Software Project Management Tools: Making a Practical Decision Using AHP

Norita Ahmad
Phillip A. Laplante
Penn State University
{nua2|plaplante}@psu.edu

Abstract

The selection of an appropriate software project management tool, as in the selection of many other tools, is often approached using an ad hoc process. Such non-rigorous approaches often based on personal preference, intuition, or marketing hype, can lead to an erroneous result. In this paper a rigorous model for selecting a software project management tool using the Analytical Hierarchy Process (AHP) is presented. The AHP provides a flexible, systematic, and repeatable evaluation procedure that can easily be understood by the decision maker in selecting the appropriate software project management tool. Several relevant factors based on the most common features offered by commercial off-the-shelf solutions (COTS) are used as the selection criteria in ranking the software project management tools. The contribution of this work is to apply a well-known decision making procedure in a novel way to help decision makers better identify an appropriate software project management tool without having to go through a more extensive evaluation process. In addition, this work establishes a framework for comparing individual product decisions across projects, project managers, organizational groups, and organizations.

1. Introduction

The evolution of project management tools for both software and non-software applications has been accelerating at a rapid pace and the number of available products on the market has grown significantly. These product choices are accompanied by a dizzying set of product features leading to software project management tools that are available at many different levels of sophistication and prices. Because the available feature set is so rich, and the price range so wide, it is important for the project owners to choose the most appropriate tool for their project.

Project managers have used software tools to automate the administration of individual projects or small groups of projects for years. What is new, however, is that these tools now incorporate such features as risk assessment, portfolio management, best practices, email notification, collaboration, and many more, which have broader, enterprise-level impact. While it is often assumed that any commercial software project management tools will perform basic functions (and hence, a choice based on price alone is made), prospective buyers need to perform a careful selection analysis to accurately evaluate the feature sets of the many available tools on the market.

Given the current upsurge in interest in software project management tools, surprisingly limited work has been done in this area and little attention has been paid to the evaluation of the selection criteria. Most of the previous empirical research and investigation in software project management focuses on broad project situations such as construction [1],[3], testing [2], [5], risks [4], [6], or outcomes [7], [8], [11]. We could find no relevant work on software project management tool selection.

Therefore, it is appropriate to bring to bear a recognized multi-criterion, decision making approach. The Analytical Hierarchy Process (AHP) is a well-known decision theory model developed by Saaty [9]. Its primary attribute is quantifying relative priorities for a given set of alternatives on a ratio scale, based on the judgment of the decision-maker. It provides an easy way to incorporate multiple experts' opinions and control of consistency in judgments. In addition, the AHP method ensures high repeatability and scalability controls.

The objective of this paper is to introduce the application of the AHP model to the selection of tools, in particular in selecting an appropriate software project management tool. The paper will briefly review the concepts and application of the AHP method and demonstrate how the AHP model can help the decision maker to better identify an appropriate software project management tool without having to go through an extensive evaluation process.

Proceedings of the 30th Annual IEEE/NASA Software Engineering Workshop SEW-30 (SEW'06)
0-7695-2624-1/06 $20.00 © 2006 IEEE

2. The Analytical Hierarchy Process (AHP)

The Analytical Hierarchy Process (AHP) is a powerful and flexible multi-criteria decision-making method that has been applied to solve unstructured problems in a variety of decision-making situations, ranging from the simple personal decisions to the complex capital intensive decisions. AHP differs significantly from other decision making approaches used in software engineering such as Quality Function Deployment (QFD) and Goal-Question-Metric (GQM). QFD is used to represent the “voice of the customer” in the design of new products in a competitive market. GQM is an approach for designing metrics programs. AHP differs in that it is best applied in a situation where structuring, measurement, and/or synthesis are required. Some areas in which the AHP has been successfully employed include resource allocation, forecasting, total quality management, business process re-engineering, quality function deployment, and the balanced scorecard [10]. Surprisingly, little has been written on the application of AHP to software engineering decision making problems. To show how AHP is well suited for such problems is the main contribution of this work.

The application of the AHP model is carried out in three stages: (1) structuring complexity or decomposition, (2) measuring on a ratio scale or comparative judgments and (3) synthesizing. In AHP, the decision maker first structures the problem elements into a hierarchy. Once the hierarchical model of the decision process has been built, the decision maker completes a comparison matrix at each level by comparing pairs of criteria, or pairs of alternatives, starting with the lowest level and working upwards. In pairwise comparisons, two elements of the same level are compared at any given time to a particular element at a higher level and a ratio of relative importance is assigned to each paired comparison, usually according to the Saaty linear nine-point scale*, {1/9, 1/8, 1/7, 1/6, 1/5/, 1/ 4, 1/3, 1/2, 1, 2, 3, 4, 5, 6, 7, 8, 9}*. By starting at the lowest level, the expert gains familiarity with the details of the higher level decision attributes before making those higher level paired comparisons.

After all matrices are developed and all pairwise comparisons are obtained, eigenvectors or the relative weights (the degree of relative importance amongst the elements), global weights, and the maximum eigenvalue (λ_{max}) for each matrix are then calculated. The λ_{max} value is used as a reference index to screen information by calculating the consistency ratio of the estimated vector in order to determine if the pairwise comparison matrix provides a completely consistent evaluation. The consistency ratio is calculated as follows:

1) Calculate the eigenvector or the relative weights and λ_{max} for each matrix of order *n.*

2) Compute the consistency index (CI) for each matrix of order *n* by the formulae:

$CI = (\lambda_{max} -n)/(n-1)$

3) The consistency ratio (CR) is then calculated using the formulae:

$CR = CI/RI$

Where RI is a known random consistency index obtained from a large number of simulations run and varies depending upon the order of matrix.

The acceptable CR range varies according to the size of matrix, that is 0.05 for a 3 by 3 matrix, 0.08 for a 4 by 4 matrix, and 0.1 for all larger matrices, $n >= 5$ [9]. If the value of CR is equal to or less than the acceptable value, it indicates a good level of consistency in the comparative judgments represented in that matrix. In contrast, if CR is more than the acceptable value, inconsistency of judgments within that matrix has occurred, and the evaluation process should therefore be reviewed, reconsidered, and improved. An acceptable consistency property helps to ensure decision-maker reliability in determining the priorities of a set of criteria [9].

3. Identification of Software project management tool selection criteria

We loosely define software project management as those techniques used to identify, measure, and control various aspects of a software project throughout its life cycle. Many different software project management tools are used by managers to manage and support these activities by providing a means for planning, executing and controlling the project.

It is true that most commercial project management tools will perform the basic functions of software project management such as scheduling and resource management. Hence, a choice based on price alone is made. But prospective buyers need to perform a careful selection analysis to accurately evaluate the feature sets of the many available tools on the market.

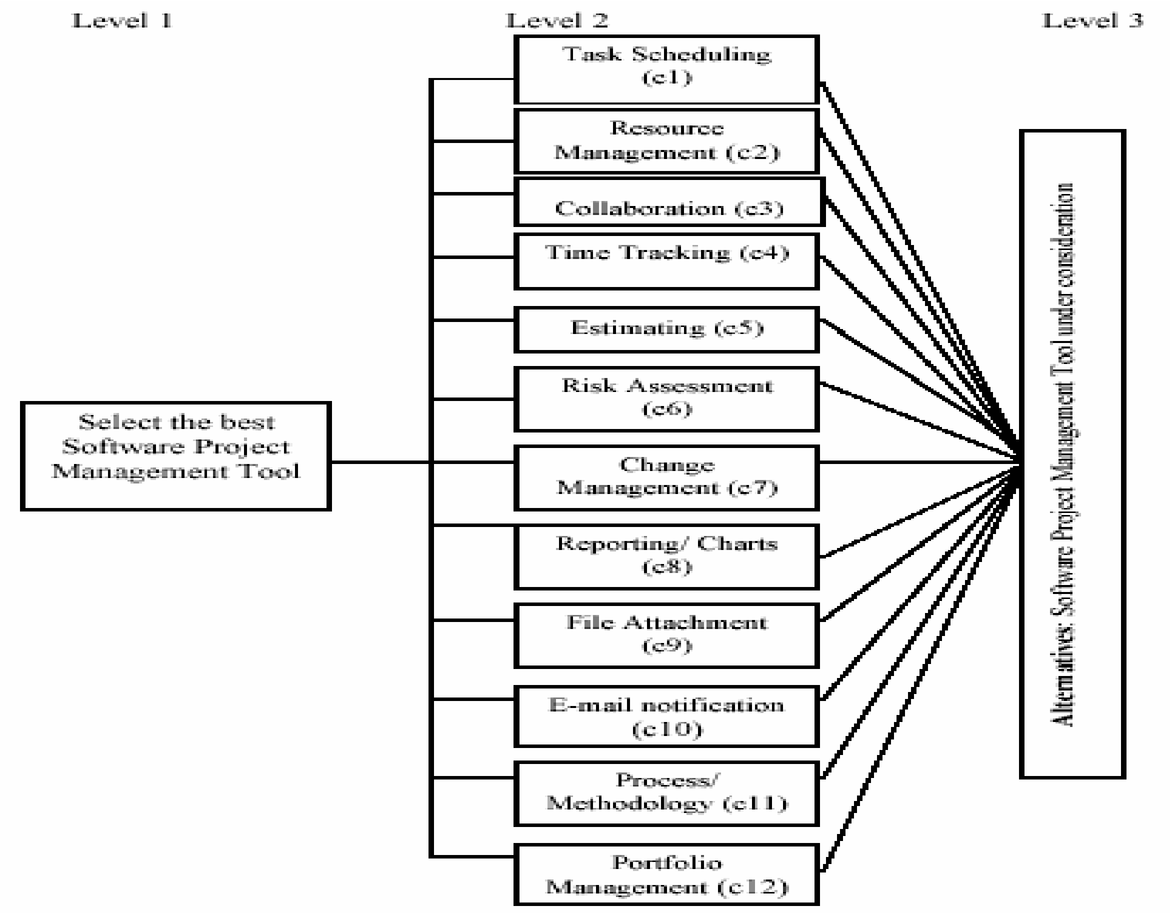
We will use twelve criteria to compare and evaluate software project management tools as shown in Table 1. The criteria we chose are by no means the only ones possible. This set of criteria was selected, however, because it was identified as suitable for making a project management tool selection [12].

4. Tool selection using AHP

In order to apply the AHP model to the selection of a software project management tool, we must state a precise goal for the selection. In this case, the design of the AHP hierarchy must satisfy the goal of developing a model that will allow a software project manager to decide the tool that is more appropriate for a particular project.

Table 1. Software Project Management tool selection criteria

	Criteria	Description
1	Task Scheduling	Task scheduling refers to the assignment of start and end times to a set of tasks. This feature lets software project manager track important project milestones and note who is responsible for each task.
2	Resource Management	This feature lets the software project manager organize and trace requirement details to ensure that proper resources are committed to the project. The software project manager can establish information relationships between multiple documents, assign attributes to the information, such as task assignment, priority and status, and change these over time to reflect changes in the project.
3	Collaboration	Collaboration enables both structured and free-flow sharing of knowledge and best practice. It includes project status reports that are accessible via a Web page, integrated e-mail or discussion boards.
4	Time Tracking	Time tracking allows recording, analyzing and reporting associated with project working routine. Software project manager can use time tracking feature to manage employee timesheets and expenses, calculate salaries, prepare project estimates, and get invoices based on personal or client work rates
5	Estimating	The estimate feature allows the project manager to generate, manage, and validate estimates of effort for a wide variety of projects. It evaluates the project plan, project requirements, information about working environment, and even different aspects of company's culture.
6	Risk Assessment	Risk assessment helps the software project manager in identifying and planning for potential project risks. It could also help software project manager to describe the various risk factors and how to score them.
7	Change Management	This feature lets software project manager control schedules, resources, and deliverables of project. It can manage the impact that changes have on project objectives, and it lets software project manager trace changes to see how each requirement's changes affect multiple other requirements.
8	Reporting/ Charts	In addition to Gantt or PERT charts, some tools provide hundreds of charts and reports. In addition some tools allow users to develop a custom report format that suits the organization.
9	File Attachment	File attachment let users customize project tasks, like file sharing and document management systems, and Web page authoring. Some tools help users control document version and checkout as well.
10	E-mail notification	Most software project management tools provide integrated e-mail notification to keep team members informed of the current status such as defects, change in documentation issues and requests, and other related issues.
11	Process/ Methodology	Process/Methodology features allow the software project manager to develop and implement a consistent and standardized process workflow.
12	Portfolio Management	Portfolio Management feature helps the software project manager manage multiple projects that are related, such as infrastructure technologies, desktop applications and so on, and allocate resources accordingly.



Note: Each alternative in level 3 is connected to every criterion in level 2.
Figure 1: Hierarchy Structure for the selection of Software Project Management Tool.

Note that the formulation of the question in this way helps to avoid a “one size fits all” solution. This is important because each project tool may call for a very different set of features and price sensitivity.

The structure of this research model consists of a three level hierarchy as shown in Figure 1. The top level is the goal, which is to identify the best software project management tool for a particular project. The second level is the twelve criteria identified in section 3. The third level is the alternative or the software project management tools to be evaluated.

5. An Example Application of AHP

We will present an example of the AHP methodology applied to simulated data here to illustrate the technique. To simplify calculations, only five different alternatives or software management tools are used. We will identify these tools as T1, T2, T3, T4, and T5.

First, the twelve evaluation factors (labeled c1,…,c12) from the second level of the hierarchy (Figure 1) are compared with each other to determine the relative importance of each factor with regards to the overall goal. The easiest and most structured way of doing this is to prepare a matrix with the factors listed at the top and on the left, as shown in Figure 2. The decision maker then fills the matrix with numerical values denoting the importance of the factor on the left relative to the importance of the factor on the top. A high value means that the factor on the left is relatively more important than the factor at the top. In Figure 2 for example, Task Scheduling (c1) is considered to be eight times as important as Resource Management (c2), whereas Estimating (c5) is only half as important as Reporting/ Charts (c8). When a factor is compared with itself the ratio of importance is one, resulting in a diagonal line across the matrix.

Goal	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12
c1	1	8	2	3	4	6	8	2	9	2	1	8
c2	0.125	1	0.5	3	1	6	3	0.25	7	1	0.5	8
c3	0.5	2	1	3	2	7	8	1	9	2	1	8
c4	0.333	0.333	0.333	1	3	7	7	1	9	1	0.333	8
c5	0.25	1	0.5	0.333	1	7	3	0.5	7	1	1	8
c6	0.167	0.167	0.142	0.142	0.142	1	1	0.167	1	0.25	0.2	1
c7	0.125	0.33	0.125	0.142	0.333	1	1	0.167	1	0.25	0.2	1
c8	0.5	4	1	1	2	5.999	5.999	1	4	0.25	0.333	3
c9	0.111	0.142	0.111	0.111	0.142	1	1	0.25	1	0.25	0.111	1
c10	0.5	1	0.5	1	1	4	4	4	4	1	0.333	3
c11	1	2	1	3	1	5	5	3	9	3	1	8
c12	0.125	0.125	0.125	0.125	0.125	1	1	0.333	1	0.333	0.125	1

Figure 2: Pairwise comparison of criteria in Level 2

Goal	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12
c1	0.211	0.398	0.273	0.189	0.254	0.115	0.167	0.146	0.145	0.162	0.163	0.138
c2	0.026	0.050	0.068	0.189	0.064	0.115	0.063	0.018	0.113	0.081	0.081	0.138
c3	0.106	0.099	0.136	0.189	0.127	0.135	0.167	0.073	0.145	0.162	0.163	0.138
c4	0.070	0.017	0.045	0.063	0.191	0.135	0.146	0.073	0.145	0.081	0.054	0.138
c5	0.053	0.050	0.068	0.021	0.064	0.135	0.063	0.037	0.113	0.081	0.163	0.138
c6	0.035	0.008	0.019	0.009	0.009	0.019	0.021	0.012	0.016	0.020	0.033	0.017
c7	0.026	0.017	0.017	0.009	0.021	0.019	0.021	0.012	0.016	0.020	0.033	0.017
c8	0.106	0.199	0.136	0.063	0.127	0.115	0.125	0.073	0.065	0.020	0.054	0.052
c9	0.023	0.007	0.015	0.007	0.009	0.019	0.021	0.018	0.016	0.020	0.018	0.017
c10	0.106	0.050	0.068	0.063	0.064	0.077	0.083	0.293	0.065	0.081	0.054	0.052
c11	0.211	0.099	0.136	0.189	0.064	0.096	0.104	0.220	0.145	0.243	0.163	0.138
c12	0.026	0.006	0.017	0.008	0.008	0.019	0.021	0.024	0.016	0.027	0.020	0.017

Figure 3: Normalized matrix for comparison of criteria (Level 2)

Once we have completed the comparison, the decision-maker uses the numbers from the matrix in Figure 2 to get an overall priority value for each factor. In order to do this, the evaluator must first normalize the matrix (as shown in Figure 3) by dividing each element of the matrix by its column total. For example, in Figure 1, the total for the first column (c1) is 4.736, so the normalized value (0.211) in the first entry of the matrix in Figure 3 is obtained by dividing 1 by 4.736. The priority vector in Figure 4 can be obtained by finding the row average of the normalized matrix (Figure 3). For example, the priority vector for Task Scheduling (c1) is obtained by dividing the sum of the rows (0.211 + 0.398 + 0.273 + 0.189 + 0.254 + 0.115 + 0.167 + 0.146 + 0.145 + 0.162 + 0.163 + 0.138 = 2.36) by the total number of the criteria, in this case 12. The value of consistency ratio in this example is 0.08 which is less than 0.1, indicating an acceptable judgment.

After we have completed setting up and determining the priority vectors for the comparison attributes, the next step in the AHP is to perform a comparison of each alternative based on each comparison attributes.

In our example, we need to compare T1 versus T2 versus T3 versus T4 versus T5 on each of the twelve comparison attributes. The alternatives comparison process is exactly the same as the attribute comparison process. The priority vector for the alternatives is shown in Figure 5.

Criteria	Priority Vector
c1	0.197
c2	0.084
c3	0.137
c4	0.097
c5	0.082
c6	0.018
c7	0.019
c8	0.095
c9	0.016
c10	0.088
c11	0.151
c12	0.018

$\lambda_{max} = 13.242$
CI = 0.122
RI = 1.54
CR = 0.08

Figure 4: Priority Vector and consistency ratio (CR) for criteria (Level 2)

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12
T1	0.125	0.143	0.248	0.111	0.125	0.172	0.231	0.289	0.027	0.271	0.167	0.284
T2	0.250	0.143	0.063	0.222	0.250	0.128	0.198	0.193	0.243	0.271	0.167	0.284
T3	0.250	0.143	0.143	0.111	0.125	0.256	0.198	0.079	0.243	0.027	0.167	0.053
T4	0.125	0.286	0.243	0.444	0.250	0.222	0.198	0.360	0.243	0.160	0.333	0.053
T5	0.250	0.286	0.303	0.111	0.250	0.222	0.176	0.079	0.243	0.271	0.167	0.326

Figure 5: Priority Vector for the alternatives (Level 3)

The last step in AHP is to obtain the overall ranking of the five alternatives by mathematically combining the two priority matrices from Figures 4 and 5. Let us denote the matrix in Figure 4 as A and the matrix in Figure 5 and B. The overall ranking is then calculated by multiplying the two matrices, A*B. Figure 6 shows the overall ranking of the tools.

Tools	Overall priority
T1	0.181
T2	0.194
T3	0.150
T4	0.257
T5	0.219

Figure 6: Overall Priority Matrix for Software Management tools

The result in Figure 6 shows that software management tools indicated as T4 has been given the highest overall rating of 0.257 and T3 has been given the lowest rating of only 0.150. Therefore, we can conclude that T4 is the best tool to be used for this particular project and T5 is the worst tool to be selected.

The analysis of the problem as demonstrated in our example can be performed much easier using professional commercial software like Expert Choice. In our example, the analysis was performed using a simple MS Excel macro.

6. Summary and Conclusions

In this paper we have demonstrated how the AHP technique can be applied to facilitate software project manager in evaluating and selecting an appropriate software project management tool. We have considered twelve relevant criteria in making the decision based on the most common features offered by most commercial off-the-shelf solutions (COTS).

Of course, this criteria set can be easily changed, and the goal of this work was not to create a definitive set of features, but rather a representative one to illustrate the process.

In any case, the AHP hierarchy designed in this study required setting up the criteria (factors) in a hierarchy that correctly reflects the process of arriving at the goal of selecting the best software project management tool.

The AHP method provides a common method of hierarchical structuring of complexity into homogeneous clusters of factors. The AHP evaluates the qualitative data through experts' opinions in specifying the weights. Accordingly, the AHP helps the software project manager to identify the principal competitors of the software project management tool and to assess the performance of the tool relative to its principle competitors. Therefore, this work establishes a framework for comparing individual product decisions across projects, project managers, organizational groups, and organizations.

7. References

[1] Basili, V. and Rombach H.. Tailoring the software process to project goals and environments. In ICSE '9: The Ninth International Conference on Software Engineering, 1987, pp 345-359.

[2] Belassi, Walid & Oya Icmeli Tukel, "A new framework for determining critical success/failure factors in project", *International Journal of Project Management*, Vol. 14, No. 3 1996, pp. 141 – 151.

[3] B. W. Boehm and R., "Theory-W software project management principles and examples," *IEEE Transactions on Software Engineering*, vol. 15, no. 7, July 1989, pp. 902 - 916

[4] Barry Boehm, "Software Risk Management: Principles and Practice," *IEEE Software*, vol. 8, No.1, 1991, pp. 32-41.

[5] Cynthia Cohen, Stanley Birkin, Monica Garfield, and Harold Webb, "Managing Conflict in Software Testing,"

Communications of the ACM, vol. 47, no. 1, January 2004, pp. 76-81.

[6] Mustafa, A. M & Al-Bahar F. J, 1991. “Project Risk Assessment Using the Analytical Hierarchy Process,” *IEEE Transactions on Engineering Management*, 38 (1): 46-52

[7] Pinto, J. K., & Slevin, D. P. 1987. “Critical factors in successful project implementation”. *IEEE Transactions on Engineering Management*, 34 (1): 22-27

[8] Pinto, J. K., & Prescott, J. E. 1990. “Planning and tactical factors in the project implementation process”. *Journal of Management Studies*, 27 (3): 305-327

[9] Saaty, T. L. *The Analytic Hierarchy Process*. McGraw-Hill, NY, 1980

[10] Saaty T. L. & L. G. Vargas. 1991. *Prediction, Projection and Forecasting in Applications of the Analytical Hierarchy Process in economics, finance, politics, games and sports*, Boston: Kluwer Academic Publishers.

[11] H. Sharp, et al, "Tensions around the adoption and evolution of software quality management systems: a discourse analytic approach“, *Journal of Human-Computer Studies*, vol. 61, 2004, pp. 219-236.

[12] Donna Davis, “The PM Prescription”, *Software Development*, September 2004, pp. 58-75.