State > City > Zip Code.  However, you may consider an explicit hierarchy instead when your needs are subject to change and don’t always formulate consistent levels, such as when you have varying territory levels or varying salesperson levels.

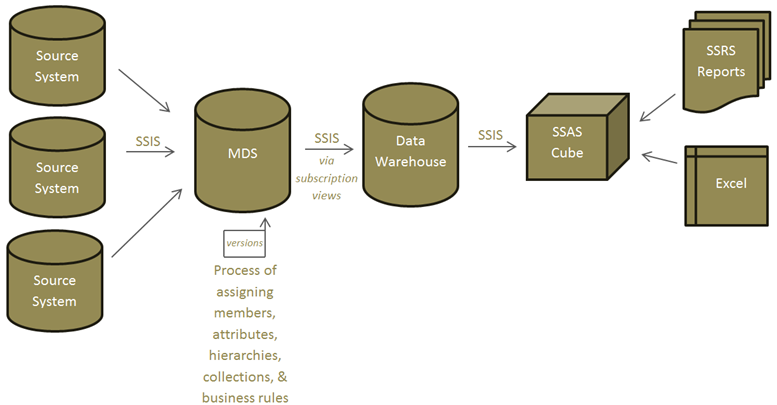
**Hierarchies in Analysis Services**

The types of hierarchies in SSAS are:

1. **User-Defined Hierarchy.**  A user-defined hierarchy is created to organize hierarchical structures and provide the end user with navigation paths in the cube.  A user-defined hierarchy should have accompanying attribute relationships.  These attribute relationships would mirror the Derived Hierarchy relationships in MDS.
2. **Parent-Child Hierarchy.**  A parent-child hierarchy is formed by a single parent attribute which has a self-referencing relationship.  This is equivalent to an explicit hierarchy in MDS.
3. **Attribute Hierarchy.**  An attribute hierarchy is just the individual attributes in a dimension.  Because it stores the Leaf Level and an All Level, it’s considered a two level hierarchy.  There’s no direct comparison to this concept in MDS.

**Pulling It All Together**

Following is how I tend to manage the promotion of data:  from MDS, to a relational DW, then into an SSAS database.



In the current SQL Server 2008 R2 version, you need to set up user-defined hierarchies and parent-child hierarchies in the SSAS Dimension Designer, completely independently from what was done in MDS.  Be sure that the attribute relationships in SSAS are the same as the derived hierarchy relationships in MDS.