

**Pentaho and Online
Analytical Processing (OLAP)**

HITACHI

Inspire the Next

Change log (if you want to use it):

Date	Version	Author	Changes

Contents

- Overview..... 1
 - Before You Begin 1
 - Use Case: Schema Creation..... 1
- Schema Setup 2
- Handling Dimensions 3
 - Date and Time Dimensions 3
 - Fact Table Values 3
 - Add Geographic Annotations 4
- Measures and Dimensions 5
 - Configuring Measures and Dimension Keys..... 5
 - Configuring Hierarchies 5
 - Using Shared Dimensions..... 6
- Development Recommendations 7
- Related Information 7

This page intentionally left blank.

Overview

This document is intended to provide best practices around how to design and build your Pentaho Online Analytical Processing (OLAP) solution for maximum speed, reuse, portability, maintainability, and knowledge transfer.

Topics are arranged in a series of groups with individual best practices for that topic explained. Some of the things discussed here include schema maintenance, schema object naming, specific dimension handling, and measure, dimension, and hierarchy definitions.

Software	Version(s)
Pentaho	6.x, 7.x, 8.0

The [Components Reference](#) in Pentaho Documentation has a complete list of supported software and hardware.

Before You Begin

This document assumes that you have knowledge about Pentaho and OLAP and that you have already installed and set up the Pentaho software.

Use Case: Schema Creation

Janice is a database administrator at a company that has recently added some new personnel. She needs to create a schema to organize her database and keep things sorted and viewable only to those who need to see them.

Janice has decided to create a schema using the Pentaho Data Source Wizard at first, for its ease of use. After she has created the schema, she has decided to edit it further using Schema Workbench, to take advantage of its added functionality.

Schema Setup

When you set up your schemas, you have a choice among several methods, including the Pentaho Data Source Wizard, Schema Workbench, and certain XML editors.


However, while this choice does exist, we do have a specific recommendation for your schema setup. Pentaho Data Source Wizard is a very user-friendly tool for you to set up your schema, but it provides only about 70% of the functionality that you can get by using Schema Workbench or an XML editor.



We recommend you use the Data Source Wizard to create your Analyzer schema, and then switch to Schema Workbench or an XML editor to use from that point onward. This will get you the greatest ease and functionality from your schema setup,

In addition to this recommended methodology, we recommend the following for your schema(s):

Table 1: Schema Setup Recommendations

Recommendation	Rationale and Details
<p>Limit the number of schemas per database connection to one.</p>	<p>By using multiple cubes per schema, you get better caching performance, easier maintenance, and security integration.</p>
<p>Ensure multiple cubes in a single schema share the same connection.</p>	
<p>Define multiple cubes for each fact table and use shared dimensions that are conformed across the multiple fact tables.</p>	<p> <i>Do not use the Data Source Wizard for this, as it only allows you to create one cube per schema.</i></p>
<p>Use name and caption for flexibility in data display.</p>	<p>This way, you can change the schema caption later without affecting your code. If you change the name of the schema instead, your code will be affected. Use caption when you want to display a different header from what the actual name is (in the data). You may change, over time, what the header is displayed to the end user, but you don't have to rewrite queries in reports. Use name column when you want to display the data value that is different from what the code is.</p>
<p>Use the same case for the naming, without spaces in the name.</p>	<p>If you do this, you can write your MDX query without using [].</p>

Handling Dimensions

Once you have your schema set up, you need to configure your dimensions. We have several recommendations to help you with this part of the process.

You can find more information on these topics in the following sections:

- [Date and Time Dimensions](#)
- [Fact Table Values](#)
- [Geographic Annotations](#)

Date and Time Dimensions

The Pentaho software contains extra features available to time dimensions, and to each level type. For example, these features and identifications mean that Pentaho can calculate year to date (YTD), month to date (MTD), and so forth.



For these calculations to work, make sure that your date/time dimension and levels are identified to Pentaho Analyzer.

Analyzer goes further by providing a handy GUI interface for relative date filtering *only* to those hierarchy levels that have `AnalyzerDateFormat` properly defined. To use this feature:

1. Ensure your date dimension is of type `TimeDimension`.
2. Each level in all time hierarchies should have the `levelType` attribute set to one of the `TimeXYZ` types.
3. Add `AnalyzerDateFormat` annotations to each level down to the day level as described in *Define Custom Actions in Mondrian*.



Sub-day annotations are not supported at this time.

4. Populate your dimension table with every date between your data's start and end date. This allows you to analyze dates that do not have data, which can be very useful, but difficult without date members for each day.

Fact Table Values

During data load, populate only those dimensions that you have data for. For all dimensions *other than time dimensions*, do not populate your dimension table with values that do not exist in the fact table.

Dimensions have many more possible values than data is provided. Values that cannot be analyzed will increase overhead and memory usage; in contrast to time dimension values since the time dimension is unique. Whereas there may not be data for a date, there will always be dates in times.

Add Geographic Annotations

Identify your geographic levels to Analyzer by adding `Data.Role` and `Geo.Role` annotations to the schema. This will make sure that each level is recognized by the Geo-mapping visualization in Analyzer as described in [Add Geo Map Support to a Mondrian Schema](#), and allows Analyzer to know how to drill up and down on geographic dimension levels.



If you have run the Data Source Wizard, it should identify and add those annotations for you.

Measures and Dimensions

We recommend the following actions regarding measures and dimensions:

- [Configuring Measures and Dimension Keys](#)
- [Configuring Hierarchies](#)
- [Using Shared Dimensions](#)

Configuring Measures and Dimension Keys

Create all possible varieties of a measure and dimension keys in a cube. While these additional measures might not be requested or used in Analyzer, having them provides you with maximum flexibility for data analysis without having to modify the schema later.

Define the following measures, dimension, and primary keys:

- `Sum`, `min`, `max`, and `avg` measures for all facts.
- `Distinct count` for all dimension key fields.
- `Count` and `distinct count` measures for the primary key of the fact table.
- *Optional*: Create a spread (`Max-Min`) measure to provide a range of values to be used for distribution analysis.

Finally, use the `AnalyzerBusinessGroup` annotation to organize measures into subgroups. This helps users navigate and find measures in the Analyzer UI, especially if you have many measures. [Add Business Groups](#) in Pentaho Documentation describes this process.

Configuring Hierarchies

Create multiple hierarchies in dimensions where different start points and drill-downs are necessary, especially when you will need to analyze a lower level of a hierarchy independent of the higher levels (most common with time/date). A day of month level allows day by day analysis without regard to month or year levels. In that case, a day of month level is the topmost level in a new hierarchy.

Analyzer schemas support multiple hierarchies per dimension. Avoid too many single level hierarchies in a dimension. Many single level hierarchies should be converted into `Member` properties. Each child level of a hierarchy should have more rows than the parent and be directly related to the parent level. Independent levels should be on their own hierarchy in the same dimension.

Analyzer uses the `approxRowCount` attribute to determine how to load and cache members. If this attribute is not specified for all levels of every hierarchy, Analyzer must then build queries to go after this data at runtime, impacting performance.

Schema Workbench or an XML editor will allow you to specify the estimated number of rows using this attribute. It does not have to be exactly the correct number or magnitude (correct number of zeroes).

Using Shared Dimensions

Shared dimensions allow Analyzer to cache members and reuse them across multiple cubes in the same schema. This improves performance and reduces memory required, so make frequent but appropriate usage of shared dimensions.

Don't use a shared dimension that is not used across more than one cube in a schema. If it is only used on one cube, then it is not shared, so make it a local dimension.

If all cubes have only conformed dimensions, then consider just having one cube. Normally each cube has at least one dimension that is not conformed.

This practice assumes you are using multiple cubes within one schema. A schema should house all cubes related to that database connection.

Development Recommendations

We recommend the following practices as you develop your schemas. Both of these suggestions help you avoid hard-to-trace errors on the server, and allow you to provide only functioning schemas to users.

Table 2: Schema Development Recommendations

Recommendation	Rationale and Details
<p>Extend Analyzer schemas incrementally, creating the most basic schema first and then expanding.</p>	<p>When too many pieces are added to a schema at once, the true error can be hard to track down. When you expand your schema, add only one or a few measures, cubes, dimensions, hierarchies, or levels before testing again, so you can more easily identify the problem.</p>
<p>Test your schema before publishing it to the Pentaho Server.</p>	<p>You will want to make sure the schema works before you publish it. To do this, run a test MDX query on each cube:</p> <ol style="list-style-type: none"> 1. In Schema Workbench, go to File > New MDX Query. 2. If you can connect to the schema, that means that Analyzer can at least parse the schema. Many errors are caught this way. 3. For a further test, you can use a simple MDX query : <code>SELECT Measures.members on COLUMNS from [MyCubeName]</code>. This simple test will prove you can connect to the cube and database and get the default members for all hierarchies.

Related Information

Here are some links to information that you may find helpful while using this best practices document:

- [Pentaho Components Reference](#)
- [Add Business Groups](#)
- [Add Geo Map Support to a Mondrian Schema](#)
- [Define Custom Actions in Mondrian](#)