Integrated Design and Process Technology, **IDPT-2006**
Printed in the United States of America, June 2006

Process Improvement with Human Performance Technology

Timm J. Esque

Esque Consulting, LLC
1443 E. Colt Rd.
Tempe, AZ 85284

ABSTRACT

Formal documentation and enforcement of a "required process flow" is a common management response when product development is going bad. This paper will describe a real case at Intel Corporation where the required process flow did not improve performance, but another approach did. A method evolved where each product development team defined their own specific process flow, and committed to it, with the entire team in a room together for one or two days. This was one element of getting the team working in the "3 Performance System conditions" which led to improvement of productivity by 30-50%, significantly greater quality and less overtime.

The Performance System conditions and related tools are from the field of Human Performance Technology (HPT). This field sheds light on the key factors that impact performance as well as why traditional project management so often results in mediocre performance in today's fast-paced, uncertain world. The presentation will include some key HPT principles and a real case with pre and post performance data.

NOMENCLATURE

A = accomplishment

B = behavior

C = consequence

P = performance

W = worthy performance

INTRODUCTION

Thomas F. Gilbert, know to some as the father of Human Performance Technology, defined a technology as "an orderly and sensible set of procedures for converting potential into capital"(1978). If that is true, then conventional project management, as represented by "A Guide to the Performance Management Body of Knowledge" (Duncan, 1996) for example, is certainly a technology. The very ubiquity of Project Management Technology (PMT) would suggest it is adding significant value, but is it still doing so in today's rapidly changing and uncertain world? One simple test would be the extent to which when implemented properly, PMT reliably ensures that a project meets its goals. In this paper, the test just suggested is applied to a related but lesser known technology called Human Performance Technology (HPT). In HPT, the science of human performance is applied "…to bring about changes to a system, in such a way that the system is improved in terms of the achievements it values" (Stolovitch, 1992). This paper describes how HPT was applied to improve a product development system, which purported to value time, quality and economy, but which often failed in regards to one or more of these values, even with the aid of PMT.

A technology of performance cannot exist without a clear definition of performance and a way to reliably measure it. In his book, "Human Competence: Engineering Worthy Performance" (1978), Gilbert, recognized that reliable measurement of performance would be a function of how it was defined. Gilbert begins with the observations that behavior is a core component of performance and that the only reason for engineering behavior is to achieve some desired end or consequence. Hence, performance (P) is a transaction involving behavior (B) and Consequence (C), or:

$$P = B \rightarrow C \qquad (1)$$

It is not just any consequence that is desired, but a valuable one – an accomplishment (A). So valuable performance is a transaction involving both behavior and accomplishment, or:

$$P = B \rightarrow A \qquad (2)$$

Gilbert's definition is critical to measuring performance because it is almost always easier to measure the quality, timeliness, and cost of accomplishments (outcomes, deliverables, etc.) than it is to measure the quality, timeliness and costs of behavior. Gilbert takes this a step further with his theorem that worthy performance (W) "…is a function of the ratio of valuable accomplishments (A) to costly behavior (B), or:

$$W = A/B \qquad (3)$$

It is this theorem that stands project management technology on its head. PMT emphasizes the analysis of tasks, durations and effort to plan and predict project performance. When schedules are not being met, one of the prime strategies is to increase effort by adding resources and/or having performers work longer hours. But according to Gilbert's theorem, all other things being equal, more effort (behavior) only adds to the cost of worthy performance. On the other hand, when managers and the performers are focused on achieving planned accomplishments, they are much more likely to course correct with different behaviors, rather than more of the same ones. The case study that follows will illustrate this important principle.

Gilbert applied HPT by taking an accomplishment-based approach to analyzing performance problems and designing jobs. Brethower (1970) demonstrated the power of applying HPT proactively to plan and manage accomplishments over time in classroom environments. Feeney (Jordan,1973), Daniels (1995 and 1997), and Esque (1997, 1999 and 2001) have since demonstrated the impact of accomplishment-base management in for-profit organizations. In each accomplishment-based management demonstration, three common conditions are created: 1) accomplishment-based plans are derived from organizational goals and articulated to the individual performer level (clear expectations), 2) each organizational level reviews progress against planned accomplishments at short intervals (frequent self-monitored feedback), and 3) differences between planned and actual accomplishments are used to manage resources for the next interval (control of resources). These three conditions will be referred to as Performance System (PS) conditions.

If the three PS conditions are so important to reliable and productive human performance, than it is notable how unlikely it is to find performers working in these conditions in typical organizations. This paper describes the behaviors and performance of a highly visible product development team at Intel Corporation before and after these three PS conditions were in place.

A CASE STUDY

In the spring of 1994, Intel delayed the launch of one of its major microprocessors more than two months because a supporting product, a chipset, was not ready to launch. The chipset group had been reporting that the chipset would be ready right up until a few weeks before the planned launch. The group's senior management, and all the other senior stakeholders were surprised by the slip, but they shouldn't have been. An examination of the performance of the three previous chipsets revealed that typical performance against three key success indicators had been equally poor (see Table 1). Performance against scheduled completion dates was averaging 30-50% slips. Three or four design revisions were occurring before high volume manufacturing even though project plans stipulated one design revision. In regards to output quality, it was typical for design errors to warrant another redesign not once, but twice, after the product was already in high volume manufacturing

The chipset group was put on notice that their performance better improve – or else. The first attempt to improve performance followed a common strategy in product development. A hand-selected cross-functional team was assembled to clarify the process by which chipsets should be designed. Their output was a process flow document, listing the required sequence of outputs and key success factors for each output required to successfully produce a chipset**.** In several cases, the specific tactics and tasks (best known methods) for producing functional team outputs were also documented. Each of the functions now knew explicitly what was expected of them at each design stage, according to this select task force. This "required process flow" document was made available to be followed by the entire organization including everyone on and supporting the team designing the next chipset.

Unfortunately, this next chipset project performed about the same against the three defined indicators of success. The performance of this chipset project went somewhat unnoticed because it was not supporting a major microprocessor launch. However, the next chipset project would support a major microprocessor launch. This created a situation where the development manager and the project manager for the next chipset project were very open to new ideas about how to ensure that the next chipset project succeeded. The author and a colleague, being from outside the chipset organization but having been observing and helping since the initial fiasco, offered to help.

First, the managers were given some objective feedback about some of the behaviors that seemed to be associated with the poor performance results. In hindsight, it is understood that these behaviors are common on projects where plans are relatively uncertain and high schedule pressure exists. Esque (1999) describes this pattern of behaviors in detail and refers to them as the project management vicious cycle. To sum up the pattern in a couple sentences, management inadvertently increases schedule pressure on the project teams without providing the necessary support to meet the schedule. In response, team members tell management what they want to hear, rather than what is really going on, and documented schedules and progress reviews turn into a game where everyone involved is mostly concerned about avoiding blame.

Along with this feedback about what was going on, the managers of the next chipset project were offered an alternative approach. The vicious cycle behaviors create just about the opposite of the 3 Performance System conditions. The managers were instructed on the PS conditions and offered assistance in getting these conditions in place on the next project. They decided to give it a try.

Condition #1: Clear Expectations. A best known method already existed at Intel for setting project goals and cross-functional expectations. The method, called map day, involved getting the entire project team into a room together to create a high level project plan. The map day process was adapted to begin to establish very clear expectations for each project team contributor. The agenda for this adapted map day is shown in Figure 1. Note that it begins with a clarification of the few key goals that will define the success of this project, as well as an explanation of the business imperative for achieving these goals. At the end of the map day, the team has a high level, accomplishment-based project plan for the entire scope of the project. In addition, a few of the early deliverables are clarified in more detail, with input of all stakeholders. Sub-teams are only asked when they can complete there first deliverables after this clarification and negotiation. Dates for later deliverables are not defined until current planned deliverables have been completed (sometimes referred to as horizon planning). Another way of describing what happens in map day is that the members of the project team create their own agreed upon process flow document, just for this project. Instead of a glossy color graphic of how some select team expects all chipset projects to proceed, they have a chart on the wall covered with yellow sticky notes, representing how they have decided they should proceed, and who is going to produce what, in order to meet this project's specific goals.

The outputs of map day are necessary but not sufficient to ensure clear expectations. Truly clear expectations define what each individual team member needs to accomplish this week (or even more frequently) in order to stay on the project plan. This level of clarity was achieved with further planning in each of the functional subteams (e.g. marketing, design, test, manufacturing, etc.). For example, the design subteam's first major deliverable was the micro-architecture spec. On past projects, subteams had started with a high level Gantt chart and delineated more specific tasks along with estimated durations of each task. To put the focus on accomplishments, the subteams on this project were asked to break their accomplishments (deliverables) from map day down into several sub-accomplishments that could be completed by one person in one week or less. For example, the design subteams which tended to divide up their work into functional unit blocks (or FUBs) decided they could break FUBS into sub-FUBs that met the one person, one week criteria. Instead of a detailed Gantt chart, their plan took the form of a list of sub-FUBs, each with an individual owner and week for completion (see Figure 2). To the extent that each individual had similar clear expectations, he or she was operating with clear expectations.

Condition #2: Frequent (Self-monitored) Feedback. Remember that on past chipset projects, the data in the schedules had become irrelevant, meaning not reflecting reality and therefore useless for making decisions. In an environment where there is uncertainty, it is critical to make good ongoing decisions about how to get/keep a project on track. This requires accurate status information which requires frequent self-monitored feedback. Frequent self-monitored feedback means that each team member (and hence each subteam, and the overall team) knows where he stands against the project plan at all times.

As mentioned in the introduction, planning and focusing on accomplishments (vs. tasks and activities) is critical to effective measurement of performance. A subteam of design engineers need to elicit a broad array of behaviors and activities in order to produce an effective micro-architecture spec. It would be very difficult to watch these design engineers and determine if they were doing just the right behaviors and doing them well. On the other hand, once each design engineer has signed up to produce specific sub-FUBS, and the quality criteria for those sub-FUBs has been agreed upon, it is relatively easy to determine if those sub-FUBS are in fact done, or not done, when promised. Typically, each subteam would assemble each week for about 15 minutes to determine if their planned sub-accomplishments were on track to be done or not, and to clarify the expected sub-accomplishments for the following week.

Of course, if the subteam was not on track, these meetings would shift to deciding how to get back on track (and whether help from outside the subteam was needed). But once clear expectations and frequent self-monitored feedback was in place, most subteams were on track the vast majority of the time. This is because team members did not wait until the end of the week to speak up if they needed help. No one had twisted anyone's arm to promise deliverables faster. It was perfectly clear what each team member needed to accomplish to stay on track and everyone knew they'd be reporting out at least weekly. So as long as team leaders responded appropriately when a team member asked for help (which they were coached to do), people did speak up, issues were quickly addressed, and the overall project generally stayed on track. It should be noted that the project manager and all the subteam leaders also met weekly to review status on the high level deliverables and address any anticipated cross-functional issues.

Condition #3: Control of Resources. Control here refers to course corrections taken when it is determined that an individual, subteam or whole team is, or is likely to soon be, in jeopardy of missing specific expectations. While this seems the most applicable definition in a management context, control can also be about getting other things and people to behave in accordance with ones own desires. Someone who has absolute control of all the required human and other resources needed to complete a project would seem to be in a good position to ensure a successful project. Indeed, the vicious cycle behaviors mentioned previously seem to come partly from the assumption that project sponsors and project managers should be able to control the resources they need to successfully complete a project. However, the process being followed here assumes the opposite, that it is very unlikely to attain absolute (or even dominating) control over required resources, especially the human ones. However, it is possible for each member of a team to control him or herself very well. And in order to do this, they need a sense of control over the resources required to meet their own expectations. This individual sense of control comes mostly from the response team members and team leaders (and even the project manager) get when they ask for help to stay or get back on track.

So, while the first two conditions have mostly to do with the behavior of the performers, the third condition has more to do with the behavior of all levels of managers. The appropriate management behaviors (for getting into the 3 PS conditions) begin back at the early goal setting stage. The chipset group managers were under tremendous schedule pressure (from their bosses and customers) and they were acting as if setting or encouraging very aggressive goals is the most important step to ensuring a fast and successful project. At least one body of research suggests that this is true to a point, but that goals can also be too aggressive. McGrath (1970) describes the impact of task pressure on performance with an inverted U-shaped curve. More task pressure correlates to better performance up to a point, and then any more pressure beyond that is correlated with reduced performance. In this case, it was recommended to the chipset group managers that while continuing to monitor and look for ways to strive for market driven schedule goals, they should let the project team manage itself to the goals they came up with in map day.

It was suggested that only by letting the team operate against what they considered reasonable goals could they expect to ever get any accurate status information from the team. This is of course necessary to make good management decisions and increase the chances of successful performance. Although senior management in the chipset group did not always follow this recommendation, the project manager and subteam leaders did, and largely protected the team from what was considered unreasonable schedule pressure. When individuals or subteams proactively spoke up about goals being in jeopardy, the response was generally one of working to get them the resources required to meet the stated expectation. In many cases this involved primarily giving individuals more control over their own time, or negotiating better cooperation from other team members or functions. In other cases, data was produced to show that resources were inadequate to achieve the stated goals, and upper management was given the choice of providing the resources or changing the stated goals accordingly. An example of the later is when it was shown with test time data that there were inadequate test stations to run all the planned tests in the given timeframe, even running around the clock on shifts. In this case, more work stations were produced within 24 hours, and the test goals were met.

In the context of implementing the 3 PS conditions, control of resources is about effectively encouraging team members to speak up when they don't think they have what they need to be successful, and for managers to always respond constructively, such that resource issues are resolved or project and schedule goals are altered to be consistent with available resources. It should be noted that when too big a gap exists between available resources and the desired goals, the appropriate business decision might be to discontinue the project, or reconfigure multiple projects so that the remaining ones are set up for success. These are difficult decisions and it is senior management's role to make them as best they can. Making these decisions well also requires accurate status reporting and effective self-management of each project and sub-team.

The chipset team had committed to a complete its first milestone five weeks out from the initial map day. They had specified a handful of specific functional team deliverables, each with dates and quality specifications. The subteams were coached on defining and reviewing weekly deliverables, and the team leaders and project manager were coached on responding appropriately to early warnings of possible issues. When the team completed the first milestone on time, they grew committed to the 3 PS conditions. They held another map day and committed to another milestone several months out this time. In all there were four map days prior to the completion of the first design. When the project manager published the team's commitment to the complete first design (the deliverable is called tapeout), upper management pressured him to "pull it in". He told his bosses that the whole team was looking for ways to get done faster, and he was open to rational plans for doing so, but he was not asking the team to simply get done faster without a rational plan to do so. Somehow he survived as project manager in spite of this. About six weeks before the team completed the first design, two key resources were pulled form the project to "save" other projects. Meanwhile, the chipset team members were starting to believe that they might actually meet their commitments. A situation they hadn't experienced in a long time.

RESULTS

Table 2 shows the performance of the chipset team operating in the 3 PS conditions compared to the performance of the 5 previous chipset projects. While the team did not achieve the desired top down deadline, they missed it by only 10% rather than the previous 30-50%. An indicator that the previous projects didn't have was a team commit date. By hitting this date, the team had accurately predicted their completion date several months in advance, in contrast to past teams being unable to predict their slips even weeks in advance. A final note about schedule performance is that the team met their commit date even though two key resources had been "pulled to save other projects" with six weeks remaining in the schedule.

A key reason the team performed so much better against schedule was because they had only one design revision before going to high volume production. Although one design revision was always assumed in the plan, this was the first chipset project team to accomplish it. In post mortem focus groups, achieving product specifications with one design revision was largely attributed to clarifying deliverables and negotiating their required quality in detail at the early stages of planning. It was also suggested that by mitigating some of the top down schedule pressure, the subteams didn't feel the need they had in the past to "trade off quality for schedule."

The key test of whether going to high volume manufacturing was a wise decision or not, was if any design revisions were required after high volume began. Unlike previous chipset projects, this chipset had zero design revisions after high volume production. In actuality, the quality of this product was evident to key customers even in the early samples. As a result, at least one top tier customer cancelled orders for a previous generation chipset and chose to wait for this one to ramp up production. This contributed to this chipset becoming Intel's highest revenue producing non-microprocessor product.

One other very interesting pattern emerged in the post-mortem focus groups. In the post-mortems of some of the previous less successful chipset projects, the notion of "panic mode" had come up frequently. Team members talked about how early in the project they had to go into "panic mode", meaning increased top down schedule pressure and lots of obligatory overtime. When this chipset team was asked if and when they went into panic mode, most of them said "never". They said that for several days prior to each commit date everyone was working extra hours, making sure that they would absolutely, positively meet their commitment. But that this was self-imposed and so didn't feel like "panic mode".

DISCUSSION

Being by nature a case study rather than a controlled experiment, this study raises a number of interesting questions, without definitively answering any. The questions chosen to explore in this discussion are: 1) was the change in performance of the chipset project team just a fluke?, 2) If not, then is HPT (and specifically management of accomplishments through the three PS conditions) a "silver bullet" technology destined to change the face of project management in all environments?, and 3) Is HPT being recommended as a replacement for PMT?

If the results of this chipset project team were a fluke, then they are a replicable fluke. Although many of the potential variables have not and probably could not be controlled, several other project teams have demonstrated similar turnarounds against similar project goals when the 3 PS conditions are consciously introduced and set up in the project environment. At Intel, one notable microprocessor development subteam excelled against the project wide goals while surrounded by other development subteams using traditional methods and getting mediocre results. Esque (1999) documents an Intel IT project where this author applied the 3 PS conditions to help a team distributed across 25 geographic locations successfully pull in the launch of a major new administration system by a month during the last six months of the project duration. A major designer and manufacturer of large scale routers achieved product development goals almost identical to the chipset project team, prompting the CEO to say "With this approach, when I'm told it will be done, I can count on it". The point being that similar applications of the 3 PS conditions and similar results have been achieved several times since the chipset project team turnaround in 1996.

At the same time, it is way too soon to view this particular application of HPT as a silver bullet. While the 3 PS conditions, when systemically implemented, seem to always lead to reliable and improved results, many team and organizational leaders have expressed an interest in achieving those results, while being unwilling to change the behaviors necessary to achieve them. In the years following the chipset project success story at Intel, many Intel managers requested help implementing map day and the 3 PS conditions. There was a common pattern of these managers supporting the notion of a whole team planning meeting, then insisting that the team commit to an end date for the project in the first map day (preferably the "suggested" commit date). While they loved the idea of subteams and project teams rigorously tracking and reporting progress on a weekly basis, many were less enamored with the idea that if a rational plan for resources didn't support the goals, then the goals had to change. Resistance can also come from the bottom of the organization. Sometimes fierce enough to prevent the 3 PS conditions from ever getting established. One of the predictors of the level of resistance seems to be the extent to which the target team/organization sees itself as failing. For example, there was very little doubt in the chipset organization after missing the major launch that something significant was broken. Other organizations exhibit vicious cycle behaviors, but still perceive themselves as successful enough within their target markets to risk making any major behavioral changes (especially when the economic stakes are high). Even when confronted with their own internally documented failures against stated project goals, they justify those failures and resist any significant change.

Equally disturbing as outright resistance to change, is the fact that a number of teams and organizations have achieved and documented significant performance improvements with the 3 PS conditions, only to revert back over time to more traditional management technology. William R. Daniels, who has been implementing the 3 PS conditions for over 20 years, is attempting to overcome this "reversion effect" by getting and keeping at least three levels of the target organization engaged in the 3 PS conditions together and at the same time. Daniels and Esque, (2006) propose an explanation for this reversion effect and describe recent efforts to counteract it. What is most relevant about that explanation to this paper is the focus on the autonomy and control of human beings. To summarize the key point, human beings desire autonomy, while managers in organizations often desire and attempt to control them in order to achieve organizational goals. Documenting and attempting to enforce a required process flow is a common manifestation of this desire to control. It is posited that the 3 PS conditions, when implemented simultaneously at multiple levels of organization, can create an effective balance of this human need for autonomy and control.

These issues may also shed some light on the extent to which HPT and the 3 PS conditions conflict with and/or complement project management technology. First, it is important to clarify that the issue here is not whether PMT was ever a value added technology. The question is the extent to which it helps and/or hinders project performance, specifically in environments under substantial schedule pressure and where plans for getting from here to there necessarily contain a lot of uncertainty. The possibility that PMT may not be as effective in these environments comes from the observation that struggling product development (and other types of project) teams are almost always using some of the basic PMT principles and tools (e.g. Gantt style plans with careful delineation of tasks, dependencies, durations and effort required, rigorously updated at short and regular intervals). In some cases, these projects are being supported or run by certified PMT experts who are also applying more advanced principles (e.g. risk management techniques, buffer management, etc.) . While this suggests that PMT may not be helping much or even hurting performance against project goals, it is also true that some of the teams that have turned around project performance with the 3 PS conditions, continued to use PMT tools to some extent.

A significant variable in this line of inquiry would be the extent to which PMT is being used properly (as intended and documented by the project management profession – in the PMBOK for example). Resolving that is outside the scope of this paper, but there are other reasons to believe that PMT may not be particularly effective in the environments in question.

First, PMT prescribes that an entire project plan should be delineated in some detail at the very outset of a project, in order to estimate the likely completion date and required resources for that project. Experienced project participants will tell you that this first pass estimate, or something close to it, is often used later in the project to apply schedule pressure, even in an uncertain environment where it is known that a number of far reaching assumptions must be made to come up with any estimate at all. From an HPT perspective, truly clear expectations require that commitments be made only as far out on the project horizon as team members can see with reasonable certainty. So there is an agreement up front that estimates used for roadmap planning and marketing purposes, are separate from the dates that the team is holding itself accountable to.

PMT, the way it is commonly practiced, also relies on supporting project analysts to collect, collate and display status data so that a few select managers can use it to decide how best to use project resources going forward.

With today's automated project management tools, it is understandable that managers would see value in keeping a detailed and updated model of the project, and using it to quickly determine how the schedule slips on certain tasks presumably impact other tasks and resources. However, in an uncertain and rapid changing environment, task duration data and effort estimates become meaningless when separated from conversations that team members are having several times a week about what they have done and what they need in order to stay or get on track to the accomplishment-based plan. This author has had many experienced project participants report that they feed data to project analysts when required, even when they consider the automated project plan to be irrelevant.

These are examples of how PMT is often used with good intentions, but without achieving the intended consequences that are valued by the systems in which it is applied. It is being suggested here that these are also examples of failing to find the proper balance between human autonomy and control. The way it is often applied, PMT is a technology that assumes that by getting all the individual team members to contribute data just about their individual work, and using automated tools to organize and report this data, a somewhat separate and select group of individuals can then make the key decisions to ensure project success. In other words, there is a belief that with the proper data and tools, these select decision makers can control all the project resources, including the human ones, to their desired ends. HPT, through implementing the 3 PS conditions, assumes that in high pressure and uncertain environments, it is impossible for a select few to analyze, comprehend and control complex project resources and determine how to best utilize them to achieve specific ends. The correct balance between autonomy and control comes from every member of the team controlling him or herself really well, as occurs when operating in the 3 PS conditions.

REFERENCES

Brethower, D. M., 1970, "The Classroom as a Self-modifying System," PhD dissertation, U. of Michigan.

Daniels, W. R., 1995, "Breakthrough Performance: Managing for Speed and Flexibility, ACT Publishing, Mill Valley CA.

Daniels, W. R. and J.G. Mathers, 1997, Change-ABLE Organization: Key Practices for Speed and Flexibility, ACT Publishing, Mill Valley, CA.

Daniels, W.R. and Esque, T. J., 2006, "Performance Improvement: Enabling Commitment to Changing Performance Requirements" in J. Pershing's (Ed.) "Handbook of Human Performance Technology: Principles, Practices and Potential, 3rd Edition, Pfeiffer, San Francisco, CA.

Esque, T. J., 1999, "No Surprises Project Management: An Early Warning System for Staying on Track", ACT Publishing, Mill Valley CA.

Esque, T. J., 2001, "Making an Impact: Building a Top-performing Organization from the Bottom Up", CEP Press, Atlanta GA.

Esque, T. J. and McCausland J., "Taking Ownership for Transfer: A Management Development Case Study", Performance Improvement Quarterly 10, no. 2.

Gilbert, T. F., 1978, "Human Competence: Engineering Worthy Performance" McGraw Hill, New York NY (republished in 1996 by ISPI, Washington, D.C.)

Jordan, P. (Director), 1972, "Business, Behaviorism and the Bottom Line" [Motion Picture], The B. F. Skinner Film Series, CRM Productions, Carlsbad CA.

McGrath, J. E., 1970, "Settings, Measures and Themes: An Integrative Review of Some Research on Social-psychological Factors in Stress", in J. E. McGrath's (Ed.) Social and Psychological Factors in Stress. Holt, Reinhart and Winston, Inc., New York, NY.

Stolovitch, H.D. and Keeps, E. J., 1992, "Handbook of Human Performance Technology: A Comprehensive Guide for Analyzing and Solving Performance Problems in Organizations" Jossey-Bass Publishers, San Francisco CA.

FIGURES AND TABLES

Table 1 Typical Performance on Four Consecutive Chipset Development Projects

Indicator	Performance
Performance against top-down deadlines	30-50% slip
Design revisions prior to hi-volume production	4
Bugs discovered by customers	2

- Intro/Housekeeping
- Business Imperative/Project Goals
- Customer Deliverables
- Internal Deliverables
- Build and Validate Deliverables Map
- Internal Deliverable Quality Criteria
- First Horizon Commit Dates
- Next Steps

Figure 1 Adapted Map Day Agenda

FUB	Sub-FUB	Owner	Commit Date	Done?
1	1A 1B 1C 1D 1E	J. A. J. A. D. O. C. F. L, J.	Week 21 Week 21 Week 21 Week 21 Week 21	
2	2A 2B etc.	J. A. D. O.	Week 22 Week 22	

Figure 2 Example of Planning and Tracking Sub-accomplishments

Table 2 Chipset Development Performance Before and After Implementing the 3 Performance System Conditions

Indicator	Performance Before	Performance After
Performance against top-down deadlines	30-50% slip	10% slip
Performance against team commit date	Didn't have one	0 slip
Design revisions prior to hi-volume production	4	1
Bugs discovered by customers	2	0