EARLY SIZING OF REQUIREMENTS FOR ESTIMATION PURPOSES

MODULE 3: EARLY SIZING & NON-FUNCTIONAL REQUIREMENTS (NFR)

Tutorial by Dr. ALAIN ABRAN

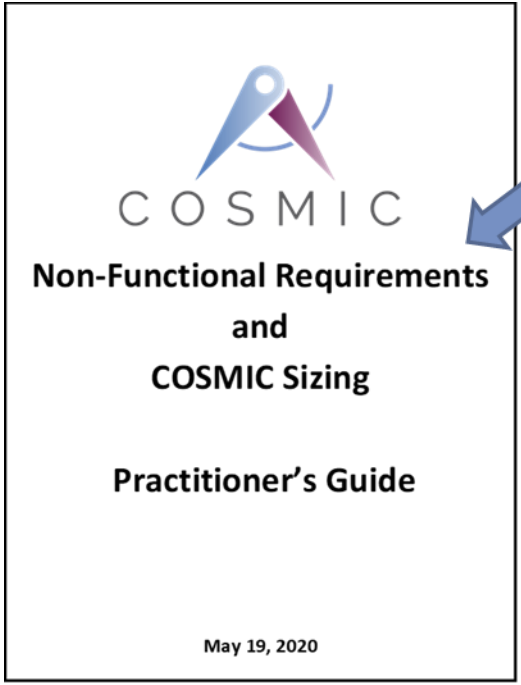
IWSM-MENSURA 2022, IZMIR (TURKEY)

A Tutorial with COSMIC Sizing – ISO 19761

Introduction

- Presents Early Sizing of software functions derived from System NFR (Non functional requirements)

  - This tutorial **does not** include **effort estimation**.



COSMIC

Non-Functional Requirements
and
COSMIC Sizing

Practitioner's Guide

May 19, 2020

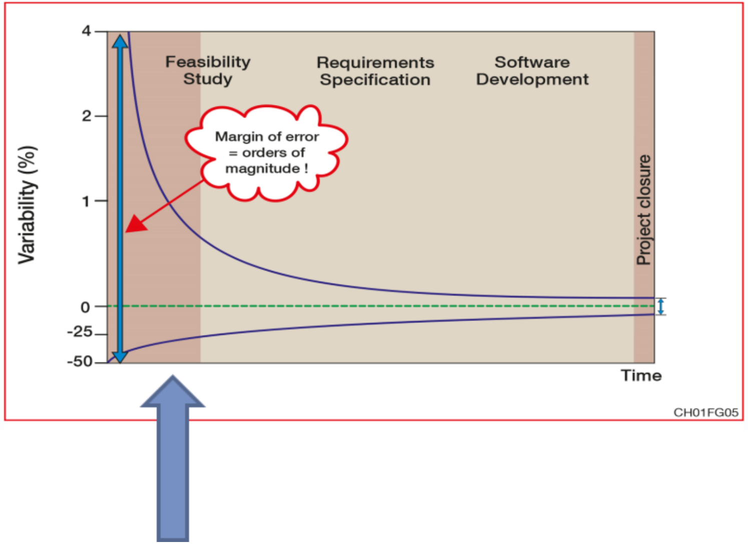
List of Topics

1. Key Concepts
2. From System NFR to Software Functions
3. Examples
4. EcoSystems & Architecture implementing Systems NFR

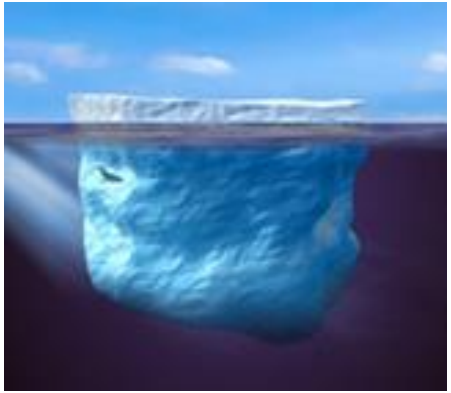
Key concepts

Early in the lifecycle:

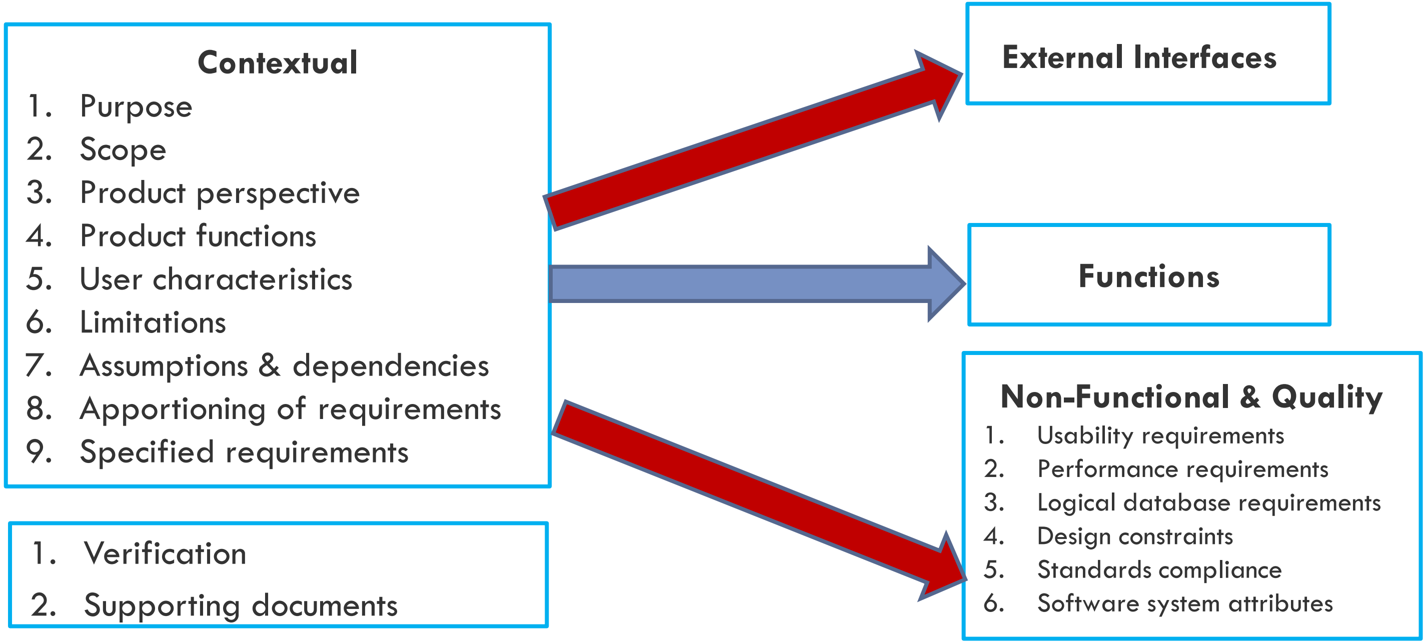
- ➢ Requirements do not describe the full scope of functionality of the software with all the necessary functional details.
  - ➢ Most of the time: requirements will be detailed & changed as the project moves through the life cycle.



Key Concept: from Simple to Complex Software



What is visible at Early requirements Phase



Contextual

1. Purpose
2. Scope
3. Product perspective
4. Product functions
5. User characteristics
6. Limitations
7. Assumptions & dependencies
8. Apportioning of requirements
9. Specified requirements

1. Verification
2. Supporting documents

External Interfaces

Functions

Non-Functional & Quality

1. Usability requirements
2. Performance requirements
3. Logical database requirements
4. Design constraints
5. Standards compliance
6. Software system attributes

INTERNATIONAL STANDARD **ISO/IEC/ IEEE 29148**
First edition 2011-12-01

Systems and software engineering — Life cycle processes — Requirements engineering

Ingénierie des systèmes et du logiciel — Processus du cycle de vie — Ingénierie des exigences

ISO-IEEE 25148

List of Topics

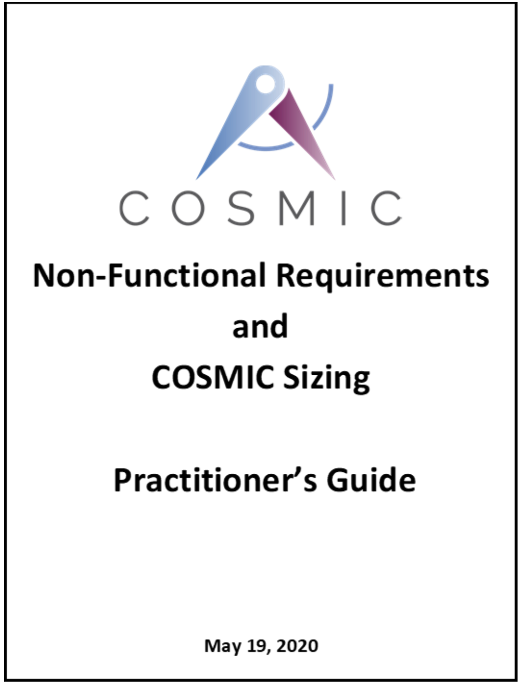
1. Key Concepts
2. **From System NFR to Software Functions**
3. Examples
4. EcoSystems & Architecture implementing Systems NFR

Non-functional requirements (distinctions)

- There are types of requirements that cannot become functional:
  - Organizational constraints (location of operations, equipment used, compliance with standards)
  - Certain environmental constraints (e.g. interoperability)
  - Implementation constraints (development language, delivery date)

- There are types of requirements that can be non-functional and become operational:
  - Quality constraints (reliability, efficiency)
  - Environmental constraints (safety)

Non-Functional Requirements (NFR)



COSMIC

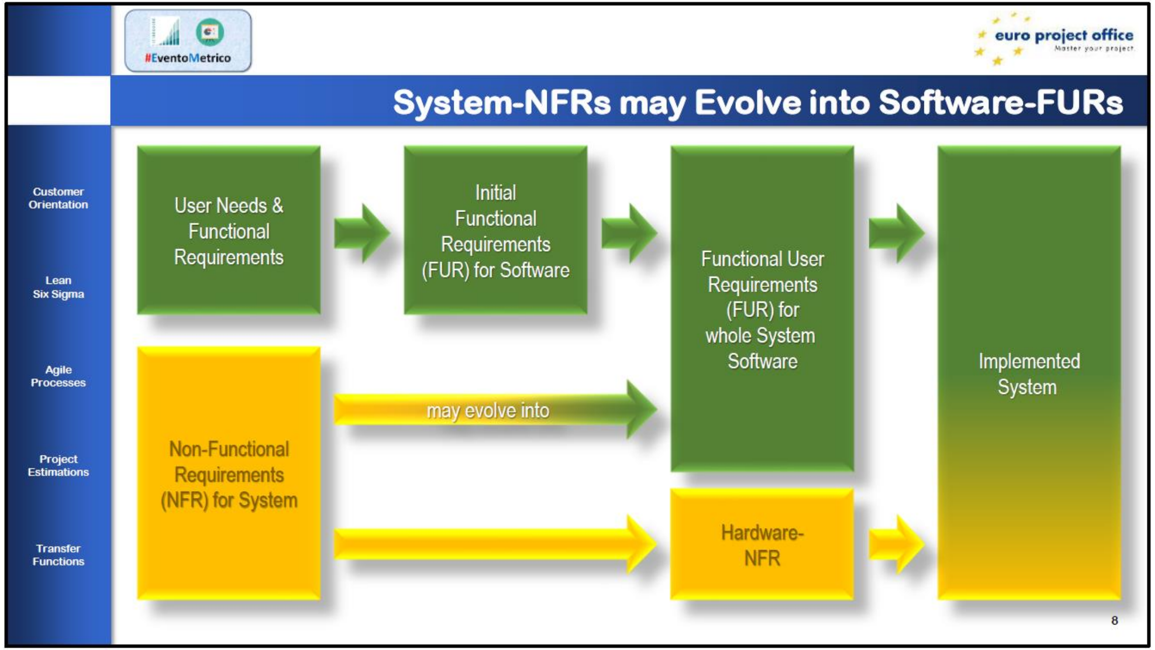
Non-Functional Requirements

and

COSMIC Sizing

Practitioner's Guide

May 19, 2020



#EventoMetrico

euro project office
Master your project

System-NFRs may Evolve into Software-FURs

Customer Orientation

Lean Six Sigma

Agile Processes

Project Estimations

Transfer Functions

User Needs & Functional Requirements

Initial Functional Requirements (FUR) for Software

Functional User Requirements (FUR) for whole System Software

Implemented System

Non-Functional Requirements (NFR) for System

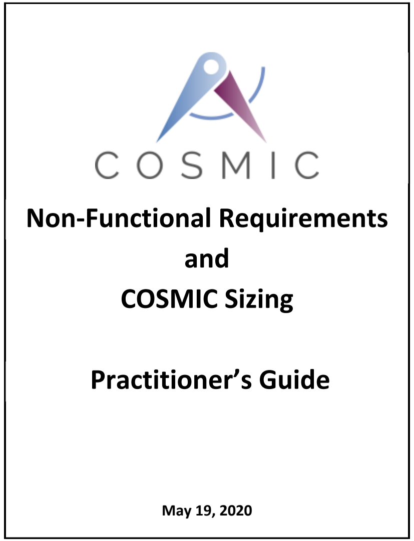
may evolve into

Hardware-NFR

8

Dr. Thomas Fehlmann, Dec. 2020

NFR in Systems Standards



COSMIC

Non-Functional Requirements

and

COSMIC Sizing

Practitioner's Guide

May 19, 2020

A- List of System-NFRs

Id.	NFR	Standards used
1	Performance	ECSS-ISO-IEEE
2	Maintainability	ECSS-ISO-IEEE
3	Portability	ECSS-ISO-IEEE
4	Security	ECSS-ISO-IEEE
5	Reliability	ECSS-ISO-IEEE
6	Interfaces	ECSS-ISO-IEEE
7	Operations	ECSS-ISO-IEEE
8	Adaptation & Installation	ECSS-ISO-IEEE
9	Safety	ECSS-ISO-IEEE
10	Resources	ECSS-ISO-IEEE
11	Human Factors	ECSS-ISO-IEEE
12	Data Definition & Data Bases	ECSS-ISO-IEEE
13	Configuration	ECSS-ISO-IEEE
14	Design	ECSS-ISO-IEEE

Some System NFR can be allocated to software & sized

Contextual

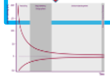
1. Purpose
2. Scope
3. Product perspective
4. Product functions
5. User characteristics
6. Limitations
7. Assumptions & dependencies
8. Apportioning of requirements
9. Specified requirements

External Interfaces

Software Functions

Functions

Detailed Software Functions



Non-Functional & Quality

1. Usability req.
2. Performance req.
3. Logical database req.
4. Design constraints
5. Standards compliance
6. Software system attributes

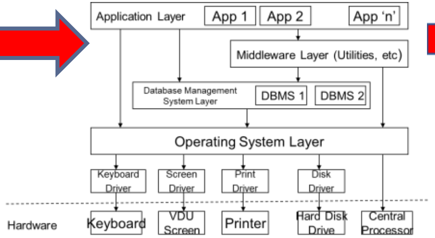
COSMIC

The COSMIC Functional Size Measurement Method
Version 4.0.1

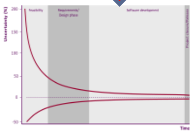
Guideline on
Non-Functional & Project
Requirements

How to consider non-functional and project requirements in software project performance measurement, benchmarking and estimating

Version 1.
November 2015



Software Functions

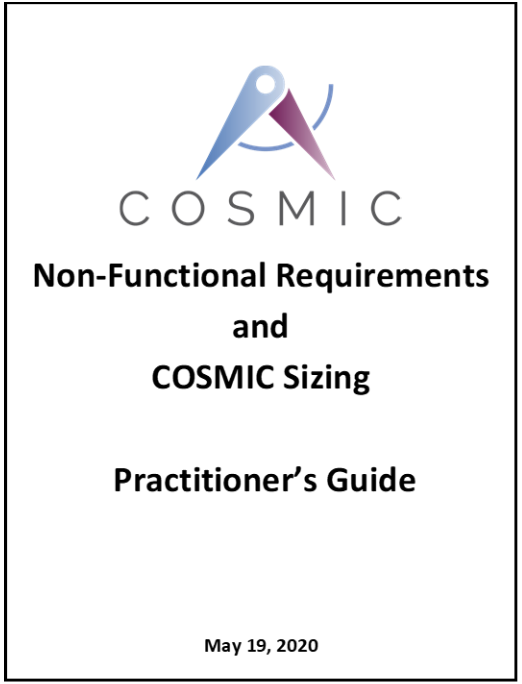


1. Verification
2. Supporting doc.

List of Topics

1. Key Concepts
2. From System NFR to Software Functions
3. **Examples**
4. EcoSystems & Architecture implementing Systems NFR

NFR Example: System Security Requirements

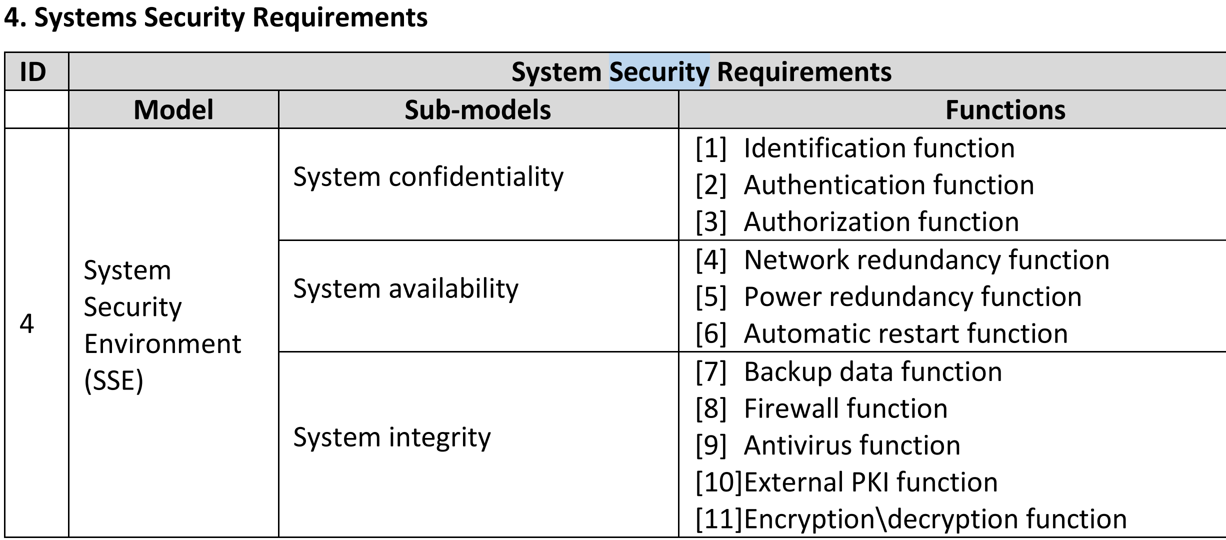


COSMIC

Non-Functional Requirements
and
COSMIC Sizing

Practitioner's Guide

May 19, 2020



4. Systems Security Requirements

ID	System Security Requirements		
	Model	**Sub-models**	**Functions**
4	System Security Environment (SSE)	System confidentiality	[1] Identification function [2] Authentication function [3] Authorization function
		System availability	[4] Network redundancy function [5] Power redundancy function [6] Automatic restart function
		System integrity	[7] Backup data function [8] Firewall function [9] Antivirus function [10]External PKI function [11]Encryption\decryption function

System Security NFR -Example

ATM System security requirements defined in Meridji *et al.* :

- Requirement 1: the customer must insert his bank card into the ATM, allowing the latter to identify the customer.
- Requirement 2: After the customer inserts the card, the system extracts the encrypted PIN and asks the user to enter their 8-digit PIN using the keypad to authenticate their identity.
- Requirement 3: If the customer is authenticated, the ATM system must ensure that the customer's daily withdrawals do not exceed $800. Once verified, the customer can access the system.

Security Identification of requirements & control plans

ATM Security Services has a number of control plans:

- User-plan **Authentication**: Determines at the start of the connection that the calling and/or called party identities are authentic.
- User-plan **Privacy**: Provides cryptographic mechanisms that protect "user" data on a VC (Virtual Channel) from unauthorized disclosure.
- User-plan **Data integrity**: Provides a mechanism to detect modification of data values or sequences of data values, even in the presence of malicious modification threats.
- User-plan **Access control**: Requires mechanisms to convey access control information used during connection establishment, as well as mechanisms within ATM components to use this information to determine whether access to the connection must be granted.

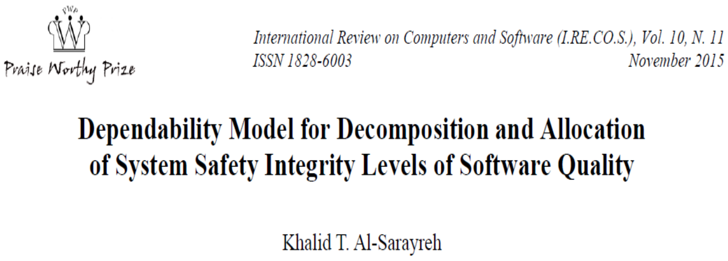
Measurement of requirements

- User-plan **Authentication**: Determines at the start of the connection that the calling and/or called party identities are authentic.
  - To identify the caller, we have an Entry, a Read and an Exit (*).
- User-plan **Privacy**: Provides cryptographic mechanisms that protect 'user' data on a VC from unauthorized disclosure.
  - To do this there is an Entry (user code), 2 Reads (user and VC) and an eXit (recognition/non-recognition of the user) (*)

(*) Actually more complex. Here we simplify a lot by considering only 1 or 2 data groups.

Safety (NFR)

Example from: Khalid T. Al-Sarayreh



Praise Worthy Prize

International Review on Computers and Software (I.RE.CO.S.), Vol. 10, N. 11
ISSN 1828-6003 *November 2015*

Dependability Model for Decomposition and Allocation of System Safety Integrity Levels of Software Quality

Khalid T. Al-Sarayreh

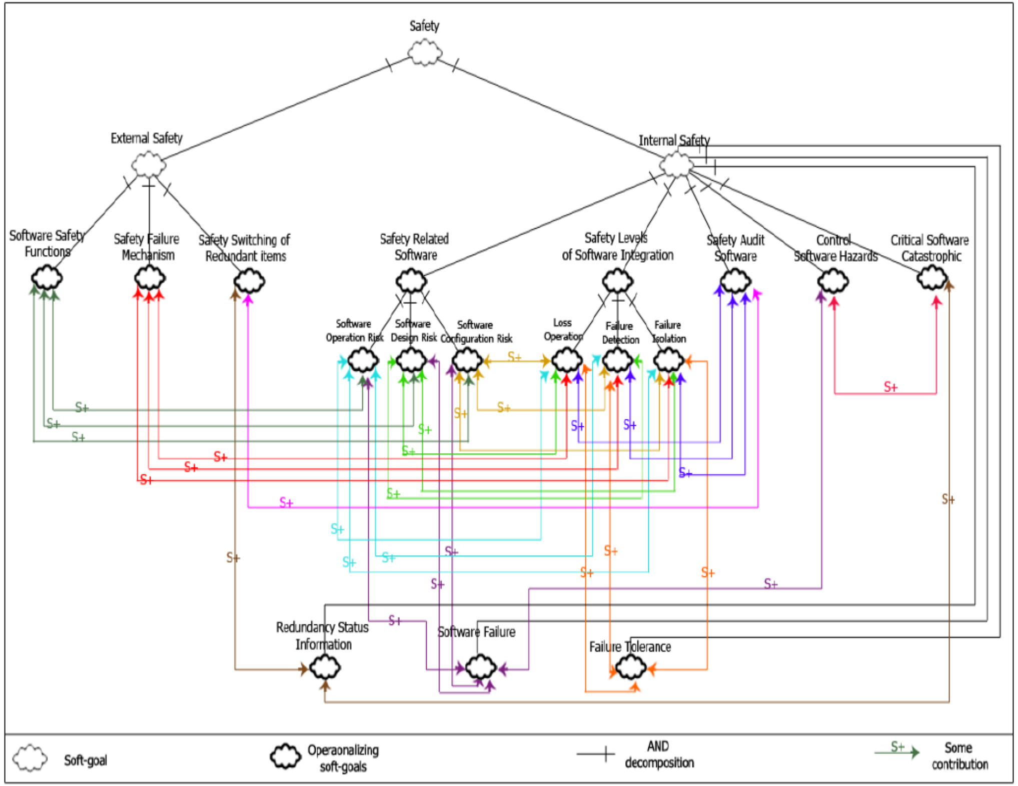
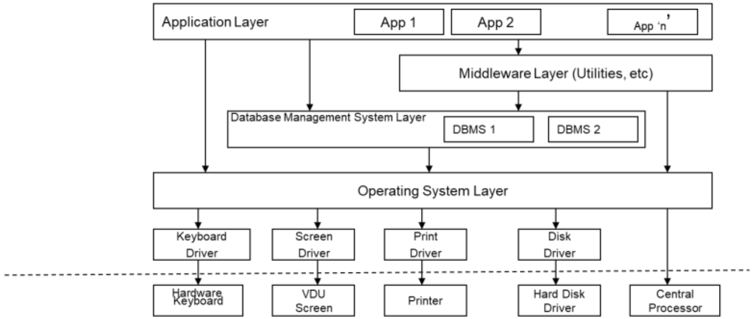


Fig. 9. System Safety Quality Model Allocated to Software Safety Requirements

Portability (NFR)



Computer Standards & Interfaces 35 (2013) 380–395

Contents lists available at SciVerse ScienceDirect

Computer Standards & Interfaces

journal homepage: www.elsevier.com/locate/csi

A standards-based reference framework for system portability requirements

Alain Abran [a,*], Khalid T. Al-Sarayreh [b], Juan J. Cuadrado-Gallego [a,c]

A. Abran et al. / Computer Standards & Interfaces 35 (2013) 380–395

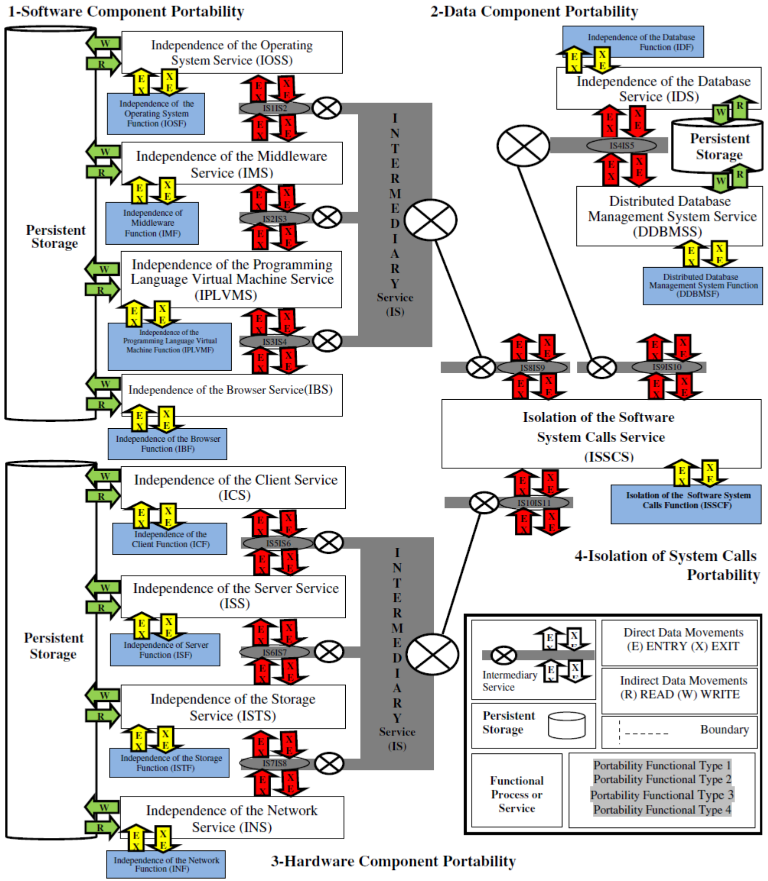
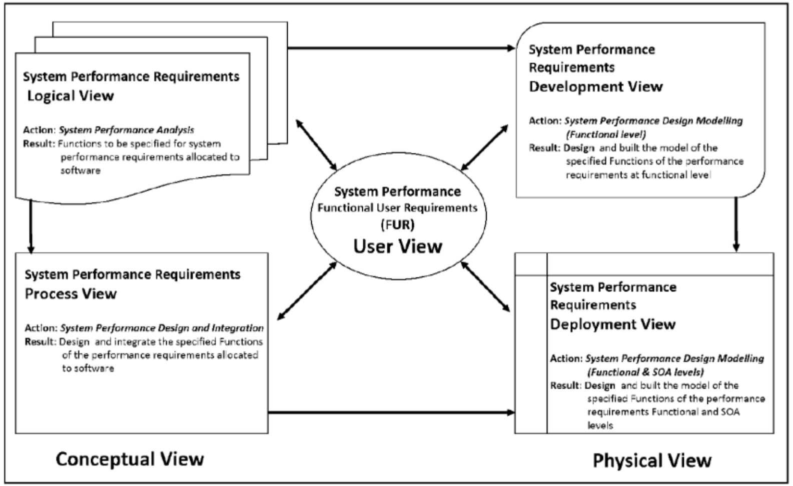

1-Software Component Portability
Persistent Storage
W
R
E X
X E
Independence of the Operating System Service (IOSS)
Independence of the Operating System Function (IOSF)
IS1IS2
Independence of the Middleware Service (IMS)
Independence of Middleware Function (IMF)
IS2IS3
Independence of the Programming Language Virtual Machine Service (IPLVMS)
Independence of the Programming Language Virtual Machine Function (IPLVMF)
IS3IS4
Independence of the Browser Service(IBS)
Independence of the Browser Function (IBF)
INTERMEDIARY Service (IS)
2-Data Component Portability
Independence of the Database Function (IDF)
Independence of the Database Service (IDS)
IS4IS5
Persistent Storage
Distributed Database Management System Service (DDBMSS)
Distributed Database Management System Function (DDBMSF)
IS8IS9
IS9IS10
Isolation of the Software System Calls Service (ISSCS)
Isolation of the Software System Calls Function (ISSCF)
IS10IS11
4-Isolation of System Calls Portability
Persistent Storage
Independence of the Client Service (ICS)
Independence of the Client Function (ICF)
IS5IS6
Independence of the Server Service (ISS)
Independence of Server Function (ISF)
IS6IS7
Independence of the Storage Service (ISTS)
Independence of the Storage Function (ISTF)
IS7IS8
Independence of the Network Service (INS)
Independence of the Network Function (INF)
INTERMEDIARY Service (IS)
Intermediary Service
Direct Data Movements (E) ENTRY (X) EXIT
Indirect Data Movements (R) READ (W) WRITE
Persistent Storage
Boundary
Functional Process or Service
Portability Functional Type 1
Portability Functional Type 2
Portability Functional Type 3
Portability Functional Type 4
3-Hardware Component Portability

Fig. 11. COSMIC-SOA reference architectural model of system portability requirement.

Performance (NFR)



System Performance Requirements
Logical View

Action: *System Performance Analysis*
Result: Functions to be specified for system performance requirements allocated to software

System Performance Requirements
Development View

Action: *System Performance Design Modelling (Functional level)*
Result: Design and built the model of the specified Functions of the performance requirements at functional level

System Performance
Functional User Requirements
(FUR)
User View

System Performance Requirements
Process View

Action: *System Performance Design and Integration*
Result: Design and integrate the specified Functions of the performance requirements allocated to software

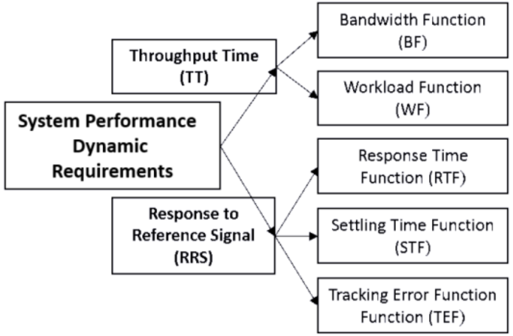
System Performance
Requirements
Deployment View

Action: *System Performance Design Modelling (Functional & SOA levels)*
Result: Design and built the model of the specified Functions of the performance requirements Functional and SOA levels

Conceptual View

Physical View

Figure 4. System performance requirements from four different views



System Performance Dynamic Requirements

Throughput Time (TT)

Response to Reference Signal (RRS)

Bandwidth Function (BF)

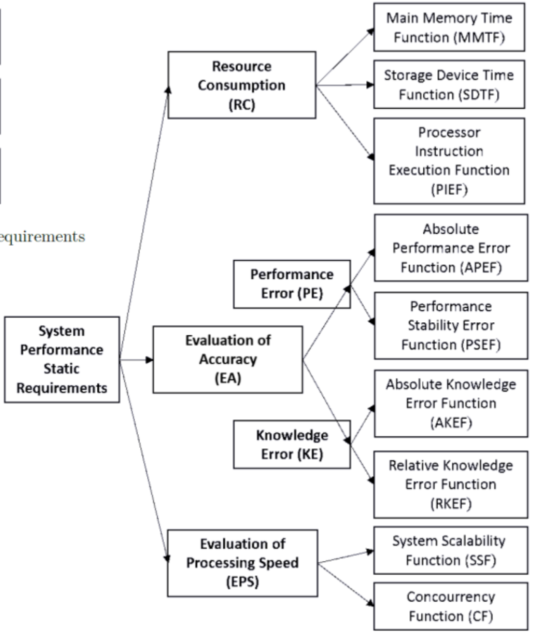
Workload Function (WF)

Response Time Function (RTF)

Settling Time Function (STF)

Tracking Error Function Function (TEF)

Integrated model of ECSS and IEEE system performance dynamic requirements and related functions



System Performance Static Requirements

Resource Consumption (RC)

Evaluation of Accuracy (EA)

Evaluation of Processing Speed (EPS)

Main Memory Time Function (MMTF)

Storage Device Time Function (SDTF)

Processor Instruction Execution Function (PIEF)

Performance Error (PE)

Knowledge Error (KE)

Absolute Performance Error Function (APEF)

Performance Stability Error Function (PSEF)

Absolute Knowledge Error Function (AKEF)

Relative Knowledge Error Function (RKEF)

System Scalability Function (SSF)

Concourrency Function (CF)

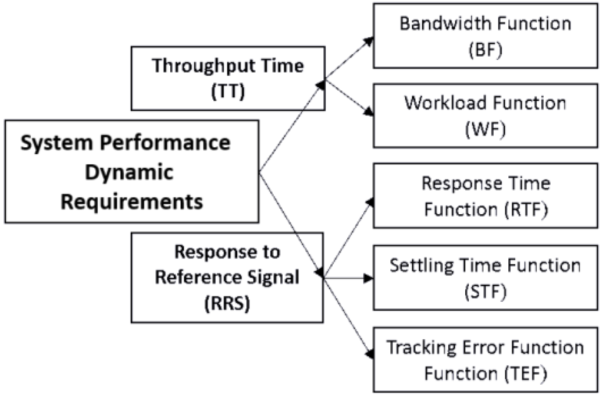
Performance Example

System requirement: 7-day software **availability** level - Target = 95%

- The data is in a log. Develop a weekly report from the log data on the WEB site.
- Detailed requirements (& data movements):
  - an Entry for the information coming from the log
  - an eXit for the information presented to the user
  - a Read and an Exit for the error message
  - ➢ Functional = 4 CFP (COSMIC Function Points)

Exercise: What if availability and date belong to two different data groups? Could you do the CFP calculation.

Performance (NFR)


System Performance Dynamic Requirements
Throughput Time (TT)
Response to Reference Signal (RRS)
Bandwidth Function (BF)
Workload Function (WF)
Response Time Function (RTF)
Settling Time Function (STF)
Tracking Error Function Function (TEF)

5. Integrated model of ECSS and IEEE system performance dynamic requirements and related functions

Example 2: Performance NFR that can be translated into different functional requirements:

- Bandwidth Function:
  - Bandwidth history log to be created.
- Workload Function: A history of the workload.
- Response time: A log of the response time.
- Setting the time function:
  - an interface to set the time function and
  - a process to show the results
- Tracking Error Function: A log of tracking errors

We identify at least 6 functional processes.
Exercise: How many CFP?

Exercise

- This exercise is about defining four quality standards and showing how you can measure them as a functional process.
  - If this is not possible, explain why.
  - Use a document you know.

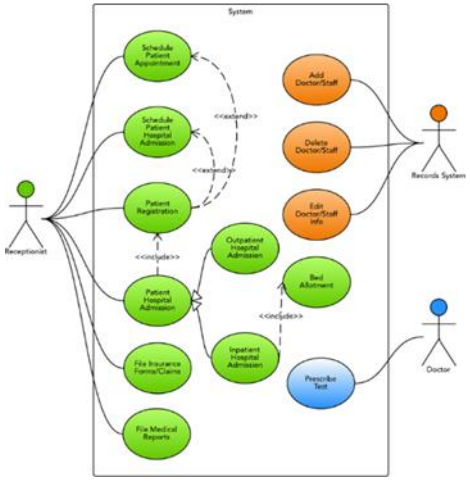
Note: it is always important to keep traceability.

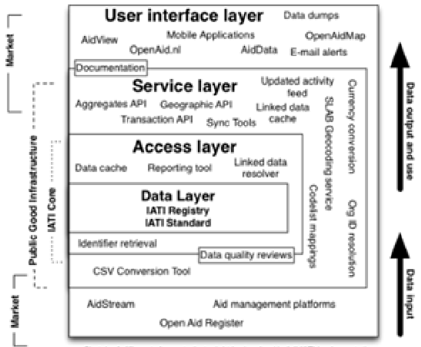
COSMIC

List of Topics

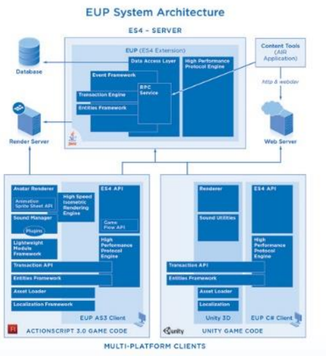
1. Key Concepts
2. From System NFR to Software Functions
3. Examples
4. **EcoSystems & Architecture implementing Systems NFR**

Software: Applications & Infrastructure





Sketch of different elements that might be involved in full IATI implementation.
Tim Davies (@timdavies), July 2012



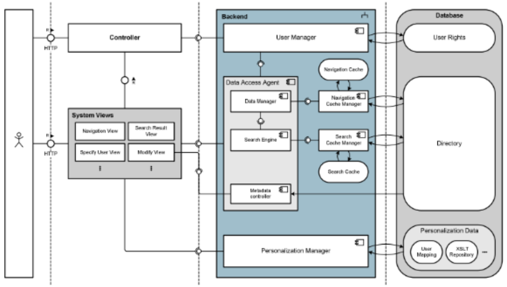
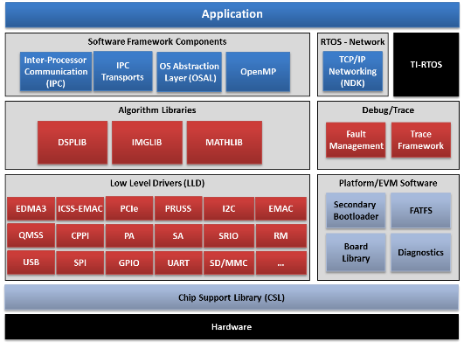
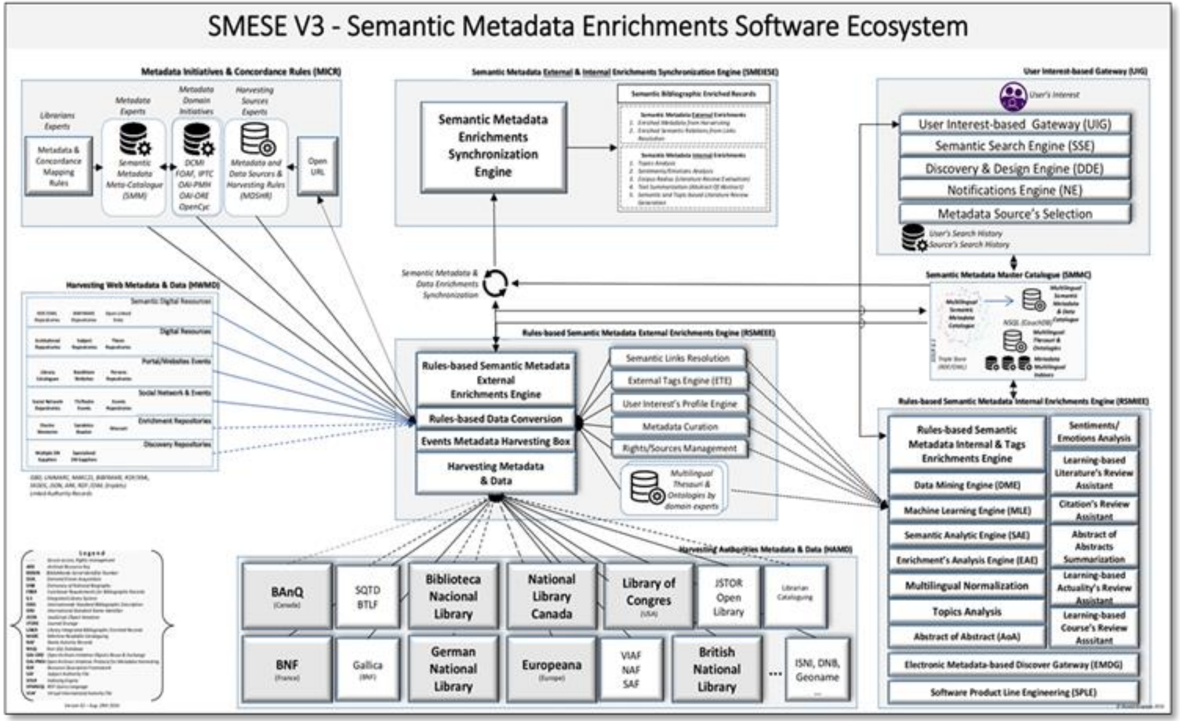


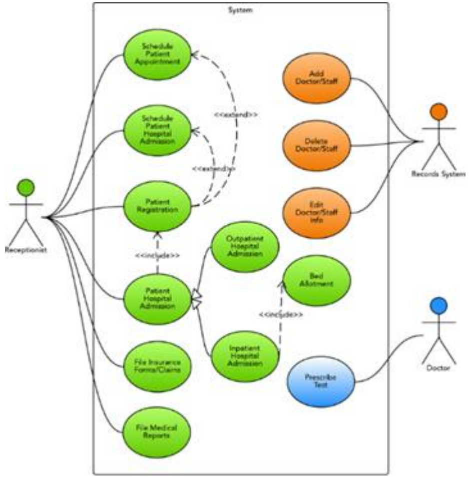
Figure 2–2 Design Level Component/Block diagram

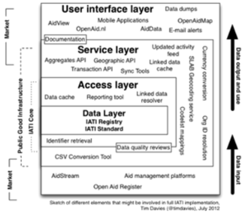


Software Ecosystem

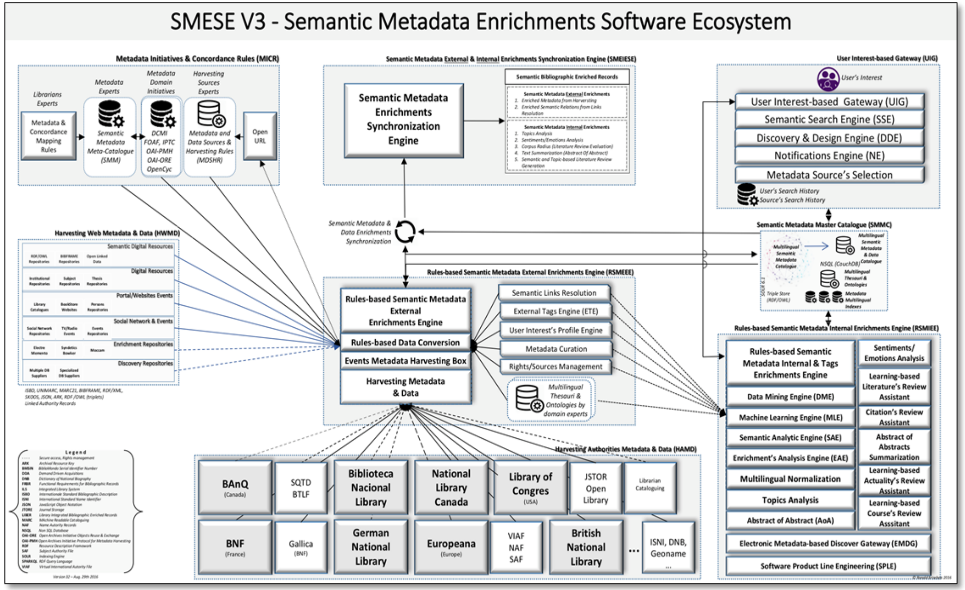

SMESE V3 - Semantic Metadata Enrichments Software Ecosystem
Metadata Initiatives & Concordance Rules (MICR)
Librarians Experts
Metadata & Concordance Mapping Rules
Metadata Experts
Semantic Metadata Meta-Catalogue (SMM)
Metadata Domain Initiatives
DCMI FOAF, IPTC OAI-PMH OAI-ORE OpenCyc
Harvesting Sources Experts
Metadata and Data Sources & Harvesting Rules (MDSHR)
Open URL
Semantic Metadata External & Internal Enrichments Synchronization Engine (SMEIESE)
Semantic Metadata Enrichments Synchronization Engine
Semantic Bibliographic Enriched Records
Semantic Metadata External Enrichments
Semantic Metadata Internal Enrichments
User Interest-based Gateway (UIG)
User's Interest
User Interest-based Gateway (UIG)
Semantic Search Engine (SSE)
Discovery & Design Engine (DDE)
Notifications Engine (NE)
Metadata Source's Selection
User's Search History
Source's Search History
Semantic Metadata & Data Enrichments Synchronization
Semantic Metadata Master Catalogue (SMMC)
Harvesting Web Metadata & Data (HWMD)
Semantic Digital Resources
Digital Resources
Portal/Websites Events
Social Network & Events
Enrichment Repositories
Discovery Repositories
Rules-based Semantic Metadata External Enrichments Engine (RSMEEE)
Rules-based Semantic Metadata External Enrichments Engine
Rules-based Data Conversion
Events Metadata Harvesting Box
Harvesting Metadata & Data
Semantic Links Resolution
External Tags Engine (ETE)
User Interest's Profile Engine
Metadata Curation
Rights/Sources Management
Multilingual Thesauri & Ontologies by domain experts
Rules-based Semantic Metadata Internal Enrichments Engine (RSMIEE)
Rules-based Semantic Metadata Internal & Tags Enrichments Engine
Data Mining Engine (DME)
Machine Learning Engine (MLE)
Semantic Analytic Engine (SAE)
Enrichment's Analysis Engine (EAE)
Multilingual Normalization
Topics Analysis
Abstract of Abstract (AoA)
Sentiments/ Emotions Analysis
Learning-based Literature's Review Assistant
Citation's Review Assistant
Abstract of Abstracts Summarization
Learning-based Actuality's Review Assistant
Learning-based Course's Review Assistant
Electronic Metadata-based Discover Gateway (EMDG)
Software Product Line Engineering (SPLE)
Harvesting Authorities Metadata & Data (HAMD)
BAnQ
SQTD BTLF
Biblioteca Nacional Library
National Library Canada
Library of Congres
JSTOR Open Library
BNF
Gallica
German National Library
Europeana
VIAF NAF SAF
British National Library
...
ISNI, DNB, Geoname
Legend

Software: Applications & Infrastructure & Ecosystem





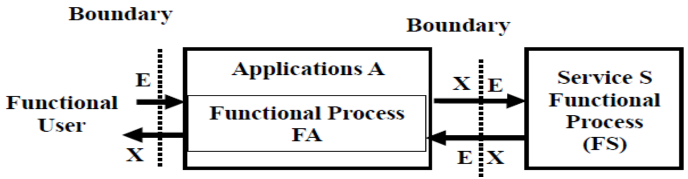
Sketch of different elements that might be involved in full IATI implementation. Tim Davies (@timdavies), July 2012


SMESE V3 - Semantic Metadata Enrichments Software Ecosystem
Metadata Initiatives & Concordance Rules (MICR)
Librarians Experts
Metadata & Concordance Mapping Rules
Metadata Experts
Semantic Metadata Meta-Catalogue (SMM)
Metadata Domain Initiatives
DCMI FOAF, IPTC OAI-PMH OAI-ORE OpenCyc
Harvesting Sources Experts
Metadata and Data Sources & Harvesting Rules (MDSHR)
Open URL
Semantic Metadata External & Internal Enrichments Synchronization Engine (SMEIESE)
Semantic Metadata Enrichments Synchronization Engine
Semantic Bibliographic Enriched Records
Semantic Metadata External Enrichments
1. Enriched Metadata from Harvesting
2. Enriched Semantic Relations from Links Resolution
Semantic Metadata Internal Enrichments
1. Topics Analysis
2. Sentiments/Emotions Analysis
3. Corpus Radius (Literature Review Evaluation)
4. Text Summarization (Abstract Of Abstract)
5. Semantic and Topic-based Literature Review Generation
User Interest-based Gateway (UIG)
User's Interest
User Interest-based Gateway (UIG)
Semantic Search Engine (SSE)
Discovery & Design Engine (DDE)
Notifications Engine (NE)
Metadata Source's Selection
User's Search History
Source's Search History
Semantic Metadata & Data Enrichments Synchronization
Semantic Metadata Master Catalogue (SMMC)
Multilingual Semantic Metadata Catalogue
Multilingual Semantic Metadata & Data Catalogue
NSQL (CouchDB)
Multilingual Thesauri & Ontologies
Triple Store (RDF/OWL)
Metadata Multilingual Indexes
SOLR 6.1
Harvesting Web Metadata & Data (HWMD)
Semantic Digital Resources
Digital Resources
Portal/Websites Events
Social Network & Events
Enrichment Repositories
Discovery Repositories
Rules-based Semantic Metadata External Enrichments Engine (RSMEEE)
Rules-based Semantic Metadata External Enrichments Engine
Rules-based Data Conversion
Events Metadata Harvesting Box
Harvesting Metadata & Data
Semantic Links Resolution
External Tags Engine (ETE)
User Interest's Profile Engine
Metadata Curation
Rights/Sources Management
Multilingual Thesauri & Ontologies by domain experts
Rules-based Semantic Metadata Internal Enrichments Engine (RSMIEE)
Rules-based Semantic Metadata Internal & Tags Enrichments Engine
Data Mining Engine (DME)
Machine Learning Engine (MLE)
Semantic Analytic Engine (SAE)
Enrichment's Analysis Engine (EAE)
Multilingual Normalization
Topics Analysis
Abstract of Abstract (AoA)
Sentiments/ Emotions Analysis
Learning-based Literature's Review Assistant
Citation's Review Assistant
Abstract of Abstracts Summarization
Learning-based Actuality's Review Assistant
Learning-based Course's Review Assistant
Electronic Metadata-based Discover Gateway (EMDG)
Software Product Line Engineering (SPLE)
Harvesting Authorities Metadata & Data (HAMD)
BAnQ (Canada)
SQTD BTLF
Biblioteca Nacional Library
National Library Canada
Library of Congres (USA)
JSTOR Open Library
Librarian Cataloguing
BNF (France)
Gallica (BNF)
German National Library
Europeana (Europe)
VIAF NAF SAF
British National Library
...
ISNI, DNB, Geoname ...
Legend

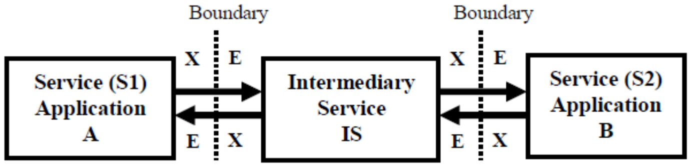
Brisebois, Abran, Nadembega, 'A Semantic Metadata Enrichment Software Ecosystem (SMESE) Based on a Multi-Platform Metadata Model for Digital Libraries', Journal of Software Engineering and Applications – JSEA, Vol. 10, April 2017, pp. 370-405.

Service Oriented Architecture Sizing with COSMIC

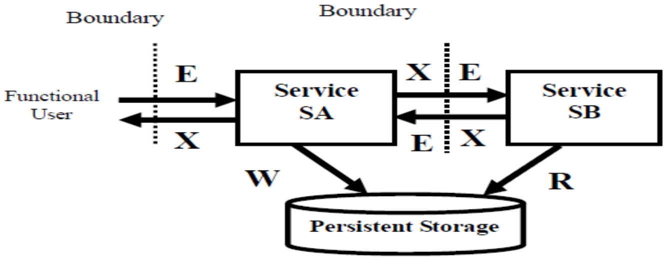
- COSMIC-SOA Exchange Messages

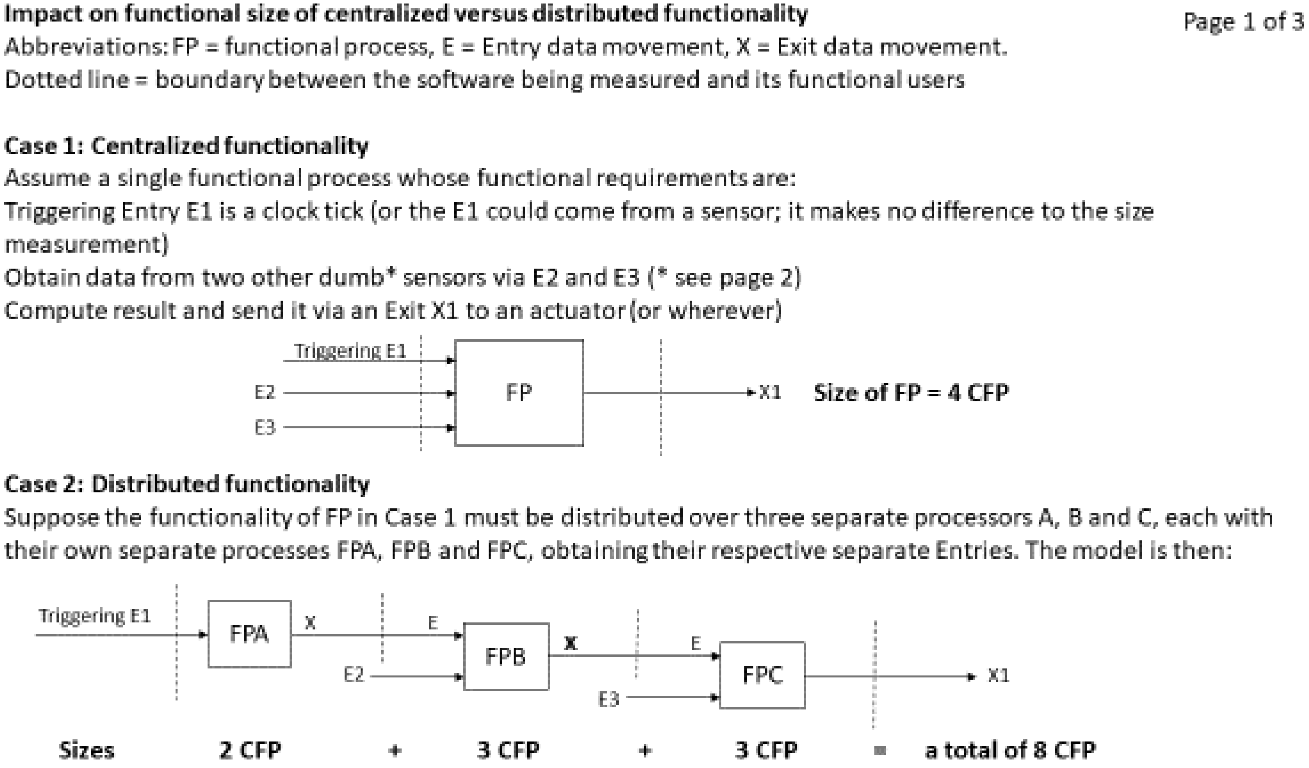


- COSMIC-SOA Intermediary Services



- COSMIC-SOA Exchange Data





Impact on functional size of centralized versus distributed functionality

Page 1 of 3

Abbreviations: FP = functional process, E = Entry data movement, X = Exit data movement.

Dotted line = boundary between the software being measured and its functional users

Case 1: Centralized functionality

Assume a single functional process whose functional requirements are:

Triggering Entry E1 is a clock tick (or the E1 could come from a sensor; it makes no difference to the size measurement)

Obtain data from two other dumb* sensors via E2 and E3 (* see page 2)

Compute result and send it via an Exit X1 to an actuator (or wherever)

Size of FP = 4 CFP

Case 2: Distributed functionality

Suppose the functionality of FP in Case 1 must be distributed over three separate processors A, B and C, each with their own separate processes FPA, FPB and FPC, obtaining their respective separate Entries. The model is then:

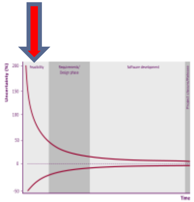
Sizes 2 CFP + 3 CFP + 3 CFP = a total of 8 CFP

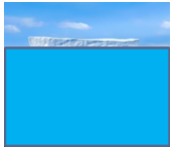
Source: Charles Symons – Sept. 2018

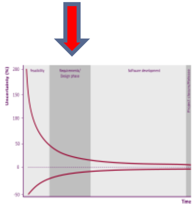
Conclusion

- A certain number of system requirements, which appear at first glance without measurable software functions, can be translated into software functional requirements and be measured with the COSMIC method.

COSMIC Guides through the lifecycle





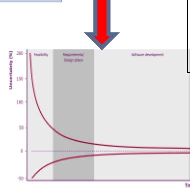


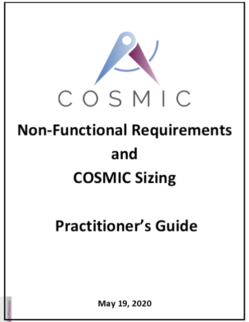
INTERNATIONAL STANDARD ISO/IEC/ IEEE 29148
First edition 2011-12-01

Systems and software engineering — Life cycle processes — Requirements engineering

Ingénierie des systèmes et du logiciel — Processus du cycle de vie — Ingénierie des exigences

ISO ISO-IEEE 25148



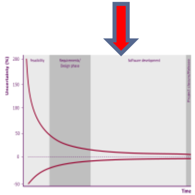


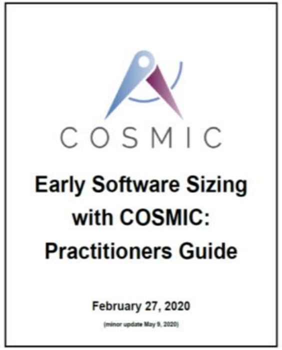
COSMIC

Non-Functional Requirements and COSMIC Sizing

Practitioner's Guide

May 19, 2020



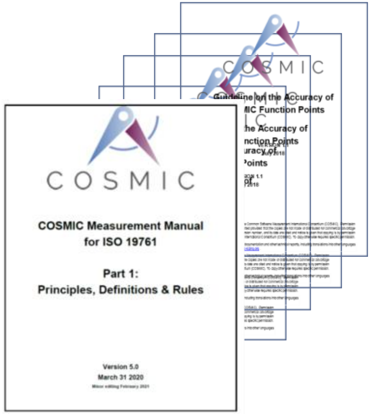


COSMIC

Early Software Sizing with COSMIC: Practitioners Guide

February 27, 2020

(minor update May 9, 2020)



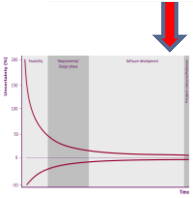
COSMIC

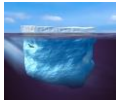
COSMIC Measurement Manual for ISO 19761

Part 1: Principles, Definitions & Rules

Version 5.0
March 31 2020
Minor editing February 2021

Guideline on the Accuracy of COSMIC Function Points





COSMIC Key Resources Available

- ✓ Sizing with accuracy : ISO rules in ISO 19761
- ✓ Early sizing techniques
- ✓ System non-functional req^{mts} (NFR) allocated to software functions
- ✓ Case studies
- ✓ Industry examples

Free www.cosmic-sizing.org

COSMIC

QUESTIONS?