High Maturity Pays Off

It is Hard to Believe Unless You Do It

Girish Seshagiri, Advanced Information Services Inc.

Abstract. Is high maturity worth it? Yes, if executive management sponsors the long-term process improvement initiative with constancy of purpose and makes quality the number one goal. We provide a business owner's view of high maturity. We provide hard data on high maturity's impact on customer satisfaction, company profitability, and business strategic decision making as well as intangible results such as self-directed teams, low staff turnover and joy in work. We show that with higher maturity (CMMI® level 5), our processes came under statistical control, and we were able to create a strategic business model with competitive game changers such as firm fixed price contracts and performance guarantees including lifetime warranty on software defects. To the skeptics, we say, "It is hard to believe, unless you do it."

1. Background

Advanced Information Services Inc. (AIS) is in the software application development business. I founded AIS in Peoria, Illinois in 1986 to fill a niche in the market place by creating a workforce with skills in emerging modern programming languages such as C, C++ and Visual Basic and programming paradigms such as object oriented programming. My business objectives were to grow and be profitable while creating much needed high wage, high technology middle class jobs in the U.S. heartland.

I brought to the business a proven track record of successfully managing technical teams in a large corporation. I was a practitioner of the principles of leadership, management and quality articulated by Philip Crosby [1], Edwards Deming [2], and Peter Drucker [3].

2. The Early Years

In the early years of AIS, the company was not profitable because our projects were not predictable. Most of our projects were firm fixed-price contracts. Schedule and budget overruns were normal outcomes. People worked long hours, and heroic efforts were needed to complete projects. Although quality was not measurable, the significant amount of rework necessary at the end of a project was a clear indication of our quality problems and the resulting customer dissatisfaction.

3. The Improvement Initiative

I realized that we had to change the way we managed the software work. What we needed was constancy of purpose with quality as the number one goal. In 1992, I attended a conference on software process improvement based on Watts Humphrey's Managing the Software Process [4]. I returned from the conference convinced that I should sponsor a long-term process improvement initiative to achieve these goals:

- Improve profitability and customer satisfaction by delivering nearly defect free products on predictable cost and schedule.
- Provide a continuing management focus on the progress and visibility of each project from initial commitment to orderly progression through the development lifecycle phases and customer acceptance.
- Continuously improve the software development process through a changed organizational culture biased towards rapid implementation of many small incremental improvements as opposed to a few large changes.

I communicated this vision to everyone in the organization.

4. The Improvement Strategy

We named the initiative Continuous Process Improvement (CPI). We chose the CMM® as the process maturity framework to improve organizational process capability and IEEE standards as the guidelines for software engineering. We were the early adopters of the Personal Software Process (PSP) as the enabling technology to improve individual engineer performance and productivity [5].

We utilized a simple and effective mechanism of Process Improvement Proposal (PIP) to gradually evolve process maturity and ensure company-wide participation in the process improvement journey across maturity levels. We established a Software Engineering Process Group (SEPG), utilizing the skills and experience of many engineers, all on a part-time basis, to evaluate and implement the PIPs. Additionally, we used Watts Humphrey's Managing the Software Process [4] book as a guide and to establish a common vocabulary and communication means.

Our approach was to:

- Conduct self-assessments with a focus on action to achieve measurable results for rework reduction, early defect removal, improved customer satisfaction and predictable outcomes for schedule, cost and quality.
- Maintain organization awareness of improvement efforts through quarterly status reviews.
- Improve continuously and forever.

5. Tracking the Improvement: the AIS Balanced Scorecard

We used the Balanced Scorecard (BSC) method [6] to communicate strategy to the entire organization, link individual accountabilities to strategic objectives, and provide a method to systematically measure progress. The BSC approach identified what we could measure earliest in the software development process which in turn impacts the results later in the process. We then linked the process performance metrics to business objectives for shareholder, customer, and employee satisfaction. We constructed a cause and effect hypothesis: "When individuals follow the PSP, they will develop work products with targeted percent of defects removed before peer review which will lead to work products with zero post development defects as well as work products with less than or equal to targeted rework effort thereby achieving the strategic internal business process objective—individuals achieve the highest possible quality in their work products. This in turn helps project teams deliver nearly defect free product on time which leads to delighted customers which in turn helps us achieve profitability, revenue growth and joy in work."

We made the SEPG responsible for gathering, analyzing and reporting the BSC measurements in quarterly status review meetings that I chaired.

6. The Results 1992-1999

6.1 The Three Eras

We map the results in three distinct eras corresponding to the changes in the organization's process maturity due to the improvement initiative during 1992- 1999. The first era was the time period before the start of the CPI initiative in 1992 (pre-model era). In the second era, from 1992 to 1995, the focus of the initiative was to stabilize the organization's project planning and tracking processes and implement rigorous requirements engineering and change management processes (CMM-only era). The third era began in 1995 when the initiative focused on improving individual engineer performance and productivity (CMM+PSP era).

6.2 Schedule and Effort Predictability

Figures 1 and 2 graphically depict the impact of the CPI initiative on projects' schedule and effort commitments. Each data point represents a new development phase using the date when the project development phase started, as the horizontal coordinate. The process limits are calculated using a moving range. The process limits in Figures 1 and 2 show the dramatic change in capability to predict schedule and effort from 1988 through 1998.

The data indicate that the average schedule deviation improved from 112% in the pre-model era to 41% in the CMM era to 5% in the CMM+PSP era. While improving the schedule performance met the customer's needs, much of the work was contracted on a fixed-cost bid and, if effort was not predictable, the project phase was not profitable. When AIS began using the PSP, effort became more predictable because of the quality practices the PSP-trained software engineers applied. The data reflect the average effort deviation improved from 87% in the pre-model era to 37% in the CMM era to -4% in the CMM + PSP era.

6.3 Quality and Productivity

PSP-trained engineers working in a mature process improvement have few or no acceptance test and usage defects. The consequent near elimination of rework time and effort flows directly to the company bottom line. A useful metric is System Test duration measured in number of days per thousand lines of code (KLOC). Figure 3 shows the dramatic reduction in system test duration in the CMM+PSP era.

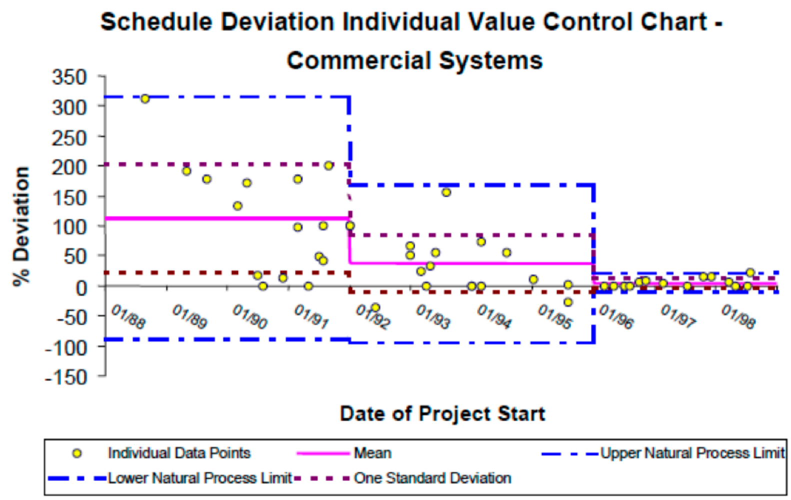


Figure 1: Impact of CPI initiative on projects' schedule deviation

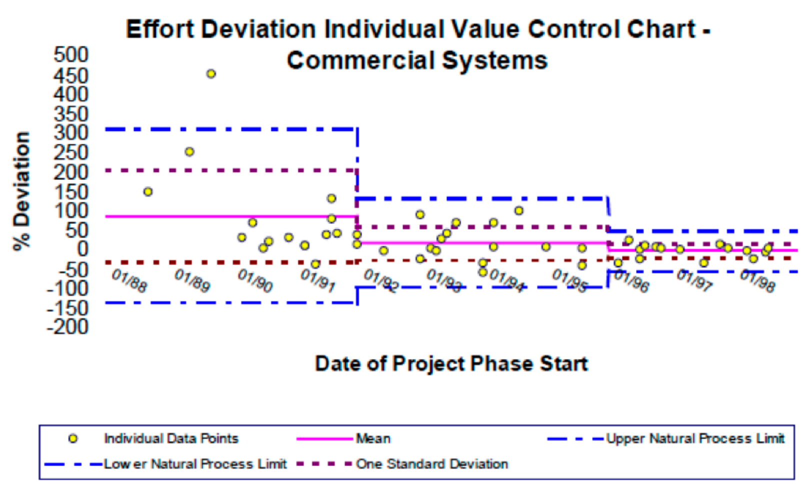


Figure 2: Impact of CPI initiative on projects' effort deviation

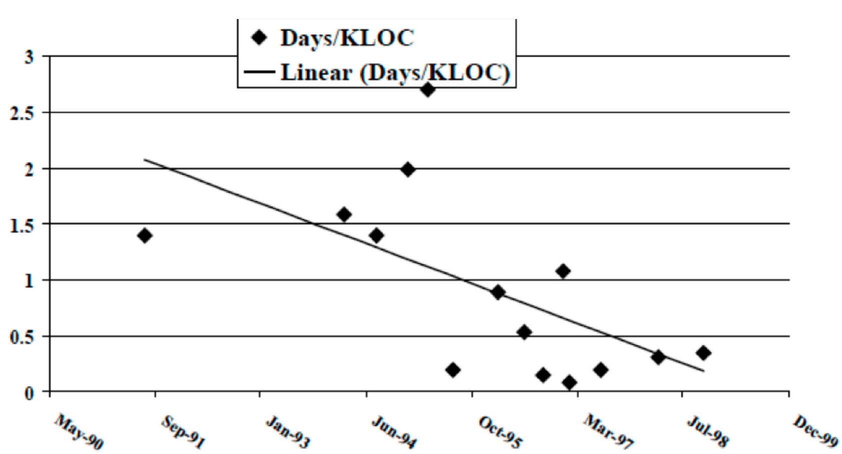


Figure 3: Projects' system test duration measured in number of days per KLOC

It is well known, though seldom practiced, that it is cheaper to remove defects earlier in the lifecycle phases such as requirements, design, and coding than later phases such as integration, system, and acceptance test. A useful metric is the percent of defects removed prior to test. AIS teams of PSP-trained engineers consistently deliver very high quality product into test. Figure 4 shows that in the CMM+PSP era, AIS projects through the consistent use of mature processes such as team inspections and personal reviews removed more than 75% of defects prior to test.

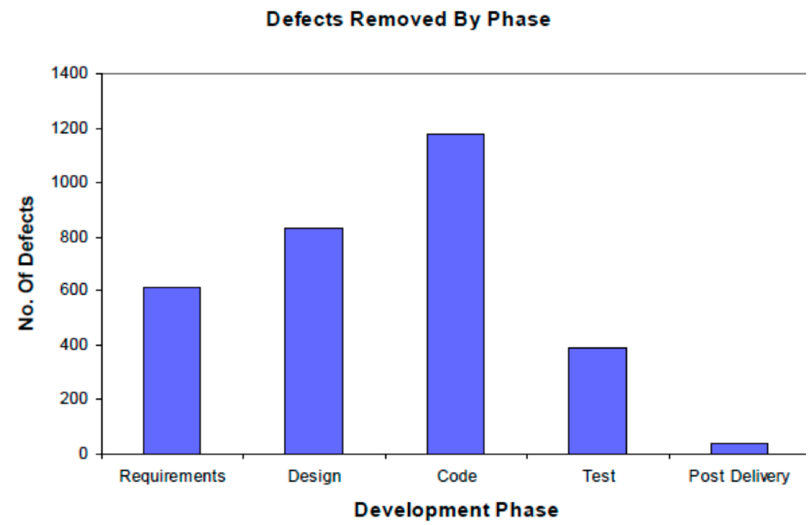


Figure 4: Defects removed by lifecycle phase reported by projects 1995 - 1998

6.4 Project and Company Profitability

In the era before 1992, AIS was profitable in only one out of the five years. Individual project profitability and overall AIS revenue and profits improved significantly as the CPI initiative progressed. In the CMM only era during the years between 1992 and 1995, profit as a percentage of revenue averaged 5.7%. In the CMM+PSP era, profit as a percentage of revenue averaged 9.9% primarily due to reduction in test and rework. These gains occurred at the same time that the AIS organization was experiencing growth from 21 people in 1990 to over 140 people by the end of 1998.

6.5 Peer Recognition

In 1999, the software professional community recognized the AIS engineers and managers with the IEEE Computer Society Software Process Achievement Award, which is similar to the Malcolm Baldrige National Quality award [7]. In addition, the business community recognized AIS with a Blue Chip Enterprise Initiative award given by the U.S. Chamber of Commerce and Mass Mutual. The awards recognized the impact of the CPI initiative in significantly improving AIS financial performance.

7. Holding the Gains

The initiative's results convinced the entire organization that quality had the biggest impact on schedule, cost and profitability and quality work was more predictable. At the same time, we recognized that in order to hold the gains, we needed to focus on training, teamwork, and making disciplined commitments while meeting customers' increasingly demanding time to market needs.

Our actions included:

1. Implemented a Software Engineering Certificate program and required that all AIS engineers and managers must complete the following courses:
    a. Requirements engineering and management
    b. Software Inspections
    c. PSP for Engineers
    d. Managing the Software Process
    e. Managing Technical People
2. Piloted the Team Software Process (TSP) during 2000 -2001[8].

8. Making Disciplined Commitments

After the successful pilot, AIS adopted the TSP. Software project teams utilize the multi-day team launch mechanism of the TSP to ensure that teams tailor the AIS CMMI Level 5 process and create detailed granular project plan containing hundreds of individual tasks. AIS policy requires that all team members participate in the team launch. Team members with the help of a TSP coach, estimate effort and schedule based on personal historical data. The teams successfully negotiate an aggressive and realistic schedule with the stakeholders.

9. The Results 2002 - Present

9.1 CMM/CMMI Assessments

Table 1 shows the history of AIS's CMM/CMMI assessments and the progression to CMMI Maturity Level 5. AIS is only one of 18 organizations and the only small business in the U.S. assessed at CMMI Maturity Level 5 [9].

Date	Levels Assessed	Levels Satisfied	Assessor
Nov 2002	SW-CMM Levels 2 to 4	3	External
Nov 2004	SW-CMM Levels 2 to 4	4	Internal
Dec 2005	SW-CMM Levels 2 to 5	5	Internal
Dec 2007	CMMI Maturity Levels 2 to 5	5	External
Dec 2010	CMMI Maturity Levels 2 to 5	5	External

Table 1: History of AIS CMM/CMMI Assessments

9.2 Understanding What We Do and How We Do It

The CMMI Level 5 process has given us a greater understanding of how the organization does the software application development work, the capability of the organization's defined processes and sub-processes, the ability to analyze the common and special causes of variation and institutionalize defect prevention activities. Such an understanding has helped the organization make business decisions such as firm fixed price contracts, offer performance guarantees, and provide greater value to customers.

9.3 Changed Organizational Culture

AIS engineers submitted more than 1,400 individual PIPs and evolved the AIS process from SW-CMM Level 1 to Level 5, and then to CMMI Maturity Level 5. The AIS staff is to be commended for broad based involvement in continuous improvement of the AIS defined process and achieving measurable results for rework reduction, early defect removal, improved customer satisfaction and predictable outcomes for schedule, cost and quality. One of the benefits of a high maturity workforce is what we call Level 5 behavior that ensures that the organization will improve continually and forever because of its people.

9.4 Self-managed Teams

The TSP practices assisted by a coach enabled AIS project teams to jell at project initiation and to manage themselves throughout the project duration. The external CMMI appraiser identified the following organizational strengths in the appraisals conducted in 2007 and 2010:

- TSP coaches provide continuous mentoring for project team members.
- Process focus at all levels of the organization.
- Open communication.
- Self-managed team structure and roles.
- Individuals with:
  - * Strong quality focus
  - * Commitment to customer and organization
  - * Sense of ownership
- Opportunity for involvement with multiple groups within the organization.
- Empowered to make decisions that affect the organization.

9.5 Processes Under Statistical Control

Project teams utilize statistical process control charts to analyze variation in results for schedule, effort, inspections, and defects. The teams supported by the SEPG use the charts to analyze the common and special causes of variation and verify the impact of changes to the process.

The following charts are from the SEPG presentation in a quarterly status review meeting in December 2010.

9.6 Predicting Quality

TSP teams make detailed quality plans during the TSP team launch. The quality plan is based on historical data on defect injection rates, and removal yields by lifecycle phase. Figure 9 shows the planned vs. actual defects removed by lifecycle phase. This data is valuable in helping teams determine whether or not they are meeting the quality plan. Teams use the data to report weekly the estimated number of defects in systems and user acceptance test and take corrective action, if needed, to meet project's schedule and effort commitments.

One of the more significant payoffs from high maturity is the ability to predict if a component is likely to have defects in downstream integration, system, and user acceptance testing. It is extremely useful to know which components are likely to be error-prone so that we can be pro-active and take corrective action.

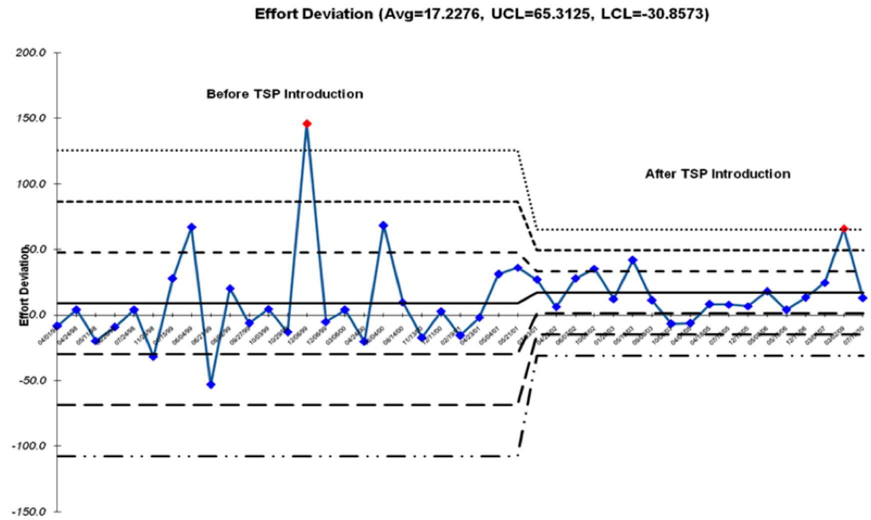


Figure 5: Impact of TSP on projects' effort deviation

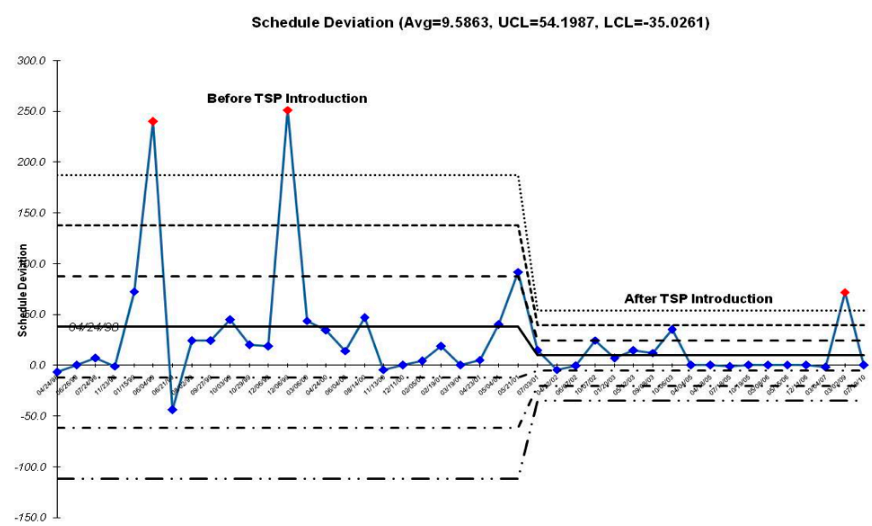


Figure 6: Impact of TSP on projects' schedule deviation

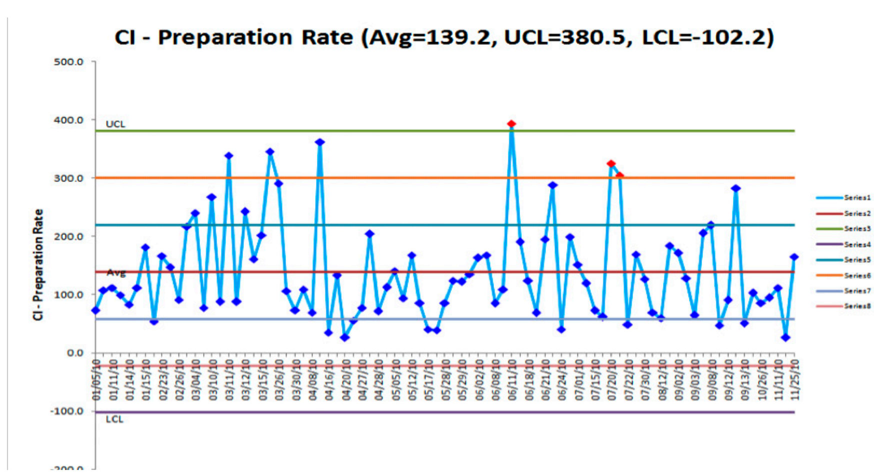
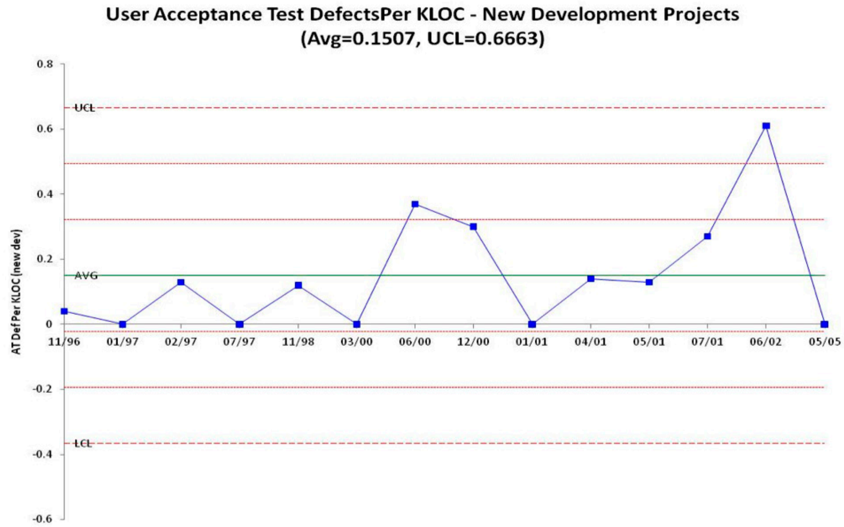


Figure 7: Ensuring effectiveness of peer review (inspection) process



Figures 8: Defect density of new development projects

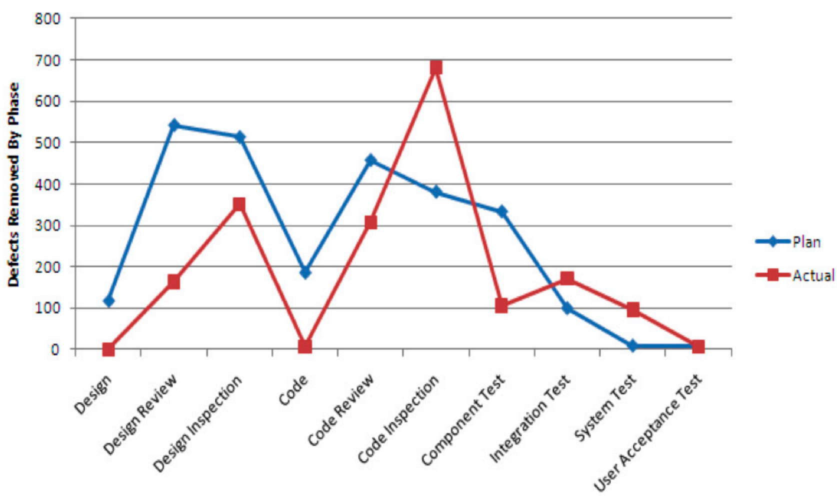


Figure 9: Planned vs. actual defects by lifecycle phase reported by a project in 2009

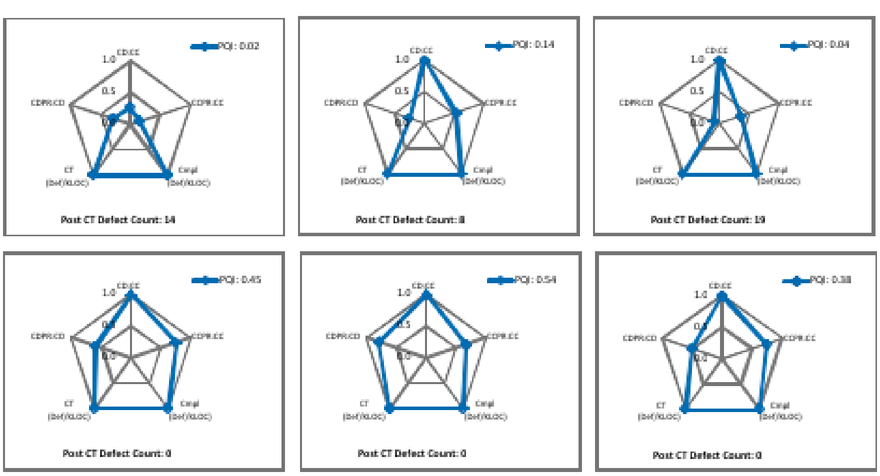


Figure 10: Component quality profile and PQI

The TSP uses a Process Quality Index (PQI) metric to predict whether components that have been unit tested will have downstream defects in integration, system and customer acceptance testing. The PQI is calculated by multiplying together five components of the quality of a component to give a value between 0.0 and 1.0. Experience to date shows that, with PQI values above about 0.4, components typically have no defects found after development. By plotting five quality measures on radar charts (as shown in Figure 10), a component's potential quality problems can be identified in time to take corrective action.

Figure 10 shows a real life example from an AIS TSP team project. The components that had PQI less than 0.4 had integration and system test defects while the components that had PQI greater than 0.4 had zero defects in integration and system defects.

9.7 Scaling Up to Larger Teams, Ensuring Zero Vulnerability and Low Staff Turnover

With processes and sub-processes under statistical control, AIS is able to field larger team sizes and maintain schedule, cost, and quality within known AIS process capability. An AIS team of 17, recently delivered more than 500,000 lines of VB.Net and SQL code on time to a federal agency on a firm fixed price contract. To ensure compliance with the Federal Information Security Management Act, we conducted two independent tests to detect vulnerability in the code. Both tests reported zero vulnerability. Such unprecedented quality performance is due to high maturity practices and the constancy of purpose of the AIS staff with quality as the number one goal. There has been no staff turnover on this project since 2008. For about half the team members, this was their first job straight out of college. We are confident we can scale up our high maturity practices up to 1 million lines of code developmental effort with a team size greater than 25 and maintain AIS historical averages for schedule, cost and quality performance.

9.8 Voice of the Customer

When AIS teams complete a project phase, the AIS SEPG conducts project phase review with customer input on whether the AIS team met or exceeded customer expectations and needs for quality, value, and timeliness. Table 1 shows a summary of the SEPG reports in quarterly status review meetings since 2001 indicating that AIS teams consistently meet or exceeded customer needs and expectations.

Percent customer responses indicating	Quality	Value	Timeliness
AIS team met or exceeded needs and expectations	90.7	95.6	90.4
AIS team needs to improve	9.3	4.4	9.6

Table 2: Customers' phase review responses

9.9 Performance Metrics That Matter

AIS averages for performance metrics that matter are superior to industry averages (Table 2). Customer benefits include significantly less time in acceptance test (agility), and lifetime warranty on defects found in production use (quality). In AIS projects, cost of finding and fixing bugs is no longer the number one cost driver in software development. Customer and AIS staff have more time for new features, enhancements, and technology solutions (innovation). Reduced rework and predictable development schedules lead to work/life balance. High performance jelled team environment leads to zero to low staff turnover (joy in work).

Performance Metrics That Matter	Industry Average	AIS Average
Schedule deviation	>50%	<11%
No. of defects in delivered product 100,000 LOC	>100	<15
% of design and code inspected	<100	100
Time to accept 100,000 LOC product	4 months	5 weeks
% of defects removed prior to system test	<60%	>85%
% of development time fixing system defects	>33%	<10%
Cost of quality	>50%	<35%
Warranty on products	?	Lifetime

Table 3: Benchmarking performance metrics that matter

10. The Future

10.1 Business Strategy Based on Excellence

AIS is now able to align its business strategy with the federal government's move to a larger percent of firm fixed price performance based contracting. Based on the performance of AIS teams during the past 10 years, we now offer guarantees for cost, schedule, agility and quality (Table 4).

Cost
Firm fixed price upon acceptance of requirements specifications
Schedule
Not to exceed 10% of committed schedule
Weekly status reporting with ability to detect as little as one-day schedule slip
Agility
Time in test significantly less than customer's historical average
Rework time significantly less than customer's historical average
Quality
Acceptance test defects significantly lower than customer's historical average
AIS *will fix defect found in production use free for the life of the product!!!*

Table 4: AIS performance guarantees

10.2 Joy In Work: the Ultimate Pay-off

High maturity by itself may not provide sustaining competitive advantage in the knowledge economy. Jelled teams that are self-directed and find joy in work will continue to out perform industry averages for knowledge work.

10.3 High Maturity: Not the End, but the Beginning

For us, high maturity is not an end, it is just the beginning in helping us understand the leadership challenge in the knowledge economy, make the connection between agility, quality, innovation, joy in work, profits and human values, and begin transformation to a life of greatness.

It is Hard to Believe, Unless You Do It

We conclude with our belief that constancy of purpose and making quality the number one goal are more important than the specific methods or models used. We realize that for many of the skeptics, no amount of data will convince them that high maturity is worth it. To them, we say, "It is hard to believe, unless you do it."

Disclaimer:

CMMI® and CMM® are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

ABOUT THE AUTHOR

Girish Seshagiri is CEO of AIS, a winner of the IEEE Computer Society Software Process Achievement Award. AIS is one of the very few organizations in the U.S. whose software process capability is assessed at SEI CMMI Level 5. Seshagiri has more than 30 years experience managing technical teams. He has an MBA degree from Michigan State University.

Girish Seshagiri
Phone: 703 426-2790
E-mail: girish.seshagiri@advinfo.net

REFERENCES

1. Crosby, Philip B., "Quality is Free", McGraw-Hill, 1980.
2. Deming, W. Edwards, "Out of the Crisis", MIT Press, 1982.
3. Drucker, Peter F., "The Effective Executive", Harper Colophon, 1985
4. Humphrey, Watts S., "Managing the Software Process", Addison-Wesley, 1989.
5. Humphrey, Watts S., "A Discipline for Software Engineering" Addison-Wesley, 1995.
6. Kaplan, R., Norton, D., "The Balanced Scorecard", Harvard Business School Press, 1996.
7. IEEE Computer Society, " IEEE Computer Society Awards ", <http://www.computer.org/portal/web/awards/processachievement>
8. Humphrey, Watts S. "The Team Software Process", Technical Report, CMU/SEI- 2000-TR-023, ESC-TR-2000-023, Nov, 2000.
9. Wilson, Hal., "Publicly reported data begging for analysis", CMMI Technology Conference, November 1999.