

RAWeb Installation and Usage

- INSTALLATION 2**
- DOWNLOADING AND INSTALLING 2
- UPDATING 2
- RUNNING 2
- REVIEWING A REQUIREMENTS DOCUMENT 4
- USAGE 6**
- MANAGING PROFILES 6
- Creating a New Profile* 7
- Updating an Existing Profile* 7
- Deleting a Profile* 8
- Input Types* 9
- Duplicate entries 9
- Actors 9
- Actuators 9
- Functions 9
- Glossary 10
- Sensors 10
- Attributes 10
- Topics 11
- Sinks 12
- Sources 12
- Systems 13
- MANAGING REQUIREMENT SETS 14
- Creating a New Requirement Set* 14
- Importing Requirements Documents* 14
- Manually Adding Requirements* 14
- Modifying/Removing Requirements* 14
- Preparation of a Requirements set* 15
- RUNNING AN ANALYSIS 16
- VIEWING REPORTS (OUTPUTS) 17
- Evaluation criteria* 18
- The results* 20
- Analysis of defects: step by step process* 21
- TROUBLESHOOTING 23**

Installation

RAWeb is currently delivered as a zipfile containing all of the files necessary to run. This includes an embedded Derby database, and Tomcat application server. It is quick and easy to download and install.

Downloading and Installing

If you have never installed RAWeb before, follow these instructions for downloading and installing the full application.

1. To download RAWeb, you can browse to the following link:
ftp://raweb:ra1web@theemphasysgroup.com/RAWeb_v1.5.0.zip
2. Save the file onto your local drive
3. Expand the zip file to any location on your C: drive **NOTE: THE PATH THAT RAWEB IS LOCATED IN CANNOT HAVE ANY SPACES.** This will automatically create a folder called RAWeb_v1.5.0
4. That's all there is to installing the application. There is no installer to run since it is completely self-contained within the unzipped folder structure.

Updating

If you already installed a previous version of RAWeb, follow these instructions for updating the version to v1.5.0

1. Download the update package:
ftp://raweb:ra1web@theemphasysgroup.com/RAWeb_v1.5.0_update.zip
2. Unzip the package to a temporary directory
3. Make sure your current application server is stopped by running <install dir>/bin/shutdown.bat
4. Copy the *webapps* and *shared* folder from the unzipped package to the <install dir>. It should ask if you want to overwrite existing folders. Click Yes to All
5. You are now ready to run RAWeb v1.5.0

Running

RAWeb is a web application that runs on your local machine, so part of running the application is launching the embedded web server.

1. Cd to <install_dir>\bin and double-click startup.bat. This will create a "DOS Window" on your screen, which you can minimize if you like, but it must remain open for the web server containing the RAWeb application to continue running. The window will show quite a few text messages as the web server starts up. This is normal. It's the way the Tomcat server is designed. You will know RAWeb is ready to be accessed when you see the message: "INFO: Server startup in xxxx ms."
2. Before you run an analysis, you need to make sure that you have a valid license file in the <install_dir>\shared\lib folder. Please contact your RAWeb representative to get a valid license file if you do not have one.
3. After the web server has finished starting, open a browser to <http://localhost:8080/RAWeb> in Internet Explorer or Firefox.
4. This will take you to the RAWeb application start page.

5. When you're through running RAWeb, if you want to stop the Tomcat server, you can run shutdown.bat from the same folder that contains startup.bat. A second DOS window will open and then both it and the original window will close after a few seconds.

Reviewing a Requirements Document

This is not an attempt to write a paper on reviewing a document. Other people like Karl Wiegers, (“More about software requirements”, Chapter 8, page 69-73) or Suzanne Robertson (“ An early start to testing: How to test requirements” www.itmweb.com/essay505.htm) have done this.

After reading these papers, there are still questions left:

- What will be the content of our review?
- What words or topics do we expect in the requirements we need to review?

To find answers it is sometimes worthwhile to search for a marketing brochure on the system that is to be developed, and to read about its features, performance or functions.

In other words we make an attempt to look at this system as a system engineer or a potential buyer of the system and wonder, “what makes this system worth building?” Is it, for instance, more accurate than what was built before? Will it have more features? Is it simpler to manipulate?

While contemplating this, take notes on characteristics (like reliability, diagnostics, response time), and how these characteristics could be measured (physical units, like hour, seconds, kilobyte, etc). This is no more than our first best guess, but usually worthwhile.

A similar approach can be done by browsing through the requirements document, making notes and diagrams, etc. It is an attempt to come up with topics that should be addressed in the requirements document, and the question during the review is: are these topics indeed addressed in the requirements document?

One might argue that the author has gone through this process already, so we can assume it is covered. But alas, the author finds it hard to maintain this helicopter-view, and gets entangled in details, and misses crucial aspects (sometimes). As a reviewer we contribute in the development process by providing a fresh look at the document, so that the development process might run smooth: without “oh, sorry, we forgot”, or “when we discover an error in a test-phase: sorry, Chapter 4 and 8 of the document were written by different authors, and therefore the reliability figures are different”.

Sometimes a word count of the words in the requirements document may also provide inspiration for key words that need the reviewer’s attention. This gives certainly an idea what not to review: words that occur so frequently that it will be prohibitive to spend time on these words.

Sometimes an organization has a history of making errors and omissions in certain aspects of the system. By analyzing cost or schedule overruns we may have found for instance: that the lack of proper requirements for “altitude control” caused a lot of trouble. So, this is an excellent reason to analyze this aspect very carefully. By consequence algorithms, accuracy, etc will be topics worth reviewing. Therefore look for units like degrees, radians, etc.

We may even obtain some suggestions from the author(s) of the requirements document. “Would you be so kind to look for A, B, C, since we had a very inexperienced person writing these Chapters?” In the light of producing a good product, as soon as we can, this is beneficial for the team.

Topics that are usually forgotten by development organizations, like diagnostics, error reporting and maintenance are also worth adding to our lists of review interests.

Yet, another way to find topics for Requirements Analysis is to look in the Table of Contents of the document under review. While glancing through the Table we will find areas of interest. The Contents may even raise some questions for further investigation. Jot them down. This is the time to be curious: Is topic A also present in the document? Is its performance stated?

The Review of the requirements is a wonderful opportunity to revisit the problem we are trying to solve. If we are catching defects in this early phase we are greatly assisting the success of our project. This Section is an invitation to think “in the box” and “out of the box”, within the scope of the project.^{1 2}

What is needed is a thorough discussion between authors, reviewers and users of requirements documents. What are important aspects to consider? Not all errors are equally important. Not all issues or defects are equally important either. How will tools assist these user communities?

So a suggestion is: take time to reflect on the review and its keywords. Writing requirements is not easy, and reviewing requirements takes time.

Our goal: Detect important defects early.

¹ Michael Jackson, Problem frames

Analyzing and structuring software problems, page 9, 51:

“Over-generalising just a little, you can state this as a general principle: *the problem is not at the computer interface – it is deeper into the world, further away from the computer.*”

Addison-Wesley (2001)

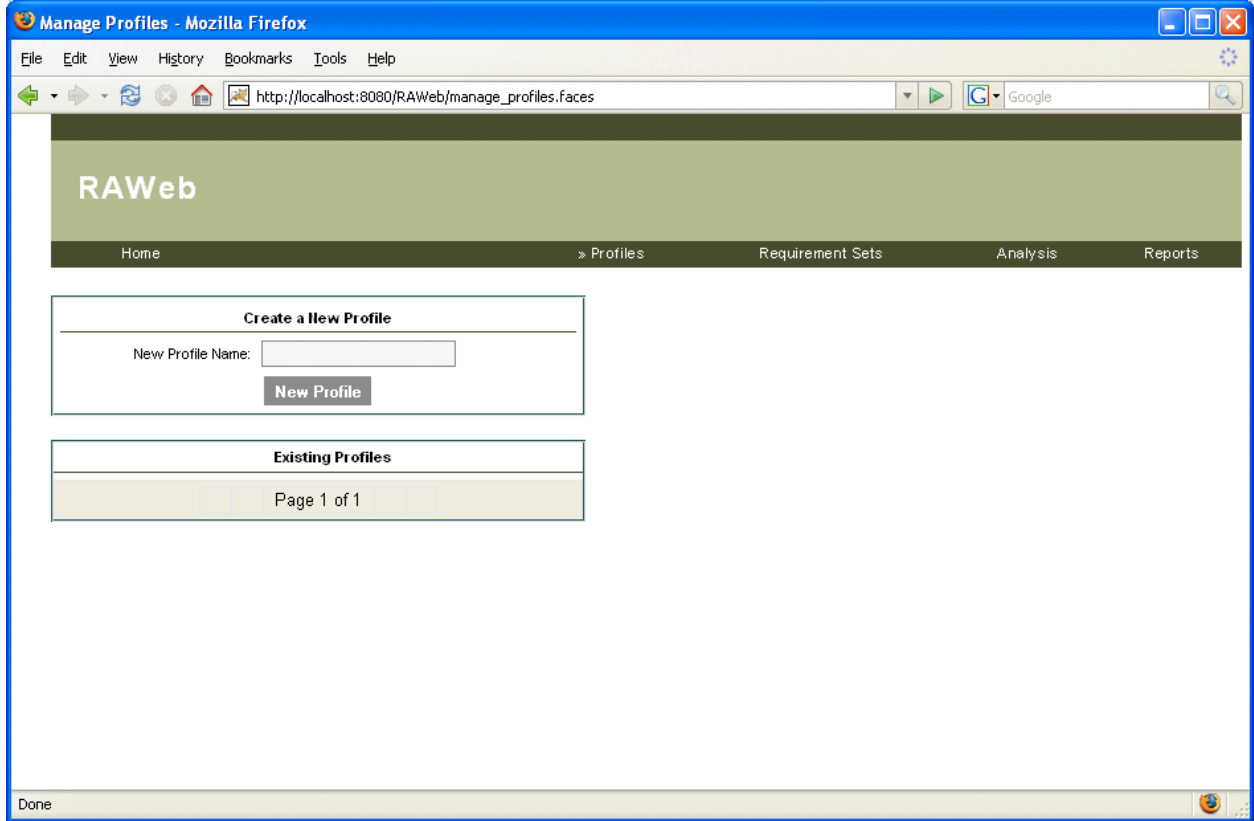
² Edward de Bono, Six Thinking hats

Little, Brown and Company (1985)

Usage

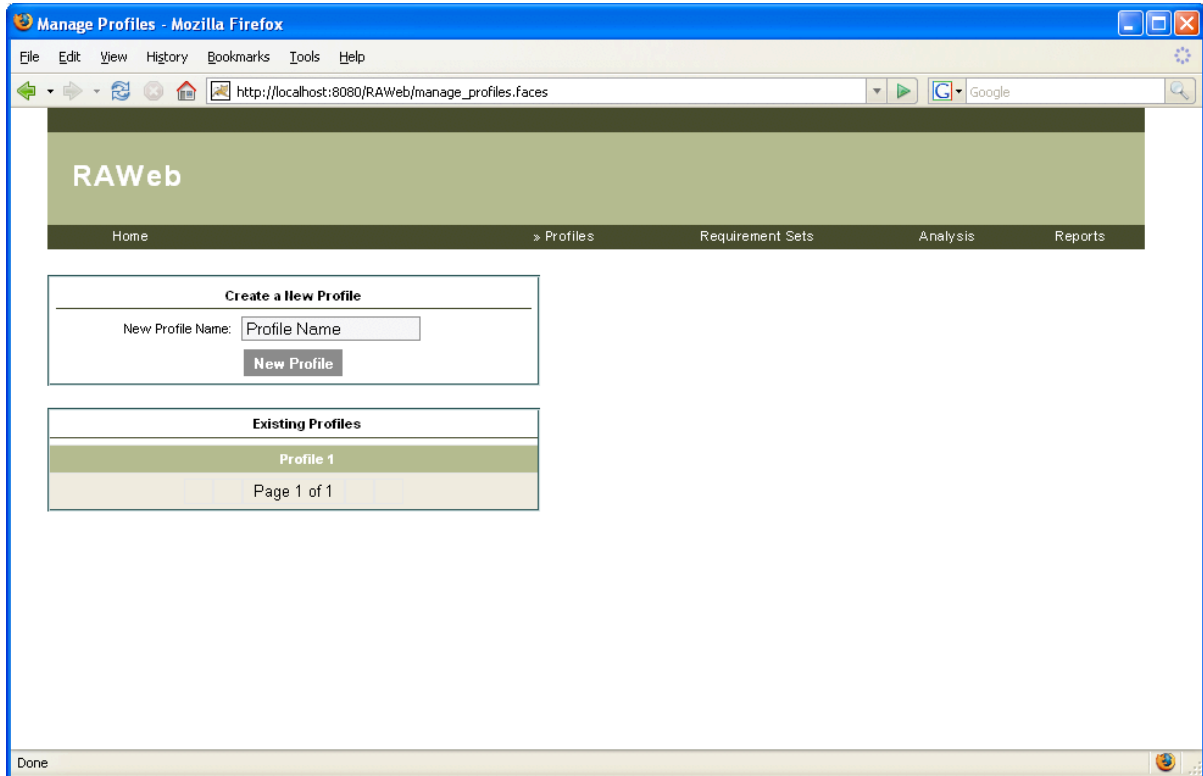
Managing Profiles

Profiles are a way to capture the various types of inputs for RA. All scenarios in this use case are contained in the Profiles section of RAWeb.



Creating a New Profile

1. To create a profile, enter a name for the new Profile and click New Profile



2. This creates a new profile, which will be listed in the Existing Profiles section. The Existing Profiles section lists all profiles in Alphabetical order.
3. To add content to the new Profile, click on the name of the profile and go to the next section.

Updating an Existing Profile

1. To update an existing profile, first click on the name of the profile to update
2. You can manually enter information in any input area
3. Click Update Profile when finished
4. You can also import an existing formatted input document. It must be formatted for RA. You do not need to specify the type of input, RAWeb will read the file and understand the input types. The file could contain any number and types of inputs.
5. When manually entering inputs, RAWeb will store the inputs exactly as entered. When an analysis is run, the inputs will be handled in differing ways, depending on what type of input it is. See the table below to understand how RAWeb handles multiple words on one line, based on input type (Refer to the RA user's manual for a description of each input type):

Input Type	Supports Multiple Words
Actors	Yes
Actuators	Yes
Attributes	No
Glossary	Yes
Functions	No
Sensors	No
Sinks	Yes
Sources	Yes
Subsystems	No
Systems	Yes
Topics	No

For input types that support multiple words, RAWeb will take any number of words on one line and create one entry in the input file. For words that do not support multiple words, RAWeb will break each word into a separate input.

Deleting a Profile

1. To delete a profile, first select the profile from the Existing Profiles list
2. On the Profile Details page, there is a button to delete the profile (at the bottom next to Update Profile)

Input Types

Duplicate entries

Duplicate entries are permitted in a file, but it is not recommended:

- 1) It will increase the execution time of the program without providing additional benefits
- 2) Some sentences are saved in the program, for further investigation. If a sentence is saved several times for the same entry, the duplicate logic may produce double output results.

Actors

Actor is an actor in the system (not necessarily humans).

Actor is one word.

Example :

system

user

subscriber

subscribers

cchs

pto

engineer

Actuators

Actuator is hardware that the actor(s) work on.

So, an actor might be an operator, and an actuator a display.

Actuator is one word.

Example :

engine

motion detector

Functions

This is a list of functions that the system has, or should have (in the reviewer's opinion).

RAWeb will search for these functions, if they show up as requirements or not. The program will also search for inputs, and outputs for each function.

Example:

initialize
initiate
terminate
abort
pause
validate
verify
logging
housekeeping
shutdown

Glossary

This contains the glossary of terms in the requirements document. (if provided).
The file only contains the glossary terms, not their definition.

Example:

product
product family
product instance
product owner
product provider
product retailer
ticket
usage rules

Sensors

Sensor is a list of sensors that are in the system, or that should be part of the system.
The plural of the sensor is taken care of in the program.

Example:

acs
css
dss
thermistor
thermometer
tracker
sensor

Attributes

These are attributes that are reviewed when the word “system” is found in a requirement.

This entry may also show up in Topics . The entries in Attributes may reveal hidden requirements, not necessarily expressed as requirements.

Example :

function
size
weight
safety
security

Topics

This is the main input-file.

It contains the topics that the reviewer would like to analyze.

A suggestion is to enter the special or specific review aspects in the top of the milfile.txt, and to save the more general aspects or words (like function, functionality, hardware) for the end of this list of topics.

Note: If “system” shows up 400 times in the requirements text, then the reviewer might consider not to use this word as a Topic, since highly likely (s)he is not going to review 400 sentences. In that case, look for a more meaningful word.

It is similar to not providing an index in a book for the word “the”.

Note: A suggestion is to limit the number of entries to max 80. RAWeb will still execute when you provide more entries, but it will take execution time. For practical purposes: How many items can you really review, and find issues for?

Example :

human
workmanship
security
guarantee
diagnostics
loading
vibration
health
magnetic
stress
transient
functionality
system

Note: The program will work when a sensor entry is put into the Topics but that entry will not obtain the benefits of a thorough sensor-analysis.

Note: The first entry of Subsystems, and the first entry of Systems, and an entry in Topics should be the same name, for example: “xyz”, to obtain the benefits of this analysis. If they are not the same, usually we will miss some analysis, but usually this is not our major concern.

Sinks

Sink: Source and Sink are an attempt to catch source and sink of information (messages) in the system.

Example:

levying body
 charging unit
 roadside enforcement equipment
 roadside enforcement subsystem
 trusted element
 vehicle appliance
 back-office
 pto
 ptos

Sources

Source: The source of information (message) that flows in the system. Note: a source can be a sink as well, and a sink can also be a source.

Example :

pacemaker
 charging unit
 roadside enforcement equipment
 roadside enforcement subsystem
 trusted element
 beacon
 back-office

Subsystems

A list of the names of subsystems of the system.(if known)

Example:

sewaco
 rnc
 dan
 gic
 board

Note: The first entry of Subsystems, and the first entry of Systems, and an entry in

Topics should be the same name, for example: “xyz”, to obtain the benefits of this analysis. If they are not the same, usually we will miss some analysis, but usually this is not our major concern.

Systems

An entry for the name of the system, and equivalent names.

Example of the file:

the system
et2000
et/bts

Note: The first entry of Subsystems, and the first entry of Systems, and an entry in Topics should be the same name, for example: “xyz”, to obtain the benefits of this analysis. If they are not the same, usually we will miss some analysis, but usually this is not our major concern.

Managing Requirement Sets

Creating a New Requirement Set

1. To create a new Requirement set, click on Requirement Sets across the top menu
2. Enter a name for the set and click New Set
3. The new set will appear in the Requirements Set List, sorted alphabetically based on the set names.
4. Click on the name of the new requirement set to go to the details page, where you can either import requirements documents or manually enter requirements.

Importing Requirements Documents

1. Once you are viewing the details of a given Requirement set, you can import an existing requirements document into the set. The document must already have been formatted according to RA standards. See the RA User's Manual for a description of the formatting tool.
2. Select a formatted file, and select Import
3. Depending on the size of the file, this can take some time. When it is done, the individual requirements will be listed, sorted by Tag.
4. You can import as many documents as you want.

Manually Adding Requirements

1. To manually add requirements, you input the tag and requirement text in the two boxes at the bottom of the page
2. Click Add, and RAWeb will refresh, with the new requirement in the list based on the Tag value

Modifying/Removing Requirements

1. You can modify existing requirements, renumbering tags, changing text, consolidating requirements.
2. When you are finished, click Update Set to have the changes applied to the requirement set.
3. You can "delete" a requirement by removing the tag and clicking *Update Set*.

Preparation of a Requirements set

The text file to be analyzed should be in .txt format. It may imply that during conversion some of the table information in the document may be lost, and the figures will also be lost.

If the original file is a Word-file then the conversion to a .txt file is relatively simple. Convert to .txt, US ASCII, MS-DOS and no CR/LF at the end of the line.

Check if the Word-to-txt-conversion went OK, for instance look for strange characters like ú, ó, and correct them.

Add at the start of each sentence a unique requirement identifier: For example sr1234. Two letters: sr (hard coded for convenience) and 4 digits at the beginning of each sentence, paragraph heading, etc. So comments/remarks/defects can be reported uniquely for each sentence.

Example:

sr1000 This document defines the requirements for an ABC simulator, hardware and software.

To help you to add the tag, a program is available in the map Req_id. It reads the requirements set (already in a .txt file format), and adds the tag in front of each sentence. The tagged requirements file is the input set for RAWeb.

Running an Analysis

Running an analysis means taking a pre-defined requirement set and running RA against that set, using an existing input profile. To run the analysis, you must have the RA key and flash drive properly installed in the computer.

1. Click on Analysis
2. Enter the name for this new analysis run
3. Pick a profile and requirement set
4. Click Run Analysis
5. This will cause RA to launch in a separate window. Once it is launched you need to click File->New
6. Click Ok on the next two dialog boxes that appear
7. RA will now do its processing
8. When it is completed, another dialog box appears saying that it is finished. Click OK and close RA.
9. RAWeb will continue to process the output for a few more seconds before going to the Report tab.

RAWeb will generate the dynamic input files onto the flash drive and call RA to run the analysis. The actual analysis can take several minutes to complete.

After the analysis is complete, RAWeb reads in the raw output file and stores it in the database for future reporting. It also runs the raw output through the HTML output formatter, and stores that output in the database as a base report.

When the run is completed, RAWeb will go to the Reports page. You can click on the run to view the results.

Viewing Reports (Outputs)

By clicking on the Reports tab, you will see a list of all analysis runs that have output. You can sort by the columns by clicking on the column header. When you find the report you want, click on the row and it will take you to the report output. In the Report Output page you will find the HTML formatted output. There is also a link to view the raw output from RA. This will take you to another page with the raw output.

Evaluation criteria

Criterion	Subcriterion	Requirements Assistant Implementation
		Requirements Assistant Output (in other words: how can we find in the output file of Requirements Assistant that this type of error exists in the requirements document?)
<u>CONSISTENCY</u>	under-reference Consistency check	under-reference inconsistent consistency conflict? conflicting
	Non-duplicated	duplicate overlapping overlap
<u>COMPLETENESS</u>	uniqueness Incomplete	unique incomplete missing
	underspecified	underspecified
<u>ACCURACY</u>		precision
<u>READABILITY</u>	Imperatives	imperative constraint?
	Correctness	miss spelled? inconsistency in spelling
	Completeness	indefinite pronoun: pronoun;
	Positively stated	-not- imperative -not- double negative (not-not)
<u>TESTABILITY</u>	Multiplicity	more than one Multiplicity Multiple
	Conditional	conditional

Continuances	Condition
Optionality	continuance
	non-requirement
	creep?
	option?
Weak Phrases	Poor word + Collective noun
	Poor word:
	Poor word;
	Poor verb;
	Poor?
	List Poor words:
	/poor word?
	<poor><word>
	Many Poor words in paragraph
	Poor verb:
	Nominalization
Under specified	underspecified

The results

To make the output of Requirements Assistant more pleasant to read the results are regrouped into four views using Java and Web browser techniques in RAWeb.

View 1

The requirements are ordered by the defects mentioned in the Evaluation criteria : completeness, Readability, etc with their subcategories.

View 2

The requirements are ordered by their original sequence number. Each individual requirement can be selected and its defects/comments are shown.

View 3

The requirements are grouped by the topics we are interested in. So for each entry of the input file milfile.txt we will see the requirements, if the entry shows up in the text.

View 4

The requirements are grouped by units (quantity, like meter, m/s, etc) we are interested in. So for each unit we will see the requirements, if the unit shows up in the text.

For advanced users: Note that the output file “perftext.out” of Requirements Assistant has more comments/defects than those shown in RAWeb.

Analysis of defects: step by step process³

How to find (potential) errors in a specification (requirements document) using the output of RAWeb?

So now you have the results. And I would not be surprised if you are a little overwhelmed. So, where do we start? Please do not shoot the messenger!

Step

1. Before RAWeb is run, it is worthwhile to create a list of topics that should have been addressed in the requirements document, including the required performance of the system.

As an example for a luggage transportation system at an airport: required time of transport from A to B, the max number of missing bags, (max) number of bags that are moved to an (in)- correct location, percentage of bags that is correctly identified with tags, etc. Focus your effort on what is needed for a review. See also page 3 and 4 of this user manual.

In principle Topics for investigation can be put Topics, in Sensors , in the Function, etc. My advice is to put a topic, when it is a function, in the function, and a topic, if it is a sensor, in the list_sensor, etc

RAWeb uses the entries in Sensors to investigate sensor aspects, etc

The program will work when a sensor entry is put into the Topic but that entry will not obtain the benefits of a thorough sensor-analysis.

2. Since correct spelling is a pre-requisite for automatic analysis, a spelling check needs to be done on the input requirements document, and the spelling errors need to be corrected. After RAWeb is run, look for: the term “miss spelled” in the output. Correct the spelling errors and run RAWeb again.
3. Look in View 1 to see if Topics are missing, like growth etc. It merely says that growth, and its synonyms do not show up as a requirement. We may have to run RAWeb with a richer set of synonyms, or alternatively search the specification for equivalent words. A missing topic like growth serves as focus for further analysis.
4. The word “creep” in the output of RAweb may indicate that the present requirements do not sufficiently scope the effort.
5. The word “ambig*” in the output indicates potential discussions on scope. Check each ambiguity: might it lead to an incorrect system?

³ In case you would like to comment on RAWeb or when you would like to share suggestions for this manual, please let us know: contact page of www.requirementsassistant.nl or www.emphasysgrp.com. We are curious about your findings.

6. The output “no shall” may indicate a sentence in the document, that should have been a requirement.

Hopefully the author(s) of the requirements document and their Quality Control organization did this quality control already, before (s)he submitted the document for review. Under pressure it sometimes is neglected.

8. It is important to execute this crude analysis first. A sentence by sentence analysis will certainly provide more detail, VIEW 2, but it is also much more tedious! Although RAWeb does not take much execution- time to analyze a document, but in my experience it is worth taking time to reflect on its findings. Did we grab the essence of what this document is trying to tell us?
9. View 1. Look at the “almost double” sentences. Has the input document under review duplicate requirements?
10. Look at View 1 “Inconsistent?” Did we find inconsistent requirements?
11. Use the Evaluation criteria, as presented in View 1, to check on completeness, testability, etc. What is in your opinion the most important criterion, and start to work from there, until you find enough issues to proceed.
12. The purpose of this exercise is – in my opinion – not necessarily to find all the issues, but to find enough issues to speed the development process. And to learn as an organization how to avoid the important issues.
13. Have fun, and let us know if we can assist you more.

Troubleshooting

If you have any problems running RAWeb or understanding the results that are reported, please contact us.

Please provide us with:

- a) The problem you encountered
- b) The version of RAWeb you were using (it will say in the upper left corner of your web browser)
- c) The raw output (just save it to a text file)

Please send your bug reports to raweb@emphasysgrp.com, or call us at 888-784-7759.