**xVelocity in SQL Server 2012**

**SQL Server 2012**

xVelocity is Microsoft’s family of in-memory and memory-optimized data management technologies in SQL Server 2012. The xVelocity in-memory analytics engine and the xVelocity memory-optimized columnstore index feature are the first two members of this family.

[**xVelocity in-memory analytics engine (VertiPaq)**](javascript:void(0))

The xVelocity in-memory analytics engine is the next generation of the VertiPaq™ engine that was introduced in SQL Server 2008 R2, with PowerPivot for Excel 2010 and PowerPivot for SharePoint 2010. Vertipaq™ is an in-memory columnstore engine that achieves breakthrough performance for analytic queries by employing techniques such as columnar storage, state-of-the-art compression, in-memory caching and highly parallel data scanning and aggregation algorithms. In SQL Server 2012, the xVelocity in-memory analytics engine has been enhanced to support both self-service BI (PowerPivot) as well as corporate BI (Analysis Services tabular mode) scenarios.

The xVelocity engine has two usage scenarios in the context of Business Intelligence:

* Information workers can use PowerPivot for Excel to integrate data from a number of sources, cleanse and model the data, enrich the data with business logic, analyze the data and build reports and visualizations. Since PowerPivot uses the xVelocity engine under the covers, information workers are not limited by the restrictions of Excel. They can work with several millions of rows of data and still benefit from split-second response times.
* BI developers and IT professionals can use SQL Server Data Tools to create an Analysis Services Tabular project and build a BI Semantic Model. The model can contain data from a number of sources, business logic expressed in the form of DAX calculations, role-based security and large data volumes that can be managed using partitions in the xVelocity engine. When the model is deployed to an Analysis Services server, information workers can use tools like Excel and Power View to interact with the model and achieve split-second response times from the xVelocity engine.

|  |
| --- |
| **NoteNote** |
| In SQL Server 2012, the name VertiPaq has been replaced by xVelocity in-memory analytics engine (abbreviated to xVelocity where appropriate). The only exceptions are a few instances in the product (parameters, event columns, and messages) that continue to use the VertiPaq name. |

In Analysis Services, the xVelocity engine is enabled when you choose Tabular mode during an Analysis Services installation. You can then author and deploy tabular model projects by using the Analysis Services Tabular Project, Import from PowerPivot, or Import from Server project templates in SQL Server Data Tools. xVelocity is not available with Analysis Services Multidimensional and Data Mining solutions. For more information about installing and using tabular mode, see [Install Analysis Services in Tabular Mode](http://technet.microsoft.com/en-us/library/hh231722(v=sql.110).aspx) and [Tabular Modeling (SSAS Tabular)](http://technet.microsoft.com/en-us/library/hh212945(v=sql.110).aspx). For more information about PowerPivot for Excel, see [Install PowerPivot for Excel](http://technet.microsoft.com/en-us/library/gg413462(v=sql.110).aspx) and [PowerPivot for Excel](http://technet.microsoft.com/en-us/library/ee210644(v=sql.110).aspx).

[**xVelocity memory optimized columnstore index**](javascript:void(0))

The xVelocity columnstore index feature is used to enable high performance query processing in SQL Server data warehouses and data marts. Data is kept in column-wise fashion in an xVelocity columnstore index, with each column stored separately, rather than storing all columns of a row on the same page. In addition, xVelocity includes a new, vector-based query execution technology called “batch processing” that tremendously speeds up query processing in conjunction with the columnstore index.

The xVelocity columnstore index feature is memory-optimized because it stores data in memory in a special representation built for fast processing, rather than simply in disk page images, like traditional database systems use. Data is brought in to this memory-optimized cache on demand, so not all data has to fit in memory. Yet pure in-memory query performance levels are achieved when all data needed by a query has already been brought in to memory. The xVelocity columnstore can give you in-memory performance, yet allows the flexibility to achieve superb performance on systems where the main memory size is smaller than the database.

To use an xVelocity columnstore index, you use T-SQL or SQL Server Management Studio to create the column store. A columnstore index then appears as an index on a table when examining catalog views or by using Object Explorer in Management Studio. For more information, see [Columnstore Indexes](http://technet.microsoft.com/en-us/library/gg492088(v=sql.110).aspx).