Project Performance International

http://www.ppi-int.com/newsletter/SyEN-035.php

SYSTEMS ENGINEERING NEWSLETTER

brought to you by

Project Performance International (PPI)

SyEN 50 – December 18, 2012

Dear Colleague,

SyEN is an independent free newsletter containing informative reading for the technical project professional, with scores of news and other items summarizing developments in the field, including related industry, month by month. This newsletter and a newsletter archive are also available at www.ppi-int.com.

Systems engineering can be thought of as the problem-independent, and solution/technology-independent, principles and methods related to the successful engineering of systems, to meet stakeholder requirements and maximize value delivered to stakeholders in accordance with their values.

If you are presently receiving this newsletter from an associate, you may receive the newsletter directly in future by signing up for this free service of PPI, using the form at www.ppi-int.com. If you do not wish to receive future SE eNewsletters, please reply to the notifying e-mail with "Remove" in the subject line, from the same email address. Your removal will be confirmed, by email.

We hope that you find this newsletter to be informative and useful. Please tell us what you think. Email to: contact@ppi-int.com.

WHAT'S INSIDE:

WHAT'S INSIDE:

Quotations to Open On

Read More…

Feature Article

Predictable Projects: How to Deliver the Right Results at the Right Time

Read More…

Additional Article

Extreme Review - A Tale of Nakedness, Alsatians and Fagan Inspection

Read More

Systems Engineering News

- The BABOK® Guide Available in French

- INCOSE 2012 Elections

- The SEI/AESS/NDIA Business Case for Systems Engineering Study: Results of the Systems Engineering Effectiveness Survey Released

Read More…

Featured Society

- Operations Research Society of South Africa (ORSSA)

Read More…

INCOSE Technical Operations

- INCOSE UK Railway Interest Group

Read More…

Systems Engineering Tools News

- Visure Announces Visure Requirements 4.4 (Previously IRQA)

Read More…

Systems Engineering Books, Reports, Articles, and Papers

- Aligning SysML with the B Method to Provide V&V for Systems Engineering

- The System Concept and Its Application to Engineering

- Metrics for Requirements Engineering and Automated Requirements Tools

- Systems Analysis and Design with UML, 4th Edition

- System Safety Engineering and Risk Assessment: A Practical Approach (Chemical Engineering)

- Systems Architecture, 6th Edition

- Systems Thinking: Managing Chaos and Complexity: A Platform for Designing Business Architecture, 3rd Edition
- INCOSE INSIGHT October 2012, Volume 15, Issue 3 Released

Read More…

Conferences and Meetings

Read More…

Education and Academia

- The Systems Engineering Research Center
- Comprehensive Modeling for Advanced Systems of Systems

Read More…

Some Systems Engineering-Relevant Websites

Read More…

Standards and Guides

- Executable UML/SYSML Semantics Project
- ANSI/AIAA G-043A-2012 - Guide to the Preparation of Operational Concept Documents - Released
- G-118-2006e - AIAA Guide for Managing the Use of Commercial Off the Shelf (COTS) Software Components for Mission-Critical Systems
- G-077-1998e - AIAA Guide for the Verification and Validation of Computational Fluid Dynamics Simulations
- G-035A-2000e - AIAA Guide to Human Performance Measurements
- R-100A-2001 - AIAA Recommended Practice for Parts Management
- S-117-2010e - AIAA Standard — Space Systems Verification Program and Management Process
- S-102-1-4-2009e - ANSI/AIAA Performance-Based Failure Reporting, Analysis & Corrective Action System (FRACAS) Requirements
- S-102-1-5-2009e - ANSI/AIAA Performance-Based Failure Review Board (FRB) Requirements
- S-102-2-18-2009e - ANSI/AIAA Performance-Based Fault Tree Analysis Requirements
- S-102-2-4-2009e - ANSI/AIAA Performance-Based Product Failure Mode, Effects and Criticality Analysis (FMECA) Requirements
- S-102-2-2-2009e - ANSI/AIAA Performance-Based System Reliability Modeling Requirements
- R-013-1992 - Software Reliability

Read More…

Definitions to Close On – Factory Acceptance Testing, Site Acceptance Testing and Commissioning

Read More…

PPI News

Read More…

PPI Events

Read More…

Quotations to Open On

This feeling, finally, that we may change things - this is at the centre of everything we are. Lose that... lose everything.
-- Sir David Hare

Change is not made without inconvenience, even from worse to better
-- Samuel Johnson

Feature Article

Predictable Projects:
How to Deliver the Right Results at the Right Time

Niels Malotaux
niels (at) malotaux.nl
www.malotaux.nl

1 Can the result and the delivery time of projects be predicted?

This text describes a very pragmatic approach that can quickly make your projects more successful in significantly shorter time, without bad stress, at the same time being able to predict what will be done when, with quite remarkable accuracy.

Does this sound impossible? That's what many people thought before they did it. Does it sound incredible? After all, you have been working in projects for many years. You have all the degrees and experience. What can it be that you missed and that will suddenly make your projects delivering better results much faster, while everybody knows that projects easily take longer than hoped for? I almost always hear the reserves like: "It can't be done",

"We've tried it before", "I've heard that before", "It doesn't work here", "Our projects are different", "We build laaaaaarge systems", or "I'm doing these things already". These arguments are quite understandable, because people simply cannot imagine that there are techniques to deliver better results in substantially less time. Otherwise they would be applying these techniques already, wouldn't they?

There are many theoretical processes that promise great results, however, only practice shows whether it works in the real world. Based on 12 years of research and a lot of experience, the approach as presented here has been tested and honed in the practice of small and large projects, in several different countries, cultures, and disciplines. Even if you don't believe it, if those people who did it became some 30% more productive within several weeks, isn't that a good reason at least to try? The proof of the pudding is in the eating. Before you taste the pudding, it's not easy to explain the taste. However, the taste can be quickly tested.

We'll first describe some background, then the different elements of the Evolutionary (Evo) Planning approach, followed by a more elaborated description of the first element of Evo Planning. Finally, we invite you to do a five week exercise to get hands-on experience, because, believe it or not, there is much more detail in it that you can imagine when you just read this text. Further background and elaboration can also be found at www.malotaux.nl while at www.malotaux.nl/downloads several booklets and presentations, a tool, as well as a video about the subject can be freely downloaded. The author invites you to discuss, to agree, to disagree, or even try to get some taste yourself by some free coaching (see last chapter). Email: niels (at) malotaux.nl

2 The importance of time and predictability
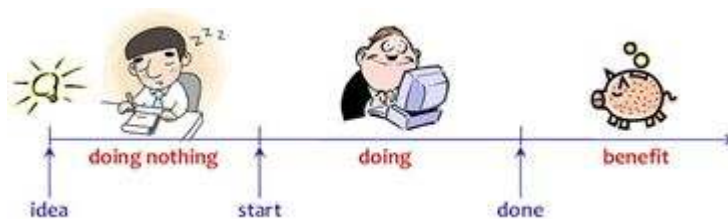
Figure 1: Once the idea of the project is born, the timer starts ticking...

Every day we start a project later, we will finish a day later (Figure 1), depriving us and/or the customer from the revenues which by definition are much higher than the cost of the project, otherwise we shouldn't even start the project. The only good reason why we don't run this project yet is that we currently are spending our limited resources on more rewarding projects.

Initially, every day of the project adds potential value, but gradually we arrive in the area of diminishing returns where every extra day adds less than the return we should expect. This means that delivery time should be appropriately designed for optimum benefit and this puts it directly into the scope of systems engineering and design. After all, the project manager is responsible for delivering the right result at the right time, but the systems engineers, designers, and implementers determine the time it takes to deliver the appropriate system or service.

Engineers often think that the project isn't finished until "all" requirements have been delivered. Well, if delivery time is also a requirement and often the most important requirement, why are all other requirements treated as being more important than the most important one (see also "The Fallacy of 'all' Requirements", www.malotaux.nl/?id=fallacyofrequirements)? The requirements from all stakeholders are often contradictory, so the design process is there to find an optimum compromise for the conflicting requirements.

A product manager at a telecom company told me that 2 people had been analyzing a project to take 4 people 10 months to complete (Figure 2). They said, however, that the development time could be shortened by 4 months if they would invest in a system costing €200k, but that the business was not prepared to invest this amount of money.
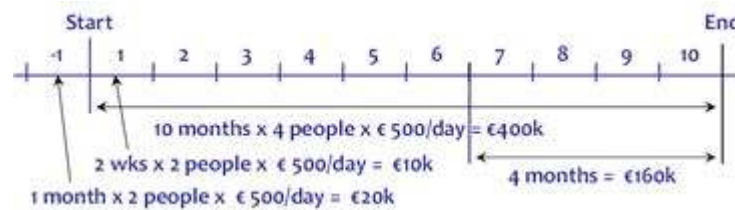
Figure 2: The importance of time

The analysts said: "Give us two more weeks to investigate further to explain better why we should invest this €200k." The product manager complained that he felt that the project should start and that no more time should be wasted on additional analysis.

I asked him: "How much do these people cost per hour?" Like most people in and around projects, he didn't know. So I suggested: "How about €500 per day?" He nodded. "That means they spent some €20k (2 people, 20 days, €500) and they want to spend some €10k more to investigate more?" He said: "Yes, but I don't want them to waste more time!" Then I asked: "What will be the benefit once the project is finished?" The answer was: "The expected benefit will be €16M per year!" (Telecom must be like gold, but of course these companies don't tell us). "Aha! So the extra 2 weeks don't cost €10k but rather some half million euro, and the investment to save 4 months doesn't cost €200k, but would rather secure some €5M extra benefit!" The product manager ran to the business people: "Invest the €200k and start the project now!"

When I start coaching on a project, one of the first things I ask is:

- "What is the cost of one day of (unnecessary) delay?" People usually don't know.

- "What is the cost of one day of the project?" People usually don't know.

- "What is your cost per day?" Most people don't even know that.

(Note: what you cost is not what you get!).

Now, how can people make design decisions if they don't know the cost of time? The exact figure isn't even important. Any reasonable figure will do to make better decisions.

Once I was in a project asking these questions and I just saw, through the small window of the door, the boss passing by. I opened the door and said to the boss: "Boss, these people don't know their cost, the cost of the project or the cost of one day of delay. How can these people make design decisions?"

He said: "I don't know their cost, but I'll find out!"

An hour later he returned, saying: "€400 per person per day"

The benefit of the project should be huge, otherwise we should do another project, but if you don't know the benefit I usually suggest to assume that the benefit is about 10 times the cost of the project.

Using this figure we calculated the cost of one hour of delay: Number of people x €400 x 10 per day. This is usually a lot more than anybody imagined and it is a good basis to start making much better-founded design decisions.

---

3 Seven options we (seem to) have to be on time

As a lot of projects deliver late, there could be some interest to know how one can deliver earlier. There are several ways we see people use to try to finish a project earlier, most of which are intuitively right, but don't work. This contradiction causes people to think that we have to accept late projects as a fact of life. After all, they did their best (most people do!), even took measures (correct measures according to their intuition), and it didn't work out.

Deming said: "Doing your best is not enough." First you should know what to do, have an approach on how to do it best, and a mechanism to constantly improve that approach, and then you do your best. There are, of course, also options that do work.

Deceptive options

Let's first do away with the 4 deceptive options. Deceptive options we see applied in almost every project, but they don't work, on the contrary, they make things worse. It's surprising that people don't learn, and they keep applying them.

1. Hoping for the best – fatalistic

If your past project took longer than expected, what makes you think that this time it will be different? If you don't change something in the way you run the project, the outcome won't be different, let alone better. Just hoping that your project will be on time this time won't help. We call this ostriching: putting your head into the sand waiting until Murphy strikes again (see 'Murphy's Law for Engineers" www.malotaux.nl/?id=murphy).

2. Going for it – macho

We don't have enough time, but it has to be done: "Let's go for it!" If nothing goes wrong (as if that ever is the case) and if we work a bit harder (as if we don't already work hard)... Well, forget it. If the time really is insufficient, it won't happen.

3. Working overtime - fooling yourself, your customer and your boss

40 hours of work per week is already quite exhausting. If you put in more hours, you'll get more tired, making more mistakes, having to spend extra time to find and "fix" the mistakes, half of which you won't. You think you are working hard, but you aren't working smart, and the result is less at lower quality. As a rule, never working overtime, so that you have the energy to do it once or twice a year, when it's really necessary.

4. Adding time - moving the deadline

Moving the deadline further away is also not a good idea: the further the deadline, the more danger of relaxing the pace of the project. This is due to Parkinson's Law (work expands so as to fill the time available for its completion or People use the time given. Parkinson observed: "Granted that work (and especially paperwork) is elastic in its demands on time, it is manifest that there need be little or no relationship between the work to be done and the size of the staff to which it may be assigned." Note that a lot of the work done in projects is 'paperwork') and the Student Syndrome (Starting as late as possible, only when the pressure of the Fatal Date is really felt (attributed to E. Goldratt). When you had to study for passing an exam, did you work hard three weeks before, or mostly the night before? Be honest). At the new deadline we probably hardly have done more, getting the project delivered even later. Not a good idea, unless we really are in the nine mothers area (Some people think that if you take nine mothers, you ca make a baby in one month), where nobody can do it, even with all the optimization techniques available. It's better to optimize what we can do in the available time before the deadline. The earlier the deadline, the longer our future afterwards, in which we can decide what the next best thing there is to do. The only way a deadline may move is towards us.

About halfway through a student project I asked the students whether they would conclude the project successfully, with a good working application. I got the four deceptive options within one minute: "I've a good feeling about it!" said one of the students (option 1: hope). "Besides, we have to be successful, so we'll make it happen!" (option 2: macho). "I'm working through nights and weekends to make it happen" (option 3: working

overtime). "Sorry, what we promised last week is not yet working properly, we'll deliver some of it next week" (option 4: moving deadlines). Result in the end: useless application, customer not happy = failure!

5. Adding people - a risk filled option...

A typical move is to throw more people at the project, in order to get things done in less time. Intuitively, we feel that we can trade time with people and finish a 12 person-month project in 6 months with 2 people, in 3 months with 4 people, or in 2 month with 6 people, as shown in Figure 3. In his essay The Mythical Man-Month, Brooks shows that this is a fallacy, a fairy tale, a myth, defining Brooks' Law (1975): Adding people to a late project makes it later
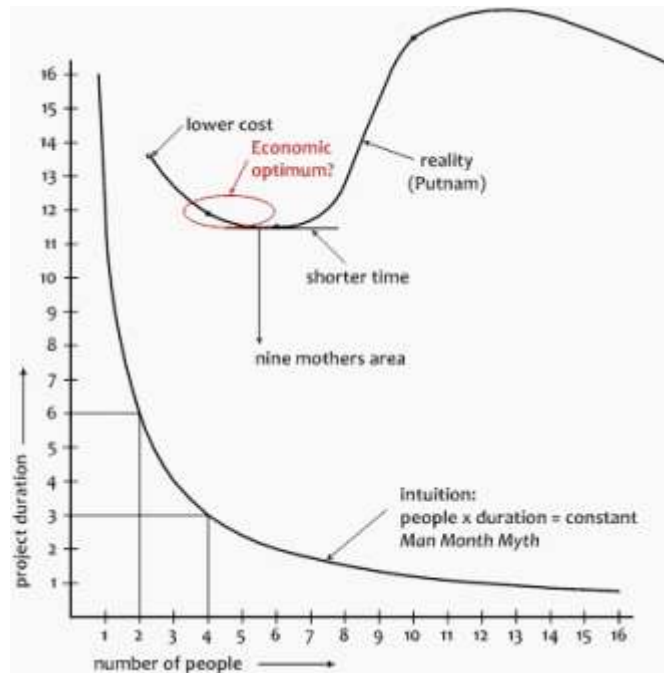

Figure 3: The Myth of the Man-Month and Putnam's results

When I first heard about Brooks' Law, I assumed that he meant that we shouldn't add people at the end of a project, when time is running out. Many projects seem to only find out that they are late by the end of the project. This is explained by Leon Festinger as the theory of cognitive dissonance (See http://en.wikipedia.org /wiki/Cognitive_dissonance). If time is running out, the added people have to learn about the project and we have to tell them, spending valuable time unproductively.

The effect is, however, much trickier: if in the first several weeks of a project we find that the speed is slower than expected, and thus have to assume that the project will be late, even then adding people can make the project later. The reason is a combination of effects. More people means more lines of communication and more people to manage, while the project manager and the systems engineer can oversee only a limited number of people before becoming a bottleneck themselves. Therefore, adding people is not automatically a solution that works. It can be very risky.

In a recent project the number of people was increased from 5 people to 20 people to ramp up productivity. The measured productivity increased by only 50%. It took project management some 10 months to decide to decrease (against their intuition!) the number of people from 20 to 10. When they finally did, the net productivity of the remaining 10 people was the same as with 20 people. They have been paying 10 extra people for 10 months with no contribution to the productivity! This was an expensive exercise, but it will

probably happen again many times at many places.

Putnam confirms Brooks' Law with measurements of some 500 projects (Figure 3). He found that if the project is done by 2 or 3 people, the project-cost is minimized, while 5 to 7 people achieve the shortest project duration at premium cost, because the project is only 20% shorter with double the amount of people. Because Time to Market is of often of huge economic value, this is probably an economic optimum.

Adding even more people makes the project take longer at excessive cost. Apparently, the project duration cannot arbitrarily be shortened, because there is a critical path of things that cannot be parallelized. We call the time in which nobody can finish the project the nine mothers area, which is the area where nine mothers produce a baby in one month. How can mega-projects where 100's or even 1000's of people work together be successful? Well, in many cases they are not. They deliver less and later than expected and many projects simply fail. There are only few companies left in the world who can design airplanes, at huge cost of time and money and usually with huge delays. If you think Brooks' Law won't bite you, you better beware: it will! The only way to try to circumvent Brooks' Law is to work with many small teams, who can work in parallel, and who synchronize their results from time to time. Every small team should be adequately managed; otherwise the overall management will be the bottleneck.

6. Saving time - the measure that always works - the low hanging fruit

Fortunately, there are ways to save time, without negatively affecting the result of the project. On the contrary, the result of the project is improved at the same time. These techniques are collected and routinely used in the Evolutionary Project Planning approach as presented here, in order to achieve the best solution in the shortest possible time.

There are many dimensions of saving time:

- **Improving the efficiency in what (why, for whom) we do**

Not doing things that later will prove to be not needed. Because people tend to do more than necessary, especially if the goals aren't clear; there is ample opportunity for not doing what is not needed. We use the business case, stakeholder management, and continuous requirements management to control this process. Every week we decide what we are going to do and what we are not going to do, before we do it. This saves time.

- **Improving the efficiency in how we do it: doing things differently**

This works in several dimensions:

  - The product - Choosing the proper and most efficient solution. The solution chosen determines both the performance and cost of the product, as well as the time and cost of the project and should be an optimum compromise and not just the first solution that comes to mind. We use short feedback cycles to check the requirements and assumptions with the appropriate stakeholders.
  - The project - We can probably do the same in less time if we don't immediately do it the way we always did, but first think of an alternative and more efficient way. We do not only design the product, we also continuously design and redesign the project.
  - Continuous improvement and prevention - Actively and constantly learning how to do things better and how to overcome bad tendencies. We use rapid and frequent Plan-Do-Check-Act (PDCA or Deming) cycles to actively improve the product, the project and the processes. We use Early Reviews to recognize and tackle tendencies before they pollute our work products any further and we use a Zero-Defect attitude because that is the only way ever to approach Zero Defects (See "Zero Defects", www.malotaux.nl/?id=zerodefects)

- **Improving the efficiency of when we do it**

Doing things at the right time, in the right order. A lot of time is wasted by synchronization problems like waiting for each other, or redoing things that were done in the wrong order. Actively synchronizing and designing the

order of what we do saves time.

All of these elements are huge time savers. Of course, we can apply these time savers even if what we think we have to do easily fits the available time, in order to produce results even faster. We may need the time saved later to cope with an unexpected drawback, in order still to be on time and not needing any excuse. Optimizing only at the end won't bring back the time we lost at the beginning. Optimizing only towards the end also means that there isn't much time to optimize anymore.

Imagine a plasterer plastering a wall in a new building. Then the electrician comes in, carving a furrow through the plaster in the wall to fit some electric wiring. The plasterer returns to repair the wall. Then the plumber comes in, cutting through the plaster and the electric wiring, in order to fit some water tubing. The electrician and the plasterer come back to repair the wiring and the plaster. If only these people made a TimeLine of their plans before they started, they would easily have seen in which order they should have done things in order not to repeat any of their work. They're not stupid. They only didn't think.

Furrows of the electrician after the paiter did his job

7. Killing the project

Of course, the faster we see that we never will get a positive return on investment, the faster we can kill the project, rather than after we've spent 3 times the budget.

4 The goal

As the universal goal of any project we use: Delivering the right result at the right time, wasting as little time as possible (= efficiently).

Or, to keep it even shorter: Quality on Time.

More formally we use as the top-level requirement of any project:

Providing the customer with

- what he needs

This is usually different from what he asks for.

- at the time he needs it

This could be earlier or later than he asks for.

- to be satisfied

Then he wants to pay.

- and to be more successful than he was without it

If he's not successful, he cannot pay; if he's not more successful than before, why would he pay? Note that the success is ultimately created by the users of the system. Our project and the customer merely provide the conditions for the users to create the success;

Constrained by

- what the customer can afford

If we start developing what the customer wants, we may fail from the start

- what we mutually beneficially and satisfactorily can deliver

It should be win - win: customer is king, but we don't have to be slaves

- in a reasonable period of time

Miracles take a bit longer.

Better focus on what we are supposed to do saves time. Anything we do in the project should support this top-level requirement, otherwise it's waste. Who wants to waste time producing waste?

5 Preflection, foresight and prevention

Of course your projects are different, but a lot of projects fail to deliver the right result at the right time. Cobb's paradox says: "We know why projects fail; we know how to prevent their failure – so why do they still fail?" Apparently, a lot of people do not know how to prevent their failure.

A lot of the things we have to do to deliver the right things at the right time are counter-intuitive. Intuition makes us react automatically on common situations. If intuition would be perfect, what we do would be perfect. Not everything we do is perfect, so apparently intuition sometimes guides us into the wrong direction. Intuition is a very strong mechanism and it's very difficult to go against it. However, once we recognize (and acknowledge!) that intuition sometimes causes us to make counterproductive decisions and inhibits proactivity, we can decide to actively do something about it.

Albert Einstein (1879-1955) seems to have said, although Benjamin Franklin (1706-1790) was earlier:

Insanity is doing the same things over and over again and hoping the outcome to be different (let alone better - my addition)

If we keep working the way we always did, we will keep making the same mistakes and our projects will still be late. Only if we change our way of working, the result may be different.

Because "hindsight is easy", we can often use it to reflect on what we did, in order to learn: Could we have avoided doing something that now, in hindsight, proves to be wrong, unnecessary or superfluous or could we've done it more efficiently? Reflection, however, doesn't recover lost time: the time is already spent and can never be regained.

Foresight is less easy, but with foresight we can imagine whether what we're going to do later will prove to be incorrect or superfluous, and then decide not to do it. We also can imagine different ways to do it to see that it is done more efficiently than if we just would start and do our best.

Reflection feeds hindsight which feeds learning, however, it's useless if we don't actively use what we learned for Preflection. Only with preflection we can foresee and thus prevent wasting precious time. Isn't it strange that we have a word 'reflection' but not a word 'preflection' which is much more important?

This is used in the Plan-Do-Check-Act or Deming cycle and it's one of the basics of Evolutionary Planning.

6 Plan-Do-Check-Act

The basis of the Evolutionary Planning technique is the good old Plan-Do-Check-Act cycle (PDCA or Deming cycle) for continuous learning and improvement.

- Do is not a problem: we "do" all the time

- Plan we do more or less, usually less

- For Check and Act, however, we have no time because we are so busy and we want to go to the next Do.

Figure 4: Intuitive cycle: PI-Do-PI-Do

Intuition is how people automatically react to situations, based on previous experience. Our sub-consciousness provides the suggestion how to handle the situation and we do it immediately as we did it before. We call this the Intuitive Cycle or 'PI-Do-PI-Do', as shown in Figure 4. It's not a conscious Plan, therefore we call it just PI.

Instead of following the PI-Do cycle, let's see how we actively can use the Plan-Do-Check-Act cycle better, as shown in Figure 5.

- First we Plan, which is twofold:

  - What do we want to achieve ('the product')?
  - How do we think we can achieve it the best way ('the project')?

- Then we **Do** according to the Plan

Figure 5: PDCA or Deming cycle

This is the first pitfall, because it means that the Plan must be Do-able and we must follow the Plan. Let's

assume that this was the case:

- Then we can go to the **Check** or **Study** phase where we analyze:

    - Is what we achieved according to the Plan? – thinking of effectiveness
    - Is the way we achieved it according to the Plan? - thinking of efficiency
    - If yes: should and can we do it better the next time?
    - If no: should and can we do it better the next time?

- Then, in the **Act** phase we decide what we will do differently the next time.

If we don't do anything differently the next time, the result will be the same. We do very small steps at the time, so that we can learn what is clearly better and what is not. Doing these small steps very quickly and frequently, we improve quickly and constantly what we are doing (the product), how we are doing it (the project) and even how we organize all that (the process). Because in the Act phase we introduce mutations of what we do and how we do it, we call this the Evolutionary approach and because evolutionary is such a long word, we use the label Evo, for everything that works better than the alternatives we know.

People often quite well know what's not going so well, and easily say: "Yes, we should do that, but ..." (management doesn't allow it, it can't be done, we've tried that before, etc). If I hear these "Yes, but ..."s, I usually suggest: "You're stuck in the Check phase. If that really is a problem, what should and could we do about it?" The other person always immediately has a suggestion what we could do about it. Then we can think what we should do about it. The problems we face are not the real problem. The main problem is that we have to move to do something about it. Not staying in the Check phase, but rather moving to the Act phase: What are we going to do about it. Not complaining what cannot be done, but rather thinking what can be done. Then you'll see that much more is possible than you thought.

7 Evaluations for fast feedback

Many organizations mandate a Project Evaluation at the end of every project, or at the end of every project stage. Some call it Retrospective or Post Mortem Analysis. Even so, few projects do the actual evaluation because they feel that these evaluations do not really contribute to better results. Why is this?

Consider a one-year project (Figure 6). People have to evaluate what went wrong and what went right (many things accidentally go right) and why, as long as a year ago. If something happened three months into the project, we have to remember it nine months later and think how we could do things better. Three months into the next project, we have to remember that we were going to do something differently, hopefully better. Result: the previous project didn't benefit, because we thought about doing things better only after the project, and the next project doesn't benefit either, because we don't remember our evaluation decisions when we need them, as they aren't ingrained into our intuition yet.
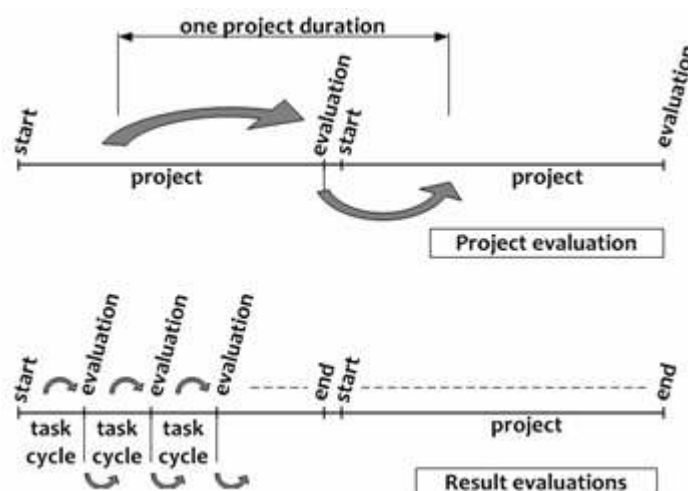

Figure 6: Project and Result evaluations

The principle is right: as with PDCA first we Plan the project, then we Do things, then we Check what we could have done better and in the Act-phase we decide what to do differently the next time. But in practice it doesn't really help us. The time-frame isn't tuned to our human capabilities. How can we tune this process, which is correct in principle, to actually work for us?

How about doing an evaluation every week? It is already difficult to remember what happened in the past week. Do you remember what you had for dinner 4 days ago? Most people don't. Well, you don't really have to know, unless you're the cook.

If we evaluate every week, now PDCA starts to work for us:

- If we tune our mind to remembering what went wrong and what went right the past week, it's not too difficult to remember it for evaluation by the end of the week. Some people use a whiteboard or a post-it to catch the issue when it happens, so that they don't forget to evaluate it later. Some people evaluate immediately, not to keep doing something that can improved immediately. After some time it becomes a way of life.

- Not too much happens in one week, so the evaluation doesn't have to take long (unless it has to).

- Because most kinds of work take more than one week, we can try out our idea to improve immediately the next week. One week later we can check: was the result better? If yes, should and can we do it better? If no, should and can we do it better?

- Because we actually do it differently the next week, the new way of working becomes internalized immediately, so that our intuition will hand it to us when we need it again later.

Now the evaluation is tuned to our human capabilities and needs, and it starts to work for us.

8 Evolutionary planning

Evolutionary (Evo) Planning is designed to continuously improve our efficiency and effectiveness, not wasting time on things that nobody will need and making sure that we will finish our endeavors ever more successfully in the shortest possible time.

In order to keep things manageable, we organize our work at several levels:

- weekly TaskCycles, to optimize the way we work, improving our efficiency;

- bi-weekly DeliveryCycles, to optimize what we deliver, improving our effectiveness;

- TimeLine, to see:

    - what will happen if we work the way we are currently working;
    - what to do if that won't produce the expected results in time, making sure we'll be on time, no excuses needed.

These elements together make our individual work more effective, efficient and predictable, which subsequently can make our projects more effective, efficient and predictable.
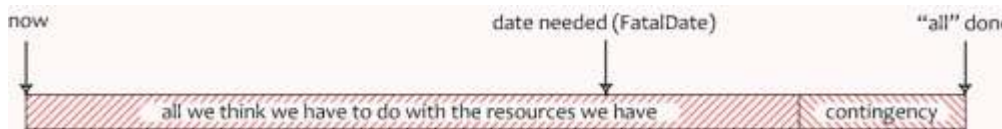
8.1 Task Cycles

In the TaskCycle we organize the work. We are checking whether we are doing the right things, in the right order, to the right level of detail. We are optimizing our estimation, planning and tracking abilities to better predict the future. We select only the highest priority tasks, never do lower priority tasks and never do undefined tasks. This improves our efficiency.

As a practical rule, we plan 2/3 of the available time and in the remaining 1/3 of the time we handle small interrupts like helping each other, project meetings and many other things we also have to do in the project. If we plan 100% of our available time, we will still do all those other things, hence we will never succeed in what we

planned (sounds familiar?).

TaskCycles normally take one week, in some special cases even less. Every cycle we decide how much time we will have available, what is most important to do, how much time it takes to do it completely (we define what completely done means) and then what we can do in the available time. We also decide what we will not do in this cycle, because there is no time to do it. Now we can focus all our energy on what we can do, making us more relaxed and more productive. Some managers fear that planning only 2/3 of the available time makes people do too little. In practice we see people do more.

8.2 Delivery Cycles

Figure 7: Standard approach: it takes what it takes, but often that's too late

In the DeliveryCycle we organize results to be delivered to selected stakeholders. We are checking whether we are delivering the right things, in the right order, to the right level of detail. We are optimizing the requirements and checking our assumptions. This improves our effectiveness.

A DeliveryCycle normally takes not more than two weeks. Novice Evo practitioners, almost without exception, have trouble with the short DeliveryCycle. They think it cannot be done. In practice we see that, without exception, it always can be done. It just takes some practice. One of the important reasons for the short length of the cycle is that we want to check our (and their) assumptions before we have done a lot of work that later may prove unnecessary, wasting valuable time. Short DeliveryCycles help us do this to minimize risk and cost.

A common misconception of deliveries is that we always have to deliver to users or customers. On the contrary, we can deliver to any Stakeholder: the user or customer, ourselves or any Stakeholder in between. This makes it easier to define deliveries. However, we must always optimize deliveries for optimum feedback: we must check what we are doing right and what we are still doing wrong. Hence, for every delivery we ask the question: "What should we deliver to whom and why?

8.3 Time Line

In many projects all the work we think we have to do is cut into pieces (some call this work packages), the pieces are estimated, and the estimates are added up to arrive at an estimate of the effort to do the work. Then this is divided over the available people to arrive at an estimate of the duration of the work, which, after evaluation of interdependencies, and adding some contingency, is presented as the duration of the project (Figure 7).

A problem is that in many cases the required delivery date is earlier. The tension between expected and estimated delivery time causes extra discussion time, however, the required delivery date doesn't change, leaving even less time for the project.

Because the delivery date is a requirement just as all the other requirements, it has to be treated as seriously as all other requirements. With TimeLine, we treat delivery dates seriously and we meet these dates, or we very quickly explain, based on facts, why the delivery date really is utterly impossible to meet, because it's in the nine mothers area. We don't wait until the FatalDate to tell that we didn't make it, because if it's really impossible we know it much earlier. If it is possible, we deliver, using all the time-saving techniques we have at our disposal to optimize what we can deliver when.

TimeLine can be used on any scale: on a program, a project, a sub-project, on deliveries, and even on tasks. The technique is always the same. We estimate what we think (We keep saying "what we think we have to do", because however good the requirements are, they will change, because we learn, they learn and the circumstances change. The longer the project, the more the requirements have a chance to change. However, what we don't know yet, we cannot plan for. So we keep improving our knowledge about the real requirements,

but we only can plan based on what we currently think we have to do.) we have to do, see that we need more time than we have and then discuss the TimeLine with our customer or other appropriate stakeholders and explain (Figure 8):

- What, at the FatalDate, surely will be done?

- What surely will not be done?

- What may be done (after all, estimation is not an exact science)?
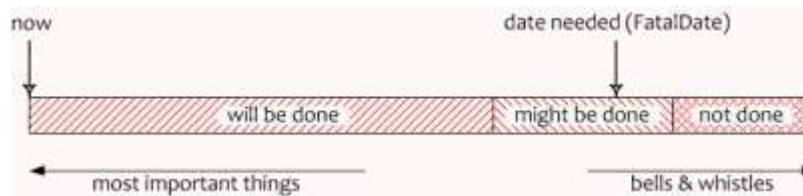
Figure 8: Basic TimeLine: what will surely be done, what will not be done, and what may be done if we would keep working the way we're working now

If what surely will be done is not sufficient for success, we had better stop now to avoid wasting time and money. Note that we put what we plan in strict order of priority, so that at the FatalDate at least we'll have done the most important things. If Time to Market is important, customers don't mind about the bells and whistles. Because priorities may change quite dynamically, we have to constantly reconsider the order of what we do when.

8.4 Setting a Horizon

If the total project takes more than 10 weeks, we define a Horizon at about 10 weeks on the TimeLine, because we cannot really oversee longer periods of time. A period of 10 weeks proves to be a good compromise between what we can oversee, while still being long enough to allow for optimizing the order in which we deliver results for optimum feedback. We don't forget what's beyond the horizon, but for now, we concentrate on the coming 10 weeks.

8.5 Delivery Cycles

Within these 10 weeks, we plan DeliveryCycles (Figure 9) of maximum 2 weeks, asking: "What are we going to deliver to whom and why?"
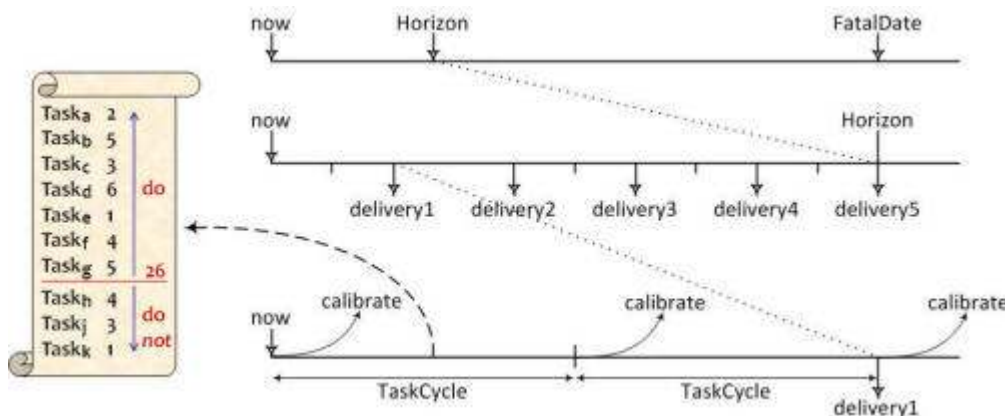
Figure 9: TimeLine: we analyze top-down what should happen, calibrate bottom-up to see what will really happen, then use our disappointment of what we see to do something about it

Deliveries are for getting feedback from stakeholders. We are humble enough to admit that our (and their)

perception of the requirements is not perfect and that a lot of our assumptions may be incorrect. Therefore, we need communication and feedback. We deliver to eagerly waiting stakeholders, otherwise we don't get feedback soon enough. If the appropriate stakeholders aren't eagerly waiting, either they're not interested and we may better work for other stakeholders, or, if we really need their feedback now, we have to make them eagerly waiting by delivering what we call Juicy Bits. How can Juicy Bits have a high priority? If we don't get appropriate feedback, we will probably be working based on incorrect assumptions, causing us to doing things wrong, which will cause delay later. Therefore, if we need to deliver Juicy Bits to stakeholders to make them eagerly waiting in order to get the feedback that we awfully need, this may have a high priority.

8.6 TaskCycles

Once we have divided the work over Deliveries, which are Horizons as well, we now concentrate on the first few Deliveries and define the actual work that has to be done to produce these Deliveries. We organize this work in TaskCycles of one week.

In a TaskCycle we define Tasks, estimated in net effort-hours. We plan the work in plannable effort time, which defaults to 2/3 of the available time (~26 hrs in case of a 40 hr week), confining all unplannable project activities like email, phone-calls, planning, small interrupts, etc, to the remainder of the time. We put this work in optimum order, divide it over the people in the project, have these people estimate the time they would need to do the work, see that they don't get overloaded and that they synchronize their work to minimize the total duration.

8.7 Top down – bottom up

Having estimated the work that has to be done in the first week, we have captured the first metrics for calibrating our estimates on the TimeLine. If the Tasks for the first week would deliver about half of what we need to do in that week, we now can extrapolate that our project is going to take twice as long, if we keep working the way we are, that is: if we don't do something about it. Initially the data of the first week's estimate may seem weak evidence, but it's already an indication that our estimates may be too optimistic. Putting our head in the sand for this evidence is dangerous: I've heard all the excuses about "one-time causes". Later there were always other "one-time causes".

One week later, when we have the actual results of the first week, we have slightly better numbers to extrapolate and scale how long our project really will take. Week after week we will gather more information, continuing top-down and bottom-up, with which we calibrate and adjust our notion of what will be done at the FatalDate or what will be done at any earlier date. This way, the TimeLine process provides us with very early warnings about the risks of being late. The earlier we get these warnings, the more time we have to do something about it. In practice the actual TimeLine process may be a bit more complicated, but the basics are as described.

Failure is not an option. The earlier we get warning signals of possible failure, the earlier we can start making sure that failure is not going to happen.

9 Evo practice example: TaskCycle planning

Not to make this article longer than it already is, I'll provide here as an example some more detail about the first thing we do when turning a project towards Evolutionary Planning.

Actually, the result of the TaskCycle planning is a to-do list for everyone in the project. Not the to-do list most of us already make from time to time. This to-do list is checked before we do the work, on feasibility (will it fit the available time?), priority (is this more important than anything else?), synchronization (does it fit with the order we should do things and with the work of others), and many other aspects. Before we do the work, we can still optimize what we are going to do and what not to do.

Afterwards we can only complain that we wasted time because we didn't check beforehand. Because what we plan is doable, people say that from the first week the stress is gone. They work much more focused, know why they are doing just this and why they are not doing other things. Productivity surges.

At the start of the weekly TaskCycle, this is what we do:

1. First determine the gross number of hours you have available for this project this TaskCycle

We define how much time we have available before we think what we have to do. If what we think we have to do takes more time than we have available, we tend to 'give' ourselves more time than we have, fooling ourselves, leading to failing to deliver as planned. The rule is that what we plan will be done, completely done, no excuses needed. Deduct all non-project time, or otherwise already planned time from your total available time.

2. Divide this gross number of available hours into:

- Available Plannable Hours (default ~2/3 of gross available hours)

- Available Unplannable Hours (default ~1/3 of gross available hours)

In a 40 hour work week, we usually use 26 hours plannable time and 14 hours unplannable time. In many projects this proves to be realistic.

We only plan those Tasks that don't get done unless planned. If it's in your plan, you have time, and after that time, the Task will be done, completely done. During planning you define what 'completely done' means and base the required time on that.

We do not plan Tasks that will get done anyway, even without planning. If they happen anyway, why waste time putting them in your planning? These are done in the unplannable time, or the time has already been deducted from the gross available time.

3. Define Tasks for this cycle

Focus on finding Tasks that are most important now and don't waste time on less urgent tasks for the moment. Based on what we learn from current tasks, the definition of later Tasks could change, so don't plan too far ahead. Use the Delivery definition to focus on what to work on in the Tasks.

4. Estimate the number of effort hours needed to completely accomplish each Task

We always estimate net effort hours. If people estimate in days, they usually come up with lead time (the time between starting and finishing the Task). If people estimate in hours, they can quickly learn to come up with effort (the net time needed to complete the Task). The reason for keeping effort and lead time separate is that the causes of variation are different: If effort is incorrectly estimated, it's a complexity assessment issue. If there is less time than planned, it's a time-management issue. Keeping these separate enables us to learn.

Only the person who is going to do the Task is allowed to define the duration of the Task. Others may not even hint, because this influences the estimator psychologically. If others do not agree with the estimate, they should challenge the (perceived) contents of the Task, never the estimated time itself. Ultimately, when we agree on the requirements of the Task, the implementer decides how much time he is going to need, otherwise there will be no commitment to succeed. If there is no commitment, there will be no pain if we still happen to fail. If there's no pain, we don't learn.

5. Split Tasks of more than about 6 hours into smaller Tasks

We split the work into manageable portions. Estimation is not an exact science, so there will be some variation in the estimates. We are not bound by the exact estimated effort hours. We are only bound by the Result: by the end of the week all committed work is done. If one task takes a bit more and the other a bit less, who cares? If you have several tasks to do and the average estimated time is sufficiently accurate, the variations will cancel out. If you have a massive task of 26 hours, it is more difficult to estimate and the averaging trick cannot save you anymore. Besides, 6 hours is about the maximum you can do in one day (2/3 of available time!).

6. Fill the available plannable hours with the most important Tasks

Never select less important Tasks. Always fill the available plannable hours completely, otherwise the unfilled

time will evaporate.

7. Ascertain that indeed these are the most important Tasks to do and that you are confident that the work can and will be done in the estimated time

- Any doubt undermines your commitment, so make sure you can deliver

- Acknowledge that by accepting the list of tasks for this cycle means accepting the responsibility towards yourself and your team. These tasks will be done, completely done, by the end of the cycle: no excuses needed.

At this point, you will have a list of Tasks that will get done (Figure 10). If you cannot accept the consequence that some other Tasks will not be done, do something! You could:

- reconsider the priorities.

- get additional help to do some of the Tasks for you. Beware, however, that it may cost some time to transfer the Task to somebody else. If you don't plan this time, you won't have time.

- If no alternative is possible, accept reality. Hoping that the impossible will happen will only postpone the inevitable. The later you choose to do something about it, the less time you have left to do it. Don't be an ostrich: in Evo we take our head out of the sand and actively confront the challenges.
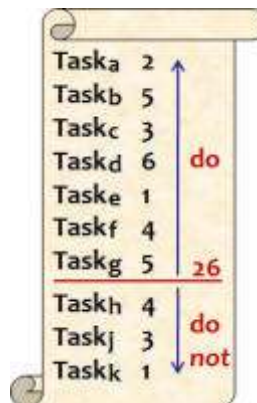

Figure 10: Task list

9.1 At the end of the TaskCycle we Check, Act and Plan:

1. Was all planned work really done? If a Task was not completed, we have to learn:

- Was the time spent on the Task, but the work not done?

- This is an effort estimation problem. Discuss the cause and learn from it to estimate better the next time. What did you first think and what do you know now.

- Was the time not spent on the Task and therefore the work wasn't done?

- This is a time management problem:

- Too much distraction or interrupts

- Too much time spent on other (poorly-estimated?) Tasks

- Too much time spent on unplanned Tasks.

Discuss the causes and decide how to change your time management habits.

Not having all Tasks completed is expected the first few cycles, but the learning should help us within a few weeks to complete all TaskCycles as planned.

2. Conclude unfinished Tasks after having dealt with the consequences

Feed the disappointment of "failure" into your intuition mechanism for next time. This is why commitment is so important: only with commitment we can really feel disappointment. We must use the right psychology to feed our intuition properly to come up with better estimates. Not for estimation's sake, but because it improves our predictability

Define new Tasks to finish what you didn't do. Estimate these Tasks, and put them on the Candidate Task List. They will surface in due time. If they do not surface immediately for the next cycle because there are more important things to do, we apparently stopped at the right time

Declare the Task finished after having taken the consequences: remember that you cannot work on this Task any more, as it is impossible to do anything in the past.

3. Now continue with planning the Tasks for the next cycle

Don't forget to apply what you learnt while planning the next TaskCycle!

9.2 Weekly 3-step process

Most projects have weekly team meetings. We found several shortcomings in these meetings and based on the experience gained we eventually arrived at a weekly 3-step process, which proves instrumental for the success of Evo planning. In this process we minimize and optimize the time used for organizing the Evo planning. Still, communication between all project people is greatly enhanced.

The steps are:

1. Individual preparation

In this step the individual team members do what they can do alone:

- Conclude current tasks

- Determine the most important Tasks for the next week

- Estimate the time needed for these Tasks

- Determine how much time is available for the project the coming week

The project manager and/or systems engineer also prepare(s) for their team what they think are the most important Tasks, what time they think these Tasks may take (based on their own perception of the contents of each Task and the capabilities of the individual) and how much time they need from every person in the team.

2. 1-on-1's: Modulation with and coaching by Project Management

In this step the individual team members meet individually (1-on-1) with Project Management (project manager and/or systems engineer). In this meeting we modulate on the results of the Individual preparations:

- We check the status and coach where people did not yet succeed in their intentions

- We compare what the individual and project management thought to be the most important Tasks. In case of differences, we discuss until we agree

- We check the feasibility of getting all these Tasks done, based on the estimates

- We iterate until we are satisfied with the set of Tasks for the next cycle, checking for real commitment. Now we have the work package for the coming cycle.

We use a data projector at every meeting, even at the 1-on-1's. Preferably we use a computer connected directly to the Intranet, so that we are using the actual files. This is to ensure that we all are looking at and talking about the same things. If people scribble on their own paper, they all scribble something different. The others don't see what you scribble and cannot correct you if you misunderstand something. Furthermore, when you are scribbling, your attention to what is said is less.

There is not just one scribe. People change place behind the keyboard depending on the subject or the document. If the project manager writes down the Task descriptions in the Task database (like the ETA (Eva Task Adminstrator tool, see www.malotaux.nl/?id=download#ETA) tool), people watch more or less and easily accept what the project manager writes. As soon as people write down their own Task descriptions, you can see how they tune the words, really thinking of what the words mean. This enhances the commitment a lot. The project manager and/or the systems engineer can watch and discuss if what is typed is not the same as what's in his/her mind. And when we are connected to the Intranet, the Task database is immediately up to date and people can immediately print their individual Task lists.

3. Team meeting: Synchronization with the group

In this step, after all the 1-on-1's are concluded, we meet with the whole team. In this meeting we do those things we really need all the people for:

- While the Tasks are listed on the projection screen, people read aloud their planned Tasks for the week. This leads to what we call the synergy effect: People say: "If you are going to do that, we must discuss …", or "You can't do that, because …" when we apparently overlooked something. Now we can discuss what to do about it and change the plans accordingly. The gain is that we don't together generate the plans, we only have to modulate. This saves time. We also see a huge increase in very efficient communication before we do things, helping each other not to do the wrong things, or to do things incorrectly or at the wrong time.

- If something came up at a 1-on-1 which is important for the group to know, it can be discussed now. In conventional team meetings we regularly see that we discuss a lot over the first subject that pops up, leaving no time for the more important subject that may come up later. In the Evo team meetings we select which subject is most important to discuss together.

- To learn and to socialize.

Project managers and systems engineers invariably say that these 1-on-1's are one of the most powerful elements of the Evo Planning approach. In a very efficient way they get a much more clear insight into the real status of the work of all individuals than they ever had before.

After a few weeks the whole planning process takes about 1 hour per person: 20 min preparation, 20 min 1-on-1 and 20 min team meeting. Do we discuss less than before? No, we just discuss the right things effectively and efficiently. Of course the project manager and the systems engineer need more time to see their whole team, but what they are doing here is what they should have been doing all along.

Recently I asked a systems engineer: "With all this planning and scheduling, do you still have enough time for your other SE work that is overseeing the systems development itself?" Simple answer: "Yes. No problem."

Some people feel that in order to improve the performance of projects we have to do a lot more than 'just' low-level planning as we do in the TaskCycle. Once we start applying the technique, however, we quickly recognize what the real issues are that plague this particular project at this particular time, so that we immediately can do something about them, rather than letting them simmer. This proves to be much more to the point than burdening the project with a lot of process that is more elaborate than what the project really needs, which would only decrease the efficiency rather than increasing it.

10 Conclusion and invitation

I have tried to paint some reasons and background, the principles, and some element of the Evolutionary Planning approach. This approach enhances communication, prevents issues before they develop, and improves the effectiveness and efficiency of our work leading to improved productivity as well as predictability.

This may leave you with a dilemma. Now you know that your project quite easily can become much more productive and predictive, you don't have an excuse anymore not to do it. Perhaps I should have warned you at the start of this text rather than at the end. ☺

The proof of the pudding is in the eating.

I invite people to do a five week exercise with me to get a real taste and to see how this works for you to become much more effective, efficient, and predictive. Because I've done this with many people before I can predict that the first few weeks you will have a hard time (seeing your current way of working in the mirror in ways you still cannot imagine now), but after five weeks you will have a much better understanding of a lot of details that really make this work. It also will give you a good feeling how this can work for your project and your organization.

This invitation may cause an avalanche of exercise requests, so I reserve the right to reject and select requests for the exercise or to put you on a waiting-list. In order to improve your chance to get some free sample coaching, explain why it would be interesting for you and for me to help you.

Start with the steps of section 9 and see what happens.

Additional Article

Extreme Review -
A Tale of Nakedness, Alsatians and Fagan Inspection

by Les Chambers BE (Hons)

Email: lchambers@ppi-int.com

Les Chambers presents worldwide PPI's popular five day Software Development Principles and Practices workshop.

Two naked babies amble across a busy freeway. Trotting after them is a guy in a suit. The guy is me. Looking back on that day it's clear that this was an omen, for within the hour I was to encounter extreme review and understand why, that for systems engineers, public nakedness is sometimes a good thing.

The Approach

Flying in from the North the scene from my starboard window was flat out stunning. The pilot dipped a wing on final approach, the blue sky rolled and the Harbour Bridge and the white sails of the Opera House rotated into view. Beautiful big brash Sydney.

Such a day. I had a right to feel good. I was carrying a million dollar proposal that, I was confident, would be accepted by a major client: IBM. My company had specialized knowledge of a small corner of the IBM software empire that guaranteed us zero competition.

It turns out that to represent the Kanji character set in IBM compiler products they needed an extra byte. Double byte enablement of IBM system software had become a cottage industry worth seven figure sums of money to small software houses. We were one.

Over the years we had developed a collaborative relationship with IBM. They were friends rather than masters. I was supremely confident, IBM was our puppy, on its back, paws in the air waiting for me to scratch its tummy.

We landed with a subtle bump and rolled to a gate. I deplaned and hurried up the air bridge into the terminal and through to the cab line.

Out of Gas

Into the cab and off, en route to IBM's headquarters at Cumberland Forest AKA KOALA Park, a 40 minute cab ride; time to focus on the day ahead and the honeyed words that might be necessary to explain away any bumps in the estimate. I don't talk to cab drivers on the way to a sales call. I'm usually too engaged in rehearsing the sales pitch. I seldom even know where I am or pay much attention to the scenery. So we had almost come to a stop when I noticed the absence of engine noise. I re-engaged with the moment just in time to hear the almost human sigh that an LPG powered vehicle gives out when it runs out of gas. The driver pulled over to the curb and there was silence.

Not a problem, there was plenty of time. We piled out of the cab and pushed it to the nearest service station only 200 meters away. Gassed up we were soon on our way with me back in my reverie.

Rescuing Naked Children

Two naked children walked onto the freeway in front of us. The driver made to change lanes and maneuver around them. Maybe he wasn't a parent. I yelled, "Stop. Stop now! Pull over!" I de-cabbed, trotted over to the kids who were about to step into the fast lane and said hello. They were no older than three. I took their hands and together we slid down the freeway embankment and across a service road towards a row of houses. They could not have walked far so I approached the nearest driveway in search of a parent. There was no sign of life but for an Alsatian who trotted out from under the house with a menacing growl and a flash of sharp teeth.

I froze in place and reviewed my risk profile. Not only was I on the dog's territory but also I probably had hold of the two youngest and most beloved members of its pack. This dog was definitely not up for a tummy scratch and we were never going to be friends. Meanwhile back in the taxi a million dollars worth of unconverted business was burning a hole in my briefcase and 30km further down the freeway was a room full of IBMers waiting expectantly. How could cruel fate seize such a day, dawned with such promise, to bury it deep in the bosom of so dark a comedy? Bill Gates had to buy a tie on the way to sell PC DOS to IBM. I have to save two naked babies and be savaged by an Alsatian? Wonderful!

A matter-of-fact voice issued from the house, "Merl the kids have got out again." The children let go of my hands and ran inside followed by the dog, his pack recovered, his fangs retracted and my work done, apparently.

The Space Shuttle Men

Back into the cab and on to Koala Park. I was half an hour late and after another half an hour of telling the story, of the sighing cab, the naked babies and the Alsatian to a fascinated array of blue pinstriped people I presented

our submission.

3 Million Lines of Code
With < 300 defects

At this point in history IBM was diversifying from pure sales of big iron to software services. IBM did write software but only of the serious kind, operating systems and on-board flight software for NASA's space shuttle. To launch their software services initiative they tasked what software development talent they had as evangelists to spread operational experience throughout the organization. So it turned out that my two reviewers were veterans of IBM's Federal Systems Division; they were space shuttle men.

It transpired that these men were not ordinary humans. They were fresh from a project that had delivered three million lines of code with less than three hundred errors. They were polite but earnest Americans with a disciplined presence about them, the kind that makes you want to sit up straight in your chair without being asked.

Their mission was to hold a torch to my belly, evaluate the quality of our submission and determine if my company could be trusted to modify IBM systems software products. Nice guys. The nicest grim reapers I've ever met.

The Extreme Review

The shuttle men had clearly read the submission thoroughly, each of them bearing a neatly documented defect list. They briefly explained the review process and mentioned a guy named Fagan1. It sounded ominous. I assumed they weren't alluding to the Dickensian character from Oliver Twist. I read the lists upside down. Each defect was classified with an acronym. There were annotations indicating severity levels and blank cells on the page for verification of corrective action. At the bottom there was a cell with the title "Defect Density" and a number.

They started out by asking me about the process we used to create the cost estimate. I didn't have a satisfactory answer because we didn't have a process. I danced around the subject until they moved on, glancing at each other.

They started at the beginning and methodically worked through each page of the document, each reviewer raising issues from his personal defect list. I had the sense of being zoomed in on with laser-like focus. Every pixel of every font examined. The inspector's gaze sliding over the curve of a P and dropping to examine the full stop. Is it necessary? Now he zooms out and takes in the flow of one word to another, one sentence to the next. Now he extracts, evaluates and judges my ideas in perfect rhythm with the odd asynchronous stabbing of the cognitive red pen on detecting an ambiguity, an inconsistency, an incorrect fact.

As the review progressed I could feel them breaking in. Boring through my outer shell to that inner softness that renders you vulnerability mortal; the wanting to be loved, the fear of criticism, the dread of rejection.

There was no love here but neither was there humiliation. Maybe just discipline born of a life lived with a horrible responsibility. The countdown to launch; the certain knowledge that screw-ups meant death for seven astronauts in front of a world audience with streaks of flame in the sky, with post mortems that run for years as the earth gives up its debris piece by blackened piece, not to mention the faces of the people, now members of the extended NASA family, partners and children at the graveside, and a life of eternal damnation for some simple mistake that might have been caught in a more effective review.

Their urgency was palpable. The consequences of failure had accelerated the evolution of their review process. The discipline was in their nature and not to be forsworn for less critical application domains. It didn't matter that this was a harmless double byte enablement project, this wasn't going to be a ticklish torch to my belly, nor was I going to be scratching anyone's tummy. This was main engine thrust.

They tore our submission apart with surgical precision, identifying elements of the quote that had been double dipped (an honest mistake on our part). I had a flashback to an epic fist fight with a class mate at boarding school. He was a trained boxer. It went on for 2 hours until he finally knocked me senseless. After a while I actually began to admire his work. His punches would appear from nowhere and slam into my face. Luckily the IBMers were more focused on my work. Their blows were precisely targeted, accurate and completely righteous. All the while they remained in good humor and, when it was over, delivered their exit decision in polite civility. "Due to the high defect density in your submission, you need to perform corrective action and resubmit for full review." We shook hands and bade farewell.

Exit

Back into a cab and out onto the freeway, we sailed past the baby house. They'd be having their dinner by now and then to bed to dream of trucks in the fast lane and the anguished face of the stranger on the freeway. Three faces had filled my day. The innocence of children, the primal pre-feeding gape of the Alsatian and the look of professionalism in peer review. In two hours I'd been dragged onto some higher, more evolved plane of systems engineering that I never knew existed. Now my company would have to come with me if we were to do business with IBM. I remember feeling no pain and wondering why.

Creativity and Pain

Reviews can be excruciating for the unevolved. In fact the entire writing process is redolent with pain. You struggle with ideas and search in vain for the words to express them. And in the back of your brain there plays a negative litany that will not go away. Elizabeth Gilbert summarized it beautifully in a TED talk:

Aren't you afraid that you're going to work your whole life at this craft and nothing's ever going to come of it and you're going to die on a scrap heap of broken dreams with your mouth filled with bitter ash of failure?

Finally words do appear. You push them around with a mouse endlessly cutting and pasting until thoughts congeal into ink and you begin to fall in love with your words on the page. They take on intrinsic beauty self evident to you and ergo to the remainder of humanity. Or so you think. Then heaven descends to earth, dreams are crushed by reality, beauty is no longer what it has been, even the gods become ordinary. You lay your work at the feet of your peers - and they savage it.

Going Naked

To submit your work to criticism is to reveal yourself. To go naked in public. Some authors liken it to inviting the world into bed with you.

For sure in polite society public nakedness is an unnatural act. Edouard Manet scandalized Paris with his painting Le Déjeuner Sur L'Herbe (The Luncheon on the Grass) depicting a naked woman picnicking with two men (see the banner above). The Paris Salon jury of 1863 called it an affront to the propriety of the time and rejected it out of hand.

In stark contrast IBM's twentieth century salon had demanded my nakedness and inspected every zit. It grabbed

me by the scruff of the neck, held my face to a mirror and made me look at myself as those, more evolved than I, saw me. I was reminded that systems engineering is not polite society. Sure, going naked is unnatural but so is every other aspect of a disciplined life. It's learned, evolved and can't be achieved without pain.

Dealing With Fear of Criticism

Afghanistan, June 11, 2010. An Australian SAS troop runs into a Taliban ambush. Ignoring withering fire from three elevated machine-gun emplacements Corporal Ben Roberts-Smith single-handedly silences two machine-guns and is awarded the Victoria Cross, Australia's highest military honor. When asked how he deals with fear he says, "Recognize it and understand it clinically. Generally, fear manifests itself as adrenaline so if you can recognize it you can control it. In my opinion, being able to control fear is what determines bravery."

The review room is not a battlefield. A high defect density in your work is probably not going to get you killed. But just like Ben, getting over the fear of criticism requires you to recognize your humanity and learn to deal with it. Accept that there is an element of vanity in everything you publish. You have an ego which blinds you to defects in your own work, you make unconscious assumptions that are not valid for all situations and you hold beliefs that become transparently illogical when voiced out loud to others.

Professionals recognize and deal with these natural pathologies by embracing peer review. Even Ernest Hemingway, an icon of American literature, invited friends over to help him remove superfluous words from his manuscripts. Always the honest professional he said that, "the first draft of anything is shit." So there you have it, if you aspire to professionalism, be honest, accept criticism and harden up – unless of course your being reviewed by bozos.

Recognizing Bozo Review

I like to put my inquisitors under pressure. Edgy reviewers are much more productive. I create positive tension by telling them the shuttle men story with the punch line, "For you, they're a hard act to follow, I hope you're up to it."

There is such a thing as bad criticism. Steve Jobs was a world champion at it. For example, he asked his marketing team to come up with names for a new Apple Computer. They responded with five options one of which was "iMac". His response was: "they suck". He began to warm to iMac but continued, "I don't hate it this week, but I still don't like it." Whenever his creatives asked him what he wanted, a common response was, "you've got to show me some stuff, and I'll know it when I see it." Worse he had a nasty habit of humiliating people in public, not only criticizing their work but also trashing their personalities, flinging indictments such as "I'm trying to save the company here and you're screwing it up" and using choice adjectives such as "stupid". At times it got so bad that wiser heads would take him aside, explain how hard everyone was working and suggest: "When you humiliate them, it's more debilitating than stimulating."

Jobs was a creative genius working in a supernatural world where futures were predicted and then delivered. He was so often right that his people endured his tantrums putting it down to creative passion. Most of us don't get cut that kind of slack. Most of us live in a world where defects are banal and more easily defined if we choose to be diligent. At the micro level it's an ambiguity, an incorrect fact an inconsistency or maybe just a missing full stop. At the macro level it's a non compliance with a pre-existing specification or it might be a violation of an agreed upon best practice.

In our normal world there is no excuse for bozo review. Bozo reviewers turn up at meetings without thoroughly reading the target document. Bozos make "this is crap" comments that denigrate the author, they gesture fecklessly at paragraphs with vague pronouncements of "I hate it", and worse, they scrawl lone red colored question marks in the margin of documents. What could this mean? Other classics from my recent past include, "rubbish", "it's tacky" and "you're trying to show off".

Suffice to say that any utterance from a reviewer that does not directly, and in the most efficient manner possible, contribute to the improvement of the work under review is bad criticism.

Reviewers! If you can't be explicit it won't get fixed, and as for character assassination; changing a person's personality can be a lifetime project, most people over the age of five are pretty much hard wired - better to

concentrate on what can be improved in the space of an hour and that is: the quality of the work.

The Reviewer as Teacher

I weep for organizations that don't review regularly. The insightful review is the most effective teaching tool we have. For a start, reviewees have their heart, soul and skin in the game. You've got their full attention. You're dealing with their baby, the piece of themselves they've nurtured into life for weeks and sometimes months. Precise, improvement oriented problem statements delivered in a non-threatening environment find an impedance match to the creative brain. Advice passes through with no component reflected back. Reflection on what went wrong followed by corrective action is the most effective learning process known to man.

Well at End

In the week following the review we fixed the defects and resubmitted our proposal. It was accepted and my company went on to develop a multimillion dollar line of business with IBM.

My day with the shuttle men happened 25 years ago yet I remember it as though it were yesterday. I remember it for the sighing broken down cab and the naked babies on the freeway but mostly for the relentless professionalism of those earnest Americans.

In a single afternoon they opened my eyes to a new way of operating. I was naked in public but felt no shame because they accepted me as I was and focused on my work with a bent to improvement. Talk about an accelerated education! Many things:

That to review is to judge and be judged, not only by what you give to your peers, but by what you do not take away; dignity and to accept critique is to stay young and open to the possibilities and beware the vanity of vanities lest they morph into conceit and perhaps then to arrogance leaving you all the while learning less and less to an end point of nothing and keep your cheek to the Earth, aware of your flawed humanity but comfortable in your own skin - naked or no, for as Baudelaire said: "The body's beauty is a sublime gift that extracts a pardon for every infamy "

Note 1:
Michael Fagan invented a formal process for finding defects in development documents such as programming code, specifications and designs. His seminal paper on this subject was published in a 1976 IBM Systems Journal. More than thirty years later his inspection process is still recognized as one of the most effective software quality assurance techniques. Fagan inspections can be applied to software development at all stages of the life cycle.

Refer:
Fagan, M. (1976) Design and code inspections to reduce errors in program development, IBM Systems Journal, Vol. 15, No. 3, pp. 182-211, [Online], Available: http://www.mfagan.com [19 May 2012]

Systems Engineering News

The BABOK® Guide Available in French 2012

The International Institute of Business Analysis (IIBA) has announced the release of the French translation of A Guide to the Business Analysis Body of Knowledge® (BABOK® Guide) at the Building Business Capability (BBC) 2012 Conference.

"The BABOK® Guide provides a standard for the concepts, tasks, techniques and skills that underlie the business analysis discipline. In our global economy, it is critical to achieve consensus on how the profession

translates and uses terminology around the world." says project sponsor Kevin Brennan, CBAP, Chief Business Analyst and Executive Vice President, IIBA.

[More information](#)

INCOSE 2012 Elections

The International Council on Systems Engineering (INCOSE) 2012 elections took place during November.

The candidates and results were:

1. President-Elect

   - David Long - elected

   - Greg Gorman

2. Secretary

   - Eileen Arnold - elected

   - Jim Armstrong

3. Director for Communications

   - Cecilia Haskins - elected

4. Director for Strategy

   - Heidi Hahn

   - Ralf Hartmann - elected

5. Americas Sector Director (Only Americas Chapter Presidents were eligible to vote)

   - Barclay Brown - elected

New officers and directors will be inducted during the International Workshop Opening Plenary on Saturday, 26 January 2013.

[More information](#)

The SEI/AESS/NDIA Business Case for Systems Engineering Study: Results of the Systems Engineering Effectiveness Survey Released

An extensive report, CMU/SEI-2012-SR-009, authored by Joseph P. Elm and Dennis R. Goldenson, has just been released, summarizing the results of a study that had the goal of quantifying the connection between the application of systems engineering (SE) best practices to projects and programs, and the performance of those projects and programs. The survey population consisted of projects and programs executed by system developers reached through the USA National Defense Industrial Association Systems Engineering Division (NDIA-SED), the Institute of Electrical and Electronics Engineers Aerospace and Electronic Systems Society (IEEE-AESS), and the International Council on Systems Engineering (INCOSE).

Analysis of survey responses revealed strong statistical relationships between project performance and categories of specific SE best practices. The survey results show notable differences in the relationship between SE best practices and performance between more challenging and less challenging projects.

Overall, the percentage of higher performance projects where little or no SE was applied was only 15%, for a medium application of SE practices the percentage of higher performance projects increased to 24%, while 57% of projects that were conducted using a high degree of systems engineering capability were in the higher performance category. The detail behind these numbers makes compelling (and convincing!) reading. The report includes correlation coefficients relating project performance with the following practices/factors:

- Project Planning

- Requirements Development & Management

- Verification

- Product Architecture

- Configuration Management

- Trade Studies

- Monitor & Control

- Validation

- Product Integration

- Risk Management

- Use of Integrated Product Teams

- Project Challenge

- Prior Experience.

The study builds upon and increases the volume and breadth of data used compared to an earlier study by Elm and others in 2007 which revealed quantitative relationships between the application of specific SE practices to projects and the performance of those projects, as measured by satisfaction of budgets, schedules, and technical requirements [Elm 2008].

This current report, made generally available immediately after completion of the study, contains only summary information of the survey results. It does not contain the distributions and statistical analyses of each survey question. That information is contained in a companion report [Elm 2013]. The companion report is available to survey participants immediately upon its release; however, it will not be made generally available for one year.

The current report may be downloaded at: http://www.sei.cmu.edu/library/abstracts/reports/12sr009.cfm

References;

[Elm 2008] Elm, J.; Goldenson, D.; El Emam, K.; Donatelli, & N.; Neisa, A. A Survey of Systems Engineering Effectiveness - Initial Results (CMU/SEI-2008-SR-034). Software Engineering Institute, Carnegie Mellon University, 2008. http://www.sei.cmu.edu/library/abstracts/reports/08sr034.cfm

[Elm 2013] Elm, J. & Goldenson, D. The Business Case for Systems Engineering Study: Detailed Response Data. (CMU/SEI-2012-SR-011). Software Engineering Institute, Carnegie Mellon University, 2013. http://www.sei.cmu.edu/library/abstracts/reports/12sr011.cfm

---

Featured Society

Operations Research Society of South Africa (ORSSA)

ORSSA is a member-based, non-profit, scholarly organization with the mission of furthering the interests of those engaged in, or interested in, Operations Research (OR) activities.

We found from ORSSA possibly the longest ever answer to the question "What is Operations Research (OR)?". But the answer is so informative that we thought we'd share it with SyEN readers.

"The name Operations Research (or Operational Research, as it is known in Europe and the United Kingdom) does not really describe the nature of the discipline. Neither do the synonyms Operational Analysis, Quantitative Management, Management Science or Decision Science, which are often used instead of the widely accepted term Operations Research. The name Operations Research stems from the collective war effort in the United Kingdom during World War II when a team of scientists was called upon "to study the strategic and tactical problems associated with air and land defense of the country". This team of scientists focused on decisions regarding the effective utilization of limited military resources, with applications including the use of the then newly invented technology of radar and the effectiveness of new types of bombing strategies. The term Operations Research was coined because the team did research on rendering (military) operations as effective as possible. Encouraged by the successes of the British team of scientists, various research teams in the United States followed suit, studying complex military logistical problems, the effectiveness of aircraft flight patterns, the planning of sea mining activities and the effective utilization of electronic equipment of warfare.

After the war, the degree of success of the various military research teams attracted the attention of industrial engineers who were keen to apply similar optimization and streamlining techniques to problems in business and industry. The modern scope of Operations Research is indeed much wider than military operations. Today the discipline is still concerned with effective system utilization and decision making, but now in business and industry, as well as in government and society at large. The decisions may be about large-scale undertakings, such as the building of a new mine, or about small ones, such as the re-routing of a local bus service. They may be concerned with long-term plans for the redevelopment of an entire inner-city area, or with the immediate problems of selecting a port to handle exports of cars, or how to balance predator-and-prey populations in the management of a game reserve.

The golden thread of commonality amongst the seemingly divergent examples of Operations Research application possibilities mentioned above is that the discipline seeks to employ the scientific method to improve the way in which decisions are made, usually with a view to optimize or streamline the effective use of limited resources (such as time, labor or money). The approach of an operations researcher during this optimization process typically entails the following elements:
Finding out what the problem at an application site really is;

- Assisting the role players at the application site to work out the objectives of the study;

- Observing the functioning and effectiveness of the application site;

- Collecting the relevant data at the application site;

- Determining the main factors influencing the problem at the application site;

- Proposing and testing various ways of solving the problem until one of them can be accepted as the best practical proposition (this often involves the use of a mathematical model); and finally

- Helping the role players at the application site to make the proposed solution work in practice.

Operations Research is not based on any single academic discipline. It draws from the physical sciences, logic, applied mathematics, industrial engineering, the social sciences, economics, statistics and computing, to name but a few, but is none of these on their own. It is typically concerned with decision problems which cut across several disciplines, and attempts to tackle problems on their merits using appropriate tools from any of these sources.

Being about change, Operations Research is not just a "hard" (i.e. quantitative) subject; it also has to be concerned with people and how they react to change. Therefore, in addition to possessing good analytical skills, an operations researcher should also be prepared to search out and try to understand peoples' attitudes, preferences and fears regarding change and improvement. In fact, today it is widely recognized that techniques and approaches within the discipline of Operations Research may be classified as encompassing both "hard" Operations Research and "soft" Operations Research.

Soft Operations Research, Problem Structuring Methods (PSM) or Soft Systems Methodology (SSM) is the result of research done by Peter Checkland and others. It is a systemic approach for tackling problematic situations. It provides a framework for handling the kind of ill-defined or not easily quantified problems (in other words messy problems that lack a formal problem definition). In terms of systems thinking, "hard" systems approaches are not very appropriate for dealing with problems which cannot be defined clearly and that do not have a commonly agreed upon set of outcomes. Problem Structuring Methods attempt to understand complexity, promote learning, to identify weaknesses and to understand relationships.

In "hard" systems approaches, rigid techniques and procedures are used to provide unambiguous solutions to well-defined problems for which an abundance of clean data are available. Hard systems approaches assume that problems are well defined, the scientific approach to problem solving will work and that technical factors will dominate.

Characteristics of Problem Structuring Methods:

- Non-optimizing; seek alternative solutions that are acceptable on separate dimensions, without trade-offs;

- Reduced data demands, achieved by greater integration of hard and soft data with social judgments;

- Simplicity and transparency, aimed at clarifying the terms of conflict;

- Conceptualize people as active subjects;

- Facilitate planning from the bottom up;

- Accept uncertainty, and aims to keep options open for later resolution; and

- Assist in giving appropriate elements of structure to a wide range of problem situations.

Problem structuring methods and "hard" OR are not in conflict. They are typically used in different stages of a problem solving process. Problem structuring methods are applied during the early stages where a problem is not well defined. "Hard" OR is applied during the later stages when problems are well-defined.

The description above of what Operations Research is, suffers from the serious drawback that, as a result of an attempt at giving a balanced and representative account of what the subject field entails, it is rather long-winded. This is typical of descriptions of what Operations Research is. Therefore, a number of shorter definitions of the discipline and what it entails are also available, as are lists of typical specialization areas within Operations Research and modern application areas of Operations Research. Finally, the Institute for Operations Research and Management Science (or INFORMS as the American Operations Research Society is known) has made a rather successful attempt at avoiding long-winded descriptions of what Operations Research is by introducing their catchy slogan that Operations Research is "the Science of Better"."

The Operations Research Society of South Africa (ORSSA) is the national, professional body tasked with furthering the interests of those engaged in, or interested in, Operations Research (OR) activities. ORSSA is continually involved in matters which concern operations researchers, such as:
Organizing conferences at which papers on OR-related topics are delivered;

- Drawing up guidelines for OR education;

- Presenting short courses on specialist topics in OR;

- Marketing OR at secondary school level;

- Providing information to the public on the nature of OR; and

- Providing information to students at tertiary level on career opportunities in OR.

At a national level, the Society is managed by an Executive Committee (EC), which is elected annually at an Annual General Meeting (AGM) of the Society. In the larger centers of South Africa, activities of the Society are arranged by regional chapters.

ORiON is the official scholarly journal of ORSSA and has been published biannually since 1985. The Society also publishes a quarterly Newsletter. A national conference is organized by the Society at different venues annually, usually during the month of September.

More information

INCOSE Technical Operations

INCOSE UK Railway Interest Group

The objectives of the "INCOSE UK Railway Interest Group" are:

- To provide a forum for those interested in Systems Engineering in rail to network in a less formal environment, to exchange good practice and to provide mutual support in an area which can require some sustained perseverance;

- To promote, improve and share the practice of Systems Engineering within the rail industry;

- To foster connections with other professional bodies within rail and thereby promote cross fertilization of knowledge and experience across sectors and community disciplines; and

- To promote awareness of INCOSE UK and encourage membership within the rail industry.

In pursuing its objectives the Group takes a broad view of rail, including heavy rail, metros and light rail, and including both the operational railway and its supply chain.

Approximately seventy systems engineering documents, events and mostly rail-related, are downloadable from: http://www.incoseuk.org.uk/Groups/Railway_Interest_Group/Presentations_List.aspx?CatID=Groups& SubCat=Railway_Interest_Group

More information

Systems Engineering Tools News

Visure Announces Visure Requirements 4.4

Visure Solutions of Madrid, Spain has announced the latest version of its Requirements Lifecycle Management solution Visure Requirements (previously IRQA). Visure Requirements 4.4 provides control of requirements specifications, and offers new features to write them, reuse them and export them. Visure Solutions states that new features include:

- Configuration Management: Restore baseline and element version. Visure Requirements provides out-of-the-box configuration management capabilities to make sure it is possible to track each change made to the elements and specifications. With new Visure Requirements 4.4, entire projects can be rolled back to their previous baselines. Additionally, it is also possible to restore individual versions of elements.

- User views: Work with the flexibility of document views, changing the sorting of the elements by simply dragging and dropping elements in the hierarchical view.

- Reporting: The Report Manager includes new capabilities to represent specifications, such as the logical sorting of elements established in the document or hierarchical view, as well as complete information on traceability, to help produce more complex traceability reports.

- Creating components: The capabilities in Visure Requirements to support reusability and product line management are improved to allow greater flexibility when creating reusable components. Now users can select only the elements that fulfill a specific criterion, such as elements applicable to a specific region, subsystem, priority, or status.

Baldo Rincón, Chief Operations Officer at Visure Solutions says "Visure Requirements software is well suited to companies that are newly implementing requirements or need to control various stages of product development that are process-driven by nature, or apply in a regulated industry."

More information

Systems Engineering Books, Reports, Articles and Papers

Aligning SysML with the B Method to Provide V&V for Systems Engineering

by E. Bousse, D. Mentré, B. Combemale, B Baudry and T. Katsuragi

Abstract. "Systems engineering, and especially the modeling of safety critical systems, needs proper means for early Validation and Verification (V&V) to detect critical issues as soon as possible. The objective of our work is to identify a verifiable subset of SysML that is usable by system engineers, while still amenable to automatic transformation towards formal verification tools. As we are interested in proving safety properties expressed using invariants on states, we consider the B method for this purpose. Our approach consists in an alignment of SysML concepts with an identified subset of the B method, using semantic similarities between both languages. We define a restricted SysML extended by a lightweight profile and a transformation towards the B method for V&V purposes. The obtained process is applied to a simplified concrete case study from the railway industry: a SysML model is designed with safety properties, then automatically transformed into B, and finally imported into Atelier-B for automated proof of the properties."

More Information

The System Concept and Its Application to Engineering

by E. W. Aslaksen

SyEN 050: News in the Field of Systems Engineering | Projec...

http://www.ppi-int.com/newsletter/SyEN-050.php

ISBN: 9783642321689

Format: Hardcover

Publication Date: 14 September 2012

Book Description (from Springer web site):

This book demonstrates how elements of systems engineering, design, economics, or ontology can be combined in order to handle the increasing complexity of engineering projects. It shows why systems engineering should not be overly influenced by the formalism of software engineering. The system concept as a mode of description is a central feature in the book.

More information

Metrics for Requirements Engineering and Automated Requirements Tools

by M. U. Bokhari and S. T. Siddiqui

Abstract. "Software requirements are the foundations from which quality is measured. Measurement enables to improve the software process; assist in planning, tracking and controlling the software project and assess the quality of the software thus produced. Quality issues such as accuracy, security and performance are often crucial to the success of a software system. Quality should be maintained from starting phase of software development. Requirements management, play an important role in maintaining quality of software. A project can deliver the right solution on time and within budget with proper requirements management. Software quality can be maintained by checking quality attributes in requirements document. Requirements metrics such as volatility, traceability, size and completeness are used to measure requirements engineering phase of software development lifecycle. Manual measurement is expensive, time consuming and prone to error therefore automated tools should be used. Automated requirements tools are helpful in measuring requirements metrics. The aim of this paper is to study, analyze requirements metrics and automated requirements tools, which will help in choosing right metrics to measure software development based on the evaluation of Automated Requirements Tools."

More information

Systems Analysis and Design with UML, 4th Edition

by A. Dennis, B. H. Wixom and D. Tegarden

ISBN: 9781118037423

Format: Hardcover

Publication Date: February 2012

Book Description (from the Amazon web site):

This edition of the book continues to offer a concise, modern and applied introduction to OO SAD. The new edition offers updated material, more hands-on exercises, and more applied examples. Furthermore a new emphasis on agile methods tackles programming issues and on business process modeling and ethics to add strategic coverage that appeals to IS majors.

More information

System Safety Engineering and Risk Assessment: A Practical Approach (Chemical Engineering)

by N. J. Bahr

ISBN: 978-1560324164

Format: Hardcover

Publication Date: September 1997

Book Description (from the Amazon web site):

As technological systems become more complex, it becomes increasingly difficult to identify safety hazards and to control their impact. Engineers today are finding that safety and risk touch upon every aspect of any engineered process, from system design all the way through disposal. Employing highly pragmatic examples from a number of industries, the book provides a comprehensive and easily accessible guide on how to build safety into products as well as into industrial processes.

Using a systems approach, the text discusses the best system safety techniques used in various industries, types of hazard analyses, safety checklists and other safety tools, as well as techniques for investigating accidents. It explains how to set up a data management system for a system safety program, and delves into risk assessment, including ways to conduct a risk evaluation. While the book provides engineers with an efficient reference in a critical area, the clarity of the writing along with the case studies and illustrations makes this book accessible to non-technical professionals needing a how to guide for the safety management of complex systems. It is also used by graduate classes involved with ergonomics and occupational safety as well as engineering.

[More information](#)

Systems Architecture, 6th Edition

by S. D. Burd

ISBN: 9780538475334

Format: Hardcover

Publication Date: August 2010

Book Description (from the Amazon web site):

Systems Architecture is the most comprehensive introduction to information systems hardware and software in business. This new edition remains an indispensable tool for IS students, emphasizing a managerial, broad systems perspective for a holistic approach to systems architecture. Each chapter has been updated thoroughly to reflect the changing nature of new technologies, and all end-of-chapter material has been enhanced and expanded.

[More information](#)

Systems Thinking: Managing Chaos and Complexity: A Platform for Designing Business Architecture, 3rd Edition

by J. Gharajedaghi

ISBN: 9780123859150

Format: Hardcover

Publication Date: July 2011

Book Description (from the Amazon web site):

In a global market economy, a viable business cannot be locked into a single form or function anymore. Rather, success is contingent upon a self-renewing capacity to spontaneously create structures, functions, and processes responsive to a fluctuating business landscape. Now in its third edition, Systems Thinking synthesizes systems theory and interactive design, providing an operational methodology for defining problems and designing solutions in an environment increasingly characterized by chaos and complexity.

More information

INCOSE INSIGHT October 2012, Volume 15, Issue 3 Released

The October 2012 Volume 15 Issue 3 of INSIGHT is ready for members to view or download on INCOSE Connect at the INSIGHT Library, along with all the past issues.

Special Feature: INCOSE's 22nd Annual International Symposium

The special feature highlights the plenary sessions, panel sessions, tracks and other activities that took place during the course of the symposium. This year's International Symposium in Rome was another important milestone as INCOSE pursues its mission to "share, promote, and advance the best of systems engineering from across the globe for the benefit of humanity and the planet."

Conferences and Meetings

System-of-Systems Workshop: An NIE Experience
January 22 - 25, 2013, El Paso, TX, USA
More information

Seventh International Workshop on Variability Modelling of Software-Intensive Systems (VaMoS)
January 23 - 25, 2013, Pisa, Italy
More information

INCOSE International Workshop IW2013
January 26 - 29, 2013, Jacksonville, Florida USA
More information

4th International Conference on Intelligent Systems, Modelling and Simulation, ISMS2013 NEW
January 29 – 31, 2013, Bangkok, Thailand
More information

Conference Digital Enterprise Design & Management (DED&M 2013)
February 11 - 12, 2013, Paris, France
More information

International Conference on Model-Driven Engineering and Software Development - MODELSWARD 2013
February 19 - 21, 2013, Barcelona, Spain
More information

29th Annual Test and Evaluation Conference and Displays
February 25-28, 2013, Charlotte, NC, USA
More information

International Symposium on Engineering Secure Software and Systems (ESSoS)
February 27 – March 3, 2013, Paris, France
More information

Integrated-EA Conference
March 4 - 6 2013, London, United Kingdom
More information

INCOSE IL 2013
March 5 – 6, 2013, Daniel Hotel Herzlia
More information

Integrated EA 2013 NEW
5 - 6 March 2013, London, UK
More information

ASTEC 2013, Asian Simulation Technology Conference
March 7 - 9, 2013, Shanghai, China
More information

Swiss Testing Day 2013 NEW
March 13, 2013, Zurich
More information

2013 INCOSE LA Mini-Conference
March 16, 2013, Los Angeles, CA, USA
More Information

The Requirements Engineering Track - 6th Edition at The 28th Annual ACM Symposium on Applied Computing (SAC 2013)
March 18 - 22, 2013, Coimbra, Portugal
More information

11th Annual Conference on Systems Engineering Research (CSER 2013)
March 19 – 22, 2013, Atlanta, Georgia, USA
More information

EMO 2013 - the 7th International Conference on Evolutionary Multi-Criterion Optimization
March 19 - 22, 2013, Sheffield, UK
More information

Spring Simulation Multi-Conference (SpringSim'13)
April 7 - 10, 2013, San Diego, CA, USA
More information

3rd International Workshop on Model-driven Approaches for Simulation Engineering
part of the Symposium on Theory of Modeling and Simulation (SCS SpringSim 2013)
April 7 - 10, 2013, San Diego, CA, USA
More information

YoungOR 18new
April 9 - 11, 2013, University of Exeter, Exeter, Unoted Kingdom
More information

Symposium on Theory of Modeling and Simulation (TMS/DEVS 2013) NEW
April 7 - 10, 2013, San Diego, California, USA
More information

International Conference on Manufacturing Systems Engineering (ICMSE 2013)
April 14 - 15, 2013, Venice, Italy
More information

ECEC'2013, 20th European Concurrent Engineering Conference
April 15 - 17, 2013, Lincoln, United Kingdom
More information

SysCon 2013
April 15 - 18, 2013, Orlando, FL, USA
More information

SETE 2013
April 29 – May 1, 2013, Canberra, ACT, Australia
More information

SATURN 2013 Conference NEW
April 29 - May 3, 2013, Minneapolis, Minnesota, USA
More information

IST-115 Symposium on Architecture Definition and Evaluation NEW
May 13 - 14, 2013, Toulouse, France
More information

5th NASA Formal Methods Symposium (NFM) 2013 NEW
May 14 - 16, 2013, Moffett Field, CA, USA
More information

Test Instrumentation Workshop: T&E on a Sustainment Budget
May 14 - 17, 2013, Las Vegas, NV, USA
More information

International Conference on Software and Systems Process (ICSSP) 2013 ⩗⩗ NEW
May 18-19, 2013. San Francisco, CA, USA (co-located with ICSE 2013)
More information

ISC'2013, 11th Annual Industrial Simulation Conference
May 22 - 24, 2013, Ghent, Belgium

8th Annual System of Systems Conference ⩗⩗ NEW
June 2 - 6, 2013, Maui, Hawaii
More information

KIM2013 Knowledge and Information Management Conference
June 4 - 5, 2013, Meriden, United Kingdom
More information

13th International Conference on Process Improvement and Capability dEtermination (SPICE)
June 4 – 6, 2013, University of Bremen, Germany
More information

10th International Conference on integrated Formal Methods (iFM 2013)
June 10 - 14, 2013, Turku, Finland
More information

COMADEM 2013
June 11 - 13, 2013, Helsinki, Finland
More information

International Conference on Business Process Modeling, Development, and Support (BPMDS'2013)
⩗⩗ NEW
June 17 - 18, 2013, Valencia, Spain
More information

81st MORS (Military Operations Research Society) Symposium
June 17 - 20, 2013, United States Military Academy in West Point, NY, USA
More information

22nd WETICE Conference (WETICE-2013) ⩗⩗ NEW
June 17 - 20, 2013, Hammamet, Tunisia
More information

4TH ACM SIGSOFT International Symposium on Architecting Critical Systems (ISARCS 2013) ⩗⩗ NEW
June 17 - 21, 2013, Vancouver, Canada
More information

6th International Conference on Model Transformation (ICMT 2013) ⩗⩗ NEW
June 18 – 19, 2013, Budapest, Hungary
More information

CM 2013
June 18 – 20, 2013, Krakow, Poland
More information

Swiss Requirements Day 2013
June 19, 2013, Zürich, Switzerland

IFAC MIM '2013 Conference on Manufacturing Modelling, Management, and Control NEW
June 19 – 21, 2013, Saint Petersburg, Russia
More information

ASEE 120th Annual Conference & Exposition
June 23 - 26, 2013, Atlanta, Georgia, USA
More information

12th International Symposium of the Analytic Hierarchy Process/Analytic Network Process (ISAHP 2013)
June 23 – 26, 2013, Kuala Lumpur, Malaysia
More information

IS 2013 – Philadelphia
June 24 – 27, 2013, Philadelphia, Pennyslvania USA
More information

16th International System Design Languages Forum Model-driven dependability engineering (SDL 2013)
 NEW
June 26 - 28, 2013, Montréal, Canada
More information

7th ACM International Conference on Distributed Event-Based Systems (DEBS 2013) NEW
June 29 - July 3, 2013, Arlington, Texas, USA,
More information

18th European Conference on Pattern Languages of Programs (EuroPLoP 2013) NEW
July 10 - 14, 2013, Kloster Irsee, Bavaria, Germany
More information

ISSS 2013: The 57th World Conference of the International Society for the Systems Sciences new
July 14 – 19, 2013, Hai Phong City, Viet Nam
More information

15th International Conference on Human-Computer Interaction 2013 (HCI International 2013)
July 21 – 26, 2013, Las Vegas, Nevada, USA
Incorporating:
10th International Conference on Engineering Psychology and Cognitive Ergonomics
7th International Conference on Universal Access in Human-Computer Interaction
5th International Conference on Virtual, Augmented and Mixed Reality
5th International Conference on Cross-Cultural Design
5th International Conference on Online Communities and Social Computing
7th International Conference on Augmented Cognition
4th International Conference on Digital Human Modeling and applications in Health, Safety, Ergonomics and Risk Management
2nd International Conference on Design, User Experience and Usability
1st International Conference on Distributed, Ambient and Pervasive Interactions
1st International Conference on Human Aspects of Information Security, Privacy and Trust
More information

Requirements Engineering Conference (RE13) NEW
July 15 - 19, 2013, Rio de Janeiro, Brasil
More information

The Eighteenth IEEE International Conference on Engineering of Complex Computer Systems (ICECCS 2013) NEW
July 17 - 19, 2013, National University of Singapore, Singapore

More information

The Sixteenth SDL FORUM - SDL2013
Date and location to be determined, 2013
More information

31st International Conference of the System Dynamics Society ⩗⩗ NEW
July 21 - 25, 2013, Cambridge, Massachusetts, USA
More information

ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference (IDETC/CIE2013)
August 4 - 7, 2013, location TBA, USA
More information

24th International Conference on Concurrency Theory (CONCUR 2013) ⩗⩗ NEW
August 27 - 30, 2013, Buenos Aires Argentina
More information

The 4th IFAC Conference on Modelling and Control in Agriculture, Horticulture and Post Harvest Industry (IFAC AGRI 2013) ⩗⩗ NEW
August 28 - 30, 2013, Espoo, Finland.
More information

10th International Conference on Preservation of Digital Objects ⩗⩗ NEW
September 2 - 5, 2013, Lisbon, Portugal
More information

OR55 Annual Conference of the OR Society
September 3 - 5, 2013, Exeter University, Exeter, United Kingdom
More information

International Conference on Operations Research
September 3 - 6, 2013, Rotterdam, The Netherlands
More information

9th IFAC Symposium on Nonlinear Control Systems (NOLCOS 2013) ⩗⩗ NEW
September 4 - 6, 2013, Toulouse, France
More information

4th WORKSHOP ON DEPENDABLE CONTROL OF DISCRETE SYSTEMS=20 (DCDS'13) ⩗⩗ NEW
September 4 - 6, 2013, York, United Kingdom
More information

APCOSE 2013
September 9 - 11, 2013, Keio University in Japan
More information

SIMEX'2013
September 10 - 13, 2013, Brussels, Belgium

30th Annual International Test and Evaluation Symposium
September 16-20, 2013, Denver, CO, USA
More information

27th European Simulation and Modelling Conference - ESM'2013

October 2013, Lancaster, UK
More information

ASTEC 2014
March 2014, Digipen Institute of Technology, Singapore
More information

ISC'2014
June 11 - 13, 2014, Skövde, Sweden

19th World Congress of the International Federation of Automatic Control (IFAC 2014)
August 24 - 29, 2014, Cape Town, South Africa
More information

SIMEX'2014
September 2014, Brussels, Belgium

INCOSE Europe, Middle East & Africa (EMEA) Sector: 1st EMEA Systems Engineering Conference 2014 (formerly EuSEC)
October 2014, Cape Town, South Africa

ASTEC'2015, Asian Simulation Technology Conference
March 2015, Japan

ISC'2015 13th Annual Industrial Simulation Conference
June 2015, St.Petersburg, Russia

SIMEX'2015
September 2015, Brussels, Belgium

ISC'2016 14th Annual Industrial Simulation Conference
June 2016, Bucharest, Romania

Education and Academia

The Systems Engineering Research Center

The Systems Engineering Research Center (SERC), a University-Affiliated Research Center of the US Department of Defense, leverages the research and expertise of senior lead researchers from 20 collaborator universities and not-for-profit research organizations throughout the United States. SERC is said by the Center to be unprecedented in the depth and breadth of its reach, leadership, and citizenship in Systems Engineering. Led by Stevens Institute of Technology, and principal collaborator, the University of Southern California (USC), the SERC provides a critical mass of systems engineering researchers – a community of broad experience, deep knowledge, and diverse interests. SERC researchers have worked with a wide variety of domains and industries and so are able to bring views and ideas from beyond the traditional defense industrial base. Establishing such a community of focused SE researchers, while difficult, promises results well beyond what any one university could accomplish.

More information

Comprehensive Modeling for Advanced Systems of Systems

The COMPASS consortium is a group of researchers and companies committed to collaborative research on model-based techniques for developing and maintaining Systems of Systems (SoS).

Modern networking technologies let systems cooperate by sharing resources and offering services to one another so that the resulting system of systems has a behaviour that is greater than just the sum of its parts. For example, the information systems of fire, police and hospital services can together offer a flexible and responsive SoS for emergency management, even though the individual systems were not intended for collaboration. At a different scale, the integration of systems on board an aircraft can offer more energy-efficient and robust flight control.

Although there are great opportunities here, the design of innovative products and services that take advantage of Systems of Systems (SoS) technology is hampered by the complexity caused by the heterogeneity and independence of the constituent systems, and the difficulty of communication between their diverse stakeholders. Developers lack models and tools to help make trade-off decisions during design and evolution leading to sub-optimal design and rework during integration and in service.

The research agenda involves:

- Developing a modeling framework for SoS architectures.

- Providing a sound, formal semantic foundation to support analysis of global SoS properties.

- Building an open, extendible tools platform with integrated prototype plug-ins for model construction, simulation, test automation, static analysis by model-checking, and proof, and links to an established architectural modelling language.

- Evaluating technical practice and advanced methods through substantial case studies.

More information

Some Systems Engineering-Relevant Websites

---

http://portal.modeldriven.org/project/foundationalUML

Open source software which is a reference implementation, consisting of software and related files, for the OMG specification called the Semantics of a Foundational Subset for Executable UML Models (fUML), is downloadable from this page. The reference implementation is intended to implement the execution semantics of UML activity models, accepting an XMI file from a conformant UML model as its input and providing an execution trace of the selected activity model(s) as its output.

The reference implementation was initially developed as part of a Lockheed Martin Corporation funded project with Model Driven Solutions in 2008, and has been maintained by Model Driven Solutions on modeldriven.org since then. The objectives for this reference implementation are said to be to encourage tool vendors to implement the standard in their tools, and to provide a reference that can assist in evaluating conformance of vendor implementations to the specification. There are now at least two commercial implementations of fUML available:

- The Cameo Simulation Toolkit for the MagicDraw tool from No Magic.

- AM|USE 2.0 by LieberLieber for the Enterprise Architect tool from Sparx Systems.

http://www.sebokwiki.org/1.0/index.php/SEBoK_1.0_Introduction

This web page introduces the Systems Engineering Body of Knowledge (SEBoK) V1.0. The purpose of the SEBoK is to provide a widely accepted, community based, and regularly updated baseline of SE knowledge. This baseline will strengthen the mutual understanding across the many disciplines involved in developing and operating systems. Shortfalls in such mutual understanding are a major source of system failures, which have increasingly severe impacts as systems become more global, interactive, and critical.

http://www.ndia.org/Divisions/Divisions/SystemsEngineering/Pages /Systems_Engineering_Effectiveness_Committee.aspx

This is the website of the (USA) National Defense Industrial Association Systems Engineering Effectiveness Committee (SEEC). The mission of the SEEC is to assist the Systems Engineering (SE) community in achieving a quantifiable and persistent improvement in program performance through appropriate application of systems engineering principles and best practices. Its current activities are to:
1. Identify systems engineering principles and best practices proven to provide improved program performance.
2. Identify a set of leading indicators that provide insight into technical performance at major decision points for managing programs quantitatively across their life cycle.

The website links to a three reports:

- NDIA System Development Performance Measurement Report, December 2011

- Systems Engineering Effectiveness Study, December 2008

- Systemic Root Cause Analysis (of program failures), March 2009.

In turn, the first reference above links to the following system development performance measurement standards, publications, guidance, legislation, directives, studies, reports:

Systems Engineering Leading Indicators Guide, Version 2.0, January 29, 2010.
http://www.incose.org/ProductsPubs/products/seleadingIndicators.aspx

Practical Software and Systems Measurement (PSM): Objective Information for Decision Makers.
http://www.psmsc.com/Default.asp

ISO/IEC 15939:2007, Systems and software engineering – Measurement process
http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44344

Technical Measurement, Version 1, December 2005, INCOSE-TP-2003-020-01
http://www.incose.org/ProductsPubs/products/techmeasurementguide.aspx

Weapon Systems Acquisition Reform Act of 2009.
http://www.ndia.org/Advocacy/PolicyPublicationsResources/Documents/WSARA-Public-Law-111-23.pdf

Better Buying Power: A Memorandum for Acquisition Professionals.
Dr. Ashton Carter, OUSD (AT&L). September 2010.
https://dap.dau.mil/Pages/NewsCenter.aspx?aid=157

Naval Probability of Program Success (PoPS). ASN RDA.
https://acquisition.navy.mil/rda/content/view/full/6601

Joint Capabilities Integration and Development Systems (JCIDS), Key Performance Parameters / Key System Attributes (Enclosure B).
https://acc.dau.mil/communitybrowser.aspx?id=267116

Systems Engineering Effectiveness Measures.

Center for Systems and Software Engineering, University of Southern California.
http://csse.usc.edu/csse/research/SEEM/index.html

Pre-Milestone A and Early-Phase Systems Engineering: A Retrospective Review and Benefits for Future Air Force Acquisition.
Air Force Studies Board, 2008, The National Academies Press.
http://www.nap.edu/catalog.php?record_id=12065

Systems Engineering Plan Outline, April 20, 2011.
Office of the Deputy Assistant Secretary of Defense, Systems Engineering.
http://www.acq.osd.mil/se/pg/guidance.html

Technology Readiness Assessment Guidance, April 2011, revision 13 May, 2011.
Assistant Secretary of Defense for Research and Engineering (ASD(R&E)).
In particular at this site:
Manufacturing Readiness Level Deskbook, July 2011, Version 2.01 System Development Performance Measurement 14.
http://www.acq.osd.mil/ddre/publications/docs/TRA2011.pdf

Manufacturing Readiness Levels Body of Knowledge.
Department of Defense Manufacturing Technology Program.
http://www.dodmrl.com

Top Systems Engineering Issues in U.S. Defense Industry.
NDIA Systems Engineering Division, September 2010.
http://www.ndia.org/Divisions/Divisions/SystemsEngineering/Documents/Studies/Top%20SE%20Issues%202010%20Report%20v11%20FINAL.pdf

A Survey of Systems Engineering Effectiveness: Initial Report.
Software Engineering Institute and NDIA Systems Engineering Division, November 2007, CMU/SEI-2007-SR-014.
http://www.sei.cmu.edu/library/abstracts/reports/07sr014.cfm

NDIA Industrial Committee on Program Management (ICPM).
http://www.ndia.org/Divisions/IndustrialWorkingGroups/IndustrialCommitteeForProgramManagement/Pages/default.aspx

NDIA Systems Engineering Conference, October 2010.
DoD presentations on Systems Engineering metrics and systemic findings of program execution issues.
http://www.dtic.mil/ndia/2010systemengr/2010systemengr.html

aiaa.kavi.com/public/pub_rev/G-043-201X_for_PR_July_2011.pdf
This website contains a .pdf download of the BSR/AIAA G-043A-201X (Revision of G-043-1992) Proposed American National Standard "Guide to the Preparation of Operational Concept Documents" This document is not an approved AIAA Standard. It was distributed for review and comment.

Standards and Guides

Executable UML/SYSML Semantics Project Prepared by Data Access Technologies, Inc.

The objective of the Executable UML/SysML Semantics Project is to specify and demonstrate the semantics required to execute activity diagrams and associated timelines per the SysML v1.0 specification and to specify the supporting semantics needed to integrate behavior with structure and realize these activities in blocks and parts represented by activity partitions. The SysML semantics will build on the semantics of a Foundation Subset for Executable UML Models, which is currently in preparation for submission as an OMG standard.

The .pdf can be downloaded here.

ANSI/AIAA G-043A-2012 - Guide to the Preparation of Operational Concept Documents - Released

ANSI/AIAA G-043A-2012 "Guide to the Preparation of Operational Concept Documents" has been released.

The standard can be purchased for US$70 from the ANSI website: http://webstore.ansi.org/RecordDetail.aspx?sku=ANSI%2fAIAA+G-043A-2012#.UMA81oVgA4M,
and for US$69.95 from the AIAA website: https://www.aiaa.org/StandardsDetail.aspx?id=12878.

The standard is available for download free to AIAA ad INCOSE members, from the organisations IONS' respective weesites.

The purpose of this Guide is twofold. First, the Guide describes a time-tested process for operational concept development. Second, it is intended to recommend how to compile the information developed during operational concept development into one or more Operational Concept Documents (OCDs) encompassing the full range of the product lifecycle (Haskins, 2010): concept, development, production, utilization, support, and retirement stages.

A recognized systems engineering best practice is early development of operational concepts during system development and documentation of those operational concepts in one or more operational concept documents. Recognizing this best practice, U. S. Department of Defense (DoD) and NASA standard procedures have required that information relating to system operational concepts is prepared in support of the specification and development of systems. In the past, the DoD has published Data Item Descriptions (DIDs), and NASA has published Data Requirements Documents (DRDs), which describe the format and content of the information to be provided. This AIAA Guide describes which types of information are most relevant, their purpose, and who should participate in the operational concept development effort. It also provides advice regarding effective procedures for generation of the information and how to document it.

More information

G-118-2006e - AIAA Guide for Managing the Use of Commercial Off the Shelf (COTS) Software Components for Mission-Critical Systems

The purpose of this Guide is to assist development and maintenance projects (teams and individuals) that have to address the use of, or consideration of, COTS products within large, complex systems, including but not limited to mission critical systems. This assistance is provided by capturing a set of information about COTS products (benefits, risks, recommended practices, lifecycle activity impacts) and mission critical systems (variety of MCS, special needs for MCS, differences between MCS and other types of systems) and then providing some linkage between these topics so that various types of stakeholders can find useful information. The document should be of value to both management and technical individuals/teams. It should also be of value to teams that are dealing with non-MCS, in that the scope is not limited to only MCS.

More information

G-077-1998e - AIAA Guide for the Verification and Validation of Computational Fluid Dynamics Simulations

This guide provides a means for assessing the credibility of modeling and simulation in computational fluid dynamics (CFD). The two main principles necessary for assessing credibility are verification and validation. This document defines the key terms, discusses fundamental concepts, and specifies general procedures for conducting verification and validation of CFD simulations. This terminology and methodology may also be useful in other engineering and science disciplines.

More information

G-035A-2000e - AIAA Guide to Human Performance Measurements

This guide suggests methods for measuring human performance for the purpose of scientific research and system test and evaluation. The information contained in this document is provided as guidance, not mandated as direction. This guide should be considered during the planning, conduct, and analysis of human performance measurement activities. The objectives of this guide are: (1) to foster human performance measurement (HPM) techniques that have proved to be effective; (2) to promote commonality across research projects and, thus, enable comparison of results across evaluations; and (3) to enable the development and use of common HPM tools for data collection and data processing.

More information

R-100A-2001 - AIAA Recommended Practice for Parts Management

This standard establishes a parts management approach that is consistent with the new acquisition reform business environment. It provides a performance-based process to address the growth in the market for commercial electronics along with the decrease in aerospace market share. The strategy provided in this document advocates managing risk up front through the use of integrated product teams and addressing parts obsolescence, diminishing sources, and technology insertion. Key elements that should be included in a parts management plan are identified, and an approach to sharing data among suppliers is described.

More information

S-S-117-2010e - AIAA Standard — Space Systems Verification Program and Management Process

This document enforces a systematic approach to planning and executing verification programs for manned and unmanned space systems based on a distributed approach that corrects fundamental deficiencies associated with the traditional centralized verification approach. Thus, this document corrects generic problems in conducting verification that existed even during post-Total System Program Responsibility or "Faster, Better, Cheaper" policy that prospered late 1990 through early 2000 for developing complex space systems. This standard is intended to help those in the space community develop reliable systems that meet requirements while ensuring proper accommodations of heritage and/or commercial systems in their developing systems. It also helps to facilitate the closely coordinated validation activities with those of verification, as the distributed systems engineering processes utilized in the latter can be easily adopted by the former activities.

More information

S-102-2-4-2009e -ANSI/AIAA Performance-Based Failure Reporting, Analysis & Corrective Action System (FRACAS) Requirements

This standard provides the basis for developing the performance-based Failure Reporting, Analysis & Corrective Action System (FRACAS) to resolve the problems and failures of individual products along with those of their procured elements. The requirements for contractors, the planning and reporting needs, along with the analytical tools are established. The linkage of this standard to the other standards in the new family of performance-based reliability and maintainability (R&M) standards is described, and a large number of keyword data element descriptions (DED) for use in automating the FRACAS process are provided.

More information

S-102-2-4-2009e -ANSI/AIAA Performance-Based Failure Review Board (FRB) Requirements

This standard provides the basis for developing the performance-based Failure Review Board (FRB), which is a group consisting of representatives from appropriate project organizations with the level of responsibility and authority to assure that root causes are identified and corrective actions are effected in a timely manner for all significant failures. Although good engineering practice suggests that most product development projects should include a formal FRB, the basic FRB functions may devolve to a single individual on small projects. Planning and reporting requirements and analytic tools are provided for contractors. The linkage of this standard to the other standards in the new family of performance-based reliability and maintainability (R&M) standards is described, and a large number of keyword data element descriptions (DED) for use in automating the FRB process are provided.

More information

S-102-2-18-2009e - ANSI/AIAA Performance-Based Fault Tree Analysis Requirements

This standard provides the basis for developing the performance-based fault tree analysis (FTA) to review and analytically examine a system or equipment in such a way as to emphasize the lower-level fault occurrences that directly or indirectly contribute to the system-level fault or undesired event. The requirements for contractors, planning and reporting needs, and analytical tools are established. The linkage of this standard to the other standards in the new family of performance-based reliability and maintainability (R&M) standards is described, and a large number of keyword data element descriptions (DED) for use in automating the FTA process are provided.

More information

S-102-2-4-2009e - ANSI/AIAA Performance-Based Product Failure Mode, Effects and Criticality Analysis (FMECA) Requirements

This standard provides the basis for developing the analysis of failure modes, their effects, and criticality in the context of individual products along with the known performance of their elements. The requirements for

contractors, the planning and reporting needs, along with the analytical tools are established. The linkage of this standard to the other standards in the new family of performance-based reliability and maintainability standards is described, and all of the keywords for use in automating the Product FMECA process are provided.

More information

S-102-2-2-2009e - ANSI/AIAA Performance-Based System Reliability Modeling Requirements

This standard provides the basis for developing performance-based System Reliability Modeling to develop mathematical or simulation models to be used for making numerical apportionments and reliability predictions based on the reliability characteristics and functional interdependencies for all configured items required to perform the mission. The requirements for contractors, the planning and reporting needs, along with the analytical tools are established. The linkage of this standard to the other standards in the new family of performance-based Reliability and Maintainability (R&M) standards is described, and a large number of keyword data element descriptions (DED) for use in automating the System Reliability Modeling process are provided.

More information

Definitions to Close On

Factory Acceptance Testing, Site Acceptance Testing and Commissioning

Factory Acceptance Testing (FAT): Testing conducted at the manufacturing facility, before transportation to the user's facility, to assure that the product as manufactured meets the mutually agreed requirements.
Source: Clive Tudge and Robert Halligan

Factory Acceptance Testing (FAT): Factory Acceptance Test (FAT) is a major project milestone in a laboratory automation project where the equipment and/or system integrator demonstrates that the system design and manufacturing meets the contract or Purchase Order (P.O.) specifications, created by the system owner/project manager/project team).
Source: http://www.labautopedia.org/mw/index.php/Factory_Acceptance_Testing_(FAT)

Factory Acceptance Testing (FAT): Factory acceptance testing (FAT) is the process of in-house (or in-factory) testing that determines the product or part meets the stated specification. This testing is conducted by the suppliers prior to shipment of the product to the customer.
Source: http://www.acutest.co.uk

Factory Acceptance Testing (FAT): FAT is typically the final phase of vendor inspection and testing that is performed prior to shipment to the installation site. The FAT should demonstrate conformance to the specifications in terms of functionality, serviceability, performance and construction (including materials).
Source: Federal Highway Administration, "Testing Programs for Transportation Management Systems – A Technical Handbook", Washington, DC, 2007.

Site Acceptance Testing (SAT): Testing conducted at the user's facility to assure that the product as delivered or installed meets the mutually agreed requirements. STA is typically conducted to ensure that transportation from the manufacturing facility, or installation in the user's facility, has not damaged or otherwise affected the product.
Source: Clive Tudge and Robert Halligan

Site Acceptance Testing (SAT): Site Acceptance Testing is acceptance testing by users or customers at their

own site, to determine whether or not a component or system satisfies the user/customer needs and maps correctly to the agreed business processes.
Source: http://www.acutest.co.uk

Commissioning: Process by which an equipment, facility, or plant (which is installed, or is complete or near completion) is tested to verify if it functions according to its design objectives or specifications.
Source: http://www.businessdictionary.com/definition/commissioning

Commissioning: A quality focused process for enhancing the delivery of a project. The process focuses upon verifying and documenting that the facility and all of its systems and assemblies are planned, designed, installed, tested, operated, and maintained to meet the Owner's Project Requirements.
Source: ASHRAE Guideline 0-2005

Commissioning: Commissioning is a comprehensive and systematic process to verify that the building systems perform as designed to meet the State's requirements.
Source: Department of General Services, State of California, USA

Commissioning: Commissioning or "hot green testing" involves bringing on the high-voltage switchgear, livening the power transformers, switching on the MCC circuit breakers, livening the MCCs, and livening all auxiliary and power supplies for instrumentation control panels and the like.
Source: The Institution of Professional Engineers New Zealand Incorporated (IPENZ) Practice Note 09, July 2007

Commissioning: The correct definition of a commissioning process is dependent on the industrial application. Generally, one may define it as a "well-planned, documented, and managed engineering approach to the start-up and turnover of facilities, systems, and equipment to the end user that results in a safe and functional environment that meets established design requirements and stakeholder expectations".
Source: The International Society for Automation / ISPE "Pharmaceutical Engineering Guides for New and Renovated Facilities, Volume 5, Commissioning and Qualification"

Commissioning: An engineering approach to site-dependent configuration (if any), start-up, check-out and handover of facilities, systems, and equipment to the end user.
Source: Robert Halligan

PPI News (see www.ppi-int.com)

Congratulations Suja and Daniel!

On behalf of everyone at Project Performance International, we would like to extend out warmest congratulations to our editor Suja Joseph-Malherbe and husband Daniel on the arrival of their baby boy Jeevan. Jeevan was born on 22 November 2012. We wish them the very best of health and happiness always and look forward to Jeevan's influence in future SyEN editions.

Festive Season Celebration - Course Discount

In the spirit of the festive season, PPI is offering our clients a 10% discount on all new 2013 course registrations received and paid for by 31 December, 2012. This discount is in addition to any other discounts available, and is applied to the amount that would otherwise be payable. This offer does not apply to our PRINCE2 or MSP® course offerings.

We wish all SyEN readers a very happy festive season and a healthy and prosperous new year. See you in January!

Robert Halligan
Managing Director, Project Performance International

CTI delivers CSEP Exam Preparation Training to INCOSE South Africa

Certification Training International (a PPI subsidary) recently delivered CSEP Examination Preparation training in Pretoria, South Africa. The workshop was an outstanding success and plans are underway to conduct another workshop to the growing INCOSE South Africa Chapter in the new year. For further information on CTI's training workshop visit: http://www.certificationtraining-int.com/

PPI Events (see www.ppi-int.com)

Systems Engineering Public 5-Day Courses

Upcoming Locations Include:

- Amsterdam, The Netherlands
- Adelaide, Australia
- Brisbane, Australia
- Rio de Janeiro, Brazil
- Munich, Germany

Requirements Analysis and Specification Writing Public Courses

Upcoming Locations Include:

- Melbourne, Australia
- Stellenbosch, South Africa
- Las Vegas, USA
- Amsterdam, The Netherlands

Software Engineering Public 5-Day Courses

Upcoming Locations Include:

- Sydney, Australia
- Pretoria, South Africa

- Amsterdam, The Netherlands

OCD/CONOPS Public Courses

Upcoming Locations Include:

- Brasilia, Brazil

- Pretoria, South Africa

- Las Vegas, USA

Cognitive Systems Engineering Courses

Upcoming Locations Include:

- Adelaide, Australia

- Las Vegas, USA

CSEP Preparation Course (Presented by PPI subsidiary Certification Training International)

Upcoming Locations Include:

- Las Vegas, USA

- Austin, USA

- Munich, Germany

PPI Upcoming Participation in Professional Conferences

PPI will be participating in the following upcoming events. We look forward to chatting with you there.

- CSD&M 2012 Conference | Participating | Paris, France (12 - 14 December, 2012)

- IDEX 2013 | Exhibiting | United Arab Emirates (17 - 21 February, 2013)

- Latin American Defence & Security International Exhibition (LAAD) | Exhibiting | Rio de Janerio, Brazil (9 - 12 April, 2013)

Kind regards from the SyEN team:
Robert Halligan, Managing Editor, email: rhalligan@ppi-int.com
Suja Joseph-Malherbe, Editor, email: sjosephmalherbe@ppi-int.com
Stephanie Halligan, Production, email: shalligan@ppi-int.com

Project Performance International
2 Parkgate Drive, Ringwood, Vic 3134 Australia
Tel: +61 3 9876 7345
Fax: +61 3 9876 2664
Tel Brasil: +55 11 3230 8256

Tel UK: +44 20 3286 1995
Tel USA: +1 888 772 5174
Web: www.ppi-int.com
Email: contact@ppi-int.com

Copyright 2012 Project Performance (Australia) Pty Ltd, trading as Project Performance International

Tell us what you think of SyEN: email to contact@ppi-int.com

If you do not wish to receive a copy monthly of SyEN in future, please reply to this e-mail with "Remove" in the subject line. All removals are acknowledged; you may wish to contact PPI if acknowledgement is not received within 7 days.