

## Requirements Management Tools A Qualitative Assessment

Rajat R. Sud and James D. Arthur  
Department of Computer Science, Virginia Tech  
Blacksburg, VA 24060 USA

{rsud, arthur} @vt.edu

### Abstract

This report is primarily aimed at people with some background in Requirements Engineering or practitioners wishing to assess tools available for managing requirements. We provide a starting point for this assessment, by presenting a brief survey of existing Requirements Management tools. As a part of the survey, we characterize a set of requirements management tools by outlining their features, capabilities and goals. The characterization offers a *foundation* to select and possibly customize a requirements engineering tool for a software project. This report consists of three parts. In Part I we define the terms requirements and requirements engineering and briefly point out the main components of the requirements engineering process. In Part II, we survey the characteristics and capabilities of 6 popular requirements management tools, available in the market. We enumerate the salient features of each of these tools. In Part III, we briefly describe a “Synergistic Environment for Requirement Generation.” This environment captures additional tools augmenting the requirements generation process. A description of these tools is provided. In the concluding section, we present a discussion defining the *ideal* set of characteristics that should be embodied in a requirements management tool. This report is adapted from a compendium of assignments that were prepared by the students of a class in Requirements Engineering, offered by Dr. James D. Arthur of the Computer Science department at Virginia Tech, in the spring of 2002.

**Keywords:** Requirements Engineering, Requirements Management, Requirements Management Tools

## **Part I**

# **A Brief Introduction To Requirements Engineering**

---

*This chapter begins with some background information about Requirements Engineering and outlines the activities involved in generating requirements. Then we set the stage for understanding the requirements management process. Note that it is beyond the scope of this report to elucidate each of the core activities comprising the requirements engineering process. We have, however, provided a brief summary of each of the core activities. For further details of these activities, the reader is advised to seek a comprehensive reference in Requirements Engineering*

# 1 Requirements Engineering

*Software Systems Requirements Engineering* is the process of discovering stakeholders, their needs and documenting these in a form that is amenable to analysis communication and subsequent implementation. There are inherent difficulties in this process. Stakeholders (including customers, users and developers) may be numerous and distributed. Their goals may vary and conflict depending on their perspectives of the environment in which they work and the tasks which they wish to accomplish.

The process of Requirements Engineering is comprised of the following *core* activities:

- Problem Synthesis
- Requirements Elicitation
- Requirements Analysis
- Requirements Specification
- Requirements Verification and Validation
- Requirements Management

## ***Problem Synthesis***

The process through which the customers' problem is identified, analyzed and decomposed, with the intent of gaining a deeper understanding into the *real* problem at hand

## ***Software Requirements Elicitation***

The process through which the customers (buyers and/or users) and the developer (contractor) of a software system discover, review, articulate, and understand the users' needs and the constraints on the software and development activity

## ***Software Requirements Analysis***

The process of analyzing the customers' and users' needs to arrive at a definition of software requirements

## ***Software Requirements Specification***

The development of a document that clearly and precisely records each of the requirements of the software system

## ***Software Requirements Verification and Validation***

The process of ensuring that the software requirements specification is in compliance with the system requirement (or customer needs or Concept of Operations document), conforms to the document standards of the requirements phase, and is an adequate basis for the architectural (preliminary) design phase

### ***Software Requirements Management***

The planning and controlling of the requirements elicitation, specification, analysis and verification activities

## **2 Software Requirements Management**

Seventy-one percent of all software development projects result in complete failure (i.e., premature cancellation or shelf-ware upon completion). Poor requirements management is generally considered one of the major causes for product failure. *If we do a poor job of understanding our customers' needs, deciding the right features to build, and writing down what we think we want out of a system, how can we possibly expect a successful project?*

“Requirements Management involves establishing and maintaining an agreement with the customer on the requirements for the software project. The agreement forms the basis for estimating, planning, performing, and tracking the software project’s activities throughout the software life cycle.”

- (CMM Practices Manual, CMU/SEI-93-TR-25, L2)

The key to requirements management is *communication*. A good requirements management process helps ensure a high level of communication between stakeholders. For the developers to fully understand the needs of the customers, they must fully understand those needs, and have an open channel of communication with the customers. To deliver a high quality application, it is imperative that customers play an active part in the development process.

Communication is also crucial when requirements change, as they do in any project. Once changes have been agreed upon, they must be incorporated into the project scope, and be communicated to developers and customers, as well. An effective means of communication is thus essential to getting the project right the first time and avoiding expensive re-work later in the development cycle.

## **Part II**

# **Requirements Management Tools**

**–A Survey**

---

*Many tools have been developed for Requirements Management until now. Table 1 introduces them with a brief description of their characteristics, features and capabilities. Table 2 tabulates organizational factors that are instrumental to reach a conclusive decision*

## Requirements Management Tools

Requirements Management Tools can assist organizations in defining and documenting requirements by allowing them to store requirements in a central location. Project teams can then access the requirements to determine what is to be developed, and customers can access the requirements to ensure that their needs were correctly specified. Customers and team members (developers) can then participate in informed discussions in case of any discrepancies, or seek further clarifications in case of uncertainties. This also aids the process of classifying and prioritizing requirements.

Requirements Management Tools

Requirements Management Tool	Requirements Traceability	Requirements Analysis	Security and Accessibility	Portability and Backend Compatibility	Configuration Management	Communication/ Collaboration	Change Management	Online Publishing	Usability	Requirements Specification/ Documentation
<b>Rational Suite AnalystStudio</b> (Use Case Modeling)	Use Cases are linked to requirements. Any change made to a use case is reflected in the requirements as well. Also supports generation of test cases, facilitated by the integrated suite	Requirements organized into packages stored in a database. From different views of the database, attributes of priority, status, difficulty and version number can be assigned	Requirements protection facilitated by providing attributes, links and user permission privileges	Compatible with MS Word providing context and formatting through a familiar interface	'Requirements Revision History' automatically keeps track of all changes to requirements and records the rationale behind any revision	Stores requirements in central locations enabling access and review by the entire development team. Web interface facilitates viewing and modification of requirements	Any change made to an element is reflected in all other elements automatically. Establishes traceability relations among requirements thereby enabling change assessment	Rational ClearQuest generates automatic change request forms on the Web to manage input from stakeholders	Integration with MS Word provides users with a familiar interface to document and edit requirements. Built-in templates provide quick means to create new projects	Requirements are elicited from use cases providing a structured approach to represent user-system interaction
<b>RDT 3.0</b>	'Requirements Allocation Matrices' forms traceability links between requirements, design, development and test components	Customizable user-defined attributes may be associated to every requirement	Workgroup privileges assign "Add, Edit, View and Delete access to users	Integration with MS Office products	Revision tracking and baseline allocation	Check-Out / Check-In enables the sharing of data between different sites, and the ability to collate this data back to the master database	Change Proposal Management enables a change proposal to be identified, and any data which will be added, changed, or deleted as a result of it being accepted	Shared files are stored on a server. RDT supports up to 255 concurrent users	Data Viewing windows enable concurrent views of different data. Document Views allow data to be edited in tabular format. These views are fully user defined	Automatic requirements capture and syntax parsing directly from existing documents
<b>RTM Workshop 5.0</b>	Defines traceability relationships between individual requirements and requirements and other system elements	Defines different attributes for different types of requirements and set attribute values for individual requirements	Users are categorized in groups and assigned access privileges	Built upon the Oracle database technology	Records project baselines before and after changes are incorporated	RTM e-mails all stakeholders on the status of all proposed changes	Maintains a complete change history including versions of impacted requirements and rationale for each change	Formulates web interface for database query, discussion threads and updating requirements attributes	Uses word-processing packages from Adobe, Interleaf and Microsoft for its user interface, providing a familiar interface	Parses a document to load requirements into a database

Requirements Management Tools

Requirements Management Tool	Requirements Traceability	Requirements Analysis	Security and Accessibility	Portability and Backend Compatibility	Configuration Management	Communication/ Collaboration	Change Management	Online Publishing	Usability	Requirements Specification/ Documentation
<b>Telelogic DOORS</b>	Provides 'user-defined, multi-level traceability for relationships' – requirements to test, requirements to design, design to code, etc.	A set of states are associated with each requirement (New, In Review, Partly approved, Approved, More information required, On Hold, Rejected and Applied)	No listed security support. Permits customization of interfaces for every user	Integrates with MS Word and Excel, as well as with other Telelogic tools and third-party applications	DOORS Database Managers manage project baselines, archiving and deleting baselines as the project progresses	On-line Change Proposals and Review Systems, Discussion Threads and distributed data management all collaboration with remote team members	Records the change history of requirements. Notifies team members of changes by email automatically	Telelogic DOORSnet? generates web interfaces to facilitate searching, sorting and editing requirements and submitting change proposals	Provides MS Windows Explorer-like view for project navigation. Customizable user interface by editing colors and graphics to highlight pertinent information	Telelogic DocExpress provides 'one-click' operations to generate requirements for complex systems
<b>Omni Vista OnYourMark Pro</b>	Adds links and tables for each requirement. These tables can have customizable attributes enabling the user to customize traceability links	Requirements analyzed from multiple perspectives – Market, Financial, Development	No listed security support	Compatible with third party application (not listed)	Baselines requirements by ensuring all stakeholders 'sign off' on the latest agreement	Online web publishing facilitates execution of customized queries and sharing of information, all through the web interface	Requirements Triage permits evaluation of impact on overall project based on individual factors	A snapshot of the requirements can be published to the web or exported to an XML format	Detailed graphical representations indicating the effects of adding features on resources (budget, time, schedule, etc.)	Tabular representation recording, tracking and annotating requirements. Permits importing requirements from third party tools
<b>Starbase Caliber-RM</b>	Multiple method of traceability visualization enable users to understand effect of impact of requirements change on other requirements, tests, tasks and/or source code	Spreadsheet views permit sorting and prioritizing requirements according to cost and value. Different views may be created based on the data collected	Centralized repository provides security, visibility and availability to all requirements data	'Open Architecture' permits requirements to be linked to external source code generators, test suites and project management applications	Independently versions each requirement. Baselines specific versions providing users the option to compare requirements versions to assess volatility	Online discussions capture information more effectively than asynchronous (email) communication Immediate notification of changes sent to all stakeholders	Each change creates a unique history record; differences between one version of a requirement and another are highlighted with the reason for the change recorded.	Provides team members with a network connection access to requirements data, imposing minimal bandwidth requirement	GUI is customizable using wizards. Complex scripting languages not required to draft requirements. Can be customized to support many requirements processes	Documents requirements and glossaries to define industry terms, project terms, corporate terms, etc.

**Table 1: A Qualitative Comparison of Requirements Management Tools**



Requirements Management Tools

Requirements Management Tool	Cost (in U.S. dollars)	Operating Expenses/ Licensing Fees	Platform Requirements			
			Processor	Disk Space (Client + Server)	Operating System	Other Requirement(s)
<b>Rational Suite AnalystStudio</b> (Rational Requisite Pro Component)	\$ 2034 per user	Includes 1 year of support	>= 300 MHz 128 MB RAM recommended	96 MB	Microsoft Windows NT 4.0, or Microsoft Windows 98, 2000 or Millennium Edition	<u>Word Processor</u> : Microsoft Word 97, 2000 or 2002 <u>Database</u> : Microsoft SQL Server 7.0 or greater, Oracle 7 or 8 (creates MS Access databases by default) <u>License</u> : Licenses can be node locked or floating. A license manager (flexLM from GLOBEtrotter) is provided with AnalystStudio.
RDT 3.0	\$ 2995 per user for software	\$ 650 per user support for 1 year	>=200 MHz >= 32 MB RAM recommended	60 MB	Microsoft Windows 95/98/NT 4.0/2000	<u>Word Processor</u> : Microsoft Word 97, 2000 or 2002
<b>RTM Workshop 5.0</b>	N/A	N/A	>= 200 MHz >= 64 MB RAM recommended	250 MB	Microsoft Windows 95/98/NT 4.0 and UNIX	<u>Word Processor</u> : Microsoft Word or Adobe FrameMaker <u>Database</u> : Oracle 7 or 8 <u>License</u> : Uses the floating networking license manager
<b>Telelogic DOORS</b>	N/A	N/A	>= 200 MHz Follow Manufacturer recommended RAM	40 MB	Microsoft Windows 98/2000/NT	<u>License</u> : Supports network (floating, node locked, etc.) licensing
<b>Omni Vista OnYourMark Pro</b>	N/A	N/A	N/A	N/A	Microsoft Windows 98/2000/NT	N/A
<b>Starbase Caliber-RM</b>	N/A	N/A	>= 200 MHz >= 256 MB RAM recommended	125 MB	Microsoft Windows 98/2000/NT Java-enabled web client	<u>Database</u> : Caliber-RM ships with a server version of the object-oriented database (OODB), Versant Object Repository, which is also commercially available <u>License</u> : Caliber-RM may be licensed either by seat or on a concurrent user basis

Table 2: Organizational Factors

## **Part III**

# **Tools supporting the Synergistic Environment For Requirements Generation**

---

*In this chapter we describe an environment for requirements generation, being developed at Virginia Tech, and list some tools that we have conceived in support of this environment*

## 1 Overview

The Synergistic Environment for Requirements Generation, being developed is the first of its kind and provides the requirements engineering process the much-needed structure that it has been lacking. Upon completion, the visual representation of the environment will provide guidelines & protocols, and the context in which the activities are progressing, while requirements generation will be facilitated by means of a set of tools and artifacts. For example, a text editor will provide means to document the requirements. An automated requirement measurement tool<sup>1</sup> will then be employed to parse the requirements specification and ensure consistency and unambiguity. The process will further be supported with tools such as *email managers*, *hierarchy generators*, *schedulers*, etc. The operation of all the tools will be under the auspices of the framework outlining the process. We provide a brief description of these tools in the section below.

## 2 Requirements Management Tools Supporting The Synergistic Environment

This section provides a list of tools being incorporated in the Synergistic Environment. It must be noted that these tools are not just applicable to the environment, but can be generically applied to any requirements management effort.

### Hierarchy Generator

The first activity performed while organizing a requirements elicitation meeting is participant selection. This selection varies depending upon the methodology employed for conducting the elicitation meeting. For example, Joint Application Development<sup>2</sup> (JAD) and Participatory Design<sup>3</sup> (PD) advocate different approaches to selecting participants. JAD promotes the selection of a cross-section of a hierarchy of users. It advocates that participants should preferably represent all ‘user classes.’ On the other hand, PD encourages the selection of participants from the same *strata*. According to PD, this helps participants to *shed inhibitions* and engage more actively in the proceedings, without having the fear of being evaluated by seniors. For the advantages and disadvantages of each methodology, the reader is advised to refer to the footnote at the end of this page.

For a requirements engineer, who is not a part of the organization, selecting participants can be a daunting task, especially in larger projects where requirements are to be elicited from a large populace. Remembering the name, contact information, role and designation of each stakeholder can be tedious.

---

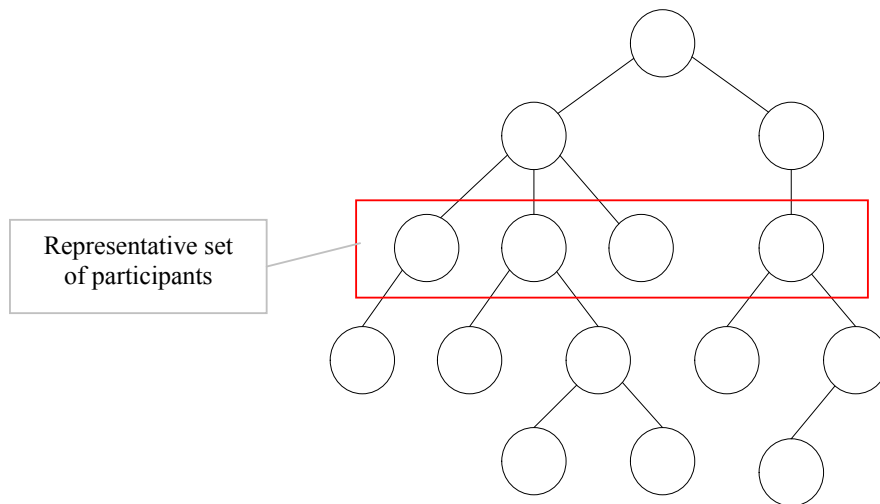
<sup>1</sup> NASA ARM - <http://satc.gsfc.nasa.gov/tools/arm/>

<sup>2</sup> For more details refer “References”

<sup>3</sup> For more details refer “References”



Similarly, the selection criterion for the PD methodology is depicted in Figure 3.



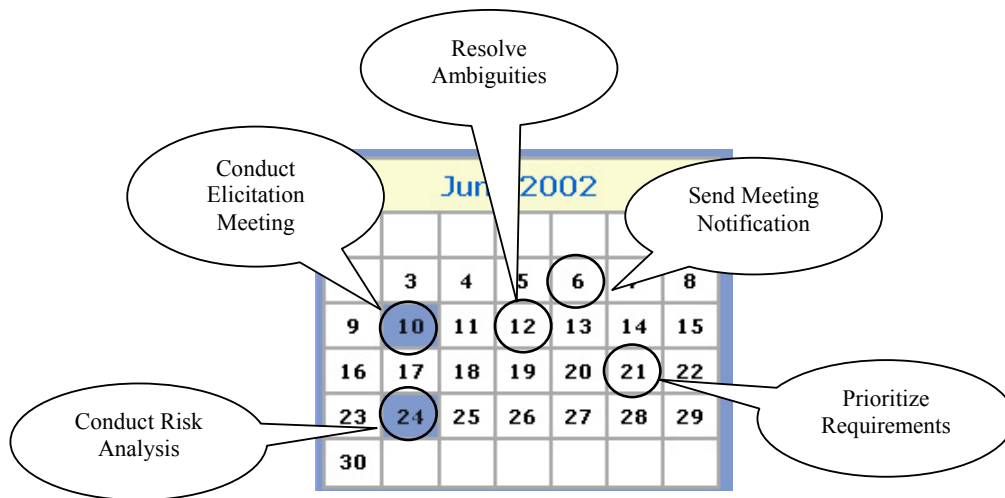
**Figure 3: Participant Selection as per the PD Methodology**

The Hierarchy Generator automates the selection of the representative set of participants and sends out the elicitation meeting notification to every participant.

## **Scheduler**

The Scheduler is responsible for *charting* the course of activities to be performed to ensure an effective requirement generation process. An underlying framework supports the Synergistic Environment. This framework dictates the sequence of activities performed and accounts for the need of an additional iteration. Simply put, the framework is responsible for scheduling and planning all activities.

The Scheduler displays the timeframe within which these activities are to be performed, as shown in Figure 4. The front-end of the Scheduler consists of a calendar that actually displays what activity is to be performed and when. The Scheduler is *dynamic*, i.e., after a particular activity, the requirements manager is asked as to whether another iteration is required. If affirmative, then the (pending) schedule is automatically readjusted to incorporate the additional iteration.

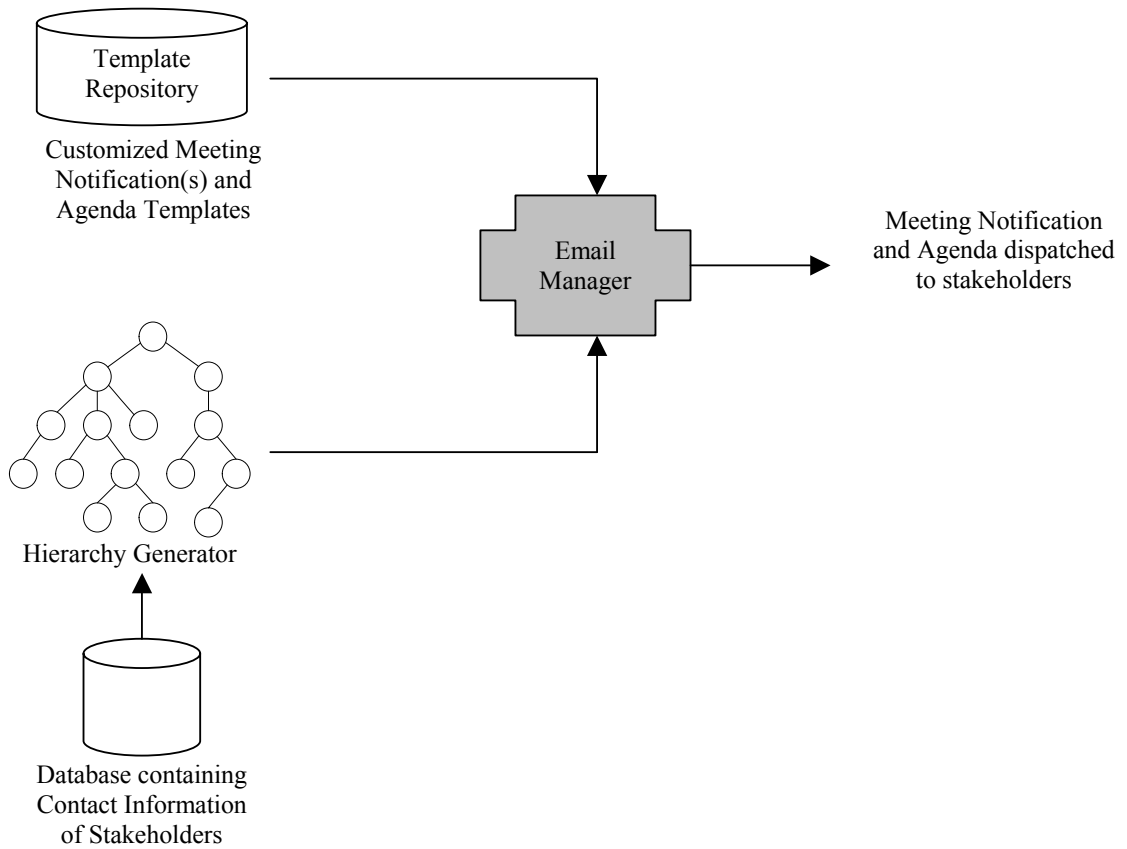


**Figure 4: Scheduler Front-End**  
*Callouts provided by mouse-over effects*

The Scheduler enforces structure to the requirements generation process and also aids planning the overall development process. It must be noted that the Scheduler is customizable, accounting for missed meetings or passed deadlines.

## Email Manager

The Email Manager supports communication amongst stakeholder. It is invoked after the selection of the participants is made (using the Hierarchy Generator) and the template for the Meeting Notification is prepared. An email notification is automatically sent to all intended recipients. A schematic representation is shown in Figure 5.



**Figure 5: A Schematic Representation of the Email Manager**

The Email Manager also facilitates all other communication between the requirements engineer and the customer/users.

# Discussions and Conclusions

---

There are numerous tools<sup>4</sup> available that support the requirements management process. This report surveys six popular tools. Some tools have more features that are more important than others. The leading products have features that make the process easier, efficient and faster for the administration and the team members.

We have enumerated some features that are crucial in a requirements management tool. We have also provided a table that provides questions to be answered while selecting a requirements management tool.

## *Ability to manage versions and changes*

At some stage of development, a project should define a requirements baseline, a specific collection of requirements that a particular release will contain. A requirements management tool must record a history of the changes made to every requirement. This will help explain previous decisions and rationales, thereby minimizing the scope for ambiguity.

## *Store requirements attributes*

Comprehensive information and attributes about requirements should be stored to promote understandability, for customers as well as developers. Attributes must be visible to all concerned stakeholders and selected individuals should be able to update values. Requirements management tools automate the generation of several system-defined attributes, such as date created and version number, and allow the definition of additional attributes of various data types. Attributes such as author, person responsible, origin or rationale, release number, status, priority, cost, difficulty, stability, and risk further augment understandability.

## *Link requirements to other system elements*

Tracing individual requirements to other system components helps ensure that developers do not inadvertently overlook any requirements during implementation. Different kinds of requirements and between requirements in different subsystems should

---

<sup>4</sup> A fairly comprehensive listing of all available requirements management tools can be obtained at <http://www.incose.org/tools/tooltax.html>



be defined. When analyzing the impact of a change proposed in a specific requirement, the traceability links reveal the other system elements that the change might affect.

### ***Track Status***

A good requirements management tool will help track the status of each requirement during development thereby supporting overall project status tracking. If a project manager knows that 55% of the requirements allocated to the next release have been implemented and verified, 28% are implemented but not verified, and 17% are not yet fully implemented, then he or she has good insight into project status.

### ***View requirement subsets***

Requirements Management tools can help sort, filter, or query database(s) of requirements to view subsets of the requirements that have specific attribute values.

### ***Control access***

Permissions should be set for individuals or groups of users. This prevents miscreants and inadvertent accesses to requirements.

### ***Communicate with stakeholders***

Most requirements management tools let team members discuss requirements issues electronically. E-mail messages can notify the affected individuals when a new discussion entry is made or a requirement is changed.

### ***Backend Compatibility with Editors***

Most requirements management tools described in this section are compatible with popular editors, such as Microsoft Word. This is helpful to distribute and edit requirements and interchange information with other tools.

### ***Publishable Web Interfaces***

Recent tools have the ability to publish information on the Internet with a click of a button. This helps sharing information and promotes remote collaboration, particularly in development efforts requiring remote collaboration

No.	Question	Answer
1	Does the tool support analysis of existing documentation?	Y / N
2	Does the tool support automatic parsing of documents?	Y / N
3	Can the tool compare different versions of the same document (detect changes)?	Y / N
4	Does the tool support the reading of requirements from external tools?	Y / N
5	Does the tool support requirements traceability?	Y / N
6	Does the tool have the capability to categorize requirements as they are elicited and documented?	Y / N
7	Can the tool capture or generate graphical representations (system architecture, analysis graphs, tables, etc.) of the system?	Y / N
8	Can the tool link requirements to other system elements (use cases, design, code, test cases, etc.)?	Y / N
9	Does the tool support the association of attributes to every requirement?	Y / N
10	Can the tool detect inconsistencies in requirements (incomplete traceability links, missing attributes, etc.)?	Y / N
11	Does the tool provide information of requirements verification (whether a requirements was implemented or not, who did it and when)?	Y / N
12	Does the tool maintain a history of requirements changes?	Y / N
13	Does the tool have the ability to baseline requirements document and then compare different baseline?	Y / N
14	Does the tool provide security to prevent inadvertent or malicious modifications to information?	Y / N
15	Does the tool support the generation of the Software Requirements Specification (SRS) in standardized formats?	Y / N
16	Does the tool check for inconsistencies (spelling errors, etc.) in the SRS?	Y / N
17	Does the tool support querying?	Y / N
18	Does the tool integrate with other CASE tools?	Y / N
19	Does the tool provide publishable web interfaces?	Y / N
20	Does the tool support communication among team members?	Y / N
21	Is support offered for single and multiple concurrent users?	
22	Which platforms and operating systems does the tool run on?	
23	Does the tool use a commercial or a proprietary database to store requirements?	Y / N
24	What are the hardware and software requirements for the tool?	
25	Does the tool provide an interface for manipulating information?	Y / N
26	Does the tool manage change by adjusting for a change in all pertinent places?	Y / N
27	Does the tool support web access to the information stored in the database?	Y / N
28	Does the tool have a warrantee?	Y / N
29	Are user manuals online or hard copies are provided?	Y / N
30	What kind of support does the vendor provide?	

**Table 3: Ready Reckoner for selecting a Requirements Management Tool**

# References

## **A Brief Introduction to Requirements Engineering**

Thayer R.H., Dorfman M., Software Requirements Engineering, Second Edition, IEE Computer Society, ISBN 0-8186-7738-4

## **Existing Requirements Management Tools**

Rational Suite AnalystStudio – <http://www.rational.com/products/astudio/index.jsp>

RDT 3.0 (Requirements Design and Traceability) – <http://www.igatech.com/rdt>

RTM Workshop 5.0 –

[http://www.chipware.com/sub1.php3?section=rtm&sub=RTM\\_Overview](http://www.chipware.com/sub1.php3?section=rtm&sub=RTM_Overview)

Telelogic DOORS – <http://www.telelogic.com/products/doorsers/index.cfm>

Omni Vista OnYourMark Pro – <http://www.omni-vista.com/Products/OnYourMarkPro/index.htm>

Starbase Caliber-RM – <http://www.borland.com>

Wiegens K.E. – Automating Requirements Management, <http://www.processimpact.com>

## **Tools supporting the Synergistic Environment for Requirements Generation**

E. Carmel, R.D. Whitaker, and J.F. George. PD and joint application design: A transatlantic comparison. Communications of the ACM, 36(4): 40—48, June 1993