How to Talk to Your CIO to Get the Data You Need

by
Joe Coté, Vice President of Marketing
Xbridge Systems, Inc.

Financial executives need data, lots of it. They need to have that information in a spreadsheet or other format, so that they can analyze it to make strategic decisions.

However, as most CFOs have discovered, getting data, on demand, in a familiar format is not easy. In most cases, the IT department controls access to the information. CFOs wait for batch reports to be run. If they need more data, or need the data in another format, they have to wait for a (probably overburdened) mainframe staff to prepare that information. All this can be inconvenient and time consuming, and potentially result in lost opportunities.

This article will explore ways that financial executives can communicate better with the IT department, so that CFOs can get the data they need, when they need it, and manipulate that information to meet their needs. The goal is to help financial executives quickly and easily get the important data they need to help make the corporation top notch.

Different Worlds

One basic thing to understand is that CFOs and CIOs live in different universes. CIOs do not concern themselves with quantitative analysis, pivot tables, ANOVA, regression analysis and correlation. If they do "what-if" calculations at all, it is infrequently. The financial department needs data no one else cares about, and probably no one else appreciates how important the data is. In the post-Enron world, the financial department is undergoing closer scrutiny; the IT department is affected little, if any. And the IT department is not required to have instant answers to questions from the CEO and Board that might range from the impact of changing the accounting method and restructuring the sales force to determining the ROI on some new purchase. In short, financial executives deal with content.

The CIO, on the other hand, deals with processes, with keeping the infrastructure going. Their world consists of protocols, operating systems, applications software and infrastructure requirements. They need to keep systems working 24 hours a day, every day of the year. Understandably, they prefer to have as many things scheduled as possible, e.g., they like doing regular reports on a regular basis. Ad hoc requests are less favorably received because they are one more disruption in their day.

Both departments are probably overworked, with incredible demands on their time. But the two departments' duties are so different, there is often a gulf of understanding – even suspicion – between them.

It is pretty safe to assume that the CIO might not understand just how important getting timely data, in an appropriate format, is to the financial operation. The IT staff might not realize how dynamic the financial environment is, and how important ad hoc requests can be.

In their minds, regularly scheduled batch reports should meet the needs of the financial department, which should be responsible for integrating data from various sources. In many, if not most cases, they are probably right as far as routine matters are concerned. Unfortunately, in today's business environment, financial executives must deal with the unexpected, as well as with routine matters. Such situations require non-routine responses.

A View into the IT World

To help bridge the division between the departments, it's essential to "speak a little geek," to be conversant enough in technology to command respect from the IT department, especially when the department puts in a funding request for some new technological initiative or when the financial department needs information quickly.

Here a little history is in order. The IT shop probably started life on the mainframe, using esoteric formats and languages like Virtual Storage Access Method (VSAM), Queued Sequential Access Method (QSAM) and Common Business Oriented Language (COBOL). Most likely the majority of the data, especially the older financial data, is still in COBOL/VSAM applications.

The IT department probably remains wedded to the mainframe, for good reason. Estimates are that up to 70 percent of the world's business data is still on mainframes. The mainframe's reliability, security and performance are unparalleled in the technological universe. And, in tight economic times, it's essential that the company leverage its investment in this proven technology.

However, the IT infrastructure now includes servers (of all shapes, sizes and platforms) and data (in a dozen different formats).

The problem is: mainframes don't "speak" Windows or any of the Internet technologies such as HyperText Markup Language (HTML) or eXtensible Markup Language (XML). And those technologies don't "speak" mainframe.

So the CIO is faced with a complicated environment. Getting data to the desktop, quickly, in a desktop application, is not easy.

Yet that data is exactly what the CFO needs. Consider the case of a large national banking firm that had undergone a merger a few years previously. The CFO was asked to finance the merger of two IT departments. Further, he was asked to find a way to provide single statement sheets for all accounts that individual customers had with the bank. Single reporting was obviously desirable, for it offered the potential of significant cost savings and improved customer service.

To help him accomplish these goals while maintaining his already crushing work load, the CFO wanted to be able to access information on demand, without having to wait for a batch process. And, because he did some traveling, he wanted to be able to access the data not only from his desktop, but also from over the Internet.

All this is easier said than done. But it is possible.

In trying to meet the CFO's needs, the CIO would probably recommend at least one of five major approaches: an Enterprise Resource Planning (ERP) system, a data warehouse/data mart, an all-relational database, some ad hoc approaches, or a direct, real-time method. The alternatives could be used singly or in various combinations to achieve the desired result. Unfortunately, most of the alternatives are expensive and time consuming, and suffer from technological and implementation constraints.

Alternative Methods of Getting Mainframe Data to the Desktop

ERP
An ERP system tries to integrate an organization's functions and departments – including manufacturing, order entry, accounting and human resources – into one computer system. ERP is very complex as typically each area considers different data to be important, and treats that data differently than the rest of the company.

Organizations can choose to develop their own ERP system or purchase one from companies such as SAP, PeopleSoft and Siebel Systems. Whether purchased or built, an ERP system is expensive in terms of money, effort and time. Implementation requires changes to the entire system currently in place. Hardware, operating systems, databases and applications may all have to be rewritten, retuned or expanded. If the bank decides to go this route, it might need to consider other shorter-term approaches as well.

Data Warehouse/Data Mart
A data warehouse is a central repository for all or large portions of an enterprise's significant data. The goal is decision support; the data is made accessible for analysis, querying and reporting. In practice, data warehouses are usually housed in a mainframe or other large server. A subset of a data warehouse, a data mart is targeted at a single function or department.

If considering a data warehouse, the bank would have to factor in the maintenance of the system and the data, and managing the capacity and performance of the data warehouse. Keeping the data current and consistent is a never-ending task. (Data marts frequently grow larger than the company's original concept of the entire data warehouse.) When considering a data warehouse or data mart, an organization needs to have the long view. It typically is far more expensive (and complex) to maintain a data warehouse than it is to build one.

All-Relational Data
A relational database creates relationships between files by comparing information, such as name, accounts or addresses. IBM's DB2 is a good example of a relational database for a multi-user environment.

While an all-relational database would be one step to providing the integrated statements the bank desired, moving to all-relational data has several drawbacks, including:

- Performance. (DB2, for instance, is not as fast as native VSAM.)
- Time. (Most applications have to be rewritten to run under the relational database.)
- Expense. (Relational database licenses are priced based on CPU size.)
- Hardware. (Servers and other computers may have to be upgraded or replaced.)
- Security. (Security is an extensive headache.)

Ad Hoc Processes

FTP, NFS, screen scraping or "sneakernet" are all ways IT organizations have adapted to get data from the mainframe for use on the desktop. Ad hoc processes are typically stopgap "solutions," designed to solve an immediate problem as quickly and easily as possible. Companies use ad hoc methods because they cannot, or will not, commit to a more elegant and effective solution.

Although ad hoc approaches might appear cost-effective at first glance, upon closer scrutiny the true costs become clear. In virtually every case, the user has to ask the mainframe staff for the data. The request is typically acted upon in batch mode, and the data is forwarded the next day. The data is usually outdated, is not really the data that is needed, or is not in the desired format. By themselves, ad hoc methods could not meet the financial department's need for data-on-demand or for integrated statements.

Direct, Real-Time Access

A direct, real-time access method probably holds the most promise for getting real-time data to the desktop. The prime example of this approach, Xbridge Host Data Connect™ from Xbridge Systems, enables Windows® applications and browsers to access mainframe data in real time, and to use that data in a familiar off-the-shelf application (e.g. Excel, Word), meeting the CFO's need for current data.

The direct approach can also simultaneously retrieve data from multiple files and from multiple mainframes and join the data in Windows or browser format, making the single account statement possible. This approach is unique in that it operates without the traditional process of brokering, duplicating or migrating the data. This method provides just the specific data records or fields required. The requesting application can access the data – on demand – through intranets, extranets or the Internet without requiring additional IT resources, making Web access possible.

Summary

Whether to produce a single report for multiple accounts, or any of the other hundred reports and documents needed in the financial world, financial executives need to understand the CIO's world a little.

There are alternative methods for accessing data. Many are long, arduous and expensive. Even the emerging technologies, such as the much-hyped XML, require significant development efforts as well as an in-depth understanding of the company's business processes.

Choosing the right methodology is an exercise in understanding the currency and accuracy required for the given tasks. The CFO needs to understand the ramifications of redeveloping applications using a more modern infrastructure, moving applications to an all-relational database, building a data warehouse or data mart, relying on ad hoc approaches, or utilizing a real-time, direct methodology.

By asking the appropriate questions, CFOs can help assure that any IT changes meet the needs of the financial department for current data to the desktop, when needed, in a format that the user can manipulate. Not coincidentally, by being aware of the alternatives, the financial executive will also be in a much better position to assure that any IT changes are good investments in the long-term viability of the organization.

###

Data-Access Questions (suggested sidebar)

General

- What's the best way for me to get the information I need? When I need it?
- Can I get real-time data?
- Can I get information from various sources integrated into a single report?
- Can I filter the request so I get just what I need?
- Can I get the data myself, without waiting for a batch process?
- Can I get information in a spreadsheet format?
- Will I be able to access data using my browser?

Cost/Implementation

- How much will the alternatives cost?
- Are there any hidden costs?
- How much data is likely to remain unchanged, regardless of alternative chosen?
- What do the approaches take in terms of ongoing support and maintenance?
- Do the approaches truly integrate the data from different sources and present it in a single format?
- How quickly can we implement the various approaches?

ERP

- Does this really model our business well?
- Should we buy or make an ERP solution?
- Can we afford the high cost of ERP?

Data Warehouse

- Do we have the staff and skills to keep a data warehouse/data mart properly maintained?
- Will that data be on time and accurate?
- Will it be the needed data?

All-Relational Database

- Do we really want to re-engineer our processes and rewrite our applications to utilize a relational database?
- How many applications will have to remain in VSAM anyway?

Ad Hoc Approaches

- Do these processes provide timely, accurate, flexible data?
- Do these "solutions" really meet our needs?

Direct, Real-time Access

- Does this method provide data on demand?
- Does it present the information in a Windows or browser format?
- Does this method provide a way to integrate data from multiple sources into a single report?
- Does this approach require massive re-programming or operational changes?
- How quickly can this method be implemented?