

Transferring Software Development Best Known Methods between Generational Product Lines

James R. Bindas

Intel Corporation
JFT-101, 2111 NE 25th Avenue

Hillsboro, OR 97124

Phone: (503) 264-8869

james.r.bindas@intel.com

Biography

James R. Bindas is a Software Process Engineer with Intel Corporation in Hillsboro, Oregon. His tasks include working with the software development community to help standardize and improve software development processes. James has been working with Intel for eight years in various software roles ranging from software tester to project leader. Before joining Intel James worked for RCA/GE Solid State and Harris Semiconductor as a Quality Assurance Technician. James holds a Master of Science degree in Computer Science from Steven's Institute of Technology and a Bachelor of Science in Graphic Communications from California University of Pennsylvania.

Abstract

Launching a new product line is a difficult and time-consuming task under the best circumstances. While working on a new project can be refreshing, difficulties arise in allocating sufficient resources to create and tune the new product development processes. Modifying the processes or Best-Known Methods (BKMs) of the past is one technique employed to allocate time and resources more effectively.

Transferring BKMs from an earlier product line into the current one can bring about its own set of challenges which include: defining the BKMs, producing repeatable BKMs to be deployed in varied environments, implementing techniques to log the BKMs, and placing the BKMs in a central location.

This paper discusses one approach to transfer software BKMs between generational product lines. The topics addressed include:

- Defining the project's mission and goals.
- Creating definitions and processes to support the BKM transference.
- Encountering and solving issues while implementing the processes.
- Addressing the human dynamics issues involved.

Introduction

Many words in the English language have multiple definitions with similar meanings. A prime example is the word “quality.” The term Best Known Method (BKM) also falls into this category. The difficulty in defining BKM surfaces when attempting to phrase the composition, implementation, and certification. Some persons would use another term, but other phrases cannot rival the perception and awareness of “BKM.” This paper addresses how to succeed in the challenging environment of transferring BKMs to another party.

My department is responsible for the design, development, and delivery of Electronic Design Automation (EDA), Computer-Aided Design (CAD) tools to internal customers - the Intel microprocessor design groups. Our department employs over 500 employees at four major sites to support the quickly evolving microprocessor roadmap. The department is divided into two layers of cascading development teams located in Santa Clara, California; Hillsboro, Oregon; and Haifa, Israel. We refer to these groups as generational teams. As one development team is maturing (Team-B), the other group (Team-C) is preparing to deliver solutions for the next generation of EDA tools. With the emphasis on tighter time schedules, pressure is placed on devising Team-C to produce high quality solutions, with minimal start up costs.

In order to adhere to the tight schedule, an approach adopted was to reuse Team-B’s software development processes, some of which were considered to be Best Known Methods (BKMs). While the idea is logical, no process “existed” to support the transfer. A common BKM definition and transfer infrastructure was needed between the two generations to support this effort. This project was assigned to our team, which is outside both generational organizational structures and is charted with managing several cross-organizational, tactical improvement projects. Managing the effort was my responsibility.

Preparation

The project began in March 1997 by researching Team-B’s software development processes, their current status, their documentation, and their utilization. Team-C followed by identifying which processes they required. A majority of the research had been performed by previous improvement projects; thus the job was a straightforward procedure to gather data, sort the information, and verify it with key generational process personnel. An added bonus was the early identification of key personnel with expert knowledge who were interested in championing the processes.

Forming a Working Group

The best method to gain commitment from a team is to include them early in the process. In late May, early June 1997, key personnel from both generations were invited to begin the project. Obtaining a commitment from team members was not a problem since members had their own personal reasons to support the project. For example, the return on investment (ROI) for Team-B included:

1. An opportunity to document their own processes.
2. A basis for training new hires.
3. A legacy on which to base future processes.
4. An opportunity to introduce improved Team-C processes into Team-B.

For Team-C, the ROI was straightforward:

1. Saves time and effort from developing new and untested processes.
2. Provides a single source for information on Team-B processes.
3. Provides a vehicle to improve upon lessons gained from implementing the process in previous projects.

Establishing a joint working team provided a forum to level expectations on both sides early in the process.

Defining the Working Group

In any good working group, the first item of business is to define the mission, objectives, and definition of success. The team agreed to limit itself to a four-month scope with short-term, well-defined goals because:

1. The group's focus often varied during a long-term project.
2. Management priority sometimes shifted due to uncontrollable forces.
3. Team members tended to leave over time, due to work assignment changes.

While preparing the mission statement, the group discussed whether they should monitor the compliance level

BKM Validation and Certification Team Charter

Rev 2, 6/16/97

- **Mission Statement**
 - **Develop a basic infrastructure to support the transfer of BKMs from Team-B to Team-C.**
- **Objectives**
 - **Create a basic pilot BKM transfer methodology.**
 - **Review and document any lessons learned from the experience.**
- **Definition of Success**
 - **Document methodology of a repeatable BKM transfer.**
 - **Two BKMs or KMs transferred using the above documented methodology by the end of 3Q97 (A total of six by end of 4Q97).**

Figure 1

of each BKM after it was transferred. The group decided against this, because it was a factor outside of their control. However, the team was able to control building an infrastructure that allowed BKMs to transfer between groups, where success could be measured. The final objectives included creating a repeatable process and documenting lessons learned (**Figure 1**).

Defining the Methodology

With a common purpose and goal established, the next step involved defining common terminology. BKM was the first term to be defined with its many meanings and applications. A common reaction was to create a new standard definition. After consulting a resident BKM expert, the team agreed they would have difficulty in settling on a single definition. Since the term BKM is common in people's minds, the introduction of a new term would never be accepted. The best that could be accomplished was to define BKM in the context of the Team-B/Team-C software development environment. Surprisingly, the team quickly and painlessly agreed to the following definition:

A Best Known Method (BKM) is a portable package of methodologies and supporting materials that demonstrate improved results over other known methods to solve specific problems/issues in a given environment. A BKM is composed of the following components:

- ❖ *Owner(s)*
 - The expert(s) for this BKM.
- ❖ *Description*
 - Brief description of the BKM, including its purpose.
- ❖ *Entry/Exit Criteria*
 - What must happen and/or what must be available before the procedures associated with this BKM are started, and what must happen and/or what must be available before this BKM can be considered completed.
- ❖ *Procedure*
 - How the BKM is implemented.
- ❖ *Supporting Materials*
 - Any tools, templates, scripts, references and training that are needed to provide descriptions and on-line locations.
- ❖ *Deployment History*
 - Information on when and where the BKM was deployed.
 - Metrics and/or data from previous projects can be used to measure the BKM's effectiveness or estimate future expectations.
 - Examples of past experiences.
- ❖ *Lessons Learned*
 - Documented experiences of previous BKM users.
 - Descriptions discussing why certain paths were chosen in the past.
 - Thoughts about what we might do differently next time?

Note: Until the BKM is "Certified," we will refer to it as a KM or Known Method.

Two key points are included in this definition. The initial point is that a BKM is a portable *package* of not simply methodologies, but supporting materials as well. The package includes the tools and procedure (**Figure 2**, x and y-axis's). The second key point is that the BKM documents experience (**Figure 2**, z-axis) which makes the BKM much more usable.

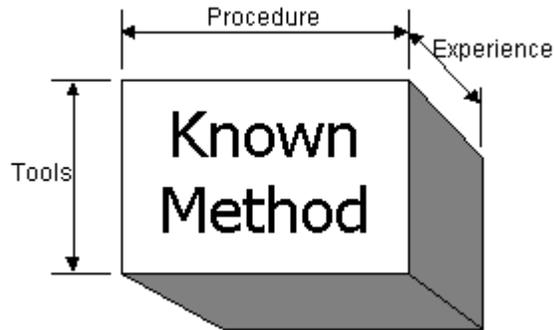


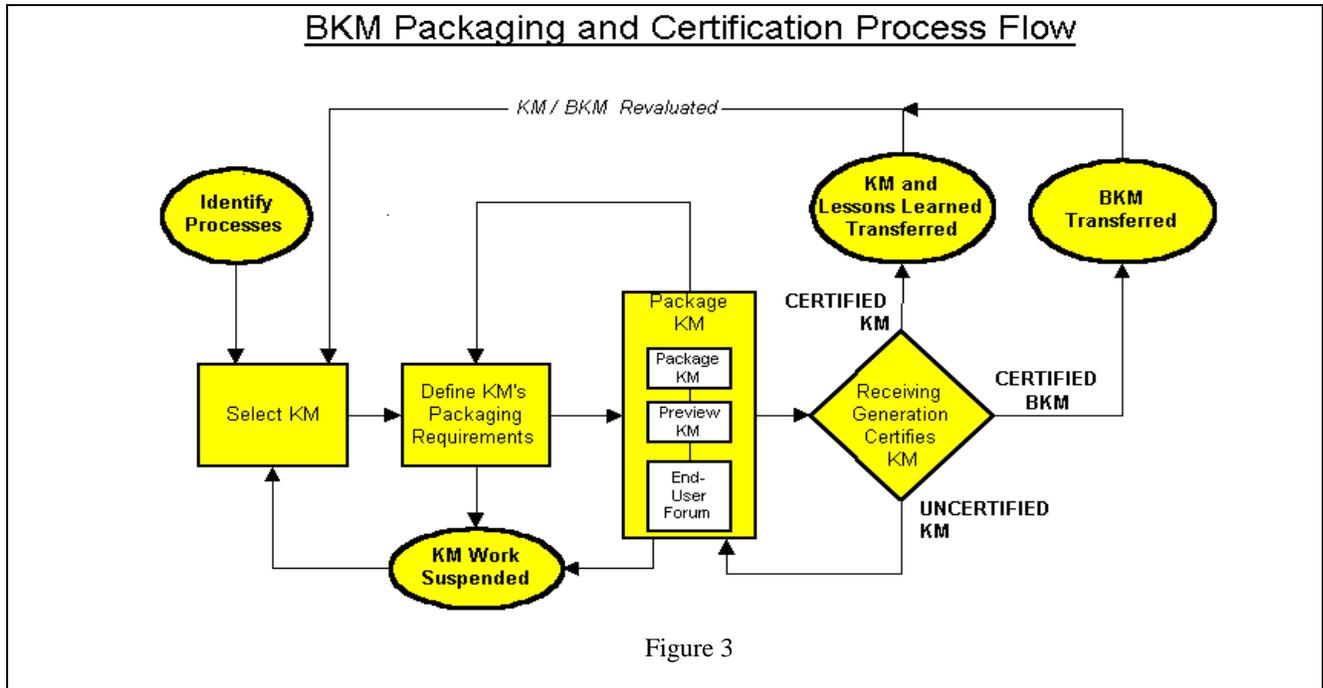
Figure 2

The next step was to create an infrastructure to support the transfer process that resulted in creating **Figure 3**. The process began simply - select the KMs to be transferred, package the KMs, verify, and certify them. Early issues arose, such as:

1. A person, familiar with the process, should package the BKM because they are in a position to ask the right questions, and include the correct components in the package.
2. Team members discussed the difference between validation and certification, suggesting that this could be a duplication of effort. The team replaced validation with the End-User Forum (defined later). Certification would be determined if the KM customer could implement it into their infrastructure.
3. Team members expressed concern about whether the KM could be certified as a BKM. Certain Known Methods are valued for their lessons learned, but not necessarily the best methods to follow. A method was needed to separate KMs from BKMs allowing both types to transfer.
4. A drive to keep the infrastructure simple existed. Too often, good processes failed because of high overhead in maintaining a complex model.

After concerns were addressed, the resulting model was formed including four key points:

1. The model was generic allowing any generation to transfer KMs to another.
2. Expectations were addressed initially by agreeing to a common definition and requirements between Team-B and Team-C, before packaging began.
3. During the packaging stage, an End-User Forum was held between Team-C and Team-B. During this time, actual Team-B End-Users provided real-life feedback about the KM; Team-C had an opportunity to ask insightful, key questions.
4. Some KMs offered valuable lessons learned, but were in such condition that they could not be used in the receiving generation infrastructure. Therefore, the method was not certifiable and was transferred to the receiving generation as a KM. The KM would then be examined and a new methodology will be constructed.
5. Certification is performed by the receiving generation, based on a set of well-defined criteria (**Figure 4**).



Implementation

One of the greatest challenges of the project was implementing the process after creating the methodology. In order to accomplish this, the team initially agreed to the high-level processes and definitions before starting implementation. This yielded a common framework to build upon, allowing the lower-level detailed processes to be specified as each step of the high-level processes was completed.

Implementation work started in late June 1997 with two KMs -

1. Software Documentation Process.
2. Software Development Life Cycle.

The selection process was kept simple because a high methodological-based approach would not produce a high ROI to justify the overhead. The KMs selected had two common attributes:

1. Team-C required them.
2. The KMs were well documented since the infrastructure was new and untested.

The second phase of the process, Defining KM Packaging Requirements, began early to mid-July. The process itself proved to be enlightening and profitable. The team soon realized that assumptions made by both Team-B and Team-C were not the same. If expectations were not leveled early, they would have led to wasted effort, and conflict within the group. This problem was solved when Team-C realized that Team-B's Life Cycle would not bring as high an ROI as previously expected. Software inspections were substituted for the Life Cycle KM.

The packaging began in early August and ended mid-September. Using the BKM definition (**Figure 4**) as a guide, various BKM components were identified and gathered. The next step was to create an hour-long (15-25 slides) presentation, describing the different components in the BKM definition and to answer questions about the KM. This presentation, or management summary, was presented at the End-User Forum in late September.

Owners were identified and assigned to package the KMs. A common packaging location was needed to act as a KM repository. The main, and perhaps only, common information link across ten time zones was our Intranet, called InfoNet. A web site was designed to be the repository.

BKM Certification Definition		
The process that ensures the BKM is in a state that can be implemented into the receiving generational infrastructure with minimal effort.		
<i>BKM Certification Criteria</i>		
<u><i>Certified BKM</i></u>	<u><i>Certified KM</i></u>	<u><i>Uncertified KM</i></u>
<ul style="list-style-type: none"> • Recommended by the packaging team with a minimum of changes. • Packaging is comprehensive. • Completed two cycles - Product Release or Tape Out. • Uniformly accepted and deployed by the sending project. 	<ul style="list-style-type: none"> • Recommended by the packaging team as a set of learned lessons from implementing the method. • Packaging is completed as much as possible. • Not completed two cycles. • Not uniformly accepted or deployed by project. 	<ul style="list-style-type: none"> • Missing or ambiguous KM components. • Out-of-date components that need to be updated.

Figure 4

The End-User Forum planning proved to be non-trivial. The plan was to keep the forum to a small, informal group of less than 15 people. The group was designed to represent a wide selection of vocal people department-wide. Getting participation for the End-User Forum was easy due to pre-selling the forum. The main dilemma was scheduling. Finding an open timeslot for 15 people proved to be a non-trivial task. Finally, after two postponements, the two and one-half hour End-User Forum was held on September 23, 1997.

A week before the forum, the team mailed the formal invitations, agenda, and sample questions alerting the members to the types of questions to be asked. The agenda was simple - present the slides and field questions from the forum.

What was refreshing about the forum was that new ideas and information were exchanged liberally between Team-C and Team-B. Although the discussions often continued on tangents, the participants offered undocumented information that only few had. This information was repackaged into a KM for future use.

The certification process was scheduled to start on October 30, 1997. Plans included splitting KM certification into two half-hour sessions. At the first session, an abridged version of the End-User slides was presented. At the second session, held weeks later, specific questions were raised. At this time, the certification team formally approved the two KMs as a BKM.

Next Series of KM

With the BKM infrastructure in place, emphasis was shifted from creation to KM transfer. Originally, four KMs were to be transferred:

- Build and Release Process.
- Unit Test.
- Flow Test.
- Methodology Development.

Because knowledge of our activity had spread, we received a request for a fifth KM: Project Estimation.

During the KM Definition, both generations discovered the need to package a “Build and Release KM” disappeared which resulted in this KM being placed on hold.

Both “Unit and Flow Test KMs” are currently being packaged and an End-User Forum was held in June 1998. Again, the End-User Forum discussion was lively and informative. Unfortunately, due to schedule conflicts, the KM will not make it on the certification team’s agenda until August.

Both generations initially agreed that Methodology Development was a potential BKM. During packaging, team members discovered it would be best to package Methodology Development as a KM. The End-User Forum was held in November 1997. Both generations were eager to share information. A follow-up meeting was agreed to share data, which would not be available until first quarter 1998. The follow-up meeting was held in April 1998, and the KM was presented to the certification team in June 1998. Further discussion was held once again, regarding the KM certification level. In the end, the team determined the KM should be accepted as a “Certified KM,” due to the process needing more maturing.

Project Estimation was agreed to by both parties, but was also delayed a few months due to a product release. Project Estimation was eventually packaged and an End-User Forum was held in January 1998. Project Estimation was then certified as a KM, because it was not widely accepted and deployed by Team-B.

Lessons Learned

PACKAGING AND AVAILABLE RESOURCES: Two KMs were committed for packaging, and resources were allocated for the next quarter due to higher priorities. As it turned out, other priorities quickly appeared and supplanted the packaging effort further down on the priority list. This resulted in lost momentum and little effort being allocated for packaging.

SCHEDULING CONFLICTS: Three KMs were delayed due to fourth quarter 1997 holidays and again due to product releases. Again, June and July 1998 scheduling furthered delay certification. These conflicts further slowed the momentum almost to a halt.

DEVELOPING CHECKLISTS: This lesson was discovered during KM certification when one KM was nearly denied certification due to confusion regarding the criteria. Looking back, unspoken assumptions, held by some of the certification team members, caused this confusion. During the next round of KM definitions, a checklist was used to unmask any hidden assumptions.

ENCOURAGING FACE-TO-FACE END-USER FORUMS: Because our department is multi-geographical, face-to-face meetings are not necessary for certain people. However, the receiving group needs to be present in person, to fully engage in the exchange of information. Some forums had problems with various attendees being heard or recognized on the telephone conference call. Technical problems prevented one attendee from participating in the forum. When the participants were drawing on the white board, a person on the phone line was trying to follow the conversation, based on verbal clues. These difficulties could hamper the knowledge transfer process, as well as, cause miscommunication between the teams.

KM PACKAGING PARTICIPATION: The Haifa, Israel representative departed Intel, and was never replaced. Because of this, the KM did not become implemented in Haifa

Conclusion

Often overlooked in a vertical organization, is the difficulty in sharing process improvements between teams. Unless action is taken, an improved software process cannot be developed and will end with the current product line, simply to be recreated in the succeeding product line bearing the same startup pains. One response to break the barriers between product generational teams is to ensure improvement upon this procedure (**Figure 5.**)

Was the project a success? Yes, if judged by the pre-determined “Definition of Success.” This definition states the team should have a documented methodology enacted and three KMs transferred. Three additional KMs should be transferred in the near future. The project’s success can be evaluated by Team C’s use of five-out-of-six KMs. One KM, that was placed on-hold, indicated that the process was preventing work from being performed on a non-valued KM. The project’s short-term success appears to be growing into a long-term success as time progresses. This success is not measured by active processes, but by people outside of our team asking to use the transfer process for their own purposes.

Quick BKM Transfer Process Overview:

1. Research Team-B's process for their status and utilization.
2. Research Team-C's needs process requirements.
3. Identify both Team-B and Team-C's stakeholders and form a Working Group.
4. Develop the Working Group's
 - a. Mission Statement
 - b. Objectives
 - c. Definition of Success
5. Working Group develops a methodology by defining:
 - a. BKM Definition
 - b. BKM Certification
 - c. BKM Transfer Flow
6. Team-C selects a KM to transfer.
7. Team-B researches the KM to determine feasibility for transfer.
8. Team-B and Team-C meet to discuss KM transfer requirements.
9. Team-B's owner packages the KM.
10. Team-C's receiver previews the KM to ensure meeting requirements.
11. Working Group sets up a centralized and accessible BKM/KM repository.
12. Working Group sells End-User Forum to prospective Forum attendees.
13. Face-to-face End-User Forum is held.
14. Team-B repackages KM with information taken from the End-User Forum.
15. Working Group presents KM to the Certification Board.
16. Certification Board certifies the KM as a BKM

Figure 5

Acknowledgment

The success of this project is due to the entire BKM team. Each member contributed his or her time, energy, insights, experience, and knowledge to this project. Three Intel software engineers should be specifically mentioned

- Doron Becker
- Greg Hannon
- Tamar Yehoshua.