Frequently Asked Questions about In-Database Analytics

Sponsored by:

TERADATA
Raising Intelligence

By Seth Grimes

Alta Plana

Table of Contents

The Case for In-Database Analytics 3

In-Database Analytics Basics 4

1. What is in-database analytics? 4
2. Why execute analytical computations in the database? 4
3. How does in-database compare with other analytics approaches? 4
4. Does every database management system support in-database analytics? 4
5. What are the business benefits? 4
6. Is in-database analytics new? 5
7. What is parallelization? 5
8. Is parallelization the same as MapReduce? 6
9. I've heard the terms "data-application server" and "analytics server." How do these notions compare to an in-database approach? 8

Using In-Database Analytics 7

10. How do I use embedded analytics routines? Can I use SQL? 7
11. Is in-database an all-or-nothing proposition? 7
12. Can in-database analytics co-exist with external systems such as Hadoop? 7
13. What problems are best run in-database? 7
14. What performance gains can I expect? 7
15. What if I'm using analytics tools and packages such as SAS? 8
16. What genres of analytics can be database-embedded? 8
17. Is this approach reliable and scalable as my data volume, number of users, analytical complexity, and business requirements evolve? 9

In-Database Analytics Development 10

18. Do I have to hire developers to use in-database analytics? 10
19. What should I look for in a DBMS programming framework? 10
20. If I want to develop my own database-embedded code, how do I go about it? Can I use a standard integrated development environment (IDE) and tools such as Eclipse? 10
21. Will I need to re-architect my analytics? 10

Deploying In-Database Analytics 11

22. How have real-world organizations deployed and benefitted from in-database analytics? 11
23. Do I need to adopt a special database schema or data model? 11
24. How do I monitor and manage the execution of analytical routines in-database? 12
25. What about security and reliability? 12
26. Can I do in-database analytics on the Cloud? 12

In-Database Analytics Evaluation and Selection 13

27. What are my options? 13
28. What are important technical considerations in selecting a platform for in-database analytics? 13
29. How about the business side? What should I look for in a vendor? 13
30. How long does it take to migrate to in-database analytics? 13
31. What are my next steps? How can I learn more about in-database analytics? . . . 13

The Case for In-Database Analytics

***In-database analytics* describes moving analytical computing – model scoring, predictive analytics, geospatial analysis, BI functions – into the database system, close to the data.** While the most visible advantage of database-embedded computations is higher performance – a very significant benefit – shops that adopt in-database analytics also win manageability, reliability, and security benefits that are by-products of centralizing analytical processing.

Moving *in-database* responds to growing demand for high-throughput, operational analytics for needs such as fraud detection, credit scoring, risk management, and on-line commerce in the face of ever growing data volumes and complexity. The technology builds on end-user database programming facilities that first emerged in the early 1990s. Enhanced frameworks and application programming interfaces (APIs) deliver greater ease-of-programming and use than ever before while allowing analytical computations to take advantage of modern, parallel database architectures.

Application and Implementation

Users have demonstrated the value of in-database analytics in addressing diverse business challenges, but really the technology can be applied beneficially anywhere there is a large volume of data, a critical need to minimize data movement and speed processing and response times, and a desire to move toward managed enterprise analytics. But the technology is not a panacea; you need to understand where, why, and how to use it.

The move in-database is not an all-or-nothing proposition. User organizations do have the option of exploiting the technology for individual applications or functions based on business priorities and on the availability of database-optimized versions of software from analytics providers. Migration can be gradual and should provide measureable performance and analytics-manageability ROI as well as enhanced analytics capabilities: the possibility of doing more with your data.

Frequently Asked Questions

The analytical database market has been transformed in the past two years with the realization that the performance and capacity of database management system (DMBS) architectures designed for transaction processing will not scale to support the modern enterprise data warehouse (EDW), which may host thousands of users and data volumes in the hundreds of terabytes with few limits in sight. Larger, well-established analytical DBMS vendors now face competition from newer entrants that have built competitive solutions around scalable database technologies that include, in some instances, analytics embedding.

This Frequently Asked Questions document will help in-database analytics users – current and prospective – cut through the clutter of competing definitions and claims. It is designed to explain the technology and, even more important, where and how it can be used to deliver business advantage. **It addresses key questions –**

> ***What is meant by in-database analytics? What varieties of analytics run best in the database? What types of problems have organizations applied in-database analytics to? How do you choose among implementation options?***

– to create clarity around the where, why, and how of in-database analytics.

In-Database Analytics Basics

We start with in-database analytics basics – definitions, rationale, benefits, and a bit of history:

1. What is in-database analytics?

In-database analytics describes analytical computations executed in the database management system (DBMS) rather than in a desktop program or application server. These computations, also known as *embedded analytics*, are coded in a programming language, whether by the end-user organization or by a third-party analytics provider. They are made available through a database query language such as Structured Query Language (SQL) and other database interfaces. They cannot typically be written directly in industry-standard versions of SQL.

In-database analytics means creating a smarter enterprise database warehouse (EDW) that boosts enterprise analytics performance, agility, scalability, and manageability.

2. Why execute analytical computations in the database?

Analytical computations – key performance indicator (KPI) calculations, data aggregations, model scoring, predictive analytics, temporal and geospatial analysis, and other functions – often touch large numbers of data records and multiple data fields. They may be carried out repeatedly. Executing these functions in the database, close to the data, avoids data movement, speeding analytics performance and increasing throughput. Further, centralizing analytics in the database facilitates version control, reduces duplication, and extends the DBMS's management, security, and auditing infrastructure to analytical data and functions.

3. How does in-database compare with other analytics approaches?

Calculations are calculations, whether executed outside or inside the DBMS. Numerically, the results will be the same.

With in-database analytics, the calculations are managed and executed in a single, centralized location, available to any user or program that needs them (subject to access permissions). Because they are exposed through SQL, they may be executed through direct SQL queries or via any front-end application or user interface that accesses the database.

End users may not even be aware that their analyses involve database extension or that their analytical software is running calculations in the database, except indirectly given performance improvements.

4. Does every database management system support in-database analytics?

No. For in-database analytics, you need a DBMS that can be extended with custom computations, typically in the form of user-defined functions (UDFs). Of course, any DBMS vendor's developers can extend the vendor's own system, but you don't want to be limited to vendor-created or -sanctioned extensions. For generalized, robust in-database analytics, the DBMS vendor must provide a framework that executes computer code created by third-parties including end-user organizations and specialist analytics providers.

Note that we're referring to analytical computations that go well beyond the standard SQL that is implemented by most systems, including the SQL ranking, CUBE, and ROLLUP functions that are available with many DBMSes although not with popular systems such as MySQL (excepting ROLLUP).

5. What are the business benefits of in-database analytics?

The business benefits boil down to cost savings and improved execution through higher efficiency, security, mission support, manageability, and usability.

- **Efficiency**: The performance boost means more bang for the hardware buck, and it means faster results delivery to users. Further, the entire analytical development and scoring process can be streamlined, allowing businesses to better use their IT and analytic personnel. Instead of performing the iterative cycle of extract, consolidate, integrate, and analyze or score, staff can simply analyze or score directly in the database.

- **Security and control**: The centralization of analytical processing in the database eliminates the security vulnerability associated with data movement and adds the ability for management to control, monitor, and audit analytics use.
- **Mission support**: The possibility of extending the database system with proprietary or specialized analytical methods creates competitive advantage. These extensions may be created by the user organization itself or through vendor partnerships with specialized, industry-leading vendors such as SAS, KXEN, and SPSS (now owned by IBM) that make proven methods available without forcing user organizations to program the routines themselves.
- **Usability**: Analytics routines become available to end users and developers in the form of extensions to industry-standard SQL and via common APIs. Users are shielded from exposure to complex algorithms.
- **Manageability**: Moving routines into the database eliminates duplicate programming and maintenance of analytics routines and eases version control.

6. Is in-database analytics new?

No. In-database analytics is based on the ability to extend the underlying DBMS with special analytical functions. In the mid-1990s, Teradata Database was one of the first commercial systems to support end-user DBMS extensions to support in-database mining. Object-relational database systems from Illustra (folded into Informix, now owned by IBM), Oracle, and IBM also brought these capabilities to their own users, including through extension "data blades" and "cartridges" created by third-party vendors. These latter systems, however, did not focus on analytical processing, nor did they provide for analytics parallelization.

With the growing popularity of analytical database systems, in-database analytics has lately become a must-have capability.

7. What is parallelization?

Parallelization is the ability to decompose a larger workload into smaller tasks that are carried out simultaneously. For instance, individual school teachers take attendance for their classes and report numbers to the school office, which tallies them for the whole school. That's an example of parallelization, in particular the execution of a single task in multiple locations with distinct data subsets.

Parallelization can help with "big data" and can speed the overall execution of larger tasks and workloads. Data-related tasks that can be parallelized include database loading, index building, queries over large data sets, and many analytical computations. Database systems researcher Joe Hellerstein recognized "inter- and intra-query parallelism"[1]: the former being the simultaneous execution of a query or transaction for multiple sets of data elements, the latter the decomposition of a given query or transaction into components that are executed in parallel with results assembled on completion of the subtasks.

DBMS pioneer Michael Stonebraker, referring to "automatically providing parallel execution on a grid of computers," recognized that "commercialization of these ideas occurred in the late 1980s with systems such as Teradata."[2] Teradata Database is indeed an example of a parallel database system, exploiting a "shared-nothing" compute grid, complemented by Teradata Parallel Transporter, which provides scalable, high-speed, parallel data extraction, loading, and updating.[3] Red Brick Warehouse, now owned by IBM but no longer widely used, is a parallel analytical DBMS that dates to the 1990s. These systems are distinguished by the integration of parallelization into mature query-optimization facilities.

1 Joe Hellerstein, "Parallel DB Primer," 1999, *http://www.eecs.berkeley.edu/~demmel/cs267_Spr99/Lectures/parDBprimer.ppt*. This presentation includes 1999 Teradata technology profile.

2 David J. DeWitt and Michael Stonebraker, "MapReduce: A major step backwards," January 17, 2008, *http://databasecolumn.vertica.com/database-innovation/mapreduce-a-major-step-backwards/*

3 "Teradata Parallel Transporter Tips and Techniques," *http://developer.teradata.com/tools/training/teradata-parallel-transporter-tips-and-techniques*

8. Is parallelization the same as MapReduce?

MapReduce, as made famous by Google for text processing over large computing clusters, is one of several approaches to parallel processing. Implementations may run on top of a file system – Apache Hadoop's open-source MapReduce implementation is one example – or they may run in-database. They do not, in themselves, guarantee data security, queryability, or system high availability. Hadoop, for instance, has the vulnerability of a single point of failure for its distributed file system. While DBMS pioneer Michael Stonebraker went so far as to call Hadoop "a giant step backward in the programming paradigm for large-scale data intensive applications," implementations since 2008 that move MapReduce into the DBMS have addressed the problems he and co-author David DeWitt described.

9. I've heard the terms "data-application server" and "analytics server." How do these notions compare to the in-database approach?

"Data-application server" is simply one analytical DBMS vendor's term describing the vendor's own database management-analytics platform. Other systems that support advanced in-database analytics similarly function as data-application servers.

An "analytics server," by contrast, involves software (and possibly hardware) that executes analytics outside the database. Like the in-database approach, an analytics server can help centralize analytics. However, it does add components to the overall computing environment that may, when compared with the in-database approach, complicate manageability, add points of failure, affect the freshness of data available to users, and diminish analytics performance.

Using In-Database Analytics

10. How do I use embedded analytics routines? Can I use SQL?

Yes, you can use SQL. One of the key points is the reduction of complexity, making basic and advanced analytical calculations available through standard SQL, both directly and via familiar query, reporting, and analysis applications that use SQL to access the database.

Really, you can use any user interface (UI) or API supported by the DBMS to use embedded analytics routines.

11. Is in-database an all-or-nothing proposition?

No. You can move as much or as little of your analytical processing into the DBMS as you wish, at your own pace, parallelized or not.

If your analytics vendor provides an in-database option for back-end, computational components, you may not need to take any active steps yourself to benefit from the technology.

12. Can in-database analytics co-exist with external systems such as Hadoop?

Certainly, with the right database system with the necessary integration hooks. These might treat an Apache Hadoop instance, or a similar system, as an external data store, accessible via an in-database UDF or a similar mechanism, making the external data store, for example the Hadoop Distributed File System (HDFS), look logically like an internal database table. Use of the external data store then becomes transparent to DBMS users, who can analyze HDFS data in conjunction with DBMS-managed data. Users get the benefits of Hadoop's (or a similar system's) parallel processing coupled with a reliable, robust, high-performance data-management and analytics platform.

13. What problems are best run in-database?

In-database is great for "big data" problems and for others that can benefit from a centralized, managed analytics architecture. The following is a sampling of the types of analytical problems that are best run in-database:

- Data intensive calculations, whether simple or complex. The act of moving data costs compute and network resources, creates security vulnerability, and takes time.
- Recurrent calculations that repeatedly touch the same data tables and value. Moving smaller amounts of data repetitively is as costly as moving big data less frequently, especially given that commercial-grade DBMSes have effective query optimization and data caching mechanisms.
- Cross-functional analytics, for instance, applications that touch both geospatial and transactional or customer-profile information. With a variety of types of analytics hosted by the DBMS, it becomes easy to join analytics across what may otherwise have remained "stovepipe" applications.

14. What performance gains can I expect?

Naturally, there's no fixed performance multiplier. Gains due to the elimination of data movement from database to application server or desktop will be in proportion to data volume. Where calculations are concerned, there's the possibility of much greater gains. Why? While it is tempting to say that a compute cycle is a compute cycle, whether run in the DBMS or on the desktop, modern DBMSes have powerful query optimization and caching mechanisms, and the ability to parallelize many operations (per Q7), that are not typically available to or present with analytics options other than in-database.

As a rule of thumb, for "big data" analytical computations that can be parallelized for in-database execution, expect performance improvements that may reach 10X to 100X. As an added benefit, you can also expect faster "speed-to-insight" given the efficiencies possible with centralized, in-database analytics management and execution.

Solution-provider literature will offer examples linked to real-world business scenarios. For instance, a Teradata Warehouse Miner brochure describes a customer that "had previously spent 312 hours to run five models against 250 million rows, extracting the data into an analytic server... Teradata converted these models into SQL models leveraging Teradata's in-database mining techniques. All five analyses ran in only 12 hours."[4] Any credible vendor will provide similar case-study examples.

15. What if I'm using analytics tools and packages such as SAS?

You're in luck. Leading analytics vendors including SAS have been working with Teradata and other DBMS vendors to move analytical processing from the application layer into the database. SAS's relationship with Teradata is the most advanced of any of its DBMS vendor relationships for in-database analytics.[5] That is because SAS has rewritten many basic procedures (Procs) and functions, in addition to complete applications such as SAS Scoring Accelerator[6], to run inside the Teradata® Database. These database-embedded functions support data preparation, modeling, and analysis tasks and are used automatically by some SAS user-facing applications when appropriate. While SAS has created SAS/Access connectors for many database management systems, SAS/Access does not cause SAS Procs or functions to be executed in the DBMS.

Leading analytical DBMS vendors typically partner with multiple, specialized analytics providers and work with the members of their partner networks to optimize the partners' software for in-database execution. These partners may support financial, marketing, sales, law-enforcement, and other analyses utilizing transactional, operational, time-series, geospatial, textual, and other data, depending on the robustness, extensibility, and capacity of the DBMS platform.

16. What genres of analytics can be database-embedded?

The only significant limitation on database-embedding of different genres of analytics relates to support for the data types (and associated indexes, functions, and methods) the analytics genres rely on. For example, it makes little sense to seek to run location intelligence work in-database if the DBMS doesn't support geospatial data types, preferably as native types.

One should, of course, look for standards compliance, for instance, when working with geospatial data, of the SQL Multimedia (SQL/MM) spatial standard, which includes Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, and GeometryCollection types. Continuing with this example, "the methods defined in the SQL/MM specification are provided and can be grouped into four categories: attribute, spatial, topology interactions, and measurement. Spatial methods return a new geometry object and calculate intersection, boundary, or the distance between locations, etc."[7]

4 Teradata Data Mining Warehouse Solution, *http://www.teradata.com/t/brochures/Teradata-Data-Mining-Warehouse-Solution-eb1897/*

5 James Taylor, "First Look – SAS In-Database Analytics," *http://jtonedm.com/2010/04/12/first-look-sas-in-database-analytics/*
The integration is further described in the Teradata magazine article, "Reduce movement, gain speed," *http://www.teradata.com/tdmo/v08n01/Tech2tech/AppliedSolutions/ReduceMovement.aspx*

6 SAS Scoring Accelerator, *http://www.sas.com/technologies/analytics/datamining/scoring_acceleration/index.html*

7 Alan Greenspan, "Location, Location, Location," Teradata Magazine, *http://www.teradata.com/tdmo/location_location_location.aspx*

17. Is this approach reliable and scalable as my data volume, number of users, analytical complexity, and business requirements evolve?

Yes. Arguably, database and analytical processing scale better with in-database analytics than without, in particular, as data volumes grow, with mixed or homogeneous workloads, large numbers of users, and evolving business requirements.

First, leading-vendor database software infrastructure is designed to scale as these conditions change. User organizations can typically simply add nodes and storage to clustered or grid systems. Because in-database calculations are managed within this same scalable infrastructure, analytics capacity will similarly scale with the addition of hardware. There are added benefits with in-database analytics, however. Data movement is reduced, which lessens the hardware and performance impact of "big data" and reduces security risks. Analytics deployment is centralized, which simplifies development processes and configuration and user management. Leading-vendor systems provide out-of-the-box analytics routines that simplify application development and boost analytics usability. These factors mean that the in-database approach will likely scale better than traditional analytics approaches.

In-Database Analytics Development

18. Do I have to hire developers to use in-database analytics?

Not necessarily. You don't need developers to use vendor- or third-party-supplied in-database routines.

If you create your own analytics routines, the developers you have on staff would simply deploy those routines to run in-database, within the DBMS vendor's in-database programming framework. Of course you'll want to maintain and adapt your established design, testing, and quality assurance processes and to extend your database security framework to the new routines.

19. What should I look for in a DBMS programming framework?

The assumption, of course, is that the framework is licensed to end users for the development of DBMS extensions.

Optimally, the framework will support the user's choice of programming language. It will support programming around a full range of data types, base and extended, including text, geospatial, and temporal. Preferably, these types will be treated natively, not as external patches to the DBMS. The framework will support parallel programming – the ability to break a calculation down into multiple threads that execute simultaneously on multiple processor cores – and be integrated with the DBMS's query optimization subsystem.

The framework should support "thread safe" execution with isolation of individual processes so the coding issues will not take down the whole system. It should provide developer "sandboxes" that segregate non-production and production code and/or data.

Further, it can be very helpful if the vendor provides software development tools, a test harness, and performance measurement and monitoring interfaces or tools. Other assets include user groups, developer forums or exchanges, strong training materials, and a well-developed partner network.

20. If I want to develop my own database-embedded code, how do I go about it? Can I use a standard integrated development environment (IDE) and tools such as Eclipse?

Modern software development is typically conducted via an IDE that encapsulates a continuous-integration methodology for agile, iterative development and deployment. It is almost always possible, however, even if not recommended, to develop and deploy code the old-fashioned way, using a text editor and textual commands.

Leading vendors including Teradata support database-application development through the Eclipse open-source IDE, an "open development platform comprised of extensible frameworks, tools, and runtimes for building, deploying, and managing software across the lifecycle."[8] Typically, the DBMS vendor will provide a freely downloadable Eclipse plug-in that utilizes the Eclipse plug-in framework and, perhaps, the Eclipse Data Tools Platform (DTP).

21. Will I need to re-architect my analytics?

If your home-grown analytics routines run serially, and if your DBMS supports parallel in-database execution, you may wish to rewrite your routines for parallel execution, which can greatly accelerate analytics performance and throughput.

You may also wish to rebuild your routines to rely on vendor- or third-party-provided analytics components designed to run in your DBMS platform. This move would allow your organization to focus on higher-level, higher-value application development by looking to a supplier for the development and support of basic application components.

In either case, analytical processing carried out in desktop and server applications will be migrated into the EDW, allowing you to simplify those user applications.

8 About the Eclipse Foundation, *http://www.eclipse.org/org/*

Deploying In-Database Analytics

22. How have real-world organizations deployed and benefitted from in-database analytics?

The best way to understand how real-world organizations have deployed and benefitted from in-database analytics is to look at examples, at typical user stories. Here are three, drawn from published materials:

- *Banking Technology*, in the April 13, 2010, article, "BoA cuts AML transaction processing time by ten hours," reports, " **Bank of America has cut its processing time for anti-money laundering by ten hours using in-database analytics.** Having initially used the system to tackle the growth in money laundering processing, it is now being used across the bank... Julie Mannor, senior technical developer in Enterprise Information Management at Bank of America... worked for the anti-money laundering team, and it saw challenges to the existing set up. 'A year ago, we were running 1.4 billion records across the network in a night, and the process ran for about 14 hours.'[9] The process in fact now runs at 4 hours... The Informatica system that had been used to extract and load data from Teradata into SAS was no longer needed, SAS data stores no longer needed to be built, and the security protocols no longer needed to be recreated."
- Analytics leader SAS reports that at specialty retailer Cabela's, prior to moving to in-database analytics, "data for modeling and scoring customers was stored in a SAS data mart. This process required a large amount of time to construct, bringing together disparate data sources and keeping statisticians from working on analytics. On average, the statisticians spent one to two weeks per month just building the data."[10] **Analytics-database integration "enables Cabela's to bring its data close to its analytic functions in seconds versus days or weeks**. It can also more easily find the highest-value customers in the best locations most likely to buy via the best channels. The integrated solution reduces the need to copy data from one system to another before analyzing the most likely indicators, allowing Cabela's to run related queries and flagging potentially ideal new prospects before the competition does." The SAS case study reports Cabela's move to productionalize "**analyzing the clickstream patterns of customers shopping online... to put the perfect offer in front of the customer based on historical patterns of similar shoppers**." According to Dean Wynkoop, manager of data management for Cabela's, "This **would not be possible without the in-database processing capabilities** of SAS, together with Teradata."
- An *IT World Canada* article, "Taking the sting out of forecasting,"[11] describes how Warner Home Video dramatically improved home-video sales forecasts by "cleverly combining the strengths of two key data mining/data warehousing technologies." Warner reportedly reduced the run-time for new-release forecasts from 36 hours to one hour and 15 minutes and for running a specialized ranking model from two hours to one minute… and that in 2006. Warner has continued its move to in-database processing, for instance by rewriting a complex depreciation function in SQL for in-database (Teradata) execution (an eight-hour rewrite) and coding "interactions and relationships" analyses in SQL, in both cases eliminating data movement across the network and centralizing analytics management. These steps have also allowed Warner to eliminate the need for special analytical data sets.

23. Do I need to adopt a special database schema or data model?

No. Design your database schema the way you would normally for whatever operational and analytical processing you'll be doing. You may have a normalized schema, or you may prefer a star- or snowflake-design that implements a dimensional data model. The key is design for performance, throughput, and maintainability given your processing's data-use profile.

9 *http://www.bankingtech.com/bankingtech/article.do?articleid=20000176982*

10 "SAS helps Cabela's reel in more customers," *http://www.sas.com/success/cabelas.html*

11 *http://www.itworldcanada.com/news/business-intelligence-taking-the-sting-out-of-forecasting/100373*

24. How do I monitor and manage the execution of analytical routines in-database?

DBMS extensions – user-defined functions or specialized routines such as Teradata INMODs (INput MODules), when properly installed, essentially become part of the DBMS, part of the EDW. A robust DBMS will automatically account and allow for these extensions within the scope of utilization, performance monitoring, user management, and auditing tools and also cover them within the DBMS's query execution planning, scheduling, and optimization facilities.

25. What about security and reliability?

Enterprises must secure sensitive and mission-critical data regardless where application code resides and is executed. If the code runs outside the DBMS, the security of each application will need to be assured, tested, and monitored. But if it runs inside the DBMS, only results, and not the data from which they're computed, will be exposed outside the database, mitigating risk and reducing or even eliminating the need to test and monitor each application.

There is very little added cost in extending DBMS security mechanisms to in-database application components. It is, however, important to choose a database platform with adequate security mechanisms. These will minimally include the ability to assign data-access privileges to users, groups, and roles and go beyond to offer robust backup and recovery mechanisms and options for data encryption.

26. Can I do in-database analytics on the Cloud?

Certainly. If your data is cloud-hosted, by moving analytics into your cloud-hosted system, you will avoid the performance, latency, and security hit created by transmitting data across the Net to non-cloud systems for analysis. And by co-locating data and analytics on the cloud, you will benefit from the full set of manageability, on-demand provisioning, cost control, and speed-to-use advantages that are associated with cloud computing. Of course, the cloud DBMS you are using must support embedded analytics.

In-Database Analytics Evaluation and Selection

27. What are my options?

End users are fortunate: the marketplace offers a variety of choices that include established vendors and start-ups. A starting point is to look at options available with your current analytical DBMS vendor. If your current provider cannot respond to your organization's technical and business concerns, you will want to survey the market to identify alternatives. You can find a variety of helpful resources via a Web search.

28. What are important technical considerations in selecting a platform for in-database analytics?

It makes no sense to move analytics into a DBMS that isn't already a strong analytical database management platform. That would be, colloquially speaking, putting lipstick on a pig.

If your DBMS does not provide the capacity, performance, throughput, scalability, reliability, and manageability you need for current and foreseeable analytical processing, first look for a new DBMS. If your DBMS meets these criteria – or once you've put in place a DBMS that does – the next step is to assess its database programming framework and tools, per Q19 and Q20. Does the framework support programming languages of interest, and does it isolate analytics processes to ensure high availability and reliability? Does the vendor provide programming, test, and management tools that support best-practices development methodologies? Can you add extensions that parallelize processing (per Q7) and take advantage of data types and associated methods of interest such as geospatial (per Q16)?

29. How about the business side? What should I look for in a vendor?

Business considerations and vendor qualities that relate to in-database analytics are really no different from considerations and quality for any analytics/data-management solution provider. First, is the vendor reliable? Does the vendor have satisfied customers and a track record delivering analytical database solutions that include, in this instance, in-database analytics options? Optimally, the vendor will have experience and perhaps focused solutions relevant to your business domain, business problems, and technical environment. Moving out a step, has the vendor developed, and does the vendor support, an ecosystem of solution, services, and technology partners that complement its own capabilities? Lastly, active user and developer communities indicate a strong vendor and can provide an essential resource for organizations that are new to in-database analytics or, for that matter, any aspect of analytical databases and enterprise data warehousing.

30. How long does it take to migrate to in-database analytics?

Migration to in-database analytics may take you no time at all: that in the case where your analytics provider has moved behind-the-scenes application components into the DBMS. The provider will have redesigned the user-facing components accordingly to automatically execute analytical processing in-database.

If you build your own analytics code and wish to move it in-database, the process is not necessarily time consuming. You can take a phased approach where you test the waters with selected routines before jumping in fully. The migration process does offer an opportunity to refactor your code – to optimize it to run efficiently within the target DBMS – and to consider replacing analytics primitives (common, lower-level routines) with vendor- or third-party supplied code. The overall migration process will be facilitated by adoption of continuous-integration methodology that supports agile, iterative development and deployment (per Q20).

31. What are my next steps? How can I learn more about in-database analytics?

Many vendors in this space have material on-line that explains their in-database analytics implementations. Look for objective articles published in the trade press, such as "In-Database Analytics: A Passing Lane for Complex Analysis"[12] by the author of this FAQ document, and by organizations such as the Data Warehousing Institute (tdwi.org). And look, in particular, for case studies published by the vendors and in the trade press that explain how user organizations implemented and benefitted from the technology.

12 Seth Grimes, "In-Database Analytics: A Passing Lane for Complex Analysis," *http://intelligent-enterprise.informationweek.com/showArticle.jhtml?articleID=212500351*

About the Author

Seth Grimes (*sethgrimes.com*) is an information technology analyst and analytics strategy consultant. He is contributing editor at *Intelligent Enterprise* magazine, founding chair of the *Text Analytics Summit* and the *Sentiment Analysis Symposium*, an instructor for *The Data Warehousing Institute (TDWI)*, and text-analytics channel expert at the Business Intelligence Network.

Seth has worked with database, BI, and decision-support applications and users for more than 25 years. He founded Washington D.C.-based Alta Plana Corporation in 1997. He consults, writes, and speaks about information-systems strategy, data management and analysis systems, industry trends, and emerging analytical technologies.

You can reach Seth at *grimes@altaplana.com*, +1 301-270-0795; *@SethGrimes* on Twitter.

Teradata Corporation

Teradata is the world's largest company solely focused on creating enterprise agility through database software, enterprise data warehousing, data warehouse appliances, and analytics. Providing powerful solutions for companies of all sizes, Teradata and its partners bring expert insight to your business through the highest-performing technology available, including:

- Appliances and Platform Family
- Enterprise Data Warehousing
- Active Data Warehousing™ Technology

Teradata
2835 Miami Village Drive
Dayton, OH 45342
Phone: +1 866-548-8348
Visit **Teradata.com**.

Active Data Warehousing and Raising Intelligence are trademarks and Teradata and the Teradata logo are registered trademarks of Teradata Corporation and/or its affiliates in the U.S. and worldwide.