

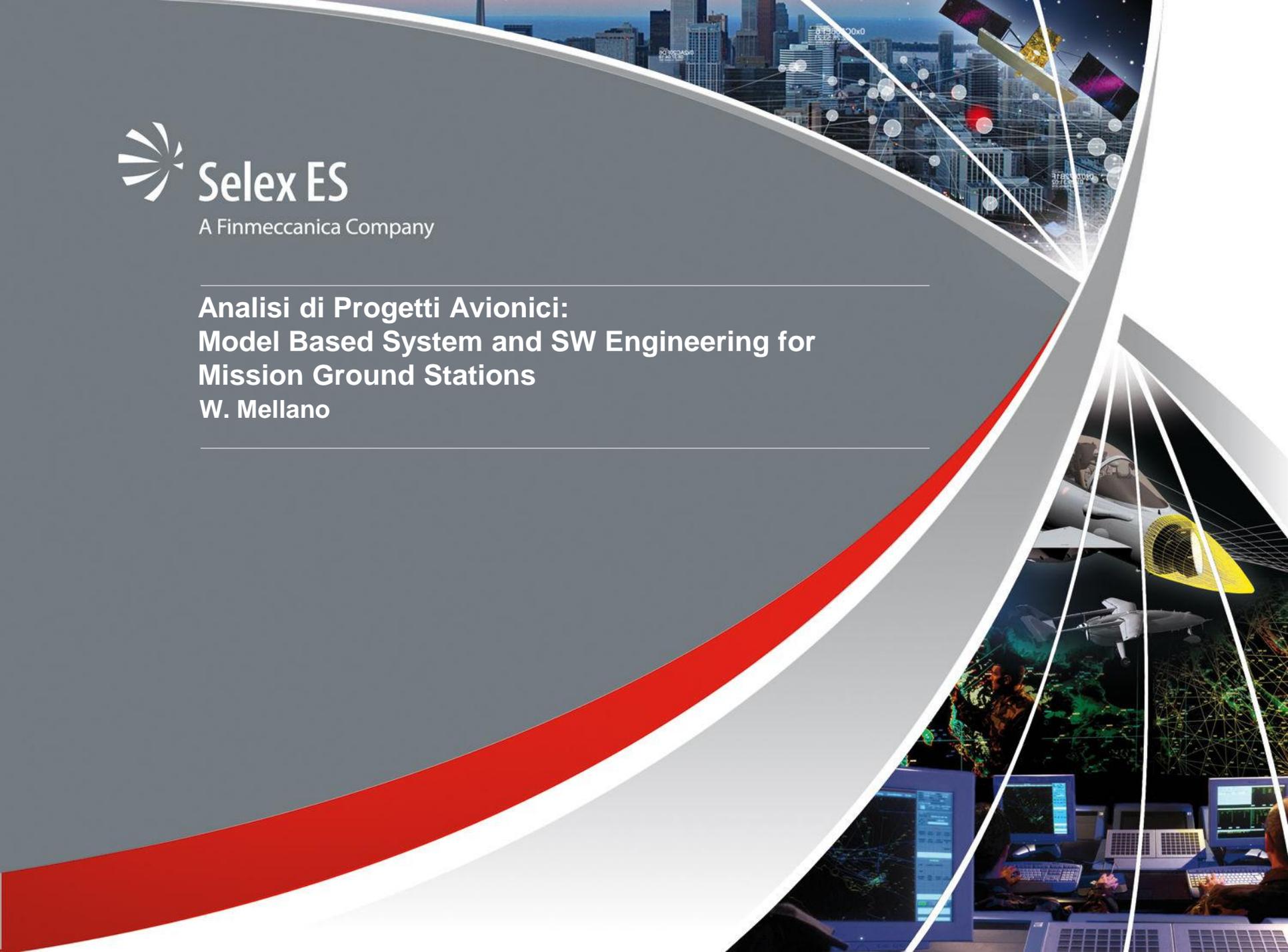


Selex ES

A Finmeccanica Company

**Analisi di Progetti Avionici:
Model Based System and SW Engineering for
Mission Ground Stations**

W. Mellano



Il contesto della presentazione

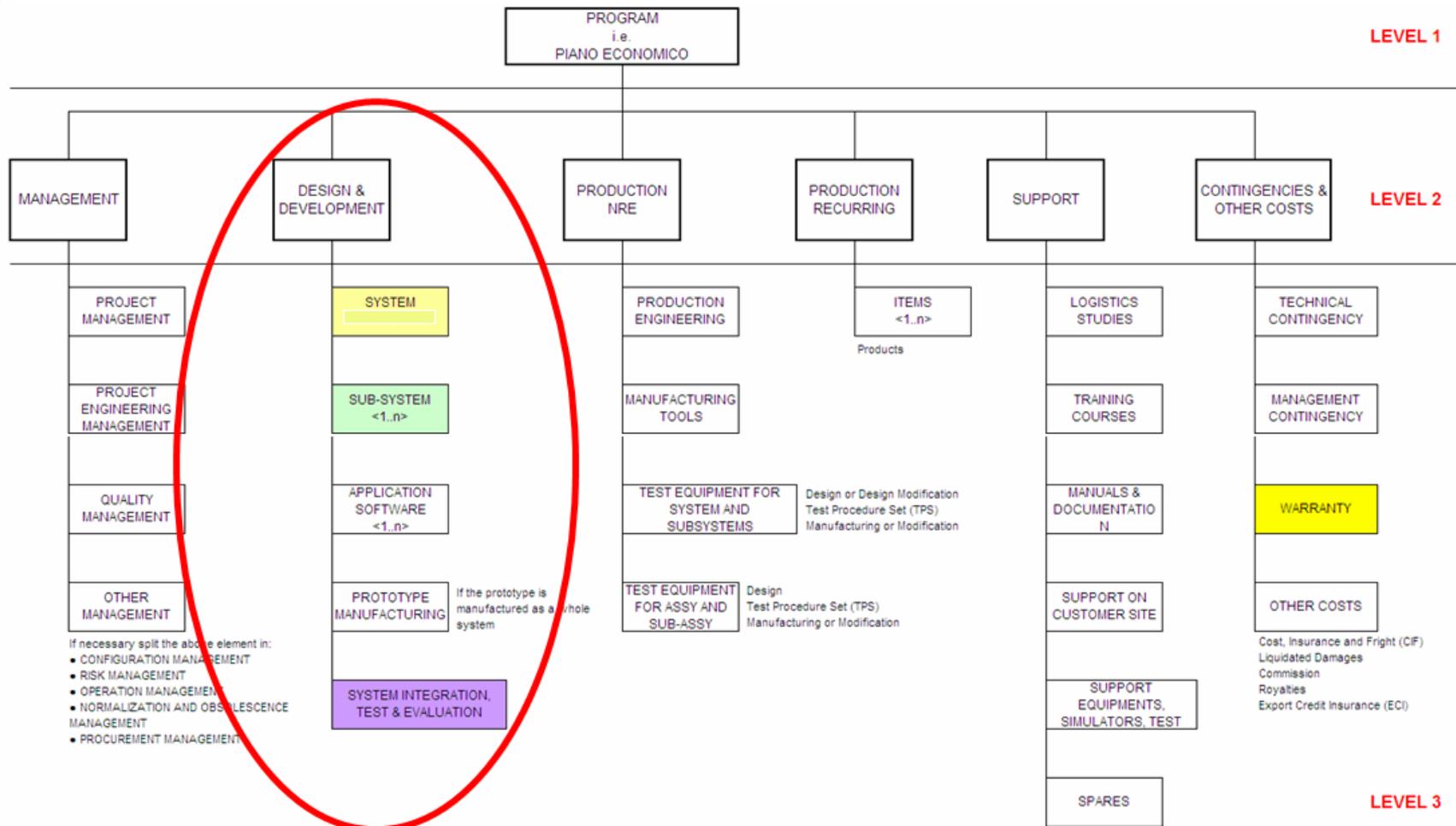
- Sviluppo di Sistemi Avionici
- Sistemi di bordo e Sistemi di Terra
- Sviluppo di sottosistemi di Terra
- Sottosistemi basati su COTS (Commercial Off-The-Shelf)
- Arco temporale degli ultimi 10 anni
- Centralità del Cliente
- Dall'offerta al Customer Training

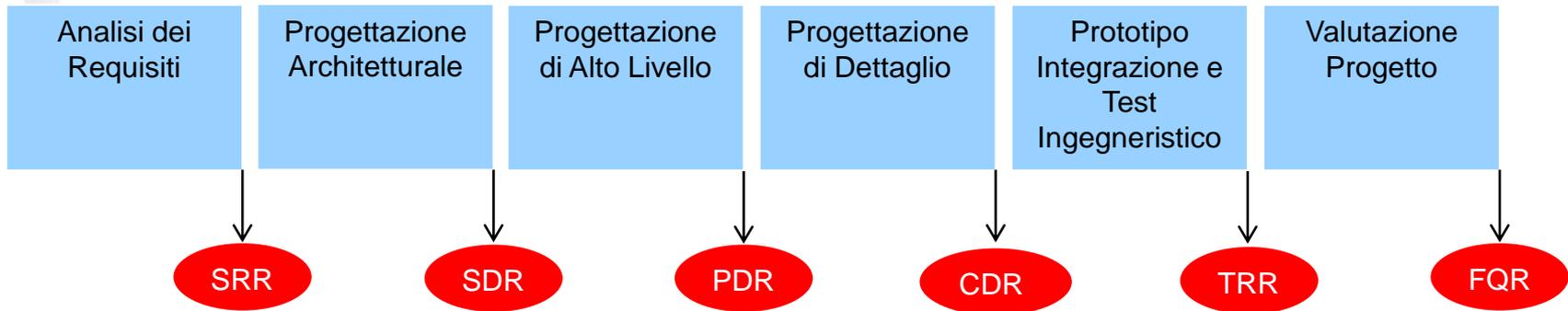
Work Breakdown Structure (WBS):

«individua l'organizzazione funzionale e gestionale dell'offerta /programma e rappresenta il prodotto di fornitura secondo una struttura gerarchica ad albero, dove vengono individuati e correlati tra loro i contributi elementari che lo compongono.»

(PRM-L2-018 - STANDARD WORK BREAKDOWN STRUCTURE [WBS] Selex ES property)

Organizzazione di Progetto: la WBS

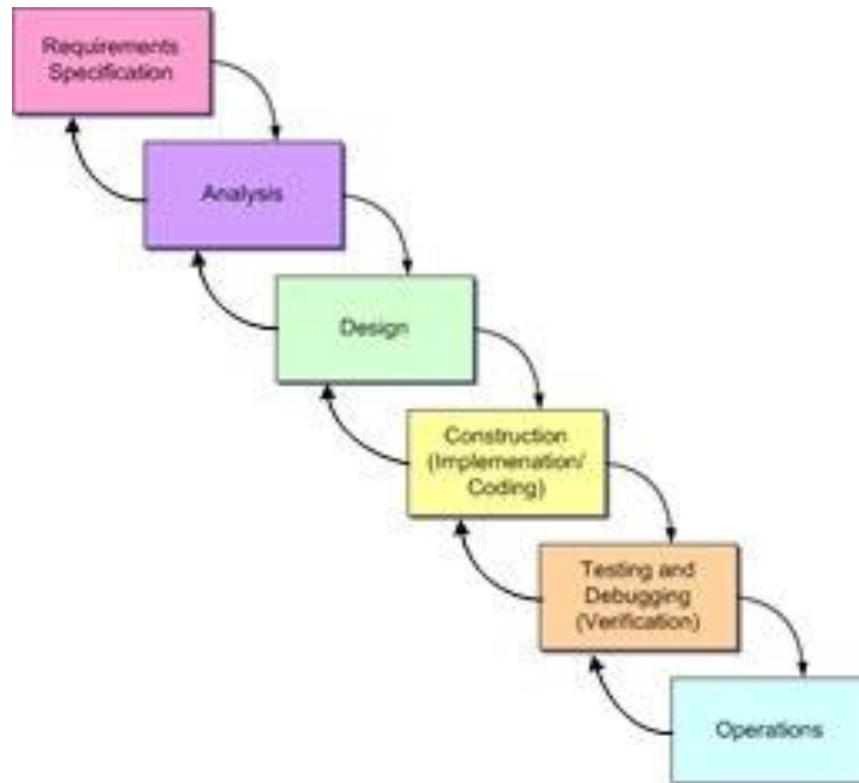


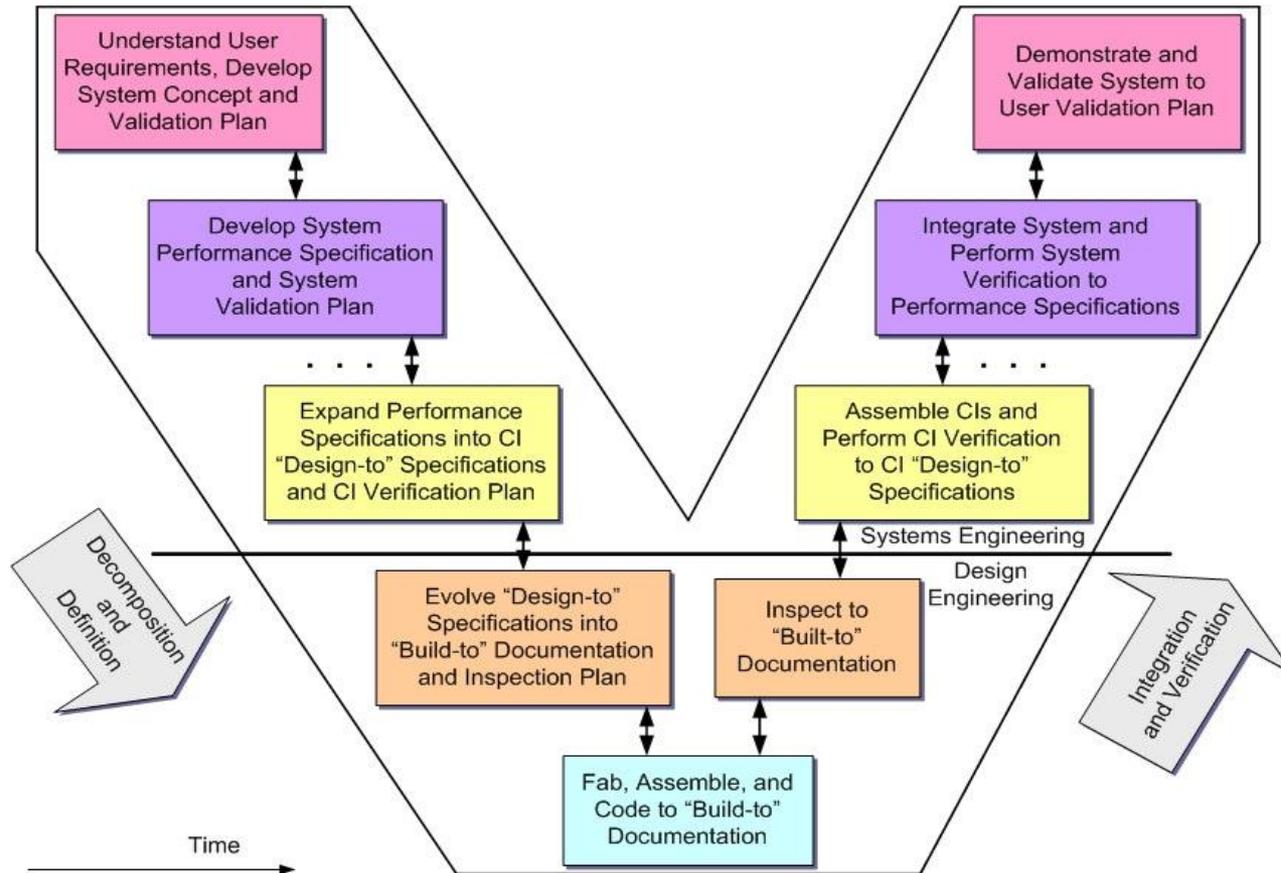


- System Readiness Review
- System Design Review
- Preliminary Design Review
- Critical Design Review
- Test Readiness Review
- Formal Qualification Review

Fonte: (PRM-L2-018 - STANDARD WORK BREAKDOWN STRUCTURE [WBS] Selex ES property)

Lifecycle Models: **Waterfall**

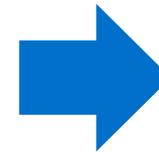
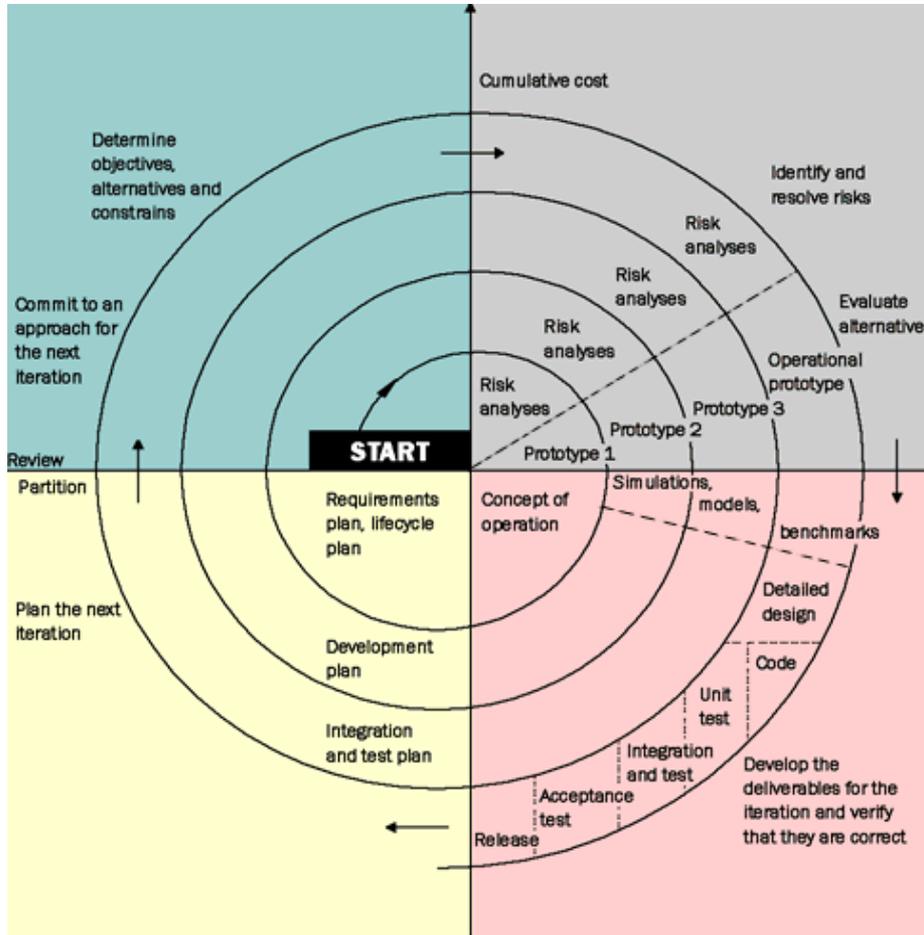




Selex-ES property - V. Mellano

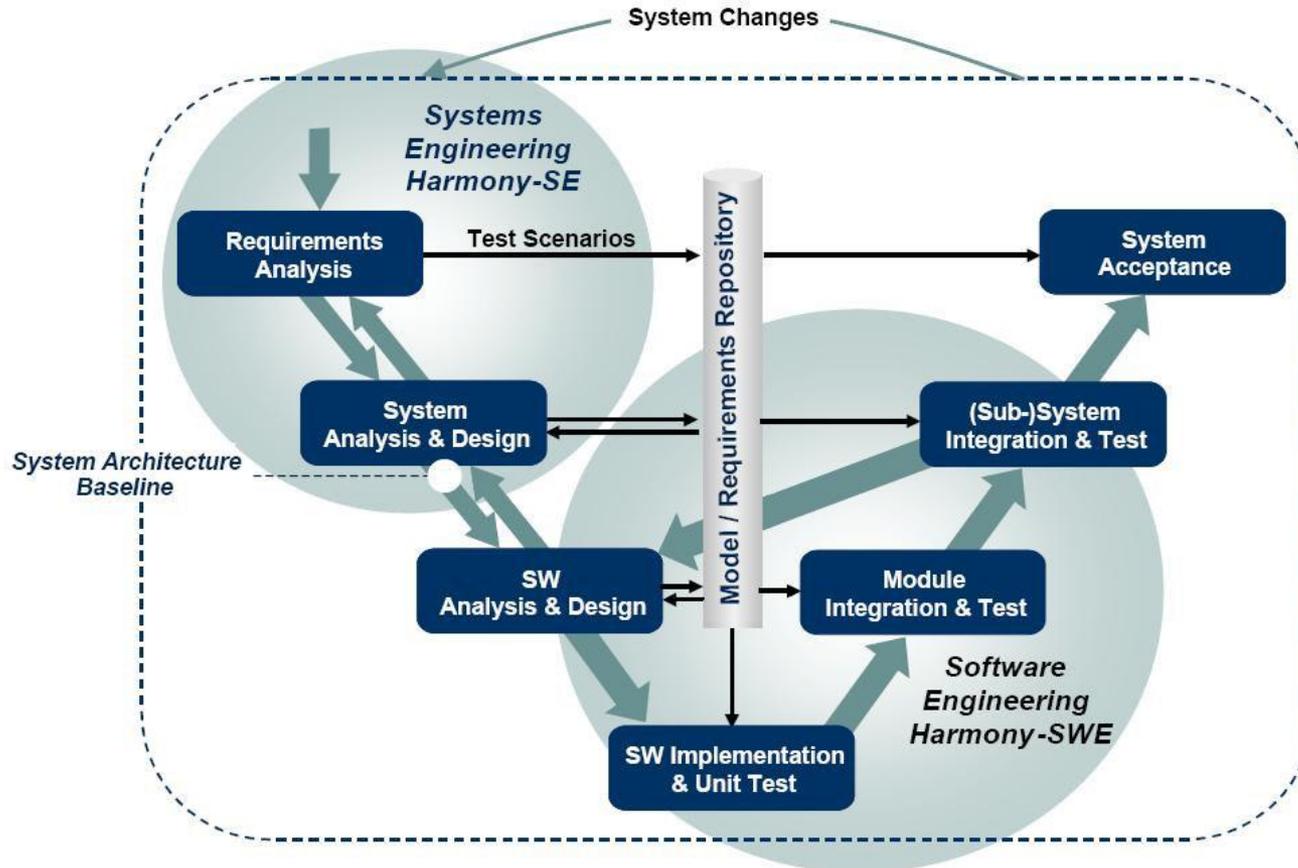
Fonte: Survey of Model-Based Systems Engineering (MBSE) Methodologies - Jeff A. Estefan - Jet Propulsion Laboratory California Institute of Technology - Pasadena, California, U.S.A.

Lifecycle Models: **Spiral**



PROGETTI A FORTE GRADO DI INNOVAZIONE

Lyfecycle Models: **Vee phases**



Selex-ES property - V. Mellano

Fonte: Survey of Model-Based Systems Engineering (MBSE) Methodologies - Jeff A. Estefan - Jet Propulsion Laboratory California Institute of Technology - Pasadena, California, U.S.A.

- **Project Documents:**

- PMP: Program Management Plan
- SDP: Software Development Plan

- **System Documents**

- SSS: System Subsystem Specification
- TS: Technical Specification
- SSDD: System Subsystem Design Document
- UIDD: User Interface Design Document
- DBDD: DataBase Design Document
- ICD: Interface Control Document
- FRS: Functional Requirements Specification
- STP: System Test Plan
- TP: System (sub-system) Test Procedure
- TR: Sistem (sub-system) Test Report
- FATP: Final Acceptance Test Procedure
- FATR: Final Acceptance Test Report
- CT: Ciclo di Trasformazione (not standard)

- **SW Documents:**
 - SRS: Sw Requirements Specification
 - IRS: Interface Requirements Specification
 - STP: SW Test Plan
 - SDD: Sw Design Document
 - SATP: Sw Acceptance Test Procedure
 - VDD: Version Description Document
 - SATR: Sw Acceptance Test Report
- SC: Source Code
- EX: Executable Code

- Dalle Specifiche di Sistema al Test di accettazione
- Come preservare l'informazione al fine dei test
- L'informazione si trasforma
- La scelta degli strumenti
- L'abitudine ai requisiti testuali
- L'automazione della tracciabilità
- Il Reporting



DOORS



**SysML,
UML**

- **System requirements: DOORS**
 - L'assoluto controllo del Requisito
 - L'assolutezza del requisito testuale
 - I documenti di Test
- **FRS, SRS, SDD -> DOORS & Rational Rose or other**
- Il "Gap" tra il requisito testuale e i metodi MBSE:
 - Le nostre soluzioni nel tempo
 - Rational Rose, Enterprise Architect, BOUML

Surveillance **I**nformation **M**anagement
(**SIM**) System, which is to be assembled to
meet the requirements of Project Sentinel
and Surveillance Australia Pty Ltd (SAPL).

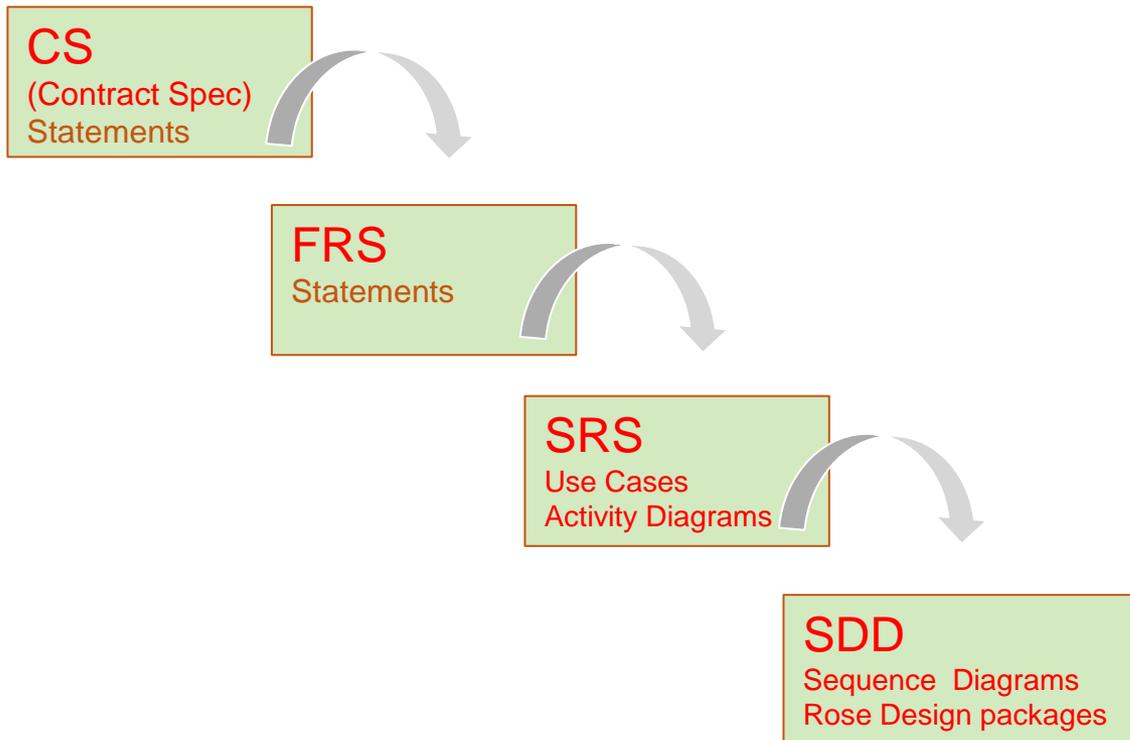


SIM-A: On board
manned Sub-System



SIM-G: Mission
Ground Station

Example from SIM-G Australia: **Flow**



Example from SIM-G Australia: **CS** and **FRS**

CS
(Contract Spec)
Statements



3.3.9.4 The SIM-A - SIM-G logical interface shall provide at least the following functions:

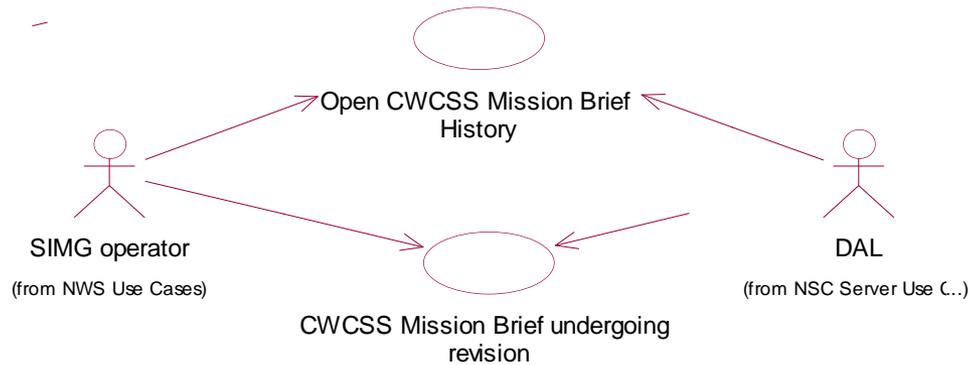
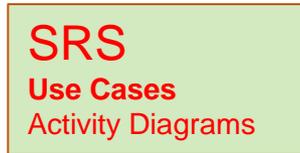
(a) Load SIM Mission Brief;

(b) ...

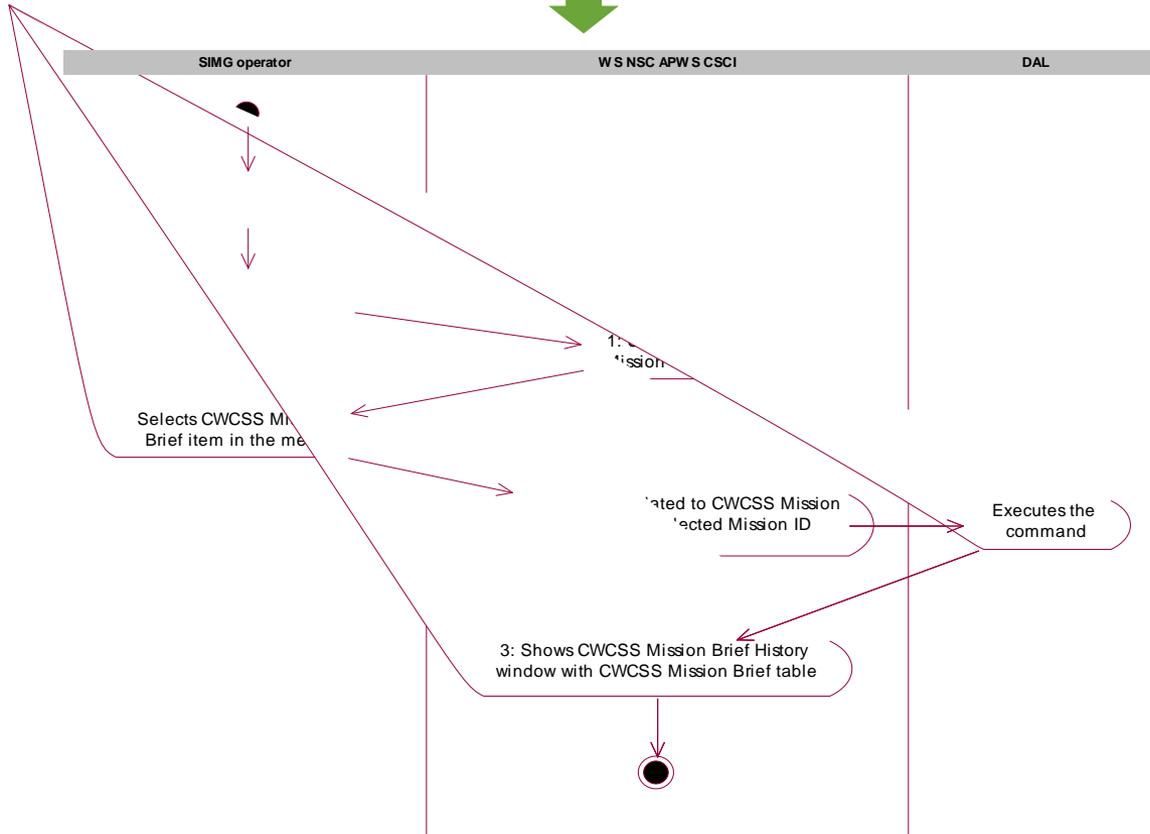
FRS
Statements



- *ID*: FRS SIM-G NSC-106
- The WS NSC shall provide a function to display Mission Brief history information



SRS
Use Cases
Activity Diagram



CSCI Components

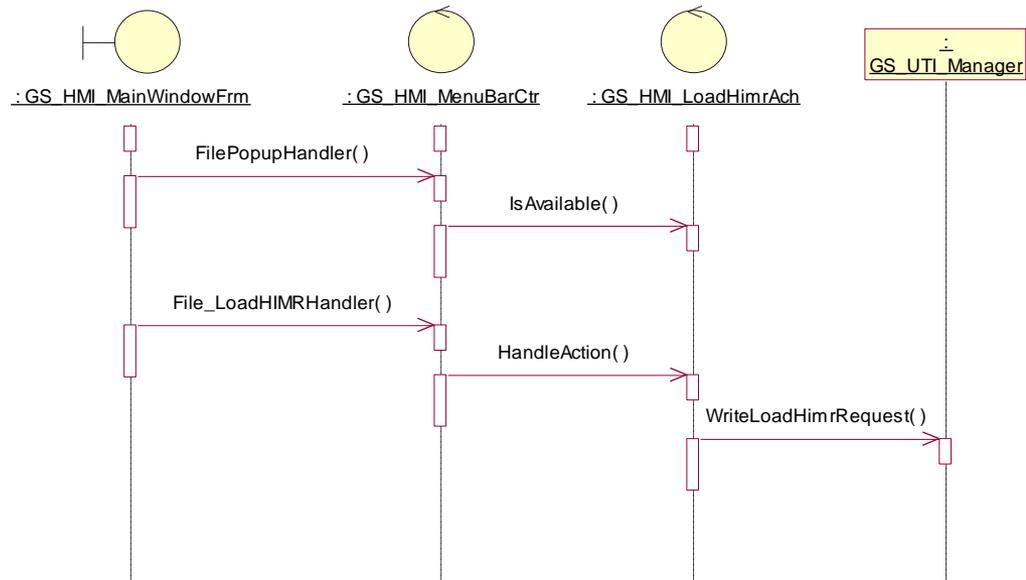
The main classes composing the CSCI are organized in these layers:

- ***Workstation Application layer***

This layer contains the software units that manage the application. The Workstation Application layer contains the following software units:

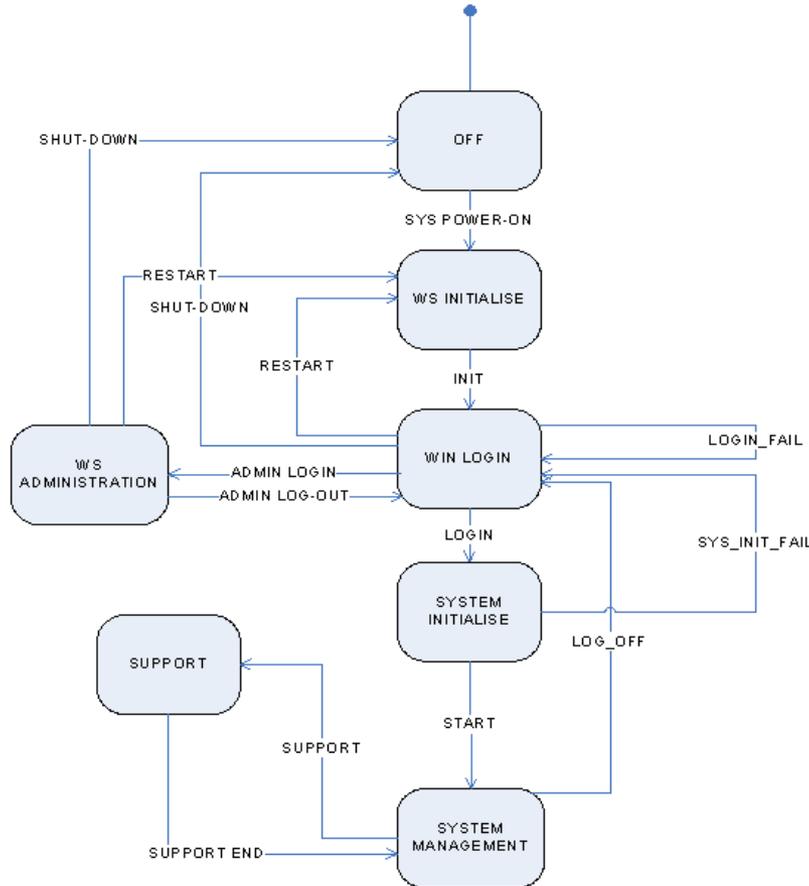
- GS_APP software unit: this SU provides the functionality to perform the main application of the system.
- GS_IMG software unit: this SU provides the base services to manage and manipulate the IMAGES: resizing and cropping.
- GS_MMS software unit: this SU provides the base services to manage and manipulate the audio/video files/sources: resize and cropping.

Concept of Execution: Menu bar Handling



Example from SIM-G Australia : the State Machine

Workstation State Machine



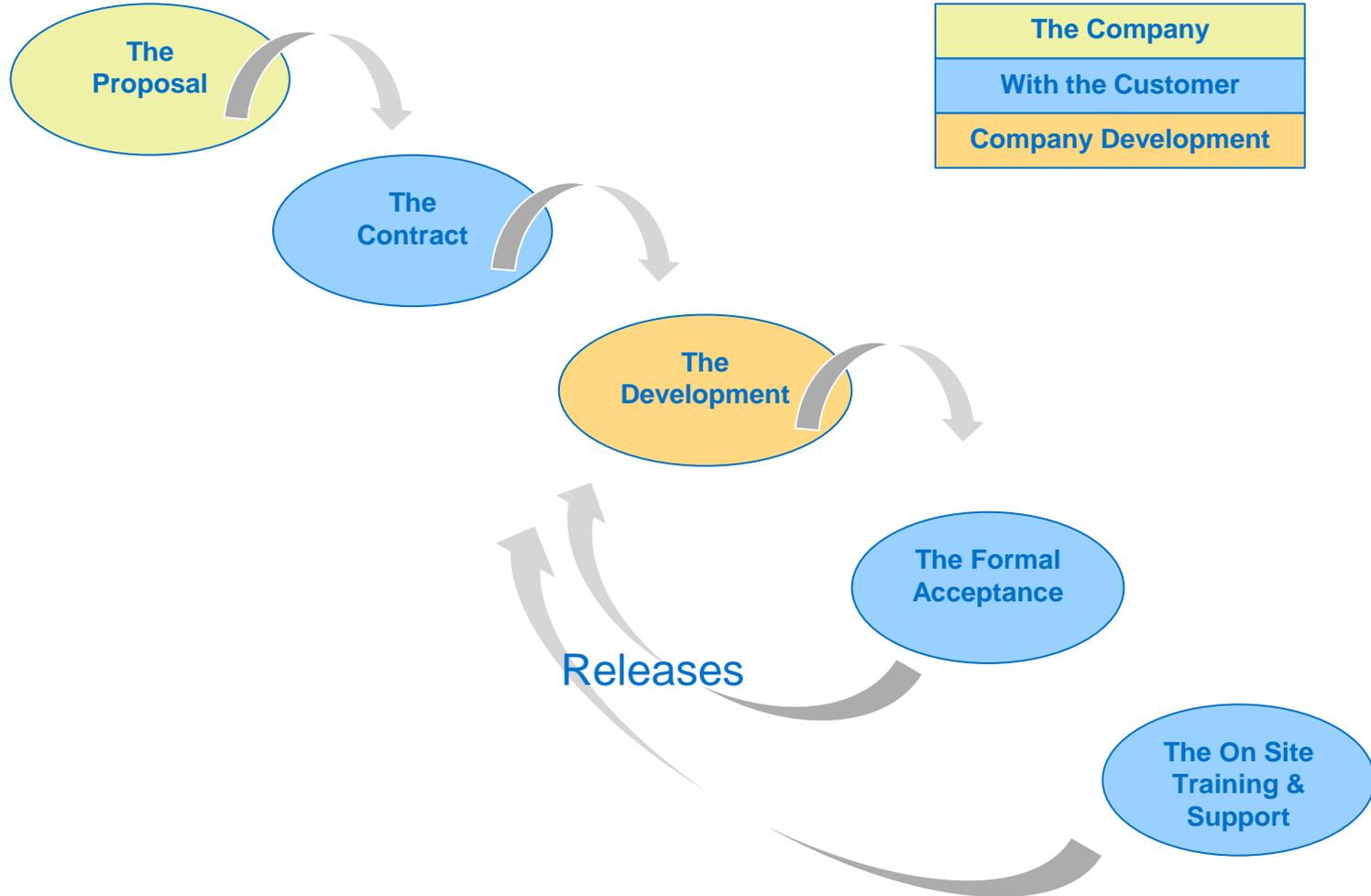
Una Macchina a Stati produce dati in uscita in funzione degli ingressi transitando attraverso una serie finita di stati in cui, per ogni stato, è definita una funzione di transizione deterministica verso lo stato successivo

The State Machine

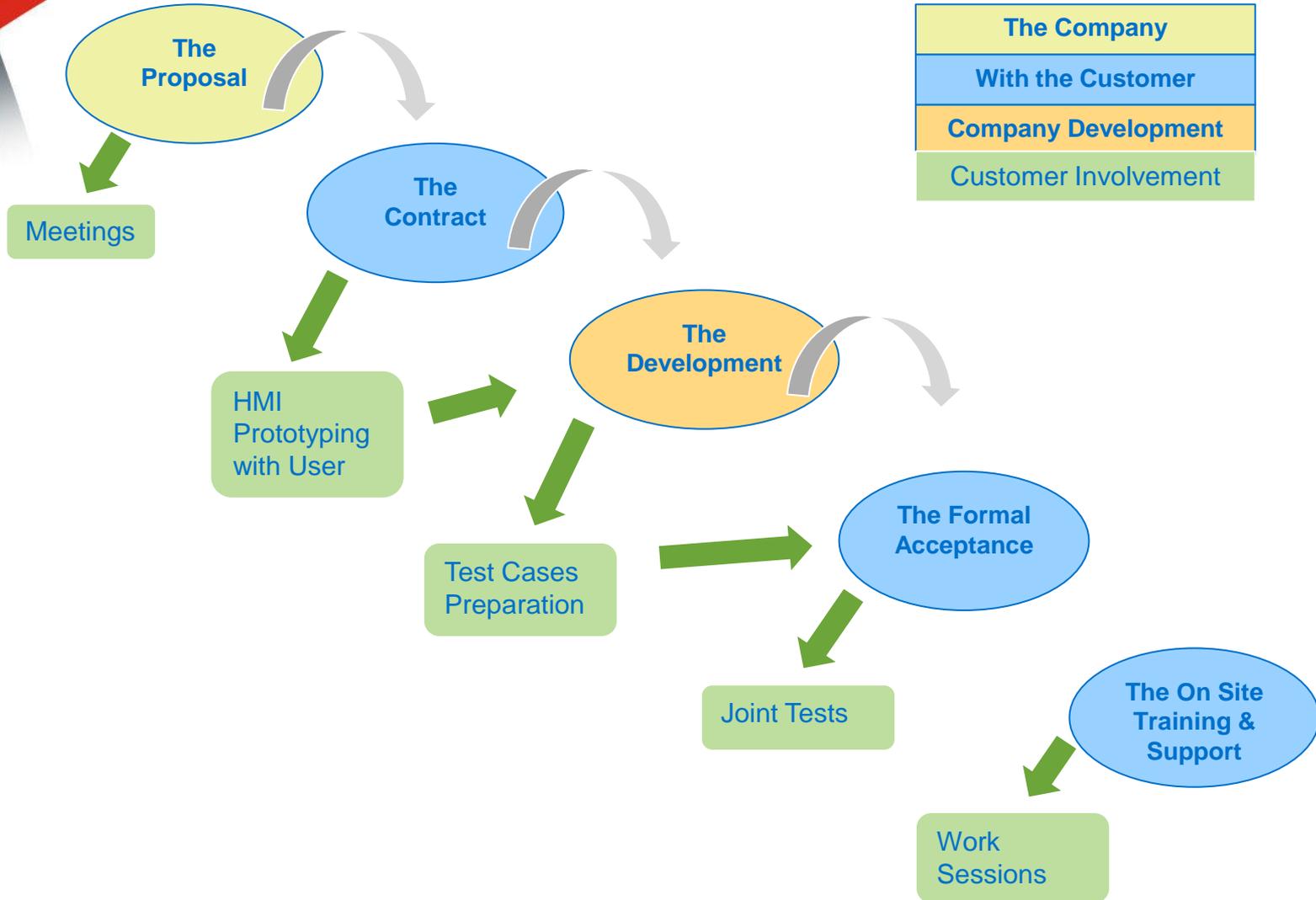
La Macchina a Stati riveste fondamentale importanza nel System design perché:

- Una Macchina a Stati globale permette di definire con precisione il funzionamento dell'HW.
- La scomposizione in Macchine a Stati di sottosistema chiarisce e delimita il funzionamento di ogni parte del sistema.
- La definizione testuale degli stati pone le basi per l'allocazione dei Requisiti.
- La Macchina a Stati è il riferimento essenziale nella definizione dei test a livello di sistema e sotto-sistema: una volta definiti i test che verificano il corretto funzionamento degli stati, possono essere eseguiti i test funzionali specifici all'interno dei diversi stati di funzionamento

The Project: Customer Involvement



The Project: Customer Involvement



The Customer Involvement: Effetti sul MBSE

HMI Prototyping with User



The
Contract

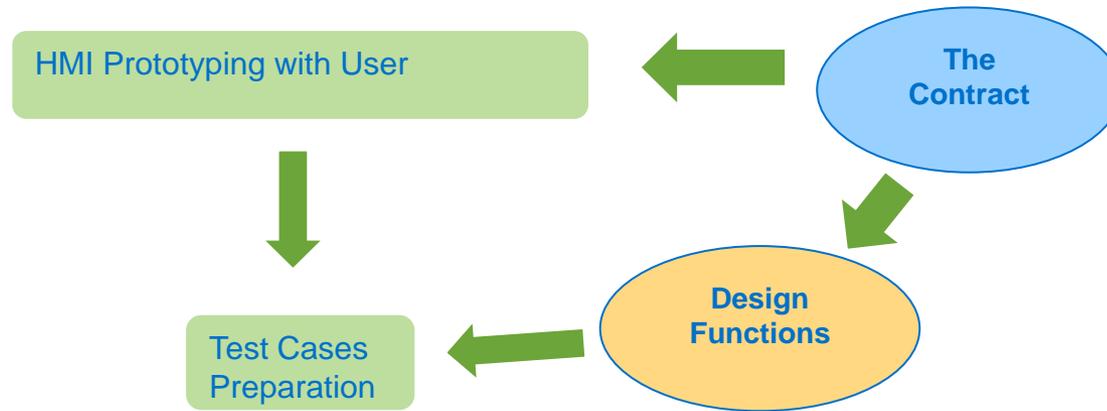
- **L'utilizzo di SysML**

- Use Cases e scomposizione funzionale

- **L'interfaccia utente**

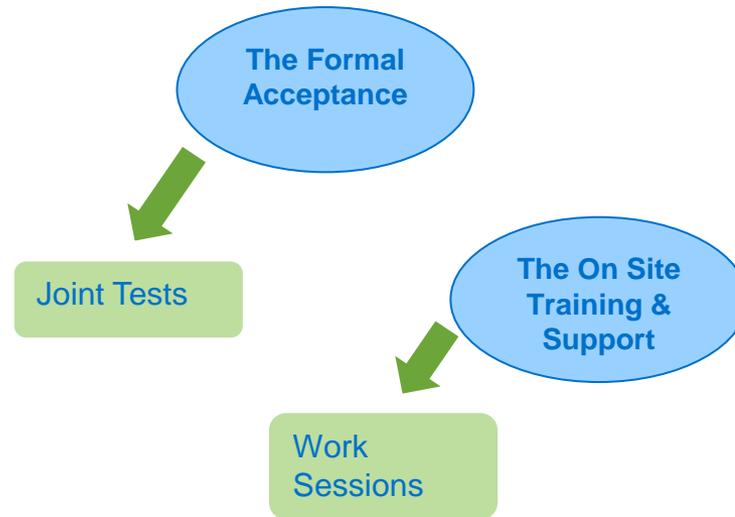
- Il Prototyping presso il Cliente
- Una attività separata con un proprio documento: User Interface Design Document (UIDD) allegato al contratto

The Customer Involvement: Effetti sul MBSE



- I Test Cases vengono preparati:
 - partendo dai System Use Cases
 - tenendo conto dell'HMI Prototyping
 - in parallelo con la definizione del sistema
- E' importante coinvolgere un Customer Representative
- Definire correttamente compiti e LIMITAZIONI del Customer Representative

The Customer Involvement: **Test & Release**



- La Formal Acceptance interna è eseguita con il Cliente e soddisfa il Contratto
- Viene rilasciata la prima versione



The Customer Involvement: **Test & Release**

- **Gli utenti finali possono essere diversi da quelli consultati nel prototyping**
- **Hanno un'idea abbastanza vaga di un “nuovo sistema”**
- **Si decide di inviare una persona a fare del training PRIMA della release**

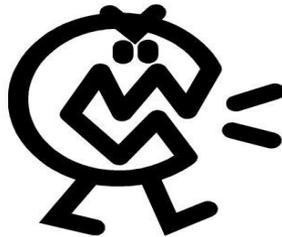


- **Una noia mortale**



The Customer Involvement: **Test & Release**

- Il sistema viene installato
- Gli utenti perdono tempo per via di alcuni problemi che vengono risolti in fretta
- Iniziano ad utilizzare il sistema seguendo (forse) le istruzioni del corso
- Sono abituati al sistema precedente, lo utilizzano comunque in tal modo
- Impegnano più del doppio del tempo ad eseguire le operazioni di routine
- Non si rendono conto dell'esistenza di una nuova filosofia di interazione
- Non riescono neanche a provare le funzioni più sofisticate
- Vengono raccolte le loro opinioni:



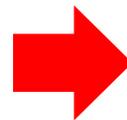
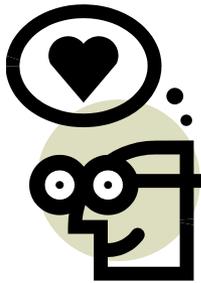
***Il Sistema è stato fatto
secondo il Contratto ma
non SERVE A NIENTE***



NO CUSTOMER SATISFACTION

The Customer Involvement: **Test & Release**

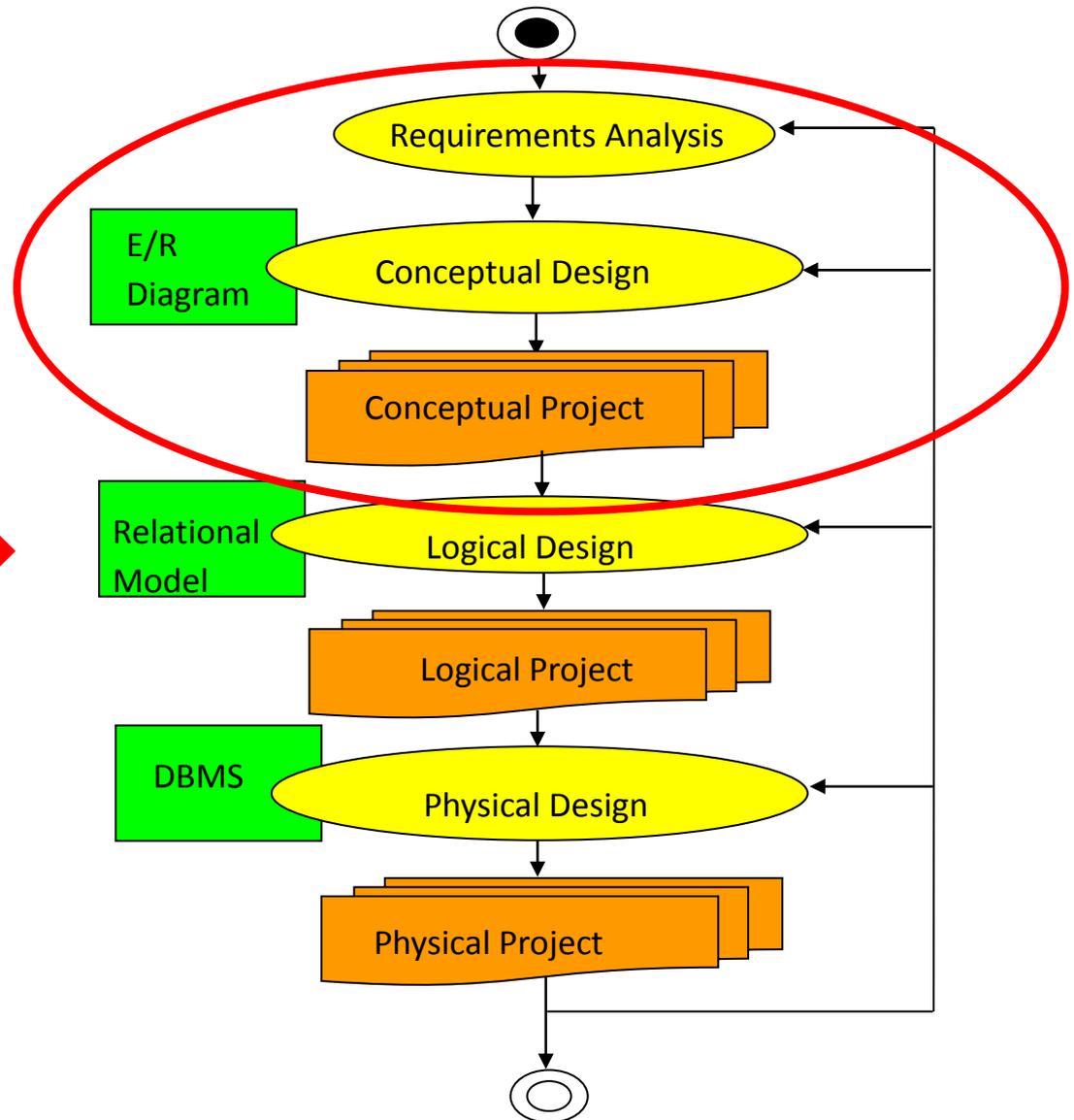
- Il sistema viene installato
- Viene inviato un sistemista esperto in loco
- Il sistemista prova a lungo il sistema e invia dei Problem Report
- Il sistemista esegue una BREVE presentazione delle funzionalità del sistema
- Si siede vicino agli utenti e osserva
- Ogni qualvolta l'utente è disorientato lo aiuta spiegandogli la filosofia e le procedure
- In caso di malfunzionamenti invia Problem Report
- Osserva egli stesso il funzionamento del sistema e tiene traccia di possibili miglioramenti:



***Il Sistema e stato fatto
secondo il Contratto e
SERVE***



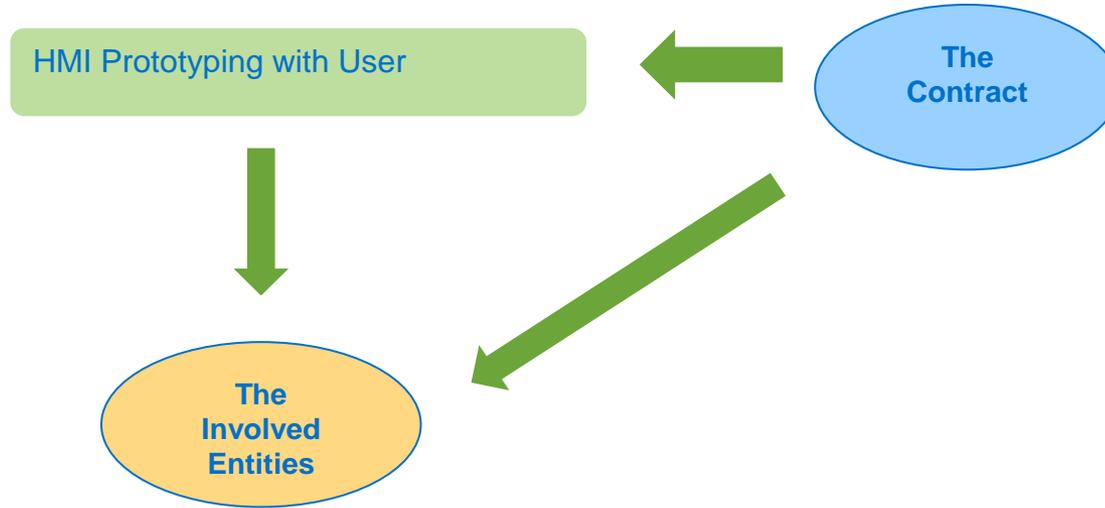
CUSTOMER SATISFACTION



The Database Design



II System Design: **Il circolo virtuoso**



**«...stat rosa pristina nomine, nomina nuda
tenemus»** (Bernardo, dal Nome Della Rosa)

- **DATA MODEL**

- a description of the objects represented by a computer system together with their properties and relationships

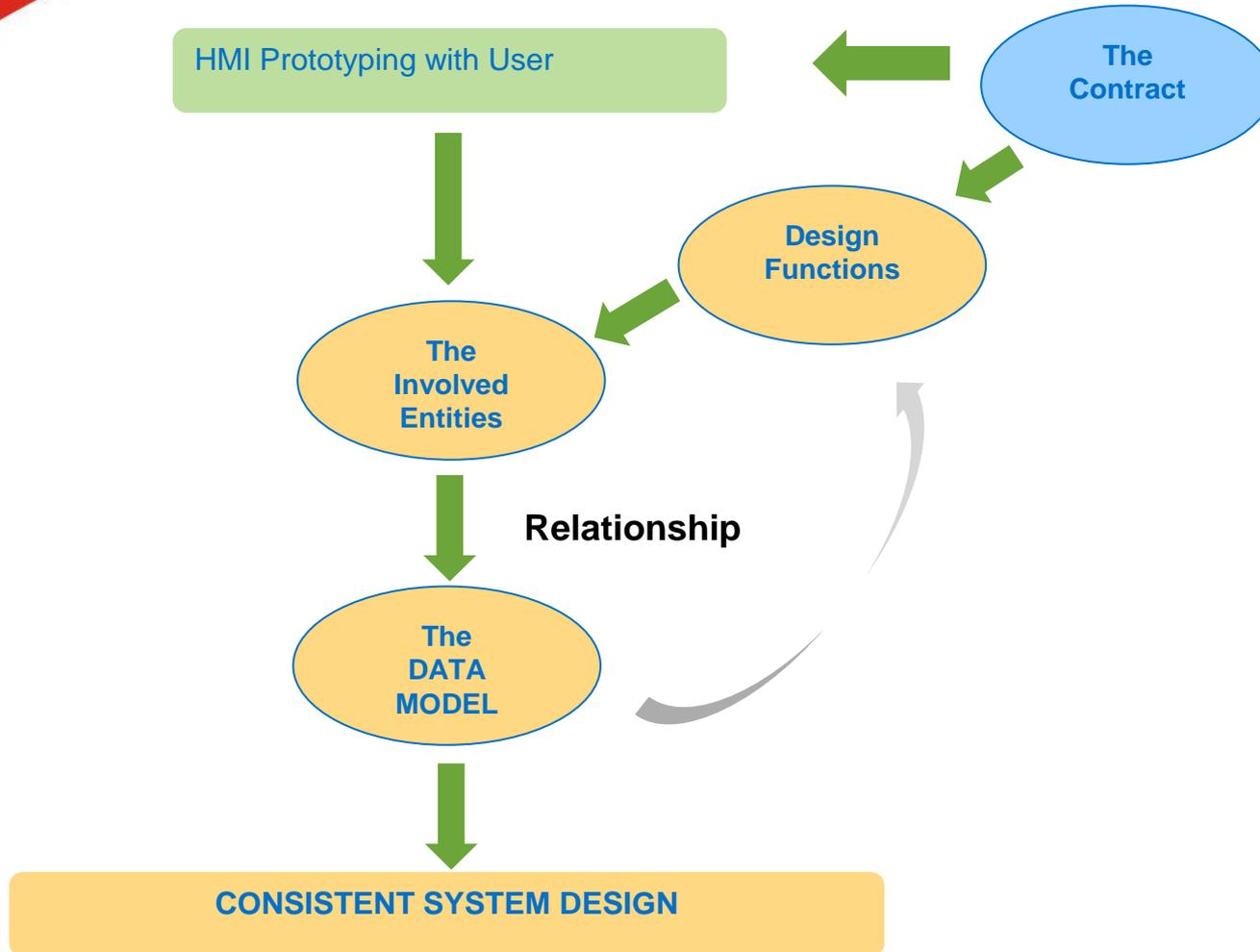
- **DATABASE MODEL**

- a collection of concepts and rules used in defining data models

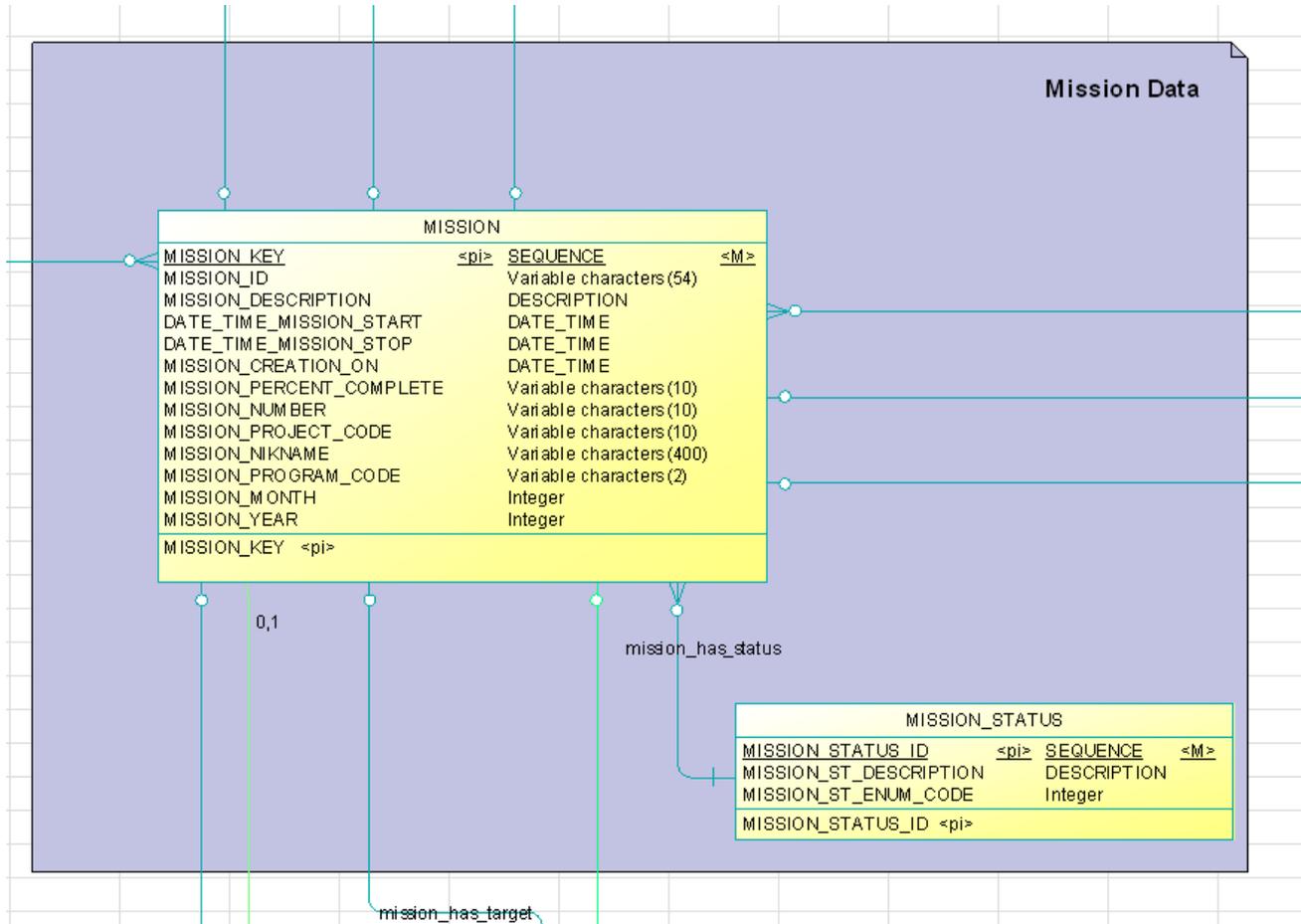
- **RELATIONAL MODEL**

- A Database Model that uses relations and tuples

II System Design: **Entities**



II System design: Relationship



Il Design Sw: Il punto critico delle Infrastrutture

- **Vincoli nella scelta:**

- Riutilizzo di SW pre-esistente
- Reperibilità delle competenze per un corretto design e sviluppo

- **I criteri:**

- Indipendenza dall'HW e dai SS. OO.
- Modularità
- Ridondanza
- Utilizzo in Rete

- **Le scelte nel tempo**

- Dalle API alla SOA

Risk Mitigation: **Prototyping del sistema**

- La scelta accurata del modello di infrastruttura
- Non serve a niente se non si è fatto prima del prototyping delle infrastrutture



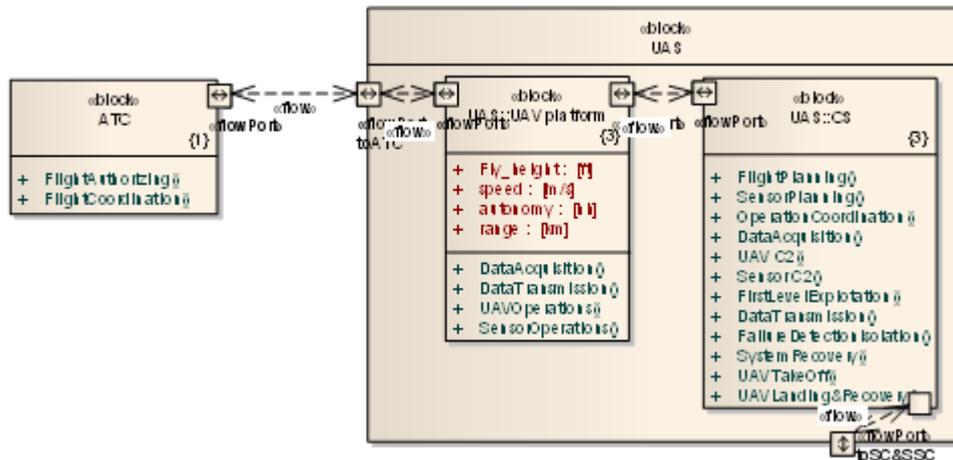
- Eseguire un prototipo che dimostri l'utilizzo delle infrastrutture:
 - Selezionare alcune funzioni significative
 - Ridurne al minimo il contenuto non necessario al prototyping
 - Implementarle
 - Eseguire dei benchmark
 - Modificare, se necessario, le infrastrutture

- Il WEB oriented e i sistemi distribuiti
- La SOA (Service Oriented Architecture)
- DDS
- RPCL
- Il Publish/Subscribe
- I Multi-Agent systems

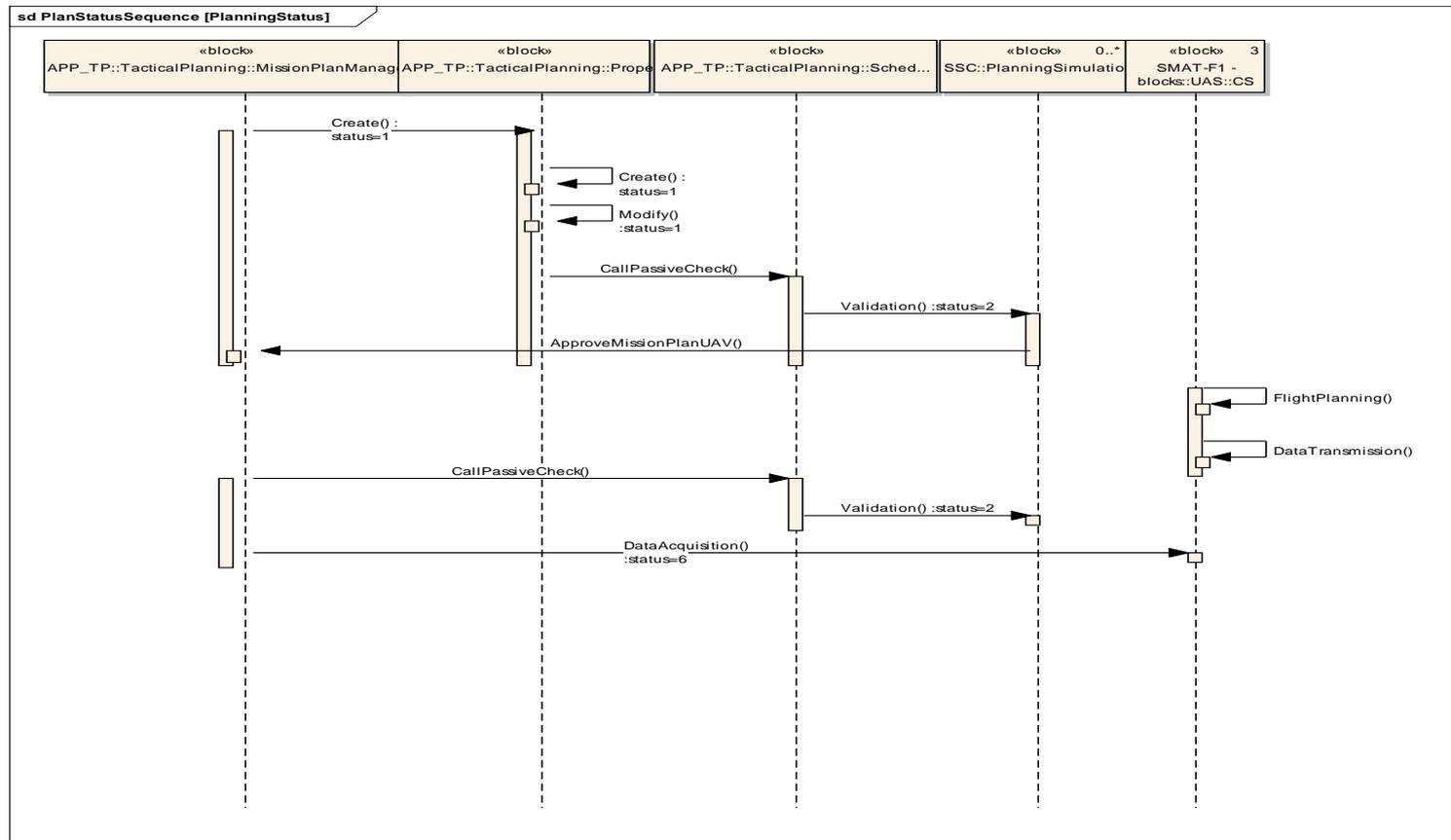
- La SOA è un Design Model che incapsula la logica di diverse applicazioni in servizi interagenti tramite un protocollo di comunicazione comune
- I protocolli di comunicazione sono in grado di smistare le richieste di servizi e la loro soddisfazione in modo articolato, che possono prevedere la negoziazione e la redirectione dei servizi

La progettazione di un SW secondo la SOA prevede:

- La scomposizione in blocchi funzionali, con una descrizione in termini di applicazioni indipendenti inglobanti servizi
- A livello inferiore I blocchi sono descritti in termini di classi “wrapper” infrastrutturali e classi inerenti al dominio

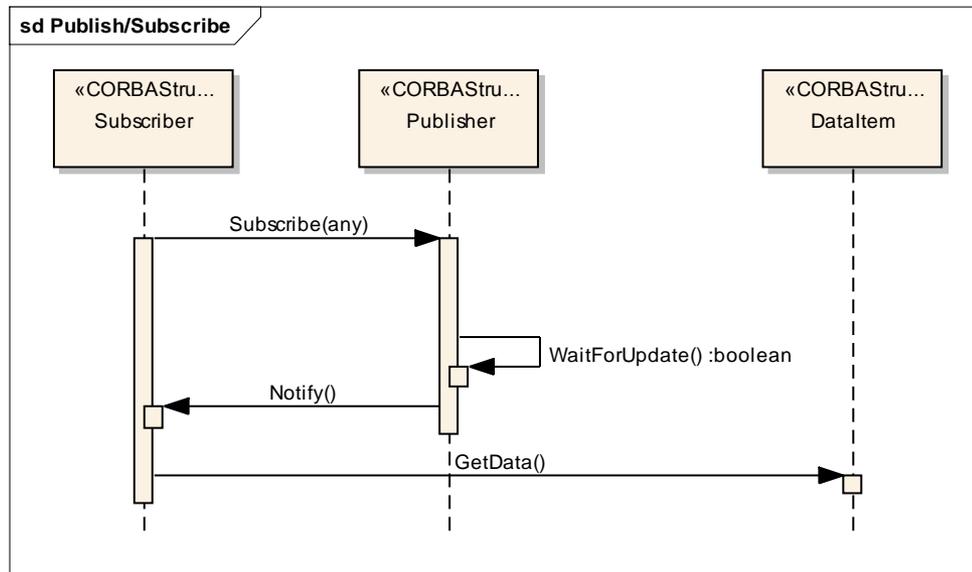


- L'utilizzo di Sequence Diagram per comprendere l'interazione della richiesta di servizi e loro negoziazione



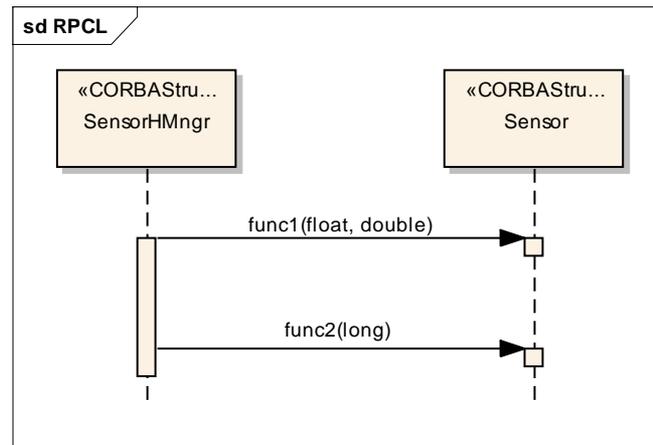
- DDS: Data Distribution Service:
- Il DDS è un *middleware* per sistemi incentrati sui dati (dall'anglosassone *data-centric systems*), ovvero sistemi distribuiti il cui funzionamento è basato sullo scambio di dati in tempo reale da più sorgenti a più destinazioni. Esempi tipici sono i sistemi di controllo, i sistemi di difesa, i sistemi di trading finanziario, ecc.

Publish/Subscribe sequence diagram:



- RPCL: Remote Procedure Call Language
- RPC:

“is an inter-process communication that allows a computer program to cause a subroutine or procedure to execute in another address space (commonly on another computer on a shared network) without the programmer explicitly coding the details for this remote interaction.” (Wikipedia)

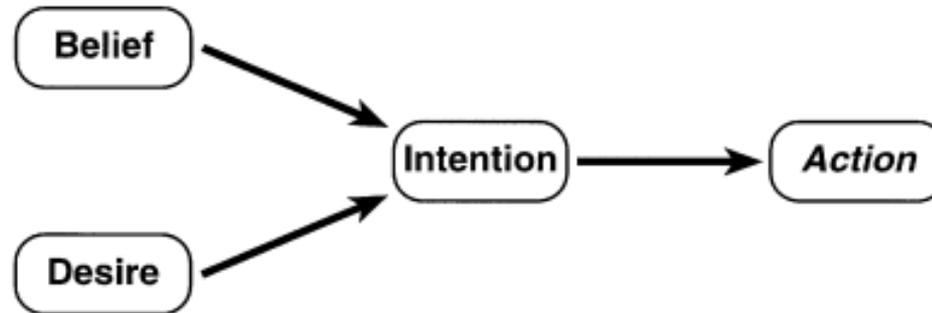


Esempi: Multi Agent Systems

- A software agent is a component within a system that, starting from an homogeneous set of incoming data, performs autonomous control of resources
- A multi-agent system is one that consists of a number of agents that interacting each other in a cooperative way

In one of the description of agents architecture (**BDI**) an agent:

- Receives data and builds **Beliefs** of the situation
- Generates different options to manage the situation (**Desires**)
- Filters the Desires to build a list of **Intentions**
- Transforms the Intentions into **Actions** when possible



Si possono utilizzare diversi tipi di diagrammi per descrivere i diversi compiti di un Agente:

- Class Diagrams a blocchi funzionali per descrivere l'insieme degli Agenti
- Class Diagrams a blocchi funzionali per descrivere le funzioni di alto livello di un Agente e i flussi dati tra le sue componenti interne
- Macchina a stati del funzionamento dell'Agente
- Activity diagrams per descrivere le procedure di selezione e valutazione dei Beliefs e delle Intentions e la pianificazione delle azioni

Quando occorre sviluppare un SW che non necessita del flusso documentativo completo, si possono ridurre i documenti e prodotti da rilasciare a:

SoftWare Description Document
Ciclo di Trasformazione del SW
Software Acceptance Report
Version Description Document
Source Code
Executable Code

SWDD: SoftWare Description Document

- Scopo
- Identificazione
- Descrizione del sistema
- Descrizione del documento
- Requisiti SW (Allegati da UML)
- Design architetturale (Allegati da UML)
 - Scelte architettureali
 - Concepts of Execution
- Design di dettaglio: SW Components (Allegati da UML)
- Interface Design (Allegati da UML)
- Coding Guidelines
- Problem Reports Management
- Code releases
- Testing
 - SW Integration Environment
 - SW Test Procedure

- SW di test
- SW di emulazione o simulazione
- SW per il training
- Prototipi per l'innovazione

...**ma in fondo**

È System Engineering...

... o System Handcrafting ?

...ma in fondo

ASSOLUTAMENTE SYSTEM ENGINEERING:

- Elevata classe di rischio
- Project Team eterogeneo e geograficamente distribuito
- Prodotto consolidato

SYSTEM HANDCRAFTING:

- Classe di rischio moderata
- Prodotto nuovo con innovazione
- Nuovi sistemi prototipali

QUINDI....

- Prima di tutto viene il Cliente
 - -> puoi avere risparmiato sui costi, ma se il Cliente non è soddisfatto il prodotto non vende
- La seconda priorità è la Qualità
 - -> per essere all'altezza della concorrenza
- La terza priorità è l'ottimizzazione dei costi
 - -> per battere la concorrenza
- La quarta priorità è l'innovazione
 - -> il prossimo Cliente vorrà un prodotto migliore

CONCLUSIONI

- Prima di tutto viene il Cliente
 - ->MBSE trasforma i suoi desiderata in componenti testate
- La seconda priorità è la Qualità
 - -> MBSE consente il controllo completo sul prodotto e sul testing partendo dagli Use Cases
- La terza priorità è l'ottimizzazione dei costi
 - -> MBSE permette di pianificare il budget correttamente e riutilizzare il SW, nonché di monetizzare ulteriori richieste del cliente
- La quarta priorità è l'innovazione
 - -> MBSE favorisce la progettazione e l'inserimento modulare di nuove funzioni