Preface

Lin Chao
Editor
Intel Technology Journal

Intel's mission for 2000 is to "do a great job for our customers, employees, and stockholders by being the preeminent building block supplier to the worldwide Internet economy." Our corporate objective has undergone a significant change with this mission. While Intel will continue to grow our core processor and computing business, we also will focus on growing new Internet businesses and market segments. The buzz at Intel and throughout the industry seems to be the changes brought about by e-Business on traditional "brick and mortar" companies.

An example of an e-Business is Intel's Web-based order management system, introduced in July 1998. In January 1998, Intel had no online customers. Orders were filled through phone, fax, and overnight parcel carriers. By June 1999, over 560 companies in 46 countries were using Intel's Web-based order management system to place orders, track deliveries, post inquiries, and get product and pricing updates. Today, this system produces nearly $1 billion in sales per month. By 2001, Intel projects that more than 90% of its orders will be handled over the Web. The shift to e-Business is a key driver of current business strategy and opens up many new opportunities for Intel.

In this Q1'00 edition of the Intel Technology Journal, we will look at how Internet information technologies (IT) are enabling e-Businesses at Intel and beyond. The first paper describes Intel's own Internet connectivity architecture including the business factors that led to its creation and the technologies developed to build and maintain it. The second paper talks about factory indicators using near real-time supply chain performance indicators from a variety of manufacturing sources. These indicators provide the necessary information to run day-to-day operations and avoid or detect problems. The third paper describes corporate Intranets and how to manage information overload.

The fourth paper looks at business-to-business (B2B) specifications for an industry-wide standards-based approach to e-Business known as RosettaNet*. The fifth paper looks at policy-based management of network services. It explores the dimensions of policy-based management, provides a pragmatic review of the technology, and discusses the deployment challenges and usage scenarios.

Applying Information Technology to Enable e-Business at Intel

By Doug Busch,
Vice President and Director, IT
Intel Corp.

Intel is rapidly becoming dependent on information technology for every aspect of its operations. Our core business processes, like those in most large companies, have been enabled by information technology for many years, but a critical change is in progress. Many of our most critical business strategies are now based on taking advantage of Internet technologies to allow integration with our customers and suppliers, to enable electronic collaboration between our employees, and to dramatically increase the speed and efficiency of our business processes. Throughout Intel, groups are developing and executing strategies for improving business using these tools. Major initiatives are focusing on comprehensive business-to-business and business-to-consumer marketing and sales; cost improvements through supply chain integration and more responsive production planning; improved virtual factory operations; and simplified delivery of services to employees.

Worldwide, this dramatic transition is happening throughout the business world. The shift to electronic business, or e-Business, is the key driver of current business strategy. e-Business is the principal opportunity, and in the hands of competitors, the principal threat facing every business. Competitive forces are driving incredibly rapid adoption of e-Business: companies are expanding their computing infrastructure to participate in e-Business, and consumers are buying and upgrading their PC's and network services to take advantage of the growing range of online businesses.

As a preeminent supplier of building blocks to the Internet economy, Intel's product and service strategies have been profoundly impacted by this transition. We are expanding our focus from clients and servers to encompass networking components and Internet services. Critical to Intel's success is the understanding, as a company, of not just how to design and manufacture these Internet building blocks, but of how to use them. Insight into the challenges of building an e-Business will enable us to deliver the best possible products and services to our customers.

Intel is recognized as a leader in the implementation of e-Business systems, conducting more than $1 billion of business-to-business commerce every month, and we intend to maintain that leadership through the aggressive yet pragmatic implementation of new e-Business technologies. The experience gained from building the e-Business infrastructure is one of Intel's most important assets. We deliver Internet connectivity to 65,000 employees worldwide, every day. We deliver information to our sales channel and customers in every location over the Internet. We are increasingly interacting with our suppliers over the Internet. In the process, we're learning first hand what it takes to build reliable, manageable systems; how to ensure they can change rapidly and scale up to meet bursts in demand; and how to integrate

hundreds of companies into a virtual enterprise.

In this issue of the *Intel Technology Journal* you will find some of the lessons we have learned in the process of making Intel an e-Business. I hope these lessons are of value to you as you develop the products, services, and business strategies that will take you into the next millennium.

Cockpit: Decision Support Tool for Factory Operations and Supply Chain Management

Paul Calame, Assembly Test Manufacturing Information Systems, Intel Corporation
Ravi Nannapaneni, Assembly Test Manufacturing Information Systems, Intel Corporation
Scott Peterson, Assembly Test Manufacturing Information Systems, Intel Corporation
Jay Turpin, Assembly Test Manufacturing Information Systems, Intel Corporation
James Yu, Assembly Test Manufacturing Information Systems, Intel Corporation

Index words: supply chain, decision support systems, performance indicators, OLAP

ABSTRACT

We live in the Internet age, where businesses like Intel sell, buy, and manage internal operations in a fundamentally different way than just a few short years ago. Speed and agility have been added to the business performance equation: those that are fast and agile prosper, those that are not, don't. Our customers demand the ability to place and amend orders at will, require minimum inventory stock and just-in-time delivery, and desire order status visibility into the Intel supply line. These expectations drive internal demands for near real-time supply chain performance indicators from a variety of sources, a potential problem alert capability, and a look-ahead and environmental scan capability to support forward business planning.

This paper addresses one Intel program that focuses on satisfying these information needs with the specific objective of improving manufacturing operations and supply line performance.

The new implementation is called *Cockpit* reflecting its ability to provide the pilot with the necessary information to run day-to-day operations and avoid or detect problems. Based on industry standard technologies (e.g., OLAP[‡]), Cockpit is a new breed of Decision Support Systems (DSS) and one that has earned the confidence of Intel manufacturing professionals.

The Cockpit team shares the development process that led to the production system: the initial challenges, the development approach, the technology choices, the enterprise-wide data partnerships, and the various architectural details that make up the system.

INTRODUCTION

It's a *brave new world* in Intel manufacturing. Shrinking product and technology lifecycles, pricing pressures, supply line and segment management requirements, increased manufacturing complexity, and the impact of emerging e-Business are just some of the new challenges for Intel's back-end components manufacturing groups. Coupled with a geographically dispersed, 17 time zone non-stop *Virtual Factory*[◊] operation, the need for breakthrough management strategies becomes imperative. The following questions are posed:

- How does manufacturing become more responsive to customer delivery, flexibility, and cost needs?
- How do we transfer and ramp products faster with low variability?
- How do we reduce cycle time to World Class, and static inventories to zero?
- How do we proactively and dynamically allocate production capacity?

In the Technology Manufacturing Group's (TMG) Assembly Test Manufacturing (ATM) organization, these challenges are being addressed through a variety of initiatives such as supply chain management; better forward planning; asset fungibility and utilization improvements; and people, team, and system resources.

[‡] OnLine Analytical Processing (OLAP).

[◊] A *Virtual Factory (VF)* is a group of factories making the same product and managed as a unit.

To enable many of these initiatives, Information Technology needs to be exploited. Assembly Test Manufacturing Information Systems (ATMis) is one of the programs chartered to deliver the necessary enablers. The ATMis mission is to facilitate manufacturing performance breakthroughs by developing and delivering three key capabilities:

- Business operations and planning data to run factory operations, solve problems, and make decisions faster.
- Real-time and asynchronous collaboration to improve *virtual* work team efficiencies, and to mitigate work-team burnout.
- A virtual information and knowledge environment to enable intra- and inter-organization sharing and reuse.

In this paper, we discuss one of these tools developed by ATMis, the ATM Cockpit. The goal of the Cockpit team is to promote more effective use of ATM resources and assets as follows:

- Deliver management-level data and information needed to run VF manufacturing and supply line operations.
- Reduce time and overhead costs to obtain this information.
- Provide tools and services in half the typical development time at minimum new cost and Cost Of Ownership (COO).

The Cockpit was designed to support three ATM customer segments: the ATM/TMG executive team, the ATM operations management team, and the VF Technology management team. Its specific deliverables are as follows:

- Put factory and supply-line historical status and look-ahead information from a variety of disparate domains at the consumer's fingertips.
- Provide a potential problem alert and notification utility.

These deliverables are provided by way of features such as data drill-down and graphical display, a personalization utility, and an easy to use Web-based front-end.

A technical core team was assembled to drive the program and work with IT and company data providers to create a standards-based architecture and technical framework. The team employed a small group of key customer stakeholders to guide implementation, utilized rapid application development and deployment (RADD) techniques to fast-track development, and employed software already licensed by Intel to create and deploy the product in time. Using an *add value, ease the pain now* deliverables approach, the team produced and deployed the pilot product in six months; subsequent updates were done in 90-day cycles. Since data is crucial to customer decision making, significant effort was paid to modeling and validation of source data. The tool is currently employed in ATM OLGA[∝]/Flip Chip Virtual Factories and in ATM operations management.

APPLICATION OVERVIEW

The ATM Cockpit is a Web-based tool that allows users to create reports on-the-fly. The Cockpit allows end-users to design reports known as views by selecting easy-to-use navigation controls. Users can define and save personal views or drill-down from high-level to detailed views. Saved views can be accessed on the user's Cockpit front page by clicking a thumbnail report icon. Alternatively, users can subscribe to these views which are then sent to their desktop. The Cockpit also allows users to manage business through an exception management system, which monitors predefined and personalized indicator trigger points. The tool notifies users when an exception triggers a fixed value or deviates from the goal. Additionally, the Cockpit provides a complimentary real-time event delivery mechanism that incorporates e-mail and channel-based notification. Configurable to individual needs, the delivery mechanism scans the business environment pushing news and other events of interest to the user for convenience and easy access.

MANUFACTURING CASE STUDY

The following case study[⊕] demonstrates how the integrated Cockpit application directly impacts management's day-to-day decision-making process. This case study focuses on a specific instance of how a VF product manager uses the Cockpit to identify and confirm a potential yield issue and proactively take the necessary steps to correct the problem.

[∝] Organic Line Grid Array.

[⊕] These examples are for informational use only and do not represent actual data from Intel.

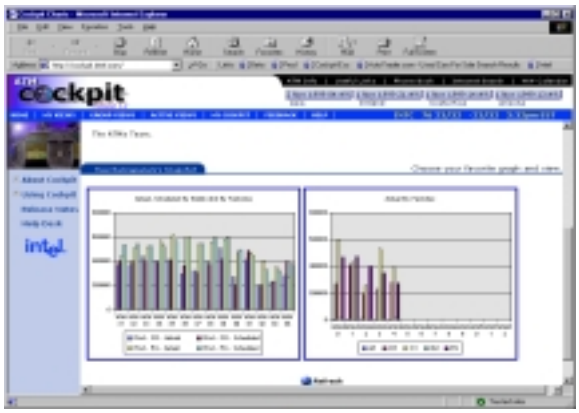


Figure 1: Cockpit user's customized front page

Figure 1 shows a typical Cockpit product manager's customized front page with thumbnail icons displayed. These icons represent favorite views the manager would want to see on a daily basis, providing insight into business operations. Each manager specifies which thumbnails to display by selecting from a standard list of defined views, or a list of personalized views previously created and saved. With one mouse click, the manager can select any of the thumbnails to examine the full *active view* displaying data from the most recent OLAP cube update.

Figure 2: Manufacturing-specific customized views of "VF Forecast, Yield by Products" and "VF Forecast, Yield by Output"

Scrolling down the front page, the manager reviews the *VF Forecast, Yield by Products* view. Focusing attention on detail, the manager sees that the VF is not meeting yield goals.

From a manufacturing perspective this means that the VF will either miss committed output or require more product starts to stay on target. The manager elects to drill-down on yield to further analyze current activities and clicks the right thumbnail, *VF Forecast, Yield by Products* to display the active view of the data shown in Figure 3.

Figure 3: Drill-down in Active View to reveal Yield trending for specific product

Shifting the time frame to the current quarter, the graph displays a downward trend. With the ability to easily change time frames and indicators and dimensions to elaborate on the current problem, the manager deselects all products except ProdX, which seems to be presenting the problem.

Figure 4: Expanded data view revealing increased product yield in Technology Development (TD) sites

In Figure 4, data were expanded so that they could be viewed by site to determine if this downward trend is unique to a site, or if the problem is spread across the entire VF. By deselecting other sites, the active view can be simplified to focus on specific sites highlighting the yield anomaly. Analyzing apparent indicators, the manager concludes that all High Volume Manufacturing (HVM) sites are displaying a downward trend while Technology Development (TD) sites display an upward trend. This prompts questioning the engineering manager to determine what has changed since the last quarter and whether TD is developing some new Best Known Methods (BKMs). Electing to analyze the data further to elaborate on possible options, the manager modifies the time frame of the chart a second time to provide a larger window. Looking at prior quarter performance to analyze how yield was trending for the product, the manager evaluates that ProdX was steadily improving as expected.

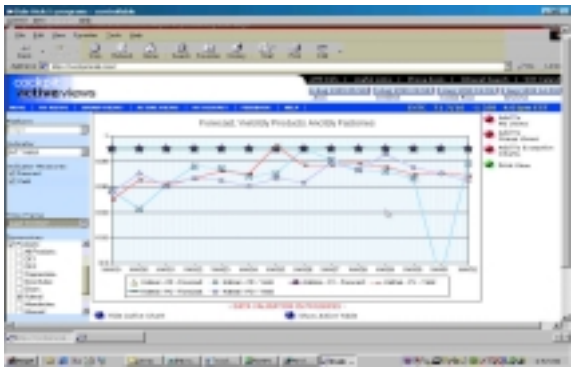


Figure 5: Enlarged time frame view revealing steady improvement with significant spike in Q2 yield

The manager now knows, as expected, that ProdX yield at HVM sites was improving as the product matured. At quarter end, however, all yields at HVM sites went down, but TD remained constant and even showed a slight increase.

Looking at a yield trend, several options are available to the manager:

- Determine course of action to improve yield.
- Consider modifying forecast (lower) to avoid over-commitment.
- Maintain the output forecast and either over-start units or lower Throughput Time (TPT) to compensate for lower yields.

In Figure 5, noticing that yield improved as expected with a maturing product, the view displays a visible *knee* at the 2nd quarter boundary. The manager wonders what new or different actions occurred: new stepping, new die, engineering change, etc.

Figure 6: Modified view to "VF Output, Actual vs. Scheduled" to reveal Yield-Output correlation

Electing to correlate the previous trend, the manager switches to *VF Output, Actual vs. Scheduled* displaying the same time period as the previous yield view. The analysis validates that while yield trended up through the end of the quarter, the VF consistently beat the output schedule.

When yield trended down, at the beginning of the new quarter, the VF consistently missed the output schedule. The manager concludes that yield and output are positively correlated. With this new information, management can take adequate action to improve yield.

Through the Cockpit's interactive live charts and drill-down capability, a correlation that could have taken days, even weeks to identify was detected in a timely manner, demonstrating the benefits of the Cockpit as a decision support system.

To facilitate early notification of similar yield-related problems, the manager can request that the application monitor the yield indicator. The manager can set a personalized threshold and be notified if yield is outside the acceptable range. This early notification will shorten the problem-solving cycle time without requiring constant indicator monitoring.

COCKPIT ARCHITECTURE

Architecture Design Guidelines

The single most important attribute of a software architecture design is the ability to handle change. The Cockpit's architecture is designed to adapt to ever-changing technology, user base, and business requirements. Optimized for application-specific requirements, the criteria driving the Cockpit's architectural design and technology selection included the following:

- maintainability, scalability, extensibility, and interoperability
- multidimensional query support and drill-down capability
- highly interactive active and static views
- various levels of user sophistication
- guaranteed and consistent performance for LAN, WAN, and off-line environments
- push/pull information delivery, coupled with the ability to provide scanning and notification information in near real-time.

The ATM Cockpit addresses these system and application-level requirements through adoption of an n-tier Internet architecture, utilizing subsystem interfaces that conform to open industry standards and the Intel internal architecture reference model.

The Cockpit's component architecture is based on Windows* Distributed interNet Applications (DNA)*, and its decision support implementation is consistent with Microsoft data warehouse and Intel DSS Utility® architecture frameworks.

Overview of Component Architecture

The Windows DNA backbone provides the foundation for the Cockpit's Component Object Model (COM) based components to easily deploy, scale, and interoperate in the Internet environment. Based on Internet Explorer*, the Cockpit's presentation layer relies on Microsoft Office 2000* Web Components (PivotTable and PivotChart) to present multidimensional data in table and graphical formats. Custom Cockpit ActiveX* objects provide user-friendly navigation that drives Office 2000* Web Components. These components facilitate seamless integration with Excel 2000*. For future extensibility of the presentation layer, the Cockpit's architecture supports replacement of these components with other vendor solutions. The Cockpit's business layer provides the security and personalization services necessary to ensure that the Cockpit application is secure and easy to use. The above services are provided through server-side components built on Site Server and Active Server Pages (ASP).

Site Server offers data persistence to the Cockpit's personalization information through a Lightweight Direct Access Protocol (LDAP) interface that provides the migration path to the future Windows 2000* Active Directory* service.

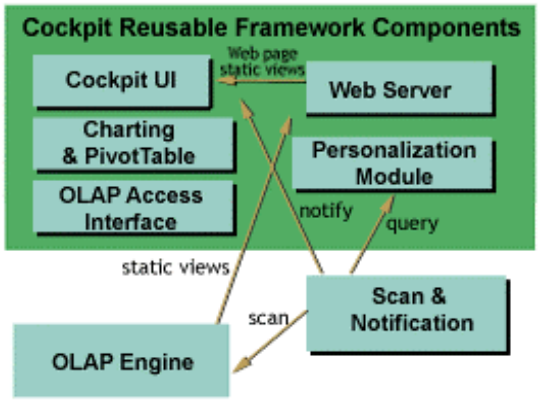


Figure 7: Reusable framework components for the Cockpit UI Indicator Reporting module

* Other brands and names are the property of their respective owners.

Reusable UI Framework

The largest benefit of the Cockpit's architecture is realized through the use of OLAP technology. The OLAP engine encapsulates multidimensional query process complexity from the Cockpit's user interface (UI) and business logic. Communication between client-side Office 2000 Web Components and OLAP cubes (data) are handled through the OLE DB† for OLAP (PivotTable Service) database standard on the client.

The abstraction layer, supplied by client-side PivotTable Service, allows the Cockpit client to connect to OLAP data on the server over the LAN, across a WAN connection, or on the user's PC. This abstraction layer aids in insulating the UI from the data source.

Any OLAP cubes implemented by other organizations are directly accessible from the Cockpit's UI and middle layer components. Other non-OLAP data sources or data structures can be accessed through a custom OLE DB for OLAP Provider.

Consequently, the Cockpit's UI and middle-tier implementation can be used as an enterprise UI framework for all OLAP applications within the company. This helps the bottom line by avoiding redundant cross-company development time and cost.

The Cockpit User Interface

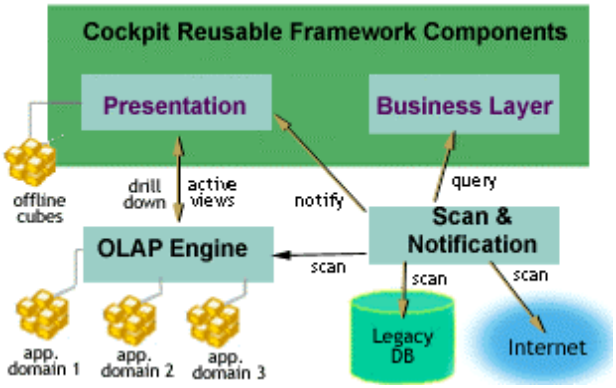


Figure 8: Reusable framework components for the Cockpit UI Scanning and Notification module

The Indicator Reporting Module

The Cockpit's Indicator Reporting module allows access to all information contained in the Cockpit OLAP

† The OLE DB for OLAP (PivotTable Service) database standard from Microsoft offers an SQL client/server solution with a standard access method to data and the ability to use various front-ends.

database. When we were considering the look-and-feel for the Cockpit user interface (UI), we initially reviewed several commercially available tools that provide Web-based OLAP analysis. Although these tools were generally quite capable, further research revealed several notable problems when it came to the Cockpit's business requirements.

Firstly, these tools are aimed at users with OLAP experience. The Cockpit's customer base of senior managers with minimal or no OLAP experience would have to undergo significant training.

Secondly, these tools allowed for little customization and would require a comprehensive single vendor solution. Clearly, our implementation would benefit from a customizable front-end and the ability to interoperate with industry standards at all levels of the application.

The Scanning and Notification Module

The Scanning and Notification module, as the name reflects, scans Intel's internal and external environment for potential problems and events of interest, and reports the results to the user via a push mechanism. The current implementation uses an active channel approach to deliver events to the user. This implementation requires a separate front-end and a small stock-ticker-like tool to run on the user's machine. We are exploring other simpler ways to disseminate the event information; Microsoft Outlook* and e-mail are two possibilities.

Overview of DSS Architecture Framework

The Cockpit's back-end data services comply with Intel's DSS Utility architecture framework. Intel IT DSS Utility partners provide OLAP cubes that are suitable for multidimensional query capability. These cubes adhere to an established extract, transform, and load process. All Cockpit indicator data resides in databases optimized for OnLine Transaction Processing (OLTP) in the manufacturing execution and other operational environments. Extracted data are sent through a cleansing process for storage in the corporate-wide data warehouse.

The Cockpit dependent data mart• is derived from the corporate-wide data warehouse. OLAP cubes are built from Cockpit's staging data mart. The resulting cubes can then be hosted on the same or on a different server to be accessed by the Cockpit's UI front-end.

DATA ORGANIZATION

Known Issues

At the outset of this project, we faced a number of challenges from a data perspective:

- Data was stored in a number of databases within the organization. To meet initial customer commitments, we needed to consolidate data from four separate systems: two manufacturing WIP systems, a planning system, and a safety and health system.
- There were no existing sources for common data, such as factory calendars, product masters, or factories.
- Use of OLAP technologies was relatively new at Intel and the selected product, Microsoft OLAP Services*, had just been released. No resources were available to draw on. There were no consultants, books, or classes.

Team Approach

To maximize utilization of resources available within the organization, and to minimize the number of developers we would need to hire, we implemented dimensional modeling techniques to create the data mart. This proved to be the easiest method to consolidate data from disparate systems, allowing for an easily supportable OLAP design. We decided to take a team approach to developing the data mart and so did the following:

- Implemented dimensional modeling techniques to create the data mart. This proved to be the easiest method to consolidate data from disparate systems belonging to different groups, allowing for an easily supportable OLAP design.
- Partnered with IT for technical assistance. IT procured and configured the hardware infrastructure, provided data loading expertise, and performed database modeling.
- Worked with data owners to understand data sources and design data validation routines. We developed cross-departmental relationships to quickly understand data sources. Because these departments

* Other brands and names are the property of their respective owners.

• *Dependent data mart* is a type of data mart that acts as a subset of a larger data warehouse which combines databases across an entire enterprise. Data marts are usually smaller than larger data warehouses and focus on a particular subject or department.

traditionally owned the data, they were interested in ensuring that it was accurately reported.

Database Server and OLAP Engine Selection

Due to the development time frame for delivery of the solution to upper management, little time was available to perform a complete evaluation of various database server and OLAP engines. With only three months to delivery date, we selected Microsoft's SQL Server 7.0* and OLAP Services*.

Many developers on the team were already familiar with the SQL Server and knew that it would provide adequate power to support a database size of 10-20GB.

We then began work on creation of the initial data mart, and we selected dimensional database modeling as the design path. Much of the Cockpit's design was based on the teachings of Ralph Kimball, an expert in the field of dimensional modeling [1].

Dimensional Modeling

Dimensional Modeling (DM) is a logical design technique that attempts to present data in a standard, intuitive framework allowing for high-performance access. Essentially, this is a relational database modeling technique, but with some important restrictions [3]:

- Every dimensional model is composed of one table with a multi-part key, called the fact table. This table typically contains one or more numeric facts or measures.
- The model also includes a set of smaller dimension tables, each having a single-part key. These tables contain descriptive information about such things as products, factories, and time. Dimension tables are used to describe the facts in the fact table.
- When these tables are *joined* together, they create the characteristic *star-like* structure, commonly called a star join.

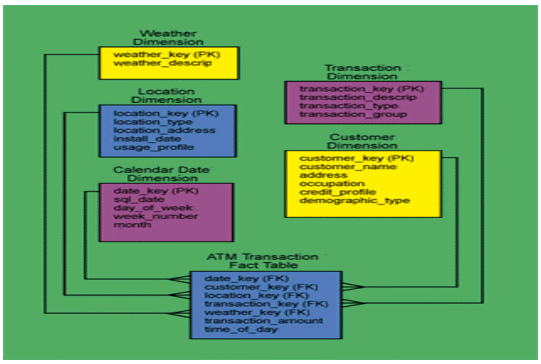


Figure 9: Typical star join dimensional model [6]

Dimensional models are inherently additive. Users rarely ever retrieve a single record from the table, but instead retrieve hundreds, thousands, or even millions of records at a time. The only useful operation to perform on such a large number of records is to add these items [4].

Figure 10 displays the facts Units Shipped. This fact is described at each intersection of the dimension tables; in this case, Products, Time, and Factories.

By simply filtering the result set, you easily obtain the number of units shipped for any combination of products and factories during a given time period. From an end-user viewpoint, this type of database design is easy to understand and use [2].

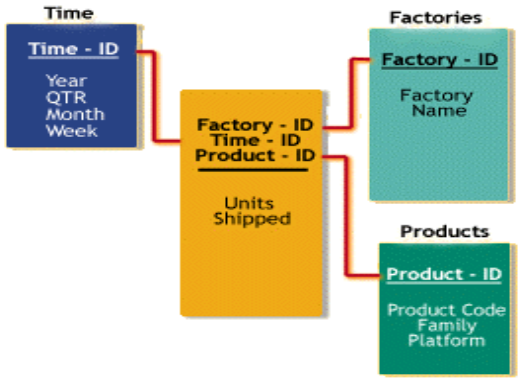


Figure 10: Dimensional Model composition displaying fact and Dimension Table relationships [2]

Data Population

Populating the dimension tables proved the most difficult part of this task. Most of the work was not technical in nature, but required a large amount of business analysis to determine how products are categorized, and how to gather the various shift schedules from each factory.

The product dimension was the most difficult table to populate, due to the need to cross-reference shop floor product codes used in manufacturing and planning systems, with common product *nicknames* used by most

* Other brands and names are the property of their respective owners.

end-users. This information did not exist in any database. Instead it resided in spreadsheets, e-mail messages, and with individuals in factories. After gathering the data, we built and populated an entire set of cross-reference tables to maintain attributes in the product dimension table.

Initial customer requirements indicated the need to view weekly data, with a possible future desire to view daily data. Viewing data at a factory shift level was not a requirement. During the design, we decided to store it at the lowest level possible, and chose to store data in fact tables at a shift level, just as a precaution. This later proved advantageous due to a change in customer requirements requesting shift level data.

After gathering requirements and identifying data sources, data analysis consumed at least 50% of the total time allocated to development. Due to the size of the development window, full up-front data analysis was not possible. During data analysis, we leveraged relationships with other groups within the organization to identify the data experts to help construct the proper queries to extract data. Once data analysis for each indicator was 70-80% complete, Extract, Transform, and Load (ETL) work started. Developers in IT used Informatica* to visually extract data from the source, transform it using a variety of business rules, and load it into the data mart. Data loading jobs were scheduled to load all dimension tables first, followed by fact tables.

OLAP (On-Line Analytical Processing)

OLAP provides organizations with a means of accessing, viewing, and analyzing data with high flexibility and performance. First and foremost, OLAP presents data to end-users through a natural, intuitive data model. Second, OLAP accelerates delivery of information to end-users viewing these multidimensional structures by preparing some computed values in the data in advance, rather than at execution time. The combination of easy navigation and fast performance allows end-users to view and analyze data more quickly and efficiently than is possible with relational database technology only. The end result is more time spent analyzing data and less time analyzing databases.

In an OLAP data model, information is conceptually viewed as cubes, which consist of descriptive categories (dimensions) and quantitative values (measures). The multidimensional data model makes it simple for users to formulate complex queries, arrange data on a report, switch from summary to detail data, and filter or slice data into meaningful subsets. Within each dimension of an OLAP data model, data can be organized into a hierarchy that represents levels of detail on the data. For example, within the time dimension, there can be years, months, and days levels. Similarly, within the geography dimension, there can be country, region, state/province, and city levels. A particular instance of the OLAP data model would have the specific values for each level in the hierarchy. Users viewing OLAP data can move up or down between levels depending on whether they want more or less detail [5]. One common misconception is that OLAP technology is similar to that of relational databases. OLAP cubes are queried using multidimensional expressions or MDX. Though MDX shares a number of keywords with SQL, such as SELECT, FROM, and WHERE, there is little similarity between them.

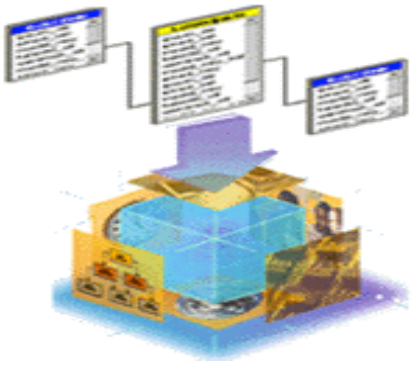


Figure 11: Multidimensional OLAP cube, dimensions and relational schema [7]

Before engaging in a full OLAP development project, it is advisable to train developers in the differences between the two. For our team, however, creating OLAP cubes was relatively easy, mainly due to the design of the underlying data mart. The majority of issues encountered involved implementing MDX inside of calculated measures. Once data was *cubed*, data validation became increasingly important. Some of the validation was straightforward, especially where standard reports existed for comparison. However, we encountered challenges validating OLAP cubes against spreadsheets maintained by users who gathered information from a variety of sources.

PRESENTATION LAYER

Application Goals

From an application requirements perspective, the Cockpit's presentation layer has two main goals: to present an intuitive user interface that allows managers to view key indicators across the VF and to inform management of events related to internal business or supply chain situations.

The Indicator Reporting module addresses the goal of presenting an intuitive UI that supports and easily maintains a catalog of important, personalized data views,

as well as performs ad hoc analysis in support of high-level decision making. The Scanning and Notification module addresses the second goal, informing the manager of events related to either internal business situations or supply chain related events. These two elements, appearing as two separate applications, work in unison and share content to provide the total solution.

Ease of Use

Although the goal was to design a very simple UI, we wanted to ensure that the design did not limit the usefulness of the tool as users gained expertise. For users willing to learn, we needed to provide additional capability. To accomplish this goal, Microsoft Office Web Components* (MSOWC) were selected as the foundation for the UI. These components are compatible with the underlying data tier and have the majority of the functionality provided by other OLAP tools. Additionally, the look-and-feel of the components is familiar to Microsoft Office* users and is part of the Intel standard application suite. Nevertheless, these components are difficult for the novice user to operate. While no knowledge of the tool is needed to start using it, full functionality is available by selecting *Expert Mode*. To this end, we built a simple-to-use auxiliary set of controls for viewing and analyzing OLAP data, consisting of two list boxes and two tree view controls.

Figure 12: Left panel displays auxiliary set of controls for viewing and analyzing OLAP data

Using these controls, users select subject area, choose measures to view, and then *slice and dice* data according to dimensions of interest. Additionally, we developed an automated time dimension matching the Intel calendar. Continuing to manage all OLAP interaction through MSOWC, these auxiliary navigation controls succeed in hiding details and expert style navigation from the user.

Metadata-Driven UI

Having elected to provide custom navigation controls, it was important to design the framework so that the number of OLAP cubes accessible by the application could be easily increased. To achieve this goal, we adopted a metadata-driven method of handling navigation information. When cubes are processed on the database server, several XML files are created to include metadata about each cube that is required for the UI to populate navigation controls.

To include a new cube in the application, it needs only to be designed according to Cockpit guidelines and hosted on the local server. A few simple modifications to an XML file will cause the UI to present the cube and all of its information to the user through the same user friendly interface.

Scalability and LAN/WAN Performance

Though the Cockpit interface was well received and provided quality performance for those with high-speed connections to the server, there were significant performance problems for those across long WAN links. This is due to our adopted rich client model. If users in Asia tried to access the Cockpit's cubes hosted on a server in California, response times for certain operations increased by several orders of magnitude. Moving all processing to the server-side would have taken away the very interactive experience provided by the rich client model.

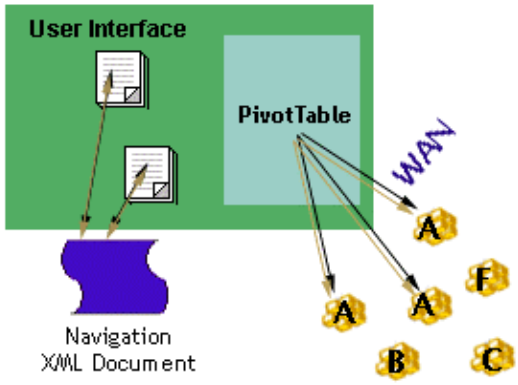


Figure 13: Metadata-driven method for navigation with XML files created during processing on database server and distributed cubes for performance

Instead, we chose to replicate the OLAP cubes on an array of OLAP servers at each remote site. Since the

* Other brands and names are the property of their respective owners.

presentation layer was primarily data-driven, we were able to add an entry into the XML cube description files to specify servers that had cube copies. Now, when a user connects, the UI can connect to the closest database server. This has significantly reduced response times at remote sites without requiring major changes to the presentation layer.

Multiple Form Factors

One of the original goals of the Cockpit application, to provide a rich and interactive user experience while analyzing OLAP data, led to the decision to host MSOWC in the browser and handle the majority of report generation on the client.

However, as the application's user base increases and the data scope widens, it will become necessary to support various client form factors in the near future. These clients may include Internet Explorer 5.0* hosting all ActiveX controls locally, a pure HTML browser with no ActiveX control capability, or even a Windows CE* or Palm OS* device.

All of these form factors should be able to benefit from Cockpit content while providing users with an appropriate level of interactivity. To support these multiple form factors, Cockpit components are capable of providing most of the reporting content in static image format consumable by almost any client.

END-USER PERSONALIZATION

Importance to Manufacturing

The Cockpit's target users belong to such domains as manufacturing, planning, logistics, materials, finance, and human resources. These users play different roles within each of these domains. For example, the manufacturing area includes factory and site managers, process engineers, quality engineers, and others with each role viewing data from a different perspective. The OLAP solution offered by Cockpit is optimal for *slicing and dicing* data, allowing users to pinpoint views of interest. It provides factory-level output figures to factory managers and provides process engineers interested in the same output with detail-level information for key constraint equipment.

Personalization Engine

Instead of providing a fixed number of static reports categorized by user types/roles, we elected to enable dynamic report generation, usage and retrieval. To support this functionality, the system should remember each user and maintain a user-specified personal configuration store. The personalization engine, a middle-tier service based on Site Server, handles personalization in the Cockpit. All registered Cockpit users have a dedicated account defined in the Site Server database with each user account holding four data segments:

User Defined Views: Users can create their own reports on-the-fly and save them for later retrieval. The user's personalization store saves, in XML format, only the necessary connection and parameter information. When a user selects a saved view, XML is retrieved from the personalization store and creates the report using the most recent OLAP cube data.

Front Page Views: Users have the ability to specify which of their favorite views will be displayed on the Cockpit's front page. Generated on the server, these views are rendered as images and served to the user's machine on demand.

Subscription Views: Users can receive selected views as images by e-mail or desktop channels.

User-Defined Exceptions: The normal trend is for users to analyze various static or active views and determine the current state of operations. Manual in nature, this process consumes valuable time. Instead of requiring users to scan reports and make decisions, the Cockpit's user-defined exception facility is designed to automate the report-checking process. Users have the ability to set exceptions at the indicator level, instructing the system to check for exceptions each time the data set changes. The system notifies the user via e-mail or desktop channels when it detects an exception, eliminating the need for manually checking of reports to identify the exception.

Scanning and Notification

The Indicator Reporting module excels at providing easy access to a wide array of data in the OLAP database. However, all data provided by the tool exists within the company, is quantitative, and requires users to proactively search for information. The Scanning and Notification module was designed to compliment the Indicator Reporting module by scanning the business environment for relevant qualitative and quantitative information, and pushing this information to the user in real-time.

* Other brands and names are the property of their respective owners.

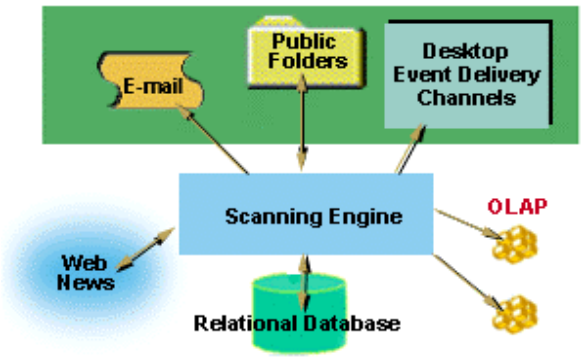


Figure 14: Scanning and Notification module architecture

Scanning Engine

The scanning engine is an automated agent that monitors a variety of sources for information on behalf of the user. The information gathered by the agent is passed to the user through a variety of channels. The two primary types of information monitored by the agent are Web-based news feeds and internal databases. News feeds are pre-defined by a system administrator and organized into public channels with sub-channel categories. If the channel is of interest, the user can subscribe to it.

For example, there is a channel that monitors news feeds for information regarding our largest customers. The agent delivers news headlines to the user for scanning. If an article is of interest, the user can click through to the complete text. Additionally, the tool can monitor internal databases, including OLAP cubes behind the Cockpit Indicator Reporting module, for a mixture of personalized and predefined trigger points.

For example, if the user is always interested in the yield of a single product at a manufacturing site, that user can create and track the exception. An exception wizard will walk the user through the process of choosing which conditions to monitor. Generally, an exception can be triggered if it crosses a fixed value or deviates from a goal by a certain percentage. Assuming all the necessary exceptions are being monitored, users do not need to poll the Cockpit's Indicator Reporting module to ensure the factory is running within acceptable limits. Instead, managers can be assured that they will be notified when they need to take action.

Delivery Channel

Once the information is found, it needs to be pushed to the user in an easily consumable, timely way. Since type of information varies by user, the Cockpit provides a variety of methods for pushing this information to the user. The first method, pushing information through e-mail, is appealing since users are comfortable with the technology, and it provides a fairly real-time delivery mechanism.

Figure 15: The Cockpit Scanning and Notification module, desktop event delivery mechanism

The downside is that important notices may get lost in the Inbox along with the huge volume of mail. Additionally, it is not very efficient for news headline scanning. To overcome these shortcomings, a second method is offered that provides a faster response time and information categorization, but requires a new application.

This application runs as a desktop event delivery mechanism organized according to channels and categories mentioned earlier. It uses a true push mechanism that notifies the user immediately when new information is available just by changing icon colors and alert lights on the application. Users can view this information at their convenience. These two choices are just the first of many that may become available.

Future implementations could easily include such channels as a Web portal or public folders. Since information agent and delivery channels are separate, this same information could be pushed through almost any channel. Users will be able to select the channel which is right for their choice of information and is the most comfortable.

BENEFITS

Impacts of the tool on manufacturing operations have been immediate, and are both quantitative and qualitative. A problem or issue that once took 3-4 days plus many people and multiple meetings can now be identified in less than 30 minutes. A factory manager can now see *upstream* in the production pipeline to view the status of production moving his or her direction, allowing for better planning, more agile response to change, etc. The VF management team now works off the same data for decision making and problem solving. Instead of a manager directing a subordinate to go to the source data

system and generate a useful summary, managers can now generate the data for themselves, instantly. With the Cockpit as enabler, the ATM VF management team can perform better, faster, and cheaper than before and proactively respond to dynamic demand and supply chain operations variability.

CHALLENGES

There are four key challenges: reliably deliver new data and product capability quickly, extend the Cockpit to new consumers, continue vigilance on data quality and reliability, and drive to near real-time data currency in support of supply chain management.

The team will continue the successful approach of letting the business drive priorities and deliver the critical few every 90 days. The tool will be expanded to support additional information domains and extend product improvements. Future plans include advanced data filtering, better graphical display, and a *thin client* version.

The Cockpit team will expand the base of consumers by teaching and helping others. A Linux*-like federated development approach will be implemented, and the product will be given to Intel's IT DSS Group to offer to other customers with similar needs.

To continue the focus on data quality, we are going to drive data owners to standards, drive use of data warehouse utility, clean up manufacturing data models and create data quality focus groups for continuous improvement. Finally, to support the supply chain *Glass Pipeline*∅ requirement, working groups will be established to identify and drive requirements for supply chain visibility.

CONCLUSION

The ATM Cockpit is adding business value today by helping ATM preclude and solve problems faster and make decisions faster. It fills a key business management capability gap for VF operations and will help ATM meet the competitive challenges of tomorrow. Along with the outstanding contributions of the extended ATMis development team, the tool and methods employed are breakthroughs that give Intel an edge in this *brave new world*.

ACKNOWLEDGMENTS

We thank David Marsing and Mike Splinter for giving us an opportunity to fail. We also thank OLGA/Flip Chip Virtual Factory for providing guidance and direction.

We acknowledge the Extended Cockpit Development Team and the IT and Enterprise Application Groups for their technical contributions. And, we thank Natasha Du Bois for the technical editing/writing, and research.

REFERENCES

[1] Kimball, R., *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley & Sons. 1996, pp. 370-388.

[2] Kimball, R., "A Dimensional Modeling Manifesto: Drawing the Line Between Dimensional Modeling and ER Modeling Techniques," DBMS and Internet Systems, August 1997. http://www.dbmsmag.com/9708d15.html

[3] Thomsen, E. et al., *OLAP Solutions: Building Multidimensional Information Systems*, John Wiley & Sons 1997, pp. 575-592.

[4] Kimball, R. et al., *The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing and Deploying Data Warehouses*, John Wiley & Sons, 1998, pp. 790-800.

[5] "Microsoft SQL Server 7.0 OLAP Services Whitepaper," Microsoft Corporation 1998. http://www.microsoft.com/sql/productinfo/olapservices.htm

[6] Kimball, R., "Help for Dimensional Modeling Helper tables let you design and manage multivalued dimensions successfully," DBMS Online, Miller Freeman, Inc. 1998. http://www.dbmsmag.com

[7] Becker, D., "How to Build Business Solutions with Microsoft SQL Server OLAP Services and Excel 2000," Microsoft Corporation 1999. http://microsoft.com/Seminar/1033/19990826TQN1007DB1/Seminar.htm

[8] Poirer, C., "The Path to Supply Chain Leadership," Supply Chain Management Review, July 1998. http://www.manufacturing.net/magazine/logistic/archives/scmr/07path.htm

* Other brands and names are the property of their respective owners.

∅ Glass pipeline is an (Extranet) public communication system with privileged and protected configuration that allows value-chain constellation constituents and their consumers to determine location/movement of goods and services. [8]

AUTHORS' BIOGRAPHIES

Paul Calame is a 23 year Intel veteran. Paul has held a variety of management roles in Manufacturing, Automation and IT. He received a B.S. degree in geology and political science from Iowa State University in 1976. His e-mail address is paul.calame@intel.com.

Ravi Nannapaneni received a B.S. degree in Computer Science from Andhra University, Waltair, India and an M.S. degree from Arizona State University, Tempe, Arizona. He has been with Intel for nine years and currently manages the Cockpit & Emerging Tools and Capabilities (ETC) group within Intel's ATM organization. Ravi's prior duties included IT Engineering design support for new Intel international site startup activities and introduction of Information Business technologies to Intel's HVM facilities. His e-mail address is ravi.nannapaneni@intel.com.

James Yu holds an M.S. degree in MIS from the University of Arizona. Before that, he also studied under the M.B.A. program at the University of Texas and holds a B.S. degree in Diplomacy from the National ChengChi University in Taipei. James Yu is the Architect for the Cockpit ETC team and has been in a software architecture role within Intel's IAL and the IT organization. Prior to joining Intel, James spent 12 years with Honeywell's Industrial Automation & Control Division in various software development, design, and architecture positions. His e-mail is james.h.yu@intel.com.

Jay Turpin received a B.S. degree in marketing from Eastern Michigan University, Ypsilanti, Michigan. He has been with Intel for three years and is currently in charge of "all things data" for Cockpit ETC group within Intel's ATM organization. Jay's prior duties included designing and developing the databases to support EATS yield analysis application and managing the corporate computer systems at Honeybaked Ham Company in Troy, Michigan. His e-mail address is jay.turpin@intel.com.

Scott Peterson received B.S. and M.S. degrees in electrical engineering and an M.B.A. from the University of Illinois at Urbana-Champaign. He has been with Intel for four years and is currently a Senior Developer in the Cockpit ETC group in Intel's ATM Organization. Prior to joining this organization, he worked on communications and networking initiatives in the Intel Architecture Lab. His e-mail address is scott.peterson@intel.com.

Intel's Internet Connectivity: Evolution, Technical Architecture, and Future Directions

Cindy Bickerstaff, Intel® Online Services, Intel Corporation
Sally Hambridge, Intel Online Services, Intel Corporation
Lynne Marchi, Information Technology, Intel Corporation
Tod Oace, Intel Online Services, Intel Corporation
Stacy Purcell, Information Technology, Intel Corporation
Jeff Sedayao, Intel Online Services, Intel Corporation
Charles Smothers, Information Technology, Intel Corporation
Ken True, Intel Online Services, Intel Corporation

Index words: the Internet, connectivity, firewalls, security, performance, measurement

ABSTRACT

This paper describes Intel's Internet connectivity architecture, including the business drivers that led to its creation and the technology developed to build and maintain it. Intel's first Internet connection was a 2400 bit per second modem used to pick up and deliver e-mail for a small community of engineers and researchers. With the advent of the Mosaic Web browser, Intel needed to develop a scalable, high speed, highly available Internet connectivity architecture that is secure, available, provides good performance, and minimizes the impact of Internet usage on Intel's internal network.

An architecture was developed and implemented that provided multihomed Internet gateways at major Intel sites. A screened subnet firewall architecture provides defense and depth. Several fail over modes were implemented to maintain connectivity even if a number of gateway components failed. To manage this distributed gateway implementation, we borrowed methods from Intel's manufacturing world such as Copy Exactly! and certain measurement and experimental methodologies and statistical techniques. We modelled our set of distributed gateways as a single system and drove configuration through a Makefile.

Our architecture has proven highly successful. It supports Internet access for tens of thousands of Intel employees' Internet access, and Intel does $1 billion a month in e-Commerce through this infrastructure. Current challenges include maintaining performance while dealing with growth and security threats. Future uses of the Internets are dial-in modem replacement, intercompany connectivity, virtual private networking, and streaming media.

INTRODUCTION

Intel initially connected to what would become the Internet in 1986 [1]. The first connection took place using a 2400 bit per second modem that was used to pick up and deliver e-mail for a small community of engineers and researchers within the company. Connectivity eventually evolved into a direct leased line into the Internet. With the advent of the Mosaic Web browser, Internet usage became mainstream. As we foresaw that the Internet would become as important a tool to business as the telephone, Intel's Information Technology group faced a number of challenges in providing Internet services:

- provide Internet services with good performance
- scale service to deal with increasing numbers of servers, services, and users
- make Internet services like Web access and e-mail highly available and robust
- ensure that our Internet gateways are secure
- minimize any impact of Internet traffic across Intel's internal network

To meet these challenges, we implemented an Internet connectivity architecture that provides Internet gateways at major Intel sites. Connecting to at least two Internet Service Providers (ISPs) at each site increased availability and performance. We implemented a screened subnet firewall architecture to provide defensive in-depth security, and we made our gateway capable of several different fail over modes in order to make it as highly available as possible.

The major challenge of a distributed Internet gateway system is managing it. We had to ensure that gateways were operating correctly, the Internet service was good, and consistent security policies were being enforced. Moreover, the staff maintaining Intel's Internet gateways was fairly small, less than ten people. To manage the Internet gateway system, Intel's IT took a number of concepts from Intel's manufacturing arm, such as Copy Exactly! [2], a technique to simplify deployment and maintenance of systems and configurations. We also used measurement and statistical methodologies developed and honed in wafer fabrication plants to measure and monitor Internet performance [3]. In addition, we simplified our maintenance by modelling our distributed gateway system as a single computer and the router and server configurations as software modules to be compiled and installed [4].

Our architecture has proven highly successful. It supports tens of thousands of Intel employees' Internet access, and Intel does $1 Billion a month in e-Commerce through this infrastructure. We have managed to scale almost all of the components of the architecture, from bandwidth and ISPs to servers and users. Policies and techniques developed while implementing this architecture have gone into Internet Standards [5] and industry white papers [6], and many of the techniques developed are being implemented in Intel's new business unit, Intel® Online Services.

The constant challenge that we have seen is keeping up with increasing usage of the services we offer and the number of new services that users demand. Another key challenge is dealing with the ever increasing and changing security threats. In the future, we see the Internet used increasingly for dial-in modem replacement, intercompany links, and for virtual private networking. We foresee a time when every site will have its own Internet connection, although we see challenges to this vision in non-US locations where bandwidth costs are problematic [7].

This paper describes how we designed and implemented Intel's Internet connectivity architecture. It covers the early drivers for connectivity, the requirements and design issues for a connectivity architecture, and the technologies and policies needed to implement and maintain the architecture that we built.

EARLY DRIVERS FOR INTERNET ACCESS

Intel first connected into what would become the Internet in 1986 [1] via an organization called CSNET. The primary driver for connectivity was the exchange of e-mail between Intel engineering organizations and consortia outside of Intel, such as the Semiconductor Research Corporation and Sematech. The Internet environment (then the ARPANET) was much different during that time period. It was designed primarily as a research environment, and there were explicit restrictions on commercial use. Direct connections through leased lines were expensive and not easy to come by, so Intel exchanged mail through a system called PhoneNET. Intel dialed up a server, dropped off outgoing e-mail, and picked up incoming e-mail. All this happened through a 2400 bit per second modem.

The power and utility of being able to communicate with people in other organizations electronically made e-mail use grow exponentially. This was despite the fact that Intel didn't offer classes or documentation on how to use the Internet and did not even advertise the fact that we were connected!

By the late 1980's, large archives of freely distributed and highly useful software began to appear. There was increasing demand for access to these archives, for which direct connectivity to the Internet was necessary. As Internet mail use at Intel grew exponentially, the dial-up charges began to approach the costs of a leased line. At that point, Intel obtained one 56 Kilobit leased line to the Internet for general use by employees. The Internet then was largely used by engineering personnel and systems administrators. Systems with Internet access were scattered haphazardly across the company and managed by a variety of groups.

The Mosaic browser and the World Wide Web brought the Internet into mainstream use at Intel. Before then, the Internet was largely text-based. It was mainly used for e-mail, FTP (file transfer), and remote login. The menuing system that preceded the Web, Gopher, was mostly textual. The Web's use of the in-line graphics increased Internet connectivity bandwidth requirements. As the Internet became commercialized and the Web began to take off, Intel needed to seriously revamp the way it connected to the Internet. At that point, we knew Internet connectivity and services were as critical as the phone. The state of Internet services at the time was haphazard and not secure. Servers used to provide Internet access

and provide Internet services like DNS were scattered across the internal network and inconsistently managed.

CONNECTIVITY ARCHITECTURE AND IMPLEMENTATION

We knew we had to move to an Internet connectivity architecture that supported Internet service as a robust, secure service used by Intel employees, customers, and suppliers. This section talks about how we designed and implemented an architecture that met that goal. First we discuss key requirements and design tradeoffs. We then describe the connectivity and security design, followed by a section on how we maintain our large distributed system. Finally we discuss how we ensure that we receive quality connectivity service, and how we crafted policies to guide Internet usage.

Key Requirements and Design Tradeoffs

With the growth of the World Wide Web and the increasing use of the Internet by all employees, we decided to view the Internet as a key utility, as basic and necessary a business tool as the telephone. Like telephony service, access to web sites and electronic mail exchange with other companies would be used by everyone, not only by a small community of engineers and scientists. With that as a model, several requirements immediately came to mind:

High Availability and Reliability. Internet services had to be as available and reliable as the telephone.

Security and Accountability. The service could not leave Intel vulnerable to intruders and could not be vulnerable to denial of service attacks. The Internet architecture also had to allow Intel to keep track of employee use so that data were available when deciding on growth. Also, Intel needed to be able to track possible misuse by employees.

Good Performance. Our architecture needed to perform well so employees could be productive in their use of the Internet. Good performance is also related to security: a system that is not performing well enough to be usable will be worked around, usually in a not secure way.

Maintainability. The staff maintaining our Internet connectivity was small and was not going to grow rapidly. Whatever architecture was implemented needed to be easily maintained remotely by a small staff. When something went wrong, we needed an architecture that could deal with it and continue to work.

Scalability. In the early years of Internet connectivity, we saw exponential growth in traffic to the Internet year after year. Whatever we designed had to be able to scale in almost every way we could imagine. We knew that the number of employees using this infrastructure and the time they spent using this infrastructure would constantly increase. Our architecture needed to be able to grow quickly to meet this demand—from increases in bandwidth to the number of servers providing service.

Impact on Intel's internal Network. Intel has a number of critical internal applications, such as chip design and semiconductor manufacturing, as well as its internal e-mail system. Use of the Internet should not impact these critical uses of Intel's extensive internal network.

Critical Decisions

A number of critical decisions had to be made at this point:

- Should the gateways be centralized or distributed.
- Should Intel connect to one ISP or several. If more than one, how many.
- How should the network be laid out so that it is highly available, secure, and yet scalable.

Centralized or distributed gateways. There are several tradeoffs between having a single gateway with centralized services and having several distributed gateways. Having a single gateway is much easier to manage. It is also better from a security standpoint, as there is only one gateway to watch and control. A single gateway is also much cheaper to provision and maintain.

Having several gateways is better from an availability standpoint, especially if you can fail over traffic from one gateway to another. Also, if traffic is directed to the closest of several Internet gateways, the amount of traffic is minimized on the Intel internal network. It is also easier to deal with traffic growth when there are several Internet gateways. If one of the gateways becomes congested, only the users of that gateway are affected. Moreover, traffic can be moved to different gateways making it unnecessary to upgrade gateway resources.

For Intel, the availability requirements and the need to keep Internet traffic off of the internal network drove us to have distributed Internet gateways. Since gateways are not free, we decided to have Internet gateways only at major Intel sites—sites with thousands of employees. This minimizes traffic on our internal network while keeping the number of gateways manageable. Two key concerns with having multiple gateways are manageability and security. These are discussed later.

ISP Connectivity—one vs. many. Another design choice was ISP connectivity. The tradeoffs between having more than one ISP at all the gateways are between manageability, performance, and availability. Having one ISP is clearly easier to manage, but having only one at all of our gateways leaves us vulnerable to a problem at that

ISP that propagates across the entire Internet and brings all gateways down. We have seen this kind of problem occur on a number of occasions. Having multiple ISPs is harder to manage from a troubleshooting and a vendor perspective, but it provides higher availability. Multiple ISPs perform better because a packet to and from our gateways could travel between Intel and other Internet sites more often without having to transit a peering point. Consequently, we decided on two ISPs. More would provide better performance, but two are manageable and cost effective and still meet performance and availability goals.

Network layout. The network layout has to be secure, scalable, highly available, and perform well. We needed a network that would still work and still be secure if a single component failed. To achieve that goal, we made each of our Internet gateways capable of handling traffic for another gateway. To implement defense in depth, we used the screened subnet firewall design [8]. To make sure that the design was scalable, we designed all of our ISPs and firewall complexes to land on a specific Ethernet segment so that additions and changes would have minimal impact.

Internet Connectivity Architecture

Let's look at the network layout in more detail. At the highest level of abstraction, our Internet connectivity architecture looks like that shown in Figure 1:

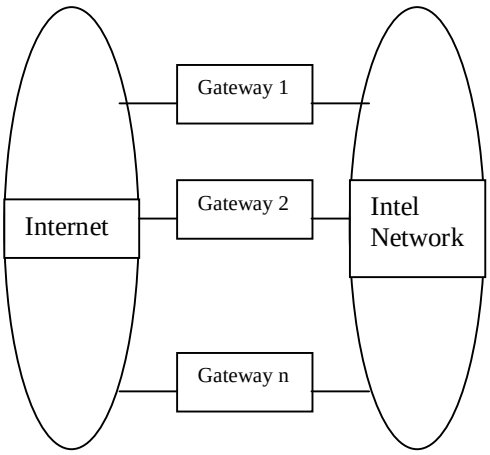


Figure 1: Gateway connectivity between Intel and the Internet

There are a number of Internet gateways between Intel and the Internet, each located at a major Intel facility. Systems within Intel are configured to use the Internet gateway closest to them (typically on the same site) when they want to utilize services such as Web access. Going to the closest gateway minimizes the impact that Internet traffic has on Intel's internal network. Also, if one gateway goes down, traffic can be automatically rerouted to another gateway. This feature increases the availability of Internet service. If necessary, the architecture can be scaled by either adding gateways or increasing the bandwidth between a gateway and the Internet.

At the next level of abstraction, each gateway looks like that shown in Figure 2:

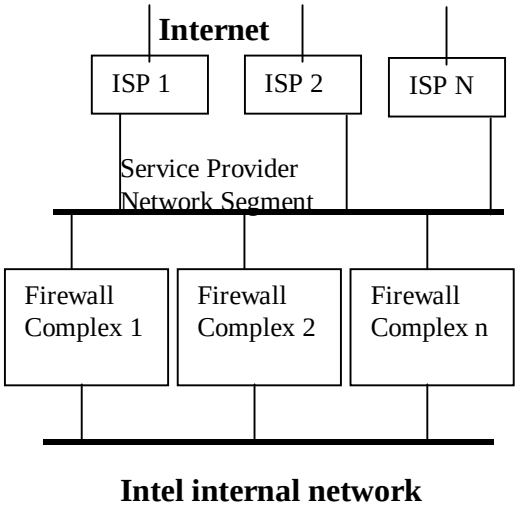


Figure 2: Architecture of an individual gateway

Each gateway consists of one or more firewall complexes connected to a service provider segment. Firewall complexes are groups of systems that have some common purpose, such as Intel's Web site or systems that provide basic Internet services. Most gateways will have one complex just for basic services, while some gateways at Intel have as many as four firewall complexes.

Two or more ISPs connect to this segment. This network layout has a number of advantages. With traffic flowing through the system, an ISP can be added, deleted, or upgraded. For ISPs that manage the router on their customer's premises, the service provider makes an excellent point of demarcation between the ISP and Intel. Also, each firewall complex can have its own individual routing policies with the ISPs. This allows us to tune performance for a firewall complex by adjusting routing.

This design can be scaled in a number of ways. The bandwidth can be increased by bringing in an ISP in parallel to the current connection and then cutting over.

Additional ISPs can be brought in if necessary or the number of firewall complexes can be increased.

This design contains a number of high availability features. If any single ISP goes down, traffic can be rerouted through another ISP. If the service provider segment goes down, then traffic can be routed through Intel to another Internet gateway. Firewall complexes are independent of each other. If one complex has a problem, that problem is usually isolated to that complex.

Security Architecture of a Firewall Complex

At the next level of abstraction is a firewall complex. We used a "defense in depth" approach in designing the architecture of our firewall complexes. Defense in depth relies heavily on the use of multiple components in the firewall to reduce the risk of an intrusion. An attacker must compromise more than one firewall component before gaining access to Intel's network—there is no single point of failure. While this approach does not guarantee the security of our internal network, it definitely makes it harder and more time consuming for an attacker.

A firewall complex is designed using the screened subnet architecture [8] shown in Figure 3.

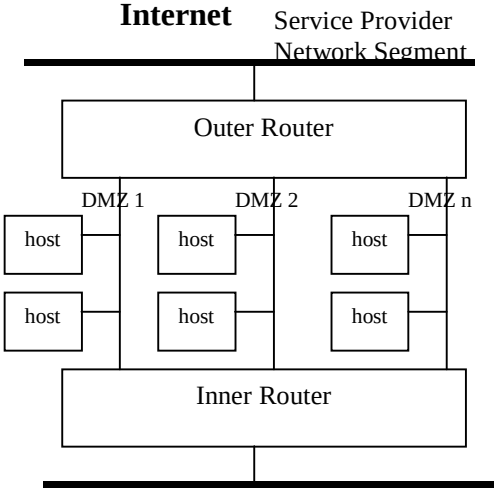


Figure 3: Basic design of a firewall complex

The firewall components are the outer router, the Demilitarized Zones (DMZs) and the inner router. The outer and inner routers are responsible for controlling both incoming and outgoing traffic going to the various DMZ network segments. The outer router allows communication between the Internet and the hosts in the DMZ while blocking direct communication with Intel's internal systems. In addition, the outer router blocks specific well known attacks. The inner router heavily filters communication between Intel's internal systems and the DMZ servers. Access by DMZ systems to Intel's internal network is restricted because the DMZ systems are exposed to attack from the Internet and cannot be trusted.

The method used by the routers to control traffic is called packet filtering. Each packet received by the router is compared against a set of predefined rules. These rules or access lists determine whether or not each packet is allowed to pass through the firewall and continue its journey towards the destination system. An access list consists of a source IP address a destination IP address, and a port number that denotes the service being requested.

The systems that sit in the DMZ are called bastion hosts. Each bastion host performs a specific set of functions. For example, one server provides proxy services, while another server acts as an SMTP mail relay and DNS server. This separation of services limits the damage that can be caused by a break-in, since the SMTP/DNS server does not run the proxy services and vice versa. In addition, the underlying operating system on all of the bastion hosts has been made more secure: unnecessary services have either been disabled or removed. Special programs have been installed that limit access or provide additional security. For example, the "sudo" program defines which administrators can execute privileged commands, and the Secure Shell program provides administrators with an encrypted tunnel so that an intruder cannot pick up passwords. We also run a program daily that compares the files on each of the bastion hosts with a set of master files. Differences are noted and sent to our system administrators for further investigation.

This design can be scaled in a number of ways. To deal with increased demand, more servers can be added to a DMZ segment. If more specialized services are needed, such as streaming media or authentication services, DMZ segments can be added.

Services Architecture

Initially, the only service provided by Intel's Internet connection was electronic mail for Intel employees. Today, Intel's Internet connections provide services for not only Intel employees, but also for Intel customers and business associates. These services allow employee access to the World Wide Web, FTP, Usenet news, and streaming audio and video. Intel customers can access Intel's Web site and can download product specifications and software. Business associates can place orders for products on our e-Commerce servers. Understandably,

the complexity of the infrastructure to support these additional services has grown.

A single mail relay server was Intel's first general purpose presence on the Internet. Later, to provide additional services such as FTP and telnet to remote hosts, several more computers were granted access to the Internet connection. Users who logged into these servers could freely access all the Internet had to offer. But even before the explosive growth of the Internet brought about by the World Wide Web, this service model did not scale well. Providing an account for Internet access to every employee is clearly not scalable.

To deal with this problem and to avoid incidents where a user might mistakenly or maliciously subvert the security of an Internet access server, the user accounts were removed and replaced with several special software agents called proxies. Each proxy provides access to a specific type of Internet service. Intel employees can access the Internet by employing these proxies. Only specialized servers within DMZs can talk directly to the Internet. This improves security and it also minimizes traffic on Intel's internal network. DMZ servers such as Web servers and DNS servers that provide services to people on the Internet talk directly to the Internet without hitting Intel's internal networks. For certain services such as e-mail and DNS, we spread servers over multiple sites and gateways. This spreads out the load and makes the services available in case entire Internet gateways go down.

Maintaining Multiple Distributed Internet Gateways

Distributed Internet gateways make their management a key challenge. Having Internet gateways located throughout the world, each with its own complex configurations, posed a significant maintenance challenge to the small staff responsible for managing Intel's Internet gateways. Configuring each system by hand one at a time would be time consuming and prone to error. Because break-ins are often carried out by crackers who exploit misconfigured devices, system and network administrators need to make sure that configurations are consistent and correct. Responding to changes in the environment would be increasingly difficult if we had to manage each system individually. We needed to create a system to help us manage all our firewall devices. Therefore, we designed, built, and currently use a system and methodologies that manage all of our router access control lists and all of our bastion host configurations. Our techniques, discussed in the following sections, greatly reduce the likelihood of a misconfiguration.

Copy Exactly!

The first key method we use was taken from Intel's manufacturing sector. "Copy Exactly!" (CE!) [2] means that each gateway is copied exactly from the previous one as much as absolutely possible. We specifically wanted to avoid engineering a solution for each gateway. In manufacturing, new factories ramp faster and more productively if a standard design is copied exactly rather than reengineered. CE! has proven to be the only way to ensure that our configurations are implemented consistently across systems. With regard to the hardware, each firewall is composed of identical sets of routers. Each bastion host is based on a small set of approved platforms. Our maintenance system ensures that the software configurations are all CE!

CE! posed an interesting challenge for writing software and configurations that are distributed into our architecture. The code on each of the Internet gateways has to be identical; yet, that code would have to operate on systems particular to each gateway. To deal with this problem, we developed a program called *which_dmz* that tells the code what gateway it is in and can return data specific to that gateway. If a code needs specific gateway information it can call *which_dmz*. This way, we can distribute identical code and scripts to all gateways and have the code use gateway-specific information.

Modelling the Distributed Gateways as A Single System

We model all of our Internet gateways as a single large system. In our model, all the configurations and software of the gateway components are considered source code. This code is compiled into an "executable" form that is then loaded into the system and run. Certain configuration files have dependencies: if particular configuration files change, then other files must change also. Just as programmers use *Make* [9] to manage compilation and installation of source code, we use *Make* to drive and manage the configuration of our connectivity architecture [4].

A single master set of host and network configurations is kept in a central repository. From this set, the configuration files for all the devices, whether hosts or routers, can be derived. Once a new configuration has been created, it can be "beta tested" by pushing out the changes to a single device. Once the configuration is tested, the remaining servers can be updated. As an added benefit, we can compare the configuration of each device in the firewall against the central copy to see if it has been tampered with (which may indicate a firewall device has been broken into). The central computer makes use of encryption and authentication to gain access to each device during maintenance. This prevents a

hacker from gaining access by impersonating the central computer or by eavesdropping on passwords that may be used in an update process.

The architecture of the *Make*-driven update process is shown in Figure 4.

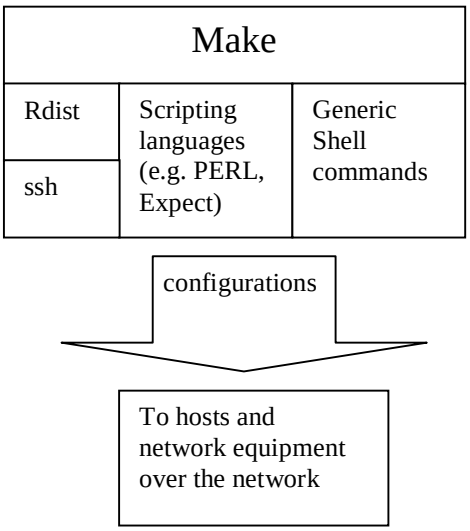


Figure 4: Architecture of the *Make*-driven update process

Make controls all of the operations of configuration maintenance. It can activate *rdist*, a file distribution program, run programs written in various scripting languages, or execute any generic shell command to build and distribute the proper configurations and software, which are sent out over the network to network equipment and server hosts. *rdist* runs over SSH, the secure shell protocol. We discuss how *Make* is used to configure network equipment and systems in the next two sections.

Maintaining Router Configurations

When Intel's Internet connectivity was just a single ISP connection through a single set of routers, router maintenance was simple. Once we began to have multiple gateways and once we also began to expect the traffic to fail over from one Internet gateway to another, we realized we would have to synchronize the router access lists that control what kind of packets can pass through our gateways. A change in the permissions at one gateway needed to propagate to other gateways. Since we stored access list entries in files, we needed to have a change in one file trigger changes in other configuration files. Since *Make* is excellent at managing the relationships and dependencies between files, it seemed to be a natural choice to manage the collection of router access lists.

Any gateway has to fail over to any other gateway. In order to do this, we use variables for the access control list (ACL) numbers that identify what access list an individual access list entry is part of. Macro definitions are used to define these variables to substitute the correct ACL number for the appropriate router. In this fashion, an access list entry on one router could appear in a different access list on another in such a way that traffic that regularly went out of one router could pass through another. Installation of access lists is driven through a router access list Makefile. *Make* is used to call Expect scripts that automatically load new versions of access lists.

Managing System Configurations

Make is used to manage the system configurations of all the servers, wherever they may be in the world. Since *Make* is so versatile, we use it to drive the remote distribution program *rdist*. In turn, *rdist* has been configured to run over Secure Shell to prevent eavesdropping by potential crackers and to prevent attacks where intruders masquerade as friendly systems in order to penetrate defenses.

Our key system configurations are organized into separate source trees, each with its own Makefile, the configuration for make, and its own Distfile, the configuration for *rdist*. There are separate source trees, containing all the executables and configurations necessary for a given server subsystem like the operating system, our IMCS software, sendmail, and other significant collections of software.

Once new software is installed into a tree, installing it on all of the servers is as easy as typing "make install." Typing "make check" shows what's waiting to be installed. Every morning we receive a report from a script that does a "make check" in each tree. This report is an extremely useful tool for keeping track of changes that are being pilot tested on one machine and will need to be installed on the rest of the machines. "make check" reports on what files have changed and what needs to be done in order to synchronize a given system with the definitive build.

By using *Make* we can easily insert hooks to massage or generate certain data files before they are distributed to the servers. For example, the sendmail tree's Makefile can automatically generate a sendmail.cw file that contains a list of all of the DNS domain names from the DNS tree.

By using *rdist*, we can easily customize which files get installed where and on which servers. We use *rdist* to automatically run commands when certain files are installed or updated. For example, when the mail alias

file on a server is updated, a program called "newaliases" needs to be run to rebuild the file into an indexed file format. *rdist* can automatically run the "newaliases" command once an alias file is updated.

Our use of *Make* and *rdist* for server management closely parallels the "Copy Exactly!" methodology discussed earlier.

MEASURING THE QUALITY OF INTERNET ACCESS

With such a significant investment in Internet connectivity, we needed to be able to measure its quality. We therefore developed the Internet Measurement and Control System (IMCS) [3]. This system measures key indices of Internet performance such as packet loss, packet delay, Web page download rate and Web page error retrieval rates, and raw traffic volumes. When these key indices exceed predefined limits, the Network Operation Center (NOC) personnel are alerted and start working through a script to debug the potential problem. We modelled this system on the statistical process control systems used in Intel manufacturing plants, where when certain metrics are out of control, action is taken.

Early in the design of our measurement systems for the Intel firewalls, we decided that the most useful architecture was a distributed one that allowed measurement systems (such as IMCS and public domain systems such as MRTG [10]) to collect their data on a system in the DMZ and publish the results as HTML pages on Web servers on the measurement system. By using Perl-based CGI scripts on the measurement systems, it was possible to drill down to individual measurement values using GET-mode URLs. The fact that the measurements were available on the Web allowed us to make some simple Perl automation scripts that would routinely fetch current measurement thumbnail graphs from all our measurement systems and display them on a browser anywhere inside of Intel. The side-by-side comparison of key metrics across firewall complexes allowed quick isolation of common faults to individual firewalls, or conversely, allowed us to see Internet-wide events in near real-time.

The common tools (ping, traceroute, whois, etc.) and controlled access to the tools' current state of interfaces on firewall routers are made available by a combination of JavaScript-enabled HTML pages and CGI scripts written in Perl and Expect. This allows us to allow NOC personnel access to tools without giving them direct access to network equipment, thus avoiding potential security exposures.

INTERNET USAGE POLICIES

We felt that we could not offer Internet access unless some guidelines covering behavior on the Internet were in place. We examined many Internet guides and wrote up a set of guidelines, which became the Intel Internet guidelines from 1993 to 1996 [11].

In 1996, Intel decided they needed one comprehensive guideline to cover employees' use of electronic mail, computers in general, and the Internet. Several of the Internet connectivity staff, as well as representatives from marketing and legal drafted the current set of guidelines in use today.

RESULTS

Our experiences with the connectivity architecture has generally been positive. In this section, we discuss how well the architecture has scaled, describe some spinoffs of the technology and the policies employed to maintain the architecture, and then outline some of the key challenges we have faced.

Scaling to Meet Demand

Our gateway architecture is quite scalable. From a service standpoint, it provides Internet services for more than 60,000 employees. The architecture is robust enough to support $1 billion per month in electronic commerce. We have scaled the components of the architecture in almost every possible direction. To deal with increasing use of services in an Internet gateway, we have increased the number of servers. To deal with different services in a firewall complex, we have increased the number of DMZs and created new firewall complexes. To deal with additional performance requirements, we added additional ISPs and additional bandwidth. To deal with acquisitions and new users, we even added additional Internet gateways.

Despite this growth just mentioned, our architecture is maintained by a small staff. At the present moment, a staff of five maintains the configurations of six Internet gateways and ten firewall complexes, which includes more than 30 routers and about 30 servers around the world. We are able to maintain thousands of lines of router access list entries. Copy Exactly!, modelling Internet connectivity at Intel as a single system, and maintaining configurations using *Make* has made this possible. The availability of the design makes the architecture functional despite the failures of individual components.

Connectivity Architecture, Tools, and Policy

The architecture, tools, and policies developed for Intel's Internet connectivity have found other uses. The

connectivity architecture for Intel's use of the Internet has been adapted into the first two data centers of Intel's new Web hosting business, Intel® Online Services (IOS). Many of the tools originating from within Intel's IT have also been implemented within IOS, including the Make update methodology and IMCS. Our experience with measuring Internet performance lead us to contribute to the Cross Industry Working Team (XIWT) white paper on measuring Internet performance [6]. XIWT is a consortium of a diverse group of industries working to improve information infrastructure.

The Internet usage policy that we described in [11] is one of the more interesting spinoffs of our work. The paper became widely distributed in a number of collections of Internet usage and security policies. At about the same time that the paper was published, the User Services Area of the Internet Engineering Task Force began looking for people and material to create a set of Internet usage guidelines to publish as an informational RFC. The work we did on Internet usage policy became part of RFC 1855 [5], Netiquette Guidelines, published in 1995.

KEY CHALLENGES

We experienced a number of key challenges as we implemented our Internet connectivity architecture. Our make-driven maintenance process, while powerful, requires tremendous discipline to use. While software and configuration changes can be rolled out uniformly across all of the Internet gateways, it is just as easy to roll out a bad configuration change as a good one. It also takes discipline to maintain Copy Exactly! and distribute changes from the central repository. Occasionally configuration changes are made on individual servers and routers and then lost when other changes are distributed from the central configuration repository.

While our architecture is designed to withstand the failure of a single physical component, the failure of a software component is much harder to deal with. Because of CE!, we deploy the same version of software and configuration files to each type of network equipment and server. If there is a failure in that software, then all the equipment with that version fails. We have seen this happen most often when there are scaling issues—where software or configurations cannot handle the demands made on them. The way to avoid this problem is to test new software or configurations under significant load. Unfortunately, not all load conditions can be simulated in tests.

Security continues to be a challenge. New threats and vulnerabilities crop up regularly and must be dealt with. Among the more difficult challenges are applications that users want to utilize across the Internet that are not well designed for firewalled environments. We have had to refuse a number of requests for services that cannot be easily proxied or do significant engineering to make those services secure.

Scaling to meet demand for Internet services continues to be a challenge. As the Internet becomes used more and more, servers and network equipment can become heavily taxed. Dealing with large numbers of firewall rule requests is also very taxing. Another demanding aspect is that changes or major projects often happen on short notice (also known as Internet time).

Our IMCS system relies heavily on active measurements, i.e., measurements that generate nonvalue-added traffic on the Internet and our infrastructure in order to obtain performance data. That nonvalue-added traffic causes a number of problems. We have received some complaints from the targets of our measurements about the measurements that we do. In one case, our measurements added significantly to a target site's bandwidth costs, prompting them to complain vigorously. We would like to measure everything we can. However, active measurements add traffic and thus costs to our infrastructure and to the infrastructure of the targets that we measure.

International Internet gateways are another challenge. Bandwidth costs are often much greater internationally than in the US, so it is not always clear that installation of an Internet gateway will reduce costs and provide better end-user performance. As noted by Cukier [7], the performance going from a country to the United States can often be better and cheaper than performance within a region such as Europe or Asia. Since more bandwidth is being put between the US and Asia and the US and Europe than is being installed locally within those regions, this trend is likely to continue, making it difficult to justify the cost of international Internet gateways.

FUTURE PLANS AND EXPECTED TRENDS

In general, we see ever increasing use of the Internet for more and more functions. Two particular trends we see are the use of Virtual Private Networking and streaming media.

Intel currently utilizes high-cost, high-maintenance legacy technology to provide connectivity between our sites, to employees when they are away from the office, and to our business associates. A new connectivity model, called Virtual Private Networking (VPN), can be utilized to lower costs and to reduce the time required to establish a connection to a new site or business associate.

With VPN, we are able to utilize the Internet bandwidth to enable employees who are out of the office to connect back in to Intel and have access to our internal resources

using their own ISPs. ISPs typically offer much wider and cheaper coverage for dial-up connectivity than Intel. Using the triple DES encryption technology provided by protocols such as IPSEC, we can ensure that Intel's intellectual property is safe as it traverses the Internet.

The majority of Intel's Wide Area Network is composed of costly privately leased lines. When we need to connect to a new site, we are often delayed due to the time required to order and install these circuits. Using VPN technologies, we will also be able to leverage the Internet connectivity that is already installed at our major sites to create an encrypted, secure link between them through the Internet. We will be able to reduce the setup time of two to three months to one day or in some cases the connection can be established within hours of receiving the request.

Connecting business associates presents a slightly different set of requirements because we only want to share a subset of information. By building an environment that limits accessibility to specific resources, we can utilize VPN technology to reduce the cost and time required to connect and implement the connection in a similar fashion to WAN replacement technology mentioned above.

We also see increasing use of streaming media such as video and audio. These type of media will substantially increase bandwidth requirements. Because these media are jitter sensitive, service-level agreements between Intel and its ISP become more and more critical.

We see passive measurements becoming much more important in the future. Mining data sources such as Web server logs and router flow statistics can yield valuable performance data without adding probe traffic. As the number of gateways increases, active measurements become more and more of a burden on the measured sites.

CONCLUSION

As the Internet evolved from a network linking a community of researchers and engineers to the mainstream medium that it is today, Intel's Internet connectivity evolved from a modem used for e-mail to the highly available, multiple service distributed system that it is today. Our design and implementation of an Internet connectivity architecture has enabled Intel to handle $1 Billion in e-Commerce per month and has spun off technology, methods, and policies used in other Intel businesses and across the Internet. Anticipating ever increasing use of the Internet, our architecture is ready to deal with new challenges.

ACKNOWLEDGMENTS

We acknowledge Suzanne Johnson, whose vision helped land our first Internet dial-up connection those many years ago. We also acknowledge and thank the many people who have helped so much with deploying and maintaining the architecture, including the CPS team, ITSP, the OSC, the WAN team, and countless folks in site IT organizations.

REFERENCES

[1] Johnson, Suzanne M., "*Internet Affects the Corporation: Experiences from Eight Years of Connectivity,"* INET 95, Honolulu, HI, June 1995.

[2] McDonald, Chris J., "*The Evolution of Intel's Copy EXACTLY! Technology Transfer Method*," Intel Technology Journal, http://developer.intel.com/technology/itj/q41998/articl Oace, Tod, Sedayao, Jeff, Wong, Clinton, *"Don't Just* es/art_2.htm.

[3] Bickerstaff, Cindy, True, Ken, Smothers, Charles, *Talk About The Weather - Manage it! A System for Measuring, Monitoring, and Managing Internet Performance and Connectivity*," 1st Usenix Conference on Network Administration, Santa Clara, CA, April 1999.

[4] Hambridge, Sally L., Oace, Tod, Sedayao, Jeff, and Smothers, Charles,"*Just Type Make! Managing Firewall Using Make and Other Publicly Available Utilities," 1st* Usenix Conference on Network Administration, Santa Clara, CA, April 1999.

[5] Hambridge, Sally. "Netiquette Guidelines," RFC 1855, October, 1995.

[6] Cross Industries Working Team White Paper, "Customer View of Internet Service Performance: Method, Methodology, and Metrics," "Customer View of Internet Service Performance: Measurement Methodology and Metrics," September 1998.

[7] Cukier, Kenneth Neil, "*Bandwidth Colonialism? The Implications of Internet Infrastructure on International E-Commerce*," INET 99, San Jose, CA, June 1999.

[8] Chapman, Brent D. and Zwicky, Elizabeth D., *Building Internet Firewalls*, O'Reilly & Associates, Inc., Sebastopol, CA, pp. 58, 66.

[9] Oram, Andrew and Talbot, Steve, *Managing Projects with Make*, O'Reilly and Associates, Inc., Sebastopol, CA 1991.

[10] Oetiker, Tobias. http://www.mrtg.org/.

[11] Hambridge, Sally and Sedayao, Jeff, "*Horses and Barn Doors: Evolution of Corporate Guidelines for Internet Usage*," Proceedings of the Seventh Systems Administration Conference, LISA '93, Monterey, CA, November 1993.

AUTHORS' BIOGRAPHIES

Cindy Bickerstaff is a 20-year Intel veteran with the first 15 in silicon process development especially in metrology and process control. She received "The Five Year (or so) SPC Marathon Award" in 1988 for driving SPC implementation in California Technology Development. She received an Intel Achievement Award in 1989 for "formulating, developing, and implementing the first statistically based process control strategy for lithography at Intel". After her 10th silicon technology development cycle she decided to try something new and moved into the Internet group in 1995. Since then, Cindy has been mapping many of the measurement and control methods she worked on in silicon processing into the Internet knowledge domain. Her email address is cindy.bickerstaff@intel.com .

Sally Hambridge began working on the Internet in 1980 when she joined the Information Sciences Institute in Marina del Rey, the home of the IANA and the RFC-Editor. She joined Intel in 1984 after a brief sojourn at Atari. She saw the advent of Internet connectivity at Intel, and has worked in the Internet group since 1992. Since 1996 she has been responsible for router access control lists and for routing in the Internet connections. She joined Intel Online® Services in August 1999. Her email address is sally.l.hambridge@intel.com .

Lynne Marchi received her B.S. degree in computer science from California State University, Sacramento. She began work at Intel in 1992 as a co-op and then joined the Corporate Information Security group in 1993. In 1996, she joined Internet Connectivity Engineering where she has focused on the secure implementation of new firewalls. Her email address is lynne_c_marchi@intel.com

Tod Oace is a UNIX and networking systems engineer. Over the past 20 years he has worked with a wide range of computer platforms from micro to mainframe, and is proficient in several programming languages and networking protocols. He pioneered the use of BSD UNIX for several enterprise applications since joining Intel in 1991. He received division recognition awards for the redesign and support of an in-house created Intranet SMTP/cc:Mail gateway, and for the co-creation of an Intranet LED readerboard display control system. He also designed the first large-scale network for the www.intel.com server complex. Tod is now working in the Firewall Engineering group of Intel Online Services. His email address is tod.r.oace@intel.com .

Stacy Purcell graduated from the Georgia Institute of Technology with a B.S. degree in computer science. He joined IT in 1994 where he has worked as a Front Line Manager, a Network Engineer specializing in the design, maintenance, and troubleshooting of LANs for the Folsom site and in new acquisitions. He currently works in the ICE team focusing on Intel's six Internet firewalls and plans to expand that number to 24 by the end of the year. His emaild address is stacy.p.purcell@intel.com .

Jeff Sedayao is a network engineer in Intel Online Services. Between 1987 and 1999, he architected and ran Intel's Internet connectivity. His primary interests are in Internet performance, security, and policy implementation. He also serves as Intel's representative to the Cross Industries Working Team's Internet Performance Team and has participated in the IETF's IP Performance Metrics Workgroup. His email address is jeff.sedayao@intel.com .

Charles Smothers is a Senior Systems Programmer for Information Technology. He joined Intel in 1982, when he wrote 8085 assembly code for testing the 27128 EPROM. With the Memory Components Division, Low Yield Analysis team he automated several of the tasks used in the visual inspection and defect categorization process. In 1990, he joined the MCD Design Engineering group as a UNIX systems administrator. Today, he specializes in Internet Proxy servers. His email address is charles.smothers@intel.com .

Ken True is Manager of Internet Firewall Engineering for Intel Online Services, Inc. He joined Intel in 1981 and has provided technical management and individual contributions in the areas of Mainframe Systems Programming, WAN Network Engineering, PC Technical Services, LAN Software Distribution (OfficeLAN), Intranet/Internet Engineering, and Internet Connectivity Engineering. He and his organization are responsible for the architecture, engineering, and security of the IOS firewall systems worldwide. His email address is ken.true@intel.com .

Managing Enhanced Network Services: A Pragmatic View of Policy-Based Management

John Vicente, Information Technology, Intel Corporation
Harold Cartmill, Information Technology, Intel Corporation
Glen Maxson, Information Technology, Intel Corporation
Shelby Siegel, Information Technology, Intel Corporation
Russ Fenger, Intel Architecture Labs, Intel Corporation

Index words: policy, policy-based management, PBM, QoS

ABSTRACT

The convergence of public and private networks and the rise of the Internet as a global medium for information exchange and economics are prompting corporations to augment existing business computing models. New information technology initiatives for collaboration and business exchange are essential to gain competitive advantage. These initiatives are being realized through emerging applications (e.g., e-Commerce, groupware applications, multimedia) over converging private and public boundaries. Enterprise strategies are requiring timely evolution of network infrastructure and management to support delivery and management of end-user and network services. Within this context, quality of service, security, productivity, and infrastructure efficiency are critical to achieving bottom-line business results. In this paper, we take a closer look at policy-based management as an enabling technology and paradigm shift for Intel's Information Technology organization. We explore the dimensions of policy-based management, and we provide a pragmatic review of the technology, discussing the deployment challenges, roadmap considerations, and practical usage scenarios.

INTRODUCTION

Current trends in corporate and Internet networks are shifting from best-effort, vertical network architecture towards a more intelligent, end-to-end, service-aware network paradigm. As evidenced over recent years, the need for enhanced network services such as virtual private networks (VPN), quality of service (QoS), security, collaboration, and directory technologies demonstrates that customers are demanding more from the core infrastructure for enabling productivity, flexibility, service differentiation, isolation, privacy, and manageability. Moreover, critical network resources must be aligned with business objectives where networks are i) more content or application-aware; ii) provide dynamic features for service creation; iii) observe and enforce network-wide policies; and finally, iv) enable control from the network provider to the administrator to the end-user. The migration to a richer network infrastructure allows corporations to be more agile and optimize infrastructure costs, while meeting the diverse requirements of emerging application demands. The following requirements are driving innovations in current network infrastructure technology.

Mission Criticality

IT organizations recognize that the availability and reliability of network infrastructure are essential to critical applications and services that rely on them. These same applications can compete for network resources (e.g., bandwidth) with other less critical and diverse applications for successful transport delivery and performance. Core network capabilities are required to ensure delivery priority, security, access control, and fair performance allocation. However, certain users (e.g., company presidents) or groups in an organization may have a more critical need for access to resources, and thus, that person or group may get priority or have a different authorization from the rest of the organization. Mission criticality is raised as organizations move towards extranets or public services for corporate business computing or service transport.

Network Architectural Agility

Customer demands and applications are growing rapidly while networks continue to be gridlocked by standards and proprietary implementations. In general, enterprise networks have been implemented with multiple vendor device solutions, while heterogeneity of bandwidth resources and legacy device inequalities are evident over geographically dispersed regions. While such infrastructure complexity exists and diverse application-network demands continue, network architectures and management frameworks must still support more rapid evolution, enabling faster service creation and deployment while maintaining legacy integration.

Service Flexibility

The application and network infrastructure need to be integrated to meet the ever-changing needs of current computing and distributed application demands. Providing dynamic network reconfiguration, timely access control, or dynamic, class-based bandwidth allocations can provide greater flexibility to the end-user or network administrator. An increase in network features allows administrators to manage service levels more effectively, while allowing application developers more control features for end-user services.

Efficiency

The increased complexity of managing network systems along with the need for service-level management requires current network management solutions to be more sophisticated. While corporations continue to grow, optimal planning and management of the distributed infrastructure are essential. Better tools for Just-in-Time (JIT) bandwidth and service provisioning, effective use of resources, and automation of management tasks can reduce the total cost of ownership while improving service-level management.

In the first section of this paper, we present the background on policy-based management including our view of the requirements for this technology. We follow this with a discussion on the evolving IT business and operational models. Next, we discuss the technology implications and challenges that IT organizations must address to realize policy-based management. Following this, we present our current progress within Intel IT, providing a transitional discussion under the e-Business model. We close with a summary and a brief look at how we can move forward with this technology.

BACKGROUND ON POLICY-BASED MANAGEMENT

As defined in [1], policy-based management is "the combination of rules and services where rules define the criteria for resource access and usage." Alternatively in [2], the authors define it as a "unified regulation of access to network resources and services based on administrative criteria." We view policy-based management as a viable technology to provide greater control and management of underlying networks via the creation and distribution of high-level policies (business rules), integrated with the enabling mechanisms of the network infrastructure. By way of automated and rapid configuration and the integration of business policies with the network infrastructure, new opportunities for managing both infrastructure and network services are introduced. To support these new initiatives, we define the following general requirements for policy-based management technology:

Service differentiation. This is the ability to control or manage the quality of the service or service delivery mechanisms in order to meet some predefined network-based performance delivery/metrics. This may extend to enabling Service Level Agreement (SLA) management for service-level validation.

Network provisioning and bandwidth management. These provide proactive bandwidth management by facilitating control and allocation of bandwidth through device configuration management: that is, facilitating manual, multi-device network configuration and performing admission control or traffic segmentation.

Integration with network management systems and legacy devices. Policy-based management must be integrated with current paradigms for managing IT organizational structures. These include existing operational models (e.g., centralized control or change management), security requirements, and business computing models. The requirement to support or address legacy systems and device limitations is also mandatory.

Scalability. The PBM technology infrastructure should architecturally scale to Intel's enterprise environment and private/public models for e-Business computing. These include the policy server environment to business computing hierarchy, the directory and database infrastructure, and scalable policy overhead in terms of administration and protocol communications.

Industry standardization. This means that policy-based management must conform to industry standards and use best practices to support network device and policy management interoperability. There must be standards for such things as policy terminology [1] and protocols (e.g., COPS [3] / LDAP[4]). Moreover, an open framework for policy management schema [5] and directory integration must exist.

- *Security*. The policy-based management tools should facilitate resource access control and authorization, and they should provide integration support for authentication and accounting.

Policy-Based Management Framework

Functional Overview

The components of a policy-based management system (PBM) include the policy console, policy server, policy database, and policy clients, all of which are shown in Figure 1 below. The policy console provides administrative and operational access to the PBM system.

The Policy Decision Point (PDP), or policy server, embodies the decision-making functionality of policy-based management. One or more such policy servers exist in a control domain, with each server configured to support policy management for some defined group of policy clients or Policy Enforcement Points (PEP) in the domain. The policy server provides each policy client with policy information; the policy client in turn carries out (enforces) the policies to the best of its abilities. The policy server's inner structure is shown in Figure 1 below.

Policy Server Structure

The policy client communication component handles all exchanges between policy clients and the policy server. The PBM client-server communications protocol uses the (proposed) IETF standard COPS protocol.

The message processing component is responsible for policy protocol message decomposition and composition, and for interpretation of wire-format objects for use in the policy server.

The core-processing component embodies the logic needed to support the policy server's policy decision making—rule processing and housekeeping. This component also logs all relevant usage by policy clients in support of accounting and billing applications. These logs may also be used for further tuning of resource control policies.

The policy console communication component gives support for communicating with the policy console. This allows multiple PDPs to be maintained from one policy console. It also allows multiple consoles to access the same PDP.

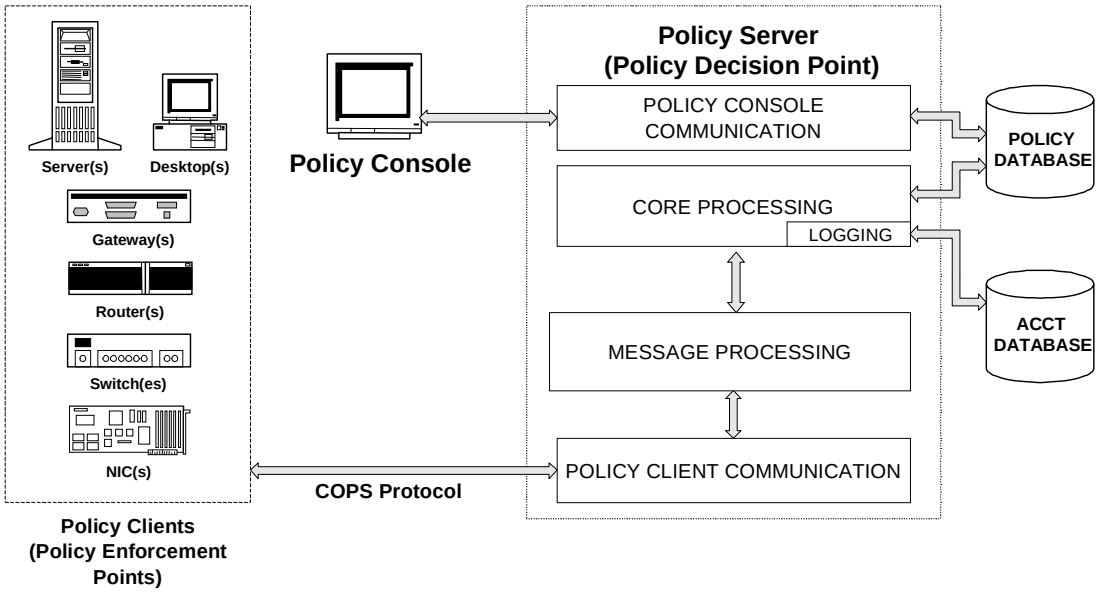


Figure 1: Policy server structure

Implementation Alternatives

There are alternatives to using a policy-based network management server as described above. For example, one can manually configure policy clients to perform policy actions on packet flows, such as Type of Services (TOS) bit settings in the packet header that tag packets for priority Quality of Service (QoS) or client-initiated RSVP (ReSerVation resource Protocol) [6] sessions. However, one of the key functions of the PBM server is the centralized management of network resources and the allocation of resources to clients based on organizational rules (the organization owning and managing the network resources.) This ensures that only authorized clients have appropriate access to and

use of network resources. This function is extremely difficult to manage solely at the client.

There are other approaches that use directories and directory protocols, such as Lightweight Directory Access Protocol (LDAP) to set up and administer policy. These approaches work well when configuring network devices and setting static rules. However, when dynamic control of network resources is required, directories are ill-equipped to deal with the complex decision-making process, based on current resources in use, priorities of requests, and administration of usage. Also, the LDAP protocol does not currently provide a robust level of information or a dynamic framework to manage network resources in real-time.

Intel Architecture Labs (IAL) is helping drive the networking industry to fully use IETF's Common Open Policy Services by making available the COPS toolkit, which enables clients to talk to policy servers. IAL has developed and is deploying a COPS Local Policy Module that enables Windows 2000* clients to talk directly to policy servers.

EVOLVING IT BUSINESS MODELS

Current trends in business computing are blurring the boundaries between the Internet and Intranet, while application demands are becoming more content rich and diverse in quality of service requirements. Network vendors are building infrastructure components (e.g., VPN) with more enabling network features (e.g., QoS) and management systems to support these new users or applications. Nevertheless, it is unclear how IT organizations should structure or restructure their business processes to map business units, application types, and transactional priorities to critical infrastructure resources. If this new paradigm is to be realized, policy-based management must emphasize and speed up the development of new or the revision of existing business processes. For IT organizations to move towards a more service-oriented and evolutionary computing model, processes will have to be developed to manage the integration of business objectives with network and distributed infrastructure through policy. Figure 2 depicts this integration with a proposed hierarchy (see also [2]) with which policy is introduced, translated, and propagated within the network infrastructure. At the highest level, business rules should dictate global (i.e., domain-specific) directives on the effective use and priority of resources to support the business objectives. These objectives are translated into network-wide policies within the administrative domain on the necessary bandwidth, communication resources, and topology requirements. Network-wide policy rules are translated to node policy rules that are specific to the required behavior of the node to manage its local resources. Finally, local policy rules are applied and enforced through one or more specific command instructions on device-level functions (e.g., scheduling and queuing parameterization).

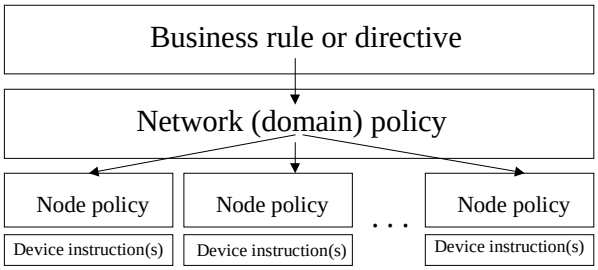


Figure 2: Integrating business rules with network policy

We propose the following methodology from which policy-based systems can be implemented as an aspect of the management process within a corporate enterprise environment or managed network domain:

1. Create a network baseline and track network usage against key applications, network services, and business unit users or usergroups.

2. Establish network domains, policy groups, group identities, and hierarchies that map to core business activities.

3. Establish organizational directives and create corresponding policy rules and service-level requirements over various applications supporting policy groups/users on the allocation or priority use of critical infrastructure resources.

4. Administer and deploy policies across the network infrastructure, typically on a domain or inter-domain basis.

5. Audit and validate network policies against service requirements.

6. Refine business directives and network policies based on policy-enforced behaviors.

7. Repeat steps above.

Operational Models—Shifting to Policy

The operational management models of today's IT environment are based on traditional models of systems and network management. These evolved from a

* Other brands and names are the property of their respective owners.

central, mainframe-orientation to a client-server and distributed systems management paradigm. A current challenge, both for IT management and solution providers, is managing both the end-to-end service model and the vertically disjointed infrastructure elements. With recent developments in network QoS capabilities and policy-based management, this challenge may be met. Operationally, however, service-level management and change management processes will need to evolve or adjust to support a policy framework. Furthermore, training of operational personnel on business-oriented policy management, network QoS/CoS, and the supporting management tools will be a requirement. These are discussed in more depth in the next section.

Change Management

A critical component of the IT operational environment is understanding infrastructure changes and managing operational schedules. For example, avoid changing a network the same day software is updated for a large population of servers or clients. Generally, the change process tends to have a high overhead in terms of administration and personnel, which is complex to manage and difficult to administer over a large enterprise environment. As we move to a policy-managed environment, using existing IT change management processes will require greater interdependency and be more complex and costly. Thus, we anticipate an evolution in the administration or change management process.

The existing change management system is built on IT being the direct provider of all IT services. The current model lacks operational flexibility or agility. Policy-based management, on the other hand, will allow organizations to map the network to the business requirements, thus changing the role of the IT organization to that of a coordinator for service provisioning. This is an evolutionary step for traditional IT organizations as the change process will directly engage customers in the change processes. Furthermore, IT project managers are generally focused on managing the customer and the expectations for their business unit, application, or area of focus. This approach works but has unforeseen consequences for the enterprise network and end user when a customer's application is fully turned on. The PBM environment should motivate business groups' project planners to drive changes in the IT project office to consolidate planning activities. The evolutionary process will require project leaders to coordinate their activities to engage stakeholders and to ensure that their application/service is properly prioritized and business rules are communicated and tested. Engaged parties responsible for sustaining business processes must understand the implications of specific agreed policies on the business environment.

As a policy is implemented, the project manager and operations change bodies must tightly integrate business requirements and the policies deployed within the operational environment. This requires cross-training personnel on the integration of business and policy management so that they understand the effects or ramifications of change policies within the infrastructure. We envision a consolidation of existing processes and tools through the integration of change and problem management with PBM systems. By streamlining processes for change and problem management groups, we will enable data sharing and true interdependence. Thus, we believe the generation of Customer Resource Management databases and tools should speed up and personalize the change/problem management system to provide customers and stakeholders a view of the relevant data that supports their activities.

Service-Level Management

Service-Level Management agreements (SLAs) are contracts between the provider delivering a service and the recipient of the service. The SLA codifies the understanding between the parties to ensure delivery of services and value for payment. A network/application-oriented SLA places value on service delivery of the key components of *availability, delay, throughput, customer service, and affordability.* A service-level policy is a method for controlling and regulating service differentiation. Together, service differentiation and service-level policy form an integral part of the service SLA function that has become increasingly important as TCP/IP networks evolve.

The shift to PBM will require IT organizations to have the tools and motivation to manage to tighter service levels on service performance, security, reliability, and customized agreements. This will require a more dynamic and granular auditing and reporting function for the underlying services and the managed environment. Additionally, the reporting function must be designed to refer to the negotiated SLA and provide a meaningful report that supplies the customer with validation that services are being delivered as agreed. Moreover, the customers must have objective proof of reliability and transaction responsiveness, service availability, as well as be able to rely on the operations supporting their core business.

TECHNOLOGY IMPLICATIONS

Policy-based management technology can be applied both to areas with obvious Returns on Investment (ROIs), such as bandwidth management (e.g., savings on WAN circuit costs) and to areas where the benefits are harder to measure, such as productivity for certain users. Since it potentially can require a large capital outlay to implement (i.e., if a large amount of legacy equipment needs to be replaced), policy-based management initially will be implemented as point solutions targeted to specific conditions with definable ROIs. In this section, we discuss the key barriers IT organizations have to clear and the key technology areas that they must address in formalizing an ROI and roadmap strategy.

Network Technology

Multi-Vendor Interoperability

The infrastructure of an IT organization is made up of technology from a host of different vendors. The type of management policies desired will determine which equipment will be involved in receiving and enforcing policy information. For example, if a company wants to give high priority to certain SAP R/3 users, it may have to configure the network composed of mixed vendor equipment, the SAP servers, the SAP application itself, the users' client systems, and possibly some auxiliary servers (e.g., database and directory servers) in order to make this happen. As mentioned above, because of the nascent level of policy-based management technology and products and the embedded base, multi-vendor issues will persist for some time. This issue is being exacerbated by the move to the Internet and e-Commerce, where policy-based management promises some of its biggest rewards but the multi-vendor issues abound.

Quality of Service (QoS) Technology

Policy-based management technology solutions and tools presently focus primarily on the enabling mechanisms for delivery of QoS and resource (bandwidth) management. QoS policy can be defined [2] based on some criteria including the endpoints of communication, route or communication path, community of interest or usergroup, application types, network or traffic characteristics, or specific time period. There are primarily two methods to support end-user QoS: signaled or provisioned. Within the IP communication model, RSVP with Int-Serv (Integrated Services) [7] performs signaling to ensure QoS on a per flow basis by using dynamic resource reservations. RSVP leverages policy-based controls to support admission control and flow regulation. Alternatively, Differentiated Services [8] or DiffServ operates on a slower time-scale. It is essentially a provisioning model to reserve or establish service classes. The DiffServ model operates on a traffic aggregate basis where flows (one or more) are bundled together according to a set policy and treated based on a negotiated class of service. The administrative provisioning model is based on SLAs translated to Traffic Conditioning Agreements (TCA) and enforced through underlying mechanisms (e.g., classification, scheduling) within a router or IP device.

Multi-protocol Label Switching (MPLS) [9] and 802.1p [10] are both layer-2 protocols that can be used in isolation or paired with DiffServ or RSVP to deliver QoS on a flow QoS or provisioning basis. MPLS label switch routers (LSRs) are positioned (among other capabilities) to support the integration between ATM and IP by using Label Switching Path (LSP) protocols to map layer-2 ATM VCI/VPI identifiers with a layer-3 IP device to deliver (not exclusively) QoS within an ATM WAN core. Within the LAN environment, the 802.1p protocol enables priority or service classes within a LAN environment, where once again, mapping of layer-2 and layer-3 service classes or traffic precedence can be supported. The original scope of the 802.1p protocol was to support the integrated services model within the local LAN.

In current vendor solutions, early QoS/CoS capabilities to support these protocols/mechanisms exist, but there are many problems with their adoption. Int-Serv/RSVP has been plagued by the scalability issue where the claim is that maintaining per flow state within the Internet or a large enterprise network breaks the fundamental IP architectural model, which is based on end-system state maintenance and control. On the other hand, Diff-Serv has not reached standardization or industry maturity. This latter point is especially true when it comes to deployment, where the need to support inter-domain SLAs has not been fully hammered out. However, the concept of the Bandwidth Broker (BB) [11] has been proposed by the DiffServ community to support provisioning within intra-network domain boundaries and across inter-domain boundaries to deliver end-to-end quality of service. This work is still in the early stages and requires more investigation. Nevertheless, the motivation is there to establish standard building blocks towards enabling QoS within the Internet/IP model.

A broader analysis of some of the aforementioned services and protocols is presented in [12] including alternative QoS-supported models for Constraint Based Routing and traffic engineering. Alternative proposals leveraging the best features or motivations behind these protocols or services have also been suggested to deliver scalable, signaled, and provisioned QoS [12, 13, 14].

Within current IT environments, proprietary network device features that support service differentiation as well as legacy devices are variables in the ROI decision process. Support for legacy systems and proprietary devices is an essential aspect of our early QoS and policy investigation. This includes looking at traffic segmentation (e.g. VLAN switching) or access control (e.g., multicast filtering) to manage traffic propagation. Moreover, over-provisioning, especially within a LAN environment, is a practical alternative in the short-term. Nevertheless, with increasing bandwidth requirements, QoS-dependent applications, and the move towards the Internet, the industry QoS models described previously may clearly be the favorable long-term choices. To deliver or manage QoS, the consensus is that policy-based controls must be integrated with device mechanisms (proprietary or otherwise) to support the provisioning, admission control, and regulation of traffic.

Network Management

As the demands for distributed and global computing increase, and Internet-based electronic businesses continue to grow, network growth (e.g., traffic volume, traffic types) and complexity (e.g. devices types and counts, network events, interoperability) increase along with it. With demand for innovative services (e.g. desktop video collaboration, knowledge-based management) extending current user communication models, the requirements for these new services will continue to increase with a corresponding increase in network growth and complexity. While this growth is essential for business and the evolution of information technology, it places rigorous demands on our infrastructure. IT managers are faced with the challenge of managing the infrastructure for optimal service delivery while at the same time reducing operational costs (e.g., manpower, operational tools). Finally, traditional network management tools are device centric and require manual configuration, which leads to duplication and task redundancy.

Because of these constraints and limitations, policy-based management is justified. PBM will abstract physical and virtual elements of the network that facilitate the automation of traditional network management tasks across multiple objects of the network. The automation feature lessens the need for human administration, thus speeding the change management process. Furthermore, abstracting network devices or components raises the issue of implementation or proprietary differences across network devices. This allows (under certain conditions) management interoperability and reduces the complexity associated with managing implementation-specific devices over alternative interfaces. However, management capabilities available in current policy-based management solutions are not broad enough to deal with the overwhelming activities associated with network management. Nonetheless, the focus on bandwidth management and QoS is certainly the right choice in the move to this new paradigm for managing networks.

Managing Network Services

A complementary view of the enabling capabilities of policy-based management is the notion of regulating or controlling distributed and abstract network resource objects in concert to deliver or manage end-user network services. Such enhanced network services include video-based distance learning, voice over IP (VoIP), quality of service, multicast services, and security. The service orientation of network management is a major shift from the traditional approach of managing networks. This has been enabled through the introduction of middleware [15] services and the corresponding abstraction or raising of the lower-level physical communication infrastructure. The notion of policy is introduced here as "the ability to administer, manage, and control access to the network resources of network elements in order to provide a set of services to clients of the network" [15].

Security Services

Interoperability

Corporations need to protect their business assets from competitors. This traditionally has been done with physical barriers and firewalls for information technology assets. E-Commerce and the use of the Internet to provide IT services (such as Web and application hosting) is blurring the line between inside and out, requiring that security be implemented on a more information-specific level. The IPSec standard, for example, allows information to be encrypted between sender and receiver, thus satisfying a privacy requirement. However, the use of this encryption may also hide pertinent information from any policy-enforcing infrastructure in the path between the sender and the receiver. Thus, some implementations of security services could eliminate the possibility of establishing other policies. We recognize that both standards groups and vendors are working to reduce or eliminate this conflict.

Policy-Driven Security

While the current focus of network-based policy management tools is on quality of service, the motivation behind policy work extends beyond the functional area within the network transport. Clearly, privacy, complexity management, dynamics, resource access, and authorization, etc. will require similar facilities or capabilities so that growth and traffic propagation in the context of a security policy can be managed. This is especially true for the e-Business model of computing.

However, just as the network vendors have been moving toward policy-based management so has the security industry. Abstractions away from managing individual access controls on objects, management of heterogeneous environments, and training of personnel on multiple administrative products all support policy-based management initiatives in the security arena. We therefore see a parallel to the network PBM toolset in the applications and server space for policy-driven security. E-Business may provide synergy for merging future policy-based management tools in these traditionally separate areas.

There is an interesting perspective that arises primarily in the application/server space that divides the policy products in half. This comes in the positioning of management of policy versus execution (implementation) of policy. In the network PBM space, typically one expects that one vendor might provide the management tools, while another vendor provides the implementation, or run-time enforcement of policy. Given the underlying infrastructure of network management protocol convergence, it is reasonable to expect a network management toolset to have the capability to manage across a heterogeneous environment.

In the application and server PBM space, on the other hand, there is no standard for administration across a heterogeneous environment. As a result, administrative tools must either be built with interfaces to many proprietary targets, or they must provide a single centralized security server implementation, and expect the secured resources to query the security server in a common language. This split delineates two significantly different architectural solutions and bifurcates the product solution space. Intel's current architecture selection is the former one: a common policy-based management tool manages multiple heterogeneous targets, and the execution-time security is native to the proprietary target environments.

As we move forward with PBM solutions in the application and network space, it is likely we will merge and simplify this overlap of administrative functions in the security space. Administrators will associate policy with people, and that policy will apply to a variety of target objects. Whether these targets were classically "servers" or "network devices" will be irrelevant. Users and administrators will both have abstracted views of the physical computing environments, defined by business intent and not by topology or by a quirk of a vendor's security implementation.

Directory Services

Historically, directory services have been designed with specific requirements in mind. The relatively static

nature of the directory information update model was accepted, and instead the design optimized the high-speed access and replication characteristics. However, the advent of directory-enabled networking, Dynamic Host Control Protocol (DHCP) leases, data flow-rate, network state, and routing statistics have made it necessary to store and distribute low-latency, transient, and dynamic data. When a directory service is capable of supporting 'dynamic' data types, it becomes useful to policy-based network operations, supporting service provisioning or delivery of network state information to applications. An active association between a user or application context stored in the directory and the network can be discovered and used. New possibilities enabled by this technology follow:

- provide secure management of information from a variety of sources, including applications and network devices
- define, register, and provide publish/subscribe features for network events
- process network events for applications, devices, and users
- expose APIs to applications to take advantage of directory-based services
- maintain state information for devices, users, and applications to support policy-based management

The core directory service acts as the single point of administration for all resources, including users, files, peripheral devices, databases, Web access, and other objects. Extended functionality such as that provided by CNS/AD [16] provides access to vendor-specific network elements and services, and securely and efficiently propagated dynamic data. High-speed replication services securely propagate cached data among all directory servers, enabling them to manage dynamic network information such as IP lease or user password. Core directory functionality relates relatively static information about users and applications to dynamic information describing a given service request and the context in which the request has been issued. In summary, directory services will play a significant part in the establishment and the long-term success of policy-based management.

TECHNOLOGY EXPERIENCE AND USAGE

Within Intel IT, our current agenda is to qualify the suitability of policy-based management for the corporate enterprise environment as well as to define the transitional models to support e-Business. We believe this approach matches current industry and product roadmaps. Our initial technology evaluation objectives will primarily focus on QoS and bandwidth management to enable service management and resource management, although we hope to gain insight into other usage areas including security and network management in a wider context. To facilitate these objectives, as illustrated in Figure 3, we have developed a Quality of Service Network for Emerging Technologies ("QoSNET"). QoSNET is a production-level and policy-managed network environment to support proof-of-concept of QoS and policy-based management technologies. QoSNET will bring together emerging technologies including multimedia collaboration technologies and other next-generation applications (e.g., VoIP) to investigate intelligent bandwidth management capabilities to support scalable and manageable deployment of these emerging capabilities. Our current activities are focused on technology evaluation to support the integration of policy and network QoS features, while recording the operational procedures necessary to support policy-based management. One of our goals is to validate the technology capability as it merges abstract policy rules with device QoS capabilities, namely 802.1p and IP precedence/TOS control mechanisms within layer-2 (switches) and layer-3 (routers) enabled devices. Using the most recent developments in network equipment, we are also investigating key QoS capabilities including rate control and application-based recognition managed through policy invocations. Secondly, through hands-on experience using policy-based management tools, we will assess potential process or operational management improvements in bandwidth management and network and service management. Interoperability between alternative policy-based management tools and multi-vendor devices is critical to our success. We are also investigating opportunities to policy manage or control standard QoS-oriented technologies being defined through the IETF to support QoS/CoS, specifically, RSVP, Differentiated Services, and MPLS. Other planned areas for technology evaluation will include policy-based routing and evaluation of directory technologies to support the integration of system or application-based policy information. This latter item will include looking at scalability issues on policy within a large enterprise network (i.e., Intel's corporate network) or across administrative domains to support Internet-based service provisioning and policy management.

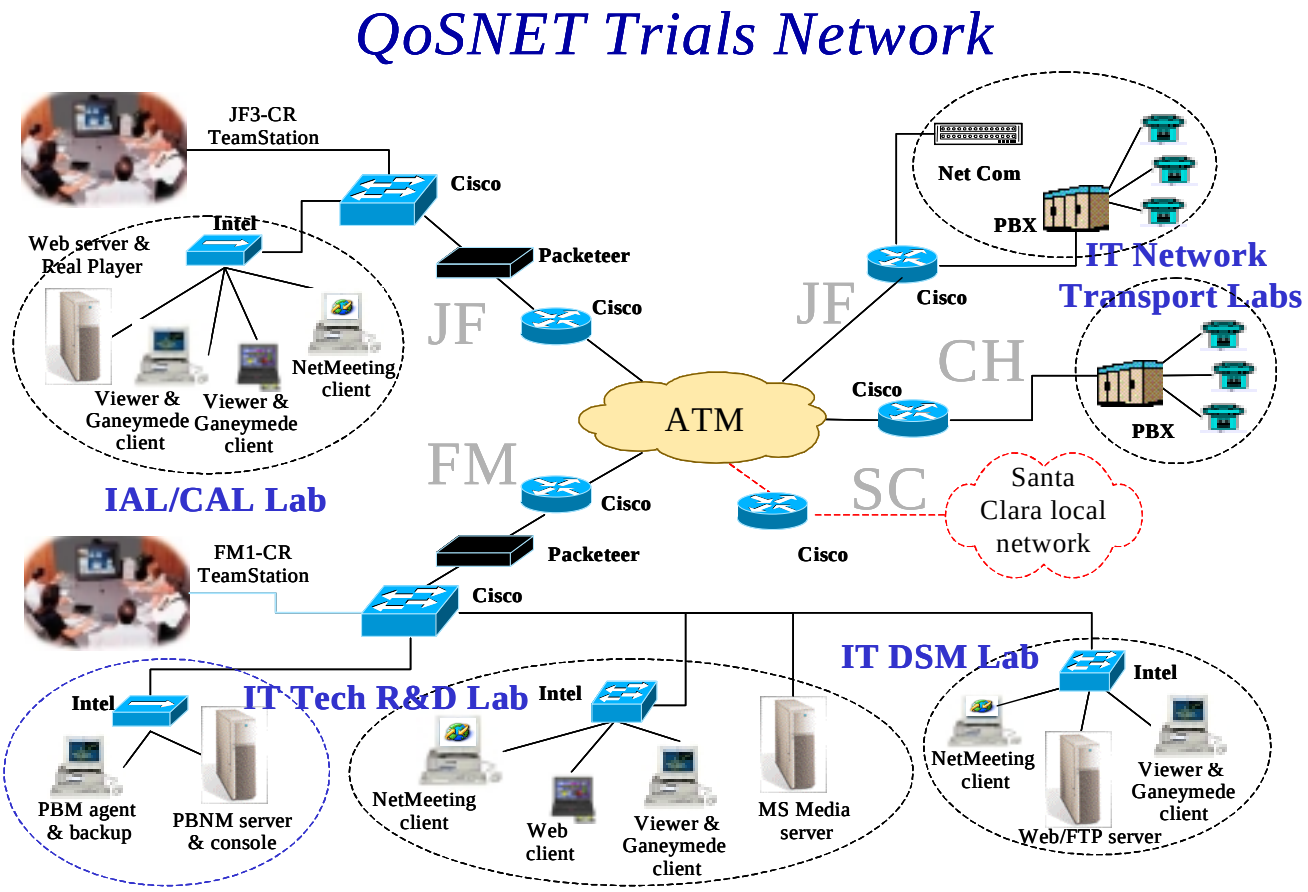


Figure 3: Information Technology QoSNET Trials Network

Usage Scenarios: e-Business

With the introduction of e-Commerce and the move by corporations towards Internet-based businesses, the information technology model will need to support a wide range of applications from a diverse IT customer base including existing internal employees, new employees from acquisitions, corporate suppliers, or individual consumers. Further, the user communication model may source from the corporate private network, the telecommuter accessing corporate resources via a remote access connection (e.g., DSL or satellite), or perhaps a corporate partner operating over an extranet via an ISP virtual private network. Information Technology must meet the diverse connectivity requirements as well as specific user requirements for productivity, network flexibility, service differentiation, isolation, privacy, and manageability. The complexity of the network along with a diverse user base is greatly increased under this new computing model. Furthermore, there will be a shift to delivery and management of services. Under this new paradigm, motivated by e-Business computing, we identify the following applied areas for policy-based management.

Service Management

As QoS technology and policy become more commonplace, the Internet will support increased business-to-business communications and inter-domain negotiations to ensure resource preservation and SLA's. This will require SLA specifications to be translated into traffic management policies. Moreover, by way of static or dynamic provisioning the SLAs would be enforced through traffic conditioning and device control mechanisms. The introduction of policy management can facilitate these changes by providing the means to translate high-level business policies into device-specific mechanisms to support SLA management. Such a model is proposed [17], including *customer policy*, *service policy*, and *flow policy*. This model closely follows the policy and provisioning administration models.

Dynamic Provisioning
There is a general industry trend towards more cost-effective models for bandwidth provisioning (e.g., VPNs). Static models, based on leased lines through network carriers, are substituted for dynamic virtual pipes or remote access connections available through either service providers or network carriers. The expectation of support for both short-lived and long-lived networks based on changing capacity and geographical domains, supporting either unscheduled or negotiated schedules, is a reasonable one for ISP customers or IT organizations. In addition to traditional office and business applications, provisioning models will require support for alternative QoS traffic types enabling emerging computing applications: VoIP, video-based technologies supporting real-time streaming applications for online training, or real-time applications to support video conferencing or desktop collaboration. The provisioning or signaling support for intra-domain and inter-domain SLAs, in addition to resource management features for load balancing or rate control, will require tools that can facilitate network control and administration automation over complex networks and alternative user models. Policy-based management tools will provide obvious value here.

Internet Pricing and Billing
Models for Internet-based e-Business will involve some form of accounting for subscribed services and use of shared resources. Although several proposals [18, 19] have been published in the area of Internet pricing, it is possible that the right model is somewhere between the shared flat rate model of Internet ISP's and the network carrier telephony model of usage-based pricing. Moreover, with alternative levels of service (i.e., QoS) becoming available to Internet users or organizational subscribers, the pricing model may be extended to support subscribed service levels, in addition to resource use. Policy-based data stores will support the availability of such information to support billing, SLA auditing, and validation. For example, a billing model based on the following policies is proposed [17]:

- Billing policy: customer type and credit associated with a request for a given service
- Charging policy: charging tables describing service, resources, time, and cost for a domain
- Accounting policy: accounting information about resources used and the cost for each customer

Security
The requirements for policy-based security focus on the integration of policies across administratively separate or heterogeneous domain boundaries. Security has to allow individual consumers, corporate suppliers, and acquisitions/mergers as well as corporate business partners to operate under a shared, and what is perceived to be, border-less infrastructure. End-end security will require a tighter integration of policy across the separate security realms, which are traditionally disjointed across networks, applications, and servers. Finally, dynamic policies will be a requirement for the administrator, providing him/her the means to perform on-the-fly control or automation of short-lived policies to secure content and communications (e.g., inter-company video conference or online supplier training sessions). Enabling such a paradigm will require a higher level of abstraction, automation, and integration across infrastructure elements. We feel that policy-based management solutions would work here by aiding in the definition of allowable security associations, integrating policy abstractions across administrative or heterogeneous security boundaries, facilitating encryption parameterization, and by rapid and dynamic configuration of boundary devices (e.g., VPN, firewall, and proxy services).

SUMMARY AND FUTURE WORK

In this paper, we introduce the key drivers for current IT infrastructure evolution: mission criticality, network architectural agility, service flexibility, and efficiency. To enable these drivers, we presented the background on policy-based management beginning with the necessary set of requirements to realize the technology. We then presented a framework for policy-based management and discussed implementation alternatives to support the proposed framework. We believe that IT business and operational models have to be augmented to transition to a policy-driven management approach; yet, we argue that the changes should help IT organizations align business objectives with more effective use of infrastructure resources. We proposed a simple methodology, which could be implemented by enterprise managers and we suggested that change management and service-level management evolve to align with policy-driven business and operational processes. The ROI decision criteria for policy-based management and technology challenges that IT organizations must address will include the selection and integration of QoS technology, vendor technology interoperability, enabling policy-based security, network management, and directory service integration.

Finally, there are still pending issues that will not be resolved until policy-based management matures industry-wide. Industry standards (primarily IETF and

DMTF) in the areas of policy-based directory schemas, QoS technologies (e.g., DiffServ, RSVP, MPLS), and policy and directory communication protocols (e.g., COPS, LDAP) are still under development and may delay full vendor adoption. Intel is very active in driving technology in the direction of these standards. Additionally, policy scalability, QoS and security conflict resolution, and interoperability will further influence IT strategies and the adoption of PBM technologies.

Over the next year, Intel IT will investigate these issues while gaining experience with policy and QoS technologies. We anticipate continued convergence in the directory arena, as this technology should serve as the foundation for the success of PBM. A widely deployed solution will depend on the eventual integration of alternative technology. The move to e-Business and Internet-based computing will force organizations as well as ISP's to focus on and speed the delivery of a policy-driven approach to managing Internet-based IT infrastructure and enhanced network services.

ACKNOWLEDGMENTS

The authors thank David Mills for his insights and his contributions to the security sections presented in this paper, as well as folks from the Intel Architecture Labs and various groups from within Information Technology including Finance, Strategy & Technology, and Infrastructure Products Engineering for their contributions.

We also thank NSD for their technology support and our external associates, including Hewlett Packard, Cisco Systems, Packeteer, and Net Com Systems for their technology support and collaboration.

REFERENCES

[1] J. Strassner, E. Ellesson, "Terminology for Describing Network Policy and Services," *Internet Draft draft-strassner-policy-terms-01.txt,* Feb. 1999.

[2] R. Rajan, D. Verma, S. Kamat, E. Felstaine, S. Herzog, "A Policy Framework for Integrated and Differentiated Services in the Internet," *IEEE Network Magazine*, Sept./Oct. 1999.

[3] J. Boyle, R. Cohen, D. Durham, S. Herzog, R. Rajan, A. Sastry, "The COPS (Common Open Policy Service) Protocol," *Internet Draft draft- ietf-rap-cops-07.txt.* Aug. 1999.

[4] W. Yeong, T. Howes, S. Kille, "Lightweight Directory Access Protocol," RFC 1777, March 1995.

[5] B. Moore, E. Ellesson, J. Strassner, "Policy Framework Core Information Model," *Internet Draft draft-ietf-policy-core-infomodel-00.txt.* June 1999.

[6] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation Protocol (RSVP)," Version 1 Functional Specification, *RFC 2205*, Sept. 1997.

[7] R. Braden, D. Clark, S. Shenker, "Integrated Services in the Internet Architecture: an Overview," *RFC 1633*, June 1994.

[8] Y. Bernet, S. Blake, J. Binder, M. Carlson, S. Keshav, E. Davies, B. Ohlman, D. Verma, Z. Wang, W. Weiss, "A Framework for Differentiated Services," *Internet-Draft draft-ietf-diffserv-framework-01.txt, work in progress*, Feb. 1999.

[9] R. Callon, P. Doolan, N. Feldman, A. Fredette, G. Swallow, A.Viswanathan, "A Framework for Multiprotocol Label Switching," *Internet Draft, draft-ietf-mpls-framework-01.txt.,* May 1998.

[10] M. Seaman, A. Smith, E. Crawley, J. Wroclawski, "Integrated Service Mappings on IEEE 802 Networks," *Internet Draft, draft-ietf-issll-is802-svc-mapping-03.txt.*, Nov. 1998.

[11] K. Nichols, V. Jacobson, and L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet," *Internet Draft, draft-nichols-diff-svc-arch-00.txt*. Nov. 1997.

[12] Z. Xiao, L.M. Ni, "Internet QoS: Big Picture," *Dept of CS, Michigan State University.*

[13] I. Andrikopoulos, G. Pavlou, "Supporting Differentiated Services in MPLS Networks," *Proc. 7th International Workshop on Quality of Service (IWQOS'99)*, London, May 1999.

[14] R. Yavatkar et al., "A Framework for use of RSVP with Diff-serv Networks," *Internet Draft draft-ietf-diffserv-rsvp-00.txt.,* June 1998.

[15] B. Aiken, J. Strassner, B. Carpenter, I. Foster, C. Lynch, J Mambrette, R. Moore, B. Teitelbaum, "Terminology for Describing Middleware for Network Policy and Services," *Internet Draft, draft-aiken-middleware-reqndef-01.txt.*, May 1999.

[16] Cisco White Paper: *Cisco Networking Services for Active Directory*, available at

http://www.cisco.com/warp/public/cc/cisco/mkt/servprod/sms/common/tech/cnsad_wp.htm

[17] T. Ebata, M. Takihiro, S. Miyake, M. Koizumi, F. Hartanto, G. Carle, "Interdomain QoS Provisioning and Accounting," *Internet Draft*, *draft-ebata-inter-domain-qos-acct-00.txt*., Oct. 1999.

[18] N. Semret, R. F. Liao, A. Campbell, A. Lazar, "Market Pricing of Differentiated Services," *Proc. 7th International Workshop on Quality of Service (IWQOS'99)*, London, May 1999.

[19]S. Shenker, D. Clark, D. Estrin, S. Herzog, "Pricing in Computer Networks: Reshaping the Research Agenda," *ACM Computer Communications Review*, pp. 19-43, 1996.

AUTHORS' BIOGRAPHIES

John Vicente is a member of Intel's IT organization where he is involved with strategy and technology in the areas of Internet-QoS, policy-based networking, multimedia, and programmable networks. John is also working with Dr. Andrew Campbell as a Ph.D. candidate at the Center for Telecommunications Research at Columbia University, New York. He received his M.S. in Electrical Engineering from the University of Southern California, Los Angeles, CA in 1991 and his B.S. in Computer Engineering from Northeastern University, Boston, MA in 1986. His e-mail is john.vicente@intel.com.

Harold Cartmill, since joining Intel in 1990, has applied his extensive LAN, WAN, server, mainframe, and system management expertise to solving administration and support problems in Intel's mainframe and Windows NT server environments. Harry has served as Technical Lead for Intel's Remote LAN, cc:Mail, Global Remote Server, and Event Management projects. His most recent accomplishments have been in the design, implementation, and support of systems management methodologies. Concurrently, Harry is actively pursuing his B.S. degree in electrical engineering at California State University, Sacramento, CA. His estimated completed date is the Spring of 2001. His email is harold.l.cartmill@intel.com.

Glen Maxson worked for the Boeing Company in Seattle, WA for 17 years before coming to Intel in 1994. While at Boeing, Glen spent the majority of his time solving the 'Enterprise Directory' problem, a quest that continues to challenge him as Intel's Directory Service architect. He completed his undergraduate studies at Pennsylvania State University in 1977. Glen holds a Certificate in Data Resource Management (1991). His e-mail is glen.maxson@intel.com.

Shelby Siegel is a network architect in Intel's IT Strategy and Technology organization where he does strategic planning for Intel's worldwide network. He received his B.S. degree in mathematical sciences and his M.S. degree in computer science: computer engineering from Stanford University in 1975. His e-mail is shelby.siegel@intel.com.

Russ Fenger is a senior architect and engineering manager in the Intel Architecture Labs where he leads the research and development of Policy Based Network Management architectures. Russ's other research interests are quality of service in the Internet and in network security. Russ has been with Intel since 1983 after graduating from Iowa State University. His e-mail is russell.j.fenger@intel.com.

Corporate Portal Framework for Transforming Content Chaos on Intranets

Atul Aneja, IT Strategy & Technology, Intel Corp.
Chia Rowan, IT Marketing, Intel Corp.
Brian Brooksby, IT Marketing, Intel Corp.

Index words: corporate portal, enterprise portal, intranet, personalization, categorization, taxonomy, search engine, information overload, repository, architecture, framework, profiles, knowledge management, content management, metadata, XML

ABSTRACT

In this paper, we describe a strategy for managing information overload on corporate Intranets. We define a corporate portal and describe the framework and components that are essential to providing the capability to organize content using categorization; to provide a Web-based interface to information; to personalize the portal, allowing employees to tailor information for their individual requirements; to search multiple repositories such as e-mail, file and data stores, and the World Wide Web; and to access different sources of information through a universal client interface.

INTRODUCTION

In the last few years, information technology and the Internet have exponentially increased the amount of information that Intel employees must process every day. Information is delivered at an astonishing pace and from a dizzying array of sources such as e-mail, news, documents, reports, articles, digital files, video and audio files, and transactional data. Yet, it is difficult to take advantage of this wealth of information because it is buried in separate, often disconnected and disorganized repositories. In addition, the volume of data leads to information overload for our employees.

Herbert Simon, an economist, describes information overload as follows:

What information consumes is rather obvious: it consumes the attention of its recipients. Hence, a wealth of information creates a poverty of attention and a need to allocate that attention effectively among the overabundance of information sources that might consume it.

We must confront the problem of information overload at many different levels, using a combination of approaches. A corporate portal will help transform some of the chaos existing today on Intel's Intranet.

INFORMATION OVERLOAD

E-mail is a key cause of information overload at Intel. But it isn't e-mail alone that creates this problem. We have more than one million URLs on our Intranet, with more than 100 new Web sites introduced every month. Much of this information is stored in a disorganized fashion resulting in some business decisions being based on incomplete or out-of-date information.

The following is a short list of key issues that make it difficult to access information in a timely manner:

- Information is scattered throughout Intel in personal documents, e-mails, transcripts of discussions, etc.
- Finding relevant, accurate information is time-consuming, difficult (if not impossible), and often requires searching multiple systems.
- Information is accessed through different methods such as Web browsers, e-mail clients, and applications.

These combined issues result in the loss of productive time spent searching for information, the increased likelihood of making decisions based on incomplete and inaccurate data, and the failure to effectively respond to important messages and information. Our research on these issues suggests a corporate portal (sometimes referred to as an enterprise portal) as a potential solution.

THE CORPORATE PORTAL CONCEPT

Corporate portals are a relatively new business concept. A corporate portal is an Intranet site that is similar in design to popular Internet sites such as My Yahoo*. It offers a single point of access for the pooling, interaction, and distribution of organizational information [4]. This browser-based system provides universal access to business-related information in the same way that an Internet portal acts as a gateway to the wealth of content on the Web [1]. A corporate portal enables a company to provide users with a single gateway to the personalized information they need to make informed business decisions [1]. Thus, a corporate portal can increase employee productivity by addressing many of the issues described in the previous section [1,2,3,4].

Figure 1 shows the corporate portal adoption rate based on a Delphi Group survey of Fortune 500 companies [4]. About 35% of these companies have implemented a corporate portal and another 30% are in the pilot/experimental stage of development.

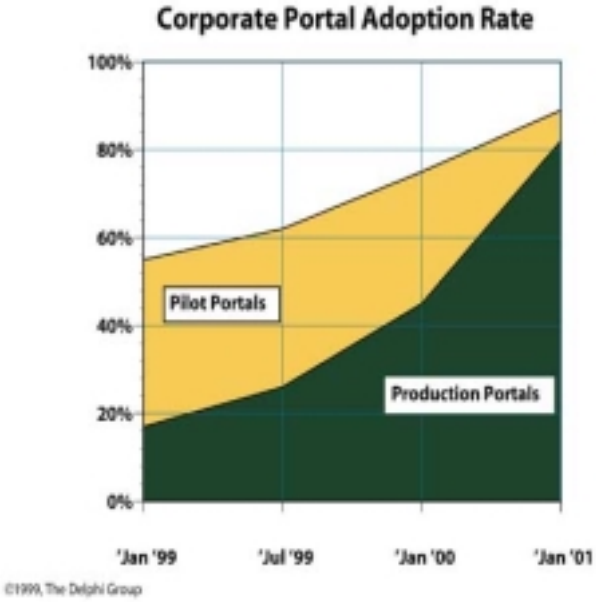


Figure 1: Corporate portal adoption rate

CORPORATE PORTAL STRATEGY

The key steps to Intel's corporate portal strategy include the following:

- Identify the content that is or will be available, and identify where this content resides.
- Leverage existing systems, resources, and repositories.
- Include both structured and unstructured information.
- Organize content into categories that can be browsed and searched.
- Integrate search functionality across multiple information repositories.
- Build a platform for publishing and subscribing to content.
- Deliver personalized content and services to users based on their preferences and roles.
- Develop the corporate portal in phases.
- Create online "communities" to connect people and enable collaborative work.
- Develop an extensible architecture that allows for extended functionality.
- Sustain a collaborative portal by " institutionalizing" it within daily business operations and weaving it into long-term strategies.
- Purchase an integrated portal product rather than building custom portal functionality.

Business Benefits

The main benefit of a corporate portal is the increased employee productivity that results from the following improvements:

- organized and structured information, which is easier to navigate
- quick access to relevant personalized news, information, services, applications, and documents
- a highly interactive and personalized interface that provides targeted information based on employees' roles and preferences
- enhanced search capabilities that reduce the amount of time necessary to find sought after information
- filtered, targeted, and categorized information so users receive just what they need

CORPORATE PORTAL FRAMEWORK AND CAPABILITIES

After an extensive R&D project the team identified an appropriate set of capabilities and infrastructure elements for a corporate portal (see Figure 2).

* Other brands and names are the property of their respective owners.

(Due to the breadth of corporate portal functional possibilities, focusing on the key requirements is critical. This framework could be adapted to fit different business requirements.)

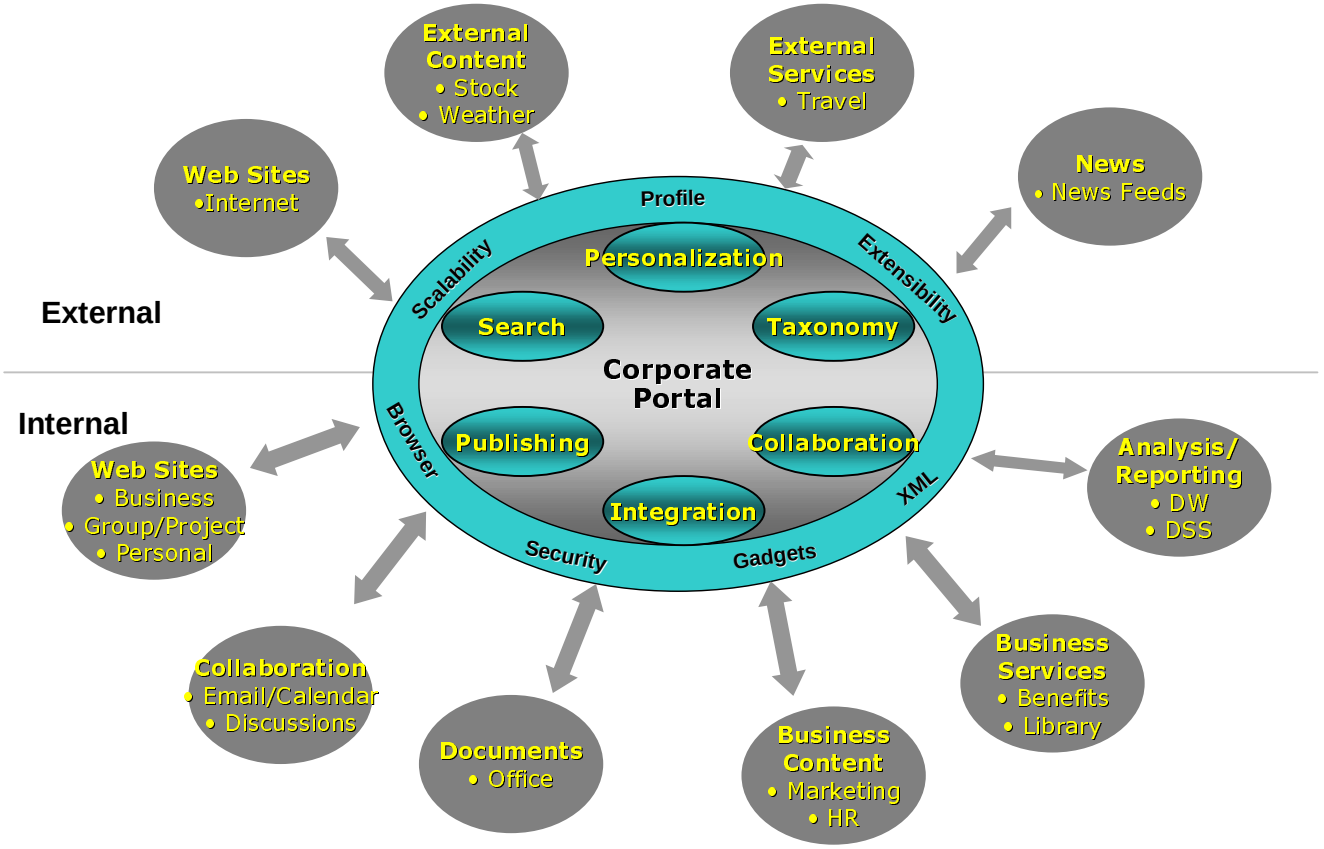


Figure 2: Corporate Portal Framework

Categorization

The sheer volume of data maintained on Intel's Intranet Web sites is expanding rapidly and is scattered throughout the company without any context. A portal can provide the structure, organization, and context necessary to transform this information into a significant Intel resource. A Yahoo*-like structure that organizes information under specific categories is an efficient way to enable employees to find specific information of interest.

Categorization of corporate information is critical because it provides employees with a navigation directory that can be browsed to find specific information. During the categorization process, each piece of information must be labeled to indicate what it contains and how it relates to other pieces of information. For example, a collection of documents can be organized alphabetically, or by subject, function, business group, geography, projects, products, etc. The challenge is to create categories that make the most sense to business users. To address the needs of different users, it can be an effective strategy to create multiple views of the same sets of documents.

The first step in the categorization process is to identify the high-level categories under which information can be organized. Once this classification, or taxonomy, is created, the next step is to define the process of evaluating how content and documents will be indexed within the taxonomy. In the past, this was done manually. Today, automatic categorization technology is maturing quickly; although it still requires manual intervention by a librarian. In either case, users must be able to use a Web browser to browse the taxonomy.

One of the key issues for searching and assessing content is establishing context: what is it, what it is about, when was it created, who created it, and for what purpose was it

* Other brands and names are the property of their respective owners.

created. Metadata can help establish this context. Metadata is information about information. For example, metadata consists of keywords and summaries that help provide context for information on the Web. In addition, metadata from different documents can be used to describe the relationships among documents.

Content Publication and Management

Content lifecycle management helps to prevent content from becoming dated. One useful practice is to specify an expiration date for all content. When the content expires, the owner is notified and can then extend the content expiration date or let the content be archived. Therefore, when creating a corporate portal, the process for authoring, approving, and publishing different types of content must be defined. Utilizing an automated workflow can ease this process.

The most effective time to capture metadata is during the authoring process because authors usually have the most knowledge about the content. Some document formats allow metadata to be stored inside the document itself. HTML contains <TITLE> and <META> elements for storing such metadata as titles, keywords, descriptions, and author information. Microsoft Office* documents can store standard summary information such as title, description, and author.

Where possible, XML tags should be used to identify each searchable attribute in a document so that search engines, robots, and applications can locate the document when searched. Using metadata to search within documents for specific fields such as author, title, or location enhances the ability to pinpoint information.

Integrated Search

Because corporate information resides in numerous places, an integrated search capability across multiple information repositories (Web/Intranet, e-mail, discussion forums, databases, and applications) is essential.

A full-text search function can produce highly accurate results. As mentioned above, the capture and use of metadata across content is another way to improve search results. Whether searching text or metadata, the search capability must function both within and among selected repositories.

Personalization

The goal of personalization is to deliver content relevant to an individual user or group of users based on their roles and preferences. The ability to present a personalized, relevant set of information is critical to portal success. Each user's portal experience must be tailored to that user's preferences, security levels, and access authorizations. Also, users should be able to subscribe to specific content, and should be informed (by a notification service) when that content changes.

Personal profiles are the architectural components that contain user information and preferences. These profiles can be created from information manually entered by the user, or by gathering user information from existing databases. User profiles can then be used by portal services to tailor their content for the user.

Several architectural considerations are necessary for personal profiles. Should there be a profile within each application that will be personalized. If yes, should the profile be based on a common schema. Or, is the personal profile commonly shared across applications. Using a common personal profile offers several advantages, including a single place to store and manage user profile data. This removes the necessity of having the same user (who needs access to different systems) in multiple profiles.

When creating a personal profile, it is often difficult for users to specify all their preferences up front. Over time, their preferences will evolve and change. Through user behavior monitoring and analysis, automatic profiling can help to address the issue of out-of-date user profile data. In addition, users need to have the ability to modify and update their profile data.

Moreover, because user profiles contain personal data, privacy is a significant issue. Privacy strategies must be established to effectively protect private data. Because many countries have specific privacy laws and requirements, an understanding of international privacy laws is also critical.

Goal-Oriented Interface and Navigation

The design of the interface and navigation scheme is key to the success of a corporate portal. Information must be available through a common, browser-based user interface. Moreover, the interface must be intuitive enough that users do not need to understand and configure a complex system in order to gain value from the site.

Developing an intuitive user interface requires significant research. The first step in the process is to define who the users of the site will be. Next, learn about their goals and

* Other brands and names are the property of their respective owners.

motivations, such as what problems are they trying to solve. An understanding of how users do their work is also very helpful. Then ask how the interface can help the user reach these goals.

Formulate your usability criteria, and then test it with the potential users of the portal site. Show the site to them and document their experiences. Watch them navigate. Note their body language as pages appear on the monitor. Most importantly, let them control the mouse. Finally, use the information you gain during the testing period to make adjustments to the interface. And, make sure the design has built-in flexibility so that adjustments can be made quickly and easily throughout the life of the portal site.

Integration

The ability to present a unified view of corporate information depends on integration capabilities. A key enabler is the definition of an information architecture for the corporate portal.

As mentioned previously, corporate information is spread across many sources within Intel. The seamless integration of information from both structured and unstructured repositories is a challenge. The appropriate level of integration must be determined. Figure 3 shows various information repositories that need to be integrated through the corporate portal.

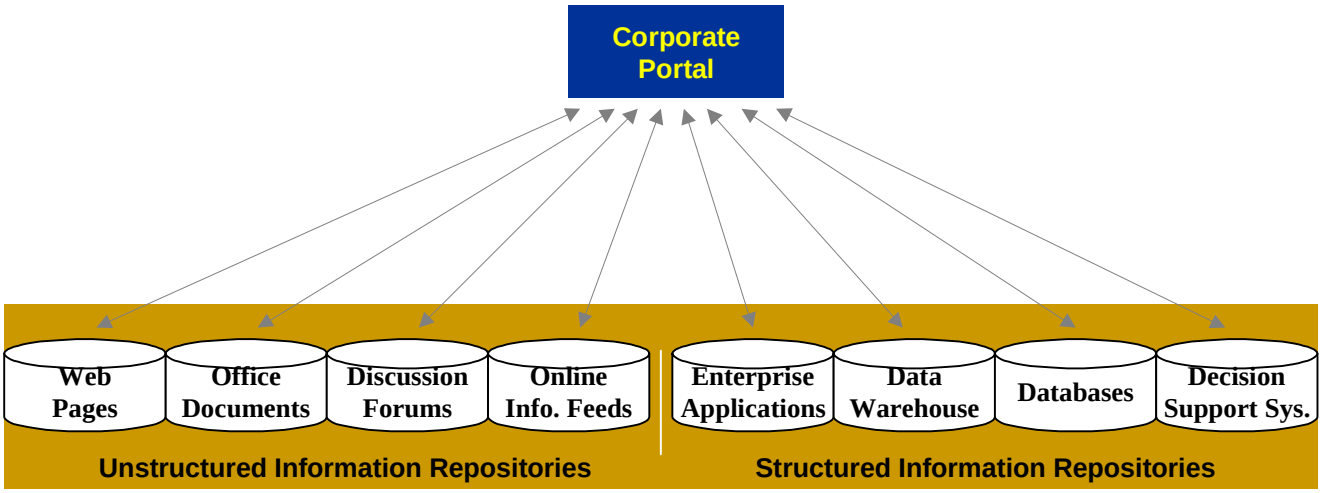


Figure 3: Structured and unstructured repositories

Collaboration

A corporate portal can effectively create a shared community across the organization. This community is critical in an age when virtual teams and groups are becoming more significant.

The corporate portal also can be used to project the organizational identity, deliver corporate messages, and interact with employees.

Advanced capabilities to build community are critical to a corporate portal. Asynchronous capabilities include e-mail, discussion forums, etc. Synchronous capabilities include online meetings, video conferencing, and chat. Network bandwidth must be considered when implementing synchronous collaboration capabilities.

CORPORATE PORTAL ARCHITECTURE

Figure 4 shows a corporate portal capability architecture. A few of the architectural principles underlying a corporate portal are described below.

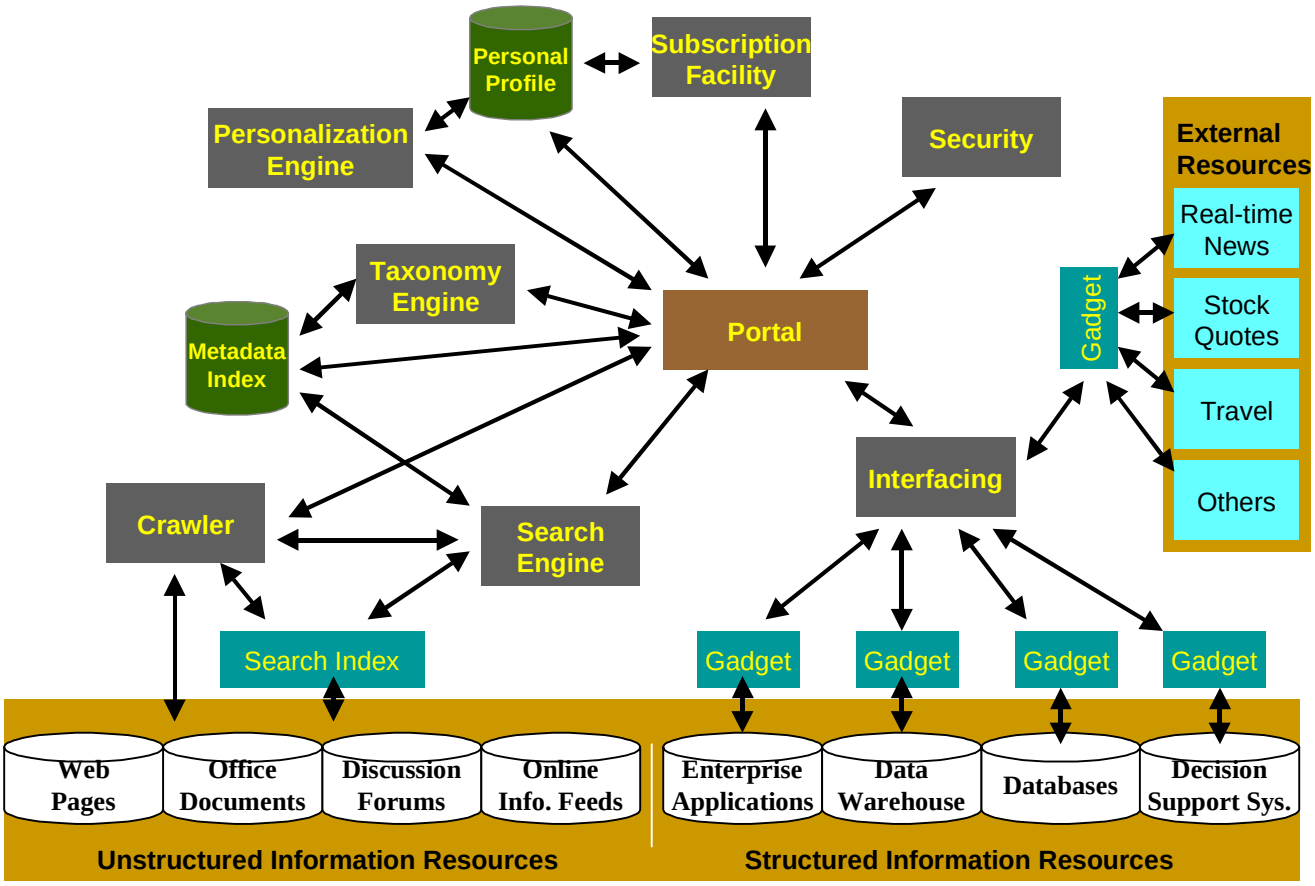


Figure 4 : Corporate portal capability architecture

Plug-and-Play

The portal framework needs to offer a plug-and-play capability that will allow additional functionality as the portal grows to meet future requirements.

Standards-Based Tools

The portal framework utilized several emerging standards, specifically, HTTP, IP, HTML, and XML.

The eXtensible Markup Language (XML) separates content from how that content is presented. Data is kept completely independent from its known, and maybe yet-to-be-determined, output format(s). The separation of data and format allows the same content to be presented in multiple views. One possible application for the corporate portal is building personalized views based on user preferences.

Style sheets determine how an XML-based document is rendered for presentation. Style sheet standards include the following:

- Document Style and Semantics Specification Language (DSSSL) provides facilities for display and transformation of SGML and XML data.
- Cascading Style sheets (CSS) provide straightforward, generally sufficient features for using styles in the context of XML and HTML.
- Extensible Style Language (XSL) is designed in a template-oriented fashion to provide powerful features to display and transform XML data.

Extensibility Through "Gadgets"

"Gadgets" are new application tools and services in the portal, provided via modular components. "Gadgets" provide the architectural construct to enable future extensibility without having to completely redevelop the portal. By using gadgets, new functionality could be easily added.

CHALLENGES

There are two key challenges facing developers of corporate portals: available technology and viability of corporate information. Today, many technology suppliers are repositioning themselves as portal vendors, but not all of these vendors have comparable solutions. A careful analysis of these vendors is necessary to fully understand their specific capabilities and technical approach.

It is also important to realize that a portal is only as good as the corporate information repositories that it accesses. If the information in these repositories is not correct or up to date, then the portal information will not be correct or up to date.

DISCUSSION

Do integrated packaged portal systems provide a better solution than custom-developed corporate portals?

Two potential approaches can be taken when creating a corporate portal. The first approach is for developers to select and use the best solution for each capability defined in the corporate portal framework. The second approach is to look for an integrated solution that offers most of the defined capabilities. Our research suggests that truly integrated corporate portal solutions contain more functionality, are easier to maintain, less expensive over the complete lifecycle, and are faster to deploy.

CONCLUSION

Corporate portals are the logical evolution of Intranet sites as organizations continue to look for ways to improve employee productivity and job satisfaction. To this end, Circuit, Intel's Intranet home page, is evolving into a corporate portal. Currently there is a team working on a next generation prototype of this corporate portal and a pilot of this new technology is planned for Q2, 2000.

Due to the fast pace of change in portal technology and in business requirements, the strategy and architecture of a corporate portal should be flexible enough to evolve over time in response to these ongoing changes.

ACKNOWLEDGMENTS

We thank IT R&D Council for allowing us to research this key emerging area. We also thank IT management for recognizing this area of importance and for believing in us. We acknowledge the cross-functional R&D Corporate Portal Team .

REFERENCES

[1] *Enterprise Information Portals*, Merril Lynch, 16 November 1998.

[2] G. Phifer, *Portal Trends Emerge Among Confusion*, Gartner Group, April 1999.

[3] M. West, *A Framework for Enterprise Portal Simplifies Intranets*, Gartner Group, April 1999.

[4] *Corporate Portals*, Delphi Group, April 1999.

[5] Reynolds, H. and Koulopoulos, T., *Enterprise Knowledge Has a Face*, March 30, 1999, Intelligent Enterprise.

AUTHORS' BIOGRAPHIES

Atul Aneja is an architect in the IT Strategy & Technology group where he leads Intel's R&D efforts in the areas of corporate portals, personalization, and e-Business architecture. His other research interests include software architecture and emerging application capabilities. His e-mail is atul.aneja@intel.com.

Chia Rowan is a manager of Circuit, Intel's corporate Intranet site and is responsible for its evolution into a corporate portal. Her email is chia.l.rowan@intel.com.

Brian Brooksby is the Technical Marketing Engineer for "Circuit," Intel's corporate Intranet site. He has designed document management and Web publishing systems for several Intel Web sites and is currently focused on developing Intel's corporate portal. His e-mail is brian.brooksby@intel.com.

Implementing an Industry e-Business Initiative: Getting to RosettaNet

Patricia J. O'Sullivan, IT Strategy & Technology, Intel Corp.
Don S. Whitecar, PE, IT e-Business Integration, Intel Corp.

Index words: e-Business, e-Commerce, standards, standards implementation, enterprise application integration, EAI, RosettaNet, B2B, trading partner automation, trading partner integration, B2B gateway, XML, WWW

ABSTRACT

As Intel looked at the cost of its own successful early implementation of Web-based e-Commerce, it became clear that an industry-wide standards-based approach to e-Business is the only way to go.

We decided to help build the right business-to-business (B2B) specifications with the right industry initiative (RosettaNet* is our main focus) and then implement those specifications. As early adopters, this has turned out to be much more of an enterprise readiness effort than initially appreciated. Team composition, technical and business knowledge coalescence, formal and informal communication channels, cross-enterprise visibility, and establishment of appropriate resource levels are just some of the challenges we face.

We anticipate that an evolving, more robust infrastructure, together with lessons learnt from pilot projects, team experience, and more mature standards will lead to the full realization of expected benefits from RosettaNet. However, we offer here a "readiness model" that we hope can be used by others to "spin up" faster.

INTRODUCTION

In early 1998, Paul Otellini, then Sr. Vice President of Intel's Sales and Marketing Group, crystallized much of our early thinking and experimenting with Internet-based e-Commerce into a simple challenge: take in $1Billion in sales orders via the Web in Q4'98. We took our first such order in July 1998 and had arrived at $1B *per month* by the start of Q4'98—success beyond our wildest dreams!

So, with that success, one may well ask what the problem is. Well, each customer's internal processes and systems are almost always different from ours, so we had no way to ensure that our applications, which worked well for us, did not introduce extra work or become otherwise burdensome for our customers. And, our customers buy many products from many suppliers in order to make up their complete product lines, so they are potentially facing extra work from *each* of their suppliers. As we moved forward with various plans to Internet-enable the way we do business with our customers, we also realized that the part of Intel that *buys* products was getting ready to establish a whole series of web-based procurement applications that we wanted our suppliers to use.

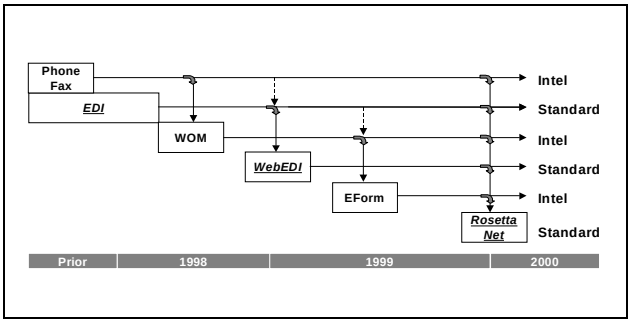
So, not only were we building a suite of applications that did not necessarily optimize e-Business for our trading partners (the phrase used generally in the e-Business arena to refer to other companies with which we do business, whether as customers, suppliers, or other), but we were facing the prospect of developing and/or buying a whole slew of applications that we (and our trading partners) would have to support and maintain over time.

What we needed were standards! However, we did **not** want standards that took years to develop; rather, we needed those that evolved at the same pace as the Internet, at the same pace as the emerging "killer app" of e-Business, and at the same pace as the technologies that underlay that growth. Furthermore, these standards had to focus on the real-world business processes of the supply chains that we are a part of, not those that tried to create a single universal e-Business solution or that sacrificed implementation to elegant technical solutions.

* Third-party brands and names are the property of their respective owners.

Of equal importance, these standards would need a sound, extensible architecture; would have to be adopted rapidly; and would have to be *demanded* by management and supported by business and technology stakeholders within our business environment.

Figure 1 illustrates the tension between some of Intel's e-Commerce solutions and our desire to use standard



solutions, as projected over time.

Figure 1: Intel standards challenge

As is true throughout history, but even more noticeable in today's Internet economy, timing is everything. In late 1997, an executive from one of the largest computing products distributors in the world approached us (as well as other leading players in the "IT Products" supply chain) with a vision and a plan. The vision was to create a common vocabulary and process set for e-Business in the context of a well understood supply chain. The plan was to pull together a fast-moving business consortium of companies representing over half the revenue of that supply chain, managed by a board of top executives from member companies, who would *precommit* to implementing the specifications that their members would jointly develop and vote upon.

At Intel, we pulled together a quick evaluation team, surveyed both our internal e-Business initiatives (such as Web Order Management (WOM), Supply Line Management (SLM), and eFORM) and the external e-Business standards/initiatives environment, and we quickly concluded that we needed to help realize this effort, and the sooner, the better. In the words of Colin Evans, Intel's Sales & Marketing e-Business architect, our competitive advantage would have to move from "first to move" to "first to standards."

And thus begins the lessons we have learned (and are still learning) about implementing business-to-business standards across the enterprise.

INTEL'S ROSETTANET IMPLEMENTATION

Today we are preparing to meet our first major commitment to have a production-level implementation of at least one of the RosettaNet "partner interface process" (PIP) specifications (which are described more fully below) running on a robust infrastructure, with at least one trading partner. Among RosettaNet members, this milestone is known as "2.2.2000," which is the date that we will all be ready to demonstrate our success. We are in the thick of this implementation, learning lessons every hour.

The main focus of this paper is on what it takes to be internally *ready* to adopt and implement the suite of specifications collectively known as RosettaNet. We try to make these observations as concrete as possible, without being necessarily RosettaNet-specific. They should be of interest to anyone who is preparing to implement any standards-based approach to e-Business.

In order to understand the magnitude of our implementation effort (both initial and longer-term), it is necessary to provide a little background on the RosettaNet business and technical architecture. The bulk of this paper focuses on our use of a readiness model to ensure that our solutions could be deployed.

ROSETTANET OVERVIEW

Although RosettaNet's supply chain scope began with IT products (e.g., boards, systems, peripherals, finished systems), it has expanded to include electronic components (e.g., chips, connectors). Intel obviously plays a role in both of these supply chains (often abbreviated as IT and EC). As maturity is gained in these environments, it is likely that RosettaNet's business scope will expand to other supply chains as well. Each supply chain's standardization efforts are overseen by a managing board composed of member company executives, who prioritize efforts, ensure synergy between supply chains as much as possible, and oversee resource allocation as administered by a paid staff.

RosettaNet focuses on three key areas requiring standardization in order to automate business interchanges between trading partners. First, vocabulary needs to be aligned; this includes both business and technical terminology germane to the transaction at hand. The RosettaNet Dictionary, drawing upon existing industry standards wherever possible, fills this need. Second, the way in which business messages are wrapped and transported must be specified. The RosettaNet Implementation Framework, which specifies the use of XML (Extensible Markup

Language), the World Wide Web (WWW), and other protocols serves this need. And third (and most important) the business processes governing the interchange of the business messages themselves must be analyzed, harmonized, and specified. RosettaNet terms these "Partner Interface Processes" or PIPs. Figure 2 shows these RosettaNet "ingredients."

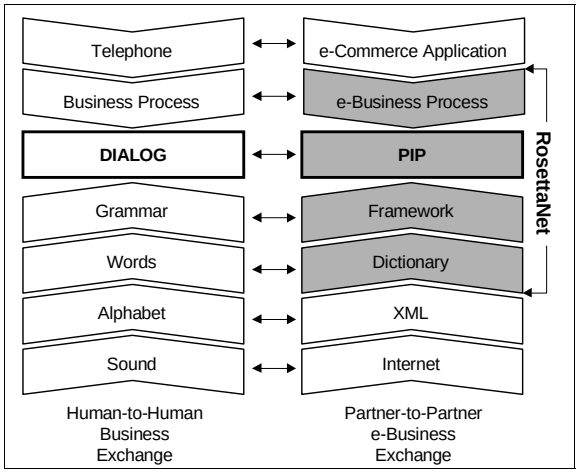


Figure 2: RosettaNet ingredients

To perform the work of analyzing, recommending, and documenting proposals for voting by the membership, RosettaNet member companies volunteer expert resources, both business and technical people, to lead and/or be a part of project teams. These people are either on part-time project duty or on detached full-time (short-term) assignments.

At present, six "clusters" of business activities (such as "Order Management") have been identified as initial targets of RosettaNet standardization efforts by the RosettaNet Managing Boards. Within those clusters, "segments" have been identified (e.g., within the Order Management cluster, "Quote & Order Entry" is one of four segments. Each segment is then analyzed in workshops that identify the necessary PIPs and document the choreography and business requirements around each PIP. RosettaNet anticipates that between 100 and 120 PIPs will result from the six clusters.

It is worth noting that when the second supply chain (EC) was added to RosettaNet's business scope, only two additional segments (and no additional clusters) had to be added. There is more synergy among related supply chains than many had guessed; this gives us more optimistic expectations for the addition of related supply chains.

Current RosettaNet clusters and segments are as follows:

- review segments: partner review; product/service review
- product introduction segments: preparation for distribution; product change notification
- marketing management segments: marketing campaign management; lead and opportunity management; design win management (EC only)
- order management segments: quote and order entry; transportation and distribution; product configuration; returns and finance management
- inventory management segments: price protection; collaborative forecasting; inventory allocation and replenishment; inventory and sales reporting; ship from stock and debit/credit (EC only)
- service and support segments: warranty management; asset management; technical support

RosettaNet member companies are increasingly realizing that, although the PIPs specify processes only at the point of interface between trading partners, the full value of their implementations will come when they align their internal processes with the PIPs as well. This makes it all the more imperative to have a tool with which to evaluate internal readiness for making the shift to standards-based e-Business.

IMPLEMENTATION READINESS MODEL

Implementing a business-to-business (B2B) message exchange environment such as RosettaNet[*] has turned out to be much more of an enterprise readiness effort than initially anticipated. Team composition (size, diversity), technical and business knowledge coalescence, communication channels, at-large evangelism, cross-enterprise visibility, and establishing appropriate resource levels are just some of the challenges.

As an early adopter, we of course experience more pain than those who will follow. More supply chain experience, B2B gateway products, internal infrastructure, pilot projects, internal experience, and standards maturation will all ease the way. However, knowing where to look for "readiness" (or lack thereof) is critical to putting together a workable implementation

[*] Third-party brands and names are the property of their respective owners.

plan. To that end, our Implementation Readiness Model has identified four primary and six secondary readiness tracks to date.

The primary readiness tracks are as follows:

1. business strategy
2. B2B infrastructure
3. business process
4. application development

The secondary readiness tracks are as follows:

1. B2B external initiative
2. trading partner
3. solution provider
4. legal
5. security
6. audit

These tracks were identified as Intel went through the following process:

- early pilot (proof of concept) initiated by our Sales and Marketing Group (Internet Marketing and e-Commerce organization) and one trading partner (completed in August 1998)
- engagement with Intel IT (e-Business Integration) to provide necessary infrastructure to ramp into production mode
- involvement of Intel Planning and Logistics Group to rationalize current business processes and ERP (Enterprise Resource Planning) systems with RosettaNet processes
- cross-enterprise collaboration to get to 2.2.2000

Our next steps will be to drive a significant increase in participation by business process and application development groups in order to further deploy RosettaNet.

We are using this Readiness Model to help us get there.

BUSINESS STRATEGY READINESS

The purpose of this track is to assess the maturity of an enterprise's B2B strategy at large. This is important because B2B solutions are currently strategically divided between browser-based (user-interface) on-demand applications and automated service applications that do not require user interfaces. RosettaNet implementers are primarily focused on trading partner automation to either rehost their Electronic Data Interchange (EDI) processes or reduce the need for browser-based applications. At present, little or no attention is being focused on feeding a B2B gateway with RosettaNet messages generated by browser-based application backend processes. Implementers should recognize that over time, as more trading partner business processes are automated, RosettaNet would reduce the need for on-demand B2B applications that provide a user interface. Therefore, many current browser-based B2B projects, funding initiatives, and roadmaps need to be re-evaluated to see whether an end of life timeframe exists. Key criteria in this track include a company's B2B strategy and a company's buy-side and sell-side motivations.

Intel, like many other companies, is using the Internet as a means to improve and simplify processes and services with its trading partners. Due to RosettaNet and similar initiatives, B2B solutions are evolving from trading partner portals or point applications requiring user interaction to automated solutions. This B2B automation evolution is enabling a shift from "engage customer eyeballs" to "customer at work," allowing customers (indeed, all trading partners) to use their own internal solutions while having immediate access to all information within their global enterprise. In other words, by enabling RosettaNet, companies should be able to reduce overall data entry and data interpretation costs. Automated data exchange and processes provide for higher quality data and faster processing time, as well as create the possibility of an event-driven global enterprise.

The motivations to implement RosettaNet may be greater within a company's buy-side or sell-side. However, implementing RosettaNet ultimately needs to encompass both the buy-side and sell-side of an enterprise's at-large B2B strategy. Implementers need to be sure to identify benefits by looking at the entire supply chain; that is, their customers' customers through their suppliers' suppliers. (This level of impact upon one's strategy will depend greatly upon how much of a company's purchased materials and finished products fall within the RosettaNet consortium scope of coverage.) Implementers should identify their other supply chains on both their sell and buy sides, and they should investigate how other B2B initiatives for trading partner automation are evolving.

The push to implement RosettaNet currently appears to be driven more by buyers in an attempt to simplify and improve productivity and margins. This may be because buyers naturally tend to engage suppliers with whom they can work more easily. However, as B2B automation spreads, we should see suppliers using their proven benefits to persuade their non-automated

customers to participate in automated services. For example, a seller should be able to provide its customers with improved pricing and availability when its customers provide real-time demand and inventory/sales out reporting instead of infrequent non-automated inputs. (Now, imagine the gains if an entire supply chain were to automate its demand and inventory/sales out reporting from end to end—this is one of RosettaNet's objectives).

A company's B2B strategy also needs to take into account the integration of mergers and acquisitions. Another of RosettaNet's benefits would be improved flexibility and agility as companies grow their core business by enabling a standard message exchange framework.

Finally, a company's B2B strategy needs to recognize the full potential of RosettaNet. Through the use of a self-describing message structure that includes a supply-chain dictionary-driven schema and meta-model for more than 100 business processes, RosettaNet supplies a strategic benefit. This message structure holds the potential of becoming a de facto message exchange standard in the near future as agent-to-service and service-to-service architectures evolve. The RosettaNet message structure is in fact sufficiently rich that it could be said to be a document database; RosettaNet messages could be used as disconnected documents passed between applications and databases within a disparate and distributed architecture.

B2B Infrastructure Readiness

The purpose of this track is to define the infrastructure and to assess the level of effort needed to achieve it. This is important because becoming RosettaNet-compliant is only a small portion of the big picture. Although RosettaNet specifies a message structure, a message dictionary, a message exchange framework, and a message exchange protocol, it does not specify the infrastructure needed nor the backend processes required to receive, process, or send messages. Infrastructure is individually managed by each trading partner. Key criteria in this track include infrastructure components, e-Business standards and guidelines, and B2B gateway capabilities.

B2B message integration involves both public and private aspects. Receiving, unpacking, and routing a message, or assembling and sending a message (the public part) is relatively easy. The private (and more difficult) part of message integration includes process automation, workflow, and application integration that link into enterprise applications. In other words, the private part is the intra-enterprise application integration portion of enterprise application integration (EAI), while RosettaNet is the public trading partner application integration part of EAI.

RosettaNet is targeted for use within e-Business applications, predominantly B2B service-to-service applications; however, much of the infrastructure needed to support this is the same as required for B2C (business-to-consumer) and B2B browser-based applications. Many infrastructure components need to exist in order to proceed. Major infrastructure elements include an e-Business "landing zone," facilities, firewalls, proxy servers, networks, routers, communication services, and web servers.

A B2B gateway is needed. It must provide for inbound message receipt, authentication, authorization, entitlement, logging, and routing to a process automation workflow tool. This gateway also must support outbound message construction, packaging and logging. The B2B gateway must be able to provide RosettaNet-compliant messaging and also should be capable of supporting other B2B specifications, perhaps even EDI, file transfer protocol (FTP) and simple mail transfer protocol (SMTP) transport protocols. The build vs. buy study needs to be completed, while looking at maturing product offerings from solution providers.

RosettaNet provides for two basic types of messages: a transaction process message (Figure 3); and a subscription model message (Figure 4).

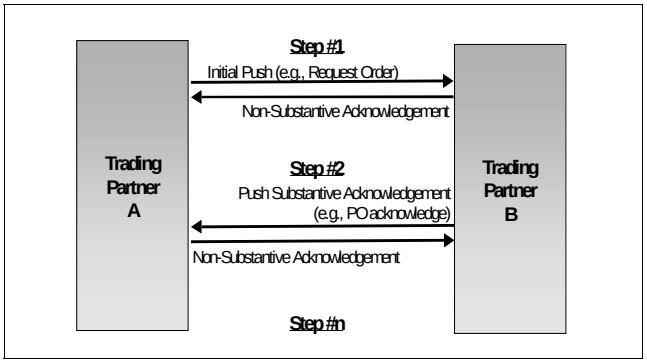


Figure 3: Transaction process model

In order to implement subscriptions, a collection of document repository, subscription, notification, and publication services needs to be provided. This is potentially a very large effort, and again, a few solution providers are working in this space.

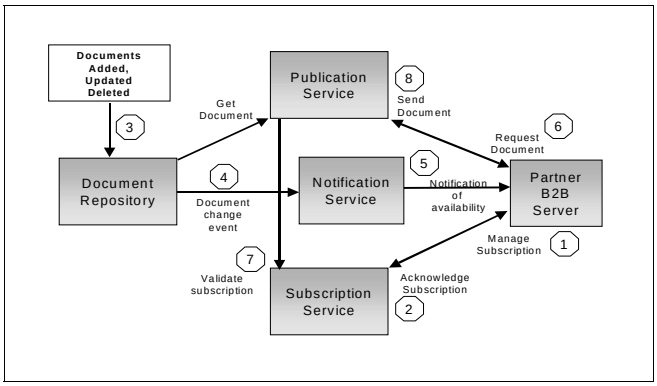


Figure 4: Subscription model

The major services within a B2B gateway include the following:

1. a trading partner database for a directory of trading partners, trading partner processes, and process parameters and their entitlements
2. non-repudiation (legal proof) archiving of message origin and content
3. public key infrastructure (PKI) repository of digital certificates and signatures for encryption, authorization, and authentication
4. PIP templates for integration to public and private process/workflow automation processes
5. virus detection capabilities for message attachments

The B2B gateway will likely coincide or integrate with existing gateways for FTP, value added network (VAN/EDI), and SMTP. Each of these gateways should comply with similar guidelines, designs, and implementations of authorization, entitlement, authentication, privacy, confidential document, and legal trade agreement practices.

RosettaNet is based upon the hypertext transfer protocol/secure (HTTP/HTTPS) protocol in an automated service-to-service framework that does not need visible or attended Web pages. Because HTTPS is needed for security, internal corporate guidelines for PKI and secure socket layer (SSL) encryption must be established. These guidelines should be compatible with existing B2B browser-based implementations that use HTTP/HTTPS.

The B2B gateway will also need a set of complementary services, such as the following:

- Receipt and routing—a public processing area that receives, authenticates, validates entitlement, archives, and routes inbound messages
- Package and delivery—a public processing area that packages, encrypts, validates entitlement, digitally signs, archives, and delivers outbound messages
- Process automation and application integration—a private processing area that provides for process automation and backend integration of inbound messages and outbound messages
- Infrastructure for non-repudiation database (NRdb), trading partner database (TPdb) and PKI
- Notification services for e-mail, pager, etc.
- XML/HTML scraping—ability to extract data from remote trading partner Web pages, in addition to or in lieu of data passed within RosettaNet messages
- Trading partner portal—a portal where trading partners can self-administer their RosettaNet processes and subscriptions
- Testing facilities—the ability for trading partners to test their RosettaNet messages against a test site; after self-testing, the B2B gateway would then promote the trading partner from "test" to "production" status, and thereby allow trading partners to control their production messaging processes
- Satellite capability—a "host" could provide its non-automated trading partners with a B2B satellite solution, thereby acting as a hub, developing and supporting a B2B application at its trading partner facilities

Figure 5 illustrates an integrated B2B gateway, complete with support for RosettaNet, other B2B initiatives, and SMTP, FTP, and VAN/EDI.

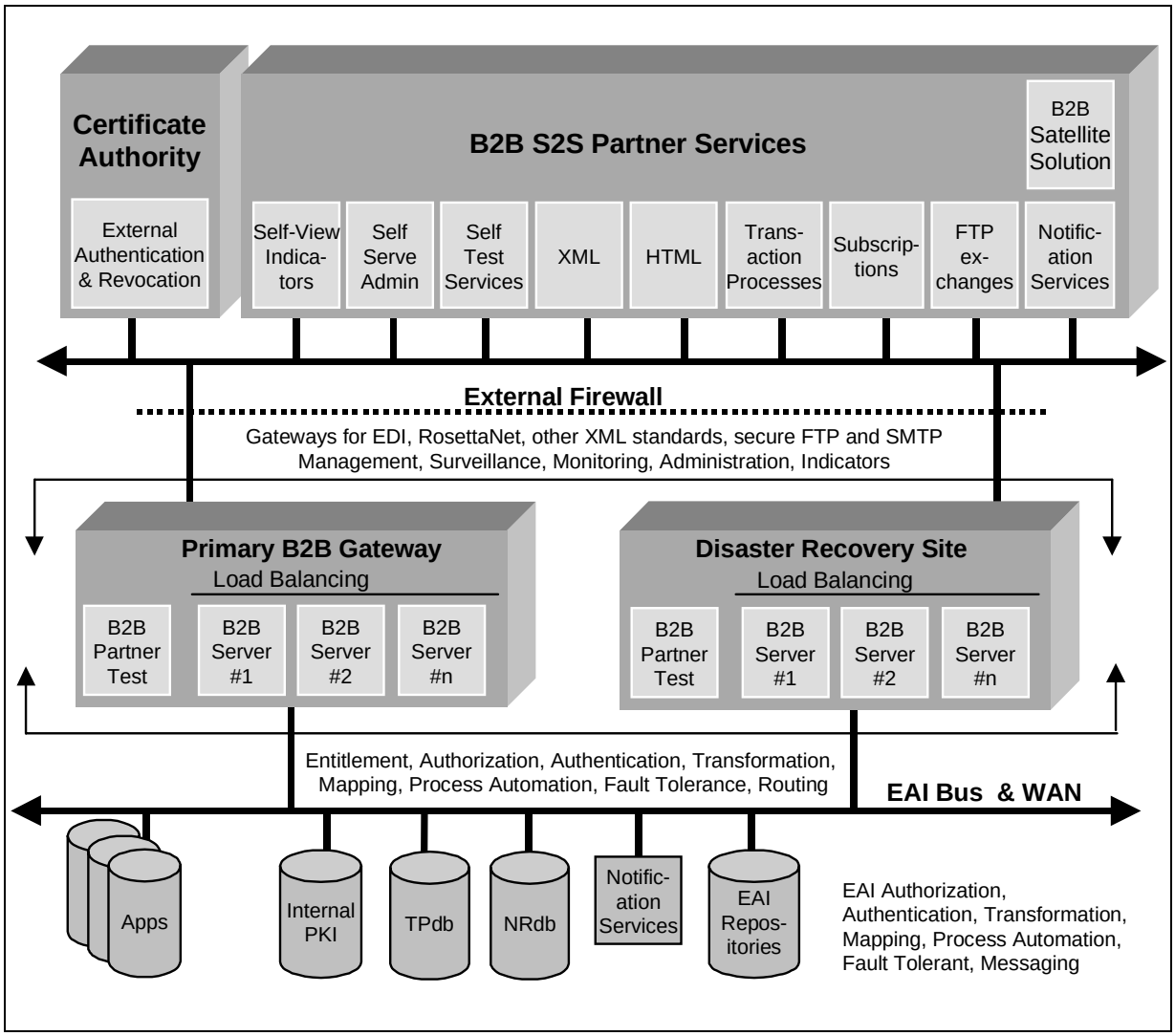


Figure 5: An integrated B2B gateway

Infrastructure readiness can be a huge task. However, it is not necessary to do it all at once. It is possible to install a solution from a RosettaNet solution provider within a few weeks and be up and running for a small implementation. Performing a comprehensive review of third-party solutions and then choosing a solution provider could take several months. However, even then one has only just started on the journey to overall infrastructure readiness.

Business Process Readiness

The purpose of this track is to assess business and technical resources, legacy applications, and business process repositories. This is important because each RosettaNet PIP provides a mutually agreed supply-chain view of key business processes. Key criteria in this track include understanding the RosettaNet message structure and process message sequencing, identifying business process architects, and defining new business processes.

Intel is completing its initial B2B infrastructure planning and design requirements gathering, which coincides with initial RosettaNet pilots. A key finding from our initial efforts is that a significant increase in participation by "business analysts" (one of two areas in which we had difficulty obtaining resources) is needed in order to forward engineer and plan for the anticipated levels of adoption.

We have also realized that process re-engineering for RosettaNet must be performed within the context of re-engineering enterprise at-large business processes that

include requirements for trading partner integration. Therefore, our RosettaNet effort is considered an integral part of our EAI initiative. RosettaNet is therefore one element of a strategy to become a real-time event-driven global enterprise. Process re-engineering is a huge task.

To appreciate why business process readiness is such a big task, we need to understand how constructing a distributed Internet application using a robust message structure with a rich meta-model impacts enterprise readiness.

A RosettaNet message is intended to be predictable (open standards-based format), somewhat human readable, and portable between trading partners. In order to produce a widely supported and long-lived message format, the RosettaNet consortium agreed to define a message structure incorporating a complete data and meta-data model common to the significant business processes within the IT and EC supply chains.

A RosettaNet message consists of several nested XML structures and data structures, namely,

1. nested XML envelopes to define action, transaction, service, agent, message, transfer, and security sections
2. XML message sections for preamble, header, and body
3. attributes expressed using XML tags based on a supply chain dictionary
4. meta-data schema structures expressed using XML document type definitions (DTDs) or XML schema consisting of attribute data type definitions, tag hierarchy, cardinality (1:1, 1:n), permissible values, and parent/child dependencies
5. data as message content

Therefore, this message was deliberately designed as a self-contained, stateful and intelligent message, complete with data, persistent state information, and a meta-data model. Conceptually, it could be used to populate an object class or produce a database structure. Moreover, it could be abstractly considered as a snapshot of a transactional sequence in a file-based database expressed using XML.

It is therefore important to recognize that a RosettaNet message contains more information than data alone. It is a rich, fully stateful, self-describing package of information.

A RosettaNet message does not include any implied, hard-coded positional, or delimited structures. On the contrary, other formats for message and document exchange (namely EDI and non-standard comma-separated values (CSV) or tab-delimited file formats) provide a lesser degree or no level of schema definition, data constraints, dictionary-driven taxonomies, and process state information.

The completeness of a RosettaNet message structure across a supply chain (as defined in PIPs) requires significant forward engineering by trading partners within the RosettaNet consortium. As a result, trading partners should expect to re-engineer their back-end systems to become RosettaNet compliant. This may involve creating processes that currently don't exist internally or mapping processes that are currently different from RosettaNet processes.

Up-front business process architects need to participate in many activities:

1. RosettaNet PIP workshops to define each process, meta-model schema, dictionary, taxonomy, message sequencing, and run-time parameters (e.g., wait times, retry duration, acknowledgements)
2. determining impact upon existing business processes and existing applications
3. optimizing existing business processes by leveraging the capabilities provided by RosettaNet within the context of an at-large enterprise process re-engineering effort
4. determining new processes and data services

Application Development Readiness

The purpose of this track is to prepare PIP implementation development plans and roadmaps. This is important because this step represents how and when existing processes and systems will be modified and rolled out to support RosettaNet. Key criteria in this track include statements of work, budgets, and plans.

As in the Business Process Readiness track, substantial participation by application development group(s) is necessary to forward engineer and plan for the anticipated levels of adoption. Application development groups realize they need to re-engineer processes for RosettaNet within the context of re-engineering enterprise at-large business processes while at the same time including requirements for trading partner integration.

Key deliverables for this track include work scope; identification of impacted systems; identification of key business analysts and process architects; determination of RosettaNet compatibility with existing processes; preparation of project budgets and schedules; setting of release dates; provision of consolidated test

requirements; definition of necessary API components; and setting of incremental upgrade roadmaps.

This track is similar to most enterprise application development efforts and can use a variety of development methodologies (e.g., traditional waterfall, rapid application development (RAD), etc). This track, more than any other, is likely to require the greatest amount of effort and resources. What's important to understand is that this group is usually the last to participate in the RosettaNet implementation planning effort, yet it has to be the first to implement the plan in order for deployment to progress. Therefore, getting up-front participation from the application developers is mandatory.

B2B EXTERNAL INITIATIVE READINESS

The purpose of this track is to assess the completeness and usability of the work of the chosen B2B external initiative (in our case, RosettaNet). Specifications, policies, and architectures provided by the initiative must be understood and evaluated against internal policies, procedures, guidelines, and strategies. This is important because implementing RosettaNet is not "only" a technology; it is part of a strategy that must permeate an enterprise's trading partner integration strategy. Key criteria in this track include review of consortium supply chain, implementation framework, and process frameworks.

Each B2B initiative provides technical specifications that present the functional design and technical frameworks for message structure, message transport, and/or message content. In the case of RosettaNet, many technical documents and specifications have been written. For example, below is a collection of guidelines and specifications that are necessary in our implementation of the "Manage Purchase Order" PIP (which covers submit, acknowledge, change, and cancel purchase orders). This material addresses one of approximately 100 PIPs.

1. RosettaNet Implementation Framework v1.1
2. Manage Purchase Order Specification (3A4)
3. 3A4 Purchase Order Acceptance Message Guideline
4. 3A4 Purchase Order Acceptance Guideline DTD
5. 3A4 Purchase Order Cancellation Message Guideline
6. 3A4 Purchase Order Cancellation Guideline DTD
7. 3A4 Purchase Order Change Message Guideline
8. 3A4 Purchase Order Change Guideline DTD
9. 3A4 Purchase Order Request Message Guideline
10. 3A4 Purchase Order Request Guideline DTD
11. Preamble Part Message Guideline
12. Preamble Guideline DTD
13. Service Header Part Message Guideline
14. Service Header Guideline DTD
15. Acceptance Acknowledgement Message Guideline
16. Acceptance Acknowledgement Guideline DTD
17. Acceptance Acknowledgement Exception Message Guideline
18. Acceptance Acknowledgement Exception Guideline DTD
19. Receipt Acknowledgement Message Guideline
20. Receipt Acknowledgement Guideline DTD
21. Receipt Acknowledgement Exception Message Guideline
22. Receipt Acknowledgement Exception Guideline DTD
23. General Exception Guideline DTD
24. General Exception Message Guideline

A given consortium's documentation is usually targeted to a specific supply chain or e-Business market segment. The consortium's pervasiveness within its target markets must be considered. Moreover, due to the relative youth of Internet e-Business, frameworks and specifications may not be as complete or thorough as they could be. Therefore, participation in and achieving time-tested experience within the initiative enables trading partners to more accurately assess the applicability of the initiative to their businesses, as well as providing a means for influencing the initiative such that it *does* deliver the needed benefits. Finally, adopting a B2B framework needs to include a review of its compatibility with best known methods (BKMs) within one's company.

Trading Partner Readiness

The purpose of this track is to assess the readiness of key trading partners. This is important because one cannot implement RosettaNet without at least one and hopefully many trading partners ready to do so. Key criteria in this track include selecting trading partners,

choosing processes, detailed integration, and achieving reliable results.

Each trading partner will need to provide a similar level of effort. It will be several years until the B2B trading partner automation technologies have matured to provide relatively inexpensive plug and play solutions; therefore, these next few years will only include trading partners who consider themselves early adopters. Trading partners must have the will and desire to deliberately re-engineer business processes based upon a rapid schedule and evolving processes. They must be able to move quickly, often with ad hoc funding and scavenging for equipment and resources. Although management commitment is essential to successful implementation of a RosettaNet-sized initiative, a skunk-works and entrepreneurial mentality in the early days can be helpful.

Selecting a RosettaNet trading partner is currently easy because only early adopters are playing; and, with a limited set of PIPs to choose from, it is easy to define a project. A key expectation is that the use of RosettaNet specifications will eliminate the currently high level of up-front trading partner analysis needed to conduct e-Business. This may lead to a rush of trading partners wishing to engage each other using RosettaNet processes (after initial successful implementations by early adopters) before the PIPs have matured and PIP implementation is a widely understood experience. At present, early adopter trading partners spend significant effort figuring out how to use the RosettaNet specifications with one another. Once sufficient infrastructure is in place, the full benefits of RosettaNet can be realized as trading partners self-administer their processes and subscriptions.

Currently, readiness must be planned with exact testing and production dates and known versions of specifications and guidelines. Legal issues need to be negotiated up front (see "Legal Readiness" below). Precise details of Global Trade Identification Number (GTIN), United Nations Standard Products and Services Classification (UN/SPSC), and Dun & Bradstreet-assigned unique corporate identifier (D-U-N-S*) must be managed. Personalized trade parameters such as part number, product lines, and interpretations of timeouts, retry and acknowledgements need to be exactly discussed. Trading partner agreements (TPAs) need to be signed. Current EDI processes with the trading partner may need to be changed. Digital certificates and digital signatures will be needed. And, as always in a new venture, backup plans will be needed.

Solution Provider Readiness

The purpose of this track is to assess the readiness of your selected B2B gateway solution provider. This is important because the tool you have selected may not provide all the capabilities needed to implement a PIP with trading partners. Key criteria in this track include review of public and private PIP processes, review of PIP templates, and concurrence of PIP interpretation.

Some solution providers provide only the plumbing to enable RosettaNet. When no PIP templates are provided, the end user must provide all aspects of PIP implementation. In these cases the tool is ignorant of the exact meaning of retry periods, duplicate messages, acknowledgements, failure to receive, and other process specifics. These build-your-own solutions will require internal infrastructure for non-repudiation database (NRdb), trading partner database (TPdb) and PKI.

Other solution providers provide a robust framework for PIP implementation where the PIP template is quite cognizant of the PIP framework. PIP implementation would be easier and faster using these tools; however, the tool must be sufficiently flexible should the PIP framework prove incomplete in any given trading scenario. These all-encompassing solutions include infrastructure for NRdb, TPdb, and PKI.

Trading partners need to assess the capabilities of their solution provider(s). Some key questions include the following. What level of compliance does the tool provide for the implementation framework and process specifications? When is beta and general availability? Has the tool been sufficiently stress-tested for a variety of PIP scenarios? Does the tool provide diverse role-based control so different groups cannot access other groups' processes? Many other questions will be on the minds of individual trading partners.

Finally, RosettaNet is working on a Solution Provider Certification program and certification standards, which should help RosettaNet implementors perform their assessments more quickly and with greater assurance.

Legal Readiness

The purpose of this track is to assess relevant legal issues. This is important because RosettaNet has expanded the capabilities of trading partner integration beyond the current terms and conditions found with EDI agreements; therefore, legal precedence has not yet been established for RosettaNet interactions. Key criteria in this track include trading partner agreements and early participation by legal counsel.

* Third-party brands and names are the property of their respective owners.

Performing RosettaNet message exchange with trading partners will require a trading partner agreement (TPA) between each pair of trading partners. These legal agreements need to be managed by each company's legal counsel. TPAs currently exist for EDI; however, a generalized RosettaNet TPA does not exist as of this writing (although creation of a model TPA is now underway). In addition, legal expertise for Internet-based e-Business using RosettaNet has not yet been attained. Experience gained in RosettaNet pilot programs will help legal counsel to understand the differences between RosettaNet and EDI and facilitate the preparation of a comprehensive TPA.

Because RosettaNet will be enabling supply chain automation across a lengthy chain of buyers (customers) and sellers (suppliers), the goal is to write the TPA from a neutral perspective. Use of such a neutral TPA may be a challenge for many companies, whose organizational practices may have dictated that they prescribe different terms and conditions within their EDI TPAs depending upon their role as buyer or seller.

The list of legal concerns is being compiled as we move forward. Although many issues have been identified, the full impact will likely not be comprehended until the infrastructure is in place and more time is spent in understanding legal ramifications. To date, some of these issues are

- encryption export to controlled countries
- frequently changing e-Commerce and e-Business legislation
- strict privacy laws
- the potential for hundreds of trading partners with varying capabilities
- restriction on use of confidential, proprietary, or trade secret information
- constantly changing landscape of trading partners, processes, messages, documents
- personalized TPAs with specific and different run-time parameters
- self-administered processes and subscriptions
- proper use of digital certificate and signatures for the accompanying document/message
- signed non-disclosure and confidentiality agreements

Security Readiness

The purpose of this track is to assess the security requirements for encryption, authentication, and authorization at both the network and the trading partner message exchange level. This is important because implementing RosettaNet means that trading partner systems penetrate their corporate external firewall and security mechanisms. And undoubtedly, most data will need to pass through the internal firewalls to core enterprise applications. RosettaNet also will enable trading partners of different types and privileges to exchange documents for many critical business processes (e.g., purchase order, quotes, product information, pricing, availability, inventory, technical specifications, trade secret and confidential documents, CAD drawings, design specifications, etc.). Key criteria in this track include an understanding of corporate security and document confidentiality policies; and encryption, authentication, and authorization.

Security needs to initially address the front-end and the back-end. Front-end security issues apply to firewalls, proxy servers, network routing and protecting the system from malicious attacks. Back-end security issues apply to the controlled access to message content to system users and intermediaries using a right-to-see approach. Unlike current point-to-point solutions where data handling is decentralized, a B2B gateway will provide for a centralized flow of critical business information; therefore, only users with the right to see specific data should be entitled. Role-based administration of the B2B gateway should be considered.

Security readiness is also a significant challenge due to the inherent solution complexities, need for managed risk, and elevated concerns. The RosettaNet implementation framework incorporates a public key infrastructure (PKI). Intel's current RosettaNet implementation is based on a single corporate guideline using multiple certificate authorities, digital certificates, and digital signatures. Intel also requires the use of 128-bit encryption, which is greater than common usage and also is prohibited for export to controlled countries. Obtaining, understanding, and incorporating these guidelines and technologies into the B2B gateway, although logically simple, has been technically difficult due to the inherent complexity of PKI.

An important aspect of security is the ability to immediately revoke the privileges of a trading partner, or of any of their processes or subscriptions. It is also important to be able to confirm that trading partners are sending messages as agreed. This includes being able to detect when a message was not correctly assembled and

transported according to the TPA in place between the trading partners. This also includes the ability to detect whether encryption, digital certificates and digital signatures were correctly used.

Audit Readiness

The purpose of this track is to assess one's readiness to be audited by internal company officials. This readiness is important because RosettaNet trading enables the interaction of critical business processes. Managers and executives should not be casual with their views of implementing RosettaNet. Key criteria in this track include understanding the seriousness of global electronic trading, and preparing for audits.

Knowing that a B2B gateway will eventually transport and manage a majority of e-Commerce transactions and e-Content interactions with trading partners, it is important to design the B2B infrastructure up front to withstand frequent and diverse auditing.

By design, RosettaNet and the capabilities it enables represent considerable risks to a company should something go wrong. Auditing is actually a good thing, as one should feel more assured that risks are under control. Some of the risks identified include

- potential to be majority revenue channel
- binding $M transactions
- binding legal agreements
- international trade with an easy global reach
- rapidly changing trade and Internet laws
- many government enforcement authorities
- sensitive and confidential document/information exchange
- many micro projects with intangible ROI where something will be unforeseen
- potential for lost potential or mistakes
- needs to be fault tolerant without data loss
- many critical success factors
- pivotal and timely information exchanges
- potential for significant impact on internal systems
- significant visibility and expectation levels
- competitors waiting for your misstep!
- centralized administration of enterprise processes and data (need for role-based administration and management using a limited right-to-see basis)

CHALLENGES

Achieving a common language for e-Business offers challenges in a number of areas, including (but not limited to) the development of the specifications themselves; correctly identifying the internal barriers to success and successfully overcoming them; and planning to keep up with an ever-changing business, technical, and standards environment.

Some of the specific challenges we see ahead include

1. *Internet Speed*. RosettaNet* is caught up in the frenzy of Internet time. As such, trading partner automation and XML messaging are very hot technologies; the leaders in this race will likely reap the greatest rewards. Most significantly, getting it done faster, better, and cheaper will remain a requirement that cannot be understated. Many challenges exist when trying to compress and accelerate planning, funding, scheduling, evangelizing, designing, building, and testing, especially when considering the Readiness Model presented above.
2. *Sustaining will (internally).* Maintaining momentum in the face of "short attention spans" seems to be a systemic symptom of today's Internet e-Business mentality. At an increasing rate, everyone seems to have less time to make informed decisions. An increased level of risk-taking will be necessary to proceed; management needs to remain committed even when the inevitable mistakes are made.
3. *Sustaining will (externally).* Early adopters of RosettaNet will find it neither easy nor inexpensive to initially embrace. Each supply-chain or endorsing adopter will face an inevitable debate of whether to continue or disengage. So far the will of the RosettaNet consortium is withstanding these stresses and the key motives for moving forward remain steadfast; however, further tests of will are likely before RosettaNet's adoption is widespread.
4. *Obtaining resources*. Planning in advance for resource needs is a challenge within any company; however, RosettaNet, like all e-Business initiatives, is driven by its constituents faster than any company could anticipate. Obtaining business and technical resources is a challenge; however, expanding to include sufficient forward-thinking resources from business analysts, technical analysts, system architects, and application architects requires resource allocation. This can be achieved either by additional funding or by cancelling other planned projects. This can be especially challenging if resources are being pulled from competing B2B initiatives.

* Third-party brands and names are the property of their respective owners.

5. *Creating the implementation plan*. Defining, planning, and estimating the scope of work to implement which of the ~100 RosettaNet PIPs across the enterprise at-large requires a diverse group of resources and a PIP-centric approach rather than a business group approach.

6. *Choosing a project management methodology*. RosettaNet implementation needs to be executed using a hybrid of rapid application development (RAD) project methodology. Determining a methodology could be challenging within companies that do not have a conscious process for selecting a methodology.

7. *Finding an optimal team structure*. Initial implementations of RosettaNet require participation from diverse groups within an enterprise (exact composition depends heavily on the PIPs chosen for implementation). Each PIP implementation becomes a mini-project within the bigger context of RosettaNet and B2B implementation. Maximizing team productivity and effectiveness is essential, especially considering that B2B and e-Business projects need to proceed at Internet speed. It will be challenging to form an optimal team structure, then clone it for the many PIP mini-projects.

8. *Managing information overload*. Implementing any enterprise-wide project (especially one which happens to affect the very way the enterprise conducts its business) is hugely complex and involves a tremendous amount of information assimilation. Implementing the same set of specifications across most of the members of an industry magnifies the problem of synchronized information assimilation enormously. Participants in the implementation process must remain current with respect to RosettaNet specifications; each of the open standards on which RosettaNet is based (e.g., XML, SSL, HTTP); software and hardware solution options; internal company guidelines; requirements and functional specifications; test plans; meeting minutes; and other common materials. Participants must also keep abreast of similar materials from trading partners with whom they are implementing the plan. Methods for assimilating and managing frequent knowledge and information change in the e-Business sphere are sadly lacking.

RESULTS

On a practical level, we have identified eight distinct roles within our B2B RosettaNet* deployment strategy. Table 1 lists these eight roles; it also shows the level of participation of each of these players within the readiness tracks discussed above. (As a point of departure for readers, the staffing levels for each role as we worked through to our 2.2.2000 deployment plans was as follows. One person each fulfilled roles 1 through 5. Role 6 consisted (in our case) of one full-time person plus parts of numerous other folks participating in PIP workshops, for another full-time equivalent. Multiple people participated for roles 7 and 8, typically one person for specific groups of PIPs or core applications. A total of 22 people participated for 2.2.2000 -- 13 from IT and 9 from the business units)

Intel performed several key tactical steps to address the diverse issues within the Readiness Model.

First we assembled the Intel RosettaNet Deployment Team consisting of six people in roles 1 through 6 in Table 1. We were slow in getting participants for roles 7 and 8 because these groups were extremely busy and up-front resource planning was required. In hindsight we recommend engaging these business analysts and application development groups in the early stages of RosettaNet planning.

Next, we engaged one trading partner (a major distributor) as part of our RosettaNet proof-of-concept pilot (August 1999) and initial implementation (2.2.2000). With our trading partner, we selected PIP3A4 ("Order Management"). Each of us selected our own solution provider and tools. This meant that four companies had to synchronize development and test plans. Since we were all first implementers, gaps and changes in the RosettaNet Implementation Framework and PIP guidelines needed to be ironed out. Infrastructure planning was a key focus from the beginning. Engineers within the core environment supporting EDI and our e-Business engineering groups worked together to integrate e-Business design requirements with existing EDI requirements. At present, we are creating a production environment that supports both EDI and RosettaNet running on Windows NT*.

* Third-party brands and names are the property of their respective owners.

* Other brands and names are the property of their respective owners.

After a few months of assessing RosettaNet readiness and formulating the Readiness Model, we prepared and sent a PIP assessment and business impact survey to all business groups having a need for trading partner automation. We are now waiting for enterprise-wide responses. These responses, and additional partner readiness discussions, will be reviewed and become the basis for our post-2.2.2000 rollout.

The following recommendations are provided to assist with first-time RosettaNet deployment:

- Look for a quick win: pick one strategic PIP with one partner. Plan for the process to take 2-4 months. Assign 4 to 6 people.
- Engage the solution providers, letting them educate you on B2B and partner integration architectures. Perhaps even contract with one of them to build a limited production pilot. Defer committing to your B2B vendor until a successful pilot is in production.
- Require the Business Manager and Technical Manager to hold weekly meetings to review progress and status.
- Incorporate the RosettaNet roadmap strategy within the company's overall B2B strategy.
- Include other B2B channels within the scope of the B2B gateway (e.g., secure file transfer, SMTP, EDI).
- Consider the impact on existing browser-based applications and partner portal strategies.

Table 1 :Participation levels of key roles in readiness model tracks

Role #	Description	Readiness Track									
		1 Biz Strat	2 Infra-structure	3 Biz Process	4 App Dev	5 B2B Initiative	6 Trading Partner	7 Solution Provider	8 Legal	9 Security	10 Audit
1	RosettaNet Business Program Management	**L**	M	**S**	S	M	**S**	M	M	M	M
2	RosettaNet Technical Program Management	M	L	S	S	S	S	L	L	L	L
3	PIP Management	S	M	S	S	S	S	M	M	M	M
4	Pilot Management	S	S	S	S	S	L	S	S	S	M
5	Application Integration Management	M	S	S	L	S	S	S	M	S	S
6	RosettaNet Standards-Development	M	M	S	M	L	M	M	M	S	N
7	Technical and Business Analysts, Business Process Analysts	S	M	L	S	S	S	M	M	S	S
8	Back-end Application Development Management	M	M	S	S	N	M	N	N	M	S

Legend: L = Leader S= Significant Participation M = modest participation N = little to no participation

CONCLUSION

Our team continually expands its understanding of what it takes to implement RosettaNet. As we complete a second-phase pilot, plan for future implementations, design the infrastructure, and expand our circle of influence, we foresee many new challenges. It is unclear when the rate of discovery of new issues and challenges will diminish. It is likely not to be until widespread trading partner/PIP implementation occurs in several years.

ACKNOWLEDGMENTS

Special thanks are due to our fellow members of Intel's enterprise-wide RosettaNet Deployment Team and to the architects, strategists, and technologists working on e-Business for their frequent business, strategic, and technical contributions to our understanding of issues

related to the implementation of RosettaNet and B2B trading partner automation capabilities.

Intel RosettaNet Deployment Team Members: Colin Evans (SMG IM&E Director, eBusiness Architecture and Operations), Alan Court (SMG IM&E, RosettaNet Program Manager); Andy Keates (SMG IM&E RosettaNet Pilot (e-Concert) Manager); Thomas Vlach (PLG, RosettaNet PIP Analyst).

IT Architects, Strategists and Technologists: Bert Cave (IT Engineering, EAI Program Manager), Ralph Nitta (IT e-Business Integration Manager), Ed Balthasar (IT Strategy and Technology).

REFERENCES

[1] RosettaNet specifications (www.rosettanet.org).

AUTHORS' BIOGRAPHIES

Patricia J. O'Sullivan is a staff architect in IT Strategy & Technology. Her current focus is on increasing productivity in the use of information, currently as applied in e-Business standards. As part of this focus, she leads the RosettaNet's Implementation Framework team. She holds a masters degree in library and information science. Her e-mail is patricia.j.osullivan@intel.com

Don S. Whitecar, PE, is IT's e-Business EAI/B2B and RosettaNet Program Manager. His goal is to provide a robust, scaleable, and secure infrastructure for B2B trading partner automation. Don is also responsible for Intel's technical implementation of RosettaNet, working closely with business groups, business analysts, and application architects. His e-mail is: don.s.whitecar@intel.com.