IT@Intel White Paper

Intel IT
Big Data and Business Intelligence
September 2013

Integrating Data Warehouses with Data Virtualization for BI Agility

Our tests show that data virtualization solutions deploy quickly and have minimal performance overhead when some initial processing is pushed down to source containers.

Executive Overview

By deploying data virtualization solutions that combine disparate data sources into a single virtual layer, Intel IT expects to increase the agility of our business intelligence (BI). This agility will enable our business groups to more quickly solve business problems, discover operational efficiencies, and improve business results worldwide. Our tests show that data virtualization solutions deploy quickly and have minimal performance overhead when some initial processing is pushed down to source containers.

As Intel IT transitions from a "one size fits all" enterprise data warehouse (EDW) to a multicontainer approach, we need new fast, cost-effective ways to access and process the data in these containers. Such tools will enable Intel IT to help our IT customers more effectively integrate data from our big data solutions, custom independent data warehouses, and traditional enterprise data warehouses.

We tested the data virtualization capabilities of two commonly used enterprise software products: an extract-transform-load (ETL) tool and a reporting tool. Using data from Intel's customer insight program, we employed these solutions on data in multiple containers and compared their performance against a baseline setup drawing data co-located in a single source. We made the following discoveries:

- The data virtualization solutions required just one week to set up compared to approximately eight weeks for a traditional co-location approach that copies all data sets to a single container.
- Optimizations that push filtering down to the source container resulted in the best performance.
- The more processing that can be pushed down to the source container, the higher the performance.

Results from these tests are guiding our deployment of data virtualization to increase our BI agility. We are also investigating more robust, dedicated data virtualization tools.

Ajay Chandramouly
Big Data Domain Owner, Intel IT

Nitin Patil
Capability Engineer, Intel IT

Rajeshkumar Ramamurthy
BI Platform Architect, Intel IT

Shankar Radha Krishnan
BI Solution Lead, Intel IT

Jason Story
BI Capability Engineer, Intel IT

Contents

IT@INTEL

The IT@Intel program connects IT professionals around the world with their peers inside our organization – sharing lessons learned, methods and strategies. Our goal is simple: Share Intel IT best practices that create business value and make IT a competitive advantage. Visit us today at www.intel.com/IT or contact your local Intel representative if you'd like to learn more.

BUSINESS CHALLENGE

As Intel IT continues to move from a "one size fits all" enterprise data warehouse (EDW) to a multicontainer model, we need new ways to cost-effectively integrate our data containers to enable greater agility in our business intelligence (BI) solutions, reduce data duplication, and deliver faster results. This is a growing problem for Intel IT and other large organizations as data volume continues to grow exponentially, data complexity increases, and the business need intensifies for real-time intelligence to make smart, timely decisions and seize new opportunities.

The Need to Increase BI Intelligence Agility

For years Intel IT addressed Intel's BI needs with a traditional centralized EDW, bringing in data through an extract-transform-load (ETL) process. Such a solution was necessary because existing data reporting tools had significant, if not severe, limitations in combining data from more than one source at a time. Data had to be co-located (in the same container) to generate insightful reports.

While co-location in a massively robust data warehouse provides a substantial performance benefit compared to manually extracting the original data from two or more locations, this advantage comes at a cost. Co-locating data from various sources is a laborious and time-consuming process. Since business groups cannot get an integrated view of enterprise data until data is brought into the EDW, this time delay directly affects data availability, BI, and the ability to make the right decision at the right time.

The current explosion of big data and its diverse data types is presenting yet another challenge. Traditional relational database approaches are no match for this voluminous mix of structured and unstructured complex data sets. Meeting big data BI needs requires the use of multiple types of BI data warehouses to provide a dynamic range of BI analytic capabilities.

To meet the needs of this new age, Intel IT is currently deploying the following solutions:

- An EDW to handle the analysis of enterprise-wide structured data
- Apache Hadoop* to provide analysis of raw, unstructured data
- An extreme data warehouse (XDW) to enable analysis of structured and semi-structured data
- Custom, independent data warehouses to analyze structured, normalized data
- In-memory solutions to deliver real-time analysis of streaming volume data sets
- Cloud-based systems for their extremely quick creation time and ability to integrate data sets external to Intel (currently under exploration)

By matching the use case to the most appropriate BI platform, we avoid inappropriate uses of the costly EDW platform, thereby achieving substantial cost savings. This new approach is also expanding the ability of business groups across Intel to mine enormous amounts of raw and unstructured data. This is enriching the decision making process and enhancing company performance.

The challenge we are now facing with this multiple data warehouse strategy is a growing need to increase BI intelligence agility through solutions enabling data federation and real-time integration of disparate data from these various sources. Traditional co-location methods continue to prove too slow. In addition, with all these disparate data types and their containers, it's difficult to identify the best container for co-locating data and performing data integration. Finally, duplicating data in another location increases storage needs, network traffic, and data management overhead, all of which significantly increase IT costs.

Data Virtualization as One Solution

A promising solution is data virtualization, a process that federates disparate systems—such as relational databases, legacy applications, file repositories, document files, website data sources, and data services suppliers—into a single data access layer integrating data services for consuming applications or users. When data virtualization is applied, its abstraction layer hides most of the technical aspects of how and where data is stored for applications, making it seem as if just one large database is being accessed. This integration enables data-consuming applications and users to target a common data access point rather than require each tool to handle all the integrated data sources separately. To resolve differences between source and consumer formats, as well as semantics, data virtualization solutions use various abstractions, transformation techniques, and data access mechanisms.

HOW DATA VIRTUALIZATION WORKS

From a design point of view, data virtualization has three basic steps (see Figure 1):

1. Connect and virtualize data sources into an abstracted format.
2. Combine and federate sources into virtual data views.
3. Publish these views as a data service for applications or web-based tools.

PROOF OF CONCEPT

Through the use of data association tools in a common enterprise reporting application and an ETL application, we compared 40 scenarios to a baseline co-location solution to find the best use cases for employing data virtualization within our multiple data warehouse strategy.

For this proof of concept (PoC), we decided to take the most cost-effective path and use the data association features of two tools Intel IT already owns and supports. Using licensed software allows us to see first what advantages in agility and performance could be gained with existing applications, rather than investing money, time, and resources in a dedicated data virtualization solution. In this paper, we will refer to our two existing tools as the following:

- Reporting tool
- ETL tool

In the PoC, we applied these tools' data association features to source data originating from two databases used for customer insight by Intel's Sales and Marketing Group. One database was housed in a standard EDW; the other was housed in our XDW.

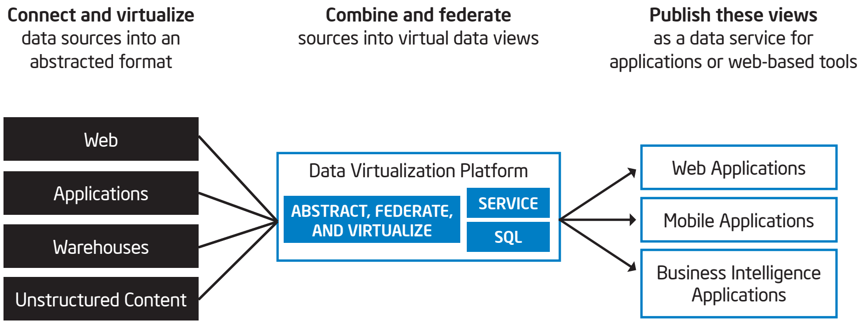
Connect and virtualize data sources into an abstracted format
Combine and federate sources into virtual data views
Publish these views as a data service for applications or web-based tools
Web
Applications
Warehouses
Unstructured Content
Data Virtualization Platform
ABSTRACT, FEDERATE, AND VIRTUALIZE
SERVICE
SQL
Web Applications
Mobile Applications
Business Intelligence Applications

Figure 1. Data virtualization federates data from disparate systems into a single data access layer, integrating data services for consuming applications or users.

Intel's Use of Big Data Solutions

Intel IT uses the open source solution Apache Hadoop* to enable the collection, processing, and analysis of large, heterogeneous data sets. Using Hadoop, we are gaining new insights from previously unexplored sets of unstructured data. This is helping our Sales and Marketing Group to enrich their understanding of customers, markets, and opportunities. Intel IT is also using big data solutions with other Intel business groups to help reduce enterprise risk and improve manufacturing efficiency.

Hadoop is a top-level open source project of the Apache Software Foundation with numerous commercial distributions available. Instead of a large supercomputer, Hadoop provides an open source framework for writing and running distributed applications that process large amounts of data. It coordinates local storage and computation across multiple servers, typically numbering in the hundreds, that act as a cluster. Each server works with a data subset. We are using Intel® Distribution for Apache Hadoop* software 2.2—a version based on Apache Hadoop and optimized for Intel® architecture—across 16 nodes. A particular Hadoop advantage is its ability to run on large clusters of mainstream two-socket servers, such as those powered by the Intel® Xeon® processor E5-2600[Δ] product family.

[Δ] Intel® processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. Go to Learn about Intel® Processor Numbers.

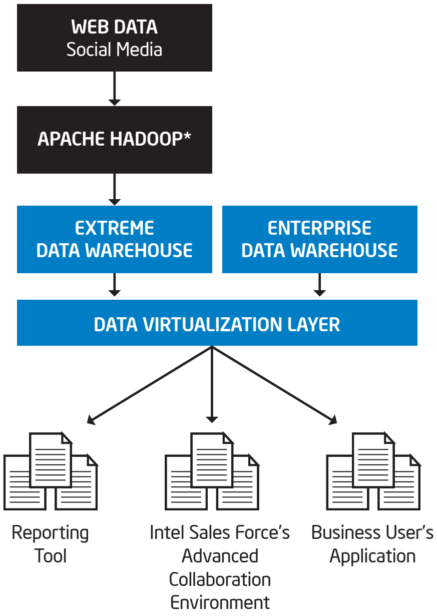
WEB DATA
Social Media
APACHE HADOOP*
EXTREME DATA WAREHOUSE
ENTERPRISE DATA WAREHOUSE
DATA VIRTUALIZATION LAYER
Reporting Tool
Intel Sales Force's Advanced Collaboration Environment
Business User's Application

Figure 2. Intel IT used this test setup for its data virtualization proof of concept, focusing on data from the enterprise data warehouse and big data extreme data warehouse used by the company's Sales and Marketing Group.

The Use Cases: Customer Insight

Intel's Sales and Marketing Group constantly seeks the most up-to-date and accurate customer insight on Intel volume customers, including original equipment manufacturers (OEMs), original device manufacturers (ODMs), and companies developing embedded solutions such as single-board computers for use in intelligent systems. To support this group, Intel IT maintains a trusted data management service that provides enterprise data governance and support, as well as a data management service that provides the latest advanced BI solutions for finding high-value line-of-business opportunities.

We use a relational database in a traditional EDW to hold the master data on business contacts and accounts, as well as maintain data on leads, opportunities, issues, and demand generation. In addition, to help give Sales and Marketing a more complete view, we support them with a big data platform using Hadoop to process web data, sales and marketing data, and logs of other customer activities (Figure 2). Through the PoC, we wanted to find the best-performing solution for integrating the data from the two sources (XDW and EDW) and enabling new reports yielding new customer insights.

TEST SETUP

Together, the data in the EDW and XDW constitute a tremendous amount of information. Intel gets 6 to 10 million web hits on a daily basis alone. A year's worth of web data on the targeted customer groups can amount to billions of rows of data. The big data platform ports its processed data to our XDW.

Typically, 80 percent of the number of tables for the Sales and Marketing Group are in our EDW and 20 percent are in the XDW. These percentages reverse when we look at volume: 80 percent of the data volume is in the XDW. This means that if we want to co-locate the data in the EDW, we have to move a large amount of data from the XDW to the EDW. Not only is this time-consuming, but it requires expensive space in the EDW.

In addition to volume, the number of records in the system are an important factor. If we are joining multiple tables and one in the EDW has 500 records and one in the XDW has 1,000,000 records, the structured-query-language (SQL) statement used must be optimized to move and join the data fast and efficiently to ensure adequate performance. The performance target is to deliver results in seconds as opposed to minutes.

Three Test Setups

To compare data virtualization using our reporting tool and our ETL tool with co-located data, we used three test setups (see Figure 3). Each setup employed the same reporting tool with web report capabilities to create reports.

- **Baseline using co-located data.** This setup enabled us to estimate the performance overhead of the data virtualization layer by drawing data from a single source into our reporting tool. All the tables were co-located in our XDW, allowing us to pull only from the XDW and measure precisely how much the additional layer adds to performance times.
- **Data virtualization using an ETL tool.** This setup used the data abstraction and integration capabilities of our ETL tool for data virtualization, drawing data from both our EDW and XDW, and then passed the results to the reporting tool.
- **Data virtualization using a reporting tool.** This setup used the data abstraction and integration capabilities embedded in our reporting tool, drawing data directly from both our EDW and XDW.

The Tests

We conducted 40 tests to see how the two data virtualization methods perform on data volumes ranging from 10 rows to 300,000 rows. The source databases had tables with up to 200 million rows of data.

In this paper, we organize these tests into three groups: simple, medium, and complex (see Table 1). Within these groups, several tests evaluated the data virtualization tools' optimization capabilities. These optimizations use filters to push down processing of select query operations into the underlying data sources. Such "pushdowns" can dramatically improve performance by using native data source capabilities and limiting the amount of intermediate data returned from a data source. For our PoC, for both our reporting tool and our ETL tool, we tested some scenarios where we pushed processing down to the source container and other scenarios where we did not.

Table 1. Scenario Descriptions

	Scenario	Description
Simple	1	A single-source and simple-query push down to the source container ▪ Simple contact profile
	2	A joining of a small table to a big table, applying a filter on the key column ▪ Low-volume contact profile data from one source and high-volume web interaction data from the other source ▪ Filter by a contact identifier (a column existing in both the source and join columns)
	3	A joining of a small table to a big table, applying a filter on a column from a table in the enterprise data warehouse (EDW) to check the data virtualization optimization capability ▪ Low-volume contact profile from one source and high-volume web interaction from the other source ▪ Filter by a contact name that exists only in the EDW
Medium	4	A joining of a small table to a big table, applying a filter on a column from a table in the EDW ▪ Build a trend relating contact profile from one source and high-volume web interaction data from the other source ▪ Filter by a contact name that exists only in the EDW ▪ Run multiple scenarios for different contacts (different data volumes and number of rows returned)
	5	A joining of a small table to a big table, applying a filter on a column from a table in the EDW; high-volume testing to build a trend ▪ Contact profile from one source and high-volume web interaction data from the other source ▪ Filter by a contact name that exists only in the EDW ▪ Run multiple scenarios for different contacts (different data volumes and number of rows returned)
Complex	6	A simple and complex query with aggregates (pre-calculated summary data derived by performing a "group by" query that creates a simple summary table) ▪ Summary information for a contact and an organization ▪ Data comes from both sources ▪ Simple aggregates to complex aggregates of sales and activity data
	7	The union of two sources (Fact Table) ▪ Merge (union of) two different transactions (web interaction and sales activity) coming from two different sources
	8	Rank Function ▪ Obtain the top 10 contacts of an account based on an engagement score derived from the number of web interactions

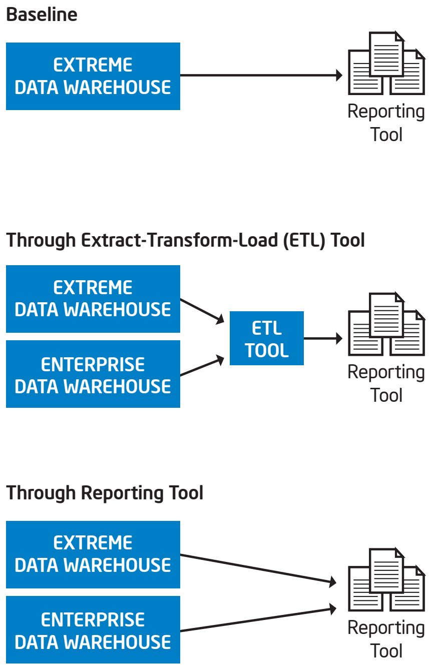

Figure 3. Intel IT tested two data virtualization methods drawing from multiple sources against a baseline approach applying data virtualization to co-located data. This baseline enabled us to measure the performance overhead of data virtualization. In each setup, we created our reports using the web report capabilities of the same reporting tool.

Optimizing Data for Virtualization

IT departments can use various optimization techniques to tune the performance of a data virtualization solution. One data optimization technique is to push down processing to the source container. Since pulling copious records into a data virtualization layer can have a major impact on performance, placing a structured language query (SQL) filter in the source container can reduce the number of records that will need to be pulled from a container to improve overall performance. Examples of query operations that can be pushed down include string searches, comparisons, local joins, sorting, aggregating, and grouping into the underlying relational data sources.

Co-location can also be thought of as an optimization technique for large data joins. Joining data from two databases in the same container is faster than joining data in two databases each located in a different container. The disadvantage of co-location is having to copy a large data set to a database and then copy this database to the same container in which the other targeted database resides. Copying data takes time and uses valuable storage space. Copying also introduces errors and creates redundancies and inconsistencies that can compromise the data's integrity.

RESULTS

Data virtualization proved an agile solution with both of our tools, requiring only one week to set up compared to the typical eight weeks it takes to implement a co-location solution. In performance, data virtualization posted better times than single source co-location in scenarios where we merged data from two data sources and used filters to push down processing to the two data sources.

Compared to the time it takes to model, design, develop, and test a co-location setup, data virtualization is much easier and less expensive to set up, test, and put in operation. This makes it a promising solution for situations where a business group needs a fast reporting solution. In addition, once a data virtualization solution is in place, adding a connection to a new source in the virtual layer is quick and easy.

For performance, results were mixed (see Figure 4). Data virtualization when implemented with our reporting tool had little performance impact for simple scenarios and for some complex scenarios where native source capabilities minimized processing at the data virtualization layer. In these cases, performance was similar to the co-location solution baseline. The graph even shows some results (the points that fall below the time axis) where the data virtualization layer outperforms the baseline. In other cases, where processing was done at the data virtualization layer rather than pushed down to the source container, we saw a performance impact.

Figure 5 provides a different view of the same data that highlights the points where the reporting tool's data virtualization solution, drawing data from multiple sources, performs better than the baseline co-location solution pulling data from the XDW only. When distributing the processing load between

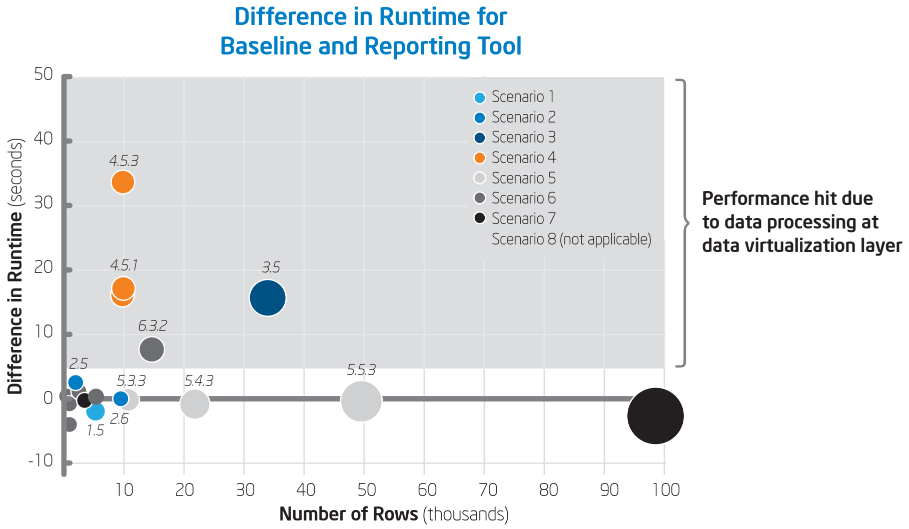

Figure 4. This graph compares the report runtime performance of a data virtualization layer implemented by a reporting tool with our baseline co-location solution running on our extreme data warehouse (XDW). The shaded area shows the performance impact encountered in scenarios 3, 4, and 6 when data processing and filtering is performed at the data virtualization layer. NOTE: The larger the size of a scenario's circle, the greater the number of records pulled. The numbers following the first number in each scenario, such as the "5.3" in "4.5.3," refer to differences in data volume and filter criteria.

two sources for scenarios such as union, this data virtualization solution outperforms the baseline. This shows the benefit of pushing down the processing (having the source systems filtering the data before it is joined) instead of trying to do all the processing in one place (single source). The more processing that can be pushed down to the source containers, the higher the overall performance.

For the most part, the two virtualization solutions—for the ETL tool and for the reporting tool—recorded similar results. The exceptions were instances where the ETL tool's data virtualization solution performed aggregation (doing a group-by-query operation to create a simple summary table) and joined the data in the virtualization layer as opposed to pushing this processing down to the sources (Figure 6). In these cases, the ETL tool's performance suffered.

CONCLUSION

While only certain scenarios in our PoC benefited from our current data virtualization solutions, the growing importance of implementing agile solutions for extracting business intelligence from big data and traditional enterprise data suggests that such methods will grow in importance and sophistication. Our findings provide initial guidance for the use of data virtualization at Intel and are driving our interest in investigating dedicated data virtualization solutions.

Based on our PoC results, when faced with a need for data integration across multiple data containers, we will now try data virtualization solutions, using rapid prototyping to see if virtualization can provide business value. If it can, we will expand its use accordingly. If we experience performance bottlenecks, complex transformations, or other issues that impact the desired results, we will switch to a co-location solution.

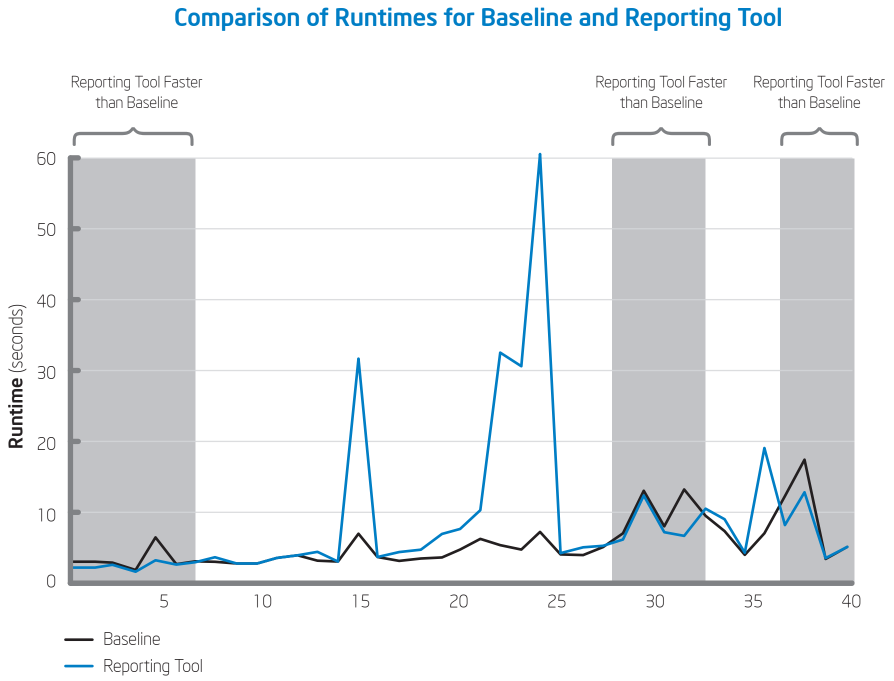

Figure 5. The grey shaded areas show where the reporting tool's data virtualization solution, pushing processing down to its two sources, performs faster in report runtime than the baseline co-location solution processing data from a single source, the extreme data warehouse.

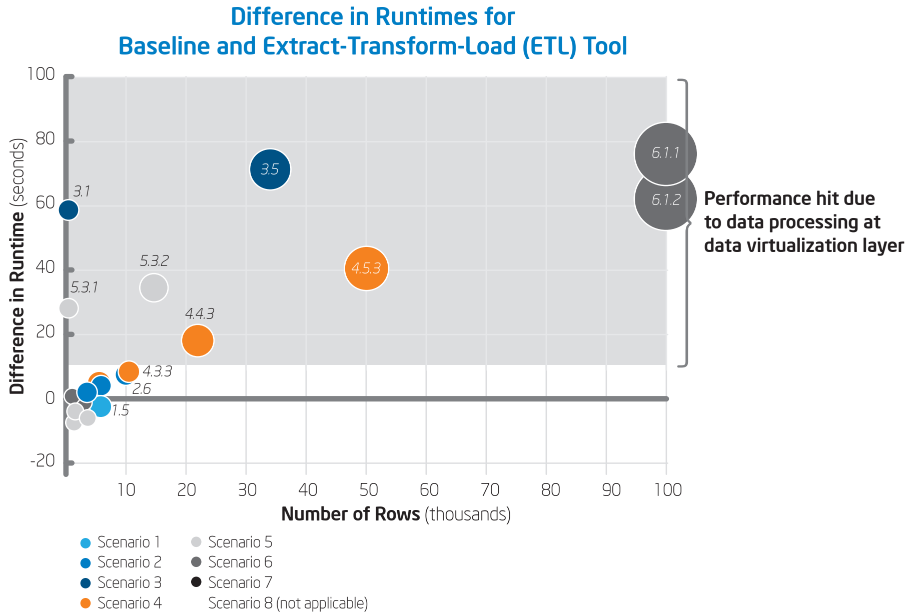

Figure 6. The grey shaded area shows the scenarios where the extract-transform-load tool's data federation functions processed data in the virtualization layer instead of pushing processing down to the sources. These scenarios had inferior performance compared to the baseline co-location solution processing data from a single source, the extreme data warehouse. NOTE: The larger the size of a scenario's circle, the greater the number of records pulled. The numbers following the first number in each scenario, such as the "5.3" in "4.5.3," refer to differences in data volume and filter criteria.

When using data virtualization to pull data from multiple containers, we see two choices with our current solutions:

- If we're using a reporting tool as the BI solution, we will employ the data federation capabilities of this tool across the containers. To improve performance, we will use optimization algorithms that push down processing, enabling us to retrieve the least data possible from each data source for final processing in the virtual layer.
- If consumption is by front-end tools other than our reporting tool, we will use our ETL tool's data virtualization capabilities.

To improve the performance of both these approaches, we are working with our suppliers, sharing our findings on performance issues. Meanwhile, Intel IT project teams are exploring the use of data virtualization in a limited capacity, particularly as implemented through the reporting tool. We anticipate our use of data virtualization will evolve along with the data virtualization capabilities of the tested tools.

We also continue to evaluate industry-standard dedicated data virtualization tools, with the expectation that using one or a mix of data virtualization solutions will enable us to increase the agility of our business intelligence, enable business groups to solve business problems more quickly, and help improve Intel's business results in each of our markets.

FOR MORE INFORMATION

Visit www.intel.com/it to find white papers on related topics:

- "Enabling Big Data Solutions with Centralized Data Management"
- "Integrating Apache Hadoop* into Intel's Big Data Environment"
- "Using a Multiple Data Warehouse Strategy to Improve BI Analytics"

ACRONYMS

BI	business intelligence
EDW	enterprise data warehouse
ETL	extract-transform-load
PoC	proof of concept
XDW	extreme data warehouse

For more information on Intel IT best practices, visit www.intel.com/it.

Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, reference www.intel.com/performance/resources/benchmark_limitations.htm or call (U.S.) 1-800-628-8686 or 1-916-356-3104.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Printed in USA ♻ Please Recycle 0913/ACHA/KC/PDF 328922-001US