SYDPIM integration of SD and PERT/CPM tools: Assessing Fagan Analysis in a large-scale software project.

Alexandre G. Rodrigues
Department of Information Systems
University of Minho
Portugal, EU
Tel.: +351 (53) 510 149, Fax.: +351 (53) 510 250
E-mail: alex.rodrigues@dsi.uminho.pt
www.dsi.uminho.pt/~alex

While Project Management is crucial to many organisations, world-wide surveys continue to show major over-runs. The causes are complex issues of human and strategic nature. System Dynamics gathers the requisites to address these issues. However, so far studies and practical applications have been implemented in total isolation from the traditional PERT/CPM models, with no attempt at exploring their relationship and formal integration. The author has developed a formal integrated methodology, called SYDPIM, which includes a model development and validation method, a process framework to articulate the combined use of the two models, and formal links of data and structure. It provides a rigorous framework for the formal exchange of data and structural readjustment of the models. In this paper, SYDPIM is briefly describe and is illustrated using a real scenario from a software project: half-way through system design, management needs to decide whether Fagan inspections is worth implementing.

Acknowledgement: this work has been developed in the course of a NATO PhD programme undertaken at the Department of Management Science, University of Strathclyde, UK. This research has been funded by Junta Nacional de Investigacao Cientifica e Tecnologica (JNICT), Portugal, and supported by BAeSEMA Ltd., UK.

Research background – an overview

The discipline of Project Management emerged in the late 1920s and over the last decodes evolved as a science of its own right (Nicholas 1990). As a consequence a well established project management framework was developed, comprising a set of tools and techniques to help with the operational issues (e.g. WBS, OBS, PERT/CPM networks, and the *earned value* concept) (Rodrigues and Bowers 1996).

However, despite its usefulness, this conventional approach has not been able to cope with all problems that can affect projects. Projects still fail across various industries, with over-runs ranging from 40%-200% being common (Morris and Hough 1980). In most cases, the root causes for these failures are to be found in areas not addressed by conventional tools: soft human factors, political and strategic issues, all of systemic nature. Conventional tools are based on a bottom-up approach where the focus is on controlling the individual elements of a project, with little emphasis on their systemic interactions.

System Dynamics (SD) emerged as a modelling approach aimed at analysing the dynamic interactions between the various components of a management system (Forrester 1961). This methodology has been applied with increasing success in the filed of project management, as it addresses the systemic issues that the traditional tools fail to capture (Cooper 1980, Abdel-Hamid 1990, Williams et all 1997). However, these applications have been implemented in total isolation from the traditional framework (Rodrigues 1994, Rodrigues and Bowers 1996). The integration of the two approaches has been proposed as the best route (Rodrigues 1994), and this has been further recognised by major practitioners (Williams et al 1995; Burdick et al 1998).

After an exhaustive review of past applications of SD to project management (Rodrigues 1994), the author has developed a conceptual integrated framework (Rodrigues and Williams 1995, 1997), and validated it within a large-scale software project (Rodrigues and Williams 1998). This was further extended it into a structured formal methodology (Rodrigues 1997, 1999), which is briefly presented in this paper.

The SYDPIM methodology – a process overview

What is SYDPIM and how does it work? The background principle of SYDPIM is a perspective of the project management process as being comprised of two main sub-processes: (1) the engineering process of product development, and (2) the management process of project control. The engineering process comprises all the working activities directly related with developing the "physical product". These activities are typically organised in a life-cycle fashion, from the definition of the system requirements to the final system testing.

The management process comprises the managerial activities of developing and updating the project plan, monitoring progress, and taking corrective re-planning actions. It also comprises the managerial activities of interacting with the external environment, in particular with the Client and sub-contractors.

The management process is divided into two main levels: (i) operational, and (ii) strategic. Both levels comprise the key functions of (re)planning and monitoring. At the strategic level, management assumes a high level view of the project and is typically concerned with the long term directions for the project. In contrast, at the operational level their focus is on detailed tasks and shorter-term issues directly related with implementation.

The SYDPIM principle is that SD project models can be used at both strategic and operational levels within the management process. An SD model to be used at the strategic level will be more aggregated and can be calibrated for the full life-cycle from the beginning of the project. This can be, for example, a single phase project model (e.g. Abdel-Hamid and Madnick 1991). The model will require few data to be calibrated to specific project scenarios. It cannot however be used to analyse the individual life-cycle phases of the project in detail. On the other hand, an SD model to be used at the operational level will assume a more dis-aggregated level of detail. This is likely to be a multi-phase project model which breaks down the project into several inter-related phases (e.g. Cooper 1980; Ford 1995, Williams et al 1995). Such model will require more detailed data to be available in order to be calibrated for specific scenarios. It is also likely that the calibration of this model will be less valid for the later stages of the project. SYDPIM assumes that managers can either use only one of these models, or both. In the later case, the two models will also be formally integrated.

In both cases, at the strategic and operational levels, the model is used in two different ways: (1) to support the monitoring function and (2) the planning function. In planning an SD model performs the roles of an estimating and risk analysis tool, working as a test-bed to anticipate possible futures and develop contingency plans. In monitoring, the model performs the role of a diagnosis and performance evaluation tool. It is primarily used to incorporate, explain, and provide extra information about the project past, and can also be used for retrospective analysis and assessment of past performance.

The SYDPIM deliverables – what does it provide to help project managers?

Although SYDPIM is a flexible process methodology, it provides project managers with well defined specific deliverables. Any project manager considering the use of the methodology will ask the following question: what does SYDPIM provide and what changes will it bring to the project management process?

The SYDPIM methodology provides three main deliverables:

(1) *an new enhanced framework for project management* – the SYDPIM framework is primarily based on the use of dynamic simulation project models. This provides a "virtual reality" environment built upon the traditional framework. In this way, the use of SYDPIM induces a shift in the project management philosophy towards a more *pro-active* and realistic paradigm. SYDPIM provides a clearly defined and structured way of managing projects;

(2) *a formal method to develop and validate SD project models* – the practical use of SYDPIM requires the availability of an SD project model. SYDPIM provides a formal and well structured method that allows the development of high quality SD project models tailored for the specific characteristics of the organisation. It should be noted that the lack of such a formal method is a major gap in the "still young" SD modelling methodology. This is also a critical requirement of SYDPIM, since the use of a poor quality project model will be more likely to misguide the management process rather than improving it;

(3) *a formal method to link SD project models with a PERT/CPM networks* – the formal implementation of the SYDPIM process requires that an SD project model is formally linked to PERT/CPM logical networks. This implies structural and data relationships, as well as data exchange. The SYDPIM provides a step-by-step structured method to specify and implement these links.

The SYDPIM process framework was just summarised. In the next section, the SYDPIM model development method is described. In the remaining sections, the SYDPIM framework is described in more detail together with a real-life example.

The SYDPIM model development method

How can project managers develop the right SD models for their project? The availability of a good quality project model is essential to the successful implementation of SYDPIM. If such a model is already available, a project manager may implement SYDPIM without having to implement the model development method. Likewise, an organisation may decide to use the SYDPIM model development method with the purpose of using an SD project model for various purposes other than implementing the SYDPIM management method.

The SYDPIM model development method is based on a structured life-cycle development method. This comprises a well defined sequence of phases with specific milestones and deliverables, as well as continuous SD modelling activities (both quantitative and qualitative).

Model validation is a critical issue in the method. The SYDPIM life-cycle comprises a final model validation phase as well as a continuous validation activity throughout. This activity is specified in the form of a set of specific confidence tests to be undertaken at each step within the development process.

The SYDPIM method comprises two main phases of: (1) model design and (2) model implementation and validation. Each of these phases has three main sub-phases as follows:

(A) Model design
- (A.1) Causal analysis
- (A.2) Requirements definition
- (A.3) Formal Design

(B) Model implementation and validation
- (B.1) Component development and testing
- (B.2) Model integration and testing
- (B.3) Final model validation

The design phase includes an important sub-phase of formal design, where the model structure is specified in the necessary detail. The implementation and validation phase is based on the development and tailoring of generic SD sub-structures. It includes a final sub-phase of formal model validation, where the fully integrated model is subjected to behaviour accuracy confidence tests. A detailed descriptopn will be available in Rodrigues (1999).

Practical Implementation of SYDPIM

Let us assume that a project manager has followed the SYDPIM model development, method and has now a good quality SD project model available to implement the SYDPIM methodology. How should the model be employed to better manage a project?

The SYDPIM methodology is essentially an *integrated* project management process, which *embeds* an SD project model within the traditional framework. This is unlike past practical applications, where SD models have been applied in total isolation and as an alternative to the traditional approach (e.g. Cooper 1993, states "... the traditional PERT/CPM and *Earned Value* techniques are a complete failure...").

It is therefore appropriate to describe first the traditional framework of project control. This is shown in figure 1a below. The process comprises the two main management functions of project control: (re-)planning and monitoring (note that the first is here considered as a wide concept, including “change management”), and the implementation process of doing the work in the field. In planning, a PERT/CPM network plan is developed and updated, specifying the work tasks, schedules, budgets and resource allocation. This plan is typically adjusted so that the key milestones are achieved on time and budget, with the available resources. The next step is to implement the plan and... wait to see what will happen!...

The work progress is periodically assessed by the monitoring function. Typically, some form of a metrics report is produced, assessing progress on a quantitative basis (e.g. ACWP, CPI, SPI, *Earned Value*). Based on this numerical assessment, a progress report for project control is compiled identifying the critical deviations from the planned targets, and suggesting corrective actions to be implemented in (re-)planning.

Overall, this is clearly a reactive approach to project control. Managers wait for the problems to emerge (and indeed for these to become *perceived*!), and only then corrective actions are considered. This type of traditional management lacks a more strategic and pro-active posture. Unfortunately, it dominates current project management practice and is commonly referred to as “fire-fighting” (Turner 1993).

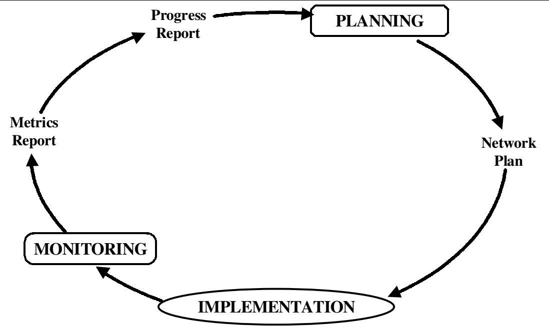

Fig 1a The traditional framework of *reactive* project control

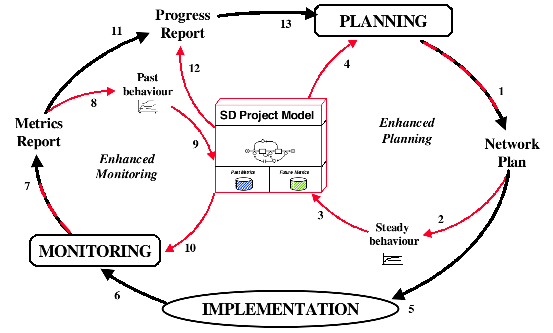

Fig 1b The SYDPIM framework of *pro-active* control

Although based on the traditional process of project control, the SYDPIM methodology takes the opposite view of *pro-actively* anticipating problems and risks, *prior* to implementation. The SYDPIM control process is illustrated in figure 2b above.

In planning, a PERT/CPM network plan is developed as in the traditional framework based on critical path analysis (step 1). Now, instead of implementing the plan “hoping it will go well”, the SD project model is used as a laboratory, or test-bed, to anticipate the likely outcome of the plan. For this, the *steady* project behaviour portrayed by the plan (Rodrigues and Williams 1997) is extracted from the network (step 2), and the SD model is first calibrated to reproduce this plan (step 3). At this stage, the SD model provides various benefits:

(a) *sanity-checking the plan – is it realistic?* Assumptions are uncovered (e.g. implied productivity and defect generation rate) and checked whether they are achievable as compared to observed and likely performance;

(b) *identifying and checking sensitivity against risks – is it robust?* Likely risks are identified and introduced in the model. The impacts on the project outcome are assessed;
(c) *improving the plan – what are the best alternatives?* Alternative planning decisions are tested, and the plan is made more realistic, more robust against risks, and with better performance. These decisions may include work re-scheduling and resource re-allocation, changing the product development process and control policies (e.g. staffing policies).

Once the plan has been tested and improved within the SD model, the network plan is updated accordingly. The SYDPIM process of enhanced planning is similar to the traditional view, except that experimentation is carried out in the model instead of in the real world! Obviously, a pre-tested plan is more likely to be implemented successfully.

In monitoring, data is collected from the project regarding actual results (step 6), and this is compiled into a SYDPIM metrics report (step 7). The project past behaviour is derived from these metrics (step 8), and the SD model is calibrated to reproduce this behaviour. At this stage, the SD model provides various benefits:
(a) *assess and evaluate performance – how well is the project doing against the targets?* During the re-calibration process, actual performance metrics are uncovered and can be compared against the ones assumed in the plan;
(b) *enhanced status information – what is the unperceived progress?* The SD project model has the ability to uncover other status information not measurable using traditional monitoring techniques. A good example is the current amount of undiscovered defects in the system;
(c) *process improvement – what are the lessons learned so far?* The model can be used for retrospective "what-if" analysis to check whether better (or worse) results could have been achieved, and how;
(d) *forecasting a new future – given this past and current plan, where is the project likely to go?* With the model calibrated to the recent past, a forecast of the likely future is produced, if the current plan is kept as is... Note that although a PERT/CPM model also provides this new forecast, the information incorportaed b the SD model regarding past results and current status is much richer (see Rodrigues and Bowers 1996, for a comparison).

The process of adjusting one type of model to the other is central to the SYDPIM project control framework. This includes calibrating the SD project model to the project behaviour portrayed by the PERT/CPM plan (past and/or future), and likewise readjusting the PERT/CPM model to a new project plan developed in the SD project model. This process is based on the quantitative links established between the two types of models, as defined in SYDPIM. The nature of these links is briefly discussed later in this paper (a more exhaustive formal description can be found in Rodrigues 1999). The concepts of "steady" and "unsteady" behaviours are also at the core of the formal integration of the two models.

SYDPIM Integration of SD project models and PERT/CPM network models

The project manager now has a good SD project model and a well defined formal process to use it within the traditional framework. The project manager also understands that, within this process, the two models are formally linked and are readjusted to each other on the basis of reproducing (SD model) and portraying (PERT/CPM) common past and planned behaviours.

However, when it comes to actually perform the operations of model readjustment, the project manager is confronted with the following questions: (1) how is data exchanged between the two models, and (2) how do the two models link in terms of structure and data? The answer to these questions is critical to the quantitative implementation of SYDPIM. The formal integration of SD project models and PERT/CPM network models was the subject of extensive research by the author over the course of several years of research and practice (Rodrigues 1994, Rodrigues and Bowers 1996, Rodrigues and Williams 1997, Rodrigues 1997, Rodrigues and Williams 1998, and Rodrigues 1999) (a detailed explanation of this subject is presented in Rodrigues 1999). A brief summary is now presented.

There are two major aspects of model integration in SYDPIM: (1) structure, and (2) data. This will lead to three types of formal links in SYDPIM:

- (i) data links,
- (ii) structural links,
- (iii) data-structural links.

The establishment of these links is a complex issue (a detailed discussion can be found in Rodrigues 1999). For the sake of simplicity, let us consider only some illustrative examples.

The detailed tasks in a PERT/CPM plan (in some projects there can be thousands!) are grouped and mapped into higher-level SD-tasks (in practice, generally not much more than 10-20). So, any detailed task in the network is captured in only one SD-task in the SD model. This mapping originates "clusters" of tasks in the network plan, with each "cluster" corresponding to a single SD-task (as shown in figure 2). This is an example of a "work breakdown structural link".

Another example of a structural links are the dynamic dependencies between the SD-tasks in the SD model (e.g. see Ford 1995 for a discussion). These can be derived from the elementary precedence relationships in the PERT/CPM model, or at least must be consistent with these. For example, in figure 2 the dynamic relationship between the SD-tasks C and D should be derived from (or consistent with) the two precedence relationships that cross the two correspondent "clusters" of tasks in the PERT/CPM model. This is an example of a "work dependency structural link".

Examples of data links are: (1) the profile of resources allocated to a group (or cluster) of tasks in the PERT/CPM plan (an output of the model), must match the profile specified for the SD-task in the SD model (an input to the model). This could be an "output-input data exchange link"; (2) the cumulative effort spend by the resources in the group of tasks in the PERT/CPM plan (an output of the model), must match the pattern produced by the correspondent SD-task in the SD model (an output of the model). This could be an "output-output data consistency link".

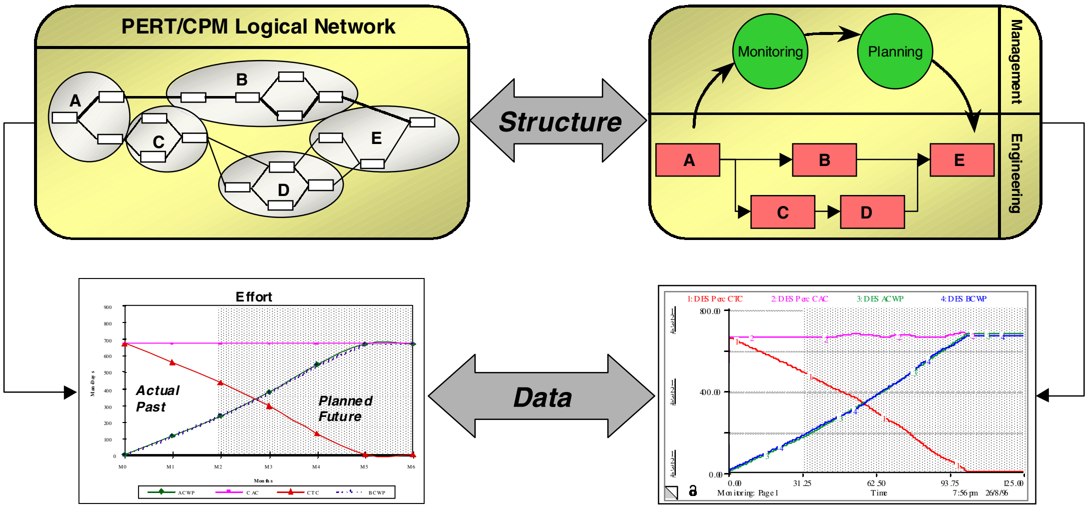

Figure 2 – An illustrative example of SYDPIM model integration

A real-life example: assessing the use of Fagan Inspections in Software Design

A real life example of how the SYDPIM methodology was used to support an important project management decision is now presented. This refers to the possible use of the Fagan inspections technique in software design.

In our real world project, the system integration phase was complex incorporating some components with new code, others with slightly re-used code and also heavily re-used components. The final delivery of the system to the Client was subjected to rigid milestones and high penalties for delays. If critical architectural problems emerged in the integration phase it could be too late. It was therefore critical to ensure the required quality of the designs. But how?

While traditional PERT/CPM tools could be used to try devising alternative plans, there were difficulties with this option: it was too time-consuming, the detailed estimates for the PERT/CPM tasks for each possible solution were not available and so "best guesses" would have to be used); the outcomes would be more imposed by these guesses than by the systemic implications of new decisions. Adding to this, quick "what-if" analysis would be difficult to implement with the gigantic PERT/CPM networks in use.

A possible solution to the problem was to implement better design reviews, where informal design reviews would be replaced by the more thorough Fagan inspections technique. The idea was attractive but there were some potential complications: the Fagan technique is highly effort-consuming and with tight time-scales, would the benefits outweigh the costs? And if so, how much more time should be given to design? What other adjustments should be made to the plan?

As mentioned, the PERT/CPM networks in use would not support the required analysis effectively because most of direct implications of using Fagan inspections could not be captured in the PERT/CPM tool explicitly, quickly, and in a way easy to understand. On the other hand, the indirect implications were of systemic nature and again these are not captured by PERT/CPM analysis.

SYPIM was used to address this issue. According to the generic structure of the methodology described above, the following steps were outlined:

(1) the SD model was prepared to capture the use of Fagan inspections: data collection and interviews were conducted;
(2) the planned behaviour was extracted from the current "non-Fagan" PERT/CPM network plan;
(3) the SD model was calibrated to reproduce this behaviour;
(4) the Fagan inspections were then tested in the model, imposed on the current plan;
(5) alternative planning Fagan-scenarios were tested to search for an optimal outcome;
(6) the benefits of Fagan inspections were compiled based on cost-schedule-quality trade-offs;
(7) depending on the decision, the PERT/CPM network would readjusted to reflect the with new Fagan plan produced in the SD model.

A detailed description of how these steps were implemented is available in Rodrigues (1999). A brief summary is presented here.

The first step was based on literature research, data collection and interviews with managers. This allowed the author to identify the critical direct impacts of using Fagan inspections. These direct impacts were then quantified and various factors were incorporated in the model. For example, defect detection was set to be 80% more efficient due to the reduced overlap of defects detected by different reviewers, and due to links to parent and related documents. Verification and validation tests were run to check the model's sensitivity to the new changes, before actually using it. The next steps were to extract the planned behaviour from the current "non-Fagan" PERT/CPM plan and calibrate the SD model to reproduce this plan. This is shown in figure 3 below.

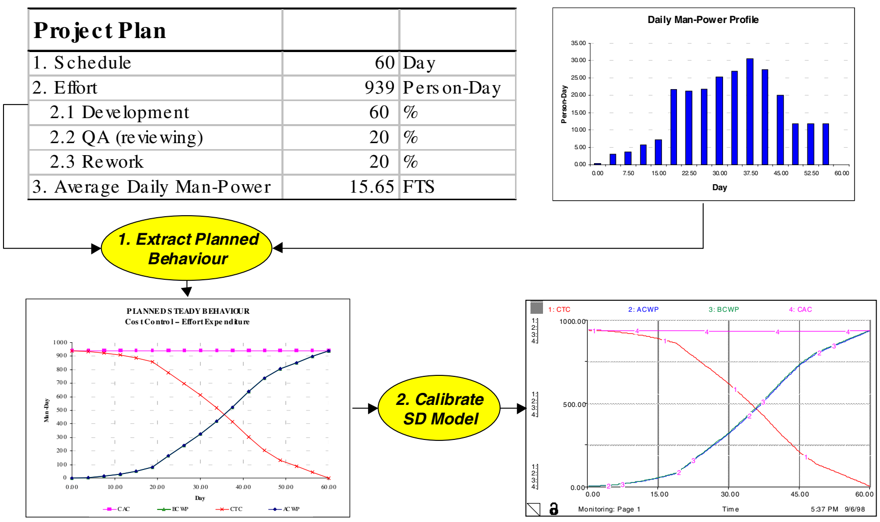

Figure 3 – Calibrating the SD model for the current plan

Then, inside the SD model various Fagan and "non-Fagan" scenarios were considered and tested in a safe-environment. Would Fagan help to improve the quality of the designs? By how much? And how much would that cost? What are the trade-offs? What would be the best way of accommodating Fagan inspections?

To answer these questions, a multidimensional analysis was carried out considering the following combinations:

(A) Which reviewing technique?
  (1) Conventional
  (2) Fagan inspections

(B) How should the design phase be re-scheduled?
  (1) 60 days with no slippage allowed (as planned)
  (2) 60 days with slippage allowed
  (3) 70 days with slippage allowed
  (4) 80 days with no slippage allowed

(C) How to readjust the QA planned budget?
  (1) Low (15%)
  (2) Medium (20%)
  (3) High (30%)
  (4) Very high (40%)

A total of 32 scenarios were tested through simulation.

The first main conclusion from the analysis was that Fagan inspections would provide significant quality benefits, and more efficiently than increasing the QA level with the conventional reviews. The numerical results were compiled. Five main scenarios are presented here:

(1) keep the plan as is and implement Fagan;
(2) as an alternative to Fagan, increase the QA level from 20% to 40% using the conventional reviewing techniques;
(3) try Fagan with the current plan but allowing schedule slippage upon staff request (i.e. low schedule pressure);
(4) try Fagan and extend the design schedule 20 days more. No slippage allowed (i.e. full schedule pressure);
(5) try 40% of QA level with the conventional techniques and extend the design schedule 20 days more. No slippage allowed (i.e. full schedule pressure);

The results from these five scenarios are shown in figure 4a below – note that Fagan scenarios are in blue with the strips to the right. Non-Fagan are in red and with the strips to the left. The results are presented as a ratio against the base-case (i.e. non-Fagan current plan).

The results show that Fagan offers significant quality benefits and, under the same circumstances, performs better than increasing the QA level with conventional techniques.

The second main conclusion was that with Fagan inspections there was more scope to improve quality by giving design more time and further increase the QA level – but this would also be at a higher cost. Figure 4b below shows the cost-schedule-quality trade-off of implementing Fagan with an increasing amount of time and effort allocated to QA activities.

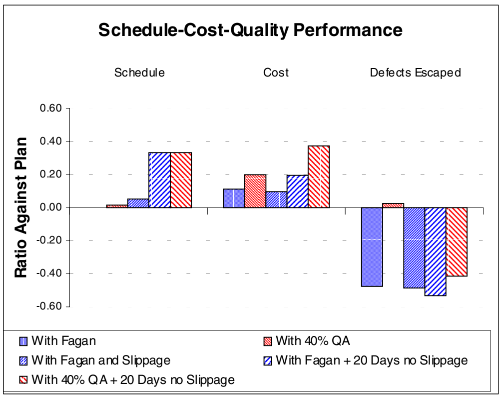

Fig 4a – Comparing Fagan with non-Fagan scenarios

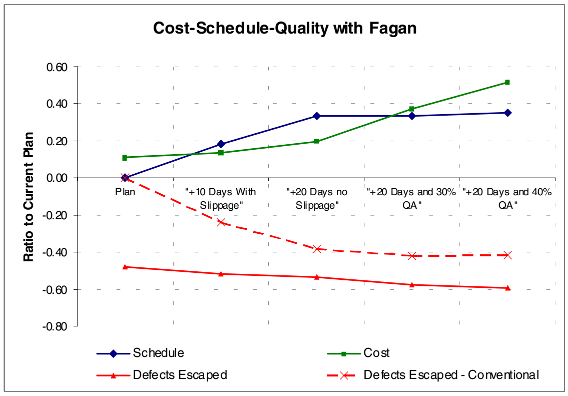

Fig 4b – Assessing various Fagan scenarios

The dashed red line (defects escaped) refers to the use of conventional techniques under similar conditions. By comparison with the thick red line of Fagan (also defects escaped), it can be seen that the performance of the conventional techniques reaches a "plateau" after which increasing more the QA effort brings little extra quality benefits. The same is not true with Fagan inspections, where quality continues to improve.

Once the "optimal" Fagan scenario was identified, the next step was to translate the plan into the PERT/CPM model, according to the analytical links established between the two models, using the SYDPIM methodology. For reasons of confidentiality the actual decision is not identified here. The figure below considers an hypothetical scenario to illustrate how this SYDPIM step is carried out.

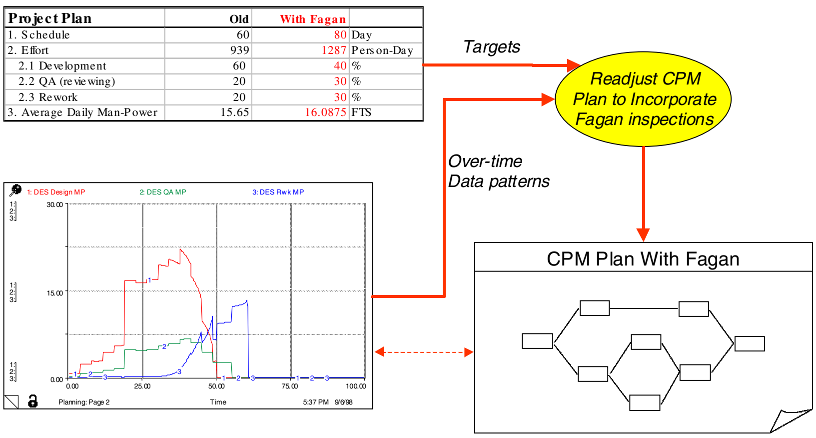

Figure 5 – Transferring the Fagan plan from the SD model to the PERT/CPM model

Conclusions and future work

From the author's experience with real-life examples, the SYDPIM methodology can provide significant benefits worth the investment of implementation. The three main benefits can be summarised as follows:

(1) Provides an enhanced Project Management process framework:

In planning:

- unrealistic planning assumptions can be uncovered;
- anticipates impacts of risks and re-planning prior to implementation;
- more realistic forecasts due to the model conceptual richness;

In monitoring:

- causes of current outcome can be diagnosed and analysed;
- improved visibility of current project status.

(2) Embeds an SD project model within the traditional framework through SD-PERT/ CPM quantitative links;
(3) Provides a formal method to develop and validate SD project models.

While useful in practice, these benefits are also important research contributions to both fields of Project Management and System Dynamics. A full presentation of this work will be described in the author's PhD thesis, forthcoming this year.

The author will continue to pursue the research and practice of SYDPIM. The automation of the SYDPIM formal process through the use of software tools is one of the most critical issues, since it reduces significantly the cost of implementing the methodology. This also applies to the SYDPIM model development method. Further practical opportunities to implement SYDPIM will be sought by the author.

References

Abdel-Hamid, T. and Madnick, S. (1991) Software Project Dynamics: An Integrated Approach. New Jersey, Prentice-Hall.

Burdick, B. et al (1998). The Impact of Software Project Management on Quality. Cutter IT Journal. 11(9), 30-38.

Cooper, K. (1993). The Rework Cycle: Why Projects Are Mismanaged. PMNETwork. February.

Cooper, K. (1980). Naval Ship Production: A Claim Settled and a Framework Build. INTERFACES. 10(6), 30-36.

Ford, D. (1995). The Dynamics of Project Management. Cambridge Mass, The M.I.T. Press.

Forrester, J. (1961) Industrial Dynamics. Cambridge Mass, The M.I.T. Press.

Morris, P. Hough, G. (1987). The Anatomy of Major Projects : a study of the reality of Project Management. Chichester: John Wiley & Sons.

Nicholas, J. (1990). Managing Business and Engineering Projects: Concepts and Implementation. Englewood Cliffs, N.J.: Prentice-Hall.

Rodrigues, A. (1999). System Dynamics in Project Management: an Integrated Methodology (SYDPIM). (*PhD dissertation thesis. Forthcoming in 1999).*

Rodrigues, A. and Williams, T. (1998). System dynamics in project management: assessing the impacts of client behaviour on project performance. Journal of the Operational Research Society. 49(1), 2-15.

Rodrigues, A. (1997) SYDPIM - A System Dynamics-based Project-management Integrated Methodology. in 15th International System Dynamics Conference: "Systems Approach to Learning and Education into the 21st Century". Istanbul, Turkey: Bogazici University Printing Office.

Rodrigues, A.G. and Williams, T. (1997), System dynamics in software project management: towards the development of a formal integrated framework. European Journal of Information Systems. 6(1), 51-66.

Rodrigues, A. and Bowers, J. (1996). System dynamics in project management: a comparative analysis with the traditional methods. System Dynamics Review, 1996. 12(2), 121-139.

Rodrigues, A. and Williams, T. (1995). The Application of System Dynamics in Project Management: An Integrated Model With the Traditional Procedures. Working Paper 95/2: Theory Method and Practice Series. Department of Management Science, University of Strathclyde.

Rodrigues, A. (1994). The Role of System Dynamics in Project Management: A Comparitive Analysis with Traditional Models. in 1994 International System Dynamics Conference. Stirling, Scotland: System Dynamics Society.

Turner, J. (1993). The handbook of project-based management. McGraw-Hill, London.

Williams, T. et al (1995). The Effects of Design Changes and Delays on Project Costs. Journal of the Operational Research Society. 46(7), 809-818.