

OLAP 的基本特征



又是一篇关于商务智能（ BI ）方面的文章，前面有几篇文章介绍了数据仓库、 多维模型和 OLAP方面的知识。这篇文章主要总结了 OLAP 具备的一些基本特征，以及其在数据的处理、展示和分析中体现的优势。

其实我们大部分时间是在模仿，参考书本或者他人的范例，而当我们去实现这些东西的时候，我们又会有自己的体验，我们需要将这些体验记录下来，当我们能够自己去总结整个实现过程的时候，其实可以认为我们已经掌握了这个知识或技能。而正是在总结的过程中，我们也许会发现原先的范例可能并不是最优的，我们会产生自己的思考和优化方案，其实到这一步的时候你已经实现了一个超越，而当你自己的方案被实践所验证时，那么可能你已经站在了一些人的前面了。而我今天要做的就是——“总结”。

OLAP的类型和基本操作

先来回顾下一些基础知识，之前的文章——数据立方体与 OLAP中介绍过 OLAP的一些基本知识，包括 OLAP的类型：ROLAP MOLAP HOLAP

以及 OLAP的基本操作：钻取（ Drill-down ）、上卷（ Roll-up ）、切片（ Slice ）、切块（ Dice ）以及旋转（ Pivot ）。

因为这些在之前的文章中都有介绍，这里不再重复了，有兴趣了解的同学直接去看我之前的那篇文章即可。

OLAP的优势

OLAP的优势包括之前提到的 丰富的数据展现方式 、 高效的数据查询 以及 多视角多层次的数据分析 。 这里再补充两点，是 Oracle 11g 的官方文档介绍 OLAP 时提到的在 Oracle 中使用 Cube 所具备的优势（当然 Oracle 里面的 Cube 指的是 MOLAP类型的数据组织方式，有点偏技术了）：

从细粒度数据到汇总数据的预聚合（ fine-grained approach to pre-aggregating summary data ）： Oracle 的 Cube 提供了基于成本的预聚合（ Cost based pre-aggregation ），也就是既不会完全不进行预先的聚合，也不会将每个维每个层次的数据都预先聚合起来；而是会去考虑对于每条记录聚合的成本，并将那些在动态聚合中相对高成本的记录预先聚合并储存起来，这样相当于权衡了立方数据加载时的压力和数据查询时的效率（过度的预聚合会使数据加载的时间和空间成本提高，而过少的预聚合则会让数据查询的效率降低）。而其最终实现的就是最具性价比的快速查询（ fast query ）。

维和立方之间的预关联（ pre-joining of dimensions to cube ）：当然这个也是基于 MOLAP所具有的优势， MOLAP是基于数组来构建的，所以维和立方之间是预关联的，也就是相比 ROLAP而言，其消除了构建索引以及建立表或物化视图时所需要的额外的时间开销，而在聚合数据的时候也避免了维和立方之间的多次关联。

OLAP的基本特征

进入主体内容，下面是我自己对 OLAP所具备的基本特征的总结，当然包括一些国外的博客和相关网站的介绍（现在打开某些国外的网站还真累）， Oracle

的一些文档资料以及自己在实际使用时的体会。其实每个特征都从不同层面上体现着 OLAP对数据的组织、处理和分析上的优势。

数据建模 (Data Modeling)

我们知道数据仓库的特征之一是面向主题的，而数据模型的构建正是为了将原本基于业务关系的数据整理成更符合人们日常观察事物的一般方式，多维模型让人们数据的观察更加得心应手，数据建模的优势就是体现在 简化了复杂的数据组织逻辑和关系。

						Measures		
Region	My Image	P	P0	P1	P2	▲ Measures[0]	▲ Measures[1]	▲ Measures[2]
-All Region[0]		0	V00			856.44 ↗	820.95 ↗	983.55 ↘
+Region[0]		0		V10		1,095.05 ↘	1,087.79	958.23
-Region[1]		1		V11		1,052.05	1,107.59	980.32
+City[0] ↘	✓	0			V20	1,088.31	884.90	1,085.73
+City[1]	✓	1			V21	1,336.43	943.27	1,117.62
+City[2] ↗	✓	2			V22	969.34	1,086.70	1,094.71
+City[3] ↘	✓	3			V23	964.30	900.58	953.65 ↗
+City[4]	✓	4			V24	915.59 ↗	1,032.98 ↘	876.19 ↘
+City[5] ↗	✓	5			V25	1,023.12	1,242.39	988.85
+City[6] ↘	✓	6			V26	949.16	941.62	1,077.48
+City[7]	✓	7			V27	1,055.92	1,095.56	1,078.26
+Region[2]		2		V12		1,027.36	1,084.88	1,009.71
+Region[3]		3		V13		972.10	932.27	836.34
+Region[4]		4		V14		919.44	926.47 ↗	873.20 ↗

多维与可视化 (Multidimensional and Visual)

多维和可视化体现在 对数据多视角多层次的展现上。其实多维模型的 OLAP 在可视化层面上主要体现在报表上的钻取、上卷、切片等操作，如果用过 Mondrian 的开源 OLAP引擎就能体验到其实就是一个类似树形结构的展开，就像 Windows 里面的资源管理器左侧列表，这个符合人们日常观察和使用的习惯。同

时大部分的报表工具都支持此类的 OLAP展示，MDX(Multi-Dimensional Expression ，多维表达式)就是专门为多维 OLAP打造的查询语法标准。

聚合 (Aggregation)

聚合的优势体现在 满足了从细节数据到高度汇总数据的不同需求 。聚合的特征在多维模型中体现为预计算 (pre-calculated)以及快速查询 (fast query)上面，能够在不同的数据粒度上对数据进行聚合汇总，满足数据的多种需求。

计算度量 (Calculated Measures)

计算度量 更加丰富地表现了各类指标的延伸、比率及变化趋势 等。最简单的计算度量就是指标间的加减乘除、排名及比例，常见的例子就是销售额减成本计算得到利润，进而根据利润对不同的产品进行排名，或者计算各类产品类型的利润所占的比例等；另外一种就是基于时间序列上计算得到的度量，比如同比增长、环比增长、期初累计、移动平均等。所以计算度量的存在让我们的分析指标有了更多的选择。

预测 (Forecast)

其实熟悉 OLAP, 用过相关 OLAP工具的朋友都知道，大部分的 OLAP都会提供预测的功能，一般是基于时间序列的预测，工具直接提供相应的预测方法，比如加权移动平均法、指数平滑法（历史数据加权平均的不断迭代的过程）等。因为在实践中没有用到过，所以这里也不便讨论起具体的意义多大，但这种不需要自己去写算法，而直接使用工具根据相应的聚合数据预测未来的趋势，至少能为我们快捷地 展现数据可能的走向，并做出可能调整 。

摘自：网站数据分析