

Scalable BI for Every Enterprise

How Open Source BI Meets Enterprise-Class Requirements.

Introduction

As Business Intelligence (BI) becomes more pervasive in enterprises large and small, the requirements placed on it—by end users, technology teams, business analysts and others—continue to evolve. Among these requirements, scalability emerged early, as organizations of all sizes realized the significant decision-making, opportunity-revealing and problem-solving value of BI.

Relevant facets of BI scalability are numerous and varied, but certain among them are common to nearly every enterprise. Because some of these scalability factors are tightly coupled with other requirements posed by growing enterprises, any examination of scalability must also address a range of other “enterprise-class” solution requirements.

For BI solutions to meet the needs of large organizations—or smaller ones with sizeable or variable reporting and analytics requirements—they must:

- Provide a minimum set of features and functionality, engineered robustly enough to stand up to the rigors of large, diverse, growing and fluid user communities, who themselves are served by busy IT groups
- Process large, fluctuating reporting volumes—both in number and size—based on large, diverse and sometimes volatile data sources
- Make optimal use of enterprise resources, both human and technological
- Help application developers and system administrators maximize their productivity and quality of output.

Not every BI provider seeks to serve the scalable enterprise, but those that do must meet these minimum criteria. It’s reasonable to assume that well known, established proprietary BI solutions meet these requirements, but the rapidly changing technology landscape may, in some cases, work against these solutions. The rise of open standards, migration to service-oriented architectures and the rapid adoption of commodity infrastructure s—not to mention the growing popularity of open source software—all pose challenges to established BI solutions created in previous technological eras.

This investigation exposes key scalability requirements for BI, and examines how the world’s most widely used BI solution—notably, one that’s open source—meets these requirements.

We'll address the requirements themselves, identify underlying BI design implications, and assess how well the Jaspersoft solution measures up.

Naturally, specific needs—and their relative importance—vary across industries and individual organizations. This report focuses on the most important—and universal—scalability requirements, and readers are encouraged to delve deeper as they assess their own organizations' specific needs.

Features and Functionality for Scalable Business Intelligence Solutions

In addition those features BI solutions must offer to adequately retrieve, format and present information to users—not addressed in this report—certain capabilities exist primarily for the purpose of enabling the solution to deliver large numbers of reports, many of them sizable and/or complex, to large numbers of users, without unduly impacting infrastructure and other applications.

Schedulers for Optimization. First, BI solutions must enable the user community to run reports on both an on-request and scheduled basis. Previously defined, on-request reports, when initiated, should not severely impact other processes and users; and there must be facilities for minimizing those impacts when they do occur. Scheduled reports must be similarly managed; for example, there must be facilities for managing large numbers of reports all scheduled for the same time slot (e.g., midnight).

Fine-grained Resource Control. Some queries and reports require large amounts of data and/or data sources; must perform complex computation, aggregation or summarization, or must produce complex output, output in large volumes, or both. Even “short,” highly summarized reports may require the retrieval of millions of rows of data. Business users, end users and even report developers may also threaten community resources by inadvertently launching so-called runaway queries.

BI solutions must provide facilities for creating these reports, but also for managing their use to avoid undue impacts on databases, other applications, bandwidth, client platforms, and other resources.

STAYING AHEAD OF ENTERPRISE NEEDS: THE JASPERSOFT BUSINESS INTELLIGENCE SUITE

Scalable BI is technology that's ready for and responsive to advancing enterprise requirements in two dimensions. First, enterprises typically widen their functionality requirements as they pursue various BI projects. They may be focused on production reporting today, for example, but realize an increasing need for embedded operational, dashboard, or analysis-focused reporting tomorrow.

Also, as an enterprise's technology landscape evolves, so may their requirements for interoperability, integration, and platform independence. In some circles, a BI solution can't be called highly scalable if it won't operate on virtually any kind of data structure and within a wide variety of technology stacks.

The Jaspersoft Business Intelligence Suite is positioned to meet these expanding needs. The suite's modular, standards-based solutions include:

- JasperReports: Advanced reporting engine—fully embeddable or accessible through a full-featured report server
- iReport: Graphical report definition toolset
- JasperServer: Full-featured, customizable report server with powerful report scheduling, ad hoc query, dashboarding and in-memory analytics capabilities
- JasperAnalysis: Highly evolved, OLAP-based Analytics solution for high-performance, in-depth analysis
- JasperETL: Easy-to-use extract / transform / load toolset for use in populating OLAP data structures, data marts and data warehouses.

The Jaspersoft suite is built on a 100%-Java architecture, can efficiently access virtually any kind of data structure, and supports a range of open integration approaches. As such, the solution serves as an ideal technology implementation for evaluating the scalability of open source BI.

Responsiveness. Enterprise users frequently develop information requirements that must be met very rapidly; vital operations or key business decisions depend on speed. Sufficiently scalable BI solutions must be able to do all of these things simultaneously:

- access and retrieve required data with reasonable efficiency
- quickly generate report content, even when lengthy and complex
- respond promptly to user requests (this may include mere notification that reports are in process, scheduled or queued)—even under large request volumes
- quickly and efficiently export output to user-requested formats such as PDF, HTML, Excel or CSV

Design Requirements for Scalable Business Intelligence Solutions

Governors. To protect the organization and its resources against runaway queries (including ad hoc reports, OLAP queries and otherwise innocuous, pre-defined reports that encounter anomalous data), BI solutions need query governors of varying types. Data-intensive queries can be managed with governors that limit row retrieval, while reports that generate unacceptably large output can be controlled with page-limiting governors.

Additionally, administrators need a way to cancel any runaway queries which do manage to launch (surprisingly, not all BI solutions offer a straightforward way to achieve this).

JasperServer enables administrators and developers to employ their choice of data (rows retrieved) and output (pages generated) governors in order to guard against runaway processes. Administrators with appropriate credentials can instantly stop any query, whether initiated by an on-demand or scheduled report, ad hoc query, or analytic (OLAP) query.

Virtualizers. Large-volume reports are a given in large enterprises (and abound in small ones). Scalable BI solutions must allow end users or report schedulers to run these reports to completion, regardless of their size, without undue impact. While schedulers and governors can be used to prevent such reports from causing problems at certain busy usage times, there must also be protection against memory overruns which could halt scheduled reports (and other processes using the same servers).

Because many BI report generators format output content in memory, memory-exceeded stoppages are a significant risk. Administrators can (and, in some cases, should) configure servers to reserve larger amounts of memory for reporting processes, but this alone won't accommodate the surprisingly common 20,000-page PDF.

Virtualizers—which reduce the amount of memory needed to store report output—are common in BI solutions, though some offer only certain types of virtualization and may not provide the flexibility needed for today’s diverse enterprise needs. Truly scalable BI solutions provide report developers the means to test and optimize virtualizer usage before reports are deployed to production environments.

It’s also reasonable to expect that large enterprises may have virtualizers already in place when they deploy a BI solution, and want to leverage their investment while minimizing complexity in the environment. While it’s likely that BI providers have more expertise in applying virtualizers to reporting processes, BI solutions should provide APIs that enable their solutions to access these third-party virtualizers.

JasperReports provides three distinct types of virtualization; report designers and administrators may choose the most appropriate kind for a given user load, reporting requirement and server environment.

- File virtualization stores report output to disk one page at a time (each page gets its own file) during the report fill process. Pages can be retrieved into memory for output export at the end of the fill phase.
- Swap file virtualization works similarly, storing fill-state output on disk—but all in a single, large file, written incrementally as the fill progresses. JasperServer can be configured to allocate initial and incremental disk space as required. This type of virtualization requires less physical I/O than File virtualization, and creates less overhead in the reporting process.
- GZip virtualization relies on compression in memory rather than paging report output to disk. The report output is compressed by a factor of approximately 10 to 1. This is the fastest, most efficient of the virtualization processes, but it may not be ideal for the largest enterprise report, which, even when compressed, may become too large to be held in memory.

Other key points: To help optimize memory usage, a single instance of JasperServer’s Swap File virtualizer process can be shared across multiple report processes. Additionally, iReport and JasperReports provides developer tools for tuning virtualization methods on challenging reports. Finally, JasperReports provides APIs designed to allow the use of non-Jaspersoft virtualizers.

Managing BI demand & growth.

Scalable BI solutions should accommodate high concurrent user counts and large volumes of queried data and report output. Relevant features can be found at compile, fill, and output stages of report creation.

Design and Compile Stage. Compiling reports isn't traditionally a scalability issue. However, some BI solutions recompile queries at report execution time for certain types of data sources, where query optimization is important, and compile times can become lengthy, adding overhead and delays to report execution. Scalable BI solutions store compiled (binary) reports and queries, enabling reports to enter the fill stage immediately upon initiation. To enable end-user flexibility (and to prevent larger-than-necessary retrieval and output), parameter value collection at run-time should be available; and compiled queries should be optimized, as much as possible, given the parameters specified.

High scalability also requires the leverage of data source-specific features; the BI solution that generates precisely the same SQL for Oracle, Microsoft SQL Server and MySQL data sources is not generating code native to each of these platforms, and may miss out on significant performance improvements.

Naturally, any scalable solution includes documentation (or other viable forms of guidance) to assist report designers in optimizing resource usage.

iReport, the graphical report definition tool, and JasperServer both create run-ready binary report objects. These JasperReports templates (JRXML files) are compiled when the report definition is created or changed, and are stored in the JasperServer repository, to be retrieved and executed as scheduled or on-demand processes. They begin the fill phase immediately once initiated.

Because JasperReports may be imbedded in other applications, it's possible that those applications might not take advantage of the pre-compiled, run-ready objects; but most development organizations make sure to use objects, not uncompiled report definitions or analysis queries, at run time, if anticipated volumes warrant.

Jaspersoft's "Ultimate Guide" documentation is useful for report designers wishing to optimize memory, processor, and data retrieval operations. The open source community also represents a valuable resource to report developers; in fact, this is one of the distinct advantages of open source and commercial open source solutions such as the Jaspersoft suite. Examples of this kind of guidance show report designers how to:

- Gain certain processor and memory efficiencies through the judicious use of what Jaspersoft calls subreports, explained fully in the documentation and discussed widely on community forums.
- Take advantage of the platform-native SQL generated by Jaspersoft for a number of popular database platforms.

- Specify known text field sizes, eliminating the need for runtime environments to compute them.
- Exploit SQL aggregation functions to (in essence) push processing down to the (usually more efficient) DBMS level.

When challenging reporting requirements demand maximum tuning, developers can set up JasperReports and/or JasperServer in an integrated development environment (IDE), set break points, and step through the fill process to see precisely how it progresses. Jaspersoft provides sample code to assist developers with this process.

The Fill Stage. Here, scalable BI solutions must be able to retrieve data quickly and efficiently—and that means highly optimized queries should be automatically generated by report and query definition facilities. It's also critical that report designers be given tools for optimizing memory and processor use, and for testing and refining results before deployment.

End users should have the ability to request ad hoc queries using a limited number of input rows or output pages, in order to ensure queries are correctly defined before initiating large-scale execution. Ideally, cross-query governors will enable administrators to manage, meter, or throttle demand on a holistic basis. Administrators with the proper authority must also be able to cancel any runaway process.

In large enterprises, it's not unusual for multiple users to be running identical (or similar) reports and queries simultaneously. Scalable BI solutions further optimize data usage (and speed report creation) by caching retrieved data and making it available across queries, where applicable; they also facilitate report / query archiving and sharing.

JasperServer's ad hoc query functionality includes a Preview mode, which enables users to limit output while they refine their queries. Additionally, on-the-fly data caching and compression enables the re-use of retrieved data across multiple users and queries under certain conditions. On-demand or scheduled operational reports do not share retrieved data in this way, but few reporting scenarios would benefit from such a capability.

Another JasperServer fill-time strategy is the ability to push the creation of certain report objects into helper classes. This can reduce the amount of memory required during the fill phase.

Jaspersoft uses its own soft class loader cache to enable multiple user sessions to share certain classes, saving additional memory. Designing reports to take advantage of this capability is a straightforward matter.

The Output stage. Any enterprise-ready BI solution must offer a range output choices: browser-ready HTML, PDF, CSV and Excel make up a minimum set. What's more important, from a scalability perspective, is how these formats are created, stored and distributed; and, to a lesser extent, how easy it will be to add new formats as enterprise (and user community) requirements evolve.

First, output should be kept available, ideally in-memory, for "second looks" and re-casting in other formats for the duration of a user session—and, ideally, for longer than that (subject to space limitations that have their own scalability impacts). The report consumer reviewing an HTML report may decide to distribute it to colleagues or business partners in PDF; there should never be a need to revert to the fill stage strictly to change formats.

Naturally, each format exporter—whatever service converts a report's essential output content into PDF, HTML, etc.—faces unique challenges, and the formats themselves place limitations on each service's efficiency and performance. That said, scalable BI solutions provide configuration and control options to help report designers and server administrators to maximize throughput without undue impact on memory, processor and other resources.

JasperServer retains filled reports in memory for the duration of a user session, which enables users to revert to earlier versions and to create new formats without the need to re-fill. This capability also facilitates dashboard presentation, where multiple report templates are involved.

Because these report objects are cached throughout a session, memory requirements can grow and complicate server clustering—an issue we'll discuss in the next session. Where this kind of high memory load is expected, however, administrators can pin sessions to specific servers and use load balancing techniques (rather than server clustering) to maintain system performance.

Jaspersoft's documentation and configuration options assist administrators in optimizing each type of format exporter—subject to the limitations of the formats themselves.

Optimizing Resources.

Data warehousing and BI industry expert Wayne Eckerson asserts that scalable BI solutions rely on scalable environments that automatically distribute processing across multiple, interconnected physical servers. Administrators should have the ability to distribute various aspects of BI services in a modular fashion; that is, report filling, rendering and output genera-

tion, and distribution processes could ideally be allocated in different ways, across different server mixes, as needed. Load balancing should be automated when appropriate, and all aspects of the load balancing should be clustered as well, so that failover routines can ensure ongoing availability by avoiding single points of failure.

Mere clustering isn't enough: the clustered software must use resources efficiently. Administrators must be able to configure for horizontal scaling (the addition of servers as demand increases) as well as report queuing (to meter scheduled report production during "popular" scheduling windows such as midnight).

We have already seen that resource optimization includes leveraging specific database platform capabilities. It's also critical that scalable BI solutions are easily integrated with—and tuned for—specialized, high-performance analytic database platforms (including column-oriented databases). In analytics scenarios, where performance is both vital and more difficult to ensure, these databases can, in some cases, enable enterprises to avoid the (costly) creation and maintenance of hypercubes and other pre-summarized data stores.

Finally, scalable BI solutions with analytics capabilities—whether in-memory or OLAP-based—should make extensive use of caching routines, and leverage pre-summarized fact tables.

JasperServer protects resources during periods of heavy scheduled reporting through the use of a queuing mechanism; administrators can, essentially, limit the number of report-generating threads running. Overnight reports are the obvious application here.

During busier periods, when more interactive reporting and analysis occurs, JasperServer's clustering capabilities come into play: as more users log in and server capacity fills up, additional servers can be added into the cluster, and traffic loads can be balanced across the cluster for optimum performance.

Jaspersoft's native query optimization for popular databases also helps optimize resources; iReport optimizes a query for platform-specific capabilities when it will increase performance to do so. As an example, an analysis or ad hoc query might take advantage of Oracle's inline table query capability.

Jaspersoft is easily integrated with many high-performance analytic databases, and also builds (and caches) pre-summarized fact tables that can dramatically reduce memory requirements. All of these facilities can enable JasperServer to support hundreds or thousands of us-

ers, depending on usage patterns, on a single hardware server. And as requirements become more complex, volumes increase or “high-demand users” enter the mix, server clustering extends this scalability significantly.

Productivity for BI developers and administrators.

Growing enterprises—or any size enterprise with growing BI requirements—will always want to optimize their most scarce resources, and IT seems always to be among the scarcest. Scalable BI solutions help developers and administrators get more done, faster.

The central contributing attribute in this dimension is the notion of “self-service” BI. The extent to which non-technical—or less technical—resources can define, execute, schedule and share actionable information is a key measure of BI scalability. Just as important is the speed with which technical resources can deliver complex new reports, analysis cubes and dashboards once requirements have been identified.

Scalable BI speeds and simplifies development tasks—integrating BI functionality with other solutions, accessing new data sources, designing reports and data marts—while streamlining administration efforts such as setting up security and access controls, adding and maintaining users and roles, and managing peak loads.

The Jaspersoft Business Intelligence Suite is built around a powerful repository that holds report and query definitions, report schedules, security (roles, authorization and authentication), binary (compiled) reports and queries, report objects, archived output and more. Developers, administrators and end users access and manage repository content easily, each appropriate to their role and permissions. Administrators can easily manage repository access. The repository significantly streamlines a wide range of developer and administrator tasks.

Many repository tasks can be achieved from external applications in addition to direct, browser-based access, which enables organizations to fully integrate development and administrative environments.

Solutions scale more readily when they make good use of other enterprise resources; JasperServer is easily integrated with external directories such as LDAP and CAS, enabling single-sign-on capabilities and simplifying administration effort.

Other productivity-enabling features include:

- In-memory analysis tools, which enable rapid data exploration without the need for complex data warehouse, ETL processes and cube definitions. In this case, end users begin using analytics immediately, while bypassing IT altogether—the ultimate IT productivity strategy.
- Multi-tenant architecture features enable users from multiple companies, departments, business units or other discrete organizations to share a single computing infrastructure while maintaining data and report isolation. This means the same number of administrators can support the equivalent of multiple JasperServer instances.
- Jaspersoft's OCBO Connect enables users to share and analyze a central data source using a familiar interface such as Microsoft Excel, reducing the need for end user support and training.

Jaspersoft's acknowledged ease of use—for all user types—helps make end-user self service a reality for organizations that have deployed it. Additionally, its ad hoc query features reduce the number of reports that IT resources might otherwise be asked to define and produce. Finally, the ease with which technical resources can develop and deploy even complex reports means user communities spend less time waiting. All of these attributes support an organization's ability to grow quickly, and to rapidly grow their use of BI.

Open source Business Intelligence: ready to scale, now.

Jaspersoft Business Intelligence solutions meet the scalability needs of large, growing enterprises, in every sense. The solutions help developers and administrators accomplish their tasks quickly and effectively, while offering the user community state-of-the-industry features and functionality—the kind of capabilities that enhance user adoption and encourage self-service.

Among other attributes, Jaspersoft's design tools, configuration options, report virtualization and full Java clustering support enable the solutions to manage high, fluctuating volumes of users, data queries and reports. Many of these same features help organizations to get more done—or, at least, to get more BI done—while minimizing the impact on the organization's computing infrastructure and resources.

One aspect of scalability that can be overlooked is cost of acquisition and ownership. In this regard, Jaspersoft enjoys the advantages of its open source foundation: low, flexible licensing costs, a large installed base, and knowledgeable user and developer communities to speed problem solving and keep maintenance costs low. Enterprises know they'll have the support they need; and they know that, as user adoption climbs, they won't be held back by software costs.

The combination of high scalability and open source flexibility makes Jaspersoft a highly attractive choice for large enterprises—and for organizations planning on becoming large enterprises.

CONTACT US

Jaspersoft Headquarters:

539 Bryant Street, Suite 100
San Francisco, CA 94107, USA
phone: +1 888.399.2199 or +1 415.348.2380
Email: sales@jaspersoft.com

Jaspersoft Europe, Middle-East, Africa:

4 St. Catherines Lane
Dublin 8
Ireland
Phone: +353 1 443 4700
Email: sales-emea@jaspersoft.com

Asia Pacific, Japan, Australia/NZ:

36th Flr. CRC Tower, All Seasons Place
87/2 Wireless Rd., Lumpini, Phatumwan
Bangkok 10330 Thailand
Phone: +66 2 625-3165
Email: sales-APAC@jaspersoft.com