

Data Mashup – Under the Hood

A comparison to traditional data warehousing approaches and a technical discussion of InetSoft's Data Mashup engine

Summary

Data Mashup is a data transformation and integration technique that puts control into the hands of the business user. Data mashup melds the flexibility of a spreadsheet with enterprise-level security, performance, repeatability, and collaboration. This paper compares data mashup technology to more traditional data warehousing approaches, discusses the particular benefits offered by the InetSoft Data Mashup engine, and details its performance characteristics.

Data Mashup or Data Warehousing?

The answer is often “both”. Data Mashup and Data Warehousing are tools, and as any carpenter will tell you, it’s important to use the right tool for the job. When all you have is a hammer, everything looks like a nail. But when you have multiple tools, you need to know how to choose the right one.

ETL (Extract, Transform, and Load) is a process that transforms and manipulates data before it is loaded into a data warehouse. Data mashup, by contrast, is a process that transforms and manipulates data on demand. This fundamental difference accounts for most of the benefits and limitations of each technology. As you will see, InetSoft’s Data Mashup technology bridges the gap between ETL and basic data mashup to address many of the needs in the middle.

The benefits of ETL are chiefly reaped at runtime. Since the “heavy-lifting” is done before loading, all that remains at runtime is to return the data. However, this benefit of ETL is also its greatest challenge: resistance to change. A considerable investment of effort is required for initial setup, and then again whenever you alter the process to address new business requirements. So think of ETL as a massive container ship. Large and heavy, it takes a lot of time and energy to get up to speed or change direction. However, it can carry a lot of cargo, so it may still be efficient if you are shipping a very large number of items.

The benefits of data mashup are chiefly reaped at design time, where users can perform ad hoc transformation and integration of data. Very little work is needed to integrate a new data source or change the way the data is processed. This is because the data remains in its source systems and is transformed and integrated on demand. However, this flexibility comes at some cost in raw performance. Processing data at runtime is slower than simply returning a pre-processed result set. So think of data mashup as an airplane. It’s lighter and faster than a container ship, but can carry less cargo.

Data mashup as a substitute for ETL

The postal service is more likely to use an airplane than a container ship, because the postal service needs to transport small packages quickly. In the same way, businesses should prefer data mashup techniques when the data size does not require ETL and it would take too long to create or adapt an ETL process.

Data mashup as a precursor to data warehousing

Building a data warehouse is usually a long and expensive undertaking, whereas data mashup is quick and inexpensive. Therefore, if you need to combine data from multiple sources, you can start with data mashup. This will allow you to quickly experiment with different ways of manipulating and combining your data. When you have settled on a final version, you can implement that logic with ETL into a data warehouse to optimize performance. In other words, first find the best path through the forest by exploring on foot, and then bring in the bulldozers to pave the road.

Data Mashup as a complement to data warehousing

Business users often find themselves with ad hoc datasets, for example, a local spreadsheet or a file from a colleague. These external datasets should not be part of the data warehouse, but you can still glean value from them by combining them (“mashing them up”) with results from the data warehouse.

Data mashup techniques allow you to view the external data set on equal terms with the data warehouse results, and to easily manipulate data from both component sources.

Data Mashup Technology

When resources are scarce, business users learn to do more with less. In environmental terms: Reduce, reuse, and recycle.

Reduce: “Push-down” Methodology

The traditional way users mash up multiple datasets is to acquire the raw component datasets and then post-process the data within a spreadsheet. InetSoft’s Data Mashup engine dramatically reduces the amount of post-processing required by pushing as much of the processing into the database query (SQL) as possible. This exploits the database’s strength in efficient data processing, and reduces the amount of post-processing to a minimum. Additionally, only the final result set needs to be transferred from the database, relieving demands on vital network resources. In many cases, all of the processing can be pushed into a single SQL query.

Reuse: Caching

The simplest way to increase efficiency is to avoid redundant effort. A recently processed dataset should be reused instead of recreated from scratch. This is especially advantageous in the context of interactive analysis, where data is being explored for nuggets of useful information. Exploratory operations such as slicing, dicing, and filtering can all be done on a cached dataset. In essence, a data warehouse is a large cache, because it contains copies of the source data. The data warehouse is updated only when the ETL process runs, but with data mashup, users can trigger a refresh whenever they wish.

Recycle: Materialized Views

The key benefit of ETL is that processing occurs before the data warehouse is queried. However, with traditional ETL, the data is transformed and pre-aggregated before even a single report has been designed. This is an open invitation to problems, because the pre-aggregates provided by the ETL process may not match the data required by reports designed later. This mismatch can then only be resolved by making adjustments to the pre-aggregation rules, or by changing the entire ETL process.

Clearly, this is putting the cart before the horse. The “pre” in pre-aggregation should mean ‘before usage’, not ‘before design’. InetSoft’s Data Mashup allows you to configure pre-aggregation after the mashup is defined. You can first discover the data that you need, and then take advantage of optimization techniques to improve performance. The key tool in flexible pre-aggregation is the materialized view.

Database Materialized View

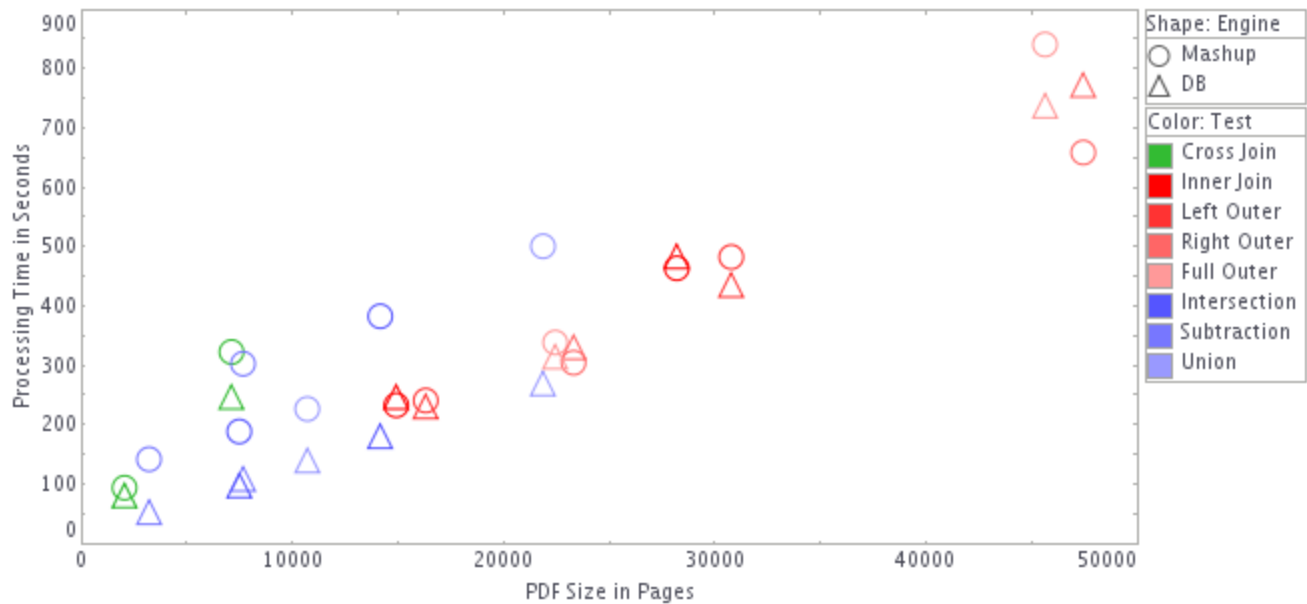
Materialization is taking a snapshot of processed data. The database engine can then re-purpose or recycle this result in the future when executing similar or related queries. Databases can keep the materialized view in sync with the raw data whenever the underlying records are changed.

Embedded Materialized View

The principle of materialization is very powerful, so InetSoft’s Data Mashup technology also provides this ability outside of the database. This means that even non-database and cross-domain queries can be materialized. Since this is done outside of the source system, the view can be refreshed with new data on a user-defined schedule rather than being triggered by changes to the underlying data.

Performance Metrics of InetSoft’s Data Mashup Engine

How well does InetSoft’s Data Mashup perform? Here are some performance numbers to give you confidence in the processing power of InetSoft’s Data Mashup engine.



Joins

The Join tests (in red) were run twice, with component datasets of 300,000 rows each and 150,000 rows each. For the database tests, the join column was not indexed because it would have defeated the purpose of the test. Notice the results are about the same whether the Oracle database or InetSoft’s Data Mashup engine processed the join. In fact, the data mashup results for the inner join and right outer join are faster than the database.

Set Operations

For set operation tests, the component datasets were 300,000 rows and 200,000 rows. The results indicate that the database is 40-60% more efficient at union, intersection, and subtraction than the data mashup engine when the datasets are large. However, the raw performance in the data mashup set operation tests were still between 23 and 47 pages per second.

Summary

The purpose of these tests is to measure the post-processing performance of InetSoft's Data Mashup Engine against a reasonable database yardstick. The data mashup process proves to be competitive, while still delivering its unique benefits of flexibility and reusability. Even with significantly large datasets, the Data Mashup results for joins were as good as, and sometimes better than, the Oracle database.

For more information on InetSoft's business intelligence application, Style Intelligence, which incorporates this Data Mashup Engine, please visit www.inetsoft.com.