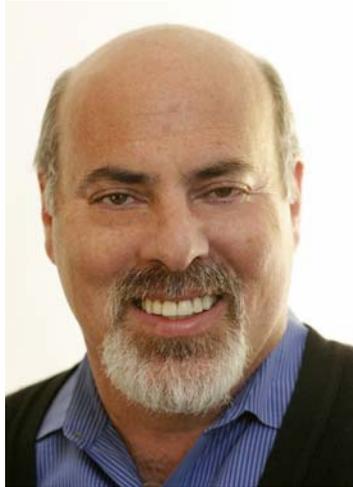


## **Enterprise Reporting Steps Up:**

The convergence of operational and analytical reporting requirements are best met with an integrated architecture, a broad set of functionalities and robust data integration capabilities

## ABOUT THE AUTHOR



Neil Raden is the founder of Hired Brains, Inc., <http://www.hiredbrains.com>. Hired Brains provides consulting, systems integration and implementation services in Business Intelligence, Data Warehousing and Performance Management for clients worldwide. Hired Brains Research provides consulting, market research, product marketing and advisory services to the Business Intelligence, Data Warehousing and Information Integration industry. Based in Santa Barbara, CA, Raden is an active consultant and widely published author and speaker on data warehousing, business intelligence and information technology strategy. He welcomes your comments at [nraden@hiredbrains.com](mailto:nraden@hiredbrains.com).

## TABLE OF CONTENTS

<b>ABOUT THE AUTHOR .....</b>	<b>2</b>
<b>TABLE OF CONTENTS.....</b>	<b>3</b>
<b>EXECUTIVE SUMMARY .....</b>	<b>4</b>
<b>INTRODUCTION.....</b>	<b>6</b>
<i>The Operational/Analytical Gap.....</i>	<i>6</i>
<i>Two Approaches to the Problem .....</i>	<i>7</i>
<b>EII AND THE DATA WAREHOUSE .....</b>	<b>8</b>
<i>Data Warehouse Limitations.....</i>	<i>8</i>
<i>The EII Solution .....</i>	<i>10</i>
<i>EII Limitations .....</i>	<i>11</i>
<b>THE CURRENT LANDSCAPE OF ENTERPRISE REPORTING .....</b>	<b>12</b>
<i>A Great Reporting Server .....</i>	<i>13</i>
<i>What Makes a Great Report? .....</i>	<i>16</i>
<i>A Word About Spreadsheets .....</i>	<i>17</i>
<b>CONCLUSION .....</b>	<b>19</b>

## EXECUTIVE SUMMARY

When it comes to getting information to people and processes, enterprise reporting<sup>1</sup> is hot. Even though data warehouses and Business Intelligence<sup>2</sup> (BI) tools have been promoted to dominate the discussion about information delivery, it is undeniable that enterprise reporting occupies the premiere place in the pantheon of methods for informing people and processes. The reasons are simple:

- Reporting didn't get hot, it always was
- Reporting is more widespread and pervasive than BI
- Enterprise reporting tools leverage standards
- The convergence of operational and analytical reporting requires use of data from multiple sources and distribution to many more people
- Newly emerging technologies, particularly Enterprise Information Integration (EII)<sup>3</sup>, Web Services and Service Oriented Architectures (SOA) play to enterprise reporting's strengths

Data warehousing and BI are powerful mechanisms for generating analytical work, such as trend analysis or customer segmentation, or to support interactive data exploration, but fundamentally, they are circumscribed by their approach, which is to identify the subjects of interest first that results in restricting the domain of inquiry. They are only useful to the extent that the data has been modeled, mapped and loaded, the latter with some degree of latency, a day at least, often longer. No matter how broad or ambitious the data warehousing effort, once completed, it cannot be expanded simply or rapidly. BI tools, especially those that recently introduced or acquired reporting capabilities, have significantly improved their ability to produce the wide range of formats and types but they have not expanded the domain of information they address beyond the data warehouse, nor can they produce the volumes of reports or scale as well as a server-based enterprise reporting platform.

Radical changes in the way companies do business over the past ten years, especially the degree to which they interact electronically with partners, suppliers, customers and regulators, have increased the volume, variety, speed and audience for reports. A data-warehouse-only approach is not workable. Accessing operational information in near real time both inside and outside the firewall is an absolute necessity and new reports based on new information sources are needed without delay. A data warehouse represents a good

---

<sup>1</sup> "Enterprise reporting" is distinguished from "reporting" by its scope, scale and capability. Almost any program can generate simple reports, but only enterprise reporting can deliver any information, anywhere in any format, and scale to thousands or even millions of reports and users per day.

<sup>2</sup> Business Intelligence tools are defined as those that provide ad hoc query, data "surfing" capabilities such as OLAP and other primarily interactive, analytical analysis support, including the embedded or added report-writing and broadcasting facilities, primarily through data warehouses.

<sup>3</sup> EII is an term without a precise definition, but for the purposes of this paper it can be defined as a set of technologies that provides an abstraction layer, or metadata, that projects a single definition of data located in many systems; handles the translation between requestor and provider; optionally provides caching and/or materialization of data; provides query federation, or the ability to query multiple data sources in a single query.

compromise between breadth of subject areas and latency of update for now, but it cannot, on its own, address the broader informational needs of today's fast-paced and connected organizations. Neither can an enterprise reporting tool, on its own, but EII technology enables reporting from a wide variety of information sources, including multiple sources, even original sources, in a single report in near real-time. Past performance bottlenecks in querying operational systems directly have been minimized through a variety of techniques including caching, scheduling and federation.

Over and over again, it's been proven that the most successful organizations excel not by framing the perfect strategy and executing it flawlessly. Rather by placing themselves, using whatever means, in a position where they see and exploit attractive opportunities before the competition<sup>4</sup>. This is only possible when those opportunities are visible, and the viewers know what they are seeing. That requires a great deal of monitoring and much of it is situational and real-time. There isn't time to take six months to model and implement the information or to adjust and adapt, a substantial portion of its reports.

There is also an artificial gap between operational and analytical processes, applications and reporting that is disappearing. This convergence is driven by changes in application architectures, particularly componentization, standards, such as Web Services, SOAP and XML, being embraced by the largest application vendors. As hybrid applications grow in acceptance, there will be an increased need for reporting solutions that are fast and flexible. The most effective solution will have the ability to manage information through an abstraction layer that leverages metadata already in place and can distribute information throughout the entire enterprise and beyond,

This paper will examine how enterprise reporting with EII meets the current and emerging needs of organizations today and in the future.

---

<sup>4</sup> For an in depth description of this phenomenon, see Clayton Christensen, *The Innovator's Solution* (Boston: Harvard Business School Press, 2003)

# INTRODUCTION

The great promise of data warehousing and Business Intelligence is the benefit of knowledge workers helping themselves to data with ad hoc query, OLAP and other self-service analysis tools. Recent market research<sup>5</sup>, however, reveals that these technologies reach fewer than 10% of the potential audience of people with current information needs. Despite the prominence of self-service tools in the industry buzz, enterprise reporting still represents the most widely used and effective means to disseminate information.

The underlying economics of reporting are very different from self-service analysis tools, and the ROI, TCO and best practices of reporting solutions are often overlooked in the press, conferences and analysts' reports. The preponderance of information disseminated within (and beyond) organizations is still delivered almost exclusively through reports, not interactive data requests, surfing, diving or mining tools. In addition, the technology behind reporting is expanding rapidly and is already quite sophisticated.

## The Operational/Analytical Gap

Partly overshadowed in recent years by more attention-grabbing technologies such as data warehouses and OLAP tools, enterprise reporting is enjoying a resurgence in interest from vendors, analysts and the press. So much so, in fact, that all of the leading BI vendors have augmented their product portfolios with new or acquired reporting tools. While one might ask, "Why is reporting suddenly hot again?" the answer is, quite simply, that it always was. The history of the application of computers to business began after WWII as primarily a record-keeping and report-generating initiative – reporting was the application. Starting with the appearance of CICS from IBM and similar on-line transaction processing (OLTP) tools, though, an ever-widening gap between operational systems and the necessary reporting and analysis of information from them appeared. As a result, Business Intelligence applications and OLTP evolved separately.

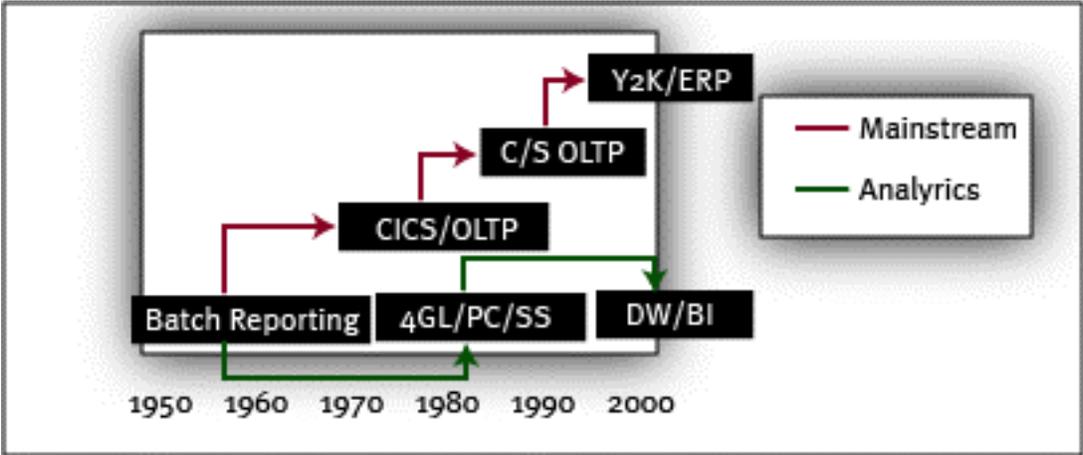


Figure 1: Gap Between Operational and Analytical Systems

Starting with fourth-generation programming languages (4GL) like FOCUS, EXPRESS and APL, followed by Decision Support Systems and PC spreadsheets, business units, with or without support from IT, developed their own reporting and analysis solutions to compensate

<sup>5</sup> *Intelligent Enterprise*, "Dashboarding Ourselves," September, 2003

for the weak reporting offered in the operational systems. None of these solutions, however, was completely effective and all of them were expensive and tedious to build and maintain. In time, data warehousing arose as a structured solution to address these intractable reporting and analysis problems:

- Performance: Systems designed for transaction processing performance, especially in an era of expensive, proprietary hardware, were unable to tolerate the added load of query and report processing
- Data quality: OLTP system data integrity was limited to the needs of the application; integrating data across systems was difficult to impossible
- Access: Initially, connectivity issues were paramount, but also security, licensing and organizational boundary disputes restricted access to information
- Stewardship of historical data: OLTP systems did not maintain historical analysis needed for variance reporting, trend analysis and even statutory/regulatory requirements

### **Two Approaches to the Problem**

Rising to the challenge, data warehousing performed fairly well, providing organizations a solution to the reporting and analysis problem by creating a separate environment where useful information could be managed for all forms of analytical and informational needs. Into this environment were introduced technologies that were not feasible before. It enabled the knowledge workers to explore and analyze the data, build their own models and reports and work interactively. The ideas were so appealing, that they attracted an inordinate and undue amount of attention. Ten or fifteen years later, it turns out that most people are not willing to change the way they work with information, still preferring information in a steady, predictable, published way and to manipulate data, when needed, in personal productivity tools, such as a spreadsheet.

Enterprise reporting (i.e. systems that deliver any information, anywhere in any format, and scale to thousands or even millions of reports and users per day), on the other hand, garnering far less mindshare than BI tools, has continued to expand its role.

Two market forces are at work pulling enterprise reporting to the “top of mind” in the industry. First, useful standards have finally emerged for data interchange, such as XML, and the plumbing to connect networked systems, such as SOAP and Web Services, coupled with the drastically expanded bandwidth of computing and storage capabilities of the past few years. This facilitates the mixing and matching of “best of breed applications,” a trend away from the single, monolithic applications like ERP that have been dominant for the past decade, forcing application vendors to componentize their products. In this environment, where no one vendor has enough influence to control the metamodel of the organization, a separation of business logic from data is necessary to provide hybrid operational/analytical features as services instead of complete applications, which in turn opens the door for other reporting “services.” Currently, reporting is often embedded in each application and supplemented by additional handcrafted reports. In a service-oriented architecture, it is much more likely that useful reports combine and integrate information from multiple services. Even in those cases where reporting is based on a single service, an enterprise-wide reporting “service” makes more sense architecturally, as components are likely to be swapped and/or modified constantly – agility is the goal of a service-oriented architecture.

The second market force is the evolving nature of information integration technologies and practice, particularly the relationship between data warehouses and EII. This topic is explored in depth in the next section.

## EII AND THE DATA WAREHOUSE

### Data Warehouse Limitations

Data warehousing emerged as a solution to the problem of reporting from too many unintegrated and non-conformed systems, by providing for a clean repository of integrated data culled from the various systems. The gathering acceptance of the idea over the past two decades encouraged existing reporting and analysis tool vendors to tune and target their products to this new entity and new companies and ideas entered the field. The field of Business Intelligence was born. But data warehousing and enterprise reporting are not always compatible. In fact, to most data warehouse practitioners, reports are viewed as a temporary solution until everyone learns how to integrate BI into their work, meaning, learning how to find and manipulate data for themselves.

There are two problems with this approach. First, a data warehouse approach, of necessity, entails understanding in advance the totality of data needed to support people's informational requirements. Understanding what subset of information is adequate to satisfy all of the informational needs of a wide and diverse audience is simply unreasonable. The source systems themselves are large and complex and difficult to map. ERP and other "enterprise" systems represent functionality breadth that is so broad, it is impossible to "warehouse" all of it.

Unfortunately, data warehouses and most BI tools do not make provision for accessing data beyond the boundaries of the data warehouse model, and expanding those boundaries is tedious and time-consuming. There is an even greater problem with metadata – to the extent that it is defined at all in a data warehouse, and it very often is not. Metadata is limited to only those elements that exist in the data warehouse. There is no provision for expanding the metadata to include elements that are beyond the data warehouse model, therefore there is no mechanism for accessing data outside the warehouse with the BI tools.

The second problem is partially a result of the first. Without access to needed information, people improvise. Projected cost savings through more efficient use of data is typically used to justify the cost of building and maintaining a data warehouse. The idea is once people learn how to use the tools properly, they will not need to rely on IT to write reports and maintain extracts for them and they will spend less time themselves in the tedious effort of finding, cleansing and reporting information. But with less than the full complement of data they need in data warehouses, people fall back to two alternatives. They either continue to rely on IT for their reports, or they use BI as merely another extraction tool, pulling data into spreadsheets, where it is merged with the complement of data not found in the data warehouse. This process is so error-prone and labor intensive that it has a name: Shadow IT. Shadow IT often drains a data warehouse project of 100% or more of its intended benefits. Booz Allen Hamilton studied the problem and drew the following conclusion:

*“Peer into the hallways of any business unit, and you will likely find ‘shadow staff,’ people performing tasks that duplicate those performed elsewhere in the organization, typically by corporate functions (e.g., HR, Finance, IT). No matter the industry, shadow staffs lurk in the corners of most large enterprises. Once brought to light these positions can add another 30 to 80% to total support staff head counts<sup>6</sup>.”*

There is another alternative. Beyond the small percentage of knowledge workers who use self-service BI tools like ad hoc query and OLAP on a regular basis, there lies a vast pool of people and applications who view formatted reports, in the myriad of forms they are rendered in, developed in an enterprise reporting platform that operates with all kinds of data sources, including data warehouses.

The purpose of a data warehouse is to provide an integrated repository of data for reporting and analysis. The fundamental concept is that the individual source systems that provide data to the data warehouse are not capable of satisfying queries for reports without degradation in their performance, that integrating data from multiple sources is essentially a batch process because it is time-consuming and the source systems do not maintain historical data, among other reasons. There is also the issue of data loss. While it may seem that extracting data from a source system and loading it into a data warehouse means that you “have” the data, in fact, data separated from its application is a facsimile of itself. At the very least, it is separated from the security scheme in place, and in many cases, its context is lost and never reassembled. Take for example, a sales order that is mapped to customer in a geography and assigned to a sales rep in an organizational hierarchy. When the order is copied into the data warehouse, it carries these assignments with it. But consider the occasion when the sales department is reorganized and this customer is assigned a new associate while the original rep takes over a new territory and customer base. The data warehouse is usually modified to support the new structure, which is fine when we anticipate the future prosperity of the new territory. But maintaining the historical context of the order, and its original sales credit is also important, especially when reviewing the original representative’s historical performance. Some ERP systems, PeopleSoft for example, can account for this case, others, including most data warehouses, must pick a perspective and stay with it.

In practice, a data warehouse is modeled first, and then data is found to populate it. The data warehouse, not the original source, now determines the meaning attached to the relocated values. The implication is that, even in those cases where data has been mapped to the data warehouse, there are reporting and analytical requirements that can only be met by working through the application layer semantics of the source systems. In this case, an EII-surround solution is needed to provide both the history and the current context of the data.

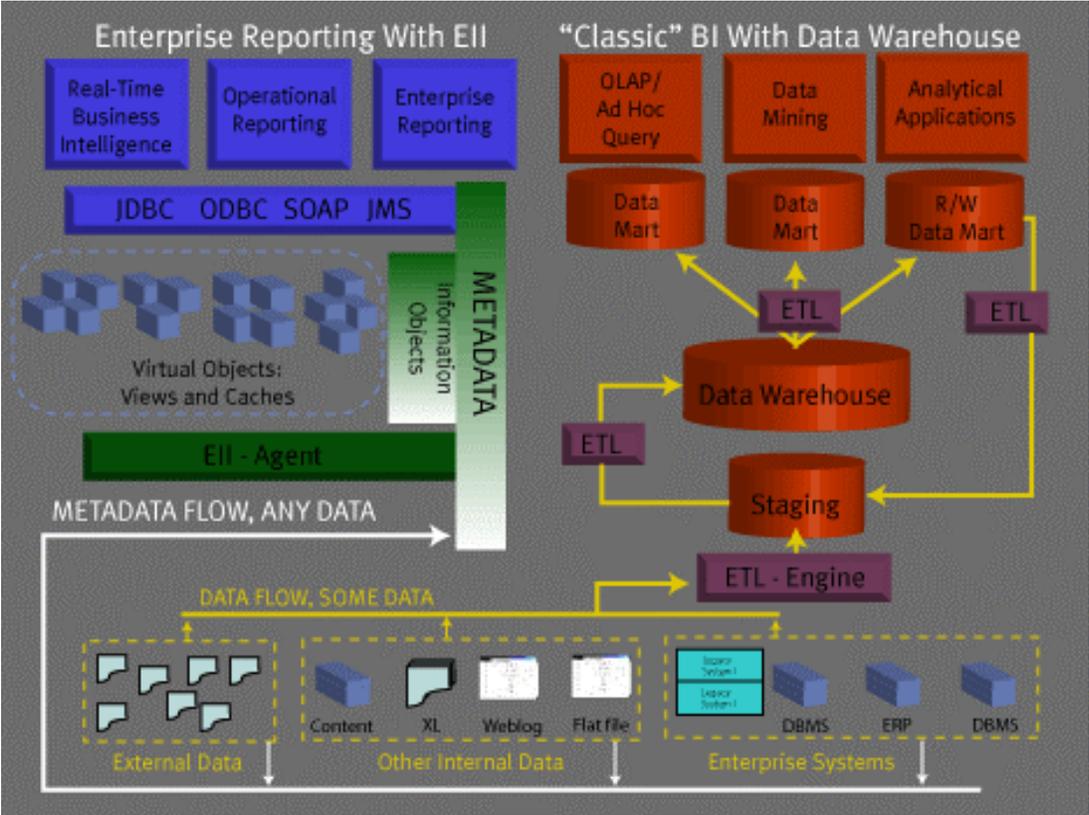
Historically, report-writing software has fallen into one of two camps – those designed to query data directly from source systems and those that presume the existence of a data warehouse. With the emergence of EII tools like Actuate’s Information Objects these distinction break down.

---

<sup>6</sup> strategy+business (a Booz Allen Hamilton Inc. publication), “Shining the Light on Shadow Staff,” 3/2003

### The EII Solution

One way to look at EII is as a layer of translations that insulate the requests for data from the complexity and changes in the physical containers of the data. In contrast, most BI tools, even those with excellent abstraction layers, still must set up pointers to the physical locations of data and process the data, for example Extract, Transform and Load (ETL), on the back end. Though there are some innovative tools emerging that are starting to address this problem, it will be quite some time before they dominate. EII does not eliminate the need for the administrator to understand the complexity of the data sources, but it does create a central set of transformations that can be used by many different applications.



Consider the diagram in Figure 2. The yellow arrows represent the actual extraction, movement, manipulation and reloading of data from source systems to the various data warehouse structures. In many cases, it is a multi-step process from the initial extraction, to a staging area, to a data warehouse and then to the data marts that are actually queried by the BI tools (the data warehouse often is too). To reiterate, even though this is a useful endeavor, there are some drawbacks:

- Latency: Data movement takes time and loading data into multiple dependent structures is subject to failure at the weakest point
- Rigidity of the model: ETL processes start with a known data model, at the physical level. Any changes require development, testing and release, with possible downstream impacts not entirely known.

- Subset of the data: The data warehouse creates the model first, and then populates the data. Access to a source system does not imply access to all its data
- Data Loss: When data is physically pulled from one system to another, its context is often diluted or lost completely. At the very least, security has to be reestablished.
- Training: BI tools require training to use properly. It may be straightforward to operate them, but learning how to effectively utilize them and apply them to work processes takes some time and effort on the part of both IT and the end users, an effort that is often given too little attention. By way of contrast, it requires no training at all to receive a report.

In addition, the data warehouse structures support a type of reporting and analysis that is more “personal” than “enterprise”. OLAP, ad hoc query, data mining and purpose-built analytical applications generally cannot scale to thousands or even millions of reports in a day. The data warehouse generally is not operational for certain portions of the day while it is being refreshed. And finally, the communication between the many components in the data warehouse architecture is largely based on proprietary protocols, though this has begun to change.

In contrast, EII architecture involves mainly the locating and/or creating of metadata, descriptive information that provides a means for the EII engine to query (and in certain cases, update) all of the data sources. One major advantage is that as soon as the metadata for a source is registered, that source is available for reporting. There is no lag for development, testing and release into production (or at least, a very truncated one, as no code is modified, no physical structures are created or dropped). This is represented in the diagram as the white arrows. The EII agent consumes metadata and composes requests for data in the most efficient manner it can. In some cases, it may make sense for caches of data to be created, for performance reasons. For example, the first three days of the month, query activity against the GL transactions may be very high, so the EII engine will create a temporary cache. The difference between this and a data mart is that the cache is expedient, it is not designed and it is not persistent. If anything above the cache in the dependency chain is changed, the cache is invalidated and disappears. There is no maintenance involved.

Also worthy of note is that EII communicates strictly in open standards. In fact, if there is a need to use XML to query a SQL source, or SQL to query a non-SQL source, the EII engine sorts through it and performs the query, unlike EAI approaches, which require each dialogue to operate at the application level (API for example). Also notice that the data warehouse structures themselves can be exposed to the EII engine through metadata.

So to summarize, advantages of an EII approach are:

- No limit to the number or size of data sources (though metadata must still be mapped and/or created)
- Near real time: Updates to source systems are available to the EII queries as they occur
- No data loss
- Recognition and honoring of the security of each application
- Combination of historical perspectives stored in the warehouse with the current performance available from the source system

### **EII Limitations**

Clearly, there are some drawbacks to an EII approach. Though the newest software applications, using the latest technologies, operate with a speed and efficiency unimaginable

even a few years ago, the problem of querying operational systems for reporting without degrading their performance is still an issue. It is an especially important issue when the source systems are older legacy systems. Care must be taken when exposing these applications to an EII engine.

The data warehouse is still the logical location for deep analytical queries, especially those involving historical data but as will be explained below, EII is not an alternative to a data warehouse; it is an addition to it. The contents of the data warehouse can be accessed using traditional BI tools as well as the EII engine. This enables the EII engine to merge the results of a data warehouse query with data not available to the data warehouse, for example. This blend allows the review of real-time status against a historical perspective to drive stronger decisions.

The major drawback of EII, however, is the degree to which it can effectively manage the problems of data integrity and data quality. These are precisely the reasons that data warehouses have relatively high latency – resolving these conflicting data issues are time-consuming.

The real power of EII coupled with an enterprise report server is the ability to provide nearly unlimited amounts of information to virtually any client in any format.

## THE CURRENT LANDSCAPE OF ENTERPRISE REPORTING

Enterprise reporting is a fuzzy term that can include everything from stacks of greenbar from the mainframe to a printed spreadsheet to a digital dashboard to an alert spoken into a voice mailbox by digitally synthesized speech. In general, anything that is prepared specifically for someone else, detailing business variables, typically numerically oriented, is a report. For our purposes, an enterprise-scale report is defined as a display, in virtually any medium, that is generated by an organized system according to some rules of content, format and distribution. Interestingly, these requirements can be applied to a wide variety of systems, both inside and outside the firewall. An



Figure 2: A "Greenbar" Report

online checking account statement is a report, for example, as is a retail store's shelf stock label.

This definition eliminates personal productivity tools, such as spreadsheets and personal databases, as *enterprise reporting* tools not because they are not capable of producing reports, but because they are not essentially systematic in nature, they are largely a manual process that cannot scale. Despite this definition, a great deal of what passes for reporting in organizations is generated by personal productivity tools, a product of Shadow IT. The problem with Shadow IT is that it is so difficult to assess how effective the IT organization is in meeting its clients' needs and how it overstates the functional cost of departments doing what is essentially IT work. In addition, these personal productivity tools, when applied beyond their intended purpose, are inefficient and an impediment to the smooth flow of information.

Though the actual display of rows and columns in an OLAP tool are, technically, a report, OLAP is also excluded from this definition of *enterprise reporting*. The "gestalt" of OLAP is one of exploring and is, essentially, personal. This distinguishes it from enterprise reporting,

*which is prepared for someone else.* To the extent that OLAP tools can generate reports for distribution, they are within the realm of enterprise reporting, but at that point, they are devoid of their extreme interactivity. In the following sections, OLAP will be positioned as an absolute necessity in the information value chain, but for the purposes of understanding enterprise reporting requirements, it is not included.

There are two other types of interactive reporting that fall within the definition of enterprise reporting. Prompted or parameterized reports are identical to fixed reports, with a fixed format and definition of the contents and calculations, but at run-time, report viewers may be prompted to supply parameters that modify or limit the range of the output. This typically involves restricting the range of dates included or other elements. This is simply a device to ease the development burden by creating a “master” report that can deliver a wide range of information without the need to develop each instance separately. It delivers controlled empowerment for the user, who seeks some amount of flexibility to the report. Prompted or parameterized reports, therefore, are wholly included in this definition of enterprise reporting. Some truly advanced reporting tools can actually use parameters at runtime to modify many other elements beyond the data set being presented, such as format and definitions of contents and calculations.

Ad hoc reporting is a little more difficult to characterize. Some use the term incorrectly to refer to prompted reporting. A truly ad hoc report involves creation of a new report, either from scratch or by modifying an existing report, but in either case, a report “author” is involved in the design, even if the design involves just small, cosmetic changes. There is really very little difference between OLAP and ad hoc reports, except that only OLAP is built entirely on an underlying dimensional model. Also, OLAP tools, by definition, include interactive navigation, something an ad hoc reporting tool lacks. Nevertheless, ad hoc reporting used by an individual to explore information in-line is not technically enterprise reporting. Like OLAP, if the developer chooses to save a view and make it available to others, that activity is clearly enterprise reporting.

### **A Great Reporting Server**

To make chicken soup, it takes water, a few other ingredients (which this writer cannot divulge, sorry, family recipe) and, of course, a chicken. Likewise, to make great reports, the first requirement is a great report server.

It may be a cliché that change is a normal part of business, or that the rate of change is increasing, but techniques for developing strategies to cope with this level of change are still being devised. Many existing reporting systems in organizations are not only based on a static view of the business, they are architecturally rigid and represent a major impediment to dealing with change. At certain levels, consistency and repeatability are not only desirable; they are often mandated, such as regulatory and statutory reporting requirements. But reporting at the operational level, reporting to inform decisions and reporting to formulate and monitor strategies must be capable of rapid adaptation. In short, for a reporting architecture to be useful, it must exhibit the features of both fixed, consistent reporting as well as on-the-fly improvisation.

This is, as the wise man said, easier said than done. Before EII was available, even the best reporting servers required some sleight of hand and a great deal of skill to appear flexible. EII solves some of those problems. In addition to flexibility, enterprise-reporting servers must have the following features:

**Multiple sources:** Many reporting applications call for data to be merged from multiple sources, such as variance reports that combine actuals from a GL and budget numbers from a planning tool. A true enterprise-reporting server is able to integrate data on-the-fly into a single logical whole and construct a report as if the data flowed from one source

**Pixel perfect:** Data warehouse reporting tools have not been able to match the quality of enterprise reporting tools' composition and rendering capabilities. Pixel perfect applies to not only the aesthetic aspect of report construction, but to the structural as well. For example, statutory reports or forms such as invoices or payroll reporting have rigid structural requirements that must be met precisely.

**Near real-time:** Any reporting server that depends largely on the relocation of data from a source to another mechanism is subject to delay in the reporting. Clearly, the ideal approach is to fetch live data from source systems as they are operating. There are cases where this is impossible, because of restrictions due to performance, access or security and other difficulties reconciling and merging data. However, the need for fresh data is pervasive, not just an exception, and without a mechanism to identify and communicate with live systems, a reporting server is not truly enterprise caliber.

**Publishing:** Reports should be distributed by subscription, by role, by exception, by practically any means. Setting up and maintaining the publishing of reports has to be a centralized function, managed by an administrator through a central point of contact and the objects managed by the publishing console must be visible to the entire system. Any changes in the system can therefore traverse dependency chains and avoid manual effort in conducting impact analysis. Despite the centralized control of the process, it is important to insure that individual groups still have the ability to manage their own reports.

**Scale:** The server must display near-linear scalability in addition to providing the capacity to generate reports with thousands of pages and distribute them to thousands of destinations. It should exhibit smart features, such as the ability to construct one large report and distribute pieces of it to different destinations rather than rerunning the report for each customer, or automatically inserting interactive tables of contents and search functions.

**Authoring:** Building enterprise reporting solutions requires careful construction and an enterprise reporting tool should provide authors with not only all the rendering and composing tools they need, but also support them through the entire lifecycle with professional development tools such as version control, change management and impact/dependency analysis.

**Security:** An EII engine inherits security constraints from the source systems, but even inheriting security has limitations. If the viewer of a report is not identified to the underlying source system(s) or if the reporting application is designed to address a function that is not supported by the systems, security in place is not adequate. For example, manufacturing may generate costing reports that pull data from the inventory system, or reports may be generated for customers who do not have access to the source systems. In these cases, additional levels of security are needed.

**Role-Based Security:** With so much data available and accessing it so easy, ensuring that it is used properly is an effort that deserves some clear thinking. Clearly, creating and maintaining access rights by person in a large enterprise is an expensive and tedious task.

Simply making the determination of what a given individual should be able to see and manipulate is difficult enough; creating and maintaining a process and structure to enable it is a very hard problem. One alternative that is gaining acceptance is role-based security. In a role-based scenario, people are assigned to a group based on what they do, their role. This reduces the number of control points for defining access rights, which doesn't reduce the complexity of the problem, but it can drastically reduce the scale, depending on how fine or coarsely these roles are defined, but the need to keep the grain of the roles thick (define a small number of roles) can have unwanted side-effects, such as over- or under-restricting access. People often complain that it is a good solution overall, but a poor one on the individual level.

One interesting effect of role-based access is that it seems to perform best at a very high level, but not very well when roles get more specific. The reason for this is that at a high level, access restrictions are more relaxed, but this may not fully meet requirements. For example, separating access by legal entity or by major division can provide everyone the same access rights within that one "slice" of the organization, such as the tax department and field customer service. Organizations also tend to over do access restrictions, such as restricting sales staff to only their own territories, when valuable insight may be gleaned by comparing or analyzing activities and outcomes in others' territories, as an example.

Role-based security makes sense where roles are clearly defined and the edges between them are crisp and well understood, such as between a pilot and a navigator of an airline or a teller and a loan officer in a bank. But when it comes to operational and, especially, analytical reporting, roles are rarely so clear and even when they are, needs change rapidly, posing a major maintenance and service problem. Actuate's Page Level Security feature can actually take this to the individual level, i.e., we specify individuals who can access and view certain pages within very large reports. This is a key feature in using Actuate for client reporting applications. The feature, which can use roles stored in LDAP directories or in our own server, has been used in rapid change environments that you describe here.

Another issue is properly notifying report requesters and readers of the filters that are in place. In 1978, the executives of General Motors were informed that they held a 58% market share of the US passenger car market, leading them to a disastrous strategy that overlooked the growing impact of Japanese automakers. Unfortunately, the filter on the report was not reported and the market share numbers reflected only *domestically produced* and sold autos, not imported ones, vastly overstating GM's market share.

Restricting access to information for non-employees, such as customers, suppliers and even stakeholders, regulators, competitors and contractors, requires a much more careful scheme. While it is possible and even desirable to train employees to use confidential data with discretion, these other classes of information consumers cannot be expected to be as responsible. In those cases, organizations are wise to restrict access to "sandboxes," separate resources designed specifically for these constituencies.

The solution is twofold. First, any reporting tool must have a robust enough security model to accommodate, efficiently, almost any security access scheme the organization desires. This should be combined with a more positive way of solving the problem by educating and encouraging employees to use information wisely and to employ human resource techniques and programs that identify and weed out problem and potential problem individuals.

## What Makes a Great Report?

Reports are a little like fish dishes, they may be simple or elaborate, but they are always best when they are fresh. Beyond that, there are some characteristics that make reports effective:

**Attractive:** Regardless of the medium for presentation, a report has to convey information in a way that facilitates understanding. Layout, color, graphics, sensible sub-totals and banding, all of these are necessary. Unnecessary clutter, too much shading or too little – there is an aesthetic that is difficult to prescribe but easy to identify as met or unmet.

**Informative:** In the same way that charts can waste a lot of ink conveying what could be shown in a table with four values, reports can fail to inform by either obscuring the pertinent information in a blur of detail or wasting paper (or its electronic variant) with too much report and too little information.

**Reliable:** Particularly with alerts that inform only when certain events occur, the notified parties will come to rely on the alerts and modify their normal routines. A missed alert can cause lack of confidence in the system or have even more serious consequences.

**No training:** Unlike BI tools, where the user population is more heavily weighted with dedicated users who learn to master its high level of interaction and navigation, reports are designed to inform a more casual user population. Any degree of training needed to operate them detracts from their essential nature. Likewise, training is not a major selling point.

**Timely:** There are two aspects to timeliness, arrival and freshness. A report is composed of its content as well as its delivery and a late-arriving report has diminished value. The content itself has to be timely as well. A detail of O-negative blood on hand as of 2 o'clock yesterday isn't much use at 2 o'clock today, or the price of an option that closes at 4PM received at 4:10PM.

**Relevant:** From the beginning, technology implementations have been plagued by problems with performance and ease-of-use. Based on current research conducted by Hired Brains, however, these issues have receded in importance. Whether it is a factor of better software products and more powerful resources, or the application of better techniques as practitioners became more sensitive to these issues, is not known for certain. What is known is that the proliferation of reporting in recent years increased the population of users and the changed demographics of this community caused a shift in the balance of concerns to two new areas: relevance and understandability.

Performance problems, such as long execution times, are easy to measure and once the problem is diagnosed, the improvement needed is easy to evaluate. The solutions aren't always simple and the measures needed to correct the problem aren't always palatable (especially from a budget perspective), but they are at least quantifiable. Ease-of-use, on the other hand, is an infinitely hard problem, not only to diagnose, but also even to articulate. In practice, though, ease-of-use complaints are usually addressed as part of the performance problem – people hate to wait and the leading cause of ease-of-use dissatisfaction is long wait times, latency of refreshes and downtime. Ultimately, though, ease-of-use is more of an aggregated problem than it is a problem that stands on its own. The parts of it that actually relate to ergonomics, user interfaces and other man-machine design components are relatively small and, given their intractability, not the most productive place to apply resources at the outset. Using an 80-20 approach, performance is a better place to start.

Without being philosophical, what does relevant really mean? In the context of reporting, relevance is “dimensioned” by a few obvious attributes: time, content, presentation, context, and understandability.

Time – when it gets there, the timeliness of the content, how long it takes

Content – the explicit subject matter, the implied models, common and/or resonant vocabulary, and complement of information needed

Presentation – fluid between media, print, spreadsheet, etc. Some tools present well on the screen but render on paper very poorly, or vice-versa

Context – does it solve end-to-end problems, fit into a workflow or process

Understandability – can the information presented be investigated? Is the process for creating the report reputable and do the recipients understand it?

Familiarity – people usually have an existing way of using information or thinking about business processes before a reporting solution is put in place. If the reporting solution supports the way they currently think or behave (as opposed to enforcing a new, “better” way to process information) then they will perceive the solution as easy to use.

It takes a very sophisticated piece of software to be able to provide all of these features and functions.

### **A Word About Spreadsheets**

It is undeniable that spreadsheets play a key role in the analysis and distribution of information in organizations. Despite twenty years of effort trying to displace spreadsheets with more centrally controlled applications, IT organizations are still searching for balance in their use and larger issues of information management. Spreadsheets and other personal productivity software, like Access databases, play a large part in a very big problem – Shadow IT. But if used properly, spreadsheets can be and are an indispensable tool.

Why are people, business people in particular, so enamored of spreadsheets? Based on our research, we’ve found two overarching characteristics of spreadsheets, beyond their actual functionality, that can explain their proliferation:

Spreadsheets are expressive: Spreadsheets are often the only tool with which people can actually create something on a computer. The number of steps between the conceiving of a report or model and completing it are absolutely minimal. Spreadsheets work. On a personal level. When evaluated at a departmental level or an enterprise level, they may be counterproductive, but for a person with a need to massage some data, they are peerless. The very aspect of spreadsheet that makes them uncontrollable is the reason why they are so expressive – they are built without foundation. It doesn’t require a library of pre-written applets and the application of industry and organizational best practices to get the job done, it just takes some typing.

Spreadsheets are subversive: Every organization of any size has official, prescribed methods and channels for doing things. Every functioning organization has unofficial back channels where the real work gets done. When someone has a good idea, or needs to communicate with or collaborate with others that are not part of the official flow, the back channels are crucial. It may take months of proposals and the appointment of a steering committee to approve changes in some internal report formats, but a spreadsheet and a back channel can do the same thing in minutes. Subversion can be a good thing and spreadsheets are the perfect vehicle for it.

If an organization wants to attack the problem of Shadow IT constructively, making sure tools like spreadsheets are used effectively is a key to success. There are many tasks to which spreadsheets are routinely applied, or, more accurately, misapplied. To get at the root of the problem, follow the guidelines below:

- A spreadsheet is not a database: Database management systems have a myriad of features not found in spreadsheets that guarantee maintainability and sustainability in addition to data management. Extracting large amounts of data to a spreadsheet sidesteps all of these controls, aids and features.
- A spreadsheet is not data management program: The "system of record" for reference data should be a centralized, sharable, scalable, controllable repository. When one individual is responsible for maintaining a set of values, such as product hierarchies, the spreadsheet may be the point of entry, but not the "system of record"
- Spreadsheets are not multi-user: Personal productivity tools lack the necessary features to be reliable as multi-user applications. This is true of personal databases, too, such as Access
- Spreadsheets are excellent personal tools: Because a clever user of a spreadsheet can do almost anything, then almost anything is doable. But it shouldn't leave the desktop as an ongoing application.
- Spreadsheets are not report servers: Tools like Actuate 8 are designed to be enterprise report servers. In the case of Actuate's e.Spreadsheet option full function spreadsheets can be manufactured and managed within a reporting server
- Use centrally manufactured and managed spreadsheets when its unique capabilities are needed
- Macros are programs: Programs are written by programmers and are subject to a rigorous set of standards. When programs are written in an uncoordinated way by non-professionals, maintenance costs soar, complicating the Shadow IT problem. Software engineering should be the province of software engineers, working with the best possible training, standards and tools. Spreadsheet macros beyond simple manipulations of a sheet are to be avoided because they are not quantifiable, controllable or maintainable.
- Spreadsheets are not usually compliant: Sarbanes Oxley and other financial reporting regulations require the identification and eliminate the potentials for fraud. Spreadsheets need locking, change tracking, and usage monitoring in order to be compliant. This means that centralized access to a corporate spreadsheet becomes mandatory.

Despite all of the negatives about spreadsheets, they are the most familiar interface to people doing analytical work. Providing a "Managed Spreadsheet" service to spreadsheet users by allowing clean, fast and simple manufacturing and delivery of spreadsheets from a reporting server is well advised. When implementing this kind of solution, though, it is crucial that people are trained in the proper use of spreadsheets, applying their strengths in a useful way and at the same time, being part of a larger program to shrink the size of the Shadow IT problem by using the enterprise reporting solution and its best features.

## CONCLUSION

Larger trends in IT, such as componentization, service-oriented architecture, the convergence of operational and analytical processing and the relentless campaign of Moore's Law, all indicate an increasingly important role for enterprise reporting. Agile organizations need to exchange information as fluidly as possible. Where in the fairly recent past, a company's ability to document and refine a key business process was a competitive advantage, going forward, that advantage may accrue to those companies that can invent and implement new processes in near-real time. The implication is that, feet-up-on-the-desk data surfing and analysis will be a luxury only a few can afford. Discovering trends and relationships will be an activity for a small number of people while the rest of the "knowledge workers" will be informed using models and rules of thumb<sup>7</sup> embedded in reports, alerts, dashboards and other manner of instantaneous information distribution. The data within those reports may come from ERP systems, newly emerging composite applications (ERP II), legacy systems, your customers, suppliers and other third parties.

Partial solutions will not suffice. An Enterprise Reporting solution must seamlessly weld functions for massive scope and scale, complexity, abstraction of data sources and easy operation with other tools like spreadsheets. Actuate 8, with its enhanced e.Spreadsheet capability and new integrated EII capabilities, is clearly poised to compete for and win these very exciting opportunities.

---

<sup>7</sup> Erlang C is an example, used in call centers to estimate volume of incoming calls in place of actually pulling the data from 30 or 40 variables and calculating the volume.