

ITEA



ITEA

Information Technology for European Advancement

Task 1.8 Requirements Management method definition *Requirements Specification template description (D 1.8.4)*

Version 01 - Public
Edited by Loek Bokhorst

Software Development Process for Real-Time Embedded Software Systems (DESS)

ITEA COMPETENCES involved:

- 1) Complex Systems Engineering**
- 2) Communications**
- 3) Distributed Information and services**

November, 2001

Revision History

Version	Date	Author	Remarks
1	2001-11-02	SLE	Shell description: added Fit Criterion explanation

CONTENTS.

1. PREAMBLE 5

2. INTRODUCTION 6

2.1 Description 6

2.2 Scope 6

2.3 Glossary of terms 6

2.4 References 6

3. OVERVIEW 7

3.1 Project drivers..... 7

3.1.1 Purpose of the Product..... 7

Background to the project effort..... 7

Goals of the product 7

3.1.2 Client, Customer and other Stakeholders 7

The client..... 8

The customer 8

Other stakeholders 8

3.1.3 Users of the Product..... 9

The users of the product..... 9

The priorities assigned to users..... 10

User participation 10

3.2 Project constraints 11

3.2.1 Mandated Constraints 11

Solution constraints 11

Implementation environment 11

Partner applications/external interfaces..... 12

Commercial off-the-shelf packages or components 12

Anticipated workplace environment..... 13

Development schedule constraints..... 14

Development environment constraints..... 14

Financial constraints..... 15

3.2.2 Naming Conventions and Definitions: the dictionary..... 15

3.2.3 Relevant Factors and Assumptions 16

External factors 16

Assumptions..... 16

4. FUNCTIONAL REQUIREMENTS..... 18

4.1 Scope of the Work 18

ITEA

	<i>The context of the work</i>	18
	<i>Work partitioning</i>	19
4.2	Scope of the Product	20
	<i>Use case list</i>	20
4.3	Functional, Component and Data Requirements	21
	<i>Functional Requirements</i>	21
	<i>Component and data requirements</i>	21
4.4	Domain Model	22
4.5	Product-line requirements	22
5.	NON-FUNCTIONAL REQUIREMENTS	23
5.1	Precedence and Priority requirements	23
5.2	Look and Feel Requirements	23
	<i>The interface</i>	23
	<i>The style of the product</i>	24
5.3	Usability Requirements	24
	<i>Ease of use</i>	24
	<i>Ease of learning</i>	25
5.4	Testing Requirements	26
5.5	Performance and Safety Requirements	26
	<i>Speed requirements</i>	26
	<i>Safety critical requirements</i>	27
	<i>Precision requirements</i>	28
	<i>Reliability and Availability requirements</i>	28
	<i>Capacity and Resource utilisation requirements</i>	29
5.6	Operational Requirements	29
	<i>Expected physical environment</i>	29
	<i>Expected technological environment</i>	30
	<i>Partner applications</i>	30
	<i>Required States and Modes</i>	31
5.7	Maintainability and Portability Requirements	31
	<i>Maintainability</i>	31
	<i>Special conditions</i>	31
	<i>Portability requirements</i>	32
	<i>Product adaptation requirements</i>	32
5.8	Security Requirements	33
	<i>Confidentialness / Access constraints</i>	33
	<i>File integrity requirements</i>	34
	<i>Audit requirements</i>	34
5.9	Cultural and Political Requirements	34
5.10	Legal Requirements and Standards	35
	<i>Legal Requirements</i>	35
	<i>Standards</i>	36
6.	PROJECT ISSUES	37
6.1	Open Issues	37
6.2	Off-the-shelf Solutions (ready-made systems/components)	37
	6.2.1 Is there a ready-made system or component that could be bought?	37
	6.2.2 Can ready-made components be used for this product?	37
	6.2.3 Is there something that we could copy?	38
6.3	New Problems	38
	6.3.1 What problems could the new system cause in the current environment?	38
	6.3.2 Will the new development affect any of the installed system?	38

ITEA

6.3.3	Will any of our existing users be adversely affected by the new development?.....	39
6.3.4	What limitations exist in the anticipated implementation environment that may inhibit the new system?.....	39
6.3.5	Will the new system create other problems?	39
6.4	Tasks.....	40
	<i>What steps have to be taken to deliver the system?.....</i>	<i>40</i>
	<i>Development phases.....</i>	<i>40</i>
6.5	Implementation procedures	41
6.5.1	What special requirements do we have to get the existing data, and procedures to work for the new system?	41
6.5.2	What data has to be modified/translated for the new system?.....	41
6.6	Risks.....	42
6.7	Costs	42
6.7.1	Project costs are described in the Project Management Plan [PMP]	42
6.8	Product- User documentation	43
6.9	Waiting Room.....	45
6.10	Ideas for Solutions.....	45
7.	PROCESS REQUIREMENTS	46
7.1	Requirement characteristics and traceability	46
7.2	Process improvement requirements	46
8.	APPENDIX A, REQUIREMENT FORM.....	47
8.1	Requirement numbering	48
8.2	Event/use case #.....	48
8.3	Fit Criterion.....	48
8.4	Satisfaction/dissatisfaction.....	48
8.5	Dependencies/conflicts.....	49
8.6	Priority	49
8.7	Status	49
8.8	Difficulty / Stability	49
8.9	Traceability	49
8.10	History	50
9.	APPENDIX B, DEFINITIONS	51
10.	APPENDIX C, USE CASE TEMPLATE	53

1. PREAMBLE

The Requirements Specification template focuses on the collection and organization of all requirements surrounding the project. For example, it may be desired to have a separate SRS to describe the complete software requirements for each feature in a particular release of the product. This may include several use cases from the system use-case model, to describe the functional requirements of this feature, along with the relevant set of detailed requirements in the Supplementary Specifications.

The Requirements Specification template is obviously related to the Vision document. Indeed, the Vision document serves as the input to the Requirements Specification. But the two artefacts serve different needs and are typically written by different authors. At this stage in the project, the focus of the project moves from the broad statement of user needs, goals and objectives target markets and features of the system to detailed requirements for the solution.

The usage of this template depends on the situation. Select all the sections in the Requirement Specification Template that apply to the project and replace the entries where applicable. Delete any sections that are not relevant. Add any applicable new sections, and any facts that are relevant to your product. See Figure 1.

Requirements Specification Template

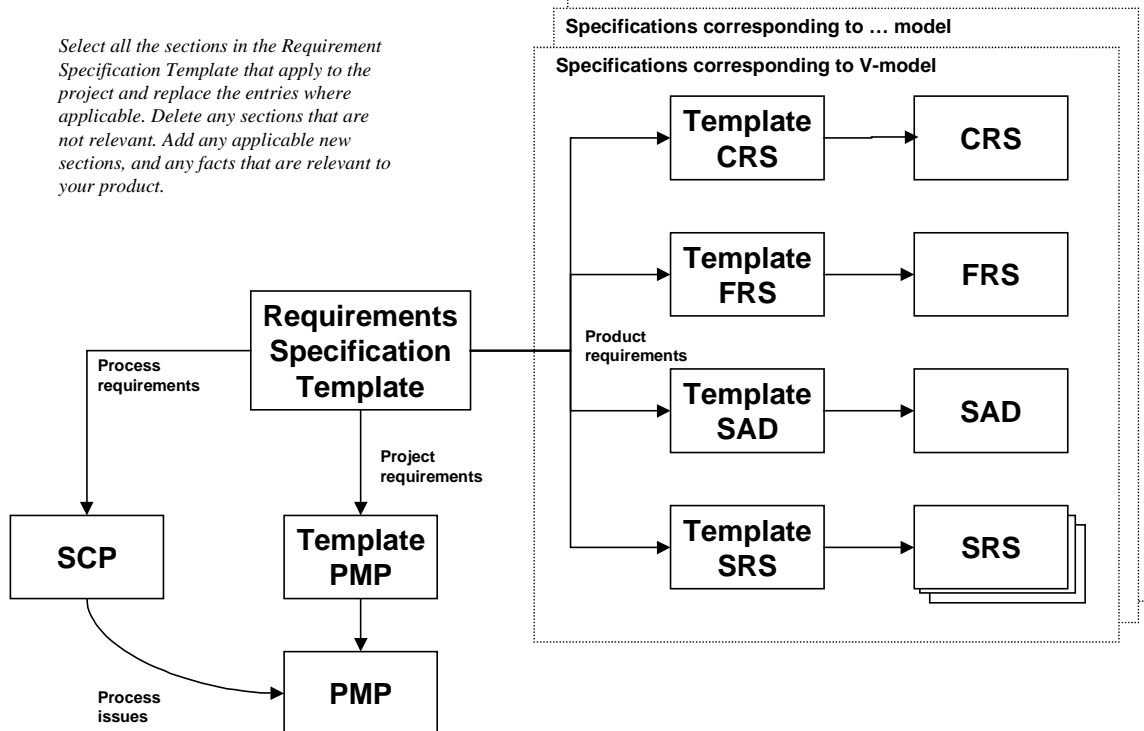


Figure 1 Requirements Specification Template

The Requirements Specification template is based on the work of the partners, participating in ITEA DESS. Furthermore, parts of the Volere template are incorporated (see note below).

2. INTRODUCTION

2.1 Description

Describe the purpose of the project, for example is to support and improve the process of constructing hardware/software systems out of reusable intellectual properties.

Each requirement shall be assigned a unique identifier to support testing and traceability.

Describe the purpose of the document for the project.

2.2 Scope

The Vision document is the starting document for all activities within the project. It serves as a kick-off for the communication between the customers and the development team. The Vision document is input for the Requirement Specifications. These specifications quantify the needs of the customer into clear and precise requirements in order to start the development of the required system.

2.3 Glossary of terms

The following is a list of abbreviations and terms used in this document.

CRS	Commercial Requirements Specification
FRS	Functional Requirements Specification
PMP	Project Management Plan
SAD	Software Architectural Design
SCP	Software Creation Process
SRS	Software requirements Specification

2.4 References

The following is a list of references used in this document.

Identification	Title, author, supplier, date and status	Reference
----------------	--	-----------

Note on Volere:

The Volere template provides sections for each of the requirements types appropriate to today's software systems. It is the intention of Volere template to adapt it to the defined requirements gathering process and requirements tool. The template can be used with Requisite, DOORS, Caliber RM and other popular tools. The Volere work is copyright © 1996 - 2000 Atlantic Systems Guild, but may be adapted for internal use provided copyright is acknowledged. E-mail to : james@systemsguild.com . Refer to: <http://www.atlsysguild.com/GuildSite/Robs/Template.html> .

3. OVERVIEW

3.1 Project drivers

3.1.1 Purpose of the Product

This section describes the purpose of the product in terms of its context and goals.

Background to the project effort

Content

A short description of the work context and the situation that triggered the development effort. It should also describe the work that the user wants to do with the delivered product.

Motivation

Without this statement, the project lacks justification and direction.

Considerations

You should consider whether or not the user problem is serious, and whether and why it needs to be solved.

Goals of the product

Content

This boils down to one, or at most a few, sentences that say, "What do we want this product for?" In other words, the real reason that the product is being developed.

Motivation

There is a real danger of this purpose getting lost along the way. As the development effort heats up, and the customer and developers discover more and more what is possible, it may well be that the system as it is being constructed wanders away from the original goals. This is a bad thing unless there is some deliberate act by the client to change the goals. It may be necessary to appoint a person to be "custodian of the goals", but it is probably sufficient to make the goals public, and periodically remind the developers of it. It should be mandatory to acknowledge the goals at every review session.

Examples

"We want to give immediate and complete response to customers ordering our goods over the telephone."

"We want to be able to forecast the weather."

Fit Criterion

An objective measure that will enable testing to determine if the goal has been met by the product.

3.1.2 Client, Customer and other Stakeholders

This section specifies the major stakeholders in the project.

ITEA

The client

Content

This item must give the name of the client. It is permissible to have several names, but more than three negates the point.

Motivation

The client has the final acceptance of the system, and thus must be satisfied with the system as delivered. Where the system is being developed for in-house consumption, the roles of the client and the customer may be filled by the same person. If you cannot find a name for your client, then perhaps you should not be building the product.

Considerations

Sometimes, when building a package or a product for external users, the client is the marketing department. In this case, a person from the marketing department must be named as the client.

The customer

Content

The name of the person who plays the role of the customer for the product. In the case of in house development the roles of the client and the customer are often played by the same person. In the case of the development of a mass market product there may be several people playing the role of customer. In the case of a product that is being developed for an international market, there might be a different customer (or customer profile) in each country.

Motivation

The customer role is ultimately responsible for deciding whether or not to buy the product from the client. The product must be built to satisfy the aims of the customer/s whilst conforming to the constraints of the client. Even if the customer/s are people who work for another part of the client's organization, they might still have the authority to decide whether or not to invest budget in the new product.

Other stakeholders

Content

The roles and (if possible) names of other people and organizations who are affected by the product or whose input is needed in order to build the product.

Examples of stakeholders include:

- *Users*
- *Sponsors*
- *Testers*
- *Business Analysts*
- *Technology Experts*
- *System Designers*
- *Marketing Experts*

ITEA

- *Legal Experts*
- *Domain Experts,*
- *Product-line Experts*
- *Usability Experts*
- *Representatives of external associations*

For each type of stakeholder identify:

- *Stakeholder Identification (some combination of role/job title, person name, organisation name),*
- *Knowledge needed by the project,*
- *Necessary degree of involvement for that stakeholder/knowledge combination,*
- *Degree of influence for that stakeholder/knowledge combination,*
- *Agreement on how to address conflict between stakeholders who have an interest in the same knowledge*

Motivation

Failure to recognize stakeholders results in missing requirements.

3.1.3 Users of the Product

This section identifies the users of the product and describes their involvement in ensuring that the product meets their requirements.

The users of the product

Content

A list of the potential users of the product. For each category of user, provide the following information:

- *User name – This is most likely to be the name of a user group like: schoolchildren, road engineers, project managers.*
- *User role – Summarizes the users' responsibilities.*
- *Subject matter experience – Summarizes the users' knowledge of the business. Rate as novice, journeyman or master.*
- *Technological experience – this describes the users' experience with relevant technology. Rate as novice, journeyman or master.*
- *Other user characteristics – Describe any characteristics of the users that have an effect on the requirements and eventual design of the product. Describe things like:*
 - *Physical abilities/disabilities*
 - *Intellectual abilities/disabilities*
 - *Attitude to job*
 - *Attitude to technology*
 - *Education*
 - *Linguistic skills*

ITEA

- Age group
- Gender

Motivation

Users are human beings who interface with the product in some way. The role of the client is to pay for the development of the product and the role of the customer is to buy the product. The role of the user is to use the product to do work. You use the characteristics of the users to define the usability requirements for the product.

Examples

Users can come from wide, and sometimes unexpected, sources. Consider the possibility of your users being clerical staff, shop workers, managers, highly-trained operators, general public, casual users, passers-by, illiterate people, tradesmen, students, test engineers, foreigners, children, lawyers, remote users, people using the system over the telephone or Internet connection, emergency workers, and so on.

The priorities assigned to users

Content

Attach to each category of user a priority rating. This gives the importance and precedence of the user. Prioritize the users into:

Key users. These are critical to the continued success of the product. Give greater importance to requirements generated by this category of user.

Secondary users. They will use the product, but their opinion of it has no effect on its long-term success. Where there is a conflict between secondary users' requirements and those of key users the key users take precedence.

Unimportant users. This category of user is given the lowest priority. It includes infrequent, unauthorized and unskilled users, and people who misuse the product.

Percentage of this type of user – this is intended to assess the amount of consideration given to this category of user.

Motivation

If some users are considered to be more important to the product, or the organization, then this should be stated because it should affect the way that you design the product. For instance, you need to know if there is a large customer who has specifically asked for the product, and if they do not get what they want then the results could be a significant loss of business.

Some users may be listed as having no impact on the product. This means that the users will make use of the product, but have no vested interest in it. In other words, these users will not complain, nor will they contribute. Any special requirements from these users will have a lower design priority.

User participation

Content

Where appropriate attach to the category of user, a statement of the participation that you think will be necessary to provide the requirements. Describe the contribution that you expect this user to provide – product domain knowledge, interface prototyping, usability requirements etc. If

ITEA

possible, assess the minimum amount of time that this user must spend for you to be able to determine the complete requirements.

Motivation

Many projects fail through lack of user participation, sometimes this is because the required degree of participation was not made clear. When people have to make a choice between getting their everyday work done and working on a new project, the everyday work takes priority. This requirement makes it clear, from the outset, that specified user resources must be allocated to the project.

3.2 Project constraints

3.2.1 Mandated Constraints

This section describes constraints on the requirements and the eventual design of the product.

Solution constraints

Content

This specifies constraints on the way that the problem must be solved. You can think of these as mandated solutions. Carefully describe the mandated technology, include the appropriate version numbers, and a measurement of how you will test compliance. If possible, you should also explain the reason for using the technology.

Motivation

To identify constraints that must be part of the final product. Your client, customer or user may have design preferences. If these are not met then your solution is not acceptable.

Examples

The product must use the current 2-way radio system to communicate with the drivers in their trucks.

The product must use the Windows NT operating system.

The product must be a hand-held device.

Considerations

We want to define the boundaries within which we can solve the problem. Be careful because anyone who has experience/exposure to a piece of technology tends to see requirements in terms of that technology. This tendency leads people to impose solution constraints for the wrong reason and it's very easy for untrue constraints to creep into a specification. If you impose untrue constraints the danger is that you do not have the creative freedom to come up with the best solution to the problem. The solution constraints should only be those that are absolutely non-negotiable. In other words, however you solve this problem you must use this particular technology. Any other solution would be unacceptable.

Implementation environment

Content

ITEA

This describes the technological and physical environment in which the product will be installed. This includes automated, mechanical, organizational and other devices. These include the non-human adjacent systems.

Motivation

To describe the technological environment into which the product must fit. The environment places design constraints on the product. This part of the specification provides enough information about the environment for the designers to make the product successfully interact with its surrounding technology.

The operational requirements are derived from this description.

Examples

This can be shown as a diagram, with some kind of icon to represent each separate device or person (processor). Draw arrows to identify the interfaces between the processors and annotate them with their form and content.

Considerations

All the component parts of the current system, regardless of their type, should be included in the description of the implementation environment.

If the product is to affect, or be important to the current organization, include an organization chart.

Partner applications/external interfaces

Content

This describes applications that are not part of the product but with which the product will collaborate. These can be external applications, commercial packages or pre-existing in-house applications.

Motivation

To provide information about design constraints that are caused by using partner applications. By describing or modelling these partner applications, you discover and highlight potential problems of integration.

Examples

This section can be completed by including written descriptions, models or references to other specifications. The descriptions must include a full specification of all interfaces that will have an effect on the product.

Considerations

Examine the work context model to determine if any of the adjacent systems should be treated as partner applications. It might also be necessary to examine some of the details of the work to discover relevant partner applications.

Commercial off-the-shelf packages or components.

Content

ITEA

This describes applications or components that must be used to implement some of the requirements for the product.

Motivation

To identify and describe existing commercial products that must be incorporated into the eventual product. The characteristics, behaviour and interfaces of the package or components are design constraints.

Examples

This section can be completed by including written descriptions, models or references to vendor's application- or components- specifications.

Re-use of customer software (who is responsible ?)

Software to be purchased:

- *Communication protocol software*
- *DBMS*
- *Drivers for hardware control*
- *...*

Allowed licence fees (once and periodically)

Considerations

The use of a specific package or component has been mandated. When gathering requirements you may discover requirements that are in serious conflict with the behaviour and characteristics of the package or component. Keep in mind that the use of the package or component was mandated before the full extent of the requirements was known. In light of your discoveries you must consider whether the package or component is a viable choice when all the requirements are known. If the use of the package or component is not negotiable, then the conflicting requirements will have to be discarded.

If all internal interfaces are left to the design, this fact shall be stated explicitly.

Anticipated workplace environment

Content

This describes the workplace in which the users will work and use the product. This should describe any features of the workplace that could have an effect on the design of the product.

Motivation

To identify characteristics of the physical workplace so that the product is designed to compensate for any difficulties.

Examples

The printer is a considerable distance from the user's desk. This constraint suggests that printed output should be de-emphasized.

The workplace is noisy, so audible signals might not work.

The workplace is outside so the product must be waterproof, have displays that are visible in sunlight and allow for the effect of wind on any paper output.

ITEA

The user will be standing up or working in positions where he must hold the product. This suggests a hand-held product but only a careful study of the users' work and workplace will provide the necessary input to identifying the operational requirements.

Considerations

The physical work environment constrains the way that work is done. The product should overcome whatever difficulties exist, however you might consider a redesign of the workplace as an alternative to having the product compensate for it.

Development schedule constraints

Content

How long do the developers have to build the system?

Any known deadlines, or windows of opportunity, should be stated here.

Motivation

To identify critical times and dates that have an effect on product requirements. If the deadline is short, then the requirements must be kept to whatever can be built within the time allowed.

Examples

To meet scheduled software releases.

There may be other parts of the business or other software products that are dependent on this product.

Windows of marketing opportunity.

Scheduled changes to the business that will use your product. For example the organization may be starting up a new factory and your product is needed before production can commence.

Considerations

State deadline limitations that exist by stating the date and describing why it is critical. Also identify prior dates where parts of your product need to be available for testing.

You should also ask questions about the impact of not meeting the deadline like:

What happens if we don't build the system by?

What is the financial impact of not having, the system by...?

Development environment constraints

Content

How do the developers have to build the system?

Motivation

To identify requirements for the development environment that is to be used for developing the product.

Examples

Tools: modelling (UML real-time)

ITEA

Design methods/techniques (SA/SD, RUP, UML, ...)

Configuration Management: Continuous, PVCS, SourceSafe

IDE: Borland, Visual Studio, LabView

Programming languages:(C, C++, LabView, IEC1131,)

Coding standard

Financial constraints

Content

What is the financial budget for the system?

The budget for the system, expressed in money or available resources.

Motivation

The requirements must not exceed the budget. This may constrain the number of requirements that can be included in the product.

The intention of this question is to determine if the product is really wanted.

Considerations

Is it realistic to build a product within this budget? If the answer to this question is no, then either the client is not really committed to building the product or does not place enough value on the product. In either case you should consider whether it is worthwhile continuing.

3.2.2 Naming Conventions and Definitions: the dictionary

This section gives definitions of all terms used in the project. Abbreviations and acronyms are listed in section Glossary of terms on page 5 of this document.

Content

A dictionary containing the meaning of all the names used within the requirements specification. Select names carefully to avoid giving a different, unintended meaning.

This dictionary should build on the standard names that your organization, or industry, uses. The names should also reflect the terminology in current use within the work area.

The dictionary contains all important names that are used by the project. For each name write a succinct definition. This definition must be agreed by the appropriate stakeholders.

Motivation

Names are very important. They invoke meanings that, if carefully defined, can save hours of explanations. Attention to names at this stage of the project helps to highlight misunderstandings.

The dictionary produced during requirements is used and added to throughout the project.

Examples

Gritter Truck -- a truck used for spreading de-icing substances on roads in winter.

Considerations

ITEA

Make use of existing references and data dictionaries. Obviously it is best to avoid renaming existing items unless they are so ambiguous that they cause confusion.

From the start of the project emphasise the need to avoid homonyms and synonyms and explain how they increase the cost of the project.

Later on, as the analysis progresses, this description will be expanded to define all the elementary terms that describe a gritter truck.

*Gritter Truck = Truck Registration Number, Truck Capacity,
Truck Service Status*

As we progress through the requirements specification each of the elementary terms will be defined in detail

Truck Capacity - the number of tonnes of grit that can be carried by a truck. From 0.5 to 2 tonnes

3.2.3 Relevant Factors and Assumptions

This section describes external factors and assumptions that have an effect on the product, but are not mandated requirements constraints.

External factors

Content

Statements describing other forces, systems, activities in the world that have an effect on this system.

Motivation

Relevant facts might contribute to requirements. They will have an effect on the eventual design of the product.

Examples

One ton of de-icing material will treat 3 miles of single lane roadway.

The existing application is 10,000 lines of C code.

Others (Logging, installation, configureability, backup, multi-single tasking, testability, ...)

Assumptions

Content

A list of the assumptions that the developers are making. These might be about the intended operational environment, but can be about anything that has an effect on the product.

Motivation

To make people declare the assumptions that they are making. Also to make everyone on the project aware of assumptions that have been made.

Examples

Assumptions about new laws or political decisions.

ITEA

Assumptions about what your developers expect to be ready in time for them to use. For example, other parts of your products, the completion of other projects, software tools, software components, etc.

Assumptions about the technological environment in which the product will operate. These assumptions should highlight areas of expected compatibility.

The software components that will be available to the developers.

Other products are being developed at the same time as this one.

Availability and capability of bought-in components.

Dependencies on computer systems or people external to this project

Considerations

We often make unconscious assumptions. It is necessary to talk to the members of the project team to discover any unconscious assumptions that they have made. Ask stakeholders (both technical and business-related) questions like "What software tools are you expecting to be available, will there be any new software products, are you expecting to use a current product in a new way, are there any business changes you are assuming we will be able to deal with....?" It is important to state these assumptions up front. You might also consider the probability of whether or not the assumption is correct, and where relevant, a list of alternatives if something that is assumed does not happen.

4. FUNCTIONAL REQUIREMENTS

4.1 Scope of the Work

This section describes the context of the work and identifies the business processes and events relevant to the use of the product.

The context of the work

Content

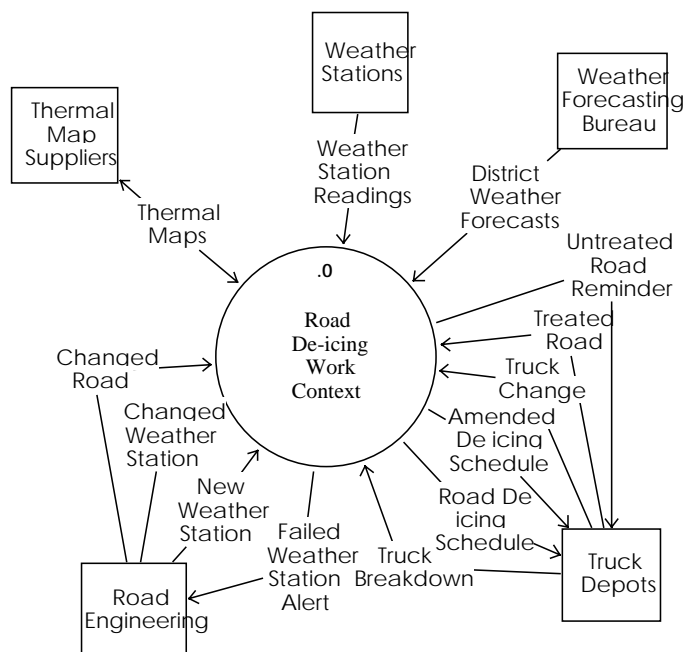
The work context diagram identifies the work that we need to investigate in order to be able to build the product. Note that this includes more than the intended product. Unless we understand the work that the product will support, there is little chance of building a product that will fit cleanly into its environment.

The adjacent systems on the example context diagram e.g. Weather Forecasting Bureau, indicate other subject matter domains (systems, people and organizations) that need to be understood. The interfaces between the adjacent systems and the work context indicate why we are interested in the adjacent system. In the case of Weather Forecasting Bureau, we can say that we are interested in the details of when, how, where, who and why they produce the District Weather Forecast information.

Motivation

To clearly define the boundaries for the work study and requirements effort. Without this definition, there is little chance of building a product that will fit seamlessly into its environment.

Examples



ITEA

Considerations

The names used on the context diagram should be consistent with the naming conventions discussed in Naming Conventions and Definitions on page 15.

Work partitioning

Content

An event list, identifying all the business events to which the work responds. The business events are user-defined. The response to each event represents a portion of work that contributes to the total functionality of the work.

The event list includes:

Event Name

Input from other systems (identical with name on context diagram)

Output from other systems (identical with name on context diagram)

Internal objects/entities that are connected to this business event. For example, both events 8 and 9 (see table below) would be connected to an internal object called road. In other words there is a need within the context to remember information about roads and that information is relevant to events 8 and 9 (and many other events as well). It is this identification of common internal objects that provides a link between events.

Motivation

To identify logical chunks of the system that can be used as the basis for discovering detailed requirements. These business events also provide the subsystems that can be used as the basis for managing detailed analysis and design.

Example

Event List:

Event Name	Input & Output	I/O
1. Weather Station transmits reading	Weather Station Readings	I
2. Weather Bureau forecasts weather	District weather Forecast	I
3. Road engineers advise changed roads	Changed Road	I
4. Road Engineering installs new weather station	New Weather Station	I
5. Road Engineering changes weather station	Changed Weather Station	I
6. Time to test Weather Stations	Failed Weather Station Alert	O
7. Truck Depot changes a truck	Truck Change Amended De-icing Schedule	I O
8. Time to detect icy roads	Road De-icing Schedule	O
9. Truck treats a road	Treated Road	I
10 Truck Depot reports problem with truck	Truck Breakdown Amended Gritting Schedule	I O
11. Time to monitor road gritting	Untreated Road Reminder	O

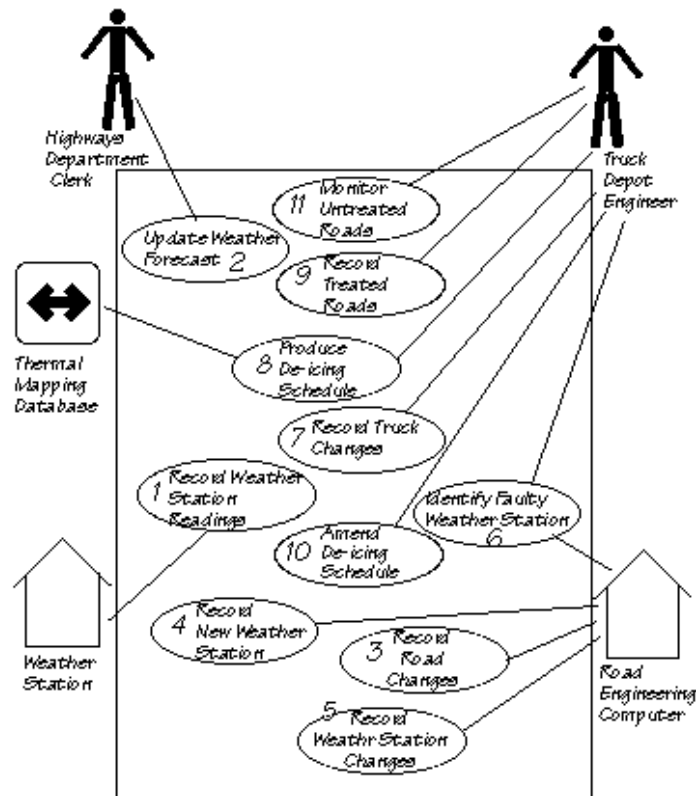
Considerations

Attempting to list the business events is a way of testing the work context. This activity uncovers uncertainty and misunderstanding about the project and helps with precise communications.

4.2 Scope of the Product

This section identifies the scope of the product in terms of the Use Cases.

Use case diagram identifies boundaries between the users and product.



You derive the product use cases by deciding where the product boundary should be for each one of the business events. These decisions are based on your knowledge of the work and the requirements constraints.

Use case list

The use case diagram is a graphical way of summarizing all the use cases relevant to the product. If you have a large number of use cases, we find 15-20, is around the limit, then it is better to list the use cases and model each one individually. For each use case on the list you should have: use case number, user/actor name, use case description and use case fit criterion. Also if you have built a use case description and/or any scenario models for this use case then this list can point to them.

Example:

Use Case 8

ITEA

User/actor name

Truck Depot Engineer

Description

Produce road de-icing schedule

Fit Criterion

Sensor readings shall be used to prepare a schedule for the de-icing trucks.

Use Case Scenarios

The description for this use case describes the normal way that it operates.

Additional scenario models could illustrate exception cases for this use case.

Each of the individual requirements that relates to this use case will contribute to meeting the fit criterion of the use case. Each individual requirement will also have its own detailed fit criterion.

4.3 Functional, Component and Data Requirements

Functional Requirements

Content

A specification for each individual functional requirement. As with all types of requirements, use the Requirements Shell. A full explanation is included in this template's introductory material.

Motivation

To specify the detailed functional requirements that must be supported by the system.

Fit Criterion

Each functional requirement must have a fit criterion. The fit criterion depends on the required action. For example, if the requirement is to record some data, then the fit criterion would say that the data must be able to be retrieved and must match certain standards. For calculations, the resulting data must conform to predicted results.

Considerations

If you have produced an event/use case list (see page 20) you can use them to help you trigger the functional requirements for each event/use case. If you have not produced an event/use case list, give each functional requirement a unique number and, to help with traceability, they can be partitioned into event/use case-related groups later in the development process.

Component and data requirements.

Content

A specification of the essential subject matter/business objects/entities/classes that are germane to the system.

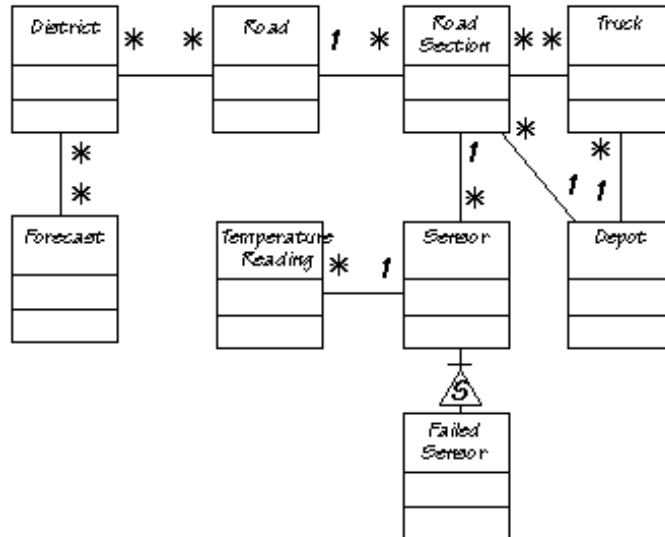
When the deliverables are one or more components, the capabilities of those are specified here.

This might take the form of a first-cut data model, an object model, or a domain model. Or it might be adequately dealt with by defining the terms in the dictionary described in section 3.2.2.

ITEA

Motivation

To clarify the system's subject matter and thereby trigger requirements that have not yet been thought of.



Fit Criterion

You can use any type of data or object model to capture this knowledge. The issue is to capture the meaning of the business subject matter and the connections between the individual parts and that you are consistent within your project. If you have an established company standard notation, then use that, as it will help you to reuse knowledge between projects.

To support your data model you would also define:

1. *Name of business object/entity (use naming convention from 5)*
2. *Statement of the purpose of the class/entity*
3. *Description of relationships between classes/entities*
4. *Attributes of the object/entity*

Considerations

Are there any data/object models for similar/overlapping systems that might be a useful starting point? Is there a domain model for the subject matter dealt with by this system?

4.4 Domain Model

A domain model a generic model of knowledge that applies to any product built for use in that domain.

4.5 Product-line requirements.

This section defines the product-line to which the product or component is to be part of.

The product-line specification and architecture specifications are identified.

5. NON-FUNCTIONAL REQUIREMENTS

5.1 Precedence and Priority requirements

Content:

This paragraph shall specify, if applicable, the order of precedence, criticality, or assigned weights indicating the relative importance of the requirements in this specification.

Motivation:

To support the definition of the contents and the functionality of the first and following increments of the product.

Example

The priorities are:

- *Basic functionality*
- *Trick mode*
- *Full functionality*
- *Full compliancy to specified standards*

5.2 Look and Feel Requirements

The interface

Content

A description of the spirit of the interface. This can be in the form of text descriptions or preliminary sketches of a possible interface.

The intention of this section is to state requirements relating to the interface. Your client may have given you particular demands such as style, colours to be used, degree of interaction and so on. This section captures the requirements for the interface rather than the design for the interface.

Motivation

To capture the expectations, the constraints and the client's demands for the interface before designing it.

Examples

The product shall have the same layout as the district maps that the engineering department uses now.

The product shall use the company colours.

The product shall be colourful and attractive to a teenage audience.

The product shall appear authoritative.

Considerations

ITEA

Interface design may overlap the requirements gathering process. This particularly true if you are using prototyping as part of your requirements process. As prototypes develop it is important to capture the requirements that relate to the look and feel. In other words, be sure that you understand your client's intentions for the product's look and feel. Record these as requirements instead of merely having a prototype to which the client has nodded his approval.

The style of the product

Content

A description of salient features of the product that are related to the way a potential customer will see the product. For example, if your client wants the product to appeal to the business executive, then a look and feel requirement is that the product has a conservative and professional appearance. Similarly if the product is for sale to children, then the look and feel requirement is that it be colourful and look like it's intended for children.

The requirements that you record here will guide the designers to produce a product as envisioned by your client.

Motivation

Given the state of today's market and people's expectations, we cannot afford to build products that have an inadequate appearance. Once the functional requirements are satisfied, it is often the appearance of products that determines whether they are successful or not. Your task in this section is to determine precisely how the product shall appear to its intended consumer.

Considerations

The look and feel requirements specify the your client's vision of the product's appearance. The requirements may at first seem to be rather vague – “conservative and professional appearance” – but these will be quantified by their fit criterion. The fit criterion in this case give you the opportunity to extract from your client precisely what is meant, and gives the designer precise instructions on what he is to accomplish.

5.3 Usability Requirements

Ease of use

Content

This section describes your client's aspirations for how easy it will be for the intended users of the product to operate it. The product's usability is derived from the abilities of the expected users of the product and the complexity of its functionality.

Motivation

To guide the product's designers into building a product that will meet the expectations of its eventual users.

Examples

The product shall be easy for 11 year-old children to use.

The product shall help the user to avoid making mistakes.

ITEA

The product shall make the users want to use it.

The product shall be used by people with no training, and possibly no understanding of English.

Fit Criterion

These examples may seem simplistic, but they do express the intention of the client. To completely specify what is meant by the requirement it is necessary to add a measurement of acceptance. We call this a fit criterion. The fit criterion for the above examples would be:

[An agreed percentage, say 90%] of a test panel of 11 year olds shall be able to successfully complete [list of tasks] within [specified time]

One month's use of the product shall result in a total error rate of less than [an agreed percentage, say 2%]

An anonymous survey shall show that [an agreed percentage, say 75%] of the users are regularly using the product after [an agreed time] familiarization period.

Considerations

Refer back to section Users of the Product to ensure that you have considered the usability requirements from the perspective of all the different types of users.

It may be necessary to have special consulting sessions with your users and your client to determine whether there are any special usability considerations that must be built into the product.

You could also consider consulting a usability laboratory that has experience with testing the usability of products that have constraints (sections Project drivers till Scope of the Work of this template) similar to yours.

Ease of learning

Content

A statement of how easy it should be to learn to use the product. This will range from zero time for products intended for placement in the public domain (for example a parking meter) to a considerable time for complex, highly technical products. (We know of one product where it was necessary for graduate engineers to spend 18 months in training before being qualified to use the product.)

Motivation

To quantify the amount of time that your client feels is allowable before a user can successfully use the product. This requirement will guide designers in how users will learn the product. For example, the designers may build elaborate interactive help facilities into the product, or the product may be packaged with a tutorial. Alternatively the product may have to be constructed so that all of its functionality is apparent upon first encountering it.

Examples

The product shall be easy for an engineer to learn.

A clerk shall be able to be productive within a short time.

The product shall be able to be used by members of the public who will receive no training before using it. They may have seen the advertising campaign.

ITEA

The product shall be used by engineers who will attend 5 weeks of training before using the product.

Fit Criterion

Fit criterion for the above example requirements are:

An engineer shall produce a [specified result] within [specified time] of beginning to use the product, without needing to use the manual.

After receiving [number of hours] training a clerk shall be able to produce [quantity of specified outputs] per [unit of time].

[Agreed percentage] of a test panel shall successfully complete [specified task] within [specified time limit].

The engineers shall achieve [agreed percentage] pass rate from the final examination of the training.

Considerations

Refer back to section Users of the Product to ensure that you have considered the ease of learning requirements from the perspective of all the different types of users.

5.4 Testing Requirements

Testing starts as soon as you start to specify the requirements. The first test is to determine if a requirement can be quantified by specifying its fit criterion. This fit criterion is an objective measure of the requirement's meaning. The criterion serves to evaluate whether or not a solution fits the requirement. If a fit criterion cannot be adequately specified, then the requirement is ambiguous or ill understood. If there is no fit criterion, there is no way of knowing if a solution matches the requirement.

This paragraph may also contain any information regarding a built-in test requirements. It shall include mechanisms for error detection, how, if at all, the error information shall be indicated and what the product shall do upon detection of a failure.

5.5 Performance and Safety Requirements

Speed requirements

Content

Specifies the amount of time available to complete specified tasks. These often refer to response times. They can also refer to the product's ability to fit into the intended environment.

Motivation

Some products, usually real-time products, must be able to perform some of their functionality within a given time slot. Failure to do so may mean catastrophic failure (for example a ground-sensing radar in an aeroplane fails to detect an upcoming mountain) or the product will not cope with the required volume of use (an automated ticket selling machine).

Examples

ITEA

“Any interface between a user and the automated system shall have a maximum response time of 2 seconds”

“The response shall be fast enough to avoid interrupting the user’s flow of thought”

“The product shall poll the sensor every 10 seconds”

“The product shall download the new status parameters within 5 minutes of a change”

Fit Criterion

Unit of measurement

Required range of values

Considerations

There is a wide variation in the importance of different types of speed requirements. If you are working on a missile guidance system then speed is extremely important. On the other hand, an inventory control report that is run once every 6 months has very little need for split second speed.

Customise this section of the template to give examples of the speed requirements that are important within your environment.

Safety critical requirements

Content

Quantification of perceived risk of possible damage to people, property and environment.

Motivation

To understand and highlight the potential damage that could occur when using the product within the expected operational environment.

Examples

The product shall not emit noxious gases that damage people’s health.

The heat exchanger shall be shielded from human contact.

Fit Criterion

Description of the perceived risk

Factors that could cause the damage

Unit for measuring the factors that could cause the damage

“The product shall be certified to comply with the Health Department’s standard E110-98. This is to be certified by qualified testing engineers.”

“No member of a test panel of [specified size] shall be able to touch the heat exchanger. The heat exchanger must also comply with safety standard [specify which one].”

Considerations

The sample requirements given above apply to some, but not all, products. It is not possible to give examples of every variation of safety critical requirement. To make the template work in your environment, you should customize it by adding examples that are specific to your products.

ITEA

If you are building safety critical systems then the relevant safety critical standards are already well specified. You will likely have safety experts on your staff. These safety experts are the best source of the relevant safety critical requirements for your type of product. The safety experts will almost certainly have copious information that you can use.

Note: if this paragraph is not applicable, it shall be stated clearly

Consult your legal department. They will be aware of the kinds of lawsuits that have resulted from product safety failure. This is probably the best starting place for generating relevant safety requirements.

Precision requirements

Content

Quantification of the desired accuracy of the results produced by the product.

Motivation

To set the client and user expectations for the precision of the product.

Examples

All monetary amounts shall be accurate to 2 decimal places.

Accuracy of road temperature readings shall be within + or - 2 degrees centigrade.

Fit Criterion

Unit of measure plus degree of precision

Considerations

If you have done any detailed work on definitions, then some precision requirements might be adequately defined by definitions in section Naming Conventions and Definitions.

Reliability and Availability requirements

Content

This section quantifies the necessary reliability of the product. It also includes the requirements on remote diagnosis, failure detection and safety-failure monitoring requirements.

This is usually expressed as the allowable time between failures, or the total allowable failure rate.

It quantifies the expected availability of the product.

A characterization of the acceptable responses to undesired events (exceptions or faults) is specified.

Motivation

It is critical for some products not to fail too often. This section allows you to explore the possibility of failure and to specify realistic levels of service. It also gives you the opportunity to set client and user expectations about the amount of time that the product will be available for use.

Examples

ITEA

The product shall be available for use 24 hours per day, 365 days per year.

The product shall be available for use between the hours of 8:00am and 5:30pm.

The escalator shall run from 6am until the last flight arrives at 10pm.

The product shall achieve 99% up time.

Fit Criterion

Time period/s of expected up-time

Considerations

Consider carefully whether the real requirement for your product is that it is available for use, or that it does not fail at any time.

Consider also the cost of reliability and availability, and whether it is justified for your product.

Capacity and Resource utilisation requirements

Content

This section specifies the volumes that the product must be able to deal with and the numbers of data stored by the product. It specifies the computer hardware resource utilisation, such as:

- 1. Maximum allowable use of processor capacity,*
- 2. Memory capacity, input/output device capacity,*
- 3. Auxiliary storage device capacity, and*
- 4. Communications/network equipment capacity.*

The requirements (stated, for example, as percentages of the capacity of each computer hardware resource) shall include the conditions, if any, under which the resource utilisation is to be measured.

Motivation

To ensure that the product is capable of processing the expected volumes.

Examples

The product shall cater for 300 simultaneous users within the period from 9:00am to 11:am. Maximum loading at other periods will be 150.

During a launch period the product shall cater for up to 20 people to be in the inner chamber.

Fit Criterion

In this case, the requirement description is quantified, and thus can be tested.

5.6 Operational Requirements

Expected physical environment

Content

This section specifies the physical environment in which the product will operate.

Motivation

ITEA

To highlight conditions that might need special requirements, preparations or training. These requirements ensure that the product is fit to be used in its intended environment.

Examples

The product shall be used by a worker, standing up, outside in cold, rainy conditions.

The product shall be used in noisy conditions with a lot of dust.

The product shall be able to fit in a pocket or purse.

The product shall be usable in dim light.

Considerations

The work environment Is the product to operate in some unusual environment? Does this lead to special requirements? Also see section Usability Requirements.

Expected technological environment

Content

Specification of the hardware and other devices that make up the operating environment for the new system.

Motivation

To identify all the components of the new system so that the acquisition, installation and testing can be effectively managed.

Considerations

Describe the hardware and other devices that make up the operating environment for the new system. This may not be known at the time of the requirements process, as these devices may be decided at design time.

It may be that the operating environment is complex, and becomes a subject of requirements study itself.

Special considerations should also be given if the product is to be embedded in a device.

If the expected operating environment is the same or similar to the current one, then this might be adequately covered in Implementation environment of the Current System.

Partner applications

Content

Description of other applications that the product must interface with.

Motivation

Requirements for interfacing to other applications often remain undiscovered until implementation time. Avoid a high degree of rework by discovering these requirements early.

Examples

“We must be able to interface with any html browser.”

“The new version of the spreadsheet must be able to access data from the previous 2 versions”

“Our product must interface with the applications that run on the remote weather stations”

ITEA

Fit Criterion

For each inter-application interface specify:

- *The data content*
- *The physical material content*
- *The medium that carries the interface*
- *The frequency*
- *The volume*

Required States and Modes

If the system or component is required to operate in more than one state or mode having requirements distinct from other states or modes, this paragraph shall identify and define each state and mode. The distinction between state and modes is arbitrary. It may be described in terms of states only, modes only, states within modes, modes within states, or any other scheme that is useful. If no states or modes are required, this paragraph shall so state, without the need to create artificial distinctions. If states or modes are required, each requirement or group of requirements in this specification shall be correlated to the states and modes.

5.7 Maintainability and Portability Requirements

Maintainability

Content

A quantification of the time necessary to make specified changes to the product.

Motivation

To make everyone aware of the maintenance needs of the product.

Examples

“New MIS reports must be available within one working week of the date the requirements are agreed”

“A new weather station must be able to be added to the system overnight”

Considerations

There may be special requirements for maintainability, such as this product must be able to be maintained by its end-users, or developers who are not the original developers. This has an effect on the way that the product is developed, and there may be additional requirements for documentation or training.

Special conditions

Content

Specification of the intended release cycle for the product and the form that the release will take.

ITEA

Motivation

To make everyone aware of how often it is intended to produce new releases of the product.

Examples

“The maintenance releases will be offered to end-users once a year. “

“Every registered user will have access to our help site via the Internet.”

Fit Criterion

Description of type of maintenance + Amount of effort budgeted

Considerations

Do you have any existing contractual commitments or maintenance agreements that might be affected by the new system?

Portability requirements

Content

Description of other platforms or environments to which the product must be ported.

Motivation

To quantify client and user expectations about the platforms on which the product will be able to run.

Examples

“The product is expected to run under Windows 95 and Unix”

“The product might eventually be sold to the Japanese market”

“The product is designed to run in offices, but we intend to have a version which will run in restaurant kitchens”

Fit Criterion

Specification of system software on which the product must operate.

Specification of future environments in which the product is expected to operate.

Time allowed to make the transition.

Considerations

Ask questions from your marketing department to discover unstated assumptions that have been made about the portability of the product.

Product adaptation requirements

Content

This section shall specify the requirements, if any, imposed on the product to make it adaptable to several environments and operational parameters that the product is required to use that may vary according to operational needs (such as parameters indicating operation-dependent targeting constants or data recording).

Motivation

ITEA

To scale a product to the target environment.

Examples

“nr of input buffers = “

“buffer size = “

“include security checks =”

5.8 Security Requirements

Confidentialness / Access constraints

Content

Specification of who has authorized access to the system, and under what circumstances that access is granted.

These requirements shall include:

- *the security/privacy environment in which the product must operate*
- *the type and degree of security or privacy to be provided*
- *the security/privacy risks the product must withstand*
- *required safeguards to reduce those risks*
- *the security/privacy policy that must be met*
- *the security/privacy accountability the product must provide*
- *the criteria that must be met for security/privacy certification/accreditation*

Motivation

To understand the expectations for confidentiality aspects of the system.

Examples

“Only direct managers can see the personnel records of their staff.”

“Only holders of current security clearance can enter the building.”

Fit Criterion

System function name or system data name

User role/s and/or names of people who have clearance

Considerations

Is there any data that is sensitive to the management? Is there any data that low level users do not want management to have access to? Are there any processes that might cause damage or might be used for personal gain? Are there any people who should not have access to the system?

Avoid solving how you will design a solution to the security requirements. For instance, don't design a password system. Your aim here is to identify what the security requirement is. The design will come from this description.

ITEA

Consider asking for help. Computer security is a highly-specialised field, and one where improperly-qualified people have no business being. If your product has need of more than average security, we advise that you make use of a security consultant. They are not cheap, but the results of inadequate security can be even more expensive.

Note: if this paragraph is not applicable, it shall be stated clearly

File integrity requirements

Content

Specification of the required integrity of databases and other files.

Motivation

To understand the expectations for the integrity of the system's data.

Examples

"The clients shall receive updated customer files once every 24 hours."

"Identical up-to-date booking information shall be available to all users of the system."

Considerations

How will the information be used? What is the impact on the customer's business if the information is out of date? Will there be a ripple effect if two different users have different versions of the system?

Audit requirements

Content

Specification of the required audit checks.

Motivation

To build a system that complies with the appropriate audit rules.

Considerations

This section may have legal implications. You are advised to seek the approval of your organization's auditors for what you write here.

5.9 Cultural and Political Requirements

Are there any special factors about the product that would make it unacceptable for some political reason?

Content

This section contains requirements that are specific to the sociological and political factors that affect the acceptability of the product. If you are developing a product for foreign markets then these requirements are particularly relevant.

Motivation

To bring out in the open requirements that are difficult to discover because they are outside the cultural experience of the developers.

ITEA

In the case of political requirements the requirements sometimes appear irrational.

Examples

The product shall be able to distinguish between French, Italian and British road numbering systems.

The product shall keep a record of public holidays for all countries in the European Union and for all states in the United States.

Our company policy says that we shall buy our hardware from company XXX.

The chief auditor shall verify all the user interfaces.

Considerations

Question whether the product is intended for a culture other than the one with which you are familiar. Ask whether people in other countries or in other types of organizations will use the product. Do these people have different habits, holidays, superstitions, cultural norms that do not apply to your own culture?

Did you intend to develop the product on a Macintosh, when the office manager has laid down an edict that only Windows machines are permitted?

Is a director also on the board of a company that manufactures products similar to the one that you intend to build?

Whether you agree with these political requirements has little bearing on the outcome. The reality is that the system has to comply with political requirements even if you can find a better/more efficient/more economical solution. A few probing questions here may save some heartache later.

5.10 Legal Requirements and Standards

Legal Requirements

Content

A statement specifying the legal requirements for this system..

Motivation

To comply with the law so as to avoid later delays, law suits and legal fees.

Examples

“Personal information shall be implemented so as to comply with the data protection act.”

Fit Criterion

Lawyers’ opinion that the product does not break any laws.

Considerations

Consider consulting lawyers to help identify the legal requirements?

Are there any copyrights that must be protected? Alternatively, do any competitors have copyrights that you might be in danger of infringing?

ITEA

Is it a requirement that developers have not seen competitors' code or even have worked for competitors?

Is there any pending legislation that might affect the development of this system?

Standards

Content

A statement specifying applicable standards and referencing detailed standards descriptions.

Motivation

To comply with standards so as to avoid later delays.

Example

"The product shall comply with MilSpec / DICOM standards."

"The product shall comply with the Microsoft XX standard".

"The product shall be developed according to ITEA DESS standard development steps."

Fit Criterion

The appropriate standard-keeper that the standard has been adhered to.

Considerations

It is not always apparent that there are applicable standards because their existence is often taken for granted. Consider the following:

Are there any industry bodies that have applicable standards?

Has the industry a code of practice, watchdog or ombudsman?

Are there any special development steps for this type of product?

The section shall define a set of qualification methods and shall specify for each requirement the method(s) to ensure the requirement has been met.

6. PROJECT ISSUES

6.1 Open Issues

Issues that have been raised and do not yet have a conclusion.

Content

A statement of factors that are uncertain and might make significant difference to the product.

Motivation

To bring uncertainty out in the open and provide objective input to risk analysis.

Examples

“Our investigation into whether or not the new version of the processor will be suitable for our application is not yet complete.”

“The government are planning to change the rules about who is responsible for gritting the motor ways, but we do not know what the changes might be.”

Considerations

Are there any issues that have come up from the requirements gathering that have not yet been resolved? Have you heard of any changes that might occur in the other organisations/systems on your context diagram? Are there any legislative changes that might affect your system? Any rumours about your hardware/software suppliers that might have an impact?

6.2 Off-the-shelf Solutions (ready-made systems/components)

6.2.1 Is there a ready-made system or component that could be bought?

Content

List of existing products that should be investigated as potential solutions. Reference any surveys that have been done on these products.

Motivation

To give consideration to whether or not a solution can be bought.

Considerations

Is it possible to buy something that already exists or is about to become available? It may not be possible at this stage to say with a lot of confidence, but any likely products should be listed here.

Also consider whether there are products that must not be used.

6.2.2 Can ready-made components be used for this product?

Content

Description of the candidate components, either bought-in or built by your company, that could be used by this project. List libraries that could be a source of components.

ITEA

Motivation

Reuse rather than reinvention.

6.2.3 Is there something that we could copy?

Content

List of other similar systems.

Motivation

Reuse rather than reinvention.

Examples

“Another electricity company has built a customer service system. Their hardware is different from ours but we could buy their specification and cut our analysis effort by approximately 60%.”

Considerations

While a ready-made solution may not exist, there may well be something that, in its essence, is similar enough that you could copy, and possibly modify, to better effect that starting from scratch. This is dangerous because it relies on the base system being of good quality.

This question should always be answered. The act of answering will force you to look at other existing solutions to similar problems.

6.3 New Problems.

6.3.1 What problems could the new system cause in the current environment?

Content

A description of how the new system will affect the current implementation environment. This section should also cover things that the new product should not do.

Motivation

The intention is to discover early any potential conflicts that might otherwise not be realised until implementation time.

Examples

Any change to the scheduling system will affect the work of the engineers in the divisions and the gritter truck drivers.

Considerations

Is it possible that the new system will damage some already existing system? Can people be displaced, or affected by the new system?

This requires a study of the current environment. A model highlighting the effects of the change is a good way to make this information widely understandable.

6.3.2 Will the new development affect any of the installed system?

Content

ITEA

Specification of the interfaces between new and existing systems.

Motivation

Very rarely is a new development intended to stand completely alone. Usually there is some existing system that the new one must coexist with. This question forces you to look carefully at the existing system and examine it for potential conflicts with the new development.

6.3.3 Will any of our existing users be adversely affected by the new development?

Content

Details of any adverse reaction that might be suffered by existing users

Motivation

Sometimes existing users are using a product in such a way that they will suffer ill effects from the new system/feature. Identify any likely adverse user reaction, determine whether we care and what precautions we will take.

6.3.4 What limitations exist in the anticipated implementation environment that may inhibit the new system?

Content

Statement of any potential problems with the new automated technology or new ways of structuring the organisation.

Motivation

The intention is to make early discovery of any potential conflicts that might otherwise not be realised until implementation time.

Examples

The planned new server is not powerful enough to cope with our projected growth pattern.

Considerations

This requires a study of the intended implementation environment.

6.3.5 Will the new system create other problems?

Content

Identification of situations that we might not be able to cope with.

Motivation

To guard against situations where the product might fail.

Considerations

Will we create a demand for our product that we are not able to service? Will the new system cause us to fall foul of laws that do not currently apply? Will the existing hardware cope?

There are potentially hundreds of unwanted effects. It pays to answer this question very carefully.

6.4 Tasks

The specification of development phases and the breakdown of the work into tasks is described in the Project Management Plan [PMP].

What steps have to be taken to deliver the system?

Content

Details of the life cycle and approach that will be used to deliver the product. A high level process diagram showing the tasks and interfaces between them is a good way to communicate this information.

Motivation

To specify the approach that will be taken to deliver the product so that everyone has the same expectations.

Considerations

Depending on the level of maturity of your process, the new product will be developed using your standard approach. However, there are some circumstances that are special to a particular product and will necessitate changes to your lifecycle. While these are not a product requirement, they are needed if the product is to be successfully developed.

If possible, attach an estimate of the time and resources need for each task based on the requirements that you have specified. Tag your estimates to the events/use cases/functions that you specified in sections Scope of the Work and Functional, Component and Data Requirements.

Do not forget data conversion, user training and cutover. We have listed these because they are usually ignored when projects set implementation dates.

Development phases

Content

Specification of each phase of development and the components in the operating environment.

Motivation

To identify the phases necessary to implement the operating environment for the new system so that the implementation can be managed.

Fit Criterion

Name of the phase

Required operational date

Operating environment components included

Functional requirements included

Non-functional requirements included

Considerations

ITEA

Identify which hardware and other devices are necessary for each phase of the new system. This may not be known at the time of the requirements process, as these devices may be decided at design time.

6.5 Implementation procedures

This section describes the activities required to ensure that the product is integrated into the user's environment and processes (design-in).

6.5.1 What special requirements do we have to get the existing data, and procedures to work for the new system?

Content

A list of the Cutover activities. Timetable for implementation.

Motivation

To identify cutover tasks as input to the project planning process.

Considerations

Will you be using phased implementation to install the new system? If so, describe the requirements that will be implemented by each of the major phases.

What data conversion has to be done? Are there special programs to be written to transport data from an existing system to the new one? If so, the requirements for this program(s) are to be described here.

What manual backup is needed while the new system is installed?

When are each of the major components to be put in place, when are phases of the implementation to be released?

This section is the timetable for implementation of the new system.

6.5.2 What data has to be modified/translated for the new system?

Content

List of data translation tasks.

Motivation

To discover missing tasks that will affect the size and boundaries of the project.

Fit Criterion

Description of the current technology that holds the data

Description of the new technology that will hold the data

Description of the data translation task/s

Foreseeable problems

Considerations

Every time you make an addition to your dictionary ask the question "What are all the places that this data is currently held and will the new system affect that implementation?"

6.6 Risks

Project risks are described in the Project Management Plan [PMP].

All projects involve risk. By this we mean the risk that something will go wrong. Risk is not necessarily a bad thing, as no progress is made without taking some risk. However, there is a difference between unmanaged risk – say shooting dice at a craps table – and managed risk where the probabilities are well understood, and contingencies made. Risk is only a bad thing if the risks are ignored and they become problems. Risk management is assessing which risks are most likely to apply to the project, deciding a course of action if they become problems, and monitoring projects to give early warnings of risks becoming problems.

This section of your specification should contain a list of the most likely and the most serious risks for your project. Against each risk include the probability of that risk becoming a problem. Capers Jones' book *Assessment and Control of Software Risks*. Prentice-Hall, Englewood Cliffs, NJ. 1994 gives comprehensive lists of risks and their probabilities, you can use these as a starting point. For example, Jones cites the following risks as being the most serious:

- Inaccurate metrics
- Inadequate measurement
- Excessive schedule pressure
- Management malpractice
- Inaccurate cost estimating
- Silver bullet syndrome
- Creeping user requirements
- Low quality
- Low productivity
- Cancelled projects

It is also useful input to project management if you include the impact on the schedule, or the cost, if the risk does become a problem.

6.7 Costs

6.7.1 Project costs are described in the Project Management Plan [PMP]

The other cost of requirements is the amount of money or effort that you have to spend building them into a product. Once the requirements specification is complete, you can use one of the estimating methods to assess the cost, and express this in a monetary amount or time to build.

There is no best method to use when estimating. However your estimates should be based on some tangible, countable, artefact. If you are using this template then, as a result of doing the work of requirements specification, you are producing many measurable deliverables. For example:

ITEA

- Number of input and output flows on the work context
- Number of business events
- Number of product use cases
- Number of functional requirements
- Number of non-functional requirements
- Number of requirements constraints
- Number of function points

The more detailed work you do on your requirements the more accurate will be your deliverables. Your cost estimate is the amount of resources you estimate each type of deliverable will take to produce within your environment. You can do some very early cost estimates based on the work context. At that stage, your knowledge of the work will be general and you should reflect this by making the cost estimate a range rather than one figure.

As you get more knowledge about the requirements we suggest you try using feature point counting.

It is important that your client knows at this stage what the product is likely to cost. You usually express this as a total cost to complete the product, but you may also find it advantageous to be able to point out the cost of individual requirements.

Whatever you do, do not leave the costs in the lap of hysterical optimism. Make sure that this section includes meaningful numbers based on tangible deliverables.

6.8 Product- User documentation

User Manual

Content:

Describe the purpose and contents of the User Manual. Discuss desired length, level of detail; need for index, glossary of terms, tutorial vs. reference manual strategy, etc. Formatting and printing constraints should also be identified

Example:

For maintenance employees a user maintenance manual must be available with a complete description of the hardware and software maintenance.

Procedures and a reference guide with maintenance sensitive points. The maintenance manual should contain an index and a glossary of terms.

On-line Help

Many systems provide an on-line help system to assist the user. The nature of these systems is unique to application development as they combine aspects of programming (hyperlinks, etc) with aspects of technical writing (organization, presentation). Many have found the development of on-line help system is a project within a project that benefits from up front scope management and planning activity.

Installation Guides, Configuration, Read me files

A document that includes installation instructions and configuration guidelines is important to a full solution offering. Also, a Read Me file is typically included as a standard component. The Read Me can include a "What's New With This Release Section," and a discussion of

ITEA

compatibility issues with earlier releases. Most users also appreciate documentation defining any known bugs and workarounds in the Read Me file

Production and Service Guides

A document that includes production and/or service guidelines is important to a full solution offering.

Considerations:

Think of:

- *Mechanical and electrical maintenance or service manuals*
- *Software maintenance (back-up, restore,....)*
- *Documentation of purchased parts*
- *List of spare parts*
- *Construction dossier*

Training

A description of the required training and training materials

Considerations:

Think of:

- *Duration of the training*
- *For who (number of persons, which function (fitter, operator, service people, etc.)*
- *By who*
- *Where (At Supplier, Customer, ...)*
- *When (Before or after acceptance, installation)*
- *Training materials*
- *On-line or off-line (during production, or not)*

Labeling and packaging

Today's state of the art applications provide a consistent look and feel that begins with product packaging and manifests through installation menus, splash screens, help systems, GUI dialogs, etc. This section defines the needs and types of labeling to be incorporated into the code. Examples include copyright and patent notices, corporate logos, standardized icons and other graphic elements, etc

6.9 Waiting Room

This section describes requirements that will not be part of the agreed product. These requirements might be included in future versions of the product.

Content

Any type of requirement.

Motivation

To allow requirements to be gathered, even though they cannot be part of the current development. To ensure that good ideas are not lost.

Considerations

The requirements gathering process often throws up requirements that are beyond the sophistication of, or time allowed for, the current release of the product. This section is a hold-all for requirements in waiting. The intention is to avoid stifling your users and clients by having a repository of future requirements. You are also managing expectations by making it clear that you take these requirements seriously but they will not be part of the agreed product.

6.10 Ideas for Solutions

This section describes relevant ideas to ensure that they are not lost.

When you are gathering requirements you are focusing on finding out what the real requirements are, you are trying to avoid coming up with solutions. However when creative people start to think about a problem they always have ideas. This section of the template is a place to put those ideas so that you do not forget them and so that you can separate them from the real business requirements.

Content

Any idea for a solution that you think is worth keeping for future consideration. This can take the form of rough notes, sketches, pointers to other documents, pointers to people, pointers to existing products....the aim is to capture, with the least amount of effort, an idea that you can come back to later.

Motivation

To make sure that good ideas are not lost and to help you separate requirements and solutions.

Considerations

Whilst you are gathering requirements you will have solution ideas, this is a way of capturing them. Bear in mind that this section will not necessarily be published as part of the specification that you publish.

7. PROCESS REQUIREMENTS

7.1 Requirement characteristics and traceability

This section contains a number of tables related to the traceability of requirements and their characteristics.

This table can be used to add information suggested in the Volere Requirement Shell, or other useful information, such as a complexity weighting and risk.

Requirement	Source	Priority	Complexity	Etc.

Traceability from each system requirement to component requirement and vice versa could be a requirement for the development process.

7.2 Process improvement requirements

Content

Requirements for the project, resulting from any improvement program

Motivation

To enable the environment of the project to improve the way of working

Fit Criterion

X% of effort should be spent on software improvement activities

Project team members should participate in planned maturity evaluations

Considerations

Although a development project may have fixed begin- and end dates, the environment in which the project is executed may want to improve its way of working

ITEA

8. APPENDIX A, REQUIREMENT FORM

The figure below shows the format for writing each requirement.

Requirement Form

Requirement #: Unique Id	Type: Section #		
Description: A one sentence requirement statement			
Rationale: A justification of the requirement			
Origin: Who raised this requirement			
Fit criterion: A measurement of the requirement, such that it is possible to test if the solution matches the original requirement.			
Assigned To: The person who is responsible for solving this requirement			
Satisfaction:	Dissatisfaction:		
Dependencies			
Req #:	Req #:		
Description:	Description:		
References: References to documents that describe or explain this requirement			
Priority:	Status:		
Cost:	Difficulty:		
	Stability:		
Traceability			
From		To	
UC #	Description	UC #	Description
Event #		Event #	
History			
Date	Version	Author	Description

8.1 Requirement numbering

Give each requirement a unique identifier to make it traceable throughout the development process. The numbering scheme suggested in the requirement form is:

- Requirement #, is the next unique requirement number
- Requirement type, is the section number from the template for this type of requirement.

The inclusion of the section number is not absolutely necessary because we do have a unique requirement id. However it serves as a reminder of what this requirement relates to and helps to remind why the requirement is considered important. Also the ability to compare requirements of the same type makes it easier to identify contradictions and duplications.

For example:

A functional requirement comes from section 9, and the next unique number is 128.

Requirement #: 128 Requirement Type: 9

We shall record the time when we are notified of a gritter truck breakdown

A performance requirement comes from section 12, and the next unique number is 129.

Requirement #: 129 Requirement Type: 12

Gritter truck drivers shall be informed of their schedule 30 minutes before leaving the depot.

8.2 Event/use case

is the identifier of a business event or use case that contains this requirement. There might be several Event/use case #'s for one requirement because the same requirement might relate to a number of events. The terms event and use case are already widely used in the systems development world. We use the term business event to mean a business related happening that causes an event-response within the work that we are studying.

We use the term event-driven use case (or product use case) to mean a user-defined (or actor defined) piece of activity within the context of the product. Business events and product use cases provide a way of grouping business-related requirements and tracing them through into implementation; they are used throughout the Volere development process.

8.3 Fit Criterion.

You start testing requirements as soon as you start writing them.

Your first test is to determine if you can quantify the requirement by specifying its fit criterion. This *fit criterion* is an objective measure of the requirement's meaning; it is the criterion for evaluating whether or not a given solution fits the requirement. If a *fit criterion* cannot be adequately specified, then the requirement is ambiguous, or ill understood. If there is no *fit criterion*, then there is no way of knowing if a solution matches the requirement.

8.4 Satisfaction/dissatisfaction

Customer Value is a measure of how much your client cares about each requirement. Ask your stakeholders to grade each requirement for Customer Satisfaction on a scale from 1 to 5 where 1

ITEA

means mild interest if this requirement is satisfactorily implemented, and 5 means they will be very happy if this requirement is satisfactorily implemented. The stakeholders also grade each requirement for Customer Dissatisfaction on a scale from 1 to 5 where 1 means that it hardly matters, and 5 means that they will be extremely displeased if this requirement is not satisfactorily implemented. The point of having a satisfaction and a dissatisfaction rating is that it guides your clients to think of the requirement from two different perspectives, and helps you to uncover what they care about most deeply.

8.5 Dependencies/conflicts

Dependencies keep track of other requirements that have an impact on this requirement. If the dependency exists because requirements use the same information, then use of standard naming conventions and definitions (see Section 5) will implement this dependency. Other dependencies exist because a solution to this requirement has a positive or negative effect on solutions to other requirements. Capture these types of dependencies by cross referencing the requirements. Some requirements, especially project drivers and project constraints, have an impact on all the other requirements.

Conflicts keep track of other requirements that disagree with this one. Conflicts that are caused by mistake are solved simply by bringing them to the surface and resolving them. Other conflicts are because of true differences in opinion/intention. These are the conflicts that might eventually need to be addressed using negotiation or mediation techniques. There is nothing wrong with having conflicting requirements providing you know that you have them. Then you are in a position to address the conflict.

8.6 Priority

A priority is assigned to each requirement. Suggested values to be used are: **High, Medium and Low**. Priorities will help you to define the contents of the first and following increments of the system.

8.7 Status

We will keep track of the status of each requirement. Suggested values for the statuses are: **Proposed, Approved, Incorporated, Validated and Waiting room**. Depending on the requirement type other values may be possible.

8.8 Difficulty / Stability

Suggested values for these requirement attributes are: **High, Medium and Low**.

8.9 Traceability

Event # or UC # is the identifier of a business event or use case that contains this requirement. There might be several Event/use case #'s for one requirement because the same requirement might relate to a number of events. The terms event and use case are already widely used in the systems development world. We use the term business event to mean a business related happening that causes an event-response within the work that we are studying. We use the term event-driven use

ITEA

case (or product use case) to mean a user-defined (or actor defined) piece of activity within the context of the product. Business events and product use cases provide a way of grouping business-related requirements and tracing them through into implementation.

8.10 History

We follow the requirement from the date that it was created, through all its changes. We minimize future confusion by recording the rationale for making major changes. When a requirement is deleted we record when and the rationale behind the deletion.

9. APPENDIX B, DEFINITIONS

Context of the Product

The boundaries between the product that we intend to build and the people, organizations, other products and pieces of technology that have a direct interface with the product.

Context of the Work

The subject matter, people and organizations that might have an impact on the requirements for the product. The context of study identifies the intersection of all the domains of interest.

Client

The person or organization for whom the product is being built, usually responsible for paying for the development of the product.

Customer

The person or organization that will buy the product (notes that the same person/organization might play the client, customer and sometimes user roles)

Design or Systems Design

Crafting a solution to fit the requirements.

Developers

The people who specify and build the product.

Domain of Interest

A subject matter area that has some relevance to the context of study.

Non-Functional Requirement

A property that the eventual product must have.

Event

We use the term business event to mean a business related happening within a system adjacent to the work that we are studying. The happening causes the work to produce an event-response.

Fit Criterion

Objective measure for defining the meaning of a requirement, and eventually testing whether a given solution satisfies the original requirement.

Functional Requirement

An action that the product must be able to take, something that the product must do.

Global Constraint

Constraints that apply to the system as a whole.

Product

This is what we are attempting to deliver. This could be a piece of software, the installation of a package, a set of procedures, a piece of hardware, a piece of machinery, a new organization, or almost anything.

Requirement

A measurable statement of intent about something that the product must do, or a property that the product must have, or a constraint on the system.

Stakeholder

A stakeholder is a person who can affect the outcome/success of the project and/or is affected by its outcome/success.

System

The business system whose requirements are being studied.

ITEA

Systems Analysis

Detailed study of the requirements, intended to prove their workability as input to systems design.

Use case

We use the term event-driven use case (or product use case) to mean a user-defined (or actor defined) piece of activity within the context of the product.

User or End User

Someone who has some kind of direct interface with the product.

10. APPENDIX C, USE CASE TEMPLATE

The template below is an example on how to describe each use case.

Use Case Template

UC#:	Name: <A short active verb or phrase which shows the goal of the UC>
-------------	---

Context of use:	<A longer statement of the context of use, if needed>
------------------------	---

Scope:	<What system is being considered>
---------------	-----------------------------------

Level:	<one of: Summary, Primary task, Subfunction>
---------------	--

Primary Actor:	<a role name for the primary actor or description>
-----------------------	--

Stakeholders & Interests	
Stakeholder	Interest
<Stakeholder Name>	<Stakeholder interests>

Preconditions:	<What we expect is already the state of the world>
-----------------------	--

Minimal Guarantees:	<The interests as protected on any exit>
----------------------------	--

Success Guarantees:	<The interests as satisfied on successful ending>
----------------------------	---

Trigger:	<The action upon the system that starts the use case>
-----------------	---

Main Success Scenario	
Step	Action
<Step #>	<Put here the steps of the scenarion from trigger to goal delivery, and any cleanup after>

Extensions	
Step	Branch Action
<Step #>	<Condition causing action: action name of sub-use case>

Technology and Data Variations	
Step	Variations
<Step #>	<Put here the steps of the variations>

Related Information:	
-----------------------------	--

Open Issues:	
Issue	Description
<Issue #>	<Description of issue to be solved>