

Intel® e-Business Engineering Release Management and Application Landing

Alan Hodgson, IT e-Business Integration Engineering, Intel Corp.

Index words: release, environments, engineering, prioritization, testing, milestones, automation

ABSTRACT

To meet the rapid growth in the number of e-Business applications to be deployed and the demand for increasing the frequency for application releases, Intel's e-Business release cycle has gone through many evolutions and has seen many improvements. This paper examines these changes from an IT e-Business Integration Engineering perspective. The e-Business release cycle may range from 6 weeks to 20 weeks depending on the functional and technical complexity of the release. A release may include up to 6 applications and in general, each application releases 4 or more times a year. There are over 65 core e-Business external-facing applications.

In this paper, we focus mostly on Business-to-Business (B2B) direct sales and marketing, the most active e-Business area, and we deal only with the e-Business Integration Engineering group's role in e-Business release planning and management.

INTRODUCTION

Since 1998, the Intel® IT e-Business Integration Engineering organization has grown from a small group of 25 engineers supporting 50 servers running a handful of external-facing Internet applications to 80+ engineers supporting approximately 850 servers running 65 external-facing Internet applications. This server farm is split between pre-production, which in general covers development, QA test, and stress test servers; and production, which includes production, failover, and production support servers. The challenge has been to provide on-time, reliable engineering activities to support the rapidly increasing number of applications being released in the external-facing, Intel® e-Business space and to keep pace with the increasing application release frequency.

The Intel IT e-Business Integration Engineering group has progressed, out of necessity, from adhoc releases requested with minimal lead time by the application groups to a very close partnership with the business application groups that includes early engagement and notification of release priorities and plans. Out of this partnership has evolved a comprehensive release process that incorporates planning and prioritization, resource

and environment management, engineering management and release management.

Intel is dedicated to delivering new and improved functionality to its customers in a timeframe that keeps Intel and its customers competitive in the e-Business market place. To ensure reliability and speed in the release process, Intel's IT e-Business Integration Engineering group is in the process of automating many of the engineering build activities. This has reduced environment build times by up to 50% and increased accuracy and reliability. Automation is continuing with all aspects of instance designs and code migrations.

The engineering activities vary with the technical complexity of the release, but in general include instance design activities, server build activities, stress testing, day 1 testing (a rehearsal of the deployment activities), and deployment of the application into production. A complex release may also include purchasing and assembling new hardware and possibly the development and implementation of new reference designs. These engineering activities are now closely managed and aligned with the application development lifecycle activities to ensure that development, testing, and deployment milestones can be met. Daily management of release milestones has become a necessity for meeting release deadlines.

This paper covers a broad range of issues and activities relating to the management of the release and deployment of an external-facing (where Internet communication passes through a firewall) e-Business application at Intel. It is written from an e-Business integration engineering perspective and looks at the Intel e-Business release process with respect to how the e-Business Integration Engineering and the e-Business Application Development teams handle the release process for over 60 applications a year.

Intel's e-Business applications fall into one of the following business areas:

- B2B Sales and Marketing Group (SMG)
- B2B Indirect Channel
- B2C (Business to Consumer)
- B2C Core Services (e.g., Search, Registration, etc.)

- B2B Supplier
- Other non-mission-critical e-Business applications

In most cases this paper refers to the release process for the most active and mission-critical e-Business area, Intel's B2B Sales and Marketing Group.

The Engagement Process

For new applications and major enhancements to existing applications there are two phases of customer engagement with the e-Business Integration Engineering group. The first phase, shown in Figure 1, deals with finding strategic responses to business initiatives and involves participation in the Architecture Design Group (ADG). In this phase, the e-Business Integration Engineering group provides expertise in infrastructure and architecture to help find the best strategic response. This may include piloting and validating a recommendation before sanctioning its acceptance.

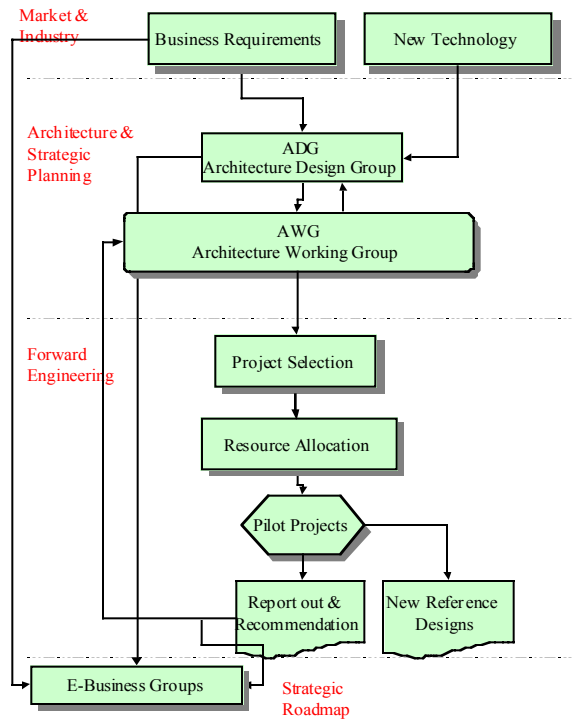


Figure 1: Initiative study and strategic architecture design

The second phase, shown in Figure 2, occurs when a project has been funded and is ready to be prioritized in the release process. In this phase, the technical complexity of the project is assessed, resources are assigned, and there is participation in the design, development, and deployment of the application.

This paper focuses on the second phase of the engagement process dealing with e-Business release planning and management.

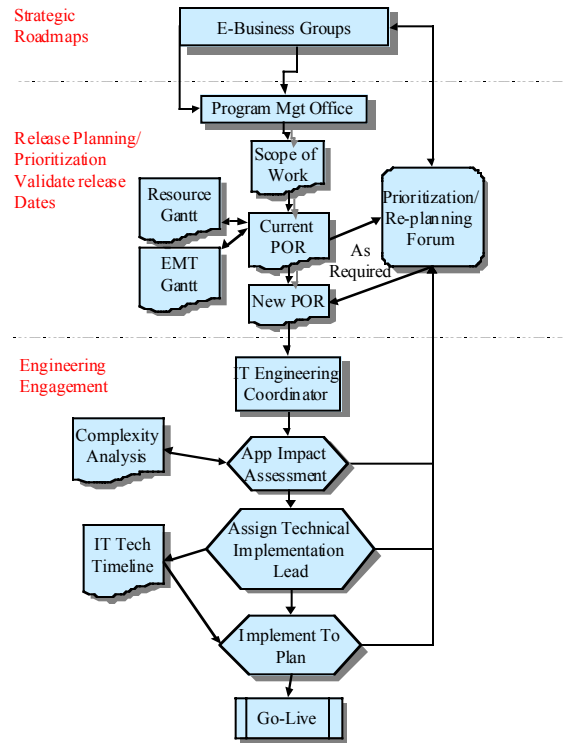


Figure 2: Release planning/management

CHALLENGES

As the number of e-Business applications increased and the need to release more functionality more frequently, there was an ever-increasing demand for hardware and engineering resources. This demand often outpaced the hardware and resources available. Environments and servers were being shared among applications and needed to be constantly rebuilt for the next release. Releases would follow so closely upon one another that a missed release date could very well have an impact on the date of the next release. Engagement with the IT e-Business integration engineers would happen towards the end of the application design phase leaving very little time for engineering to get acquainted with, and prepare for, the new technology requirements. In many cases, new hardware needed to be purchased and new reference designs developed. In addition, there was little or no business prioritization of the applications and associated releases. All this resulted in unrealistic expectations from the users that a release could always be resourced and delivered on time. From an engineering standpoint this was a no win situation and resulted in overloaded resources and missed schedules. This paper addresses the release planning and

management processes that were developed and implemented to change this scenario yet support this fast-paced dynamic environment.

ENGINEERING RESPONSIBILITIES

First, let's take a look at the Intel IT e-Business Integration Engineering organization and its role in the release process. Support for our release cycle environments requires a team of engineers who can plan, design, build, test, and maintain the environments. These engineering activities are in the critical path of the release and are dependent on certain development activities being completed on time. Likewise, the QA test teams are dependent on the e-Business integration engineering activities being completed on time. The Intel e-Business Integration Engineering group has categorized these activities as follows:

- Design Engineering
- Factory Engineering
- Pre-Production Engineering
- Production Engineering

Design Engineering includes those activities relating to adding or changing infrastructure designs. These design additions or changes may be as a result of new software, new hardware, or new releases of operating systems and databases. Design engineering may include designing new infrastructure architecture, prototyping and testing the changes, and ultimately developing reference designs that can be used to consistently set up and install the new hardware, software, or release. These reference designs are used by other engineers to build servers for new application releases.

Factory Engineering includes those activities relating to ensuring a successful deployment of the application release from an engineering perspective. These activities include developing and managing the engineering project plan and designing and documenting the builds for all servers in the environment. These design documents are called Instance Designs (often referred to as branding documents), and they provide the complete specification for building and configuring a server. They include, in part, reference designs for the

base builds of the OS and databases, but in addition include directory structures, application installation and configuration instructions, and Common Object Model (COM) specifications. Factory engineering requires the engineer to be closely involved with the application development team to understand the new application's technical designs. The engineer should be a partner in the design solutions. A factory engineer will, in most cases, use the build of the proto environment as an engineering prototyping opportunity to test out and develop the instance design documentation. This documentation will ultimately be passed on to the pre-production engineer and the production engineer to build and maintain the testing and production servers.

Pre-Production Engineering includes those activities relating to the building and maintenance of servers in the pre-production environments. These engineers are expected to build from the Instance Design documentation to ensure consistent and reliable server builds. In the release cycle, these engineers are most impacted by any slippage in application design and development activities, since they are still expected to deliver test environments to the QA and Stress test teams on time.

Production Engineering includes those activities relating to building the production servers, deploying an application release into production, and providing adequate production support for the application infrastructure. Production engineering is also a primary recipient of the Instance Designs developed by the factory engineer and validated by the pre-production engineer. A production engineer is also a primary participant in building the stress and day 1 environments and performing the day 1 execution. The day 1 execution is in effect a dress rehearsal of the deployment of a release into production and usually takes place the week prior to production deployment. There is an additional burden on the production engineer to voice concern if the application does not meet the entrance criteria for production stability and performance. Once a deployment has taken place the production engineer becomes responsible for on-call support, production environment maintenance, data center management, and troubleshooting of production issues.

Server Name	PROD	Stress 1	Stress 2	Stress 3	Stress 4	Stress 5	Stress 6	QA	PROTO
CYAN	S26	NA	S26	S26	S26	S26	S26	S26-RM	S26-RM
TEAL	S26	NA	S26	S26	S26	S26	S26	HP	S26-2
PLUM	S26	NA	S26	S26	S26	S26	S26	HP	S26-RM
ROSE	S26	NA	S26	S26	S26	S26	S26	HP	S26-RM
IVORY	S26	NA	S26	S26	S26	S26	S26	HP	S26
DIAMOND	S26	NA	S26	S26	S26	S26	S26	HP2	S26
EMERALD	S26	NA	S26	S26	S26	S26	S26	HP	S26-2
OCEANIC	S26	NA	S26	-	PC-360	S26	S26	S26	PC-350
CPSBATCH01	S26	NA	PC360	PC360	PC-360	S26	S26	HP	S26-2
CPSAMS01	4300	NA	4300	4300	4300	S50	S50	HP2	S26
JFSCODR	S26	NA	S26						
JFSIBL01-ORCL	4300	NA	S50-1	4300	S50	S50	S50	S26-RM	S26-RM
SW	S26	NA	S26	S26	S26	S26	S26		
ECLIPSE	S50	NA	S50	S50	S50			S26-RM	S26-RM
CAMELOT	S50	NA	S50	S50	S50			S26/S50	S26-RM
ECITRDB	S26	NA			S26			Consolidated on QADB	
PYRITE	S26	NA			S26			S26	
KALPLANA	S26	NA			S26			S26	
CELTIC	S26	NA	S26-RM	S26-RM	S26-RM			S26	S26-RM
SORRENTO	S26	NA							
CPSUSI	S26	NA	S26	S26	S26				
MIDAS	4400 (2) 4300 (1)	NA	4300					4300	
MIDASD2W	S26	NA	S26-RM					S26-RM	S26-RM
IBPDB1	4300	NA							S50
GREEN	S26	NA						S26-RM	S26-RM

KEY: Landed/Online To be Ordered Requires Upgrade
 Received/Not installed Order Approved: ESD: Estimated Shipping Date

Figure 3: Sample pre-production environment server map

RELEASE ENVIRONMENTS

Each functionally dependent set of applications is developed, tested, and deployed on an infrastructure of servers that has been designed and configured to support the various applications. This infrastructure typically consists of groups of servers forming environments that are dedicated to phases of the release cycle.

- A development environment consists of a set of development servers used by developers to develop and test their applications.
- A proto environment consists of servers used to prototype or test out the designs.
- The pre-production QA environment consists of servers used by an independent QA test team to perform functional and integrated testing.

- The pre-production stress testing and day 1 testing environment is the most dynamic environment. It actually comprises up to five environments called stations, which are built on an as-needed basis to stress test applications within a release. An environment station may be used for stress testing Web Order Management one week and rebuilt to stress test a channel application the next week.
- The production servers host the application in production and typically include a redundant set of fail-over servers. In addition there may also be a production support environment where production bug fixes can be made and tested without impacting a new release that is being tested in the pre-production environment.

The complete set of phased environments, including production, is called a pipe. A pipe is, in most cases, specific and dedicated to an application area, (e.g., B2B sales and marketing). A phased environment typically consists of one or more web, database, and application servers. The numbers will vary depending on how close the phase is to production. The QA test environments will be more representative of the production architecture than the development environment and will include back-end system environments that make up the integration architecture. At Intel, the e-Business, B2B Direct SMG pipe requires between 20 to 26 servers in each of the test environments and around 50 servers in production, including failover. The pipe consists of around 140 servers that need to be maintained and supported for a release. See Figure 3.

The new or changed application is moved from environment to environment as development is completed and testing begins. At each stage, entrance criteria need to be met, as shown in Figure 4, before the code can be migrated to the next environment.

E-Business Pipeline 'Promote to Production':
Environment Entrance Criteria

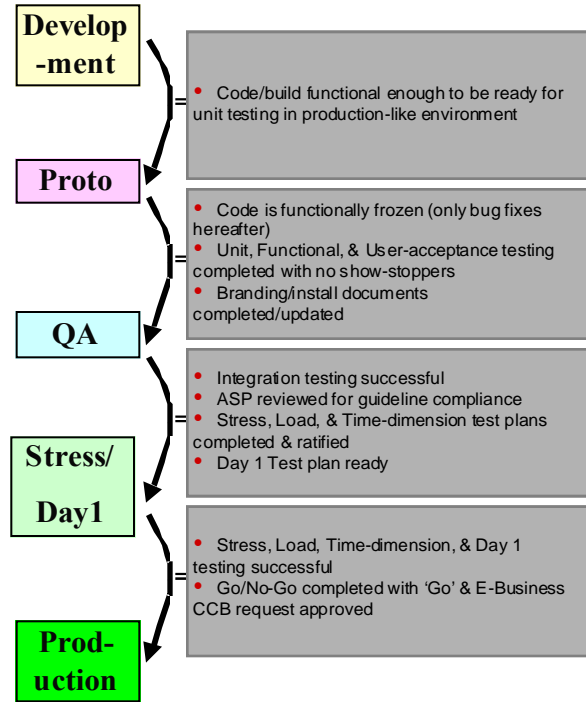


Figure 4: Environment migration and exit criteria

The Release Cycle

The release cycle for an e-Business application consists of the standard phases of planning, requirements, design, development, testing, and deployment. However, unlike the large back-end systems that often span years of development, an e-Business release takes anywhere from six months to six weeks depending on its functional and technical complexity.

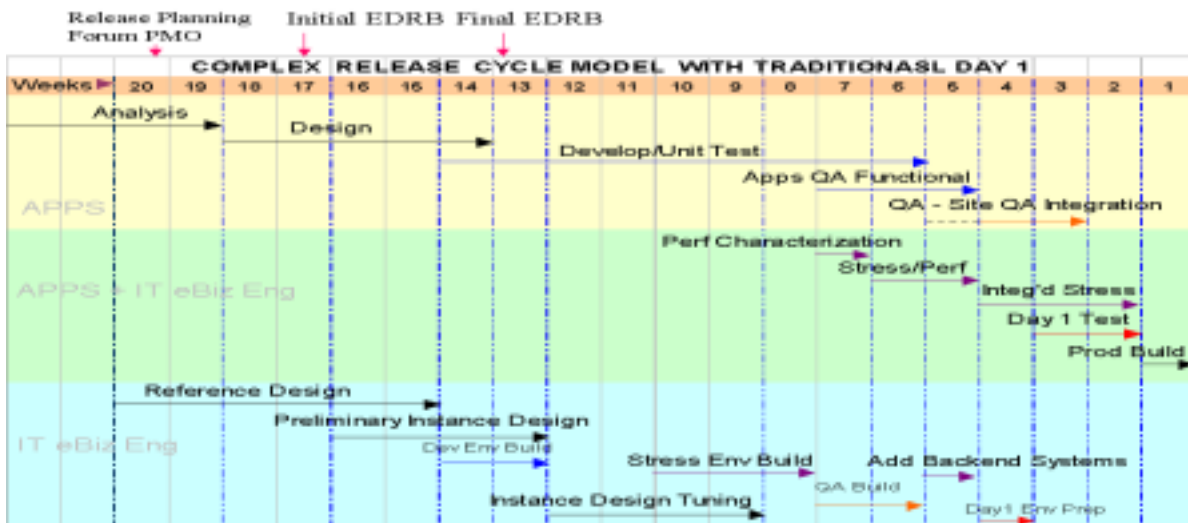


Figure 5: Complex release model

The Complex Release Model shown in Figure 5 reflects significant infrastructure changes that would include new servers and associated reference and instance designs.

A Medium Complex Release Model would not necessarily require new equipment and reference designs, but would require significant changes to

instance designs with corresponding build and test activities.

The Simple Release Model shown in Figure 6 may require significant testing of functionality but would require very little engineering. In this model, the environments stay dedicated to the application and do not need to be rebuilt or changed.

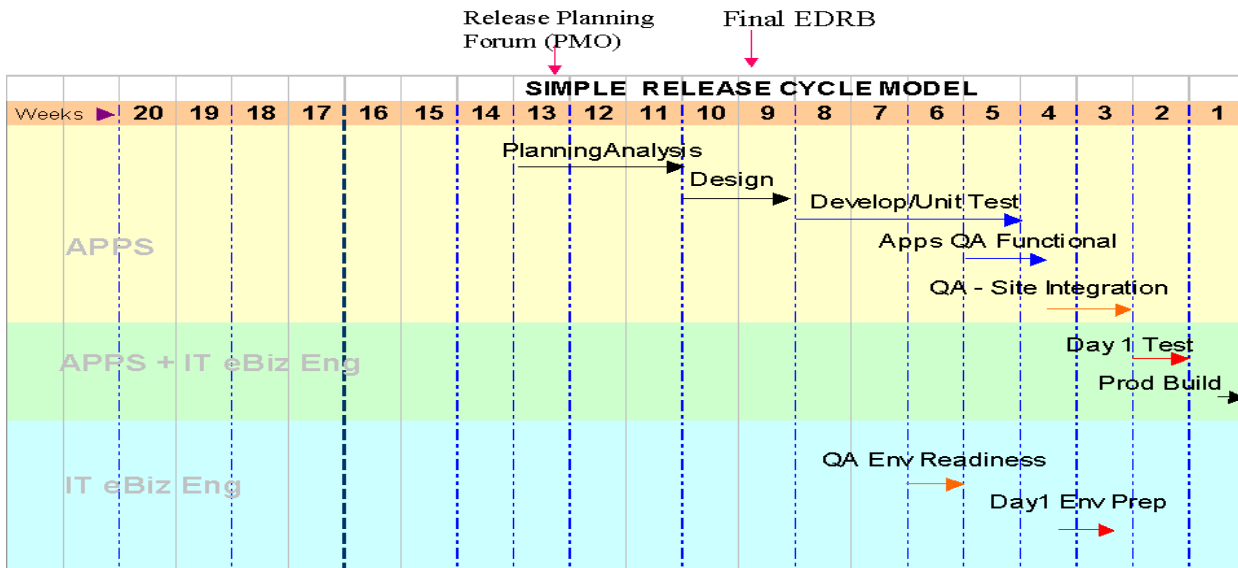


Figure 6: Simple release model

The Release Management Process

In the early days of e-Business development at Intel, the planning horizon for releases was no more than six months, and releases would occur somewhat sequentially within each business group. A release could consist of up to six applications. The dependencies between those applications were shared functionality, test environments, shared resources, or just a shared release date. This approach did have some success, but success was always dependent on the weakest application. If an application failed to pass testing criteria, the entire release might have been pushed out. This in turn had an impact on future releases that were dependent on the same resources and environments. This constant flow of critical path activity was the most stressful for the environment engineers and the QA test teams.

Figure 7 illustrates the four major processes that need to take place to manage a release.

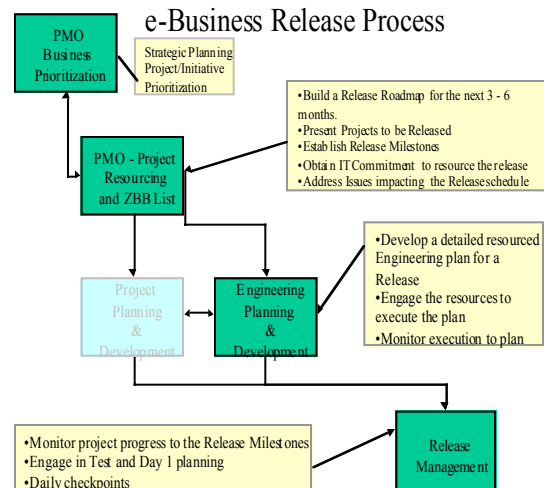


Figure 7: Major release management processes

Business Prioritization and Resourcing

In our largest e-Business area of B2B, the business organization has established a Program Management Office (PMO). The PMO is chartered with the responsibility of establishing a forum for business owners to submit release requests and join with other business owners to prioritize all requests relative to Intel's B2B e-Business priorities. Other participants in the forum include e-Business Integration Engineering management and QA/Test Management. The process is simple. The projects are assigned tentative release dates and priorities by the business owners. Each resource manager assigns resources to the projects until all resources are allocated. In the case of e-Business integration engineering, it is also necessary to determine if the shared test environments are available as required by the prioritization and release date. The business owners then determine if the projects left unresourced can be postponed. If they can, then no priority changes are made. If they cannot, then further prioritization is done or funding is allocated for additional resources or environments until the correct business prioritization is achieved. This process has brought visibility to the ratio between release demand and resource availability to implement a release and has resulted in all the key players working from the same prioritized list.

Engineering Planning and Development

From an engineering perspective, there is a lot to do in a very short space of time. We may identify a set of resources to work on the release in the PMO forum, but at that time we don't know much about the technical requirements. In the past, the application teams presented their technical requirements to the e-Business integration engineers at the end of their design phase, at the e-Business Design Review Board forum (EDRB).

This late engagement has proved to be a major issue for complex releases where new hardware needs to be purchased and possibly new reference designs developed. Today, we assign an Integration Technical Lead (ITL) to the release after the PMO forum has given the green light for the release to proceed. It is the role of the ITL to partner with the application team to get an understanding of the complexity of the application and then to put together an engineering plan to meet the requirements. The ITL is also expected to be a partner with the application development team to develop the technical design and jointly present it at the EDRB.

Daily Release Management Meetings

In the past, the daily release management meetings focused on issues surrounding the release at a particular point in time. This served a purpose and allowed bottlenecks and issues to get addressed very quickly. However, it did not lend itself to looking at the big picture or even looking a week or more ahead. Today, we still have a daily meeting but the release is managed through the implementation of milestones. These milestones,(see Figure 8 below) reflect major dependent events that need to be completed on time to ensure that the release stays on track.

Daily monitoring of milestones has proved to be quite successful and provides advanced warnings of potential showstoppers. A missed milestone results in more attention being focused on the issues rather than an immediate halt to the application or release. A continuation of missed milestones for an application may result in that application being pulled from the release.

Application Name	High Level Design	Reference Design	Detailed Design Rvw	Proto Env Ready	Prelim Inst Design	StartFunct Test	eARL Ready	Start Char'n Test	QA Env Ready	Funct Code Freeze	PROTO Exit	Char'n Testing	StartInteg Testing	Inst Design Freeze	Day 1 Env Ready	QA Exit	Integ Stress Env	StartInteg Testing	Int Stress Complete	Day 1 Plan	Start Day 1	Management Update	Day 1 Complete	Go/NoGo			
Kihei	w15	w26	w20	7/17	w30	7/17	8/14	8/14	8/14	8/14	8/14	8/28	8/14	8/14	8/28	8/28	8/7	8/14	w37	9/5	w37	9/5	w38	9/13	w38		
IBL db																											
Vulcan 2San																											
Larry+ InfoDesk															w38						w38	9/12	w38	9/12	w40	w39	w40
AM/FDBL Content Publishing Role																											
															w38						w38	9/12	w38	9/12	w40	w39	w40

Figure 8: Milestone tracking for a release

INCREASING THE RELEASE FREQUENCY

Engineering and Business Partnership

The e-Business world is constantly changing as customers require new functionality. The industry is constantly improving on technology and functionality, and on server technology; and operating systems and databases are issuing major releases almost every year. This is not a time for bureaucracy and process for process sake. It is a time for flexibility, compromise, and speed. These are most effectively accomplished by an alliance, or a partnership, with all the groups participating in the release cycle. The goal is to jointly deliver the release on time and not to demonstrate the shortcomings of the other groups' deliverables. However, the realistic requirements of each group still need to be respected and delivered with flawless execution.

In the Intel e-Business world, the partnership that exists today between the business organization and the IT e-Business Integration Engineering group is being enhanced with a much closer resource alignment to the business groups. This will provide more focused, dedicated environments and resources for each business group.

Automation

Automation is beginning to play a significant role in speeding up the release process. In the e-Business integration engineering world, we have started to automate server build scripts that can dramatically reduce the time and the amount of engineering resources it takes to build a server. Today, we have all of our OS and database base builds completed and in operation. In the future, we will be automating as many of the instance design scripts as possible.

In addition, we are also bringing in testing tools that have been successful with the back-end systems for test script automation and execution for both functional and stress/performance testing.

Pipeline Server Management

Dedicated pipelines of servers for each business group will reduce server conflicts between business group releases, but will most likely double and triple the number of servers over the next two to four years. Managing these large server farms is going to be a major challenge, ensuring redundancy for failover and reliability rates in the 99.6% – 99.8% range. To complicate the picture even more, Intel is planning a distributed data center architecture across North America, Asia, and Europe. This architecture is seen as a means to improve connectivity and performance in Asian and European countries. The release process may

ultimately include multiple releases in different time zones.

MANAGING E-BUSINESS GROWTH AND CHANGE

Resource Constraints

One of the most difficult challenges facing the e-Business Integration Engineering group today is hiring enough experienced people who can hit the ground running when they join Intel. The e-Business Integration Engineering group has grown from a 30-person organization two years ago to a 100-person organization today. Windows* NT and other operating systems, Oracle, and SQL engineers are the primary resources. The e-Business Integration Engineering group has developed boot camp classes for new hires and puts a high priority on training. We are also looking at opportunities for outsourcing the more standardized tasks to free up our experienced employees to work on the more creative design and engineering tasks.

Outsourcing

The key areas of opportunity for outsourcing that are being evaluated are application development and support, infrastructure engineering environment builds, production hosting, and first-, second-, and third-level support. Many of the outsourcing opportunities can be outsourced within Intel but there may be situations where it makes more sense to use external vendors.

Infrastructure Releases

Keeping the infrastructure current and consistent has always been a challenge in this fast moving ever-changing e-Business world. A year ago there were 75 different reference designs, many of which had minor service pack or version differences. Today, through a concerted effort to standardize and reduce the number of reference designs, the number is down to around 40 and is still going down. Scheduling an infrastructure release for upgrades to servers, operating systems, and databases has always been difficult because of the impact upgrades have on the business release schedule. To lessen the impact, infrastructure releases have been aligned with functional releases. This allows the infrastructure release to leverage the environment engineering resources and the QA test resources. This strategy has been successful, but it is sometimes viewed as an unnecessary overhead and involves a risk to a business release. As a result, Intel's e-Business engineering group is looking into having one infrastructure release a year. The business is supporting this idea because it means a more stable and reliable environment.

RESULTS

Over the last three months that these improved release management processes have been in place, there have been no release misses in the B2B space, and the IT e-Business Integration Engineering group has delivered as required. The early engagement both with the PMO and the ITL has caused teams to plan ahead and adequately resource the releases. Time will tell, however, with respect to increasing the number of releases and whether automation and outsourcing can add further value.

In the next six months there will be approximately 12 releases involving approximately 21 applications in the e-Business B2B business area. The majority of these applications have been prioritized and resourced. This is twice as many releases as in the same period last year.

DISCUSSION

The processes and improvements that have been addressed in this paper are bearing fruit, but are just the beginning of the change that needs to take place. The e-Business world is fast paced and ever changing and requires constant monitoring for new and better ways of doing business.

As we move forward with our releases it is clear that there are limits to the number of servers and people that can be thrown at the problems. The key success factor in delivering on schedule is staying closely partnered with our customers. We all have to make sure that our resources are expended on the highest priorities. For Intel, the PMO forum is a major success factor and needs to be expanded to include all e-Business groups.

It will be difficult for the business to stand by the decision to dedicate one quarter to an infrastructure release. However, without it we stand the chance of falling behind on significant improvements to our ever-growing environment.

Finally, outsourcing both internally and externally may turn out to be a key opportunity to increase the release frequency without incurring a major increase in resources and hardware.

CONCLUSION

Intel's executive management has set the goal to have 100% of its customer business Internet enabled by early 2001. To meet this goal, it has been necessary to develop and deploy e-Business applications in the shortest possible time and with an acceptable level of quality. To achieve this, we are changing the paradigm for building, testing, and deploying systems. Business-focused teams, partnerships between business and engineering groups; automation of environment builds and application testing; outsourcing of suitable functions; and a flexible but managed release management process where engineers and developers

feel each others pain when milestones are missed, are all essential to our success.

ACKNOWLEDGMENTS

Thanks to Jeff Engman for his significant efforts in driving change into the release process and steadfastly following the process.

Thanks to David Roff who has helped drive the automation of the base builds.

Thanks also to Bruce Theurer who has driven change into the release process from the user side. Bruce is responsible for both the PMO and the daily release management meetings.

AUTHOR'S BIOGRAPHY

Alan Hodgson graduated from John Dalton College, Manchester, England in 1971 with an equivalent B.S. degree in mathematics, statistics, and computing. He is a 30-year veteran in information systems and has seen the good, the bad, and the ugly in release management over the years. He specialized for many years in testing and joined Intel 3 1/2 years ago as the Test and Integration Manager for a large SAP project. Alan is currently an Engineering Manager in the IT e-Business Integration Engineering group. His e-mail is alan.hodgson@intel.com.

Copyright © Intel Corporation 2000. This publication was downloaded from <http://developer.intel.com/>.

Legal notices at <http://developer.intel.com/sites/developer/tradmarx.htm>