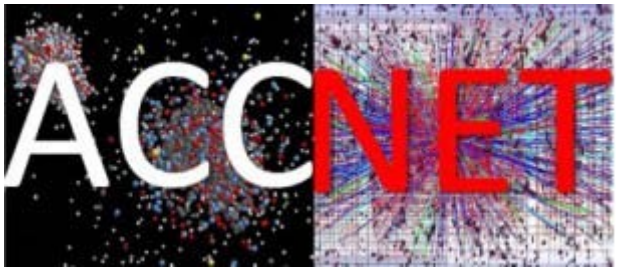
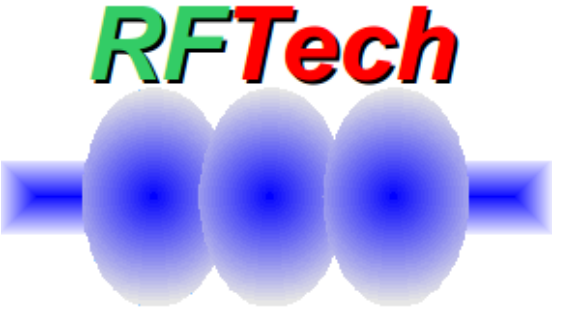
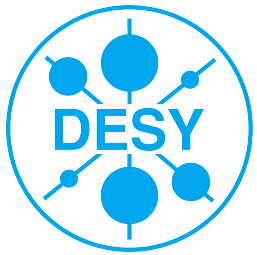




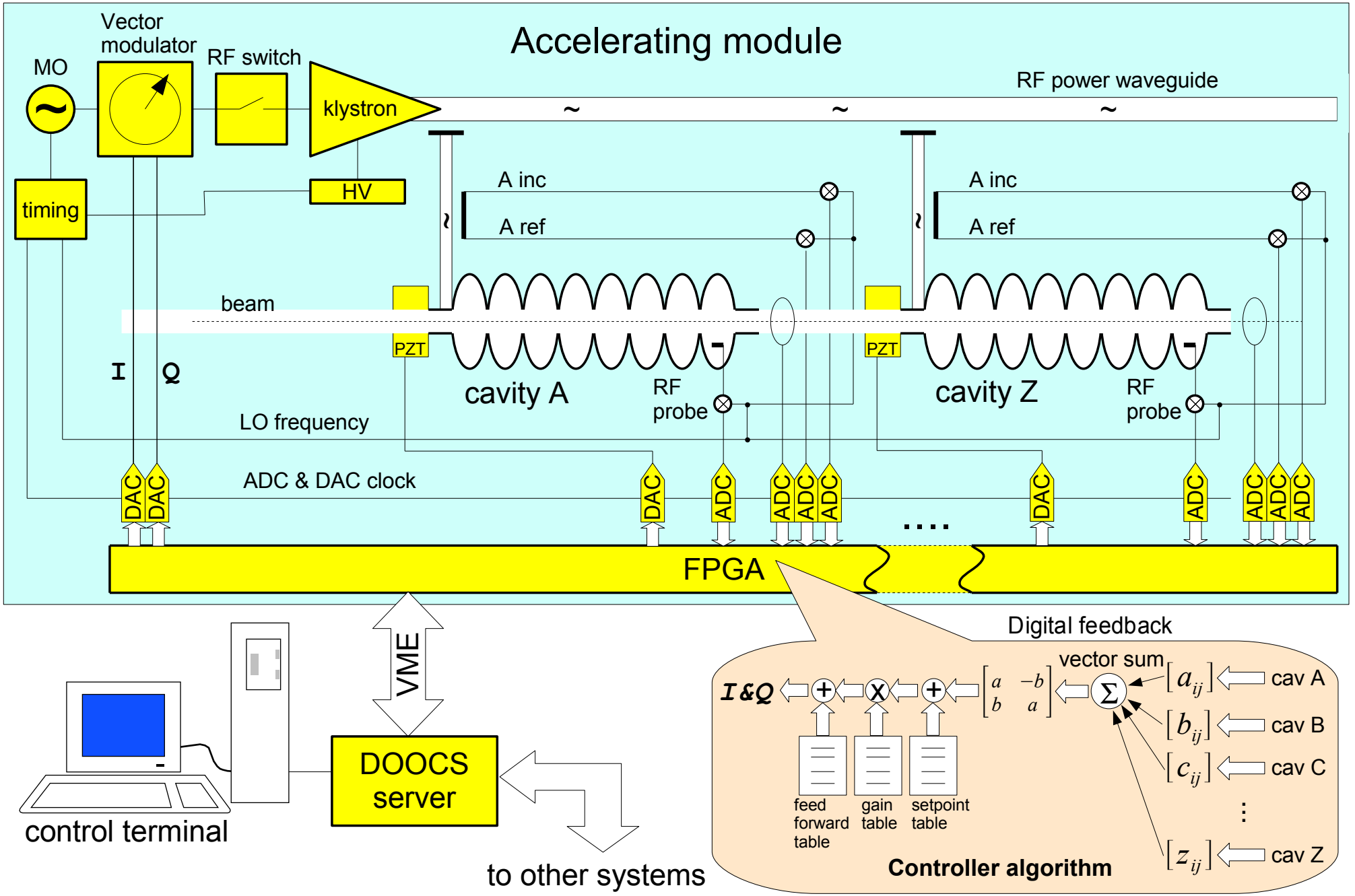
Application of SysML to LLRF system design

M.Grecki



1st annual RFTech meeting, DESY, Hamburg, 29 March 2010

LLRF system architecture



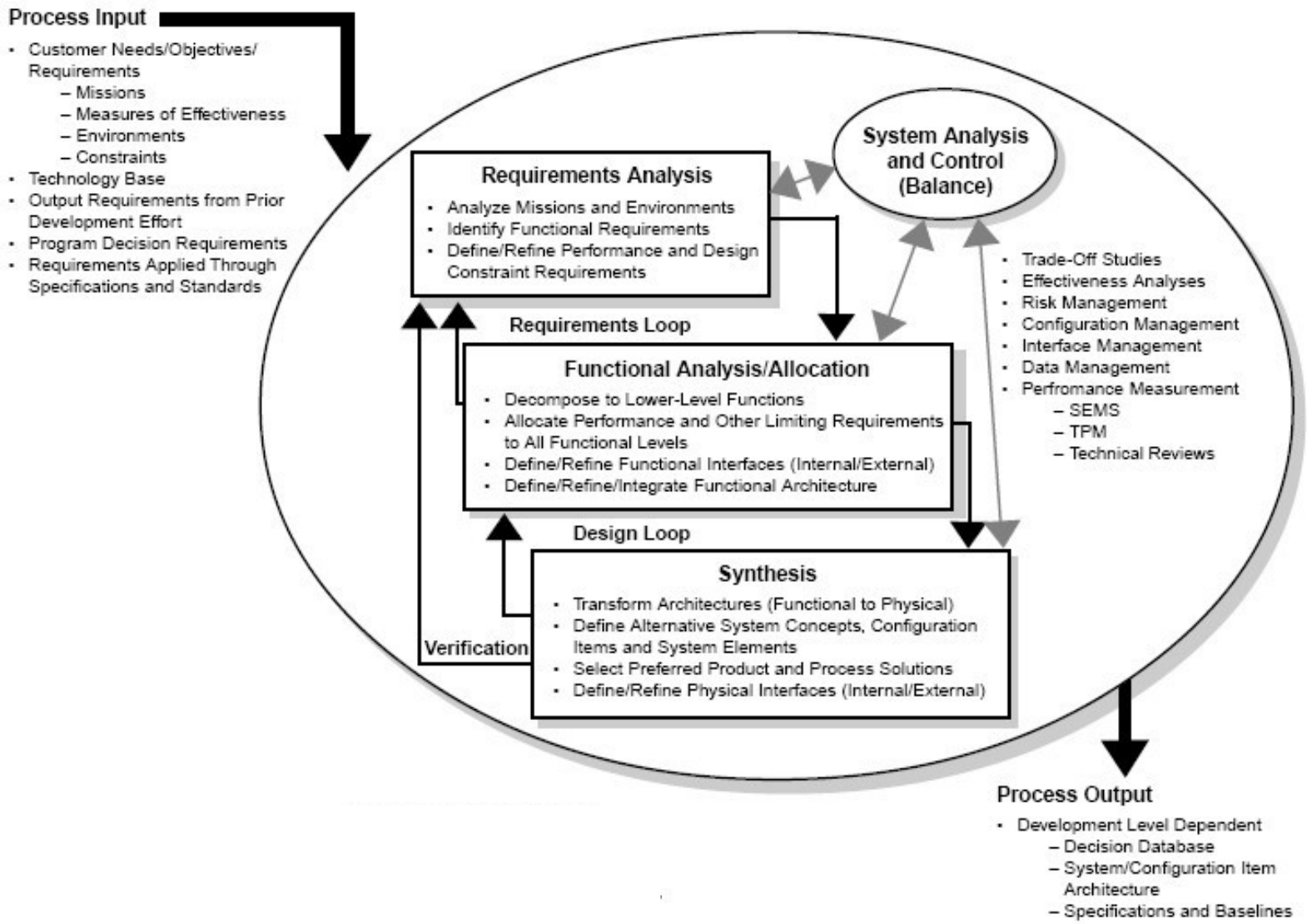
RF Control Requirements

- Maintain Phase and Amplitude of the accelerating field within given tolerances to accelerate a charged particle beam (e.g. XFEL: 0.01% for amplitude and 0.01 deg. for phase)
- Minimize Power needed for control
- RF system must be reproducible, reliable, operable, and well understood
- Other performance goals
  - build-in diagnostics for calibration of gradient and phase, cavity detuning, etc.
  - provide exception handling capabilities
  - meet performance goals over wide range of operating parameters

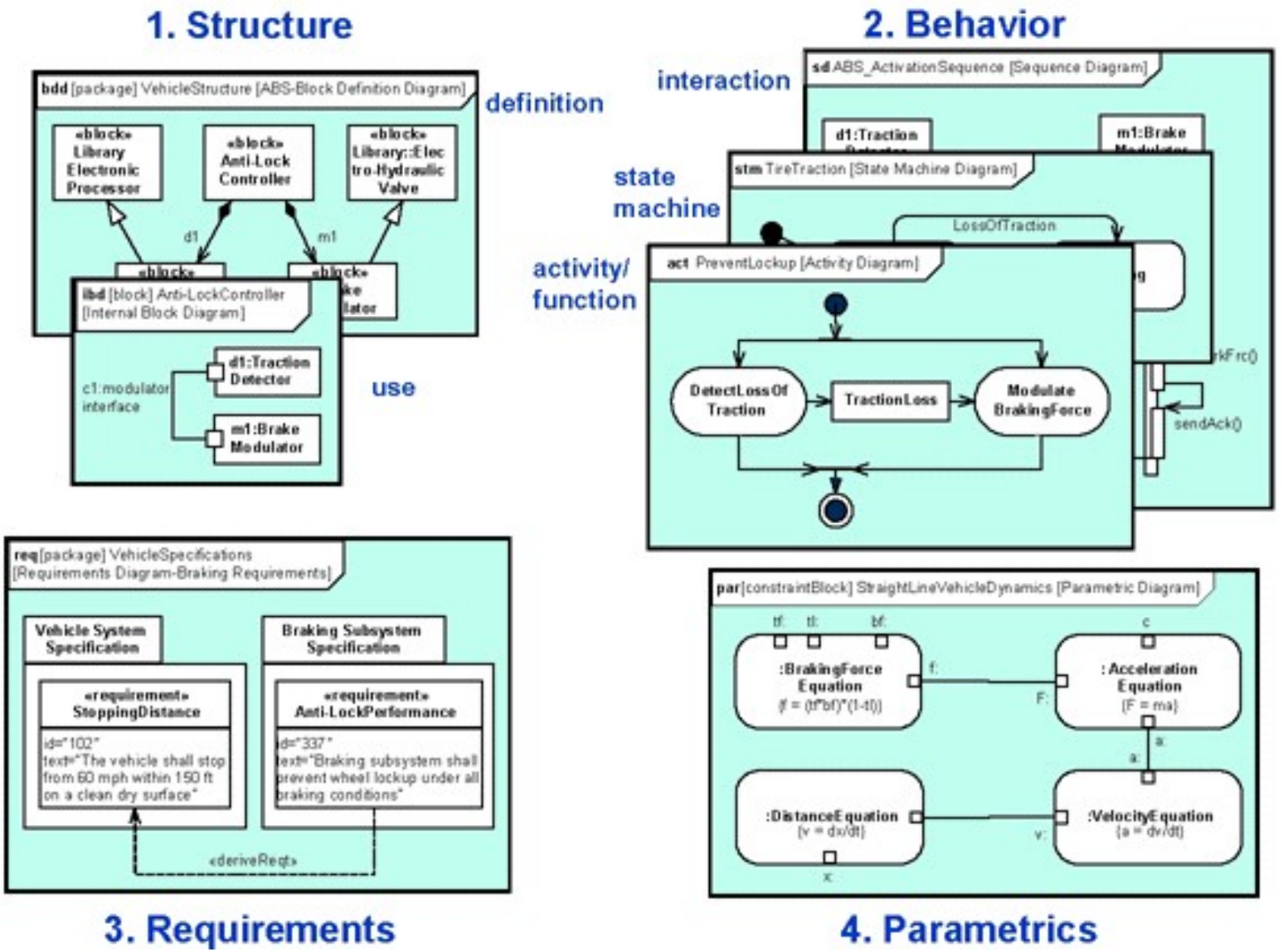
System engineering

- Understand the whole problem before you try to solve it.
- Translate the problem into measurable requirements
- Examine all feasible alternatives before selecting a solution.
- Make sure you consider the total system life cycle. The birth to death concept extends to maintenance, replacement and decommission. If these are not considered in the other tasks, major life cycle costs can be ignored.
- Make sure to test the total system before delivering it.
- Document everything.

Design methodology



SysML - Systems Modeling Language



SysML tools

- We started from Rhapsody (Telelogic – currently IBM)
  - Tool strongly oriented towards automatic code generation and simulation
  - Very complex (this was impression when we were tried to use it)
- Advised by SysML expert we switched to Enterprise Architect (Sparx) as a simpler to use
  - It is not 100% SysML compliant, in some cases it allows to use illegal expressions
- There are other tools (Magic Draw, SysML Toolkit) – we did not try them

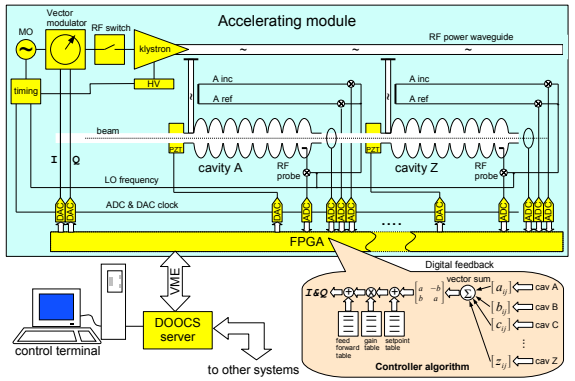
System Model

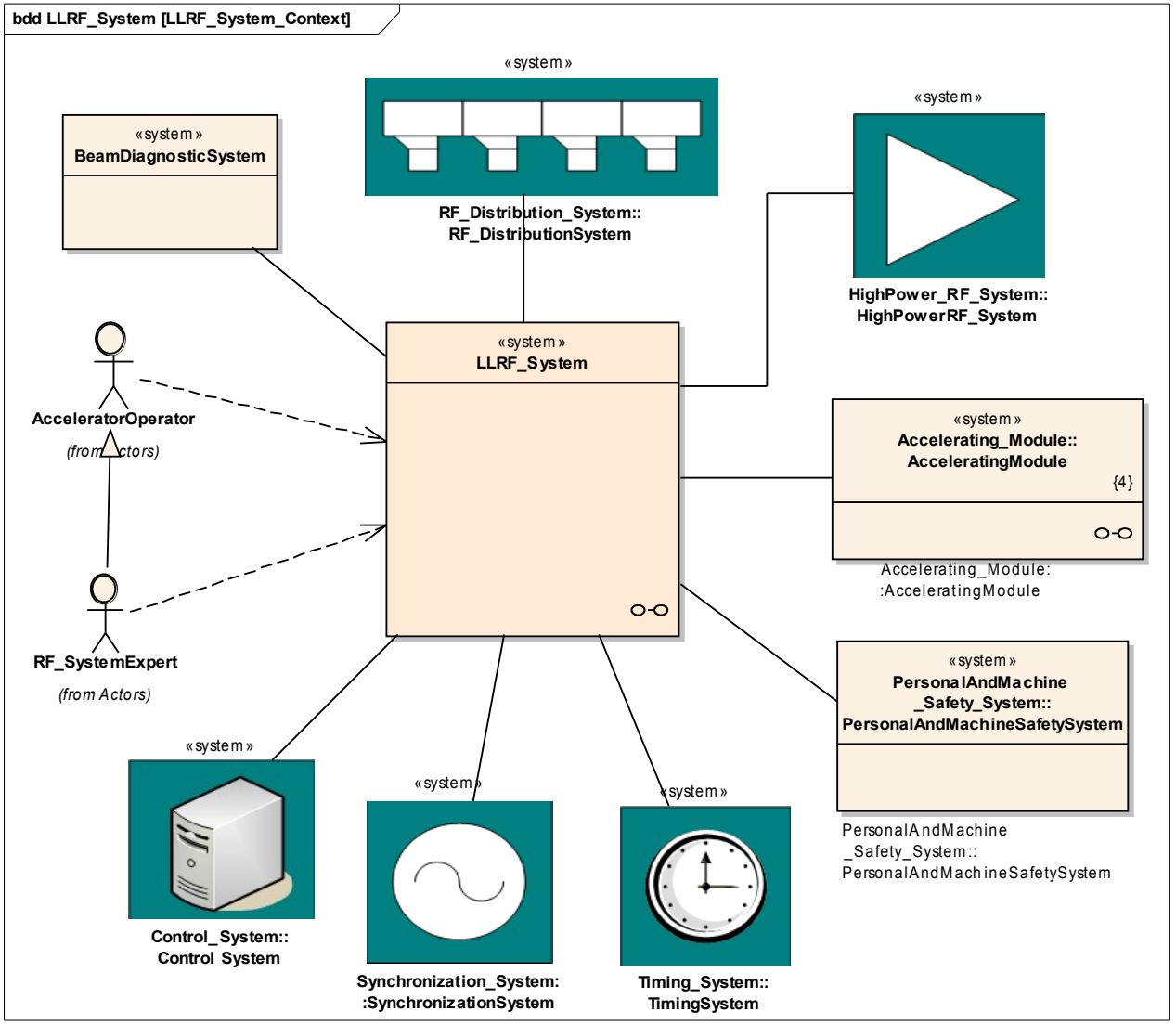
System model is a hierarchical structure consisting diagrams, objects definitions and declarations, objects relationships and their behavior.

All objects are grouped and organized by packages.

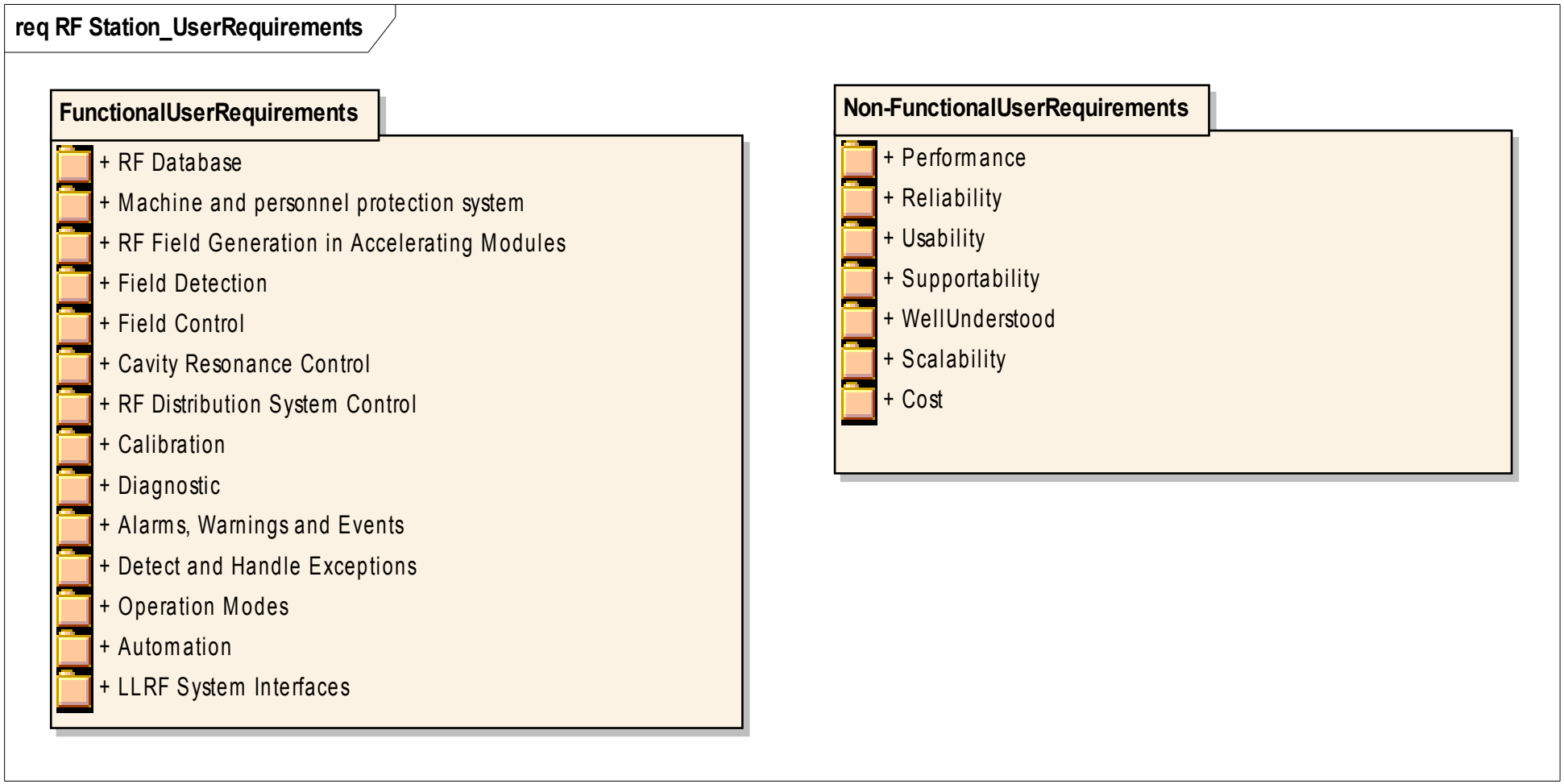
Good hierarchy helps a lot.

LLRF system – context diagram



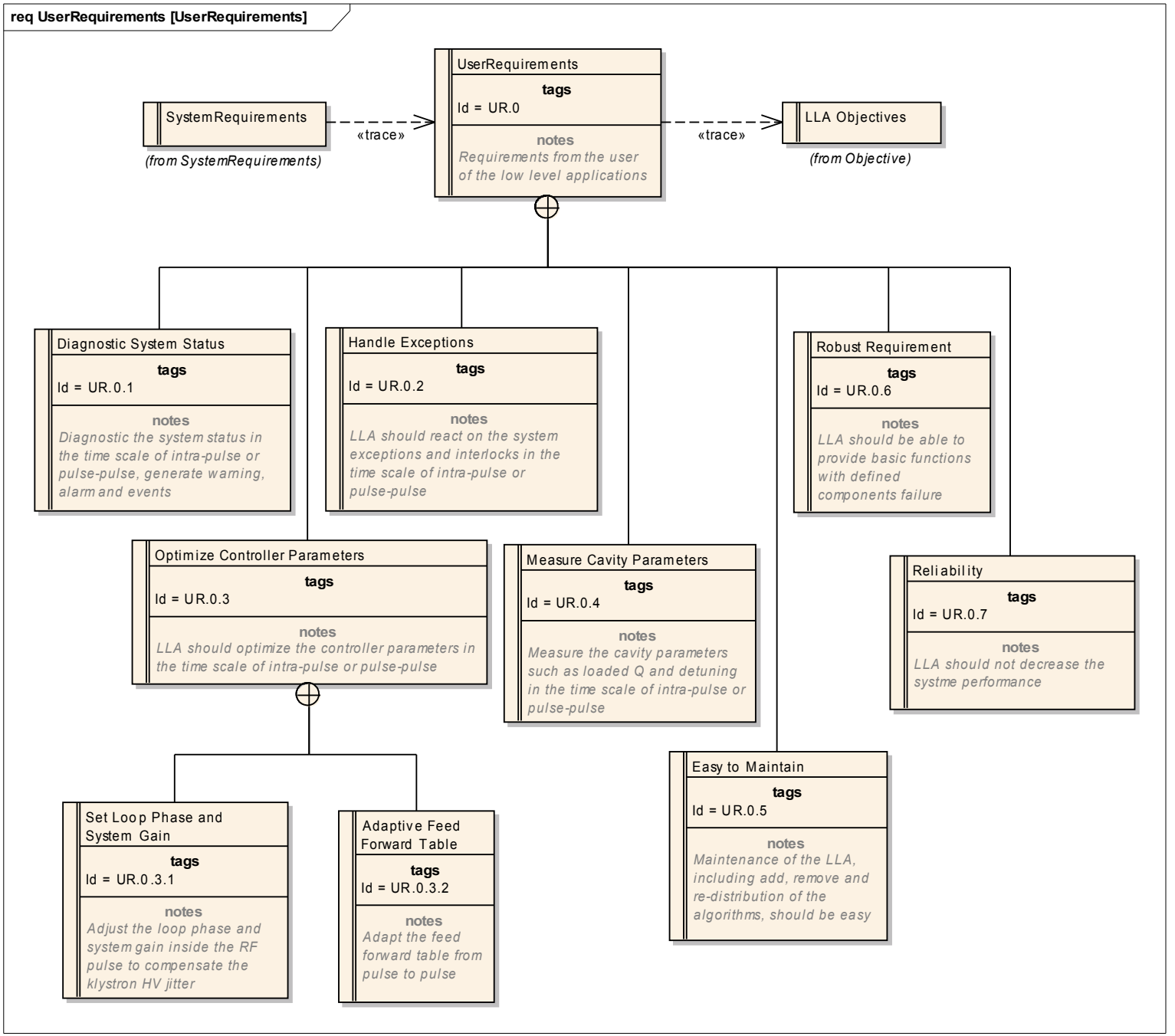


Requirements packages

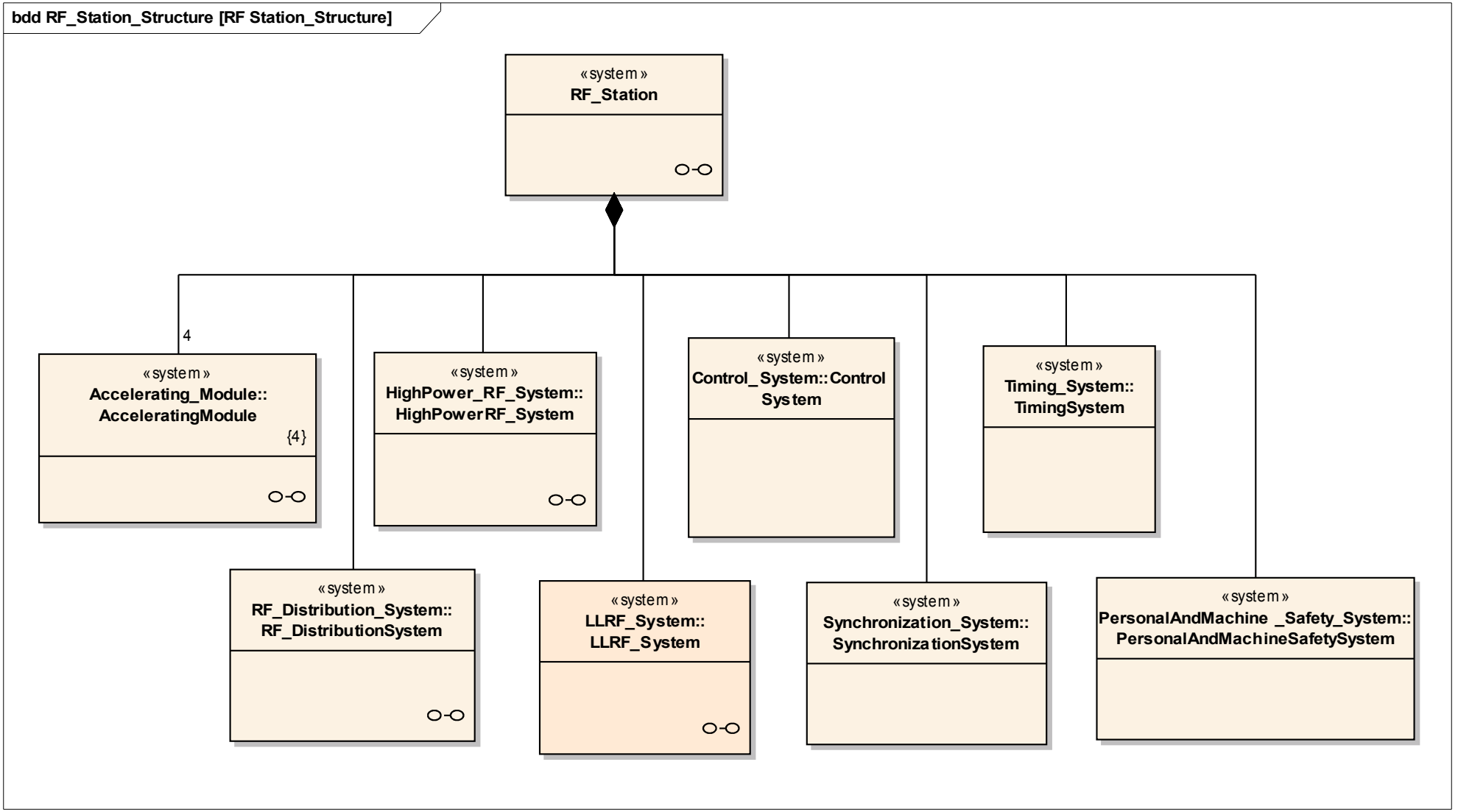


Requirements diagram

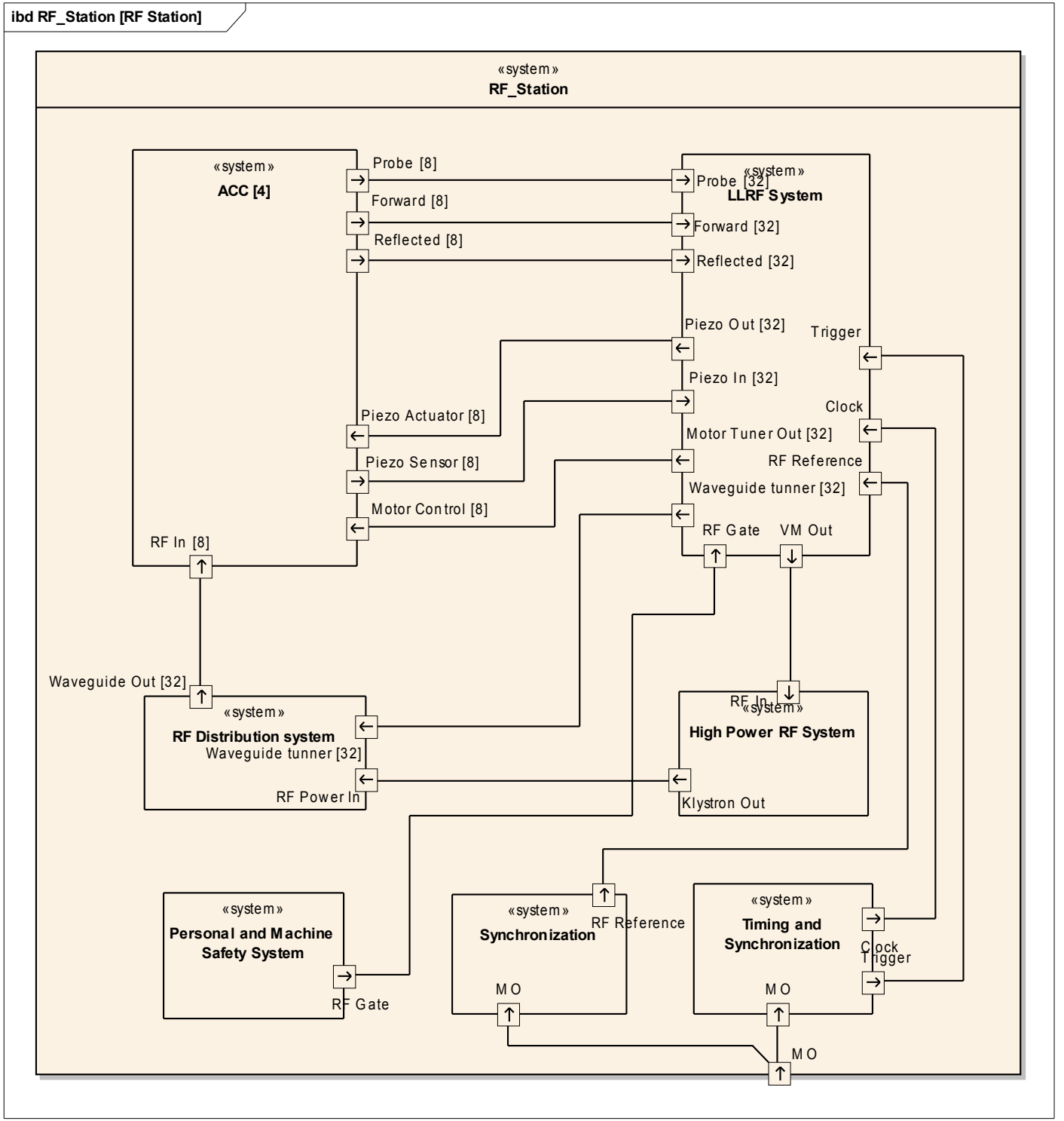
requirements diagram for Low Level Applications



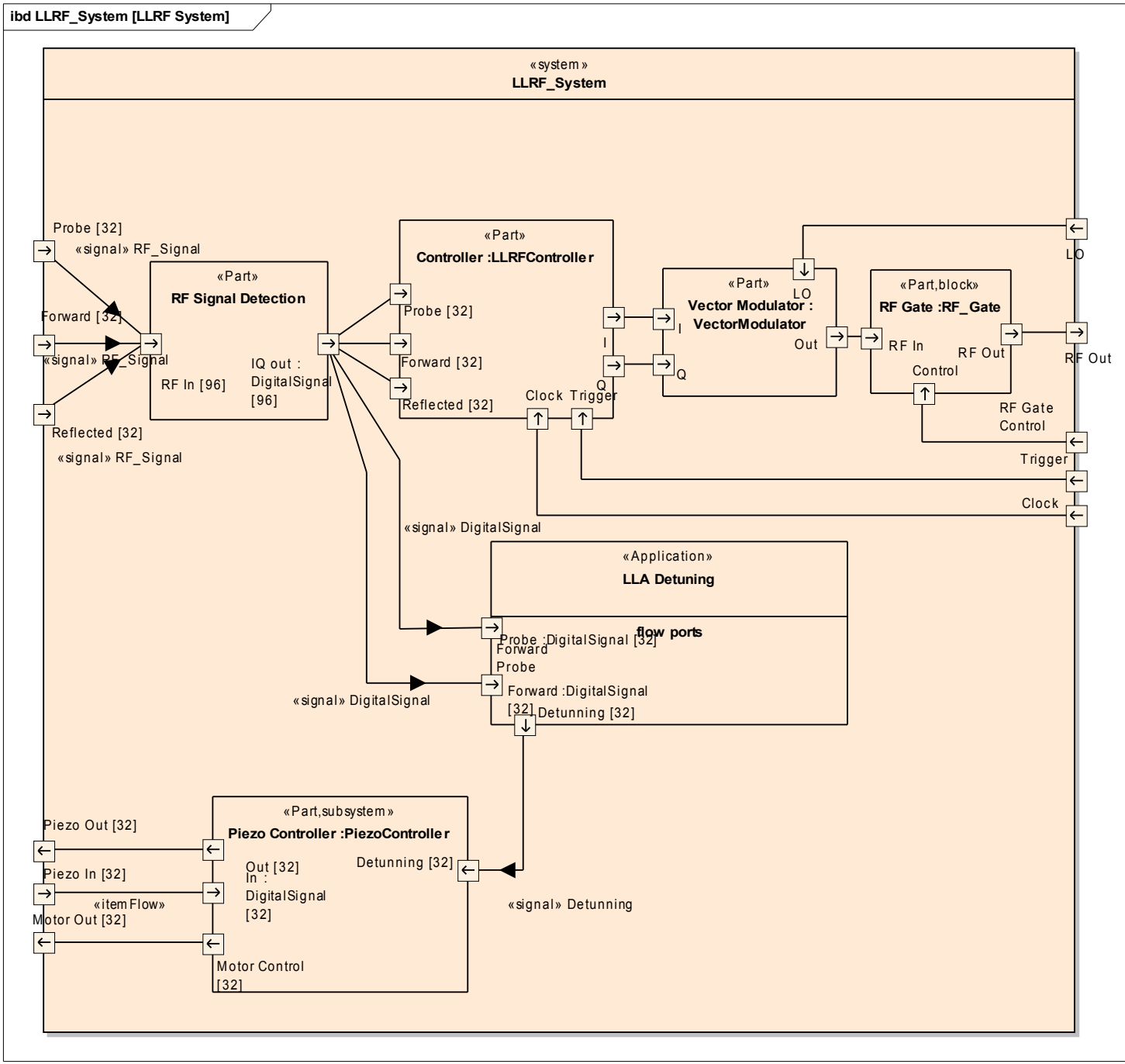
RF station – BDD diagram



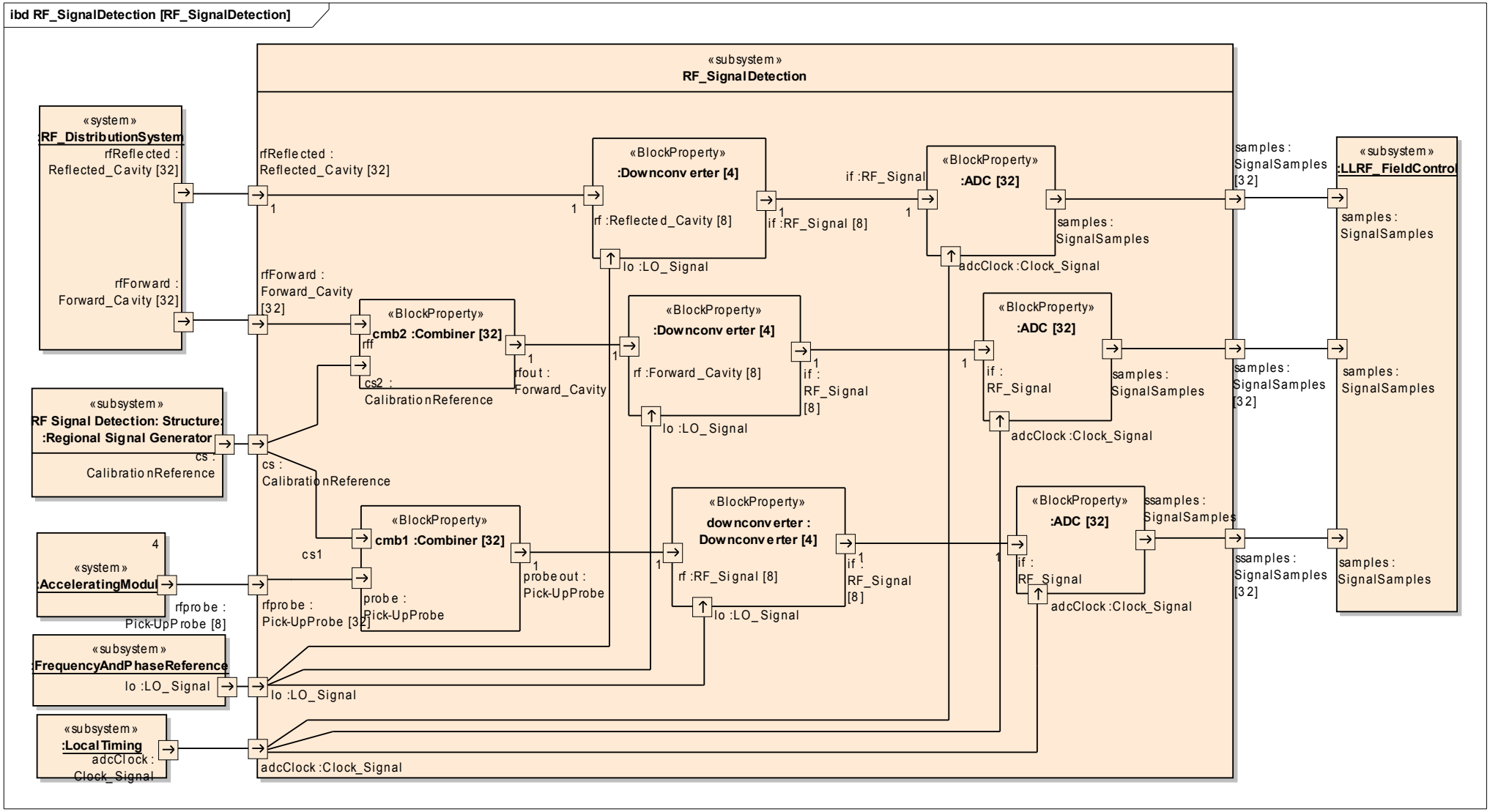
RF station – IBD diagram



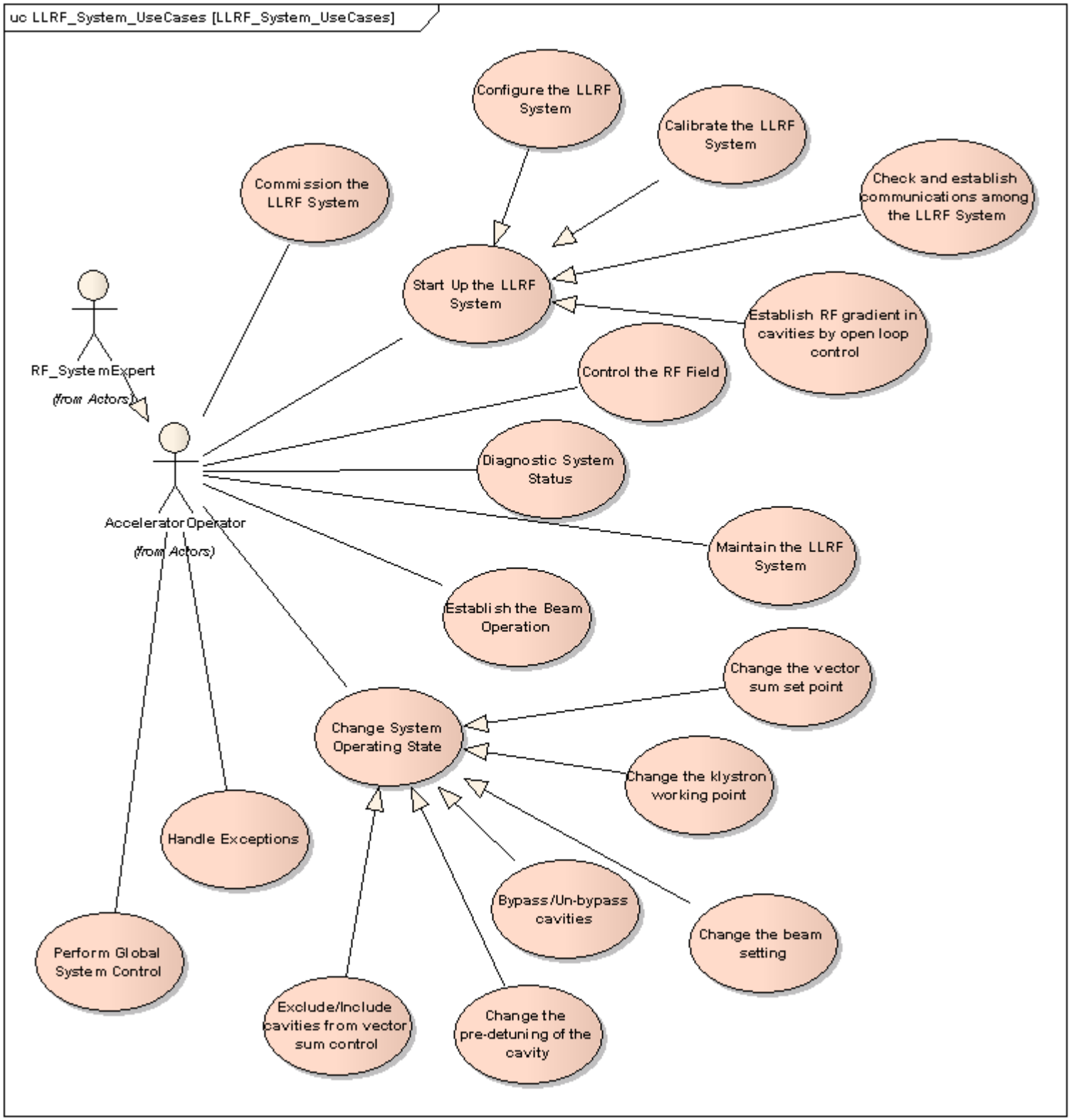
LLRF system – IBD diagram



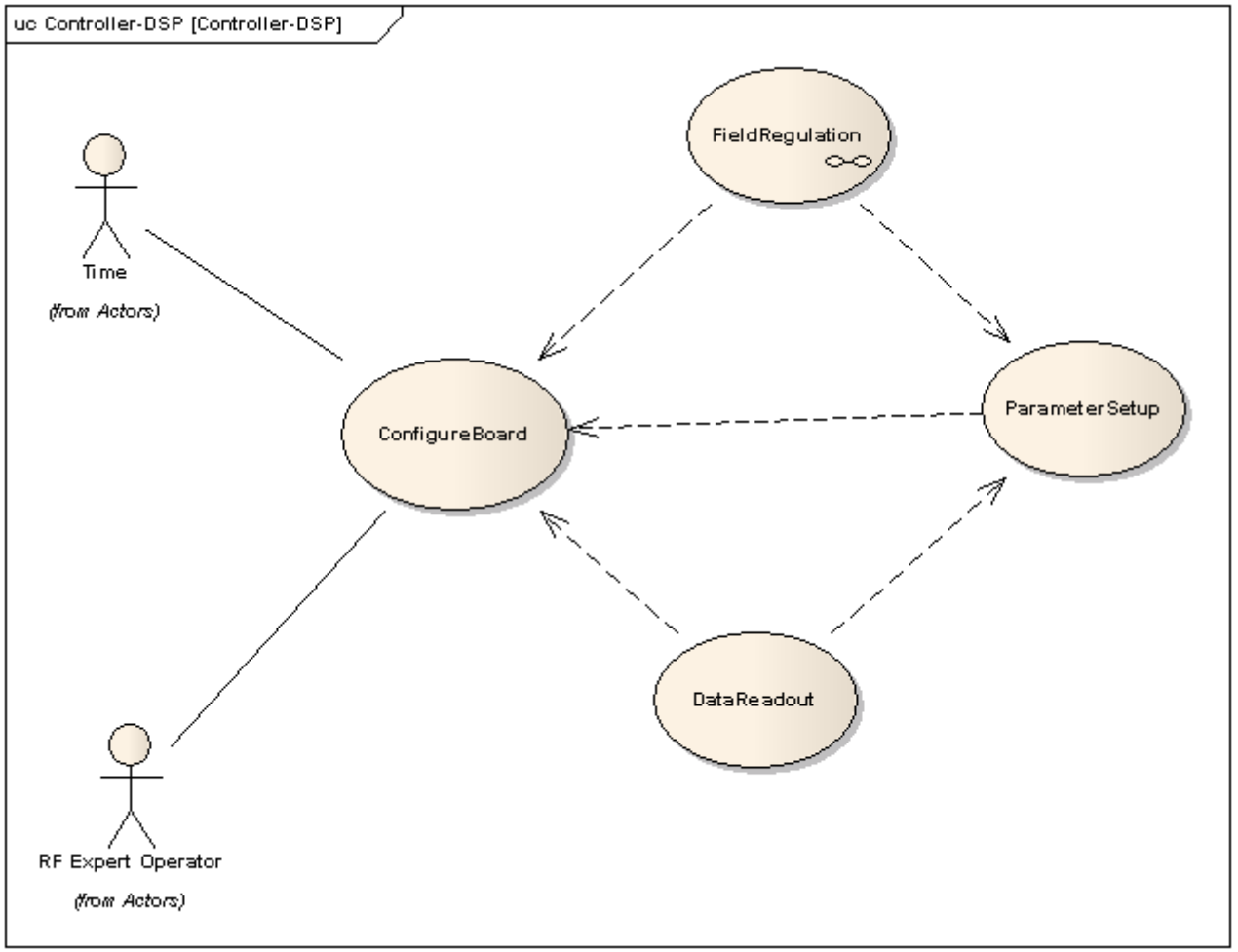
IBD diagram – RF signal detecton



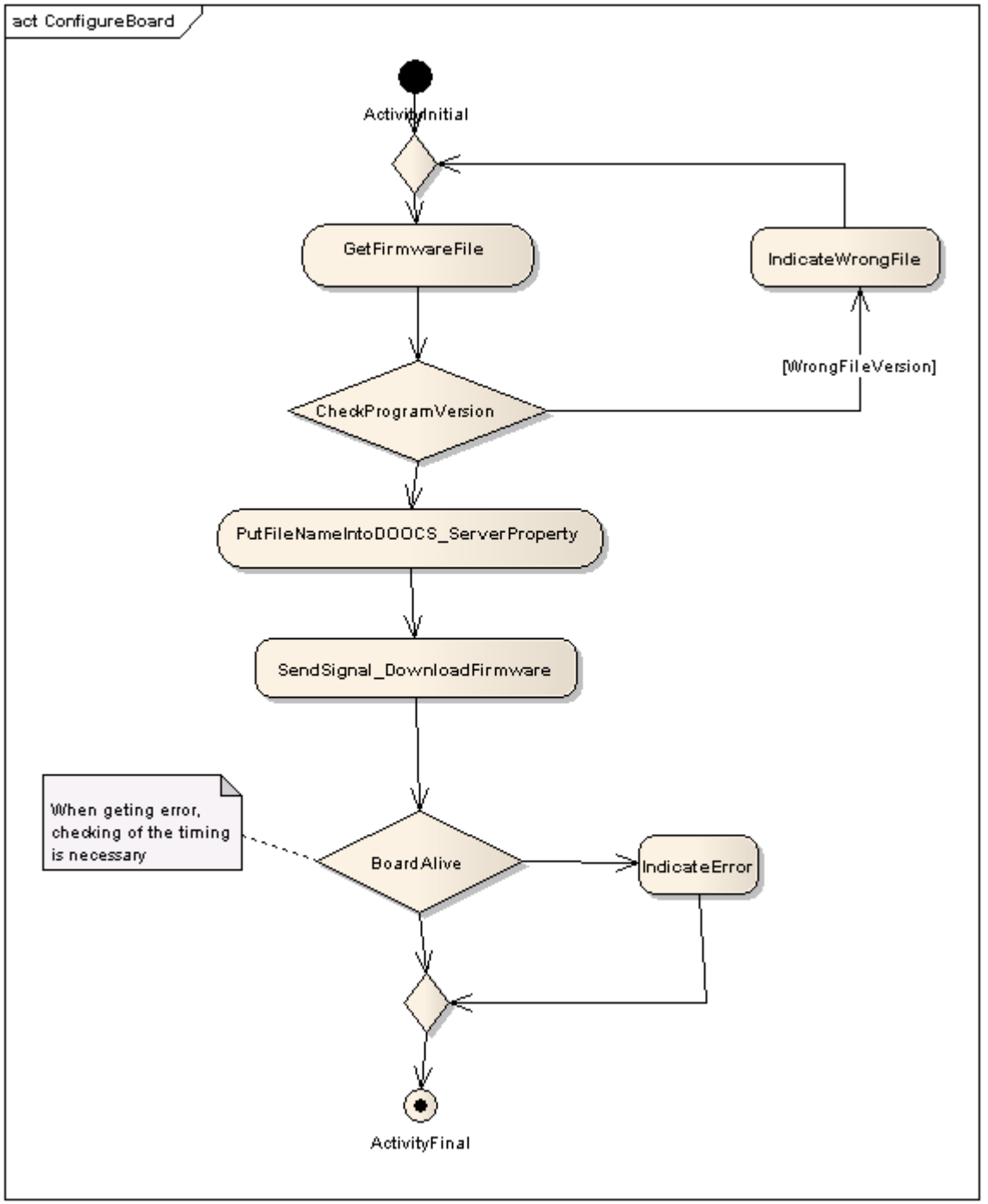
LLRF - Use Case Diagram



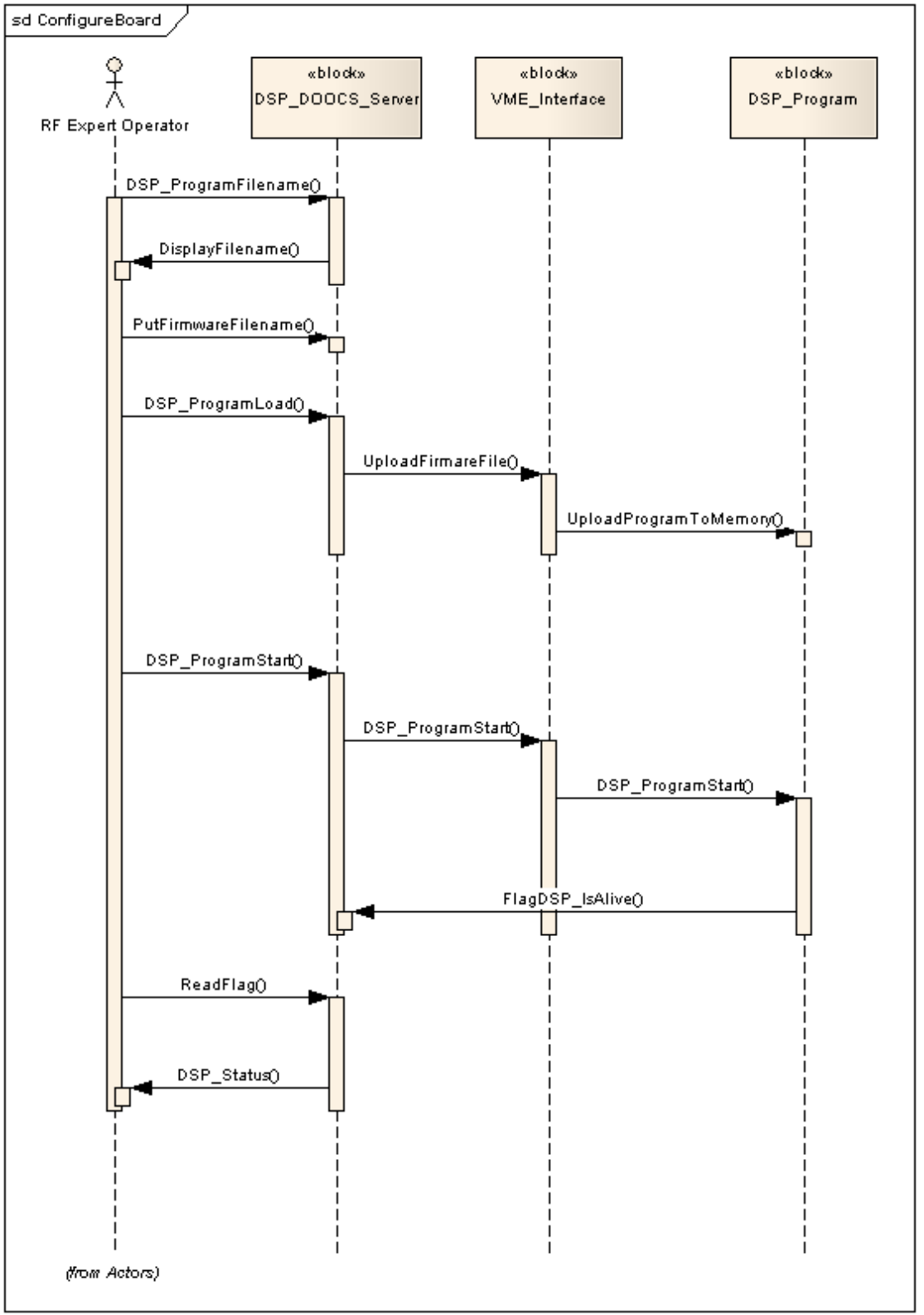
Use Case diagram – controler (simplified)



Activity diagram - configure board



Sequence diagram - configure board



Project Estimation with Use Case Points

- The number of steps to complete the use case.
- The number and complexity of the actors.
- The technical requirements of the use case such as concurrency, security and performance.
- Various environmental factors such as the development teams, experience and knowledge.

Use Case Points

$$UCP = TCF * ECF * UUCP * PF$$

- Technical Complexity Factor (TCF).
- Environment Complexity Factor (ECF).
- Unadjusted Use Case Points (UUCP).
- Productivity Factor (PF).

Technical Complexity Factors

TCF = 0.6 + (0.01*Total Factor)

Technical Factor	Description	Weight
T1	Distributed system	2
T2	Performance	1
T3	End User Efficiency	1
T4	Complex internal Processing	1
T5	Reusability	1
T6	Easy to install	0.5
T7	Easy to use	0.5
T8	Portable	2
T9	Easy to change	1
T11	Concurrent	1
T12	Special security features	1
T13	Provides direct access for third parties	1
T14	Special user training facilities are required	1

Technical Complexity Factors

Technical Factor	Description	Weight	Perceived Complexity	Calculated Factor (weight * perceived complexity)
T1	Distributed system	2	5	10
T2	Performance	1	4	4
T3	End User Efficiency	1	4	4
T4	Complex internal Processing	1	5	5
T5	Reusability	1	2	2
T6	Easy to install	0.5	0	0
T7	Easy to use	0.5	4	2
T8	Portable	2	0	0
T9	Easy to change	1	3	3
T11	Concurrent	1	2	2
T12	Special security features	1	0	0
T13	Provides direct access for third parties	1	2	2
T14	Special user training facilities are required	1	2	2
				36

TCF = 0.6 + (0.01*36)=0.96

Environmental Complexity Factors

ECF = 1.4 + (-0.03*Total Factor)

Environmental Factor	Description	Weight
E1	Familiarity with SysML/UML	1.5
E2	Application Experience	0.5
E3	Object Oriented Experience	1
E4	Lead analyst capability	0.5
E5	Motivation	1
E6	Stable Requirements	2
E7	Part-time workers	-1
E8	Difficult Programming language	-1

Environmental Complexity Factors

Environmental Factor	Description	Weight	Perceived Impact	Calculated Factor (weight*perceived complexity)
E1	Familiarity with SysML/UML	1.5	1	1.5
E2	Application Experience	0.5	3	1.5
E3	Object Oriented Experience	1	3	3
E4	Lead analyst capability	0.5	4	2
E5	Motivation	1	4	4
E6	Stable Requirements	2	3	6
E7	Part-time workers	-1	3	-3
E8	Difficult Programming language	-1	3	-3
				12

ECF = 1.4 + (-0.03*Total Factor)=1.04

Unadjusted Use Case Weight

Use Case Type	Description	Weight
Simple	A simple user interface and touches only a single database entity; its success scenario has 3 steps or less; its implementation involves less than 5 classes.	5
Average	More interface design and touches 2 or more database entities; between 4 to 7 steps; its implementation involves between 5 to 10 classes.	10
Complex	Involves a complex user interface or processing and touches 3 or more database entities; over seven steps; its implementation involves more than 10 classes.	15

UUCW – is based on the total number of activities (or steps) contained in all the use case Scenarios

Unadjusted Actor Weight

Use Case Type	Description	Weight
Simple	The Actor represents another system with a defined API.	1
Average	The Actor represents another system interacting through a protocol, like TCP/IP.	2
Complex	The Actor is a person interacting via an interface.	3

UAW – is based on the combined complexity of all the use cases Actors

Unadjusted Use Case Points

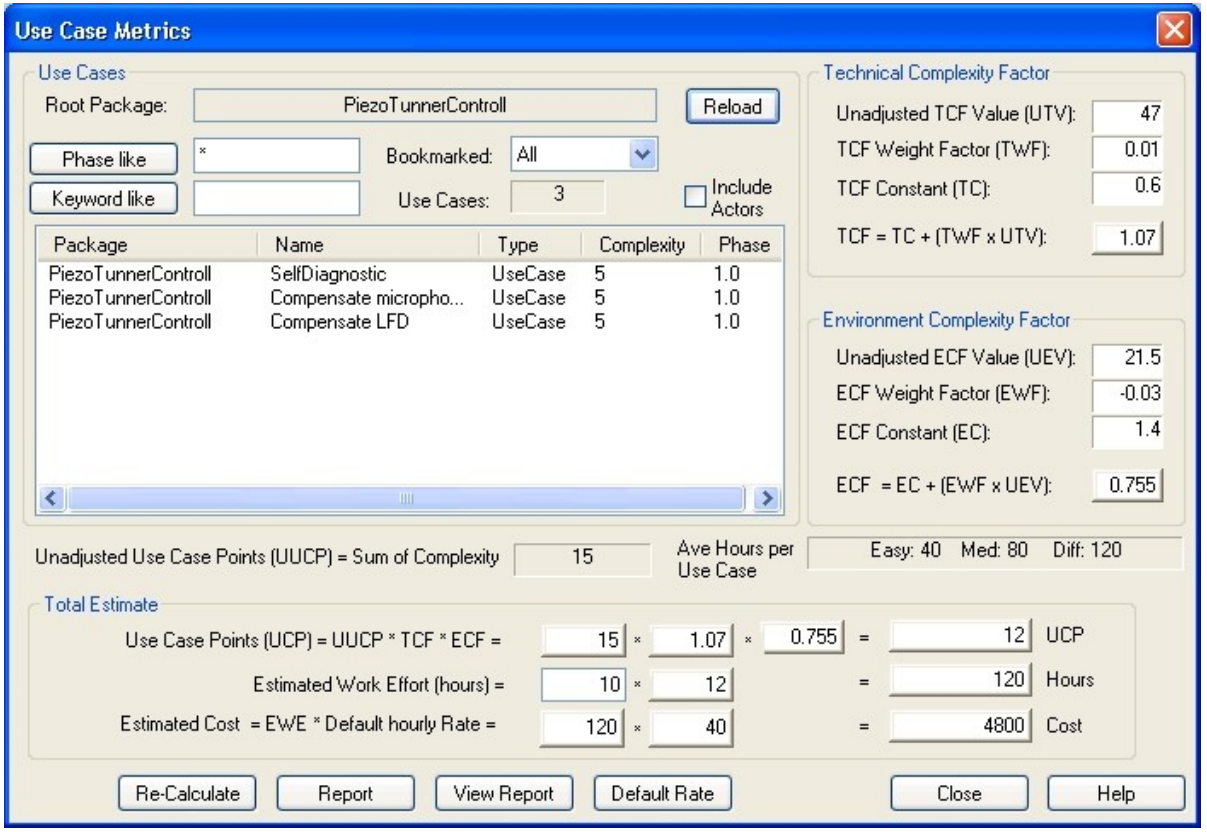
$$UUCP = \sum UC * UUCW + \sum A * UAW$$

Productivity Factor

The Productivity Factor (PF) is a ratio of the number of man hours per use case point based on past projects. If no historical data has been collected, a figure between 15 and 30 is suggested by industry experts. A typical value is 20.

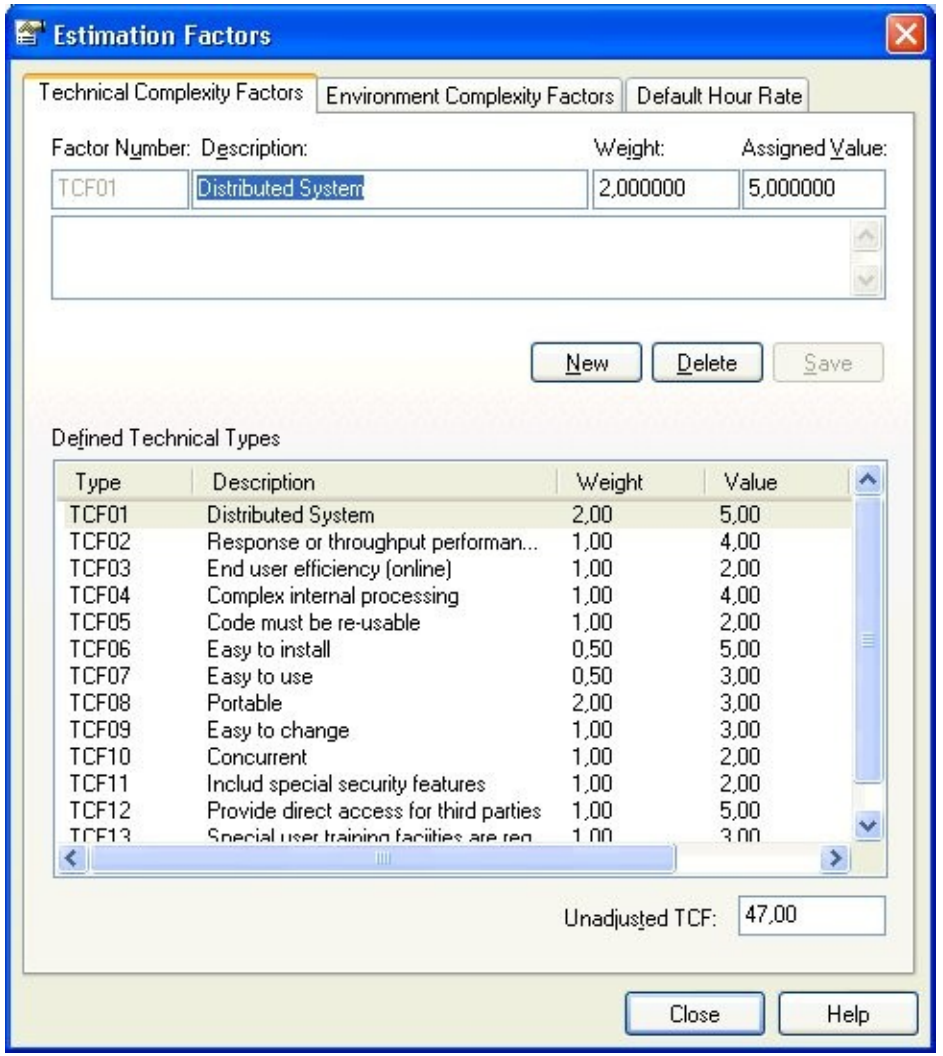
Cost estimation

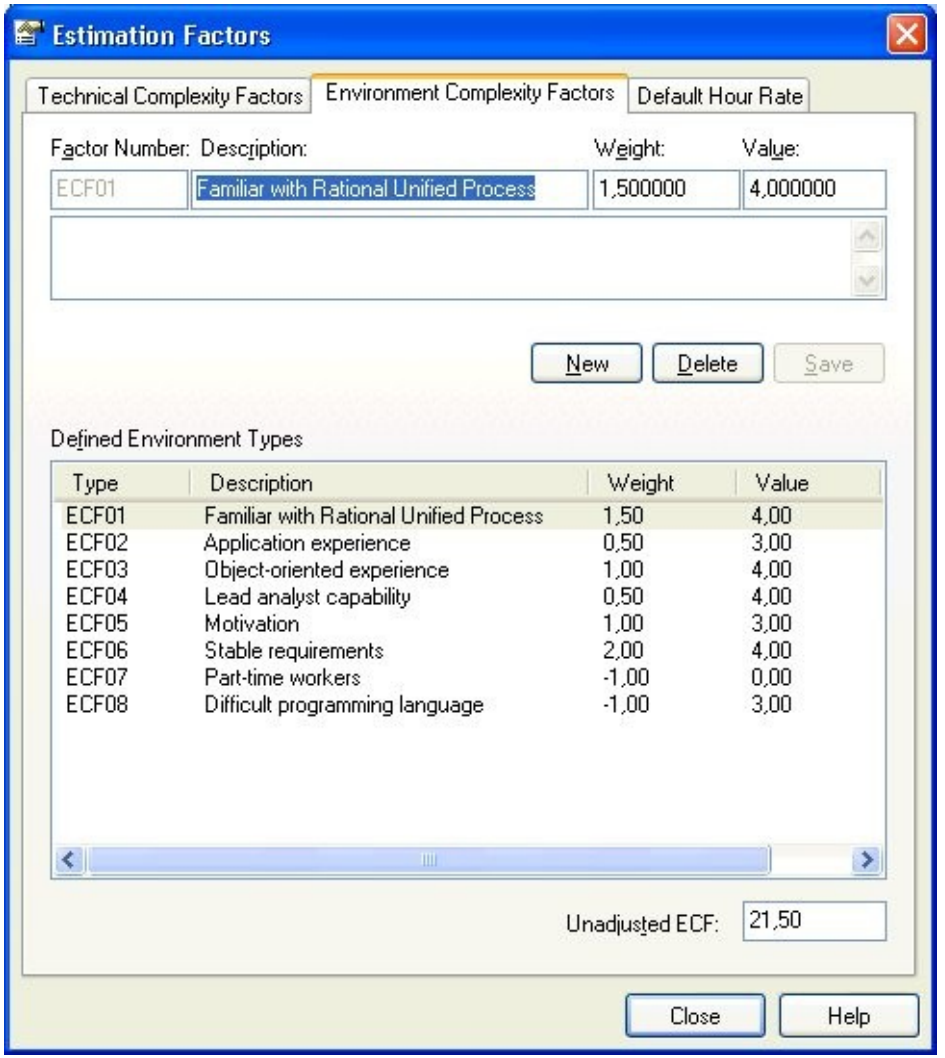
$$UCP = TCP * ECF * UUCP * PF$$



This technique is of value only once you have developed a couple of known projects to use as a baseline. Please DO NOT use the provided 'guesstimates' as a real world measure until you have some real world base lines to measure against.

Tuning parameters for UCP





Conclusion

- The LLRF control for the European XFEL requires careful and well documented design. Since it must integrate the mutual interactions between subsystems of various nature it must be documented in a way understandable by all involved designers (coming from large international collaboration between research labs, universities and industry). The SysML seems to be adequate language for that.
- Since the SysML is a new language it is not easy to start. There is a lack of good design examples one can look at and learn.
- SysML supports software cost estimation through the UCP method.