Software Process Achievement at Tinker Air Force Base, Oklahoma

Kelley Butler
Walter Lipke

September 2000

TECHNICAL REPORT
CMU/SEI-2000-TR-014
ESC-TR-2000-014

Carnegie Mellon
Software Engineering Institute
Pittsburgh, PA 15213-3890

Software Process Achievement at Tinker Air Force Base, Oklahoma

CMU/SEI-2000-TR-014
ESC-TR-2000-014

Kelley Butler
Walter Lipke

September 2000

Oklahoma City Air Logistics Center, Directorate of Aircraft, Software Division, Test Software and Industrial Automation Branches

This work is sponsored by Oklahoma City Air Logistics Center. The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

For information about purchasing paper copies of SEI reports, please visit the publications portion of our Web site (http://www.sei.cmu.edu/publications/pubweb.html).

Table of Contents

List of Figures

List of Tables

Foreword

When you read this report, I think you are in for a pleasant surprise. The software folks at the Oklahoma City Air Logistics Center are not a lock-step bunch of government bureaucrats wrapped up in a security blanket of inflexible software procedures. As their story unfolds, they emerge as a group of people trying to figure out how their software skills can best benefit the community they work and live with. If this takes them outside the normal confines of the Software Capability Maturity Model® (CMM®) and its sequence of maturity levels, they continue to persevere, and they often find that their biggest value process improvements come from proactive coordination of hardware-software processes and supplier-developer processes.

This may have slowed their progress toward becoming a Software CMM Level 5 organization, but it qualified them as Process Achievement Award winners on two highly significant counts: as catalysts for process improvement in their non-software neighbor organizations, and as early pioneers of an integrated software/system/stakeholder approach to process excellence as exemplified by the new generation of integrated CMMs.

As they contemplate the future, you can see a tension between their wish to rework their processes to capitalize on the exploding changes in information technology, and their wish to satisfy "software CMM experts" pointing them toward an ideal of a stable software process. I have every confidence that their creativity, good sense, and focus on human values will lead them to a synthesis that captures the best elements of both.

—Barry Boehm

® Capability Maturity Model and CMM are registered in the U.S. Patent and Trademark Office.

Acknowledgments

When asked what the key is to the organization's success, the answer that best sums it up is "want to." "Want to" by the employees and the management. The desire to improve and meet the next challenge is an integral part of the organization's attitude. Add to that stability in the organization's leadership, something that is very uncommon in both government and industry, and the result is success. It is impossible to put into words the contributions of all the employees and managers across the organization. To everyone in the Test Software and Industrial Automation Branches, thank you for all you have done and all that you will do.

—Kelley Butler, Quality and Process Improvement Focal Point

Going through the rigors of the Institute of Electrical and Electronics Engineers (IEEE) Software Process Achievement Award was a great experience for me, Kelley Butler, our Software Process Improvement lead, and every employee in the Test Program Set /Industrial Automation (TPS/IA) Branches. I can't say enough about the review experience and the opportunity it gave us to meet two of the icons of our industry: Dr. Barry Boehm and Watts Humphrey. We are certainly grateful to them, the other review team members (Dr. Vic Basili, Manny Lehman, and Bill Riddle), the IEEE Computer Society, (in particular, Leonard Tripp), and the SEI (Dr. Steve Cross, Director), for recognizing our software organization with this prestigious award.

—Walt Lipke, Deputy Chief, Software Division

Abstract

On May 20, 1999, the Test Software and Industrial Automation Branches of the Oklahoma City Air Logistics Center's Directorate of Aircraft Management's Software Division at Tinker Air Force Base were awarded the IEEE Award for Software Process Achievement. This report will outline the process improvement activities and successes that led to the award.

1 Introduction

On May 20, 1999 at the International Conference for Software Engineering, our name was officially announced as the 1999 recipient of the IEEE Award for Software Process Achievement. It was a wonderful moment in our organization's history. Receiving the award validated all of the effort to improve. This report describes much of the information the IEEE Review Team used in making their final decision for the award. We are the Test Software and Industrial Automation Branches of the Oklahoma City Air Logistics Center (OC-ALC), part of the Air Force Depot located at Tinker Air Force Base (AFB), Oklahoma, just east of Oklahoma City.

We are composed of approximately 360 personnel, mostly electronics engineers, organized as shown in Figure 1. Three branches develop and maintain Test Program Sets (TPSs), which are used with Automatic Test Equipment (ATE). A TPS is the software, interface hardware, and documentation used to test avionics (black boxes and circuit boards) and jet engines. One of the three branches also provides software support to jet engine, constant speed drive, and eddy current testing, along with the support of software for several automated processes associated with jet engine overhaul. Another branch participates in all software acquisitions for the B-2 weapon system, including Operational Flight Software (OFS), Ground Based Software (GBS), and TPSs. The weapon systems affected by the software developed and maintained by the Test Software and Industrial Plant Equipment Branches include the A-10, B-1B, B-2, B-52, C-141, C-5B, E-3A, -135, F-15, F-16 aircraft, Advanced Cruise Missile, and 12 jet engines.

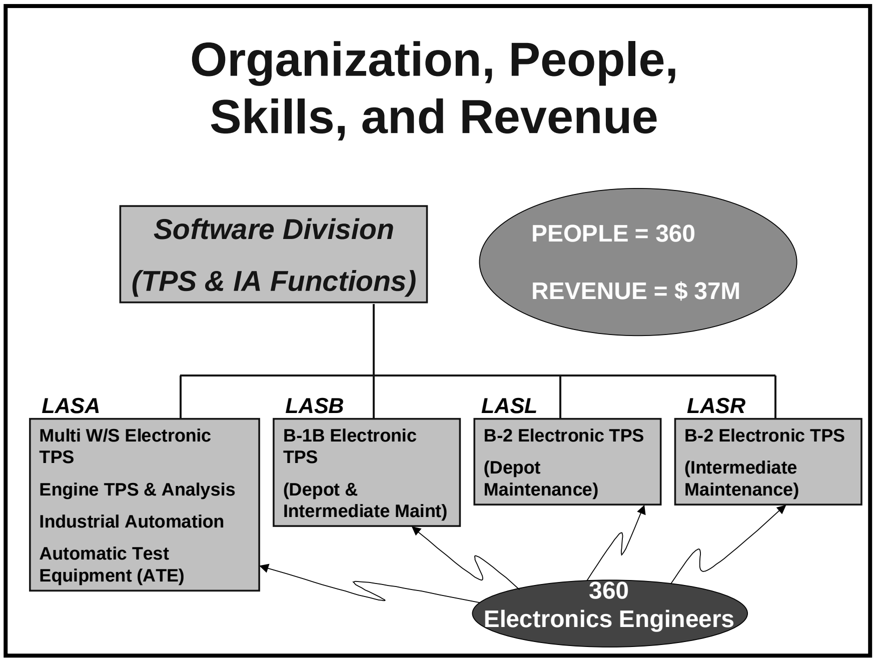


Figure 1: Organization and Mission

Delivered products and services occur in both the acquisition and operational phases of a weapon system's life cycle and include

- Test Software Development and Maintenance
- Interfacing Hardware Design and Modification
- Integrated Logistics Support Data and Services
- Technical Manuals
- Engineering/Technical Services such as Acquisition Support and Independent Verification and Validation

In Fiscal Year 1999, the Test Software and Industrial Automation (TS/IA) Branches produced products and performed services utilizing approximately 496,000 man-hours of labor and having a value of $33.5 million.

The three principal product areas of the TS/IA Branches shown in Figure 2 are Industrial Automation (IA), TPS, and ATE. Each of these areas is broken down into sub-areas. The numbers shown above the boxes are the percentages of the total effort in man-hours (i.e., where our labor is expended). As you can see, the vast majority of our work is TPSs. In the specific case of the TPS workload, the percentage above "Electronics" represents the portion of all work, not just the portion related to the subdivision of TPS. Thus, our main product line is TPSs for electronic repair.

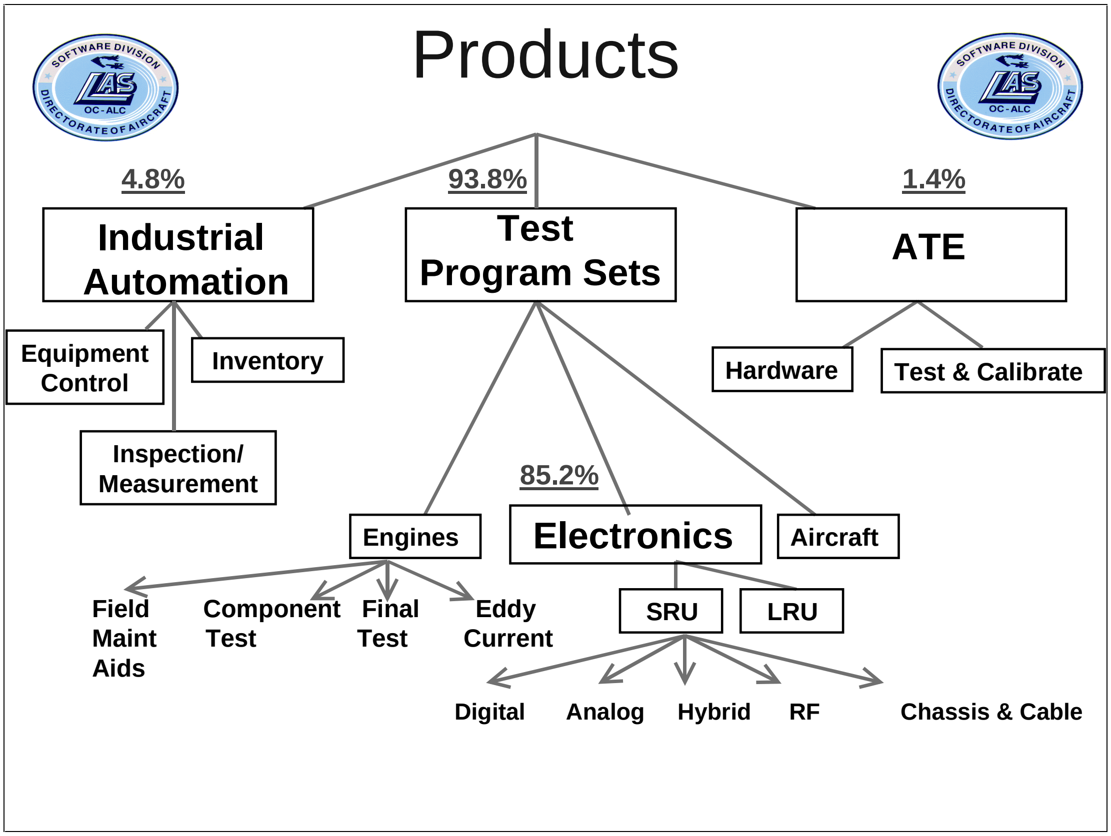


Figure 2: Products

Our primary workloads are shown in Table 1 below.

Primary Workloads	
B-2 Test Program Set Development	Development of 285 TPSs; scheduled to be completed in 2002.
B-1B TPS Maintenance and Rehost	Ongoing maintenance and rehost of over 600 TPSs in support of OC-ALC and the B-1B Operating Bases.
Jet Engine Testing, Trending, & Support	Twelve jet engines, includes engine accessories (electronics), an ever-growing workload, and support of Jet Engine overhaul process. Direct support is provided to Air Force installations world-wide.
Avionics TPS Development, Rehost, and Maintenance	Multiple weapon systems; continue to support and provide new solutions for the OC-ALC avionics repair organization.

Table 1: Primary Workloads

In contrast to the Test Software and Industrial Automation Branches production numbers of $33.5 million for Fiscal Year 1999, Tinker Air Force Base performs about $1.25 billion of depot work annually, which is truly a big business. What is amazing about these numbers is that the Test Software and Industrial Automation Branches, while less than three percent of the depot workload, touch almost every part of depot operations. From Avionics Repair, to Jet Engine Test, to Aircraft Wiring and Nondestructive Inspection, there are very few repair functions on Tinker AFB that could complete their mission without software.

To support depot operations, the Test Software and Industrial Automation Branches are located in 12 locations on the base to provide the customers with on-site support. This has been done in a very conscious effort to ensure that our customers are getting the support that they need. It has been so effective that our customers generally call us when they have a problem, regardless of what it is, because they know that we will help them get results.

2 Our Software and Process Improvement History

2.1 Early Beginnings: B-1B and C-5B Test Program Set Development

While the beginnings of software process improvement for the Test Software and Industrial Automation Branches can be traced back to the early 1970s, the first major software development project was for 67 TPSs for the B-1B Bomber Aircraft, beginning in 1985. That project, which was completed within cost and on schedule, saw the first use of Earned Value Management and was an unqualified success. One of the most impressive accomplishments was that the Test Software and Industrial Automation Branches delivered the first B-1B Test Program Set, ahead of the competitors, some of whom had begun work as much as one year earlier.

In 1988, the Test Software and Industrial Automation Branches were awarded the contract to develop 54 TPSs for the C-5B Cargo Aircraft. While there wasn't a plan and direction to the process improvement efforts, the leadership saw the need for a defined process and organized training. Even without the Software Engineering Institute Capability Maturity Model® (SEI CMM®), the organization placed emphasis on the Level 2 and Level 3 key process areas (KPAs) [Paulk 93a, Paulk 93b]. The result was a successful project that was completed with a small cadre of experienced personnel, less than 10 percent, and a large number of newly hired engineers.

Both of these projects were very much Level 1 efforts, with hints of Level 2. They had their share of heroes and cowboys, but they were successful. They were a source of pride and reputation that a very young organization could build on for the future.

2.2 The Software Engineering Institute

In May 1989, the Software Division's Deputy Chief attended the SEI Symposium. He was intrigued by the process assessments and felt that an assessment could provide focus and direction to the organization's improvement efforts. Before it was in vogue and before the Air Force required it, the Test Software and Industrial Automation Branches embarked on our relationship with the SEI and the Software Capability Maturity Model.

® Capability Maturity Model and CMM are registered in the U.S. Patent and Trademark Office.

2.3 Process Improvement Infrastructure

Just as the process improvement efforts started before the Test Software and Industrial Automation Branches began their relationship with the SEI, so did the organization's process improvement infrastructure. In the late 1980s, a group of working-level people was established to oversee the first organizational process document. The group was to be totally autonomous with no interface to management. While this was a great idea, it just didn't work in the organization's culture. Without management interface, it is difficult to have management support. It wasn't maliciousness on anyone's part, but with 20-20 hindsight, we now understand that management support is key to improvement in any organization, especially at Level 1. However, sometimes mistakes are the best places to learn. Without this failure, some key decisions, such as creating the Management Steering Team, might have been made differently.

2.4 The Management Steering Team

Learning from the mistakes of the 1980s, the Test Software and Industrial Automation Branches knew that active management support and leadership would be key to the success of their improvement efforts. One of the first actions after the first process assessment in 1990 was to establish the Management Steering Team (MST), which was composed of the Deputy Division Chief and the Test Software and Industrial Automation Branch Chiefs. The Steering Team, which met monthly from 1990-1996, now meets every other month to provide direction and focus to the ongoing improvement efforts.

The initial MST meetings would often last up to six hours, and lively would be a polite way to describe them. The meetings were a strain not only on the managers' schedules, but also on their wills. Not only did the managers not always agree as a team, they had to struggle with some of the same cowboys and heroes who made the B-1B and C-5B projects a success.

But they persevered. One amazing fact is that with the exception of change in the management of one branch and two new members due to organizational growth, the MST membership has not changed since its inception in 1990—soon to be 10 years with the same basic leadership. This is pretty amazing and, while not quantifiable, the impact of the continuity of leadership cannot be discounted when looking at the organization's achievements. We have been very lucky to have leadership that has truly dedicated their careers to the organization and government service.

One key point should be made here. The MST has, over the past 10 years, taken extraordinary measures to work through people and let them mature into their roles. They don't replace people unless absolutely necessary; that has happened less than 5 times in 10 years, and when done, it is done in a manner that allows the person's skills to be best utilized. While this may not be the fastest way to improve, it is the most effective in the long run. People change because it is the right thing to do, not out of fear.

2.5 But Where Is the SEPG?

Read anything about process improvement and you will be told that the first thing that must be done is to establish a Software Engineering Process Group (SEPG). They will develop and drive the improvements. While we tried to conform in the traditional manner, the SEPG has never materialized. There is a Quality and Process Improvement Focal Point who is the project manager for the improvement efforts, and since 1996, there has been a metrics focal point, but there has never been a traditional SEPG.

Initially, in 1990, each branch named a SEPG representative, a charter was written, and meetings were held, but as it turned out, all the progress was made in the MST meetings. Again, it may be due to the organizational culture or a poor implementation, but the SEPG never had much of an impact. Additionally, while it was originally planned that the SEPG would work on the specific improvements, it was soon discovered that the SEPG didn't always have the subject matter experts and that others needed to be involved.

What has ended up being the norm for the organization is that subject matter experts are assigned by the MST to work on specific improvements. The experts are generally key personnel whose assignment to work on a software process improvement has an impact on areas larger than the project that they are assigned to. Often times, the assignment of these key people was very painful, but the MST made the commitment years ago to make the sacrifices needed to develop and implement improvements.

Additionally, as the organization has moved up the maturity scale, the improvements have become much more incremental, as opposed to revolutionary, and they are generally communicated through process updates.

2.6 Money Talks

The Test Software and Industrial Automation Branches are a government group, so why place an emphasis on funding? We have always been a fee-for-service organization with a great deal of attention paid to our costs by the OC-ALC financial management organizations. Our profit/loss situation and expenses are looked at monthly and our sales rates are adjusted each year. What this means is that our managers are trained to watch expenses and revenue.

In 1992, the Air Force began to provide funding for the process improvement efforts at a rate equivalent to approximately five percent of the organization's staffing level. This funding could be used to pay for labor, training, and travel. While that funding has been reduced in recent years to less than one percent, it is still vital. From the beginning, it has allowed process improvement to be viewed as any other workload in the organization. No one is asked to make improvements in their spare time—something that everyone knows does not exist.

2.7 Process Improvement Is a Project

No self-respecting software organization would embark on a development project without a plan. The SEI CMM says that is wrong. But how many groups begin process improvement without a plan? Maybe most don't, but the Test Software and Industrial Automation Branches did, and when asked to plan and schedule the process improvement efforts, the Quality and Process Improvement Focal Point initially balked. How do you plan something that you've never done, how will you know how long it will take? But, with a little more than a gentle nudge from the Deputy Division Chief, the first schedule was presented to the MST in October 1993.

Now this planning is a yearly occurrence. Each year, in October (at the start of the government's fiscal year), the Quality and Process Improvement Focal Point presents the yearly process improvement plan. Then status is reported at subsequent meetings throughout the year, using the same set of metrics that are required of any project in the division.

This approach has worked amazingly well. Planning the improvement efforts does many things. It scopes the efforts to the level of funding available. It gains agreements on the yearly goals, and it helps with the assignment of resources—or at least knowing the impacts when resources are not assigned.

3 Organizational Milestones

This section describes the organizational milestones that the Test Software and Industrial Automation Branches have realized from 1990 to the present.

3.1 Level 1: The Beginning

Everything has to have a beginning. In 1989 the Deputy Division Chief decided that the Test Software and Industrial Automation Branches would undergo an SEI process assessment. There were a few issues in planning the assessment. The largest issue was that the employees of the Test Software and Industrial Automation Branches were very young, and, consequently, very few met the SEI's requirements for team member experience. Even worse, the employee chosen to lead the team had only four years of experience. Additionally, though the SEI sent the most wonderful person to coach the team (Louise Hawthorne), she is a Canadian citizen and, at the time, she was not allowed on base due to Air Force security precautions. So**,** much of the assessment took place in the apartment of one of the team members. Then, to top it off, the organization was a solid SEI Level 1, a result that wasn't truly expected by the Deputy Division Chief and his new boss. After all, many great things had happened on the B-1B and C-5B projects, surely the organization was at least Level 2!

However, from that rocky start, the assessment generated the initial set of findings that kicked off the organization's formal process improvement efforts. We were on our way.

3.2 Level 2: Proof That We Are on the Right Track

In late 1992, it was decided that it was time to reassess. The goal was SEI Level 2, a milestone that had not yet been reached by an Air Force organization.

At this time, a series of events unfolded, which in hindsight were wonderful decisions but were a little unnerving at the time. The SEI Software CMM was just being released and the SEI wanted to test their new assessment process that was based on the Software CMM. Mary Merrill from the SEI contacted our organization and requested to use us as the alpha site for the updated assessment process. Seeing this as a bit of a risk, but one with a potentially very high return, the Test Software and Industrial Automation Branches agreed.

So, in March 1993, eight SEI personnel, along with two SEI observers, arrived at Tinker AFB to perform *the first Software CMM-based assessment*. The Software CMM V2.0 had just been released a month before, and the assessment process had grown from one to two weeks.

Furthermore, to gather a greater amount of data, the SEI Empirical Methods Group requested that the entire organization complete the questionnaire.

The end result was SEI CMM Level 2, the first in the Air Force, an incredibly proud organization, a new set of findings, and, most importantly, proof that we were on the right track.

3.3 SAF/AQK Return on Investment Study

In 1994, Lloyd Mosemann, Deputy assistant secretary of the Air Force for communications, computers, and logistics (SAF/AQK), decided that the best way to make a business case for process improvement was to perform an independent study. The study was to determine the benefits of process improvement in terms of return on investment. The Test Software and Industrial Automation Branches were chosen for the study due to the achievement of SEI CMM Level 2. The data gathering and analysis was performed by Software Productivity Research (SPR). To our knowledge, this is the first and only independent verification of SEI CMM-based software process improvement that has ever been performed. The outcome, at that time, was that we had seen a 7.5 to 1 return on investment (ROI) as a result of our process improvement efforts from 1990 to 1994. This was determined by baselining to a pre-improvement project and analyzing three subsequent projects to determine their costs if no improvements had been implemented. The study was very significant in that it was a view, independent from the SEI and the Air Force, that showed the cost benefits of process improvement. It showed that there was more to process improvement than obtaining a CMM level—that process improvement could positively affect the financial bottom line!

3.4 Quality Air Force Assessment

In May 1996, the Test Software and Industrial Automation Branches were awarded an Outstanding rating, the highest possible, by the Air Force team sent to Tinker AFB for the Quality Air Force Assessment. This was not only great for the organization; it was a wonderful boost heading into the 1996 SEI CMM assessment.

3.5 Vice President Gore Hammer Awards

Vice President Al Gore established the “Hammer Awards” as part of the reinventing government initiative. These awards are named for the $1000 hammers for which the government was grossly overbilled. The award is a shadow box with a hammer and note from Vice President Gore. In September 1996, a group from the Test Software and Industrial Automation Branches won the Hammer Award for the State of Oklahoma for their efforts in pioneering the way for the introduction of new TPS technologies. Their efforts not only benefited their customer, but contributed to the $220 million savings that the Test Software and Industrial Automation Branches helped the B-2 program realize for their TPS efforts. In 1997, another project group placed second in the competition for their efforts in jet engine diagnostic trending. Once again the Test Software and Industrial Automation Branches’ efforts were honored.

3.6 Level 4: Now, Improvement Begins

Three and a half years after the organization achieved Level 2, it was time to reassess. The assessment date was November 1996. It is important to note that this assessment occurred two years prior to the 1998 SEI CMM Level 3 achievement goal established for Air Force organizations and contractors. The target was Level 3, but, while the key process areas for Level 3 were in work, there had also been significant progress toward Levels 4 and 5.

The reason for the significant progress in Levels 4 and 5 was that the organization struggled with Organization Process Definition (OPD). Even though there were earlier process documents to start with, OPD was the most difficult KPA for the Test Software and Industrial Automation Branches to address.

We struggled with the format of our process document, eventually settling on a variation of a process model termed Entrance, Task, Verification, and Exit (ETVX), using inputs, activities, products, and reviews. We also struggled with getting people focused on the writing efforts, with the final solution being the relocation of the authors (i.e., the subject matter experts) away from their assigned work location, until their assigned writing tasks were complete. The authors were relocated near the focal point for process improvement which greatly facilitated communication and focus on the task at hand. While the final solution, (or TPS Life Cycle Guide) has been an unqualified success, it took us much longer to satisfy OPD than we originally planned. While this was a bit of an annoyance at the time, it did allow for more progress to be made on the Level 4 and Level 5 KPAs.

Judah Mogilensky of Process Enhancement Partners and Michael Reed of the Air Force Communications Agency (AFCA) led the assessment. At the time, the AFCA was the focal point for software process improvement in the Air Force. The other six members of the team were from the Test Software and Industrial Automation Branches.

Seemingly, all assessments, as they progress through the two-week process, come down to one KPA. In 1993, it was Software Quality Assurance (SQA). In1996, it was Quantitative Process Management (QPM). The team struggled with QPM and what it meant to the Test Software and Industrial Automation Branches. The final solution for the team was to put a “draft” finding up during the final findings dry run and see the reaction from those being assessed. The reaction was overwhelming—the people of the Test Software and Industrial Automation Branches were adamant in the usefulness and institutionalization of the measurement program.

The result was SEI CMM Level 4, with two KPAs, Technology Change Management (TCM) and Process Change Management (PCM), at Level 5 also satisfied. This result, while secretly hoped for, nonetheless thrilled the Test Software and Industrial Automation Branches and our leadership. We had done something that had not been done by any government group and by a relatively limited number of private industry organizations worldwide. We were very proud when Mr. Mogilensky told our commander, Major General Perez, that he had a “world class” software organization.

Even so, we still had a finding: Defect Prevention was not satisfied, and, to further temper our joy and remind us to stay focused, Mr. Mogilensky told the organization, “Congratulations, you’ve achieved Level 4, almost Level 5. *Now*, you are ready to begin process improvement.”

He was right. We had worked for over six years to get our processes in place along with the appropriate measurements. Only now were we really ready to begin data-driven improvement.

3.7 ISO 9001/TickIT

After Level 4, what’s next? Never ask a goal-oriented group of managers “what’s next.” They’ll answer.

Toward the end of 1996, the idea of International Standards Organization (ISO) 9001/TickIT registration began to gain support, not only as a Tinker AFB goal, but also due to the Test Software and Industrial Automation Branches’ Foreign Military Sales customers. Fundamentally, it was another step that the organization needed to take to stay competitive.

TickIT is an additional accreditation for software organizations. TickIT auditors are required to take an additional examination and face a board to obtain this auditor credential. Essentially, with a TickIT auditor, you know that your auditor will understand software and how to audit a software organization. With our auditor (British Standards Institution, Inc.) software organizations are assured that the lead auditor, at minimum, is TickIT qualified.

To be very honest, the Test Software and Industrial Automation Branches initially looked at ISO 9001/TickIT as a box to check. We were SEI CMM Level 4, we did great work; what could ISO 9001/TickIT offer us? The answer: plenty.

ISO 9001/TickIT forced us to take a broader view of our improvement efforts and tighten up some loose ends. ISO 9001/TickIT is shallower than the SEI CMM, but much broader in scope. The ISO 9001 standard causes a software group to look further into their hardware business, if they have one (which we do), and it places a greater emphasis on the customer. Also, ISO 9001 requires, at minimum, twice-yearly surveillance audits by the Registrar.

ISO requires that a quality manual be written. Essentially, the quality manual depicts how the organization’s processes and procedures implement the “shall” statements that make up the 20 clauses of ISO 9001. Developing the quality manual showed us several areas where we were lacking. One area was Contracts, which takes the Requirements Management KPA a step further, and another was Corrective and Preventive Action, which fits in with the Defect Prevention KPA. ISO 9001 also requires an increased awareness of customer complaints.

Additionally, the twice-yearly registrar surveillance audits are incredibly beneficial. First, they help keep the organization on track with regular external audits. This is important because, even with the best intentions, it is still easy to occasionally lose focus. [As an aside, this was a primary focus of the IEEE Award Review Team. We were told that it is not uncommon to see back-sliding after an assessment, even at the higher maturity levels]. Second, our primary auditor, Chuck Herold, has an uncanny ability to find our weaknesses and expose them, so that we can correct them. The bottom line is that we believe in ISO 9001/TickIT; it has benefited us and it has helped maintain enthusiasm in the improvement efforts.

On 12 November 1998, the Test Software and Industrial Automation Branches were officially registered to ISO 9001/TickIT; we have subsequently had our registration reaffirmed three times, once every six months.

One additional thought, ISO 9001/TickIT has helped prepare the organization for the new CMM - Integrated - Systems/Software Engineering (CMMI[SM]-SE/SW) since that too broadens the scope of the improvement efforts [CMMI 00].

3.8 The IEEE Award for Software Process Achievement

We did it! We won the 1999 Institute of Electrical and Electronics Engineers (IEEE) award, which, in turn, is the reason for this report. We had submitted nominations for three years before winning. Each time we would get a polite letter telling us that while our improvements were impressive, there was not enough data to support us winning the award. But we persevered, each year using the IEEE committee's comments to improve our nomination.

After three years of trying, you can imagine our joy when Bill Riddle of the SEI called in December 1998 to tell us that we were a finalist. We can truly say that we were honored just to be named a finalist. That joy quickly turned to panic when Bill informed us that Dr. Vic Basili, Dr. Barry Boehm, Manny Lehman, Bill Riddle, and Watts Humphrey would visit in March 1999 to evaluate us. Could this be true? Would the icons of our industry really be coming to spend the day with us?

To help us plan for the visit, the committee sent five "issues" that they wanted us to address. The required 12-page format of the award nomination package very much limits the volume of what is written. The IEEE committee used the "issues" to examine the depth and credence of our nomination. To address the issues, our Deputy Division Chief essentially focused all of his time for two months to coordinate the response. We prepared 120 slides with over 60 back-up charts. *It took four hours to address just the first issue,* with another two hours to complete the other four.

[SM] CMM Integration and CMMI are service marks of Carnegie Mellon University.

The visit went wonderfully. The committee, who were all in attendance except Dr. Basili, quickly erased our initial nervousness. They went out of their way to make us feel at ease. Still, it was truly amazing to look around the room and see that group of esteemed gentlemen and listen to their comments and insights.

The day went great, and we anxiously awaited the committee's decision. To our delight, Bill Riddle called the next day to say that the committee, including Dr. Basili, unanimously selected us.

So, on May 20 1999, the Test Software and Industrial Automation Branches were named the fifth ever winner of the IEEE Award for Process Achievement in the six years that it has been in existence. This was truly wonderful!

4 Major Improvements

The Test Software and Industrial Automation Branches have implemented many improvements over the years, more than 90: some large, some small, many still in use, some not. Some improvements just don't have staying power and others are outgrown by the organization. This section will outline the most significant improvements implemented by the Test Software and Industrial Automation Branches. Many of the charts shown to the IEEE Award Review Team are included in this and subsequent sections.

4.1 Organizational Process Definition

As was mentioned earlier, this was perhaps the most difficult improvement ever undertaken by the Test Software and Industrial Automation Branches. The final result was our organizational process document, the TPS Life Cycle Guide, which has spun off two subsidiary documents: the Industrial Automation Maintenance Guide, and the Engine Trending and Diagnostics Guide. Additionally, a third document is in work at this time, the Information Technology Guide.

There were many reasons for why this particular improvement was such a struggle. Perhaps the most significant lesson that came from the TPS Life Cycle Guide development was that people have to be assigned full time to work on improvement efforts. "Part-time" didn't work. People spent the funding, but products did not emerge. Real progress was made only after people were relieved of their day-to-day obligations and moved to a quiet location to work on the guide. Another important point is that the people assigned to work on the guide were key personnel within the organization; more than 10 subject matter experts were assigned writing tasks. Perhaps an even greater indication of the importance of this effort was that one of the Branch Chiefs took on the editing duties, visibly showing the importance of this effort.

Finally, even though the Test Software and Industrial Automation Branches had existing development, maintenance, and program management guides to draw from, the format of this new all-inclusive document was a struggle. After many high-tech and intricate methods of process description were explored, the final document was written with a word processor in an ETVX format, the format being Inputs, Activities, Products, and Reviews.

This solution was both simple and lasting. Version 3.0 of the guide was just released this past year.

4.2 Cost and Schedule Estimation

The Test Software and Industrial Automation Branches have taken great pride in the cost and schedule estimation efforts since the mid 1980s. This is truly an area that has seen vast refinement and improvement.

Today cost and schedule estimating (Figure 3) is based on a Work Breakdown Structure (WBS) (level may vary depending on type of estimate) and is facilitated by our TPS Life Cycle Guide (LCG) or one of the derivative documents. The project scope is defined by the customer's Statement of Work (SOW), with a top-down approach for small jobs and requiring only rough order of magnitude (ROM) quotes and a bottom-up approach for formal proposals.

The Test Software and Industrial Automation Branches' principal cost and schedule drivers for development efforts are

- risk (management reserve - both cost and schedule)
- requirements (specifications and standards)
- staff availability (existing personnel as well as the challenges of hiring and retaining personnel in today's economy)
- complexity
- experience
- historical performance used for existing technologies
- adjustments for new/emerging technologies

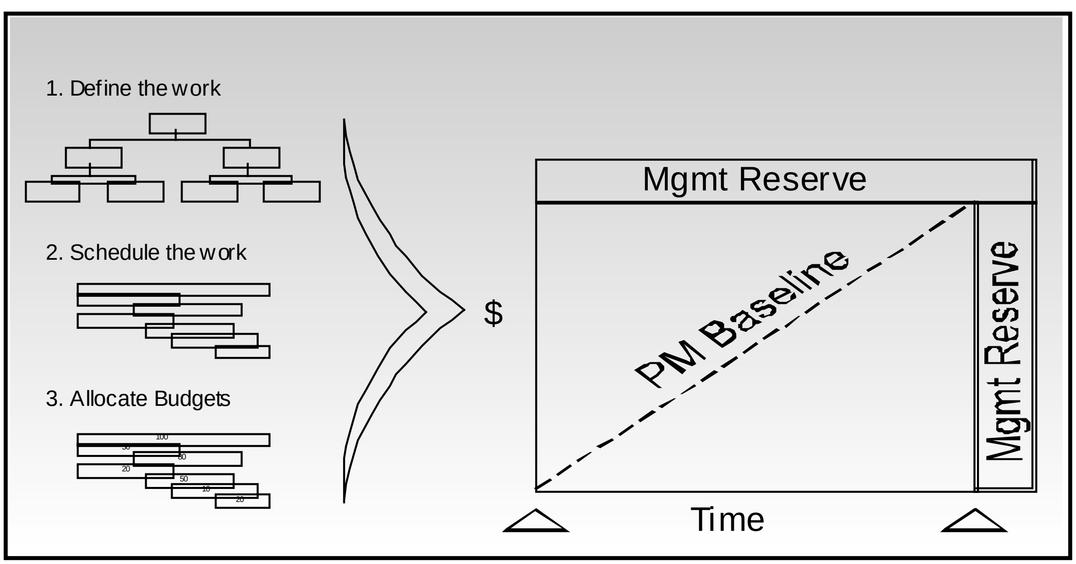


Figure 3: Cost and Schedule Estimating

An important question to be answered is "How good are the development estimates?" For our recently completed projects, we have generally been within +/- five percent on both cost and schedule. But, we have also seen that we are most likely to have the largest error in the schedule. So, new projects are placing more emphasis on planning and managing schedule reserve.

For maintenance the cost and schedule estimates are similar to development (see Figure 4):

- Estimates are task/WBS based.
- Task hours are drawn from established standards by deficiency type.
- Established standards are built by experience.

Maintenance Cost & Schedule Estimates (cont.)

LASAA C-141 IR/SDR WBS	SRU (DATSA)	LRU (607)	PROJECT HRS	NOTES
Auto Throttle SRU A2, 722-0420-002			**363**	Alan Tran
INVESTIGATION REPORT	48	68	**123**	
Assign & Investigate IR	40	60	115	98-IR-05529
IR review	8	8	8	
SOFTWARE DEFICIENCY REPORT			**240**	
Assign SDR & Setup Project	4	4	**4**	
Rewrite Code			**46**	
Rewrite procedures	8	8	8	Add re-calibration procedure
Rewrite ITA Tests	4	4		
Rewrite ID Test	4	4		
Rewrite STTO/Stim Wrap Around Tests	4	4		
Rewrite Static Tests	4	0		
Rewrite Performance Tests				
Adjusted Voltage	6	10		
Fixed Voltage	4	4		
Time/Frequency	4	8		

Figure 4: Cost and Schedule Estimates

Our maintenance projects, which are generally short in duration (200-700 hours) are consistently within +/- 8 percent cost/schedule variance. Additional data and discussion appear in Section 7 (Benefits to Us and Our Customers).

The use of measurement for project tracking and oversight is discussed in further detail in Section 6 (Measurement).

4.3 Evolution of Quality

This is perhaps one of the more interesting stories that the Test Software and Industrial Automation Branches have to share. Quality has been driven from many different directions throughout the years. From the Air Force, to the SEI, to ISO 9000, to the organizational needs, to customer requirements, the quality system used by the Test Software and Industrial Automation Branches has most definitely seen major evolution and change.

Starting in the late 1980s, the Air Force decreed that quality would be built in, that it would not be inspected in, and for this reason, all separate quality organizations would be abolished. To accommodate this direction as well as implement SEI CMM Level 2, the Test Software and Industrial Automation Branches developed an ETVX approach to quality assurance. During the 1993 SEI CMM assessment, a great deal of discussion was focused on the fact that the Test Software and Industrial Automation Branches did not have a separate quality assurance group. When the smoke cleared, the assessment team, which was composed primarily of SEI personnel (prototyping the new assessment method), determined that the approach of building quality into the process satisfied the KPA as an alternative practice. Many on the assessment team felt that the Test Software and Industrial Automation Branches had implemented an effective quality system similar to what they had seen in higher maturity organizations. One important point was that the organization was much smaller at this time, approximately 180 people, which is half of today’s size.

Between 1993 and 1996, another major step was taken in the organization’s quality evolution. Due to the increased emphasis on project management and measurement as well as the technical aspects of the project, the Test Software and Industrial Automation Branches’ projects started to have dual project leads. One lead focuses on the customer interface and project management, while the other lead focuses on quality and process. This was the “State of the Test Software and Industrial Automation Branches” practice for the 1996 assessment and continues today. Quality is still built in, but there is a much greater emphasis on project and defect prevention.

So, where does quality stand today? One hard-and-fast requirement of ISO 9001 is that the organization have an independent SQA function, although there is great leeway in the standard as to how that is done. As the Test Software and Industrial Automation Branches grew, to over 350 people today located in 12 locations on Tinker AFB, there was also a need for an oversight function to ensure process adherence and the gathering of best practices. To accomplish this oversight as well as adhere to the requirements of ISO 9001, the Test Software and Industrial Automation Branches have implemented yearly process audits. Led by the Quality and Process Improvement Focal Point and supplemented by personnel from the branches, annual process audits are performed on each project.

Additionally, the organization’s ISO 9001/TickIT Registrar audits each project at least once every two years. Larger projects are audited more often depending on the size of the project and how it is staffed within the organization. Our largest projects, B-1B and B-2 TPS, have portions audited every six months, with a complete audit of the entire project once every two years.

4.4 Evolution of Technology

Technology change and evolution has to be performed in a manner that benefits the organization. The Test Software and Industrial Automation Branches work very hard to standardize and evolve the hardware and software used for TPS development. This has to be done on two fronts: (1) as with all organizations, we need to stay abreast of the latest in technology, and (2) we must also focus on another, more difficult area—standardization across customers and weapon systems. In general, we have a greater knowledge of new technology and a broader perspective across the several weapons systems that the Branches support. Where there are opportunities for equipment and software standardization on technology upgrades, we provide those recommendations to our customers. We continually work with our customers to ensure that they are doing what is best for their systems, this is an aspect that we have focused on for years.

Areas that the software division focuses on include programming languages, digital simulators, automatic test systems, engineering workstations, and documentation tools. All of these areas have seen major change as we have evolved from a mainframe environment to personal computer (PC) workstations, as test hardware has gotten smaller, as languages have progressed from third generation to graphical, and as documentation has evolved from paper Technical Orders (TOs) to online. From 1985 to present, the technology changes have been phenomenal and we know that the future holds even more drastic change, which the organization will have to embrace and build upon.

Figure 5 below shows the progression of TPS programming languages from a Pascal-based, third-generation language in the 1980s to graphical programming languages with much greater capabilities today. The improvements in programming languages are key to our organization. Although we are a software group, we hire electrical engineers because those are the skills that are needed to determine the optimal methods to test avionics, which is our primary workload. By moving from "programming" to a "point-and-click" environment, we are able to make much more efficient use of the engineering skills and keep their focus on testing as opposed to programming.

Programming Languages

LAS continues to utilize the latest software technologies to improve software development time and supportability.

1985 **MATE/ATLAS**
- Pascal Based Programming Language
- Compiled Programming Language
- Closed Architecture

1991 **Hewlett-Packard Visual Engineering Environment (HP-VEE)**
- Graphical Programming Language
- Simplified debugging and error handling
- Improved performance/Open architecture

1996 **Program Guide and HP-VEE**
- Integrated HP-VEE with Teradyne Program Guide
- Integrated program development utilities
- Automated process Control

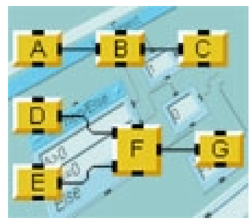


Figure 5: Technology Evolution

4.5 Improvement of Suppliers—Incremental Improvement

Any organization that is serious about process improvement soon finds that they have to help their suppliers improve if they are going to continue to see improvements in themselves. We found that the only way we could make some of our major cycle-time improvements was to work with our suppliers to improve their responsiveness. The example that we used for the IEEE Award reviewers concerned our B-1B TPS Maintenance process. Our measurements showed that we had two supplier areas making major, negative impacts to our cycle time and causing frustration not only to us but, more importantly, to our customers. The areas were configuration control and software distribution. For configuration control, the Test Software and Industrial Automation Branches established a Memorandum of Agreement, or a contract, with the configuration control organization, to speed the assignment of software revision numbers. Essentially we raised the level of management attention, showed the supplier how they were impacting us and our customers, and were able to improve the situation drastically.

Another improvement involved Time Compliance Technical Orders (TCTOs), which are required by B-1B TPS Maintenance projects when changes are made to interfacing hardware. The TCTOs provide the instruction and authority for Air Force personnel located at the operating bases to implement configuration changes to the interfacing hardware. Without the TCTOs, our software and hardware revisions were not allowed for use by the field. So, even though we had been responsive and made the revisions, our customers were not seeing our efforts. Knowing this was happening caused great frustration on our part as well as in the field. By working with the configuration manager to improve TCTO timeliness, we were able to reduce the distribution time greatly, from as long as one year to two months (see Figure 6).

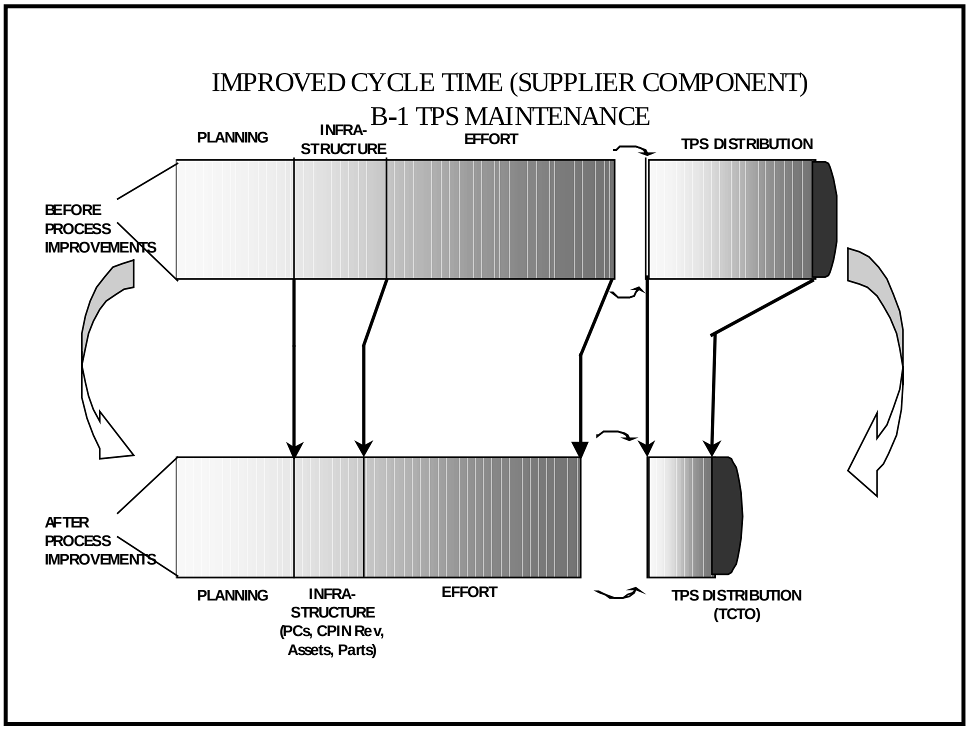


Figure 6: Cycle Time Improvements

4.6 Reduction of Backlog

It is important to anticipate customer needs and issues. One key area is responsiveness. It is an attribute that customers understand very well, so, even when a customer doesn't specify his responsiveness requirement, we try to assess what is reasonable. We realize that sometimes customers may not know what they want, but, if they don't get it, they become disgruntled and may take their business elsewhere without ever expressing their reasons why. So, we try to stay alert for potential problems.

In general, responsiveness is related to "backlog." If backlog is large with respect to the work flow rate, then there must be a lot of old work and little hope of being responsive. This was our case for both investigating and correcting problems that, once investigated, turn out to be related to the software. We recognized this, set internal goals, and, as is shown in Figure 7, Reduction of Backlog, greatly improved our responsiveness to the customer.

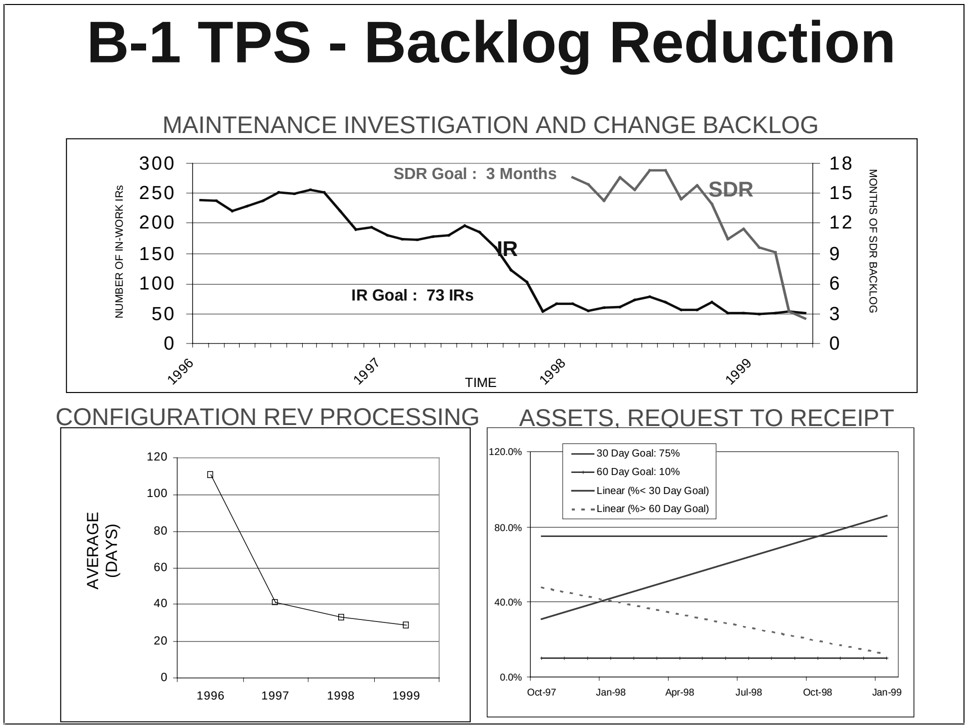


Figure 7: Reduction of Backlog

To make the necessary responsiveness improvements, we had to examine Cycle Time (CT) which has a dependency on supplier inputs. For us to complete the necessary software corrections, a configuration management process must occur to update the revision of the software, “Configuration Rev Processing” on Figure 7. Also, the item to be repaired, generally a circuit board or black box, referred to as “Assets” on Figure 7, must be available for integration and customer acceptance testing. The bottom line is, we could not have improved our responsiveness without improvements in the configuration management process as well as improvements in our ability to obtain the “assets.” The point is, we realized that you can only get so far looking internally for improvements. We examined our external dependencies too. What we found is that our suppliers can oftentimes help and they are generally willing once the problem has been identified to them.

4.7 Redefining Process and Products—Radical Change

By taking advantage of technology innovations, we have been able to make some radical changes in our products and processes. Our B-2 TPS Development project was able to take advantage of several radical changes allowed by technology, culminating in reduced cycle time. The three areas where significant impacts were made were electronic documentation, media-CD jukebox, and concurrent TPS prototyping.

As an example, one of the changes made was the elimination of paper Technical Orders and the move to electronic documentation. This was accomplished by integrating the user instructions, the Technical Order, into the software, along with the schematics and other data used in the repair of the B-2 avionics. Correspondingly, cycle time improvements are seen in both TPS development and maintenance, and software distribution is vastly improved because it can now be performed electronically through the computer networks. Also, because user instructions were available significantly earlier in the development process, training and operational feedback occurred much sooner than it ever had in the past. The application of new technology allowed the consolidation of three paths into a single path, thereby reducing management requirements, and more significantly a reduction in the fielding of each TPS by 6 to 12 months.

Figures 8 and 9 illustrate the changes and impacts on the B-2 TPS process.

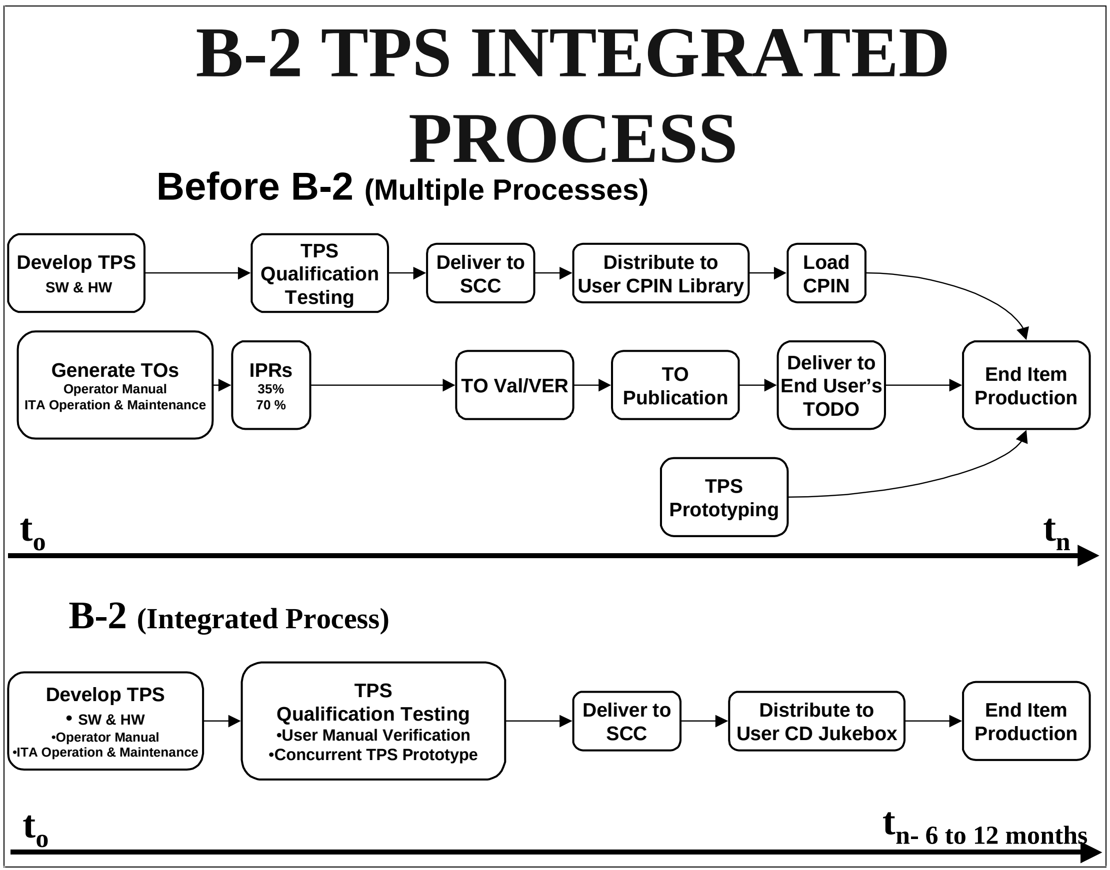


Figure 8: B-2 Integrated Process

B-2 INTEGRATED PROCESS

CHANGED PRODUCT	ADVANTAGES	DEVELOPMENT IMPACTS	MAINT IMPACTS
ELECTRONIC DOCUMENTATION	TPS DOCUMENTATION TRACKED WITH CPIN ELIMINATE FORMAL TOs CONCURRENT TPS PROTOTYPING	ELIMINATE TO INTERFACE/PROCESS RELEASE WITH TPS (6-MONTH SAVINGS)	TCTO REQUIREMENT ELIMINATED RAPID TPS DISTRIBUTION TO USER
MEDIA – CD JUKEBOX	INTEGRATED WITH SYSTEM TIGHT CONFIGURATION CONTROL ON-LINE DOCUMENTATION	NOT APPLICABLE	REDUCED SET-UP TIME VIRTUALLY ELIMINATES USE OF NON-CURRENT SOFTWARE
CONCURRENT TPS PRTOTYPING	IN-PROCESS TRAINING TO PRODUCTION PERSONNEL PARALLEL PRODUCTION AND TPS DEVELOPMENT EARLY USER FEEDBACK	SHARED PRODUCTION DEVELOPMENT RESOURCES TEMPLATE MODIFICATIONS ERRORS IDENTIFIED EARLY	NO STARTUP TRAINING ($100K) EARLY PRODUCTION CAPABILITY (3 YRS +) FEWER MAINT ACTIONS

Figure 9: B-2 Product Changes

5 Organizational Improvement Data

The total number of improvements implemented since our process improvement efforts began in earnest in 1990 are tabulated in Table 2.

These data reflect several things. First, as the data show, a Level 1 group will try many things and, while they may have short-term benefits, many of the improvements will not have staying power. A great deal of our early improvements, from 1990 to 1993, have been overcome by technology or by our own process definition and streamlining. Additionally, many of the improvements focused on a specific project because we did not do things consistently across the organization and, at the time, “not invented here” was often a problem. Both of these characteristics can be expected for a Level 1, and even a Level 2, organization. This isn’t meant to discount our early efforts; they were very important and they set the stage for our successes. However, it is very important to realize that not every improvement will last forever, especially in the beginning.

Timeframe	# Improvements Implemented	# Still in Place in 1999	Percent
1990-1993, Level 2 in 1993	45	11	**24%**
1993-1996, Level 4 in 1996	31	24	**77%**
1996-Present	22	22	**100%**

Table 2: Improvement Data

From 1993 to 1996, our focus was on process definition and bringing the organization together as a whole in both process and measurement as well as training. As Table 2 shows, the staying power of the improvements improved drastically from 24% to 77%. These improvements focused on Level 3 and 4 KPAs and were much better planned and managed. We were moving from the obvious quick gains to more lasting change within the organization.

From 1996 on, the number of improvements has dropped drastically. As was cited by the IEEE review team, the “easy-to-obtain” improvements are complete, and we must now focus on the “hard-to-obtain” gains. What we are seeing now is much less “revolutionary” improvement and much more “incremental improvement,” as would be expected in a higher maturity organization. The majority of the 10 improvements cited, with the exception of ISO 9001 registration, focus on block revisions to our standard processes and our measurement program. We have built a living, evolving process as well as a method for introducing change in an orderly manner. The organization continues to change, but in more subtle ways.

6 Measurement

6.1 Evolution of Measurement in the Organization

Nothing stays the same. If something is useful, then it will change as the organization matures and reacts to changes. The organization's process will change and so will the measurement program. The Test Software and Industrial Automation Branches have been measuring and reporting project progress since the early 1980s, but those measures have been drastically refined and improved, especially since 1995.

6.2 Management Reporting Indicators

The measurement set that each branch reports monthly is called the Management Reporting Indicators (MRI). The measures that are reported for both software development and maintenance are funds management, cost and schedule status, delivery trends, productivity, and rework. Additionally, for maintenance we track the status of the investigations as well as the Software Deficiency Report, or software correction, backlog. In general, maintenance actions are a roll-up of several maintenance actions, whereas the development indicators are project specific.

The funds-management data consist of several Air Force-specific items such as how we are charging our time and leave usage. Also we monitor the status of our customer funds, how much money they plan to obligate to us, and how much has been obligated and expended. We also monitor overtime and the unexecuted portion of our funding (i.e., how long could we work if no more funding was received).

Each individual project also reports cost and schedule status, using Earned Value Methods. We also show delivery trends. Are the products, generally TPSs, being delivered on schedule? This is important because development projects consist of multiple TPSs and we deliver the TPSs as they are completed; we don't wait until the end to deliver the entire set of products. This method allows the customer to begin using the TPSs considerably sooner. Productivity is monitored showing both effort and cycle time; likewise, rework is reported for all the effort expended as the result of the correction from either internal or formal customer reviews and functional testing. Additionally, our maintenance projects report their backlog (man-hours of effort) of problem investigations and resolutions.

Red, yellow, green indicator ranges are set for each indicator and goals are set yearly. Explanations and corrective actions are required when an indicator goes to yellow or red. Annual baselines are established from past performance along with management goals.

6.3 Management Reporting Indicator Data Flow

Each level of management, from project leader to the Deputy Division Chief, needs a different granularity of data. The project engineers provide data for complete and in-work tasks. The project leader prepares a project status report by aggregating the data of the project engineers, which is given to the section chief. Monthly, the section chief assembles the MRI data for all of the separate efforts in his section and, subsequently, provides it to the branch chief who, in turn, similarly prepares the monthly MRI briefing for the Deputy Division Chief. Each level sees a roll-up representation stemming from the data provided by the project engineers.

There is one exception to the general flow description. Development projects are not aggregated together as the data are assembled at higher levels in the organization. Each development project stands on its own with respect to funds management, cost, schedule, and delivery status.

6.4 Infrastructure Tracking

When the B-2 TPS Development project began, we quickly realized that the set of requirements for planning, including building the infrastructure, was actually a project in itself. The project was much larger than anything we had ever done before. For quite some time, the Test Software and Industrial Automation Branches have tracked the planning of projects, including delivery of the planning documents, but B-2 was still of a magnitude that had never been attempted. Over $2 million was budgeted just to plan the project.

As shown in Figure 10, the infrastructure areas planned and tracked are staffing, training, facilities, automatic test equipment, organization, support software, equipment, contracts, internal standards and tools, data, and long lead time parts. Three years into the project, the infrastructure items are still tracked and reported monthly to ensure that they are healthy and that they will be addressed if, for example, issues develop in staffing or facilities. This process, which developed out of necessity for B-2, was seen to have broad application and has been added to our organizational process.

INFRASTRUCTURE

G

ELEMENTS			OCT	NOV	DEC	JAN		FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
			FY99												
STAFFING	COMPLETE 10/96	G	C	C	C	C	→	Y	Y	R					
TRAINING	COMPLETE 03/97	G	C	C	C	C									
FACILITIES			Y	Y	Y	Y	→	Y	R						
ATE			B	B	B	B									
ORGANIZATION			Y	Y	Y	Y									
ASSETS			Y	Y	Y	G									
SPT SOFTWARE			B	B	B	B									
EQUIPMENT			B	B	B	B									
CONTRACTS			B	B	B	B									
INT STDS/TOOLS			B	B	B	B									
DATA			B	B	B	B									
LONG LEAD PTS			B	B	B	B									

→ PROJECTION

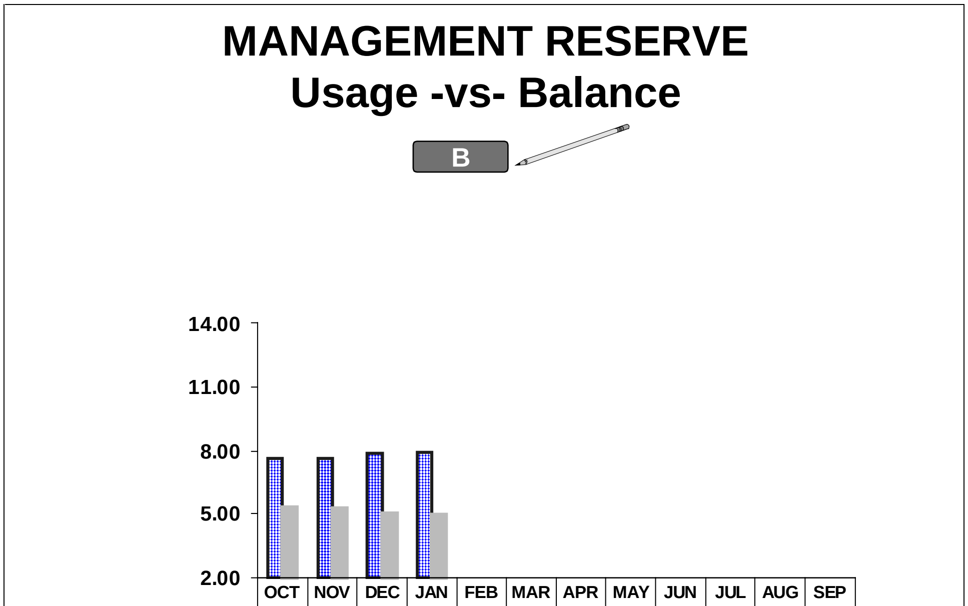
B 3 Consecutive months of Green G No Concerns Y Potential Impacts R Impacts

Figure 10: Infrastructure Status Tracking

6.5 Management Reserve

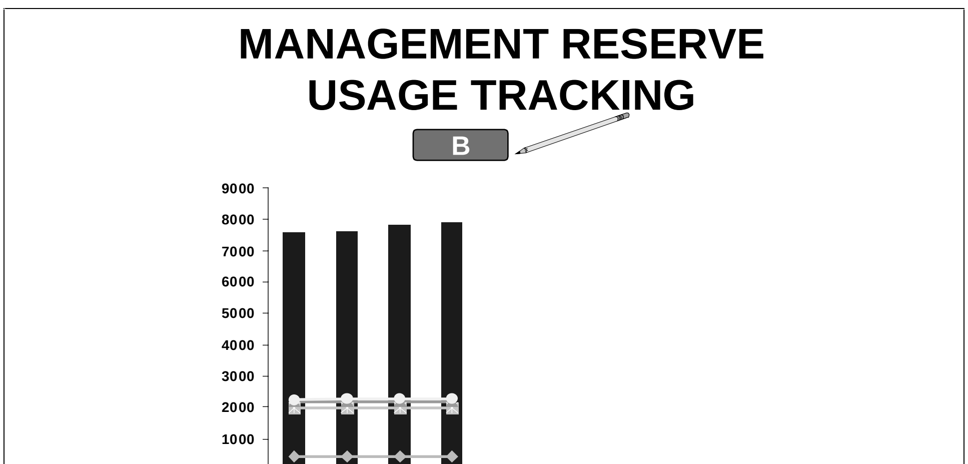
All of our projects are planned with management reserve (MR), but again, B-2 was a first with a very large sum allocated to MR; in fact, the MR was more than it takes to complete many of our smaller development projects. What was even more interesting is that the customer insisted on the large MR. The customer foresaw more risk than what we understood or considered. We are guessing, but we believe this was due to the customer's past software development experiences or lessons learned from other programs.

Given this large MR, it was vital that we tracked and reported the MR status and usage. The following two figures were developed for the B-2 project reporting, the MR usage and balance (Figure 11) and a breakout of the MR usage (Figure 12). These indicators were refined during a prototyping period and then were incorporated into our process. They are now used for all of our development projects.



	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
CUM USAGE	7.58	7.62	7.85	7.90								
CUM BALANCE	5.30	5.26	5.03	4.98								

Figure 11: Management Reserve Tracking



	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
MR USE TOT	7582	7620	7853	7905								
INITIAL DELAYS	1995	1995	1995	1995								
NON-SPO	2188	2188	2188	2188								
ATE	2245	2251	2254	2254								
ESCORT DUTIES	329	329	329	329								
FACILITIES	126	127	367	370								
MOVE	421	421	421	421								
RESEARCHING DATA	229	249	260	271								
GFE DEFICIENCIES	9	9	9	9								
TPS COST PROPOSAL	40	51	61	66								
CONTRACT EVAL	0	0	0.5	2								

Figure 12: Management Reserve Usage

6.6 Cost Indicators and Schedule Performance Indicators

The most recent measures added to our metrics system and monthly review are the Cost and Schedule Performance indicators. These indicators are defined as the Cost Ratio vs. the inverse of the Cost Performance Index and the Schedule Ratio vs. the Inverse of the Schedule Performance Index (see Figure 13).

The ratios are derived from the customer expectations versus the project plan, while the cost and schedule performance indices come from the body of management known as “Earned Value Management.”

As long as the inverse of the cost and schedule indices are below their respective ratios, then the project can be completed within the budget; however, if either rises above its ratio, then management actions need to be taken as suggested in Figure 14. These concepts are described in much greater detail in the March 1999 *Crosstalk* article, “Applying Management Reserve to Software Project Management” [Lipke 99].

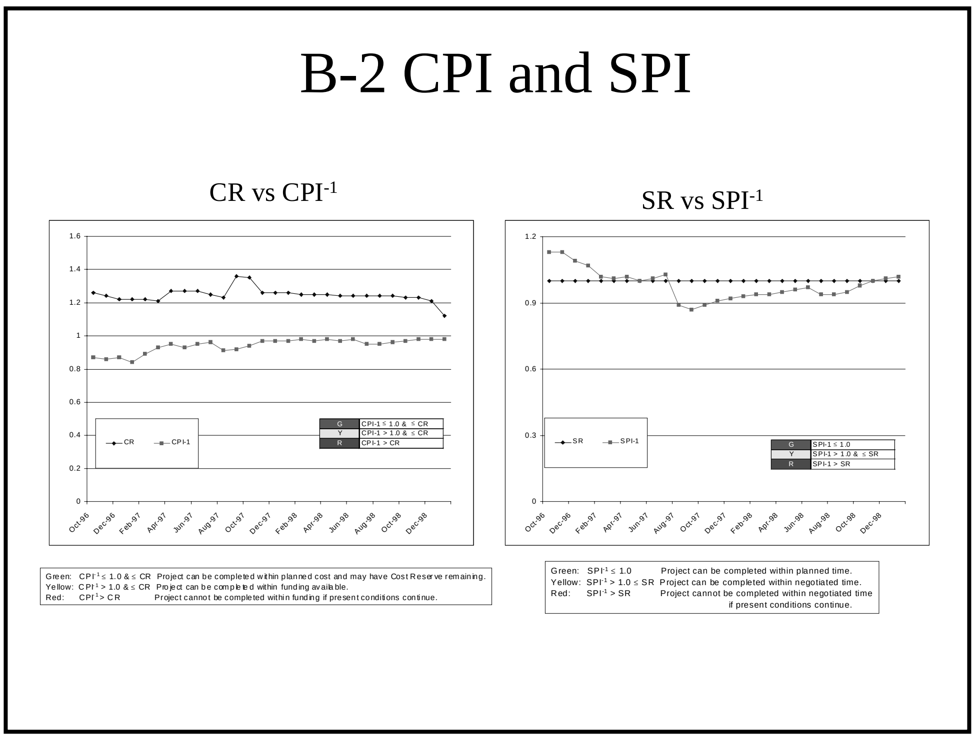


Figure 13: Performance Indices

Management Actions

CR vs CPI^{-1}	SR vs SPI^{-1}	Action
Green	Green	Reward Employees
Green	Yellow	Increase OT
Green	Red	Increase OT/People
Yellow	Green	Decrease OT
Yellow	Yellow	Review & Adjust Assignments
Yellow	Red	Adjust Assignments Consider Negotiation (Schedule)
Red	Green	Decrease OT/People
Red	Yellow	Adjust Assignments Consider Negotiation (Funding)
Red	Red	Negotiation (Funding/Schedule/Rqmts) Fire Manager

For Schedule Recovery

$$PE = \frac{BAC - BCWP_{CUM}}{CAR * PTR}$$

where PE is the Personnel Equivalent required for the remaining of the project,
CAR (Cost Accrual Rate) = Total avg cost/per person/year,
and PTR is Project Time Remaining (years)

$$OT = \frac{1 + OT_P}{TCSI} - 1$$

where OT_P is the planned overtime rate.

For Cost Recovery

$$PE = \frac{BAC - ACWP_{CUM}}{CAR * PTR}$$

$$OT = \frac{1 + OT_P}{TCPI} - 1$$

Figure 14: Management Actions

6.7 Rework

We have chosen to focus on rework as opposed to defects for our project reporting. We don't specifically track defects at the organizational level. Our measure is "rework." We believe it provides a more comprehensive view of both process and product. Our definition of "rework" is all corrective work performed after a review or test.

6.7.1 TPS Development Rework

For the following discussion concerning TPS development rework, reference Figure 15. The background of Figure 15 (i.e., the dashed lines and print) provides information about our process improvement efforts. The bubbles and horizontal lines are related to specific projects. The lines correspond to the time span of the project. (The beginnings of the early projects pre-date the chart scale.) The number in parentheses inside the bubbles is the percentage of rework experienced by that specific project.

The early projects pre-date our measurement program and were derived from graybeard recollections and archived records. We believe the numbers to be fairly accurate. In fact, the C-141 project, which began in 1984, ended up being virtually a complete "start-over." So, it was estimated only 25 percent of the original work was salvaged.

Clearly, it's seen that the improvement in rework for TPS Development has been remarkable. Our present process controls rework, nominally, to only three percent. We don't believe it's worthwhile to seek additional improvement in this attribute. To use a cliché, it's about "as good as it gets." A further reduction in rework is not considered to be cost effective for our customers.

How did we get the improvement? We can't show a one-for-one correlation, but our intuition says it's strongly related to process definition and subsequent refinements such as adding "peer reviews," and, as with many items, simply measuring the attribute.

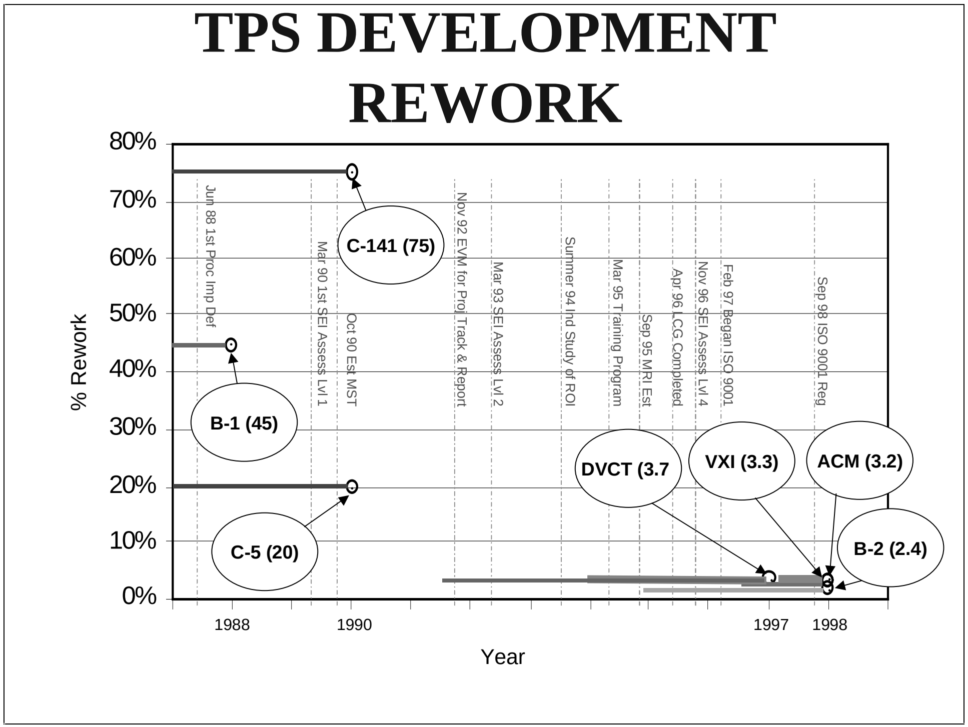


Figure 15: Development Rework, by Project

6.7.2 Review vs. Rework Costs

The rework data we are seeing has caused some self-examination. At three percent rework, we became concerned that, possibly, we were investing too much in quality. It wasn't that we wanted to go down the path of fielding inferior quality software; we just wondered if we were "gold-plating."

Prior to the IEEE review, we had begun looking into what we were getting in return for the quality investment. The question we were attempting to answer was, "Could we speed up the process and save some money, and at the same time keep the risk of a customer acceptance failure at a tolerable level?" Anyway, we described the tradeoff to the review team and the analysis that had been done. The results of this analysis are shown in Figure 16. At the time, we had not been able to ascertain the increased probability of having a product acceptance failure from dropping a review, so we hadn't made a decision to take any action.

In searching for an answer, the question was posed to all project managers in both development and maintenance areas. One response was unique. It came from our engine test software group. For this area, we must have a qualified test system operator at the controls of the test equipment when a “live” run of a jet engine is made. It’s a safety issue that we don’t want for ourselves. So, for us to perform the integration testing, we have to schedule a qualified test system operator—it’s expensive at $150 per hour. The project manager in this area explained, “We treat the integration testing phase like it was final product acceptance. We want everything to be as right as it can be.” We knew that this group spends lots of time in test simulation, but we never really explored why. The project manager explained, “It’s a matter of reputation. Because of the safety requirement, integration test is our first contact with the customer for the changed software. We don’t want him coming to the final test believing we are selling him junk.” So, besides the tradeoff of cost and schedule vs. percent rework, there is customer perception that must be considered.

The idea here is “Don’t discount ‘good’ measurements, they can lead to improvement, too.”

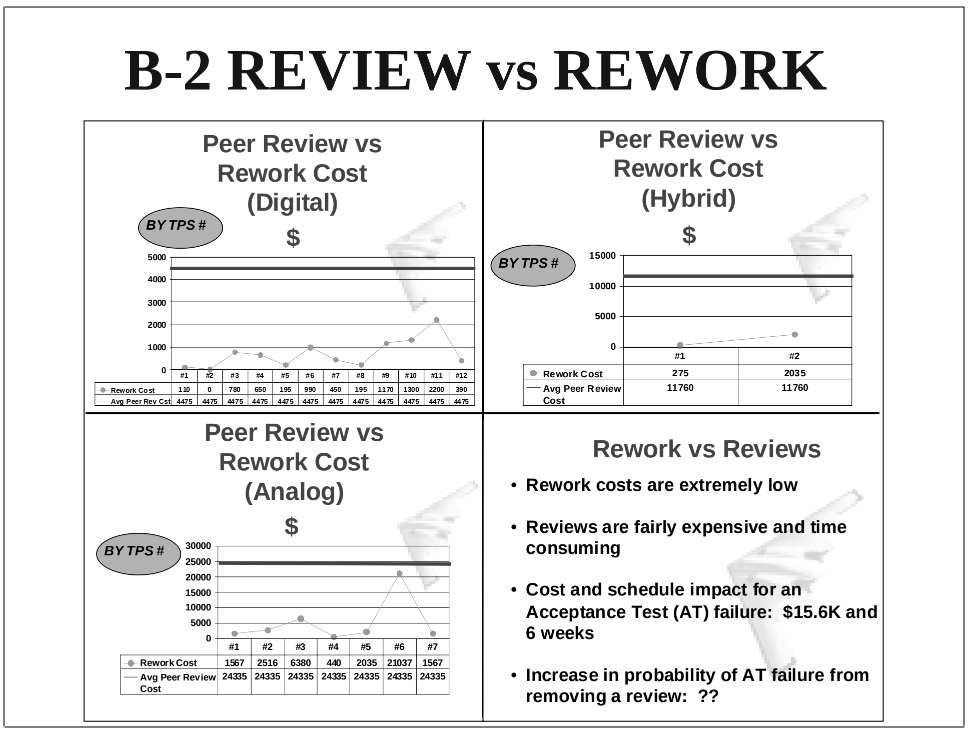


Figure 16: Reviews vs. Rework

6.7.3 Development and Maintenance Rework

The previous discussions concerning rework are for TPS development only. The portrayal in Figure 17 is sliced a little differently. These are plots for both development and maintenance from monthly composite data across all current projects within a specific branch of our organization.

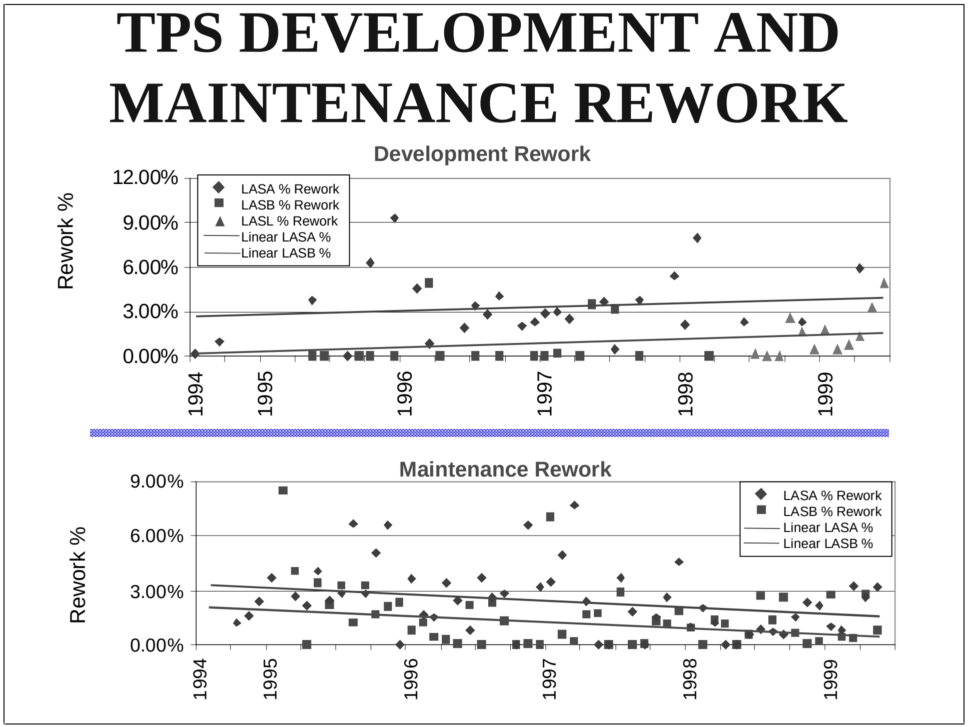


Figure 17: TPS Development and Maintenance Rework

The rework measure was piloted in 1994 and wasn't earnestly reported until early 1995. It was our most difficult measure to initiate as there were many suspicions that took several reporting periods (months) to overcome. Mostly, employees wondered, since this measure is 'negative' by nature, if there would be personal repercussions from a poor indicator. It took a while for them to realize that it was a measure of process efficiency and that it could lead to further process and individual improvement. Once we overcame the concerns, which took about a year, doubts ceased about the accuracy of the data. To facilitate the collection of re-work data, we had to implement changes in our Work Breakdown Structures (WBSs) and Earned Value systems.

From 1997 on, there is little difference in the rework values across the organization, for either development or maintenance. Development runs slightly higher, but both hover around 3%, which is significantly lower than the 40% value nominally reported for the software industry.

7 Benefits to Us and Our Customers

7.1 Organizational Growth

Since 1984 the organization has seen approximately 20 percent yearly growth. This is significant growth for any organization, but it's even more impressive for a government organization that faces constraints on workload and hiring. Figure 18 shows the growth and the associated workload and improvement milestones.

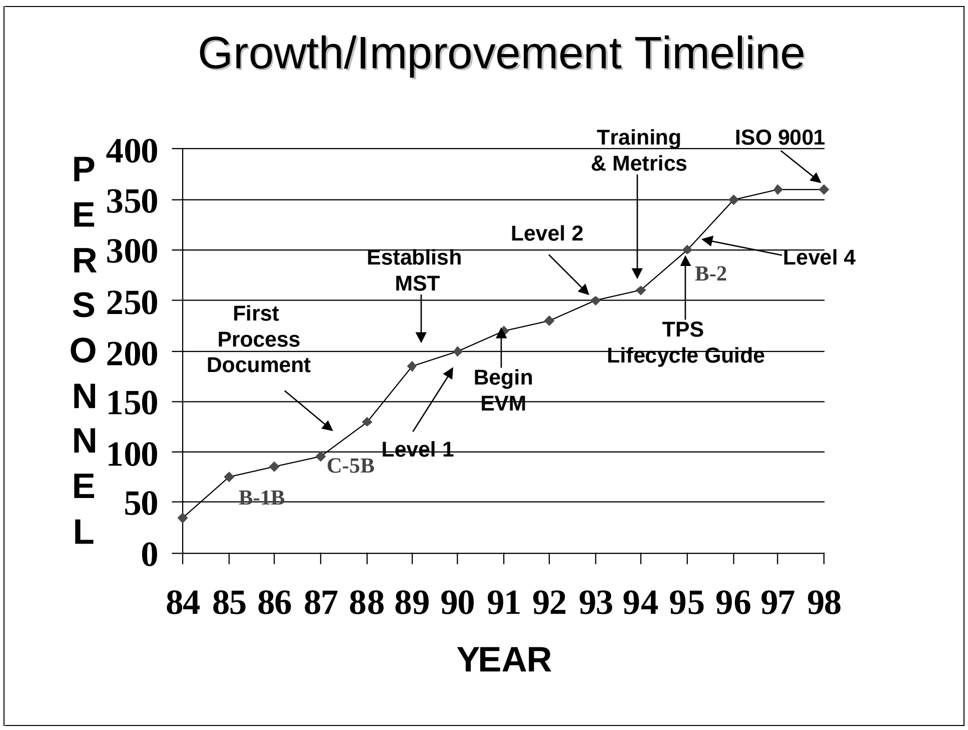


Figure 18: Growth/Improvement Time

The bottom line is that our customers want us to do their work. Our customers do have a choice as to who performs their software development and maintenance. They can use a private contractor or us. We have to show that we provide the highest quality product for their software dollar—and that we do it as efficiently as possible. In addition to our government customers, we have been approached by private contractors who want to subcontract or partner with us. While those are issues that we have yet to overcome due to laws and funding rules, it is still a sign of the respect that we have earned in the software community. Our reputation is a good one, and one that we are proud of.

7.2 The B-2 Story

One of our primary goals is to do what is best for our customers and do our part to hold down the cost of government. The Test Software and Industrial Automation Branches are able to do only a portion of the software that our customers need, but we can be a very positive influence by providing our expertise to weapon system managers during their acquisitions. This includes providing “should-cost estimates” for use as points of reference.

Personnel from the Test Software and Industrial Automation Branches were involved early on, since 1989, to develop the Automatic Test Equipment and TPS acquisition strategies. In 1992 a review performed by the B-2 System Program Office (SPO) showed that it would be more cost effective to award the TPS to us. However the SPO was resistant to do so; if the work was not awarded to the prime contractor, then the contractor would no longer be held accountable for the performance of the weapon system. Certainly the SPO faced a very difficult decision. The final decision was to award the work to the contractor and maintain his accountability for B-2 performance.

At that time, the Oklahoma City Air Logistics Center Commander took exception to that decision, stressing our capability and the SEI CMM Level 2 rating that we held at that time. The final result was a split of the TPS development between us and the contractor, with a Memorandum of Understanding (MOU) to define the agreement.

Unfortunately, our commander retired, the MOU was dissolved, and we were asked to help the SPO contract with the prime for ATE and TPS acquisition. While preparing the contract documents, it was determined that the TPS Development costs were grossly underbid by more than a third. Even with reduced requirements, the SPO did not have enough money to complete the necessary work.

Our staff, in conjunction with the B-2 SPO, developed a “vendor breakout strategy” that would make the program executable. For this strategy, the Test Software and Industrial Automation Branches would perform a portion of the work and assist the SPO in contracting the remainder of the TPSs to private industry.

The cost avoidance on this option was considerable. But, more important than the cost savings, it allowed the SPO to obtain the needed avionics test capability within available funding.

What is the significance in terms of process improvement? If we had not obtained SEI CMM Level 2 and developed a credible reputation, we never would have been seen as a viable option for the SPO. Process improvement gave us a business advantage.

So, how has it all turned out? Three years into the project, we are 3.5 percent under cost, 3.5 percent over schedule, and 7 percent ahead on projects delivered. We delivered the first B-2 TPS and we have been awarded additional work, essentially doubling the value of the project.

7.3 B-1B Data: Us vs. the Competition

One of our earliest successes was the development of 67 TPSs for the B-1B (1985-1988). While we were very much a Level 1 organization, we did perform, to some degree, the Level 2 KPAs. Various contractors developed the other 558 TPSs. Additionally, we have performed all maintenance on the 625 TPSs since they went into service. So, we have quite a bit of data.

In late 1998, our senior manager asked for data concerning our B-1B TPS Maintenance Support and we sent him the data shown in Table 3.

	Number of TPSs	**Number Exhibiting Defects (fraction)**	**Defects Identified (per Exhibit)**	**Correction Effort (mhrs per TPS)**
LAS	67	24 (0.353)	50 (2.08)	6,140 (91.6)
Contractors	558	370 (0.663)	995 (2.69)	114,079 (204.4)

Table 3: B-1B TPS Data

We also supplied the following analysis:

- Our B-1B TPSs are nearly one-half as likely to have defects.
- Our TPSs have a 30 percent smaller defect density.
- Maintenance investment per TPS is less than one-half for our products.
- Maintenance savings for our TPSs is $450 thousand.
- Maintenance savings from our involvement and efficiencies is $11 million.

While we thought this was great information, we were still a little apprehensive when our director sent the data and analysis to our customers to get their response. The completely positive feedback could not have been anticipated. Some of the responses follow:

- Harrision Pennel, Engineer, OC-ALC/LIIRN - "*LAS consistently provides much more inexpensive bids for TPS development and maintenance while maintaining high quality work. LAS has proven to be a very competent, flexible organization that has given LII very good support at a reasonable cost*."
- Sam George, Branch Chief, OC-ALC/LIIR - "*We are getting good service from LAS*."
- Col A. B. Decker, Deputy Director, OC-ALC/LI - "*We appreciate your LAS support and find your service is priced right and is extremely flexible for meeting our needs*."
- TSgt Hal Ingram, Dobbins ANG - "*Let your guys know they've been doing an outstanding job on the Engine Instrument SCDU software. The last rev has been great at detecting LRU failures previously missed. Keep up the great work.*"

7.4 Improvement of Project Estimates

One of the really good things we accomplished was applying Earned Value Management (EVM) to software maintenance. EVM has two myths:

1. It has such high overhead it can only be applied to large projects.
2. It can't be applied to software projects, and certainly not software maintenance.

Although these arguments are accepted nearly everywhere ,we had one lone voice within the organization that persisted. It was his opinion that small software projects could be managed using EVM. We decided to give it a try; EVM was piloted in one area and subsequently migrated throughout the organization.

The top portion of Figure 19 basically says we know much more about our maintenance process today—our output is very predictable. The use of EVM has facilitated this.

As a crosscheck, the question could be asked, "Okay, you can plan and execute to plan, but are you building plans with a lot of fat to insure that outcome?" The answer is "No" and is in the bottom portion of Figure 19. It shows the decreases in effort and cycle time that were seen over the same period.

EVM is a good technique and is applicable to both software development and maintenance, regardless of project size. We recommend it.

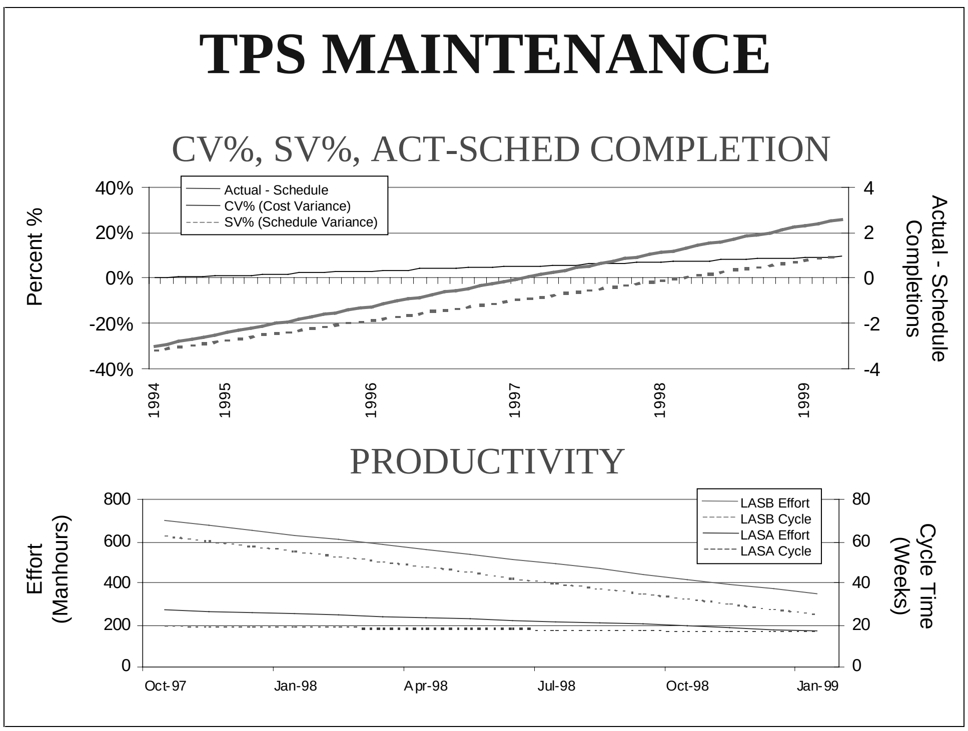


Figure 19: Estimates vs. Actuals

7.5 Integration Time and Effort Reduction

Because integration is such a major portion of development, we thought we should see it diminish percentage-wise if we were truly getting better. It just stands to reason that if requirements are clearer and more attention is paid up-front in the process, then integration time should decrease.

As with the illustration (Figure 15) shown previously for rework, data for the early projects came from graybeard estimates and archived project records. The initial data we received indicated that the integration times for some of the recent projects weren't much better than the old ones. When asked why, the response was, "We do so much more now."

During the integration of a TPS for an electronic circuit card assembly, circuit component failures are physically inserted and the software is executed to see if the fault is correctly identified. We want to know if the technician who will use the TPS will get the correct repair instructions. In the late 1980s and early 1990s, we inserted only about 20 percent of the possible failures. This is tedious work. Today we can do more with circuit simulators to give us confidence prior to actual fault insertion, but still the "rubber must meet the road" at some point. We must insert faults to ensure that we are providing a good product.

In examining Figure 20, it is obvious that there's been improvement when you see the greatly increased quality of the TPS emerging from integration. The TPS being produced today has two to four times the number of possible faults tested for the same percentage of total project effort. Additionally, the total effort for developing a TPS has been reduced by about 37 percent; so, in fact, product quality has increased while reducing effort. Certainly the B-2 project is worth noting. It exhibits both reduced effort along with greater quality.

Beyond the impacts to development, the fact that we are more certain of correctly identifying the faulty circuit component means there will be fewer TPS maintenance actions down the road. Our expectations were achieved and then some.

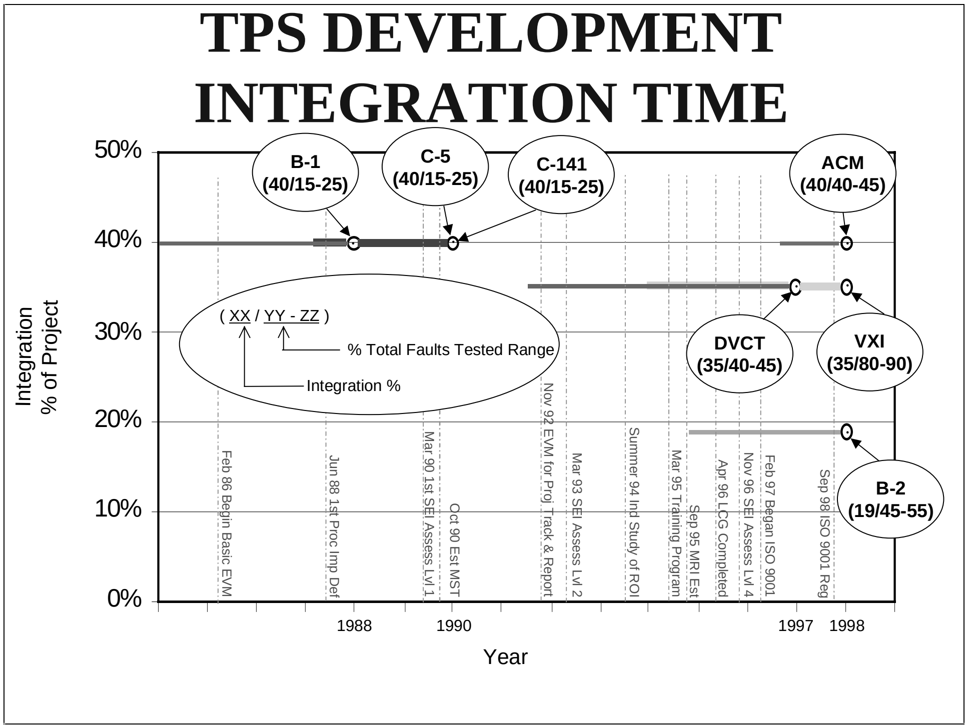


Figure 20: TPS Development Integration Time

7.6 Process Improvement Return on Investment

For an improvement effort to truly be considered successful it must show a quantifiable return on investment. We have examined ROI three different ways: the first, and perhaps the most important, focuses on productivity and defects. The next ROI calculation shows cost avoidance, and the final ROI calculation is required by the Air Force to justify our process improvement funding. Each is discussed below.

7.6.1 Productivity/Defect Improvements

As shown in Table 4, we have seen a steady improvement in our productivity and defect data for the past several years with the exception of this past fiscal year where we saw a productivity decrease (which we will explain below).

	Delivered Defects per KSLOC	**TPS Development Effort (man-hours)**	**TPS Development Cycle-Time (months**
1993	3.35	1600	13
1996	0.35	1200	12
1997	0.03	1150	12
1998	TBD	923	12
1999	TBD	1081	18

Table 4: Defect and Productivity Data

Only in the past four years have we been able to obtain reliable productivity data. Additionally, it takes time, generally almost a year, from when we finish development to when the TPS is actually used in the production environment, thus allowing us to collect data on delivered defects. While this seems long, it is actually a vast improvement from the past when several years elapsed between development and production. The good news is, our TPS delivered defect rate continues to drop.

The 1999 productivity decrease is attributed to the B-2 TPS project. The customer has levied a significantly elongated quality process, which has raised both the effort and cycle time per TPS. Our non-B-2 projects are either maintaining or improving their productivity.

7.6.2 Cost Avoidance Calculation

In addition to productivity and defects, we wanted to evaluate cost avoidance. We asked ourselves this question, “If we had not gained efficiency, what would the cost have been to accomplish the same effort?” The results are significant, a reduction in effort of 765,000 manhours, with a corresponding reduction in cost of $50.5 million. When compared to the software process improvement (SPI) investment of $6 million, the ROI is computed to be 8.4 to 1—fairly impressive.

Looking over the graph in Figure 21, there are a few associations that come to mind. The cost avoidance didn’t really take off until we began our association with the SEI. For our case, with increasing maturity, the graph indicates that increased cost avoidance can be expected. Admittedly, these conclusions are rough and probably need more refinement. Nevertheless, besides making the statement, “We have something to show for our investment,” they give a fairly strong endorsement of the value of SPI and the SEI.

7.6.3 Air Force Calculation of Process Improvement Return on Investment

The third method we use to determine ROI is provided to us by the Software Technology Support Center (STSC) at Hill AFB, Utah. This method is used to justify the Air Force funding for process improvement. The STSC spreadsheet baselines our costs, starting in 1992, and then captures data each subsequent year concerning the number of software items developed/changed and productivity. The spreadsheet also captures the funds invested in process improvement. As of 1999, our ROI since 1992 is 7 to 1.

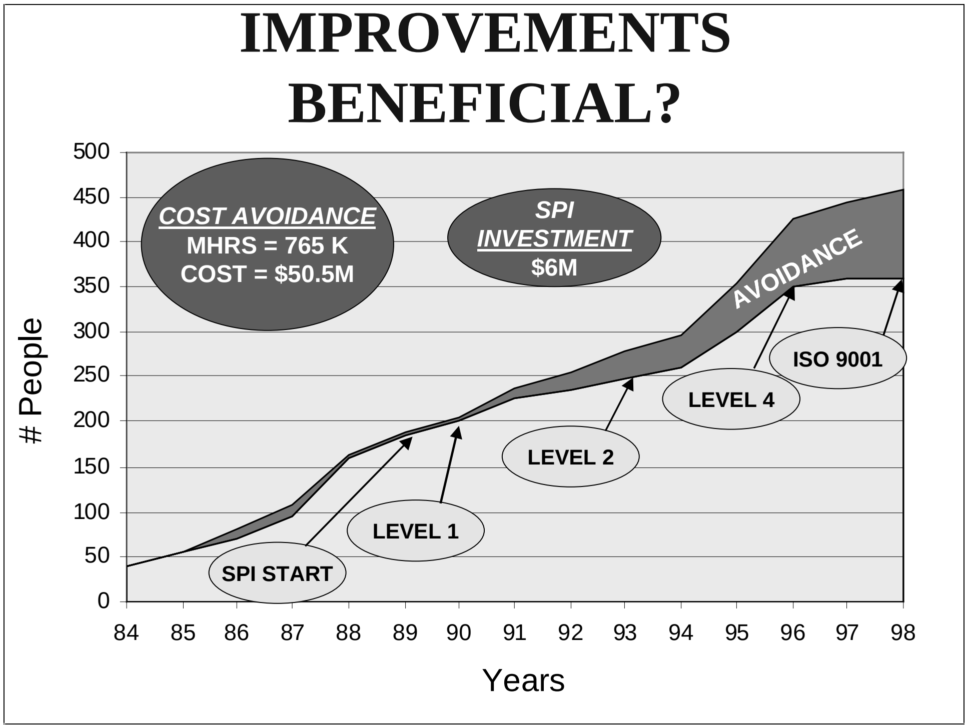


Figure 21: Cost Avoidance

8 Sharing

Since beginning our improvement efforts, we have actively shared information with others in both the public and private sectors. We hope that our publications, presentations, and consultations have been useful to others in their process improvement journey and perhaps have helped them eliminate some of the blind paths and pitfalls. We know that each organization is different, but it often helps to know how others approached an improvement.

8.1 Papers, Studies, Presentations

Articles and Reports

- June 2000, *CrossTalk Journal of Defense Software Engineering,* "Statistical Process Control Meets Earned Value," W. Lipke and J. Vaughn.
- Jan/Feb 2000, Aerospace Acquisition 2000, *"Earned Value Helps Air Force Software Division Excel,"* W. Lipke.
- March 1999, *CrossTalk Journal of Defense Software Engineering*, "Software Project Management, Applying Management Reserve," W. Lipke.
- October 1997, *CrossTalk Journal of Defense Software Engineering*, cited by U.S. Air Force: Lt. Gen. William J. Donahue, Deputy Chief of Staff, Communications, and Information, as the **Department of Defense's premier software development provider**.
- May 1997, *CrossTalk Journal of Defense Software Engineering,* "Process Lessons Learned While Reaching Level 4," K. Butler.
- July 1995, *CrossTalk Journal of Defense Software Engineering,* " The Economic Benefits of Software Process Improvement," K Butler.
- September 1994, Software Productivity Research, *"An Analysis of Software Process Improvement,"* J. Nowell.
- August 1994, SEI Technical Report, CMU/SEI-94-013, "Benefits of CMM-Based Software Process Improvement Initial Results"
  > **Only government organization included is OC-ALC/LAS; other groups highlighted in the report included Hughes and Texas Instruments.**
- December 1992, *CrossTalk Journal of Defense Software Engineering*, "Software Process Improvement: A Success Story," W. Lipke, K. Butler.

8.2 Sharing of Improvements

- Process document samples provided to National Security Agency, April 2000.
- Chosen for as Benchmarking Candidate for Air Force Aerospace Data Facility, March 2000; initial data sent in April 2000.

- Metrics definitions provided to Ron Radice for book that he is writing, October 1998.
- Metrics definitions provided to Software Productivity Consortium for July 1997, "CMM Level 3 and 4 Metrics," SPC-97054-MC.
- To date, OC-ALC/LAS (Oklahoma City Air Logistics Center Directorate of Aircraft Software Division) has freely shared information on their improvement process and efforts, including specific improvements and documents, with over 40 Air Force, DoD, government, and private organizations. Specifically, OC-ALC/LAS has worked closely with the 552nd Airborne Warning and Control System (AWACS) Computer Group to help them achieve SEI CMM Level 3 in 1997 and continue to provide assistance. Also shared metrics information and philosophies with Ogden Air Logistics Center (OO-ALC) in the year prior to their July 1998 SEI CMM Level 5 assessment. Continually provide information and materials to the Federal Aviation Administration (FAA) Mike Monroney Center in Oklahoma City.
- In addition to many presentations within the Air Force, LAS has made requested process improvement presentations to groups at Hewlett-Packard, the FAA, the National Weather Service, and the University of Oklahoma.

8.3 Conference Presentations and Papers on Software Process Improvement

- 2000 Software Technology Support Center (STSC) Software Technology Conference. Salt Lake City, UT
- 2000 College of Performance Management Conference, Clearwater, FL
- 1999 International Integrated Management Conference, Washington, DC
- 1999 SEI Symposium, Pittsburgh, PA
- 1998 STSC Software Technology Conference. Salt Lake City, UT
- 1997 3rd Annual Conference on Software Metrics, Washington, DC
- 1997 International Conference on Software Engineering, Boston, MA
- 1995 Test Facilities Working Group, Las Vegas, NV
- 1994 STSC Software Technology Conference, Salt Lake City, UT
- 1993 SEI Symposium, Pittsburgh, PA
- 1992 National Quality Symposium, Dallas, TX

9 Critical Success Factors

When asked by Watts Humphrey what we attributed our success to, one of our section chiefs, Rick McIver, spoke up and said, “Sir, it’s everything,” and that truly is the answer. The mythical silver bullet doesn’t exist; improvement must be broad and must touch every aspect of the organization to be successful and lasting. But, we did feel that a few items were key, and they are highlighted below.

9.1 We Wanted to Improve

This sounds simple: everyone wants to get better, right? That may be true, but while everyone wants to get better, few want to make the investment and overcome the obstacles. We wanted to improve. At the beginning, we may have not all been on the same page, but, loose as it was at times, everyone wanted to get better. Today, having been successful, we’re much more focused and, fortunately, the desire remains. There is a continual focus on how to do things better, smarter, and more efficiently.

9.2 Leadership

This can not be emphasized enough. The Test Software and Industrial Automation Branches have had the good fortune of stable leadership throughout the organization. While there has been a fair amount of turnover at the working levels (not unusual in today’s economy), the organization has been fortunate not to have to continually “sell” process improvement to “the boss.” With leaders who are dedicated to their jobs and the people who work for them, process improvement works!

9.3 Funding

Because we are a fee-for-service organization, we are required to account for labor, time, and other costs. This is important to say because many feel that government workers have unlimited funding and time. How we wish that were true! Our people don’t have “spare” time.

Since 1992, the Air Force has provided process improvement funding. This funding is key because it allows process improvement to be tracked and managed at the same level as any other workload. It also helps facilitate the use of “key” people on the improvement efforts. We know that we would have never seen the success our organization has experienced without the funding, and, even though it is less today, it is still vital to our ongoing efforts.

10 The Future, What's in Store

The IEEE Review Team asked hard questions. The primary focus was our commitment to continuing the improvement effort, to ensure that we hadn't relaxed our focus since achieving SEI CMM Level 4.

In contrast to the IEEE Review Team's focus, Judah Mogilensky, our lead assessor for the 1996 Level 4 assessment, made this statement following the assessment, "Now you are ready to begin improvement." Very succinctly, Mr. Mogilensky was telling us that future product and process improvements would be made using data as rationale, depending less on intuition. We believe this report bears out Mr. Mogilensky's words.

The following paragraphs are descriptions of our current areas of process improvement interest.

10.1 Statistical Process Control for Project Management

The software industry continues to struggle with Statistical Process Control (SPC). In 1996, when we were rated SEI CMM Level 4, the application of SPC to software development was rarely discussed. At that time, and continuing today, most of our indicators are in the form of trend charts. Certainly, trend charts are a viable form of SPC. However, over the last three or so years, there is a growing consensus that software process control cannot be achieved without the use of control charts. In fact, the general thinking today among the Software-CMM experts is that achievement of SEI CMM Level 4 implies that the organization has a stable process. Well, how do you know that your process is stable if you are not using control charts? The answer is, you don't. So there is increasing pressure for existing Level 4 and 5 organizations to show that they are using the method.

Today, there are a few software organizations attempting to apply SPC. Most, because of the quality connotation, are employing the method in conjunction with coding reviews. At least from the anecdotal stories circulated, the application of SPC to software development, so far, is not a success. Yet, the pressure to apply SPC continues to grow.

Our endeavor to apply SPC is merged with the methods we use to plan and track our projects (i.e., the practice of Earned Value Management). Within EVM are indicators describing the efficiency of achieving the project cost and schedule commitments. We chose to apply SPC to these indicators. The SPC charts from one of our projects are illustrated in Figures 22 and 23.

The application of SPC to software development holds a considerable amount of promise. In the application we've developed, it is an additional software project management tool for quantification of performance, recovery, planning, risk, and process improvement. We are presently prototyping the tools and ideas. If our application of SPC proves to be beneficial, we will implement it on all of our development efforts. For more information, refer to the June 2000 *Crosstalk* article, "Statistical Process Control Meets Earned Value" [Lipke 00].

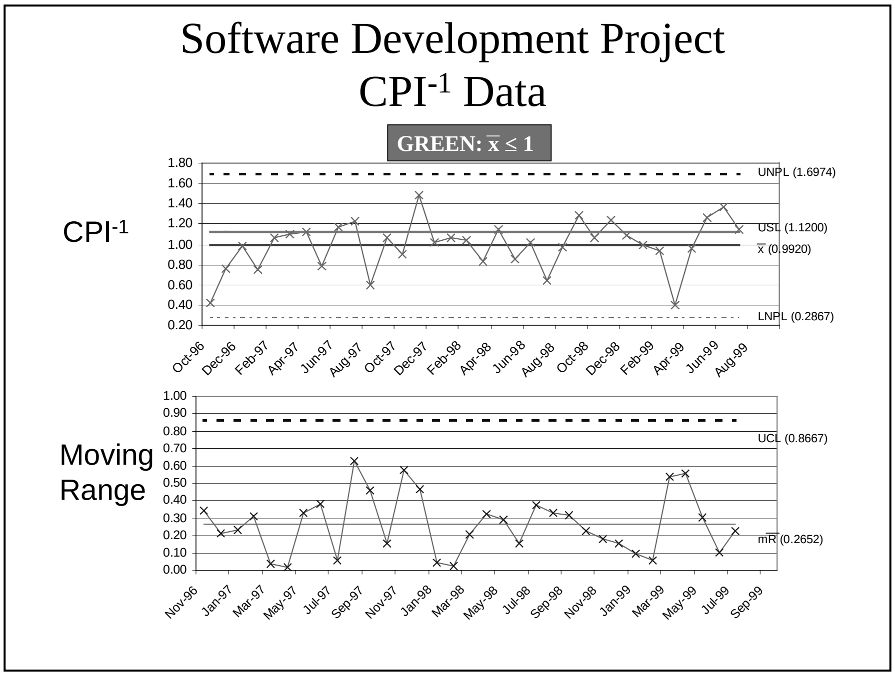


Figure 22: Software Development Project Cost Performance Index (CPI^{-1}) Data

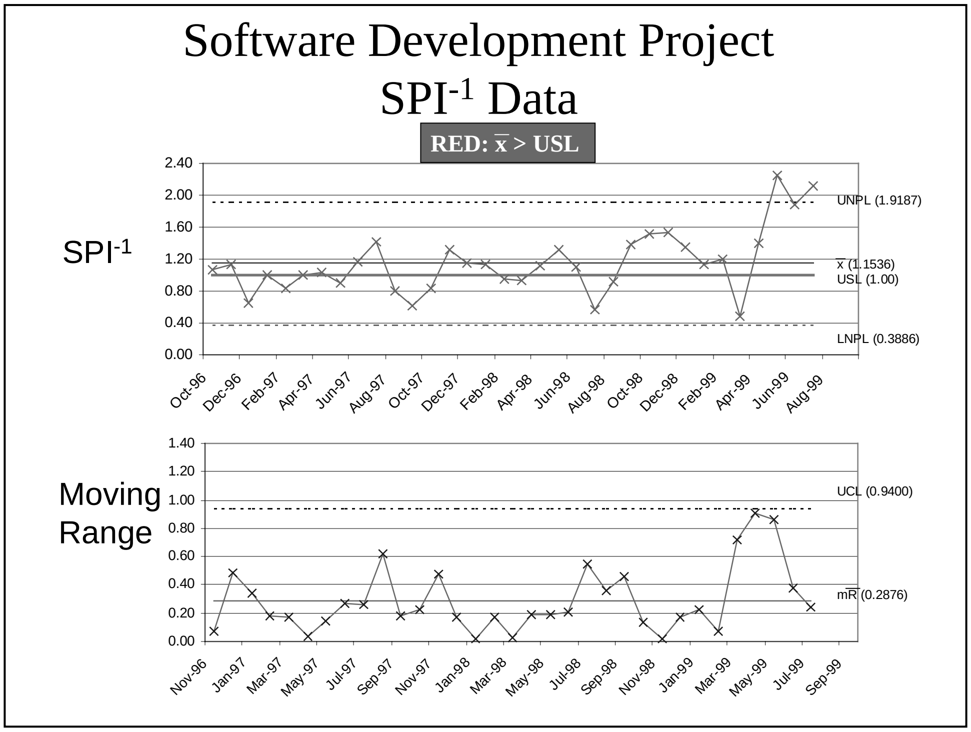


Figure 23: Software Development Project SPI^{-1} Data

10.2 Capability Maturity Model - Integrated - Systems/Software Engineering (CMMI-SE/SW)

As with all software organizations, CMMI has to be a focus for us [CMMI 00]. We feel that our ISO 9001/TickIT efforts not only helped us place focus on Defect Prevention, the one key process area that we did not satisfy in 1996, but these efforts also helped lay the foundation for CMMI. We will be examining our processes and documentation to see what changes will need to be made and what areas we need to focus on.

10.3 Information Technology (IT) Process Improvement

This is an area that is not unique to our organization, Tinker AFB, or the software industry in general. Just as software exploded in the 1980s, IT has exploded in recent years. The personal computer and utilities such as email have gone from “nice to have” to necessities. Networks have evolved into very complicated systems that are changing how we do business. We are working not only to improve this area internally but also to help extend our process improvement experience and lessons learned to the Tinker AFB information technology area. As technology continues to “explode,” we must continually strive to do what is best for the organization and our customers, realizing that at times this will destabilize our process, but that is what process improvement is really about.

References

[CMMI 00] CMMI Project. *Capability Maturity Model Integration (CMMI)* (online). Available WWW <URL: http://www.sei.cmu.edu/cmm/cmms/cmms.integration.html> (May 2000).

[Lipke 99] Lipke, W. "Applying Management Reserve to Software Project Management." *CrossTalk Journal of Defense Software Engineering* (March 1999).

[Lipke 00] Lipke, W and Vaughn, J. "Statistical Process Control Meets Earned Value." *CrossTalk Journal of Defense Software Engineering* (June 2000).

[Paulk 93a] Paulk, Mark; Curtis, Bill; Chrissis, Mary Beth; and Weber, Charles. *Capability Maturity Model for Software, Version 1.1* (CMU/SEI-93-TR-024, ADA 263403). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, Feb. 1993. Available WWW <URL: http://www.sei.cmu.edu/publications/documents/93.reports/93.tr.024.html>.

[Paulk 93b] Paulk, Mark; Weber, Charles; Garcia, Suzanne; Chrissis, Mary Beth; and Bush, Marilyn. *Key Practices of the Capability Maturity Model, Version 1.1* (CMU/SEI-93-TR-025, ADA 263432). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, Feb. 1993. Available WWW <URL: http://www.sei.cmu.edu/publications/documents/93.reports/93.tr.025.html>.

Acronyms

ACWP	Actual Cost of Work Performed
AFCA	Air Force Communications Agency
ATE	Automatic Test Equipment
ATLAS	Abbreviated Test Language for All Systems
AWACS	Airborne Warning and Control System
B-1B	B-1B Bomber Aircraft
B-2	B-2 Bomber Aircraft
BAC	Budget at Completion
BCWP	Budgeted Cost of Work Performed
C-5B	C-5B Cargo Aircraft
CAR	Cost Accrual Rate
CD	Compact Disk
CMM	Capability Maturity Model for Software
CMMI-SE/SW	Capability Maturity Model - Integrated - Systems/Software Engineering
CPI	Cost Performance Index
CR	Cost Reserve
DATSA	Depot Automatic Test Station for Avionics
ETVX	Entry, Task, Verification, Exit
EVM	Earned Value Management
FAA	Federal Aviation Administration
IEEE	Institute of Electrical and Electronic Engineers
ISO	International Standards Organization
KPA	Key Process Area
LCG	Life Cycle Guide
LNPL	Lower Natural Process Limit
LRU	Line Replaceable Unit

MATE	Modular Automatic Test Equipment
MOU	Memorandum of Understanding
mR	Moving Range
MR	Management Reserve
MRI	Management Reporting Indicators
MST	Management Steering Team
OPD	Organization Process Definition
OT	Overtime
PC	Personal Computer
PCM	Process Change Management
PE	Personnel Equivalent
PTR	Project Time Remaining
QPM	Quantitative Process Management
RF	Radio Frequency
ROI	Return on Investment
ROM	Rough Order of Magnitude
SEI	Software Engineering Institute
SEPG	Software Engineering Process Group
SOW	Statement of Work
SPC	Statistical Process Control
SPI	Schedule Performance Index
SPI	Software Process Improvement
SPO	System Program Office
SQA	Software Quality Assurance
SR	Schedule Reserve
SRU	Shop Replaceable Unit
STSC	Software Technology Support Center
TCM	Technology Change Management
TCPI	To Complete Performance Index
TCSI	To Complete Schedule Index
TCTO	Time Compliance Technical Order

TPS	Test Program Set
TO	Technical Order
TS/IA	Test Software and Industrial Automation
UCL	Upper Control Limit
UNPL	Upper Natural Process Limit
USL	Upper Specification Limit
WBS	Work Breakdown Structure

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (LEAVE BLANK)	2. REPORT DATE **September 2000**	3. REPORT TYPE AND DATES COVERED Final
4. TITLE AND SUBTITLE **Software Process Achievement at Tinker Air Force Base, Oklahoma**		5. FUNDING NUMBERS C — F19628-95-C-0003
6. AUTHOR(S) **Kelley Butler, Walter Lipke**		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA 15213		8. PERFORMING ORGANIZATION REPORT NUMBER **CMU/SEI-2000-TR-014**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) HQ ESC/XPK, 5 Eglin Street, Hanscom AFB, MA 01731-2116		10. SPONSORING/MONITORING AGENCY REPORT NUMBER **ESC-TR-2000-014**
11. SUPPLEMENTARY NOTES		
12.A DISTRIBUTION/AVAILABILITY STATEMENT Unclassified/Unlimited, DTIC, NTIS		12.B DISTRIBUTION CODE

13. ABSTRACT (MAXIMUM 200 WORDS)

On May 20, 1999, the Test Software and Industrial Automation Branches of the Oklahoma City Air Logistics Center's Directorate of Aircraft Management's Software Division at Tinker Air Force Base were awarded the IEEE Award for Software Process Achievement. This report will outline the process improvement activities and successes that led to the award.

14. SUBJECT TERMS **Capability Maturity Model® (CMM®), software process improvement, IEEE Award for Software Process Achievement**	15. NUMBER OF PAGES **57**
16. PRICE CODE	

7. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	**UNCLASSIFIED**	**UNCLASSIFIED**	**UL**

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102