Why Do I Need All That Process?
I'm Only a Small Project©

Mark Brodnik, Robyn Plouse, and Terry Leip
Intel Corporation

At Intel's Information Technology (IT) department, we developed extensive processes for our projects. While the large projects get the glory, the majority of our projects are less than six months long, have small teams, limited scope, and low risk. We found that we have a variety of project sizes but a single set of processes originally built for larger projects. So how did we fix that issue?

The Intel IT department, like many other organizations, was seeing issues with completing software projects on time and satisfying our customers. Using the Software Engineering Institute's Capability Maturity Model Integration (CMMI℠) as a basis to tackle those issues, management challenged us with the goal of being at Maturity Level 3 in 18 months. While we acknowledged that this goal was overly ambitious, we still took a shot at accomplishing the objective. A small team using the CMMI as their guide built large and formal processes and mandated their use. The inevitable result was that this approach was not well received or implemented by the project teams, especially those at the smaller end of the scale. The IT organization also realized that there was no business justification for a CMMI benchmark. To address this issue, we launched a project to streamline the processes; we took a different look at the CMMI, listened to the stakeholders, and focused on our business objectives. The end result was a set of processes 70 percent shorter and much less prescriptive.

Accelerated Process Improvement (API)

Based on stakeholder feedback, audit results, and process coach input, we identified a number of the work processes that were complex and difficult to use. The organization gathered 35 representatives including key stakeholders, engineers, process coaches, and auditors for an intense three-and-a-half day face-to-face meeting. In preparation for the session we mapped business and model requirements to our existing processes in order to determine what portions were

eligible for simplification. In addition, we created a priority list of feedback from audit, coaching, and pending process improvement requests to determine top *pain points* and focus areas. Since we had limited face-to-face time to complete the work, we were willing to accept less than perfect results and less adherence to the model to streamline the business process.

A key component of the successful collaboration was sequestering the team at an off-site facility to maximize focus and support a collaborative environment. The agenda for this event was driven by the stakeholders rather than the process engineers. The stakeholders drove the vast majority of the changes with a primary focus on accomplishing business results rather than textbook model compliance. The team reviewed each work process against the business requirements and the CMMI model to identify portions of the process that could be considered for deletion or simplification. The group was broken down in to three sub-teams each addressing a different process area. Daily report out sessions and synchronization sessions helped ensure consistency between the teams. Our CMMI model expert floated between the teams to answer model related questions.

One of the themes we noticed was that the initial process development team had implemented many of the sub-practices in the CMMI model believing they were a required model component. For example, based on sub-practices from the configuration management process area, we required each document to have unique identifier labels both internal to the document and in the file name – a convention that was unnecessarily complex and added little value for our typical 8-to-12 person project teams. We also defined and mandated specific elicitation techniques to meet the intent of the sub-practices in the requirements development process area; these techniques were overkill for our small project teams which

worked closely with their customers and stakeholders. Finally, we eliminated the requirement for a separate formal commitment for resources by combining this with one of the key milestone decisions. This came about by an incorrect interpretation of the sub-practices in the project monitoring and control process area.

Figure 1 (see page 20) shows the old processes for scope, schedule, and resource management. These separate processes totaled 29 pages and 17 process steps. The new process was able to combine scope, schedule, and resource management into one process document that was five pages long with eight process steps. This was accomplished by simplifying the wording, combining or eliminating process steps, and removing instructional text which we thought would substitute for skills expertise from the process documents.

Overall, we achieved a 70 percent reduction in document length, with one process shrinking to just 17 percent of its original size. Detail of other process and template reductions can be seen in Table 1 (see page 20). The original and combined processes can be accessed via the online version of this article.

When we evaluated the outcome of the effort, one of our project managers said the following:

> The API face-to-face session was great for analyzing the weak spots (in the documentation and in the processes), as well as acknowledging the flaws of the previous release. The teams addressed the issues with great teamwork that allowed for development of some creative and tangible action plans.

In the end, we reduced the workload of our processes and improved documentation so that it would more effectively deliver key information. The result of process simplification was the ability of smaller projects, originally deemed out of scope,

Original Flows for Scope, Schedule, and Resource Processes

**Combined and Simplified:
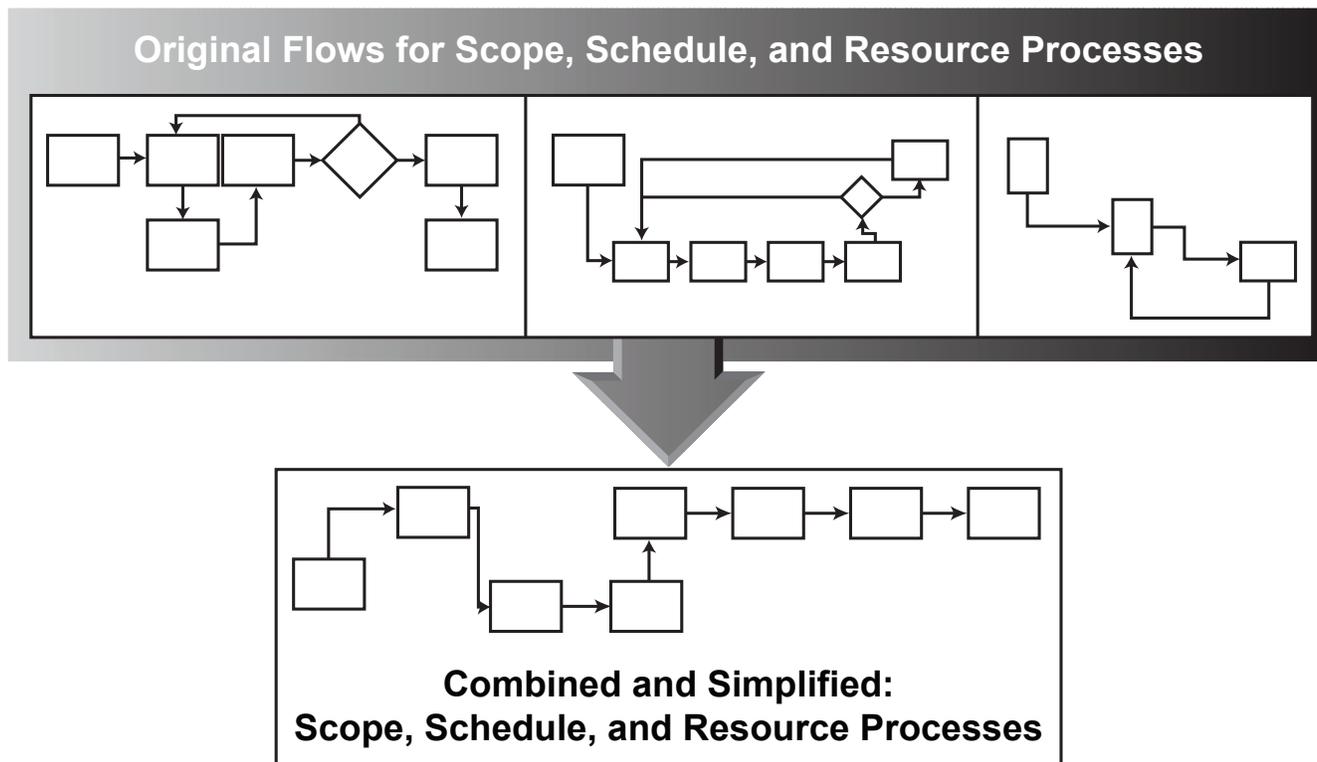Scope, Schedule, and Resource Processes**

Figure 1: *Example Simplification and Reduction*

to adopt the processes. Our number of eligible projects went from 36 percent to 50 percent. Our process coaches could now support an average of 18 projects per coach, up from 14 projects per coach before the simplification. In addition, the training course delivery was streamlined to half of their original duration.

What About Very Small Projects?

As senior management started seeing the benefits of standard processes, they made the decision to require all projects greater than eight weeks in duration to adopt the processes. As the adoption rates for these projects came closer to 100 percent, we began to look for ways to help the projects originally deemed too small for our processes. These very small projects, less than eight weeks in duration, were considered low risk and therefore were not required to follow any formal processes or standards, although some did have

their own self-imposed methods of running their projects. A result of this was that management did not have sufficient visibility into these projects and some of the larger projects would attempt to *chunk* their work into less than eight weeks so they could avoid the full process and external visibility into their work.

In our discussions with project managers, we also realized that the eight-week rule was too limiting, since many projects were only slightly longer (8 to 12 weeks), but had just two or three fractional resources. These project managers felt that the overhead of the processes were still too large compared to the amount of overall effort of the project, and they wanted to also take advantage of a smaller set of processes. They argued that they spent most of their time filling out forms or compliance checklists rather than focusing on the task at hand. We also discovered through audits that project managers in this class of projects would often

run their projects as usual and then go back after the fact and create the required process collateral. Clearly, there was need to provide processes that addressed very small projects.

Proof of Concept (PoC)

Before we deployed a new set of processes across all of IT, we decided the best approach was to work with a limited set of these very small projects through a POC in one or two IT groups. We would then analyze the data to determine if an IT-wide deployment was warranted or if we needed additional pilots. Finally we planned to take a thorough look at our entire process library to implement a more robust project characterization approach that right-sizes the processes and tools used by our project teams.

The first area we looked at was how our process deliverables were organized. Our projects' process deliverables are defined at two main levels. The first level is what a project needs to review at a milestone decision meeting with management (ex. requirements, design, and schedule). The second level is deliverables that are typically produced as a result of project work (e.g., various supporting plans, internal compliance scorecards, review and approval of records, change records). In order to address the issue of excessive process overhead for small projects, the

Table 1: *Old Versus New Document Sizes*

Document	Old Size	New Size
High-Level Requirements Process	9 pages – 7 steps	4 pages – 5 steps
Detailed Requirements Process	11 pages – 6 steps	4 pages – 5 steps
High-Level Requirements Template	10 pages	2 pages
Detailed Requirements Template	14 pages	6 pages
Scope/Schedule/Resource Process	29 pages – 17 steps	5 pages – 8 steps
Project Plan Plus Supporting Documents	4 documents	1 spreadsheet
Baseline Change Control Process	6 pages – 13 steps	5 pages – 9 steps

approach to the PoC was to determine the major process deliverables that a project should produce and require only those deliverables. To speed up the PoC, we started with the first-level deliverables and decided to leave the creation of second level process deliverables up to the discretion of the project manager. Ultimately, the goal was to have project managers focus more on the product deliverables and less on the process deliverables.

The next item we looked at was combining the templates for the various process deliverables into a single, simplified template that contains the basic information the projects are required to complete. This allowed decision makers to review this document at milestone decisions rather than requiring the project manager to duplicate information in a separate, formal presentation.

The last major challenge we wanted to work through in the PoC was the cutoff point between these very small projects and our larger projects. In the end, we decided that a more pragmatic approach was to look at several attributes to consider for the cutoff such as team size (one to three team members), overall effort (around 500 hours), and project risk (no financial systems or software exchange impact) to provide a more balanced set of classifications.

Results

While the model is not our primary focus, we are still measuring ourselves against its requirements to make sure we have not missed a best practice. Through a series of Standard CMMI Assessment Method for Process Improvement℠ (SCAMPI) Cs, Bs, and As performed over the past three years we are seeing increased compliance with the model. We have been measuring progress toward Capability Level 3 in Configuration Management, Project Planning, Project Management and Control, Requirements Management, and Requirements Development, and we are also beginning to measure compliance for Measurement and Analysis, Process and Product Quality Assurance, and Verification and Validation.

From a business perspective, we are seeing positive results in shortening the overall duration of our projects and improving our ability to meet our committed delivery dates. This is done by paying particular attention to the planned versus actual time in each individual phase.

In addition, our quality assurance audit team discovered that process compliance had improved as a result of the simplification. Since the processes had changed so dramatically, it was not possible to do an apples-to-apples comparison of the audit results prior to simplification, but both auditors and project team feedback indicated that compliance was better and easier to achieve.

Summary

In our process engineering journey we learned the following things:

- First and foremost, we need to get our stakeholders involved in every step of the development process. This ensures quicker acceptance by project teams and uncovers usability problems earlier in the process.
- Second, it is essential to have a business focus rather than a model focus. It is easy to lose sight of business objectives when trying to comply with CMMI requirements.
- Third, we should start by building processes as small as possible and add to them only as needed. We learned that by building large processes first, the true cost extends beyond the development and requires significant effort to support and maintain.
- Finally, time is the most precious commodity in process improvement. Getting a quick win that the organization would accept is more important than perfection.

In the end, it is all about being both efficient and effective.◆

About the Authors

Mark Brodnik, Project Management Professional (PMP), is a project manager at Intel's IT group where he specializes in process development and process improvement. He has 15 years of IT software applications development and management experience and holds a bachelor of science degree in management information systems from the University of Arizona.

Intel Corporation
M/S C11-123
6505 W Chandler BLVD
Chandler, AZ 85226
Phone: (480) 552-7028
E-mail: mark.brodnik@intel.com

Robyn Plouse, PMP, is a SCAMPI lead appraiser and a quality auditor at Intel's IT group where she also leads the CMMI Community of Practice. She has 10 years of project management and quality experience with Intel and nine years of experience as a systems engineer at Lockheed Martin. She holds a bachelor of science degree in aerospace engineering from San Diego State University and a masters degree in engineering management from Santa Clara University.

Intel Corporation
M/S RR5-505
1600 Rio Rancho BLVD S.E.
Rio Rancho, NM 87124
Phone: (505) 893-6319
E-mail: robyn.plouse@intel.com

Terry Leip is the lead quality assurance auditor at Intel's IT group. He has more than 20 years of software development and quality experience with Intel and Lockheed Martin and holds a bachelor of science degree in biology and computer science from Grand Canyon University.

Intel Corporation
M/S C11-123
6505 W Chandler BLVD
Chandler, AZ 85226
Phone: (480) 552-2079
E-mail: terry.leip@intel.com