How to deal with Critical Quality Requirements, Design and testing using Agile Development.

Tom Gilb –
Roots, Bergen
Monday April 27th 2009
1400-1700 (3 hours)

Workshop Content

- How to deal with Critical Quality Requirements, Design and testing using Agile Development.
  - This practical workshop will give the participant practical specification templates, processes, rules and other standards   for requirements and design.
  - The requirement/design templates will show a practical way to connect all requirements, at all levels, their corresponding design and architecture, and their corresponding test plans.
  - The participant will experience a powerful planning language for requirements and design.

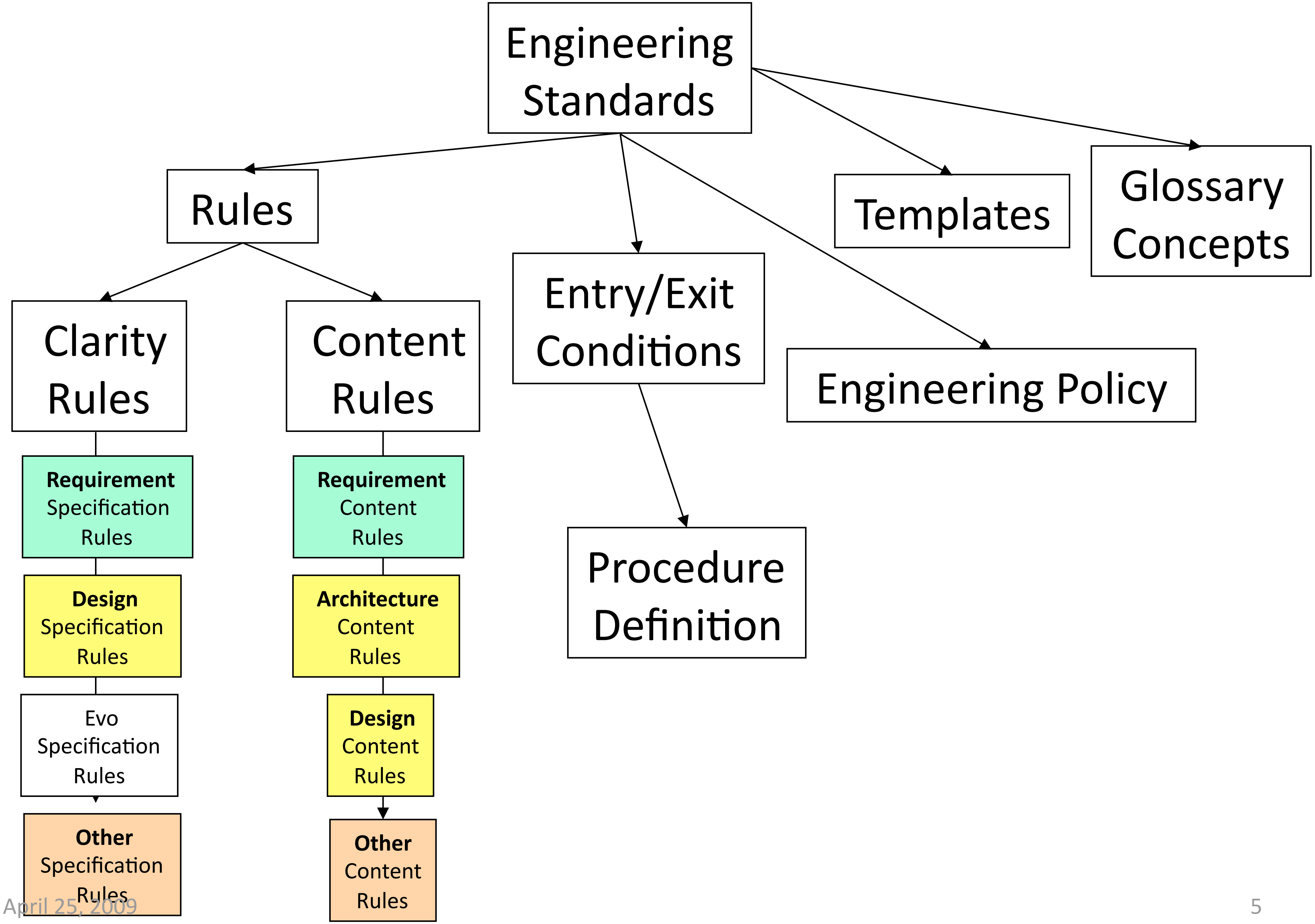
Source: Competitive Engineering Book

TOM GILB
COMPETITIVE
ENGINEERING
A HANDBOOK FOR SYSTEMS ENGINEERING, REQUIREMENTS ENGINEERING, AND SOFTWARE ENGINEERING USING PLANGUAGE
ELSEVIER
GILB

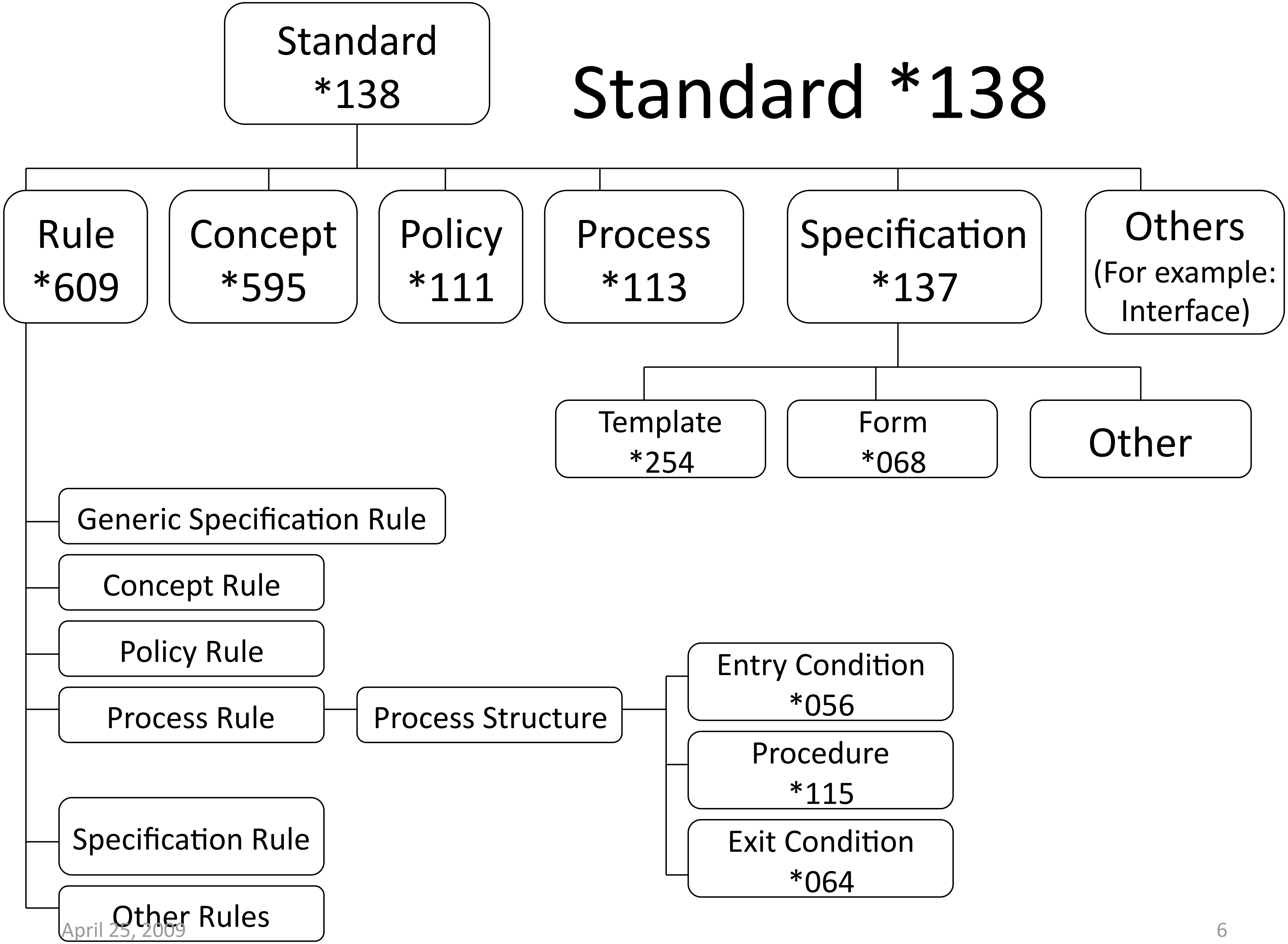
Rough Timing

- 14:00 – 14:50 Requirements
- Break
- 15 – 15:50 Design
- Break
- 16- 1700 Impact Estimation Tables
- End

Generic Standards Overview



Standard *138



14:00 – 14:50 Requirements

Process: Requirement Specification. <-CE 2

Tag: Process.RS. Version: X. Owner: TG. Status: Draft.

Procedure

P1: Define the system scope and the overall scope of the requirements.

P2: Identify relevant (critical and profitable) stakeholders.

P3: Determine the requirements of each type of stakeholder. Ensure all specification statements are sources referenced.

P4: Categorize requirements by type (The major requirement types are function, performance, cost and constraint).

P5: *(‘Stakeholder Value’)* Specify *Functional Requirements* (Process.FR. See Chapter 3).

P6: *(‘Stakeholder Value’)* Specify *Quality Requirements* (Process.QR. See Chapter 4) including identifying or creating a *Scale Definition* (Process.SD. See Chapter 5).

Specify other Performance requirements (Capacity and Savings) in a similar manner.

P7: *(‘Stakeholder Constraint’)* Specify *Cost Requirements* (Process CR. See Chapter 6).

P8: *(‘Stakeholder Constraint’)* Identify and question any constraints. (Are they real or was something else intended?) Specify the necessary *Constraints* (Process CT. See Ch. 6).

P9: Specify all known significant relationships of the requirements to any other relevant requirements specifications (external or internal to the system). *You need to identify where there may be overlap or conflict or double accounting over benefits. There may even be synergy or a chance to ‘subcontract’ parts of the system development.*

Use Planguage terms such as {Source, Depends, Assumption, Authority, Impacts, Risk, Impacted By}.

P10: Get stakeholders to approve written requirements’ specifications that specifically affect them.

P11: Carry out quality control on the requirement specifications. At least, analyze them by sampling. Using Specification Quality Control (SQC), they can exit at an appropriate low level of remaining major defects/page (such as a maximum of 1 major defects/page).

Note: this is an appropriate point in this procedure to carry out quality control. However, don’t let this prevent you from carrying out quality control at other times. Far better you find out that there is a problem after writing three pages than after thirty pages.

Note: For the majority of the procedures in this book, the exit and entry conditions serve to remind you about the need for quality control: explicit reference to quality control within the main procedure is omitted.

P12: Once the requirements have exited quality control, review them with the aim of obtaining the relevant management approval. *(SQC checks the specification quality, ‘review’ checks the business relevance.)*

Entry Conditions

E1: Generic Entry Conditions apply. The specification quality control (SQC) entry condition applies to any source information, such as contracts and marketing plans.

E2: Key stakeholders are available for questions and reviews to resolve any uncertainty about sources and exact specification.

Exit Conditions

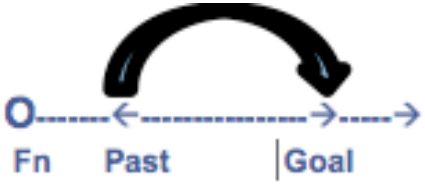
X1: Generic Exit Conditions apply. The requirement specification must have exited SQC.

X2: There is management *review* approval of the requirement specification.

‘Requirement’: Defined

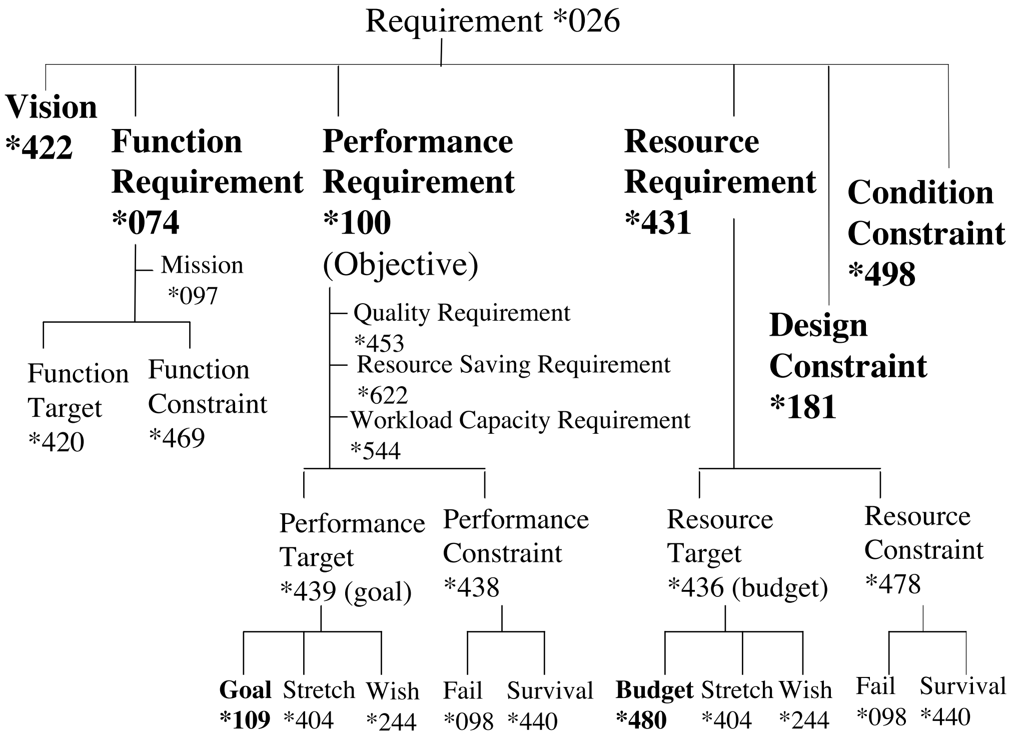
Concept *026 Version January 23rd 2008

- A ‘requirement’ is a
  - **“stakeholder-prioritized future state”.**



- **Some consequences of this definition:**
  - **requirements are *not* ‘absolute’**
  - **a requirement’s effective priority’ is *variable*, and depends on *many* factors, like**
    - **Value of doing it, cost of doing it, related constraints,**
    - **stakeholder power, formal requirement inclusion.**
  - **Planguage helps you intelligently manage requirement priorities, so that you get maximum value for your limited resources (= ‘competitiveness’).**

Some Formally Defined Requirement Concepts and types



Requirement *026

Vision *422

Function Requirement *074

Mission *097

Function Target *420

Function Constraint *469

Performance Requirement *100 (Objective)

Quality Requirement *453

Resource Saving Requirement *622

Workload Capacity Requirement *544

Performance Target *439 (goal)

Performance Constraint *438

Goal *109 Stretch *404 Wish *244

Fail *098 Survival *440

Resource Requirement *431

Resource Target *436 (budget)

Resource Constraint *478

Budget *480 Stretch *404 Wish *244

Fail *098 Survival *440

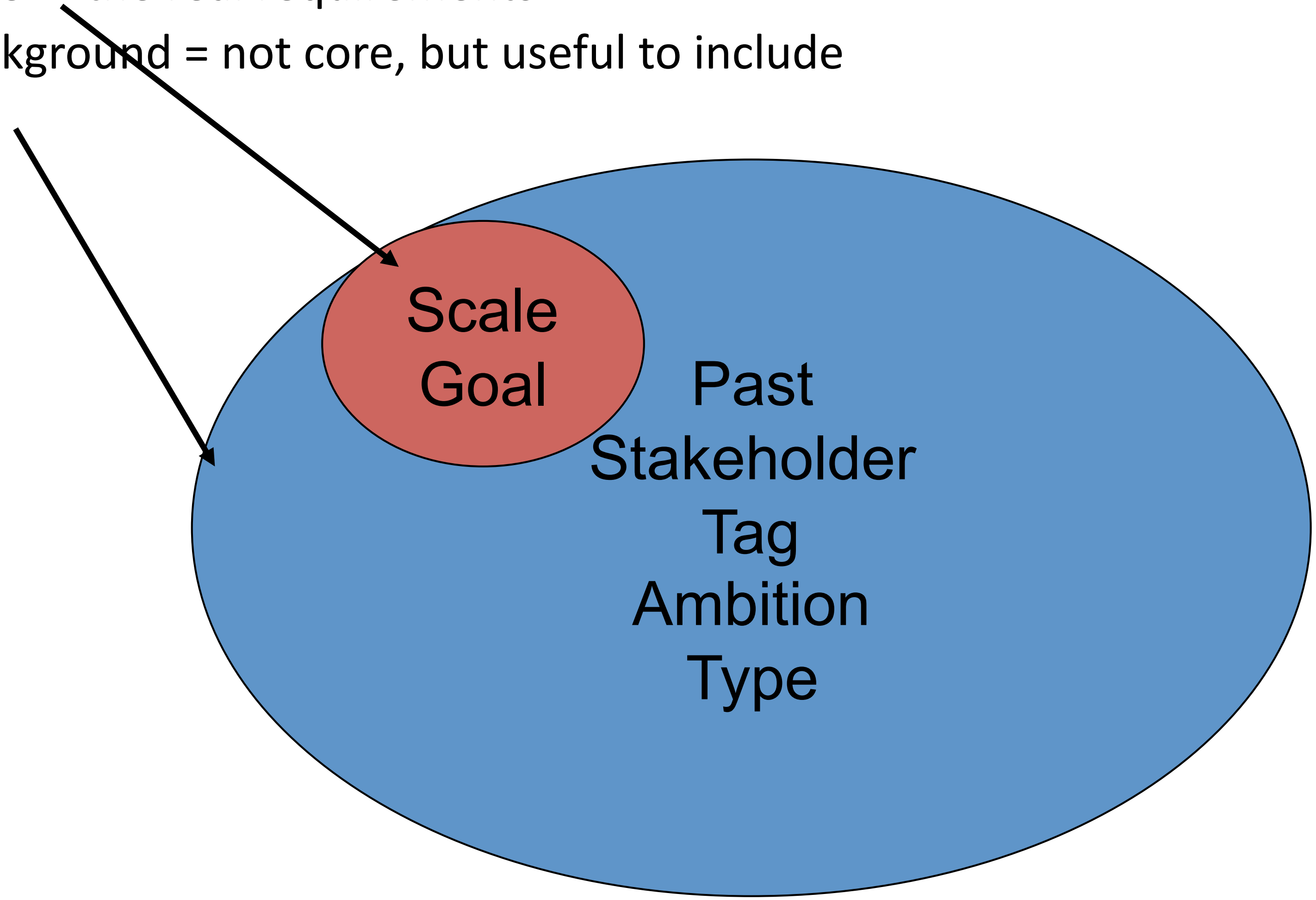
Design Constraint *181

Condition Constraint *498

Background and Core

Core = ‘the real requirements”

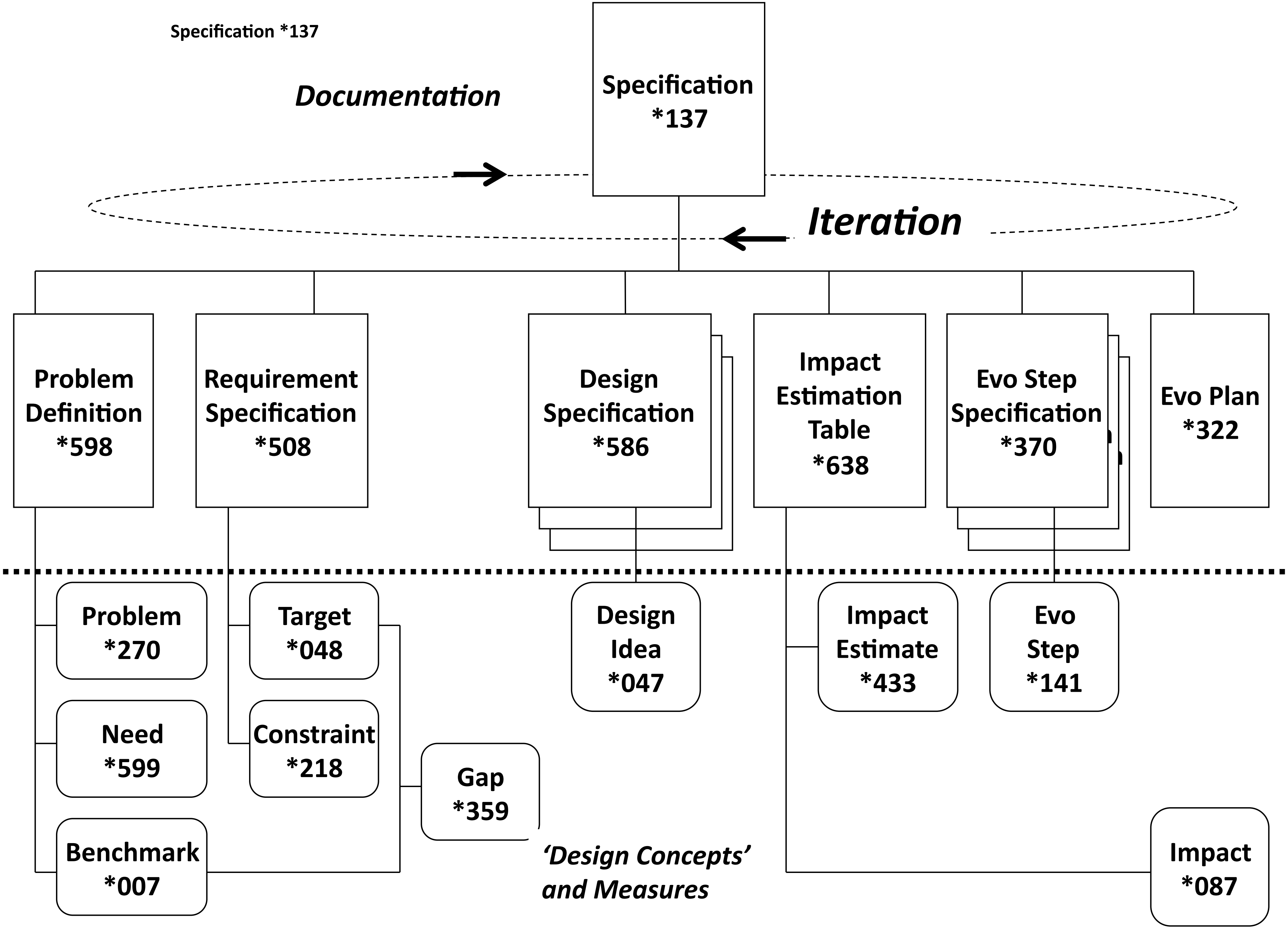
Background = not core, but useful to include



Scale
Goal

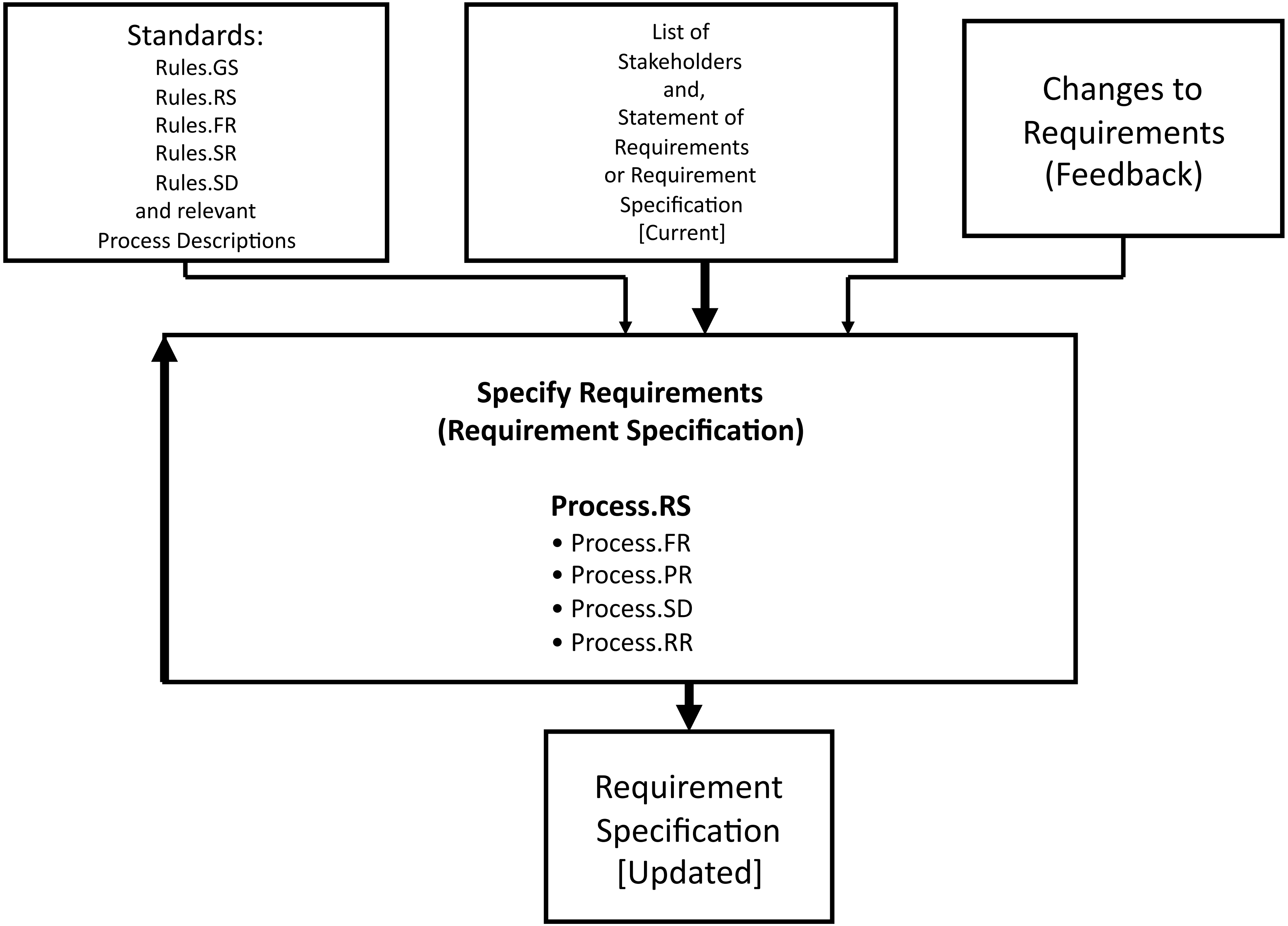
Past
Stakeholder
Tag
Ambition
Type

Specification *137



1. Planguage standards for specification

- ◆1.1 Policies
- ◆1.2 Rules
- ◆1.3 Checklists
- ◆1.4 Models
- ◆1.5 Templates
- ◆1.6 Forms
- ◆1.7 Process
  - ◆(Entry Conditions, Procedure Description, Exit Conditions)



Standards:
Rules.GS
Rules.RS
Rules.FR
Rules.SR
Rules.SD
and relevant
Process Descriptions

List of
Stakeholders
and,
Statement of
Requirements
or Requirement
Specification
[Current]

Changes to
Requirements
(Feedback)

Specify Requirements
(Requirement Specification)

Process.RS

- Process.FR
- Process.PR
- Process.SD
- Process.RR

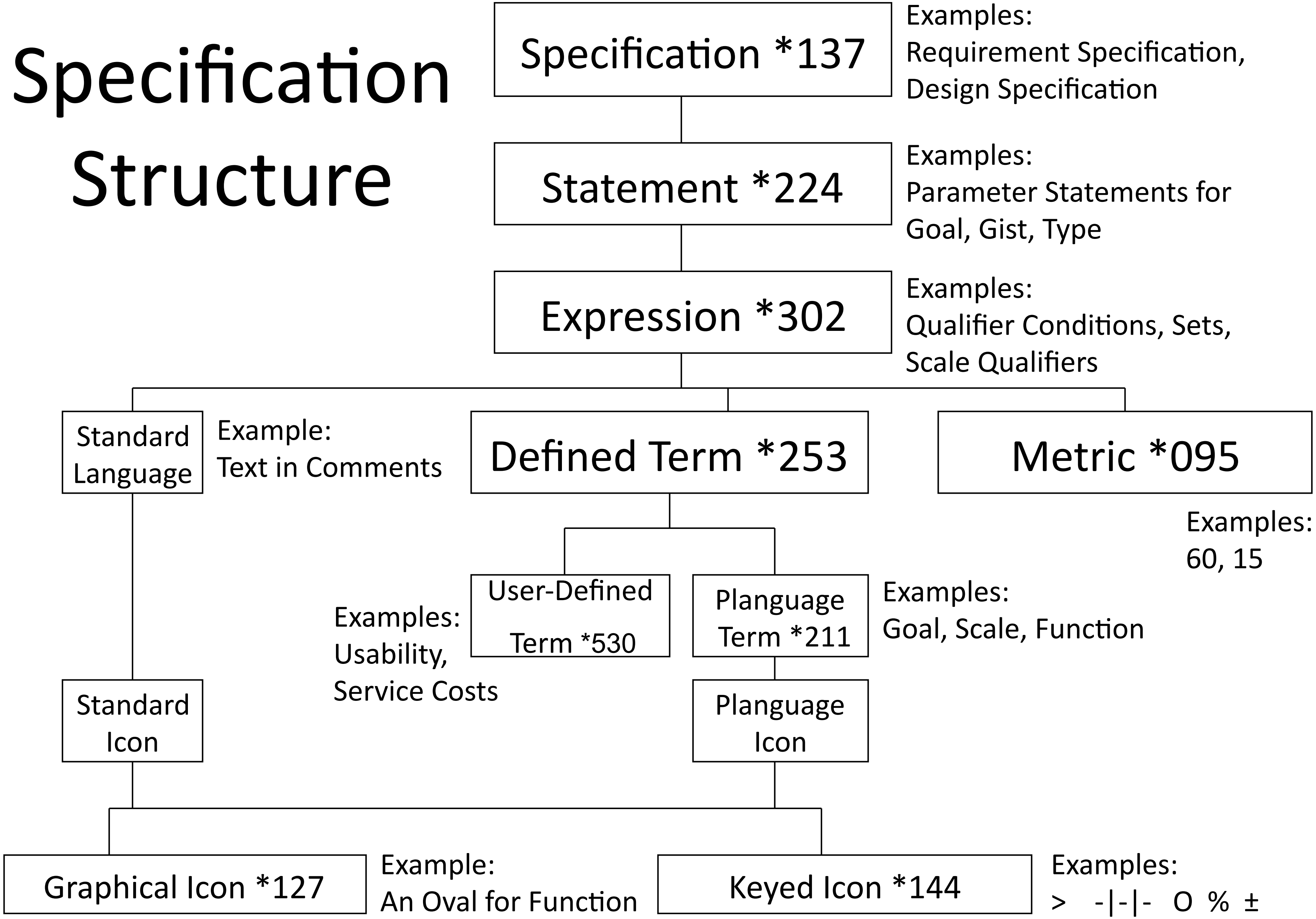
Requirement
Specification
[Updated]

Requirements Process

Analysis of dangerous common requirements practices.

1. Mixing real requirements with ‘designs’
2. Failing to quantify competitive qualities
3. Failing to derive implied requirements from designs
4. Failing to set requirements at appropriate level
5. Failing to include source information
6. Failing to include priority information
7. Failing to specify all critical stakeholders requirements
8. Failing to distinguish between wishes and their profit
9. Failing to consider all limited resources as requirements
10. Failure to update requirements evolutionarily
11. Failure to consider time, location and conditions with requirements
12. Failure to plan a time series of quality levels

Specification Structure



Specification *137

Examples:
Requirement Specification,
Design Specification

Statement *224

Examples:
Parameter Statements for
Goal, Gist, Type

Expression *302

Examples:
Qualifier Conditions, Sets,
Scale Qualifiers

Standard Language

Example:
Text in Comments

Defined Term *253

Metric *095

Examples:
60, 15

User-Defined Term *530

Examples:
Usability,
Service Costs

Planguage Term *211

Examples:
Goal, Scale, Function

Standard Icon

Planguage Icon

Graphical Icon *127

Example:
An Oval for Function

Keyed Icon *144

Examples:
> -|-|- O % ±

Configuration Management Headers (Owner, Version & others)

Tag:

Type: <fill in Attribute Description>.

Stakeholders: { }

Version:

Owner:

Here is an example of a client’s function definition **(edited for confidentiality):**

Emergency Stop:
- Type: Function.
- Description: <Requirement detail>.
- Module Name: GEX.F124.
- Users: {Machine Operator, Run Planner}.
- Assumptions: The User Handbook describes this in detail for all <User Types>.
- User Handbook: Section 1.3.5 [Version 1.0].
- Planned Implemented: Early Next Year, Before Release 1.0.
- Latest Implementation: Version 2.1. “Bug Correction: Bug XYZ.”
- Test: FT.Emergency Stop.
- Test [System]: {FS.Normal Start, FS.Emergency Stop}.
- Hardware Components: {Emergency Stop Button, Others}.
- Owner: Carla.

TEMPLATE FOR FUNCTION SPECIFICATION <with hints>

Tag: <Tag name for the function>.

Type: <{Function Specification,
Function (Target) Requirement,
Note: By default, a Function Requirement is assumed to be a Function Target.
Function Constraint}>.

============= Basic Information===================

Version: <Date or other version number>.

Status: <{Draft, SQC Exited, Approved, Rejected}>.

Quality Level: <Maximum remaining major defects/page, sample size, date>.

Owner: <Name the role/email/person responsible for changes and updates to this specification>.

Stakeholders: <Name any stakeholders with an interest in this specification>.

Gist: <Give a 5 to 20 word summary of the nature of this function>.

Description: <Give a detailed, unambiguous description of the function, or a tag reference to someplace where it is detailed. Remember to include definitions of any local terms>.

============Relationships=====================

Supra-functions: <List tag of function/mission, which this function is a part of. A hierarchy of tags, such as A.B.C, is even more illuminating. Note: an alternative way of expressing supra-function is to use Is Part Of>.

Sub-functions: <List the tags of any immediate sub-functions (that is, the next level down), of this function. Note: alternative ways of expressing sub-functions are Includes and Consists Of>.

Is Impacted By: <List the tags of any design ideas or Evo steps delivering, or capable of delivering, this function. The actual function is NOT modified by the design idea, but its presence in the system is, or can be, altered in some way. This is an Impact Estimation table relationship>.

Linked To: <List names or tags of any other system specifications, which this one is related to intimately, in addition to the above specified hierarchical function relations and IE-related links. Note: an alternative way is to express such a relationship is to use Supports or Is Supported By, as appropriate>.

============= Measurement=====================

Test: **<Refer to tags of any test plan or/and test cases, which deal with this function>.**

============= Priority and Risk Management=========

Rationale: **< Justify the existence of this function. Why is this function necessary? >.**

Value: **<Name [Stakeholder, time, place, event>]: <Quantify, or express in words, the value claimed as a result of delivering the requirement>.**

Assumptions: **<Specify, or refer to tags of any assumptions in connection with this function, which could cause problems if they were not true, or later became invalid>.**

Dependencies: **<Using text or tags, name anything, which is dependent on this function in any significant way, or which this function itself, is dependent on in any significant way>.**

Risks: **<List or refer to tags of anything, which could cause malfunction, delay, or negative impacts on plans, requirements and expected results>.**

Priority: **<Name, using tags, any system elements, which this function can clearly be done *after* or must clearly be done *before*. Give any relevant reasons>.**

Issues: **<State any known issues>.**

============= Specific Budgets =============

Financial Budget: **<Refer to the allocated money for planning and implementation (which includes test) of this function>.**

Source: Gilb: Competitive Engineering, Function Chapter

Set Trend Boundaries:
FUNCTION REQUIREMENT Real Example: RAF UK

Set Trend Boundaries: R016:

Owner: IRD

Type: Function Requirement

Gist: After all data used to run FIMI has been imported to the system, the minimum and maximum Boundaries for the Various trending parameters , for each individual aircraft, will be computed and stored.

FIMI: defined as : Fatigue Index Monitoring IIxxx ? <- Matt T

Boundaries: defined as: lowest and highest figure for each individual field that will be stored in the data catalogue ?? <- Rose J

Various: defined as: the set of all measurement data entry trending parameters for that table? <- RJ

Definition [Set Trend Boundaries: R016] :

Import To FIMI: Type: Process: import to Islander System Army and RAF Flight Data from the Islander Form 725 [All known present and past Version].

Correct The Data: Type: Process: edit the data within the Violation Table, which has Failed the Validation Rules

Depends on : ability to add and amend:

Violation Table: defined as: in Islander Database, called 'Violation Table'

Flight Table: defined as: a table in Islander system which has validated (according to Violation Table) flight data.

Query Aircraft Data: Type: Process: ....

Input to Trend Table: Type: Process: ....

Produce Stats: Type: Process: ....

Detailed Design Spec: <not currently available Oct 2001>

Test Plan: 'Islander Test Specifications': Owner: S Cxxx

Test Cases: <Cxxx would be able to point these out>

Stakeholders: {Ixxx Aircraft Maintenance (ASI4) : Squadron Leader Ian Txxx (successor), Ixxxr IPT (Note: uninterested), IRD (Fatigue Report Documentation?, Britten Nxxx Company (Isle of Wxxx), Axxx(produce a separated format disk holding the information), Unit Staff (for future records), Our Testers, Project Manager, Our Development Staff }

Sponsor: Wing Commander <?>

Impacts:Trend Breach Report

Impacted By:Flight Data Quality

Depends On: Fatigue Formula is not available. We don't own it, Bittan N do it. <-RJ

Risks: {Flight Data Quality, Corrupt Disk Data, IRD Operator Error, ...}

Corrupt Disk Data: Mitigation: Manual input from paper forms. OR ask A to redo the disk.

IRD Operator Error [No Violation Table detection] : Mitigation: WE NEED SOME SYSTEM OF QC WE HAVE NOT DEVELOPED YET! <-TG OCT 5 2001

NOTE: THERE IS AN OPPORTUNITY TO INVEST IN BETTER DATA VALIDATION THAN IS CURRENTLY PLANNED USING THE VIOLATION TABLE.

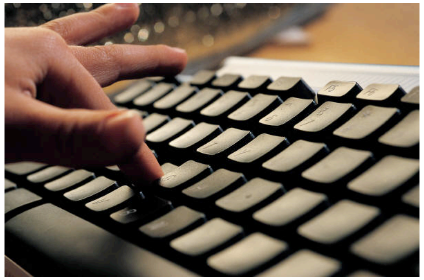
Spec Stored On: <our intranet??>

Version: October 5 2001 10:12

Design Possibility: pay Britten N Company (Isle of Wight) to run this or give us the software. <- Jamie B HQ Land Command [Oct 5 2001]

Risk: actual might be ancient and costly? <- RJ. We were not able to run it once we tried, in FORTRAN <-RJ

Simple Function Specification Example



Manual Dialing:

Type: Function Requirement.

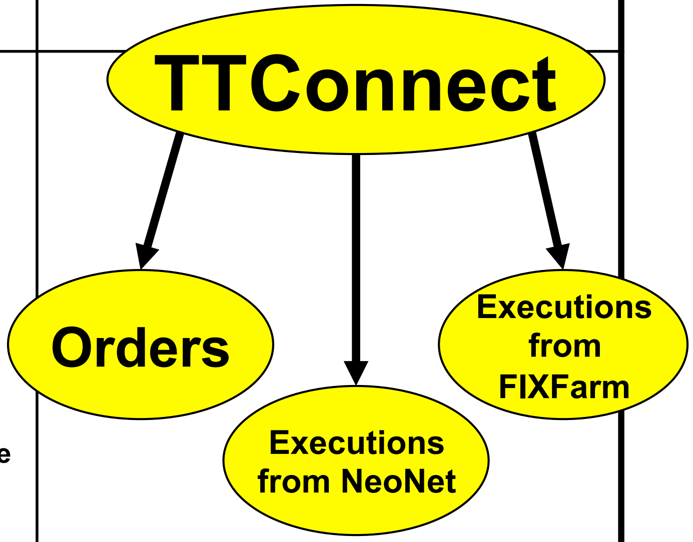
Description:

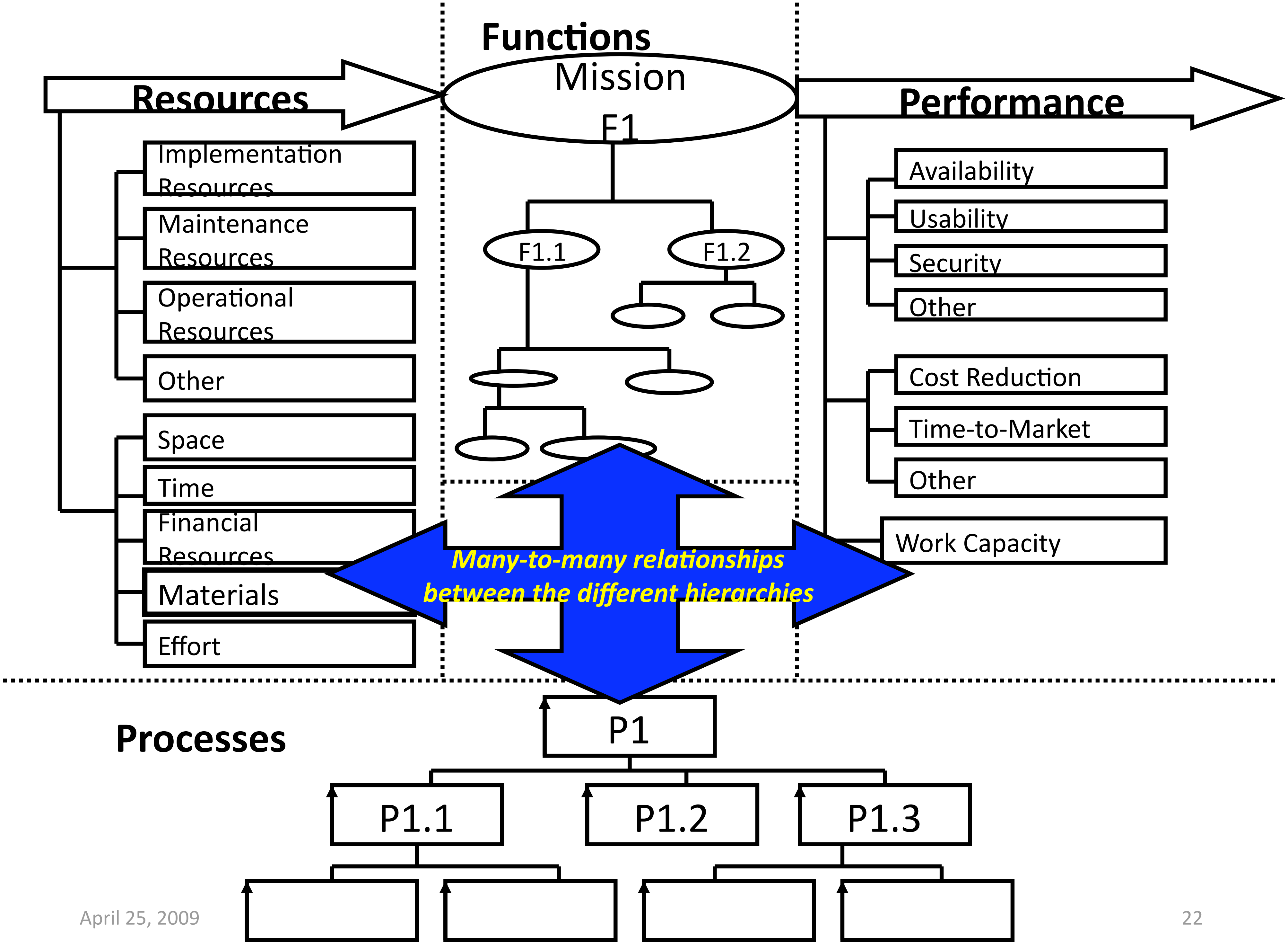
The user capability, *by any available means {finger on key, voice, name list}*, to select or provide and, transmit a {telephone or internet} {number or address} and any other required symbols, to reach and access any available services.

Scope: It specifically includes any keying or other activity needed in connection with communication, such as accessing lists.

Scope: It specifically excludes any non-communication activity such as game playing.

Function Requirement: Real Example, 2004 UK

Tag	TTConnect	Functional Requirement
Owner	**AE**	**Version 1.0**
Writer	**Laura XXX**	**Status: Draft**
Stakeholders	**AE, EGAS, Transact Tools**	
Description	**Route messages as per Agreed Structure**	
Detail	**FIX: Maintain a FIX Session with each Connection linked to Child FIX Sessions with each of NeoNet and FIXFarm (so for each customer-side FIX Session there are two linked sessions)** **Route: FIX Application Messages across linked session as described in sub-requirements.** **MQ: Maintain a single MQ connection to route FIX Execution Messages and FIX Reject Messages as described in sub-requirements.** **Connection**: defined as: **a single- or multi-customer FIX counterparty** **Child FIX Sessions**: defined as: **FIX Sessions that are only active when the main FIX Session is active.** **Note: FIX Application Messages are routed across linked session as described in sub-requirements.**	 TTConnect Orders Executions from NeoNet Executions from FIXFarm
Test:		
Assumptions	**A1: A connection for High Speed DMA cannot also support orders that are not for High Speed DMA processing.** **A2: There will be a FIX Session to NeoNet per customer. not a single shared connection.**	
Issues	**I1: Need to confirm FIX versions for connection to NeoNet**	
Includes	**Executions from FIXFarm, Executions from NeoNet, Orders**	



Functions

Mission F1

F1.1

F1.2

Resources

Implementation Resources

Maintenance Resources

Operational Resources

Other

Space

Time

Financial Resources

Materials

Effort

Performance

Availability

Usability

Security

Other

Cost Reduction

Time-to-Market

Other

Work Capacity

Many-to-many relationships between the different hierarchies

Processes

P1

P1.1

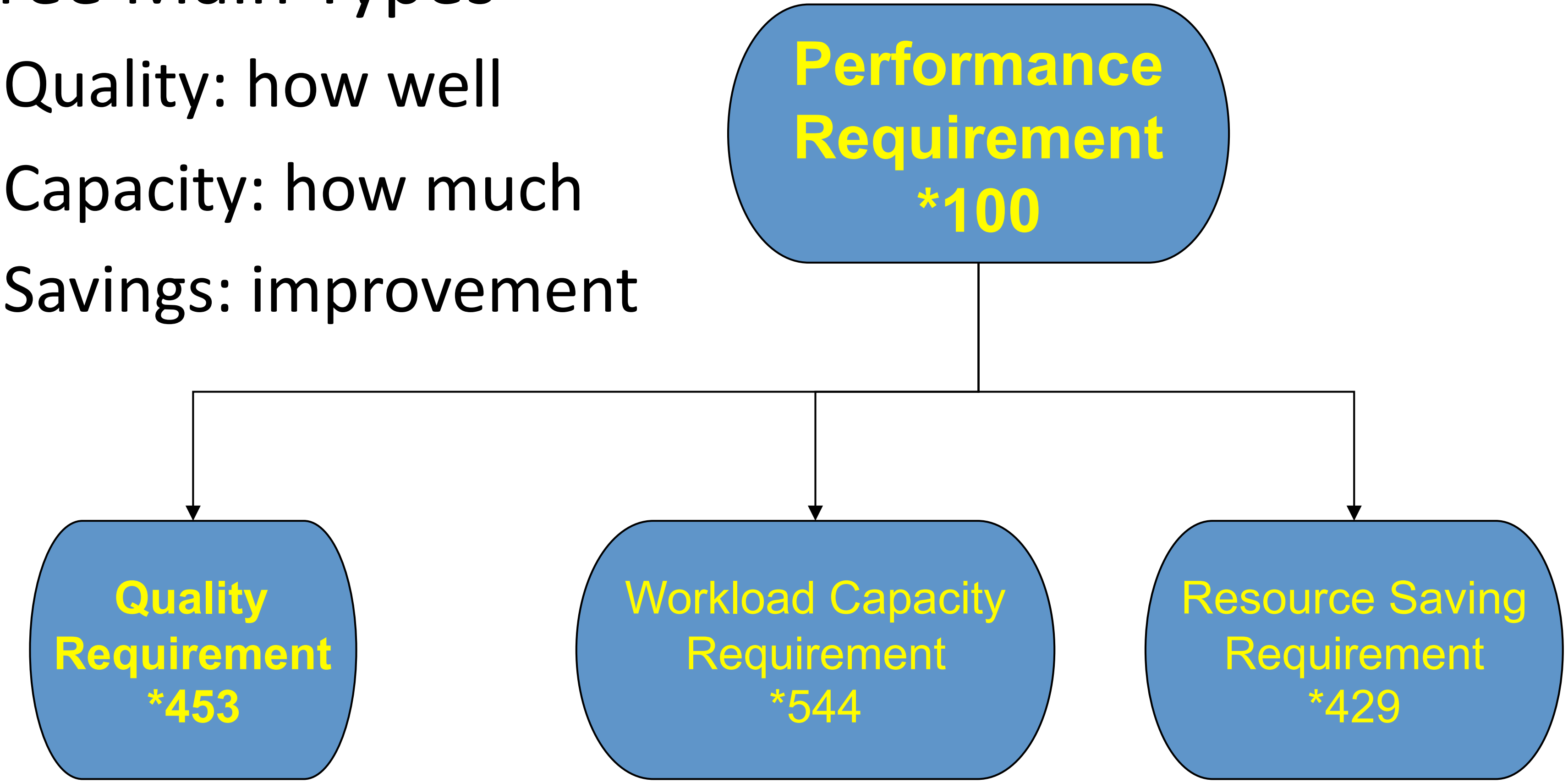
P1.2

P1.3

Many to many relationships

Performance Requirements Types

- ◆Three Main Types
  - ◆Quality: how well
  - ◆Capacity: how much
  - ◆Savings: improvement



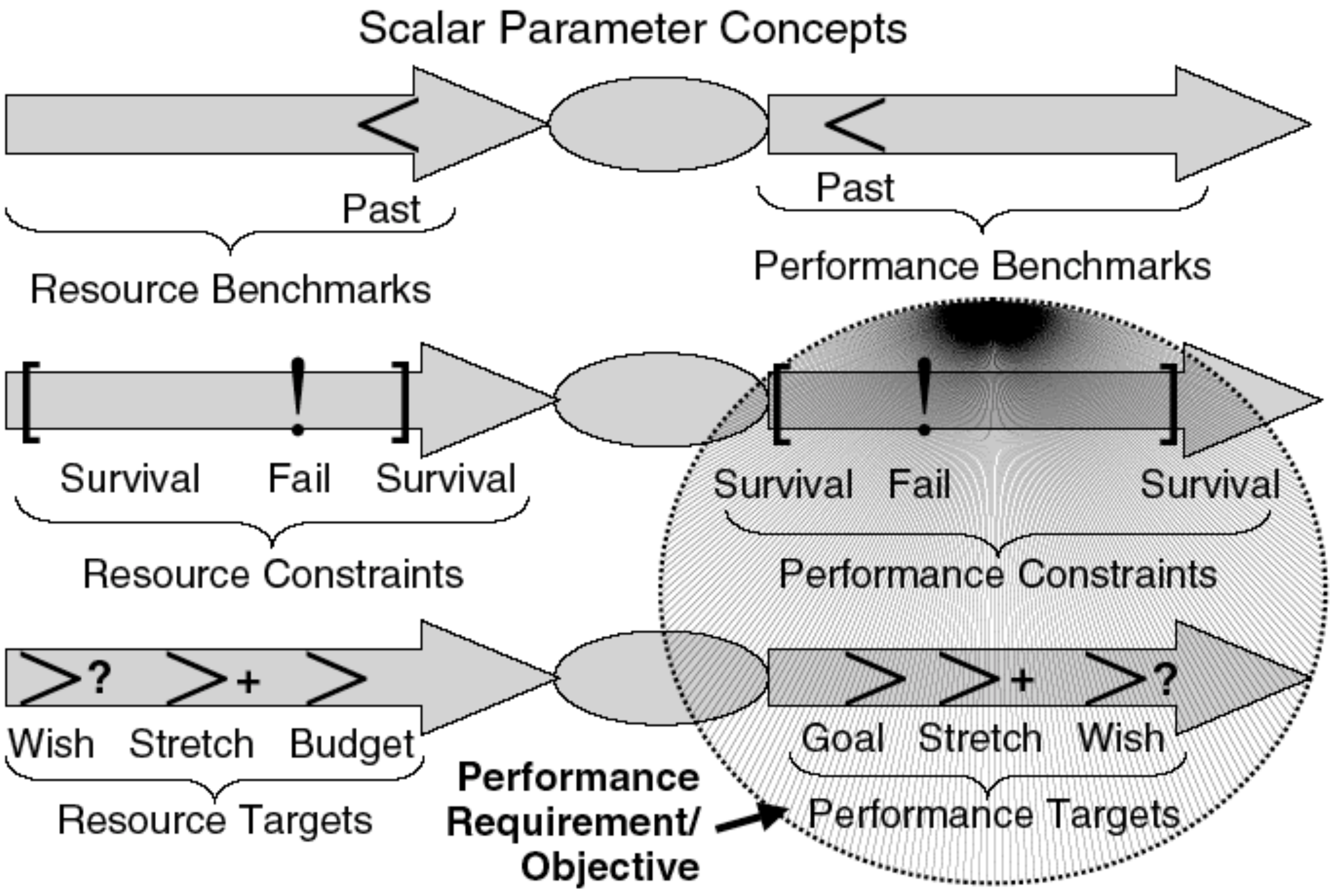
Performance Requirement *100

Quality Requirement *453

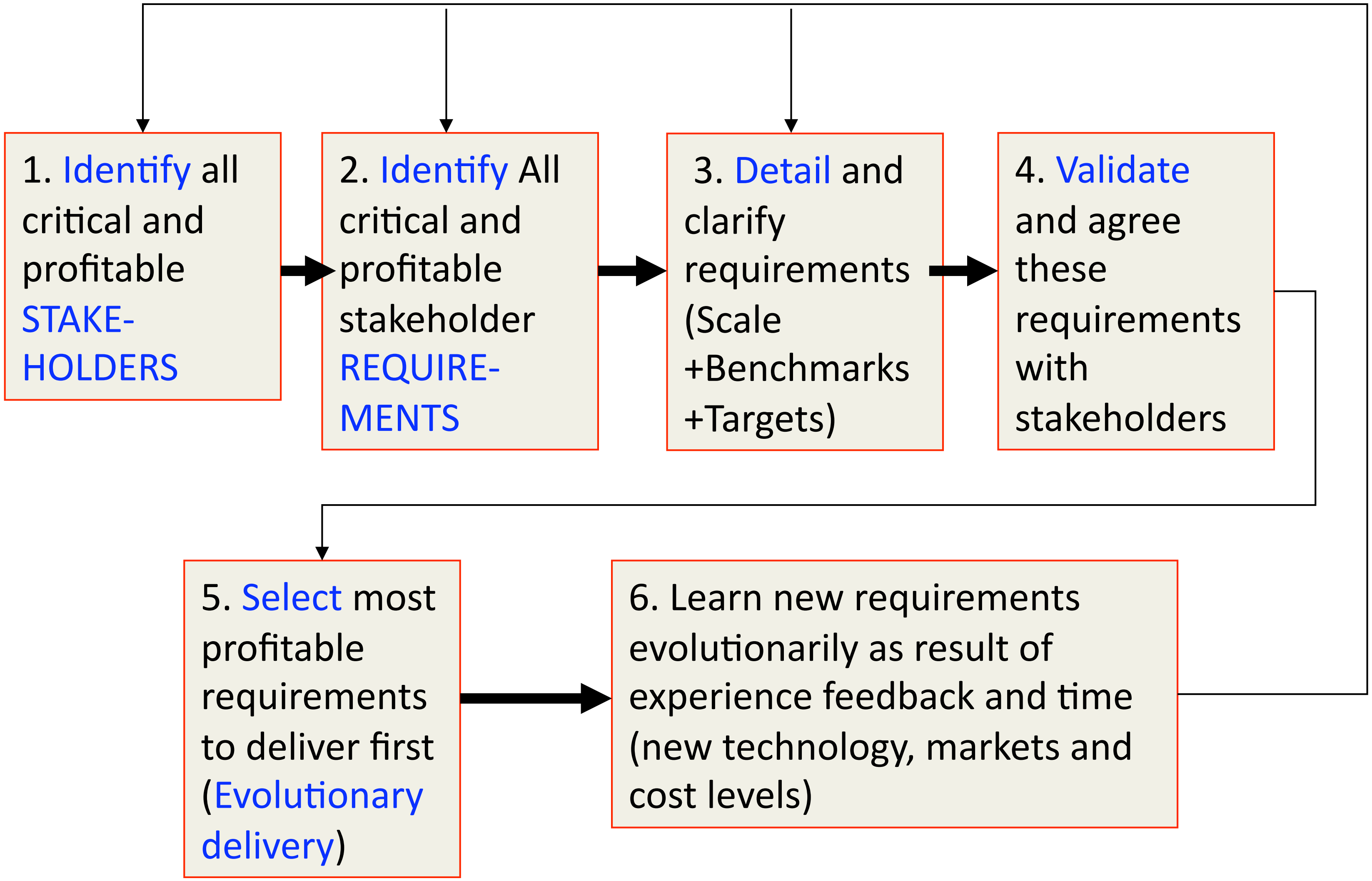
Workload Capacity Requirement *544

Resource Saving Requirement *429

Performance Requirements



Stakeholders:
How to find out about, and confirm, their requirements



1. Identify all critical and profitable STAKE-HOLDERS

2. Identify All critical and profitable stakeholder REQUIREMENTS

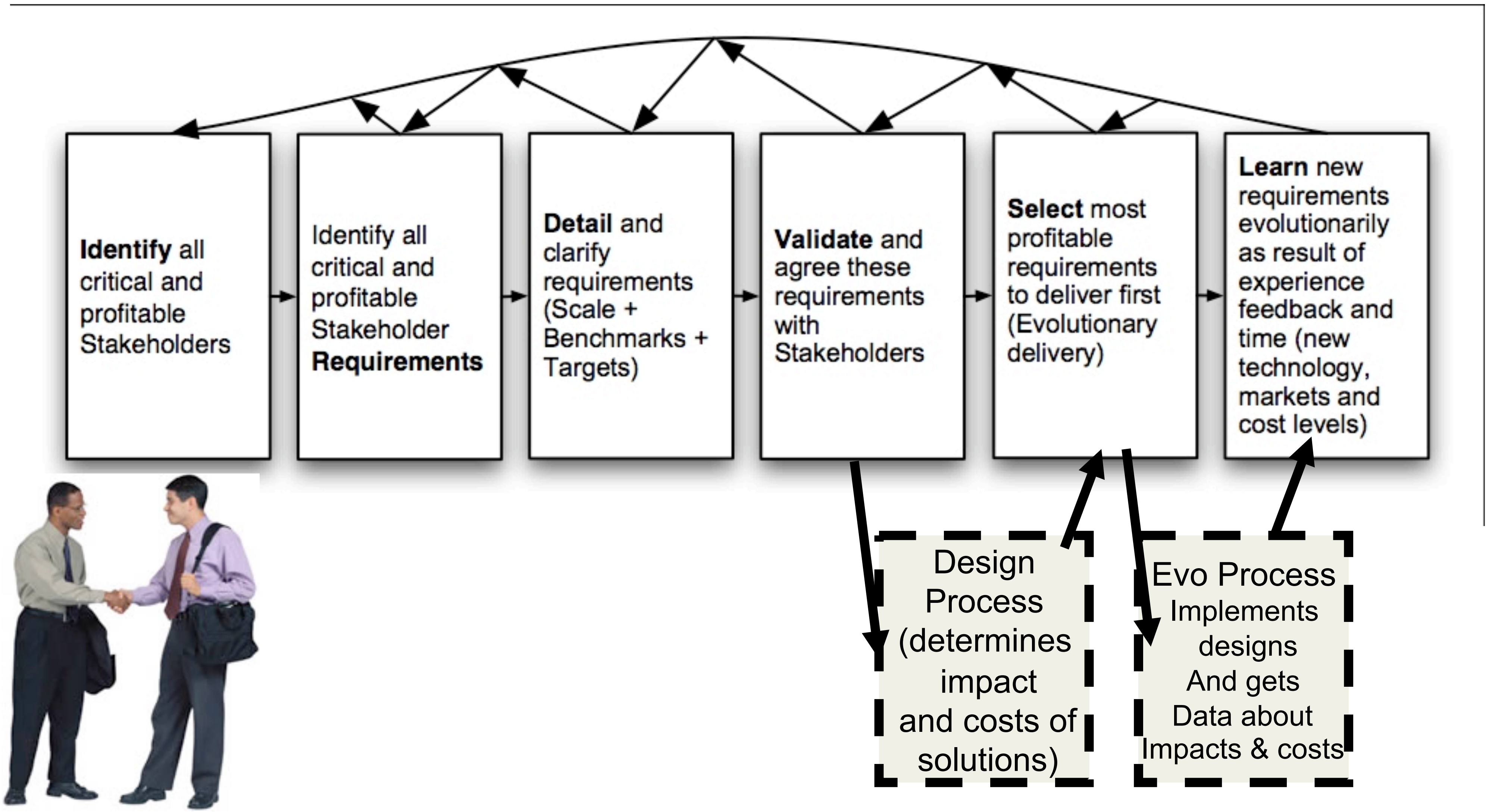
3. Detail and clarify requirements (Scale +Benchmarks +Targets)

4. Validate and agree these requirements with stakeholders

5. Select most profitable requirements to deliver first (Evolutionary delivery)

6. Learn new requirements evolutionarily as result of experience feedback and time (new technology, markets and cost levels)

Stakeholders: How to find out and confirm their requirements



Identify all critical and profitable Stakeholders

Identify all critical and profitable Stakeholder **Requirements**

Detail and clarify requirements (Scale + Benchmarks + Targets)

Validate and agree these requirements with Stakeholders

Select most profitable requirements to deliver first (Evolutionary delivery)

Learn new requirements evolutionarily as result of experience feedback and time (new technology, markets and cost levels)

Design Process (determines impact and costs of solutions)

Evo Process Implements designs And gets Data about Impacts & costs

Workshop Exercise: Stakeholders
300 seconds (15 sec each)

- Brainstorm 10 External (Use, Buy, Review, Approve) Stakeholders

- Brainstorm 10 Internal (develop, maintain) stakeholders

(Quality) Requirements Specification

Template with <hints>

<name tag of the objective>

Ambition: <give overall real ambition level in 5-20 words>

Version: <dd-mm-yy each requirements spec has a version, at least a date>

Owner: <the person or instance allowed to make official changes to this requirement>

Type: <quality|objective|constraint>

Stakeholders: { , , } "who can influence your profit, success or failure?"

Scale: <a defined unit of measure, with [scale parameters] if you like>

Meter [<for what test level?, like 'system test'>] <specify test method for this scale>.

====Benchmarks ============= the Past

Past [ ] <estimate of past> <--<source>

Record [<where>, <when >, <estimate of record level>] <-- <source of record data>

Trend [<future date>, <where?>] <prediction of level> <-- <source of prediction>

===== Targets ============= the future needs

Wish [<where>, <when>] <Stakeholder desire> <-- <source of wish>

Goal [<where>, <when>] <target level> <-- Source of Goal

Value [Goal] <refer to or state value created if we reach Goal>.

Stretch [<where>, <when>] <motivating ambition level> <-- <source of level>

======== Constraints ==================

Fail [ ] <a level where some degree and type of failure begins> <-- <source>

Survival [ ] <a level where the system is worthless, total failure> <---source

Scalar (‘How well?’, ‘How much?’) Requirement Template; Fundamental Example.

Requirement Tag:

Stakeholders:
Scale:
Meter [ ]

=== Benchmarks =============== the Past ==

Past [ ] <-
Record [ ] <-

=== Targets ====== the future value and needs ==

Goal [ ] <-
Wish [ ] <-
Stretch [ ] <-

Requirements Templates: fill it out!

Requirement Tag:
Version:
Owner:
Ambition:
Stakeholders:
Scale:
Meter [ ]

=== Benchmarks =============== the Past ==

Past [ ] <-
Record [ ] <-
Trend [ ] <-

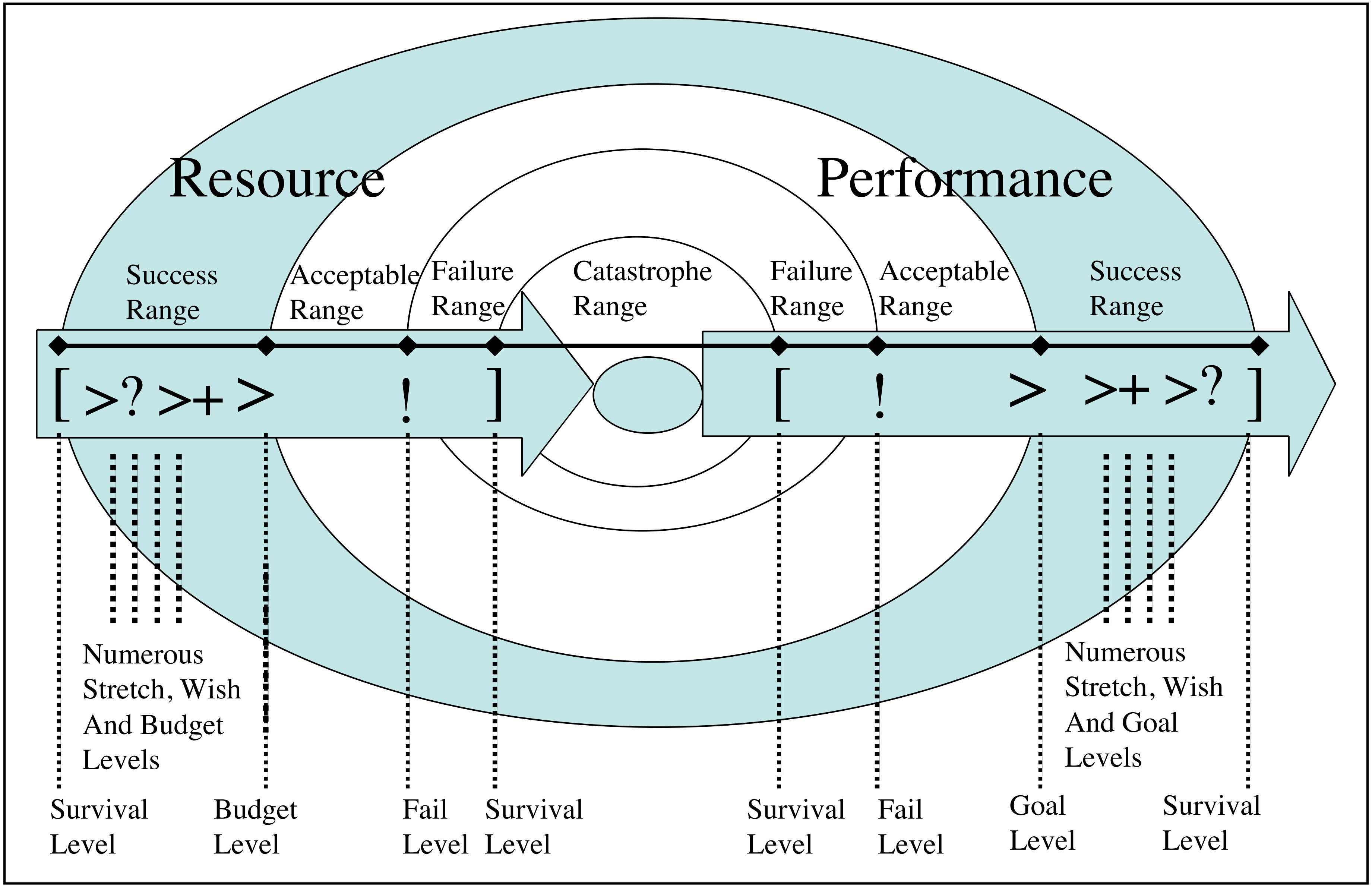
=== Targets ====== the future needs ==

Goal [ ] <-
Stretch [ ] <-
Wish [ ] <-

==== Constraints =====================

Fail [ ] <-
Survival [ ] <-

Doughnut Diagram: Ranges, Limits, and Levels



A detailed real example of Quality Specification (Oct 2004, Europe)

Design Effort [Roadbed.Drainage System, Product XX]: ‘Approved by Team’ 13:59 Tuesday (Day 1)

***Ambition** Level: 10X “at least 10 times less engineering effort than now”*

Administration:

Approved: by Team’ 13:59 Tuesday (Day 1) ok to progress to strategy phase.

Type: Product Quality [Product XX].

Version: 12 Oct 2004 10:12, 11:38

Owner: Idar

Stakeholders: Senior Road Designers, Road Designers, Drainage System Designers, Contractors.

Scale: Hours of Engineering Effort per 10 km road to Complete Roadbed Description for a defined Ideal Engineering Level: default 100%.

Assumption: the level of qualities is the same for comparative measurements. E g we do not save time, only to turn around and use it to increase quality. We still saved time for the old quality level. <-TG

--- Benchmarks------- Analysis---------

Past [VXX Our Infrastructure Design, Finland] ??? <-IK

Bad: **Past** [IEL = 20%? “<wrong mass calculations & drawings, absence of stakeout data>”, 2004, Project = <general guess IK>] <30> ±10 hours/10km <- SWAG IK

○

Good: **Past** [IEL = 90%? “<better mass calculations & drawings, some stakeout data>”, 2004, Project = <general guess IK>, Excel & our product, Swedish Users] <80> ±20 hours/10km <- SWAG IK

○

Trend [Our Product XX] <customers are more demanding, 20% is no longer a good enough level>. <- Heidi

○

Record [InXXX, 2004] <better than us in design, high Engineering Level, more consistent but not redesign> <-IK “our system is clearly worse here than the competitor – so we must improve”

--- Constraints-------- Requirement-----

Tolerate [End 2005, Road Designers, ….] Past – 25% of hours

Rationale: least powerful sales argument for selling new version.

Survival [Anytime,] <today’s level or better>

Rationale: we could lose customers to competitors.

--- Targets ----------- Requirement ----------

G1. **Goal** [IEL = 90%?, 2005 Q4, Norwegian & English] <8> ±2 hours/10 km <- Heidi, Berit, Inge

G2. **Goal** [IEL = 90%?, 2006 Q1, Swedish] <8> ±2 hours/10 km <- Heidi, Berit, Inge

--- Evolutionary Goals----------------

○

Short Term 1: **Goal** [End November 2004, Stakeholder = {SVV, Road Designers}] Past -20%?

Short Term 2: **Goal** [End 2004] Past -40%?

○ **Goal** [End January 2005] Past -50%?

○ **Goal** [End Feb 2005] Past -60%?

○ **Goal** [End Mar 2005] Past -70%?

○ **Goal** [End Dec 2005] Past – 90% =Long Term

○

Note: we lack clarity in Stakeholder to be served at each step. This decides some things to

-

Hours of Engineering Effort: net, actually applied to the task hours.

Complete: {all considerations taken, engineering quality controlled, contractor approved, to a defined % level of IEL}

Roadbed Description: defined as: {cross-section drawings, mass calculation, Geometrical Description: {existing terrain, related water and sewer, other roads, tunnel},

Requirements Templates;
Other useful Parameters

see CE index and Glossary for detail

Version:

Status:

Maturity: <see note in slide for parameters>

Owner:

Sponsor:

Author:

Sources:

Due:

Derived From:

Impacts:

Impacted By:

Sub-component of:

Sub-components: { }.

Risks:

Dependencies:

Diagram:

Picture:

Table:

URL:

Contract:

Correspondence:

Authority:

Constraints:

Supplier:

Acceptance Criteria:

Test Plan:

Test Cases:

Note:

Reference:

Confidentiality:

Scenario:

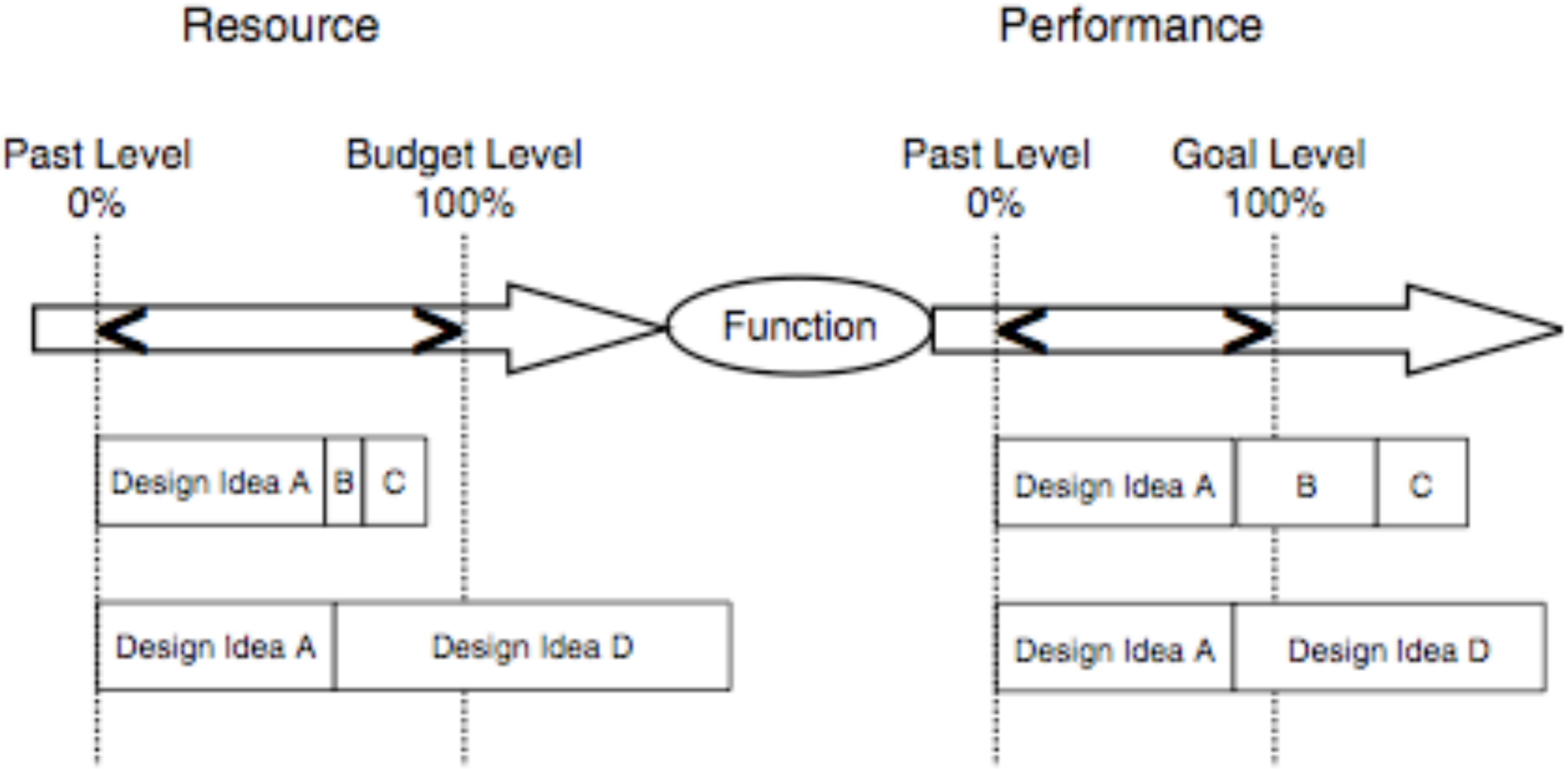
Use Case:

Model:

Subscribers List

Break 10 minutes to 15:00

15 – 15:50 Design

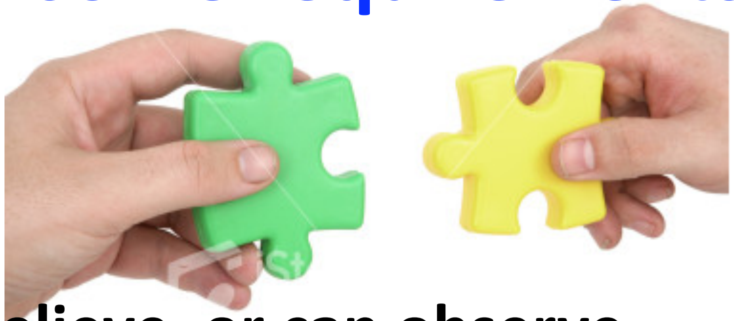


Design idea (noun):

Concept *047 2003

A design idea is

anything that will satisfy some requirements.

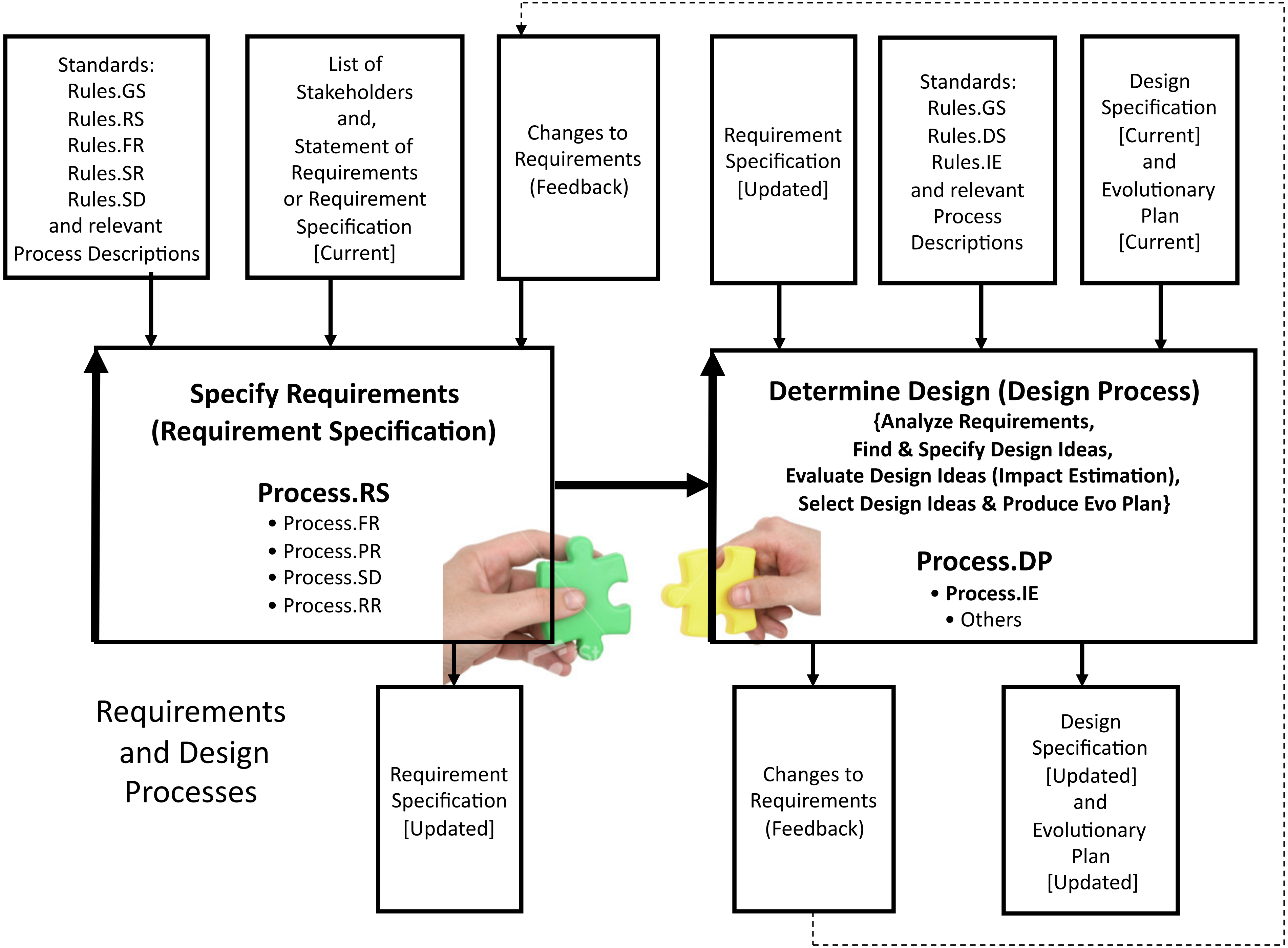


A design idea is a means to an end.

A design idea is a anything that we believe, or can observe, contributes to the observable function, performance, and cost attributes of a system.

A design idea (or 'a design') may be – 'in our minds', 'spoken aloud', 'found to exist in real systems', and it may be formally or informally *specified* (as a design specification, *586).

Requirements and Design Processes



Standards:
Rules.GS
Rules.RS
Rules.FR
Rules.SR
Rules.SD
and relevant
Process Descriptions

List of
Stakeholders
and,
Statement of
Requirements
or Requirement
Specification
[Current]

Changes to
Requirements
(Feedback)

Requirement
Specification
[Updated]

Standards:
Rules.GS
Rules.DS
Rules.IE
and relevant
Process
Descriptions

Design
Specification
[Current]
and
Evolutionary
Plan
[Current]

Specify Requirements
(Requirement Specification)

Process.RS
- Process.FR
- Process.PR
- Process.SD
- Process.RR

Determine Design (Design Process)
{Analyze Requirements,
Find & Specify Design Ideas,
Evaluate Design Ideas (Impact Estimation),
Select Design Ideas & Produce Evo Plan}

Process.DP
- **Process.IE**
- Others

Requirement
Specification
[Updated]

Changes to
Requirements
(Feedback)

Design
Specification
[Updated]
and
Evolutionary
Plan
[Updated]

What is a 'design'? (architecture, solution)

Design Idea Concept *047 March 15, 2003

- **A design idea is anything that will satisfy some requirements.**
- **A *set* of design ideas is usually needed to solve a 'design problem'.**
- A design is a specific idea about how to solve a defined design problem.
- A design (or design idea: synonym) may be in our minds, spoken aloud, found to exist in existing systems, and it may be formally or informally specified (design specification).
- A Design is a 'consciously selected means' to reach defined 'ends'.
- A design idea must be consistent with a *set* of requirements, all at once.
- It must positively serve the improvement of at least one item towards specified requirements.
- But it must also not violate any other constraint (function, condition constraint, scalar constraint) which it can impact.
- A design is different from a requirement in that it can in principle be changed at any time for a better design, which better meets the requirements.
- Design is not holy and fixed.

SCALAR REQUIREMENT SPECIFICATION

Participation: Scale: % of worldwide membership participating. Goal: 10%.

Representation: Scale: % of worldwide membership represented within defined <groups>.

Goal [Age under 25 or equating to <student status>]: 10%.

Information: Scale: % of talks rated as 'good' or better (5+ on feedback sheet scale). Goal: 50%.

Conviction: Scale: % participants wanting to return next conference. Goal: 80%.

Influence: Scale: % participants who <improve as result of the conference>.

Past: 90%, Goal: 95%.

Fun: Scale: % participants rating the conference-city quality as 'good' or better (5+ on feedback sheet scale).

Past: 45%. Plan: 60%.

Cost: Resource Budget: Scale: total cost for an individual participant including travel costs.

Fail: $2,000. Goal: $1,200 or less.

DESIGN SPECIFICATION (simple version)

Central: Choose a location in the membership center of gravity (New York?)

Youth: Suggest and support local campaigns to finance 'sending' a young representative to conference.

Facts: Review all submitted papers on <content>.

London: Announce that the conference is to be in London next time.

Diploma: Give diplomas for attendance, and additional diplomas for individual tutorial courses.

Events: Have entertainment activities organized every evening: river tours, etc.

Discounts: Get discounts on airfare and hotels.

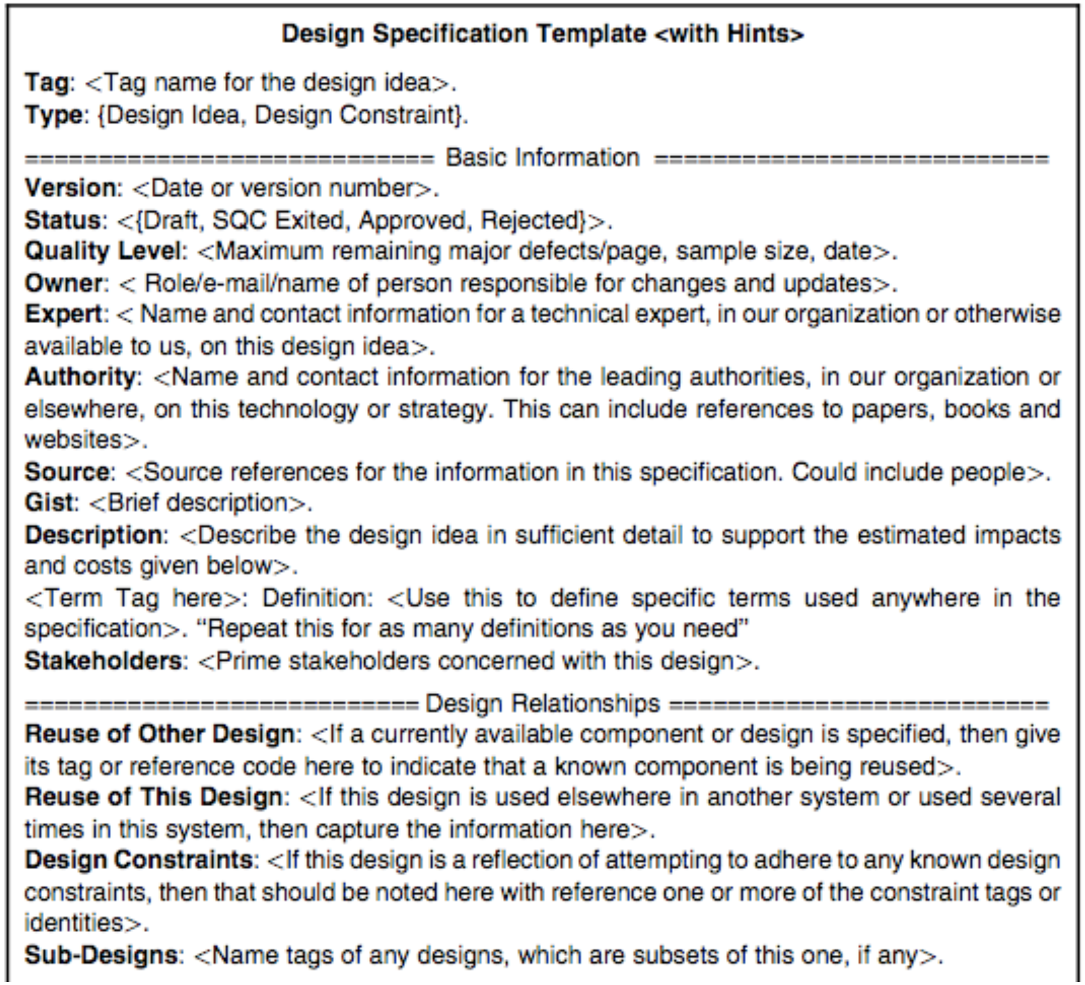
Simple Design Spec Exercise

- <Name It>:
- Type: Design/Architecture/Strategy/Solution
- Supports: { name one main requirement this is supporting }.
- Impacts: name all requirements this has non-trivial impacts on.
- Costs: Name cost areas impacted
- Short Description:
- Detailed Description:
- Risks: list any uncertainties with this design
- Designer: <who takes technical responsibility>
- Design URL: list any useful websites on this design.

Simple Design Spec Exercise: fill out form

- ______________:
- Type:
- Supports:
- Impacts:
- Costs:
- Short Description:
- Detailed Description:
- Risks:
- Designer:
- Design URLs:

Design Template Part 1 of 3



Design Specification Template <with Hints>

Tag: <Tag name for the design idea>.
Type: {Design Idea, Design Constraint}.

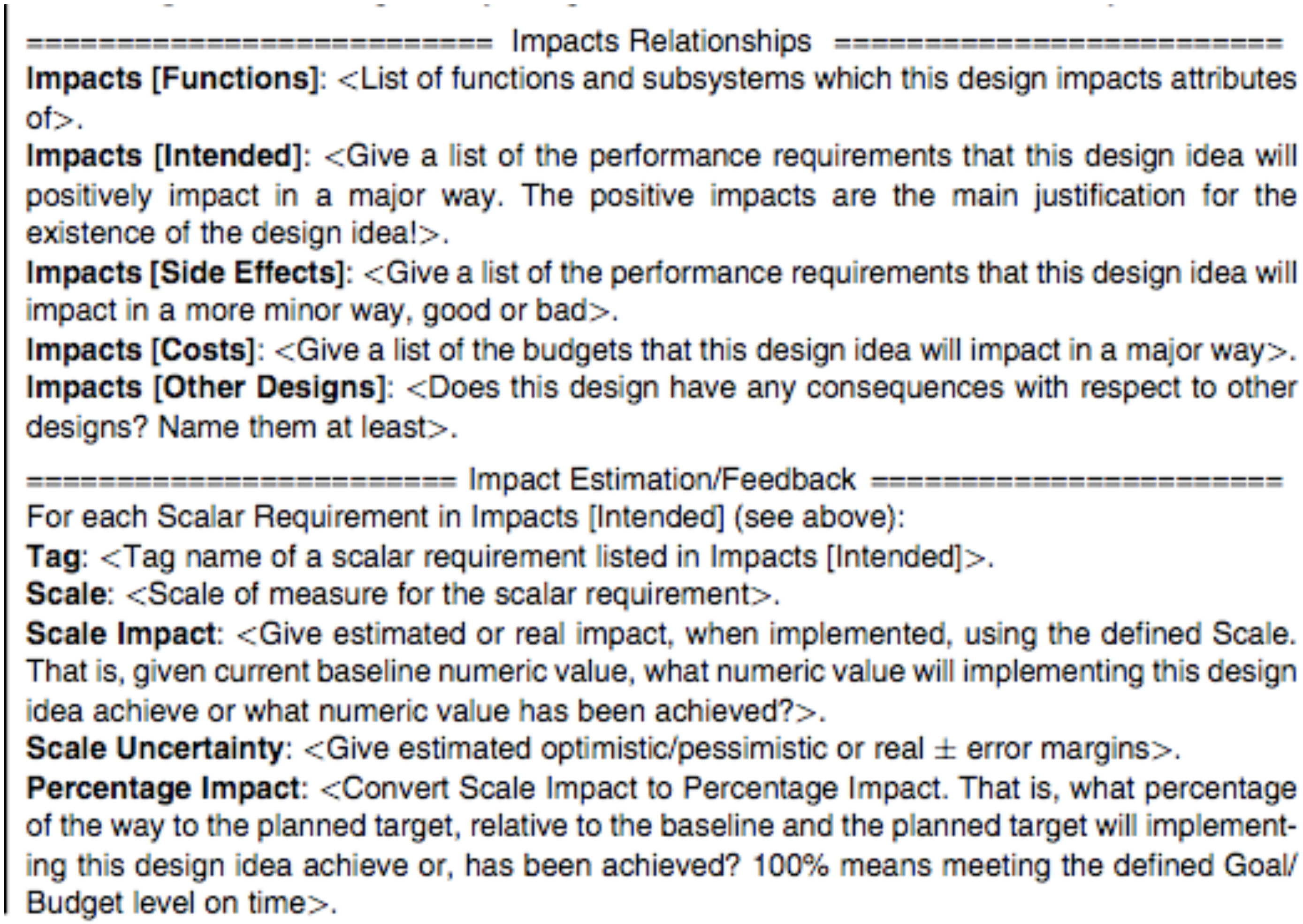
============================ Basic Information ===========================

Version: <Date or version number>.
Status: <{Draft, SQC Exited, Approved, Rejected}>.
Quality Level: <Maximum remaining major defects/page, sample size, date>.
Owner: < Role/e-mail/name of person responsible for changes and updates>.
Expert: < Name and contact information for a technical expert, in our organization or otherwise available to us, on this design idea>.
Authority: <Name and contact information for the leading authorities, in our organization or elsewhere, on this technology or strategy. This can include references to papers, books and websites>.
Source: <Source references for the information in this specification. Could include people>.
Gist: <Brief description>.
Description: <Describe the design idea in sufficient detail to support the estimated impacts and costs given below>.
<Term Tag here>: Definition: <Use this to define specific terms used anywhere in the specification>. "Repeat this for as many definitions as you need"
Stakeholders: <Prime stakeholders concerned with this design>.

========================== Design Relationships ==========================

Reuse of Other Design: <If a currently available component or design is specified, then give its tag or reference code here to indicate that a known component is being reused>.
Reuse of This Design: <If this design is used elsewhere in another system or used several times in this system, then capture the information here>.
Design Constraints: <If this design is a reflection of attempting to adhere to any known design constraints, then that should be noted here with reference one or more of the constraint tags or identities>.
Sub-Designs: <Name tags of any designs, which are subsets of this one, if any>.

Design Template Part 2 of 3



========================== Impacts Relationships ==========================

Impacts [Functions]: <List of functions and subsystems which this design impacts attributes of>.

Impacts [Intended]: <Give a list of the performance requirements that this design idea will positively impact in a major way. The positive impacts are the main justification for the existence of the design idea!>.

Impacts [Side Effects]: <Give a list of the performance requirements that this design idea will impact in a more minor way, good or bad>.

Impacts [Costs]: <Give a list of the budgets that this design idea will impact in a major way>.

Impacts [Other Designs]: <Does this design have any consequences with respect to other designs? Name them at least>.

======================== Impact Estimation/Feedback ========================

For each Scalar Requirement in Impacts [Intended] (see above):

Tag: <Tag name of a scalar requirement listed in Impacts [Intended]>.

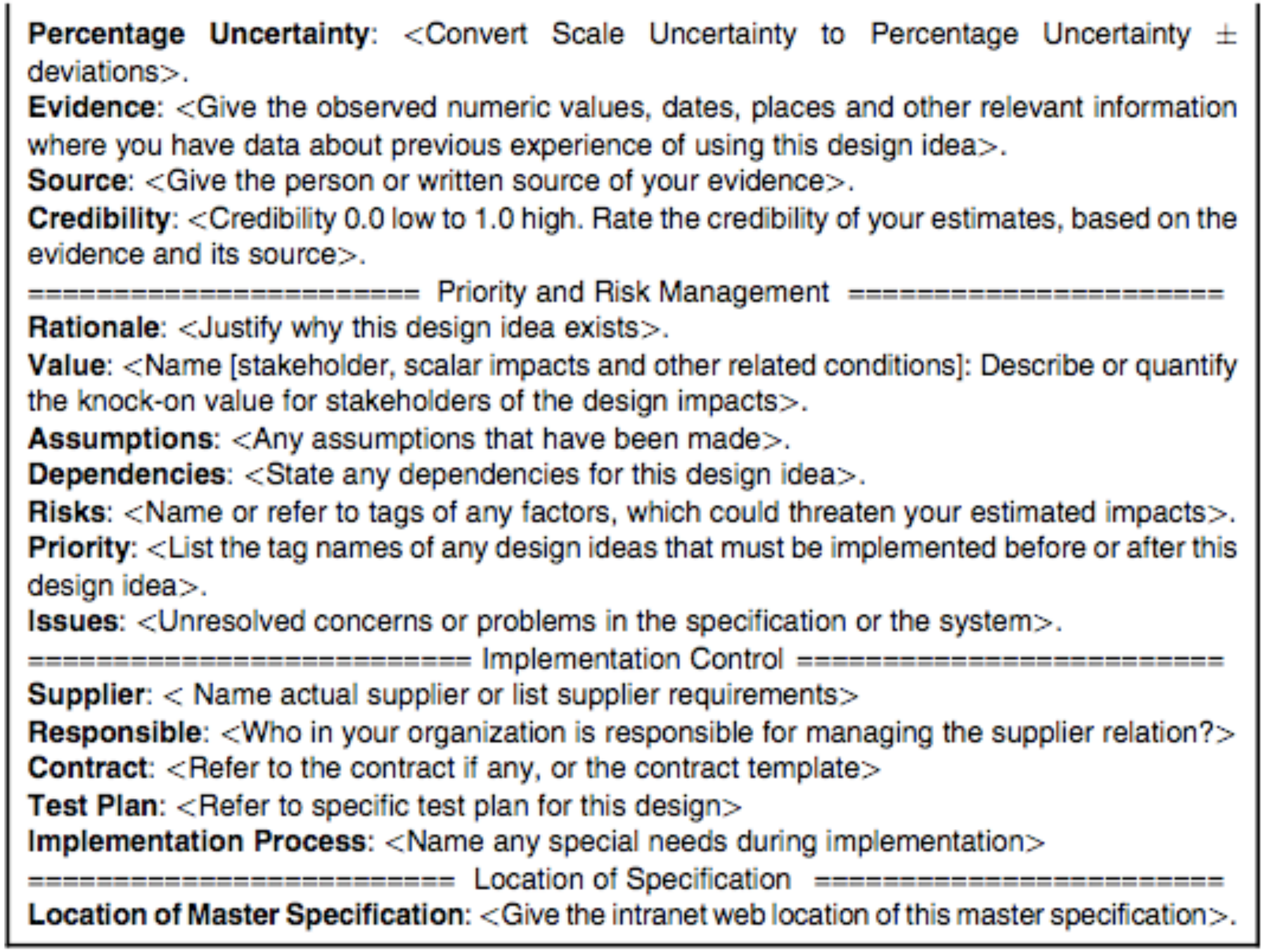
Scale: <Scale of measure for the scalar requirement>.

Scale Impact: <Give estimated or real impact, when implemented, using the defined Scale. That is, given current baseline numeric value, what numeric value will implementing this design idea achieve or what numeric value has been achieved?>.

Scale Uncertainty: <Give estimated optimistic/pessimistic or real ± error margins>.

Percentage Impact: <Convert Scale Impact to Percentage Impact. That is, what percentage of the way to the planned target, relative to the baseline and the planned target will implementing this design idea achieve or, has been achieved? 100% means meeting the defined Goal/ Budget level on time>.

Design Template part 3 of 3



Percentage Uncertainty: <Convert Scale Uncertainty to Percentage Uncertainty ± deviations>.

Evidence: <Give the observed numeric values, dates, places and other relevant information where you have data about previous experience of using this design idea>.

Source: <Give the person or written source of your evidence>.

Credibility: <Credibility 0.0 low to 1.0 high. Rate the credibility of your estimates, based on the evidence and its source>.

======================= Priority and Risk Management =======================

Rationale: <Justify why this design idea exists>.

Value: <Name [stakeholder, scalar impacts and other related conditions]: Describe or quantify the knock-on value for stakeholders of the design impacts>.

Assumptions: <Any assumptions that have been made>.

Dependencies: <State any dependencies for this design idea>.

Risks: <Name or refer to tags of any factors, which could threaten your estimated impacts>.

Priority: <List the tag names of any design ideas that must be implemented before or after this design idea>.

Issues: <Unresolved concerns or problems in the specification or the system>.

========================== Implementation Control ==========================

Supplier: < Name actual supplier or list supplier requirements>

Responsible: <Who in your organization is responsible for managing the supplier relation?>

Contract: <Refer to the contract if any, or the contract template>

Test Plan: <Refer to specific test plan for this design>

Implementation Process: <Name any special needs during implementation>

======================== Location of Specification ========================

Location of Master Specification: <Give the intranet web location of this master specification>.

Example of a (Real, partial) Design Specification using Planguage

Tag: OPP Integration.
Type: Design Idea [Architectural].
============ Basic Information ========================
Version:
Status:
Quality Level:
Owner:
Expert:
Authority:
Source: System Specification Volume 1 Version 1.1, SIG, February 4. - Precise reference <to be supplied by Andy>.

Gist: The X-999 would integrate both 'Push Server' and 'Push Client' roles of the Object Push Profile (OPP).

Description: Defined X-999 software acts in accordance with the <specification> defined for both the Push Server and Push Client roles of the Object Push Profile (OPP).

Only when official certification is actually and correctly granted; has the {developer or supplier or any real integrator, whoever it really is doing the integration} completed their task correctly.

This includes correct proven interface to any other related modules specified in the specification.

Stakeholders: Phonebook, Scheduler, Testers, <Product Architect>, Product Planner, Software Engineers, User Interface Designer, Project Team Leader, Company engineers, Developers from other Company product departments which we interface with, the supplier of the TTT, CC. "Other than Owner and Expert. The people we are writing this particular requirement for"

============= Design Relationships ========================
Reuse of Other Design:
Reuse of this Design:
Design Constraints:
Sub-Designs:
============== Impacts Relationships ======================
Impacts [Intended]: Interoperability.
Impacts [Side Effects]:
Impacts [Costs]:
Impacts [Other Designs]:
Value:
Interoperability: Defined As: Certified that this device can exchange information with any other device produced by this project.
============= Impact Estimation/Feedback ======================
Impact Percentage [Interoperability, Estimate]: <100% of Interoperability objective with other devices that support OPP on time is estimated to be the result>.
============== Priority and Risk Management =======================

Assumptions: There are some performance requirements within our certification process regarding probability of connection and transmission etc. that we do not remember <-TG.

Dependencies:
Risks: <none identified>.

We do not 'understand' fully (because we don't have information to hand here) our certification requirements, so we risk that our design will fail certification. <-TG

Priority:
Issues:
============== Location of Specification =======================
Location of Master Specification: <Give the intranet web location of this master specification>.

Design Specification Template <with Hints>

Tag: <Tag name for the design idea>.
Type: {Design Idea, Design Constraint}.
============ Basic Information ========================
Version: <Date or version number>.
Status: <{Draft, SQC Exited, Approved}>.
Quality Level: <Maximum remaining major defects/page, sample size, date>.
Owner: < Role/e-mail/name of person responsible for changes and updates>.
Expert: < Name and contact information for a technical expert, in our organization or otherwise available to us, on this design idea>.
Authority: <Name and contact information for the leading authorities, in our organization or elsewhere, on this technology or strategy. This can include references to papers, books and websites>.
Source: <Source references for the information in this specification. Could include people>.
Gist: <Brief description>.
Description: <Describe the design idea in sufficient detail to support the estimated impacts and costs given below>.
Stakeholders: <Prime stakeholders concerned with this design>.
============= Design Relationships =========================
Reuse of Other Design: <If a currently available component or design is specified, then give its tag or reference code here to indicate that a known component is being reused>.
Reuse of This Design: <If this design is used elsewhere in another system or used several times in this system, then capture the information here>.
Design Constraints: <If this design is a reflection of attempting to adhere to any known design constraints, then that should be noted here with reference one or more of the constraint tags or identities>.
Sub-Designs: <Name tags of any designs, which are subsets of this one, if any>.
============== Impacts Relationships =======================
Impacts [Functions]: <list of functions and subsystems which this design impacts attributes of>.
Impacts [Intended]: <Give a list of the performance requirements that this design idea will impact in a major way, good or bad. The positive impacts are the main justification for the existence of the design idea!>.
Impacts [Side Effects]: <Give a list of the performance requirements that this design idea will impact in a more minor way, good or bad>.
Impacts [Cost]: <Give a list of the budgets that this design idea will impact in a major way>.
Impacts [Other Designs]: <Does this design have any consequences with respect to other designs? Name them at least>.
Value: <Name or quantify value produced, and stakeholders affected by this design. Use Qualifiers>
============== Impact Estimation/Feedback ==================
For each Scalar Requirement in Impacts [Intended] (see above):
Tag: <Tag of a scalar requirement listed in Impacts [Intended]>.
Scale: <Scale for the scalar requirement>.
Scale Impact: <Give estimated or real impact, when implemented, using the defined Scale. That is, given current baseline numeric value, what numeric value will implementing this design idea achieve or what numeric value has been achieved?>.
Scale Uncertainty: <Give estimated optimistic/pessimistic or real ± error margins>.
Percentage Impact: <Convert Scale Impact to Percentage Impact. That is, what percentage of the way to the planned target, relative to the baseline and the planned target will implementing this design idea achieve or, has been achieved? 100% means meeting the defined Plan level on time>.
Percentage Uncertainty: <Convert Scale Uncertainty to Percentage Uncertainty ± deviations>.
Evidence: <Give the observed numeric values, dates, places and other relevant information where you have data about previous experience of using this design idea>.
Source: <Give the person or written source of your evidence>.
Credibility: <Credibility 0.0 low to 1.0 high. Rate the credibility of your estimates, based on the evidence and its source>.
============== Priority and Risk Management ==================
Assumptions: <Any assumptions that have been made>.
Dependencies: <State any dependencies for this design idea>.
Risks: <Name or refer to tags of any factors, which could threaten your estimated impacts>.
Priority: <List the tag names of any design ideas that must be implemented before or after this design idea>.
Issues: <Unresolved concerns or problems in the specification or the system>.
============= Implementation Control =======================
Supplier: < Name actual supplier or list supplier requirements>
Responsible: <Who in or organization is responsible for managing the supplier relation?>
Contract: <Refer to the contract if any, or the contract template>
Test Plan: <Refer to specific test pan for this design>
Implementation Process: <Name any special needs during implementation>
============== Location of Specification ========================
Location of Master Specification: <Give the intranet web location of this master specification>.

Break 10 minutes to 16:00

16- 1700 Impact Estimation Tables

How do we specify a design with *impacts*?

Design Specification Template with Annotation

Tag: <Unique Name Capitalized>

Type: Design Idea.

Version: <date and or version number of last change>

Owner: < originator, champion, expert, maintainer, architect, systems engineer>

Description: <describe the design in a dozen, or more, words. The detail should be sufficient to guarantee the expected impacts and costs estimated below>.

Reuse: <if a currently available component or design is specified, then give it's tag or reference code here to indicate that a known component is being applied>

Constraint: <if this design is a reflection of attempting to adhere to any known design constraints, then that should be noted here with reference one or more of the constraint tags or identities>.

============== Real Expected Impact Section ================

Primary Impacts: <give the main impact or impacts which this design is expected to have on an objective . These are its main justification for existence!>.

Secondary Impacts: <list expected secondary impacts, good or bad>.

Cost Impacts: <give at least rough impacts on defined budget constraints>.

============== More Formal Impact Estimation =================

Real Impact on defined Scale: <give expected impact result on the Scale defined, when implemented>

%Impact on Specific Goal: <Convert real impact to % impact relative to the main planned level: 100% means meets defined Plan level on time>.

± %Uncertainty: <give optimistic/pessimistic % deviation, like ±20%, based on best and worst real observations>.

Evidence: <give the observed numbers, facts, dates, places where you have data about this designs impact>

Source: <give the person or written source of your evidence>

Credibility: <Credibility 0.0 low to 1.0 high. Rate the quality of your estimates, based on the historic data you have>

--------- Repeat this sequence for any other major impact objectives you believe justify the specification effort here.

============= Other Useful Parameters for Design Specification ========

Risks: <name any factors, which can threaten your estimated impact or bring it to the lowest levels specified>

Assumptions: <state any implied unvoiced, threatening assumptions which if false could threaten your estimates>

Expert: < name and give contact (email?) a useful technical expert in our company or otherwise available to us on this design idea>.

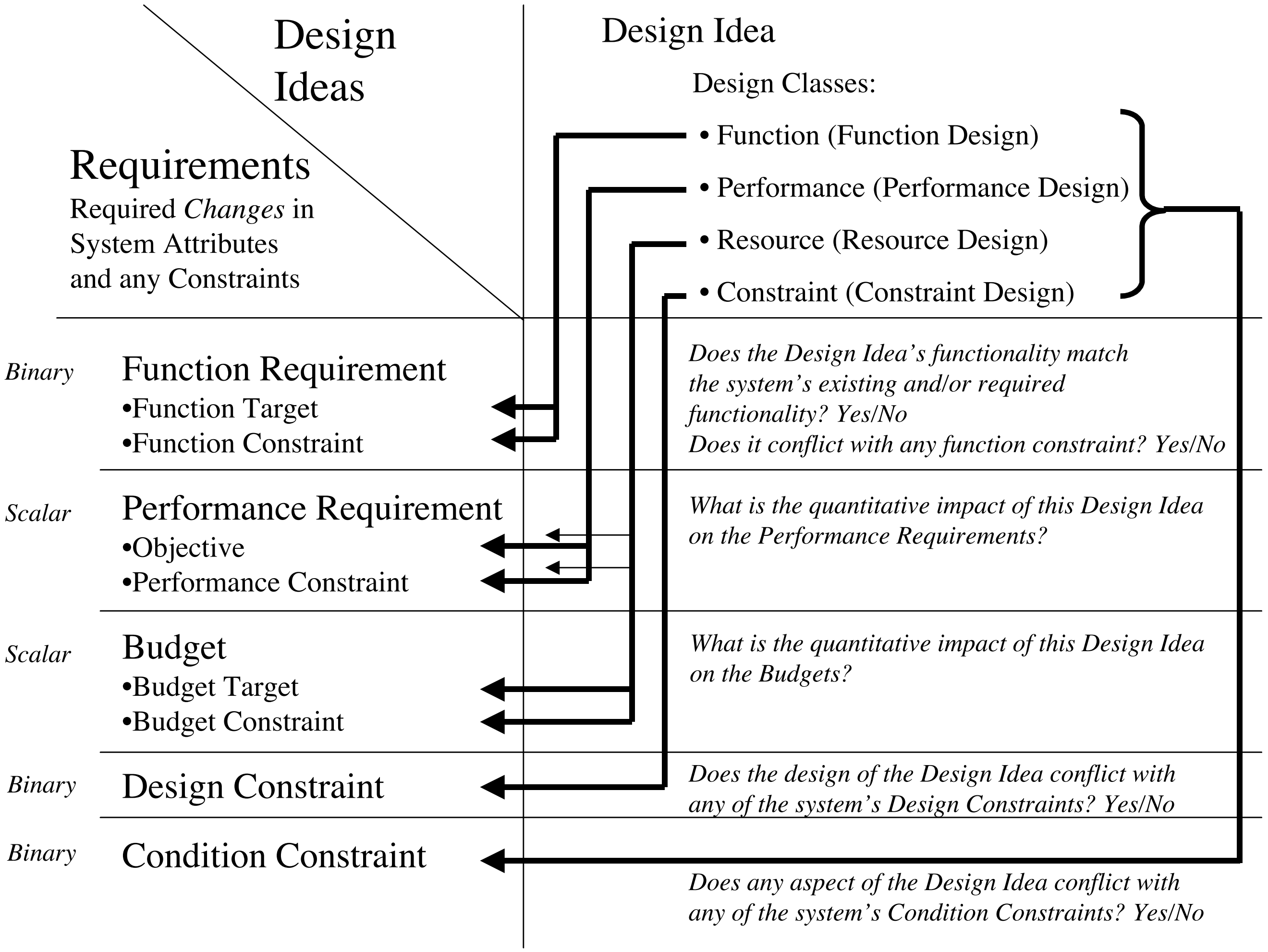
Authority: <name and give contact information to the leading authorities in our co. or elsewhere on this technology. Reference papers or books for example and websites>

Web Location of Master Specification: <give intranet web location of this master specification>.

Exercise in Impact Estimation (of designs, on requirements).

- As a Team: Write an Impact Estimation Table with at least the three Requirements and the three Design Ideas that was specified in detail.
- Each Intercepting cell should have:
  - Impact on Requirement with numbers corresponding to the Requirements Scale, Past & Goal
  - Impact on Requirement in percentage from Past to Goal.
  - ± variation on the above numbers
  - Credibility Number
  - Source
  - Evidence
- Then Add a critical Resource like Cost to develop.
- Calculate which Design Idea gives the most bang for the buck

What are the principles of evaluating a design?



Design Ideas

Requirements
Required *Changes* in
System Attributes
and any Constraints

Design Idea

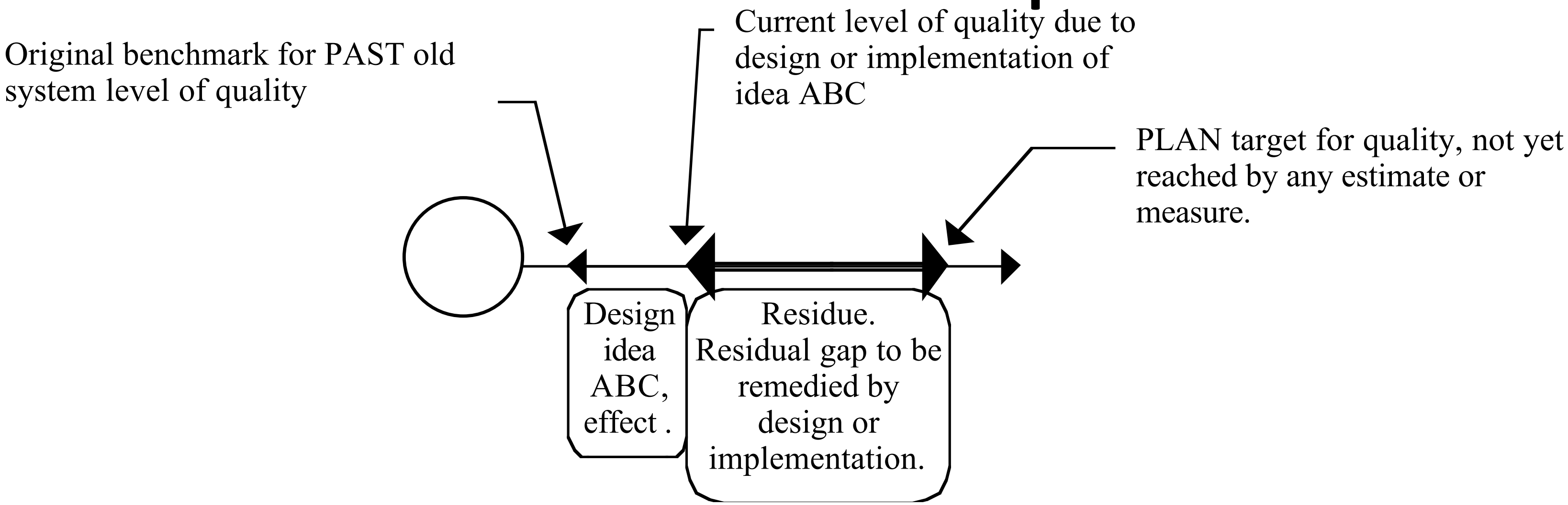
Design Classes:

- Function (Function Design)
- Performance (Performance Design)
- Resource (Resource Design)
- Constraint (Constraint Design)

	Requirements	Design Idea
Binary	Function Requirement •Function Target •Function Constraint	*Does the Design Idea's functionality match the system's existing and/or required functionality? Yes/No* *Does it conflict with any function constraint? Yes/No*
Scalar	Performance Requirement •Objective •Performance Constraint	*What is the quantitative impact of this Design Idea on the Performance Requirements?*
Scalar	Budget •Budget Target •Budget Constraint	*What is the quantitative impact of this Design Idea on the Budgets?*
Binary	Design Constraint	*Does the design of the Design Idea conflict with any of the system's Design Constraints? Yes/No*
Binary	Condition Constraint	*Does any aspect of the Design Idea conflict with any of the system's Condition Constraints? Yes/No*

- Avoid violating constraints
- Meet Target and Function requirements

How do we evaluate a single dimension of impact?


Original benchmark for PAST old system level of quality
Current level of quality due to design or implementation of idea ABC
PLAN target for quality, not yet reached by any estimate or measure.
Design idea ABC, effect .
Residue. Residual gap to be remedied by design or implementation.

- We must estimate or measure the numeric cumulative impact of the design
  - on a defined Scale,
  - using a defined Meter,
  - with respect to target and constrain t levels.

How can we evaluate all dimensions of impact?

Design Ideas **Objectives**	Central	Youth	Facts	London	Diploma	Events	Discounts	Total
Participation	80%±50%	60%±70%	0%±50%	0%±50%	30%±50%	20%±50%	30%±50%	220%±370%
Representation	80%±50%	80%±50%	10%±50%	0%±50%	10%±50%	20%±50%	50%±40%	250%±340%
Information	0%±50%	20%±40%	80%±50%	0%±20%	20%±50%	0%±50%	0%±30%	120%±290%
Conviction	0%±10%	20%±50%	60%±30%	80%±50%	10%±50%	80%±50%	0%±50%	250%±290%
Influence	0%±50%	40%±40%	60%±50%	0%±50%	80%±50%	80%±50%	0%±50%	260%±340%
Fun	50%±50%	40%±50%	10%±50%	0%±0%	0%±0%	80%±50%	0%±0%	180%±200%
Total	210% ±260%	260% ±300%	220% ±280%	80% ±220%	150% ±250%	270% ±300%	80% ±220%	
Budgets								
Cost	10%	10%	10%	10%	1%±5%	50%±50%	80%±50%	171%±105%
Benefit–to-Cost Ratio	210%/10%	260%/10%	220%/10%	80%/10%	150/1	270/50	80/80	

- We can use an Impact (Estimation) Table

What uses can we put impact estimation to?

IE can be used for a wide variety of purposes including:

1. Evaluating a single design idea. How good is the idea for us?
2. Comparing two or more design ideas to find a winner, or set of winners. Use IE, if you want to set up an argument against a prevailing popular, but weak design idea!
3. Gaining an architectural overview of the impact of all the design ideas on all the objectives and budgets. Are there any negative side effects? What is the cumulative effect?
4. Obtaining systems engineering views of specific components, or specific performance aspects.

Are we going to achieve the reliability levels?

5. Analyzing risk: evaluating a design with regard to 'worst case' uncertainty and minimum credibility.
6. Planning evolutionary project delivery steps with regard to value and cost.
7. Monitoring, for project management accounting purposes, the progress of individual evolutionary project delivery steps and, the progress to date compared against the requirement specification or management objectives.
8. Predicting future costs, project timescales and performance levels.
9. Understanding organizational responsibility in terms of performance and budgets by organizational function.

   In 1992, Steve Poppe pioneered this use at executive level while at British Telecom, North America.
10. Achieving rigorous quality control of a design specification prior to management reviews and approval.
11. Presenting ideas to committees, management boards, senior managers, review boards and customers for approval.
12. Identifying which parts of the design are the weakest (risk analysis). If there are no obvious alternative design ideas, any 'weak links' should be tried out earliest, in case they do not work well (risk management). This impacts scheduling.
13. Enabling configuration management of design, design changes, and change consequences.
14. Permitting delegation of decision-making to teams. Teams can achieve better internal progress control using IE, than they can from repeatedly making progress reports to others, and acting on others' feedback.
15. Presenting overviews of very large, complex projects and systems by using hierarchical IE tables. Aim for a one page top-level IE view for senior management.
16. Enabling cross-organizational co-operation by presenting overviews of how the design ideas of different projects contribute towards corporate objectives. Any common and conflicting design ideas can be identified. This is important from a customer viewpoint; different projects might well be delivering to the same customer interface.
17. Controlling the design process. You can see what you need, and see if your idea has it by using an IE table. For example, which design idea contributes best to achieving usability? Which one costs too much?
18. Strengthening design. You can see where your design ideas are failing to impact sufficiently on the objectives; and this can provoke thought to discover new design ideas or modify existing ones.
19. Helping informal reasoning and discussion of ideas by providing a framework model in our minds of how the design is connected to the requirements.
20. Strengthening the specified requirements. Sometimes, you can identify a design idea, that has a great deal of popular support, but doesn't appear to impact your requirements. You should investigate the likely impacts of the design idea with a view to identifying additional stakeholder requirements. This may provide the underlying reason for the popular support. You might also identify additional types of stakeholders.

How does Impact Estimation relate to Planguage?

Learning:

Ambition: Make it substantially easier for our users to learn tasks <- Marketing.

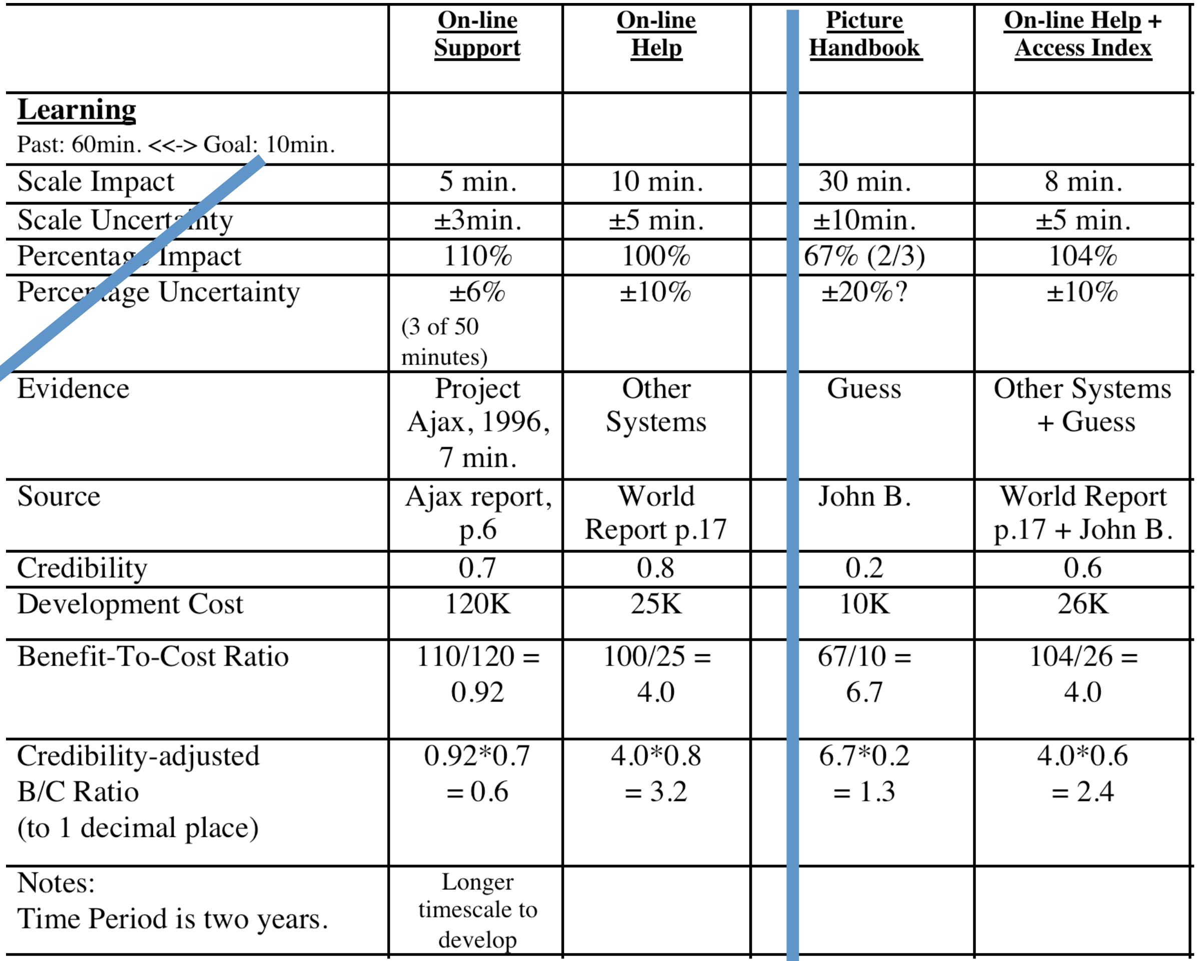
Scale: Average time for a defined [User Type: default UK telesales trainee] to learn a defined [User Task: default Response] using <our product's instructional aids>.

Response: Task: Give correct answer to simple request.

Past [last year]: 60 minutes.

GN: **Goal** [By start of next year]: 20 minutes.

GA: **Goal** [By start of year after next]: 10 minutes.



	On-line Support	**On-line Help**	**Picture Handbook**	**On-line Help** + **Access Index**
Learning Past: 60min. <<-> Goal: 10min.				
Scale Impact	5 min.	10 min.	30 min.	8 min.
Scale Uncertainty	±3min.	±5 min.	±10min.	±5 min.
Percentage Impact	110%	100%	67% (2/3)	104%
Percentage Uncertainty	±6% (3 of 50 minutes)	±10%	±20%?	±10%
Evidence	Project Ajax, 1996, 7 min.	Other Systems	Guess	Other Systems + Guess
Source	Ajax report, p.6	World Report p.17	John B.	World Report p.17 + John B.
Credibility	0.7	0.8	0.2	0.6
Development Cost	120K	25K	10K	26K
Benefit-To-Cost Ratio	110/120 = 0.92	100/25 = 4.0	67/10 = 6.7	104/26 = 4.0
Credibility-adjusted B/C Ratio (to 1 decimal place)	0.92*0.7 = 0.6	4.0*0.8 = 3.2	6.7*0.2 = 1.3	4.0*0.6 = 2.4
Notes: Time Period is two years.	Longer timescale to develop			

Picture Handbook: **Gist**: Produce a radically changed handbook that uses pictures and concrete examples to *instruct,* without the need for *any* other text.

end