PEAT- a tool to predict team performance in systems

This item was submitted to Loughborough University’s Institutional Repository by the/an author.

Citation: SINCLAIR, M.A....et al., 2010. PEAT- a tool to predict team performance in systems. IN: 5th International Conference on System of Systems Engineering (SoSE), Loughborough, 22-24 June, 7pp.

Additional Information:

- This is a conference paper [© IEEE]. It is also available at: http://ieeexplore.ieee.org/ Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Metadata Record: [https://dspace.lboro.ac.uk/2134/7230](https://dspace.lboro.ac.uk/2134/7230)

Version: Published

Publisher: © IEEE

Please cite the published version.
This item was submitted to Loughborough’s Institutional Repository (https://dspace.lboro.ac.uk/) by the author and is made available under the following Creative Commons Licence conditions.

For the full text of this licence, please go to: http://creativecommons.org/licenses/by-nc-nd/2.5/
Abstract - The paper describes the development of a tool to predict the success of a team in executing a process. It is expected to be used by systems engineers in initial stages of systems design, when concepts are still fluid, including the structure of the team(s) who are expected to be operators within the system. Currently, the tool is undergoing verification and validation; to date, the tool predicts fairly well and shows promise.

Keywords: Systems Engineering, Performance Prediction, Organizational Performance

1 Introduction

The original purpose of the tool was to provide designers of military systems with a simple tool to use in the conceptual stages of design (when variables are still variables and not parameters) to help in risk reduction exercises when considering the staffing of processes. Some sample questions for which the tool could help in providing answers are:

• What is the likelihood that this team will be successful in executing the given process? Given appropriate inputs (see below), the tool produces a likelihood of success.

• By how much can the team size be reduced, before the likelihood of success becomes unacceptable? Stepwise, the team size would be reduced, and for each step the duties of the team would be re-arranged and the tool would be used to recalculate the likelihood of success.

• By how much can the attributes of the individuals in the team be reduced, before the likelihood of success becomes unacceptable? Progressively, the ratings for each team member would be reduced, and for each reduction the tool would be used to recalculate the likelihood of success.

• Which of these two (three, …) proposed teams is likely to be the most successful in executing the process? Each team arrangement would be assessed for success by the tool; either the user could select the 'best' team, or the user could examine the trends in the likelihoods of success, and devise an even better team.

• Can an acceptable likelihood of success be achieved by increasing the quality of the individuals in the team? Progressively, the ratings for each team member would be increased, and for each increase the tool would be used to recalculate the likelihood of success.

• Can an increase in automation of tasks, or a re-arrangement of tasks, in the process bring the team's likelihood of success up to an acceptable level? For each step in the re-ordering or re-classification of tasks, the tool could be used to calculate a likelihood of success.

It has subsequently become evident that these questions could also be developed within the context of Through-Life Capability Management (for systems), with regard to the delivery of Service Level Agreements (guaranteeing performance of the system).

It will also be noted that these questions are phrased in terms of success. This is a significant point; 'success' is defined here as executing the process correctly and attaining all of the goals of the process; no reworking, no extra resources, no extra time. 'Likelihood', as usual, is expressed as a probability. Note that (1 - p(success)) does not mean 'probability of failure'; for many processes, if they show signs of failure, more resources, or more time can be allocated, and the process can be brought to some degree of successful completion.

Note also, that project success does not mean victory; the team might fire the surface-to-air missile successfully,
but it might still miss the target. 'Success' solely refers to the process execution.)

2 Development of the tool

From the outset, three constraints determined the development of the tool:

• A user-defined constraint: If the tool needs more than two pages of A4 to explain it, it will not be used. This was a unanimous view among a target group of engineers who were interviewed at the beginning of the project.

• A business process constraint: Systems designers in the UK are already familiar with, and may be using, techniques such as HEART and CREAM for assessing the reliability of individuals. The tool should incorporate these techniques, or any other in-house technique, to enhance ease of acceptance into design processes.

• A design constraint: At the conceptual stages of design, little will be known about the individuals in the team that executes the process, apart from generic attributes. Equally, the process will be undefined – perhaps just a single flow diagram sketched on a sheet of A4. Hence, the tool must make a minimum demand for input data.

The first constraint has been met, without recourse to minuscule script; a second, longer manual has also been provided to explain not just what to do, but why it needs to be done. The second constraint has been met by creating a three-stage tool; Stage 1 collects data about the attributes of individuals in the team and the intercommunications necessary for execution of the process. Stage 2 collects an analysis of the process environment, using either HEART [1] or CREAM [2], or with an in-house technique substituted for these. Both techniques are used for individuals; HEART asks the user to select a base p(error) for a given task, and then allows the user to modify this by means of Performance Shaping Factors. CREAM asks to user to complete 9 scales, and on the basis of these calculates a p(error). The probabilities of failure that emerge from this analysis are now multiplied by the PSFs for each individual in the team in his/her environment.

Stage 3 happens without necessitating user input. It now combines the values obtained above, the authority ratings, and the communications matrix to arrive firstly at what is called ‘interactive probabilities of error’ for each individual; in other words, acknowledging peer effect. The initial probabilities of error from the paragraph above are now adjusted by these extra variables. The assumption here is that a person’s performance will be influenced by the performances, knowledge, and teamworking capabilities of those who communicate with that person in the execution of the process. This might be feedforward, feedback, or the observable quality of work.

An algebraic equation performs this step, with the property that if a person communicates with very good people, then that person’s probability of error will decrease, and vice versa. It does this by first calculating the Power Distance between a person and a peer, then multiplying this by the peer’s p(error), then combining the resultant for all communicating peers, and then averaging this latter resultant and the person’s p(error).

Hence, we now have an ‘interactive probability of error’ for each team member.

Secondly, the team’s binding to the process is addressed. It was discovered that three classes of binding cover a wide range of real teams. The three classes are described below:

• The ‘aircrew’ team. Consider a helicopter, flying from A to B over hostile territory. The pilot executes the
process of flying; the rest of the crew act only as advisors (e.g. ‘A SAM has been fired’), but do not play a part in flying the helicopter.

- The ‘Boatcrew’ team. Consider a rowing eight (nine, with the coxswain). From start to finish, each person has a specific task, and cannot perform anyone else’s task. Consequently, the absence of any crew member ensures failure.

- The ‘Omnicompetent’ team. Here, anyone can perform anyone else’s task, and may do so in executing the process.

Likelihoods are presented for each of these bindings together with estimated 95% confidence limits for the likelihood; the user may choose the best-fitting example, or may calculate a specific value from the ‘interactive probabilities of error’ using the laws of probability, if the three classes are deemed unsuitable.

3 Current state of the tool

Due to the adoption of a form of Rapid Prototyping as the development methodology, the tool is now in EXCEL spreadsheet form, in its ninth revision. Functionally, the tool is considered complete save for further verification and validation, and is being made more presentable and understandable to the intended user. The user manuals have also been tested, and are considered complete. In accordance with the first constraint above, a two-page version and a more detailed version have been developed.

At the moment, the tool is constrained to deal with teams of ten or less members. This is a developmental constraint; it is expected that when the tool is deemed to have been fully tested, this constraint will be removed. With cognizance of this constraint, the tool appears to be able to deal with a number of different scenarios; as shown later in the verifications, the teams considered range from military teams to system development teams, to the Board of Trustees for a pension fund; in other words, teams of many different organisational cultures. It can accept distributed teams, and teams where some members do mainly physical tasks and others do mainly cognitive tasks.

4 Verification and validation of the tool

In other words, ‘Does it work?’ The evidence so far is that it does. A second question is, ‘Is it sensitive enough?’ Again, the indications are that it is. This sensitivity issue is discussed first below, followed by the verification and validation results.

4.1 Sensitivity of the tool

For a team of two, using CREAM to categorise the environment, 1360 values may be obtained between 0.0 and 1.0. For a team of 10, using HEART the range of values is well over 1 billion. There appears to be sufficient sensitivity for most purposes.

4.2 Verification of the tool

To date, over 400 verification tests have been carried out on these teams, changing at least one variable for each test. Three classes of teams have been investigated in some detail. Two are teams of four; the difference between them is in the communication patterns. In a ‘Linear’ team, the members are arranged linearly, with two-way communication between adjacent members (similar to a production line). In a ‘Cocktail’ team the arrangement is that of a star, with a central person communicating with all the others. The others have partial communications between them; the net effect is that each team member has a different number of communication links. The third class has explored different sizes of teams and different communication patterns, including the effect of two cooperating teams.

A further restriction on all these tests is that all the tests have been performed using CREAM to characterise the environment, since it has better theoretical underpinnings than HEART.

From all of these tests, it seems that some general conclusions can be drawn:

1. Given that the tool has been constructed as a simple technique, with no feedback loops and no ‘if-then’ rules, it is gratifying that the behaviour is as expected; trends are consistent, and no test produced a prediction outside the range 0.0 – 1.0.

2. It is striking that in all sets of tests, a well-designed working environment (i.e. well-designed process, good HMI, proper training, good support, leadership, etc.) is the biggest contributor to p(success). The other variables in the tool become more important as the working environment degrades, but they cannot make up for it. In a good working environment, team variables are relatively unimportant; even when the team is of poor quality, p(success) > 0.9. For a military environment, this is equivalent to saying that if all the US DoD Systems Engineering requirements have been met completely and in full (or in the UK, the Defence Lines of Development have delivered in full), taking full account of human factors integration, then successful performance of the process is assured (note that this is not saying, ‘Victory is assured’ – see earlier comment).

3. In poor working environments, the most important variable is the quality of the team members. As long as there are several high quality members in central roles, able to communicate with the other members, then a level of performance (p(success) ≥ 0.25) can be achieved. However, even with a high quality team, it is not possible to lift p(success) above 0.35.
4. A good team is always better than 4 individuals.

Of course, the truth content of these statements depends on the validation studies below, and more of these must be carried out. But, insofar as there is truth in these statements, findings 2 and 3 together have a significant corollary; they provide a strong argument for the importance of Human Factors/ Ergonomics in systems engineering projects. One might imagine that, given the extent and complexity of military systems being sourced for the NATO forces for future decades (particularly the USA), and the likely manning issues for these, these conclusions will be of some interest to those in military procurement, and to those in civilian safety-critical systems management.

4.3 Validation of the tool.

To date, 18 validation exercises have been carried out, summarised in Table 1. All are historical cases, with the results known; i.e. validation-by-criterion. However, in all cases but the last the criterion was subjective, since the processes were not repeated. The last was a repetitive process, allowing comparison of probabilities of success. Extremely brief descriptions of the cases are included for reasons of space and confidentiality, together with the predicted success and the source’s verbatim comment on the result. Note that in all cases except for two, the respondent in each of the tests was a Systems Engineer.

<table>
<thead>
<tr>
<th>#</th>
<th>Team process</th>
<th>Prob success</th>
<th>Comment on result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Insert new FCS in tank</td>
<td>0.76</td>
<td>“That looks good.”</td>
</tr>
<tr>
<td>2</td>
<td>Dev’t of control system for UAV</td>
<td>0.59</td>
<td>“OK, if a little generous”</td>
</tr>
<tr>
<td>3</td>
<td>Dev’t of systems Health Mgmt System</td>
<td>0.87</td>
<td>“I’m happy with that result - perhaps a little bit high”</td>
</tr>
<tr>
<td>4</td>
<td>Create engineering Technical Demonstrator</td>
<td>0.92</td>
<td>“Result is OK; perhaps a bit high”</td>
</tr>
<tr>
<td>5</td>
<td>Dev’t of comms system for navy ship</td>
<td>0.95</td>
<td>“Rings reasonably true”</td>
</tr>
<tr>
<td>6</td>
<td>Bid prep’n for US DoD ITT</td>
<td>0.45</td>
<td>“Result looks OK; wouldn’t want to argue with it; perhaps a little bit high.”</td>
</tr>
<tr>
<td>7</td>
<td>Execution of a Des &amp; Build project for M.Eng degree</td>
<td>0.49</td>
<td>“Result is a bit low”</td>
</tr>
<tr>
<td>8</td>
<td>Deliver HFI to manufacturing Technical Demonstrator</td>
<td>0.92</td>
<td>“OK; but doesn’t account for a weak team member. Got it together because of the efforts of the rest of the team”</td>
</tr>
<tr>
<td>9</td>
<td>Dev’t of guides for UK Gov’t dept.</td>
<td>0.90</td>
<td>“That’s OK”</td>
</tr>
<tr>
<td>10</td>
<td>Software development for UK NHS</td>
<td>0.60</td>
<td>“Estimate is a bit low; would have expected about 0.75”</td>
</tr>
<tr>
<td>11</td>
<td>Development of a UAV ground station</td>
<td>0.48</td>
<td>“A bit low - would have expected around 0.7”</td>
</tr>
<tr>
<td>12</td>
<td>Management team in University</td>
<td>0.83</td>
<td>“Result is OK, but this isn’t a normal team”</td>
</tr>
<tr>
<td>13</td>
<td>Pension Bd of Trustees managing fund</td>
<td>0.99</td>
<td>“That’s about right”</td>
</tr>
</tbody>
</table>
A further, Binomial test [3] was executed on those estimates deemed above and below, to test for a bias in the predictions. Table 3 shows the results of this.

According to Table D of Siegel [3], this indicates that the Null hypothesis (no bias in predictions) is not rejected (p > 0.05).

These tests indicate that the tool can produce reasonable predictions, and therefore, the inferences drawn from the verification tests are trustworthy. However, it has been arranged that three further independent, detailed validation tests will be carried out by industry. One is concerned with a new weapon system for a helicopter, another is evaluating a new mine-hunting approach, and the third is still being worked up. All are due to report later this year. It is hoped that these will corroborate the findings so far.

5 Utilisation of PEAT

Three other uses for PEAT can be envisaged, utilising its capability to predict team performance (subject to further validation). These are:

- **Through-life capability management.** In order to meet service level agreements for delivered capability, it is already the case that maintenance teams will have to be created at relatively short notice to deal with some unexpected capability readiness problem. PEAT may be useful in ensuring a suitable team can be created.

- **Project cost estimation.** Tools such as COCOMO and COSYSMO make a number of not-always-explicit assumptions about the personnel involved in delivering a project. The inclusion of PEAT in these estimation techniques may help to improve these cost estimates by providing a more precise estimate of the performance of project personnel.

- **System health monitoring.** Current techniques concentrate heavily on the technology of the system, and tend to omit much consideration of the people actively present within the system. By including PEAT in these techniques it may be possible to provide a full, socio-technical assessment of the state of health of a system, not just a technological assessment.

6 Conclusions

In recognition of the fact that a further three independent tests of the tool are under way by industry
prior to its acceptance, conservative conclusions are in order

• A tool with some predictive power has been produced. However, it is not entirely clear how powerful the tool is, nor how extensive is its range of application

• It is believed to be the only tool able to provide estimates of team performance available for designers and human factors experts to use in the early stages of design.

• Planned developments to the user interface should make the tool usable by non-experts in human factors. This is believed to be an asset; if the engineer’s own use of the tool shows that human factors issues must be addressed, this is likely to be a convincing argument.

• That the tool is in EXCEL spreadsheet form is good, but insufficient; it is necessary that a web-based version is available. This means that a more acceptable user interface could be given to the tool, and it can have a management process associated with it, both to maintain its integrity, to capture lessons from its use, and to develop the tool within the organisation hosting it. Some plans exist for this.

7 Acknowledgements

This tool has been developed within the ambience of an Engineering Doctorate. Thanks are therefore due to the Engineering and Physical Sciences Research Board and the sponsor, BAE Systems, without whose funding this would not have happened. Many, many others have provided insight and knowledge to this exercise; those most directly involved in this are Dr Siemieniuch, Prof Henshaw, and Prof. Haslam, all of Loughborough University, and Mr. Evans of BAE Systems. Their contributions are saluted.

8 References


