Reference Missile Functional Architecture, addressing design in a multinational Defense Company

Giulio Telleschi, Andrea Caroni
Missile Dynamics Italy
MBDA Italy
{giulio.telleschi, andrea.caroni}@mbda.it

Ed Willingham
Systems Design & Validation UK
MBDA UK
ed.willingham@mbda.co.uk

Pierre-Henri Pradel
System Design & Validation FR
MBDA France
pierre-henri.pradel@mbda-systems.com

***Abstract*—The need to improve design performance and avoiding rework in missile systems development has lead MBDA to set up an international team to identify a super-set of missile functionalities in any missile domain, ranging from air-to-air to anti-ship missiles. This work led to an open functional architecture that can be tailored for any project, with a generalization of functional elements based on the SysML language and MBSE approach. The Reference Missile Functional Architecture (RMFA) guides the systems architect. Taking into account the evolution of the Defense world and the multinational aspects related both to MBDA and its customers, the RMFA will provide an open, modular, robust, interoperable and cost-efficient reference for future developments. The RMFA is built using SysML and is constructed to have no restriction with respect to national security. The RMFA can be used as an appropriate framework for Knowledge Management to capture and share the company know-how at group level.**

Keywords*— *Model-Based Systems Engineering; Reference Functional Architecture; MBSE; SysML; Defense; Missile; Defense Architecture Framework.

I. Needs

During the last decade, missile design has seen a significant increase in complexity which has originated from:

- Computation resource increases and new high-performance and complex processing techniques has led to an unprecedented demand of capabilities and flexibility;
- Missile interoperability - Missiles are designed for multiple platforms and multi-missions, leading to a wider-ranging requirement set, Concepts of Operations (ConOps) and Concepts of Use (ConUse);
- More stringent safety and security constraints;
- Limited budget constraints forcing greater need for modularity and reuse policies together with commercial equipment adoption;
- Multinational projects and international customers with different cultures and understanding, adding an extra dimension to the above stated points.

Missile functional modelling & architecture should become in the future key enablers to manage the increasing complexity of the missile design ([1], [2]).

MBDA has set up an international team to identify a super-set of missile functionalities in any missile domain, ranging from air-to-air to anti-ship missiles, with the goal to create a reference model that comprises a superset of functions, services and information model which can be referenced and specialized for current and future missiles. Model adoption shall reduce development risks while enabling expertise cross-sharing and to allow for consistency across our products.

II. MBDA's Approach To Model-Based Systems Engineering

The Defense world is traditionally a driver in MBSE adoption and evolution, e.g. defining Defense Architecture Framework (DAF) artefacts using templates such as US DoDAF (Department of Defense Architecture Framework, [3]). A key differentiator between the commercial frameworks and DAF is in the provision of definitions and formalisms; DAF having far greater emphasis on explicitly defining terms in common usage and providing supporting formalisms to provide consistency in application to aid governance.

MBDA's strategy for Model-Based Systems Engineering (MBSE) empowers the advantages provided by cross-sharing and model consistency. Some examples of MBSE within the business, that are shared internationally are:

- MBDAAF, a legacy MBDA DAF;
- International working groups, e.g. RMFA is a delivery of one of these working groups;
- National Capability Teams to foster MBSE within each National Company (NatCo);
- Legacy guidelines and procedures.

Weapon and weapon systems design is usually with limited sharing across NatCos due to restricted or classified information, therefore MBSE is tailored for each project-specific solution.

III. RMFA GOALS

To adopt work sharing where applicable, MBDA encourages the use of MBSE and reference models both at missile and weapon system level (e.g. Command & Control, [4] and [5]). In particular, the role of the RMFA to act as a "reference" is:

- To share a common taxonomy on missile functional analysis;
- To share a common way of thinking a missile functional architecture across the company;
- Through a methodology, which supports how to instantiate RMFA on every specific missile project in order to:
    - Identify the required missile functional architecture;
    - Support the capture of the non-functional viewpoints like performances, safety or security in order to select the appropriate missile physical architecture;
    - Elicit the equipment functional requirements and associated behavior and performances;
    - Avoid an excess of project-specific modelling, spreading within the Company a common way of modelling;
    - Share internationally non-restricted architecture design best practices.
- Enforce top-down MBSE at missile system design level and support the product lines initiatives;
- To enhance Knowledge Management and to capture and share the company expertise at group level;

RMFA has to encompass current and future missile developments in all fields that are part of MBDA portfolio such as air dominance, battlefield engagement, ground based air defense and maritime superiority.

Then RMFA has to be open and evolutionary to be adapted to the future MBDA missile developments and the next technologies insertion.

IV. RMFA PRINCIPLES

The RMFA has been developed through an international working group with missile system design practitioners coming from UK, France and Italy. Regular meetings and collaboration from architects across the business, together with feedback from missile systems development, has enhanced and matured the architecture. This has resulted in a single RMFA being created and adopted across projects throughout our business with our European NatCos. Introducing RMFA benefits and driving the process the detrimental effects of complexity within our products has been greatly reduced.

The system architecture design has to face many challenges (Fig.1):

- It's a multi-viewpoint engineering activity;
- Functions are cross-viewpoints;
- At the end there is only one solution architecture which commits the product development and the stakeholders satisfaction;
- The challenge is to make an early and robust validation of the solution architecture.

Architects Viewpoint embraces the whole system and has to manage its complexity and every specialist viewpoint. Engineering Specialists Viewpoints are derived from each specific field that is analyzed and will identify the various functional allocation constraints and the non-functional requirements (e.g. for performances, safety, integration, V&V, security, as in [6] and [7]). Engineering specialist feedback is paramount to the evolution and development of the RMFA model, which is expressed within the non-functional views that enrich the model and turn it to be useful at every stage of design. When the design and selection processes are complete, the solution architecture is delivered and it includes Program/Project Viewpoint and Sub-contractors viewpoints to manage the project and its deliveries ("viewpoint-driven" method as described in [8]).

The main objective of the RMFA and its joint methodology is to facilitate this multi-viewpoint architecture selection process.

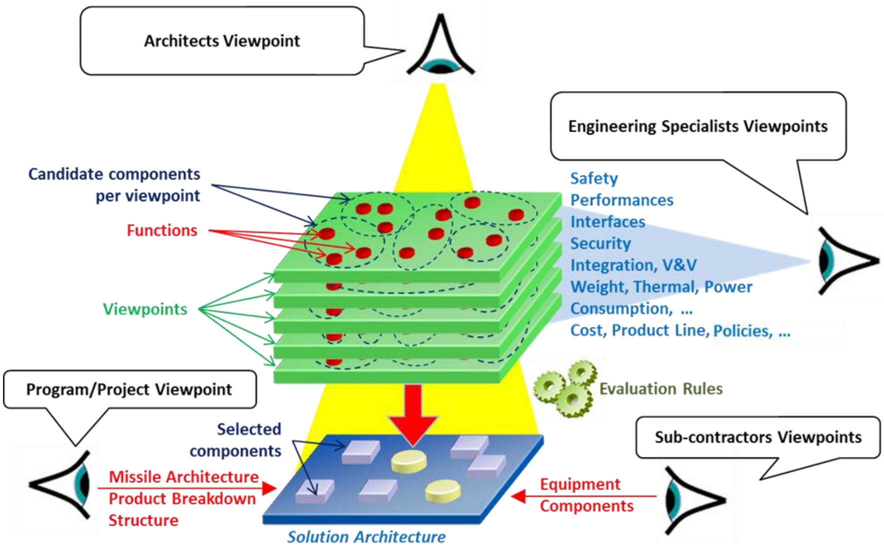

Fig. 1. The challenge of System Architecture Design

V. RMFA Development

RMFA is a missile functional architecture framework structured around:

- Missile use cases;
- Functional chains analysis;
- Missile functions;
- Missile Services (Interfaces);
- Information/Data model;
- Missile sub-system (equipment) functions captured within functional nodes;
- Architecture Examples.

Use cases are to be tailored for each project, mainly depending on the solution that has to be delivered (e.g. maritime superiority, tailoring for an anti-ship missile). Functional elements and functional chains will be shortlisted for each project, according to its specific requirements and design. A particular focus is set on functional chains as they help in defining:

- Sub-architecture of the system architecture which, along with the system use cases, structure the capture of the system functional behavior and its functions decomposition;
- Identify the functional & non-functional requirements attached to the functions;
- Functions allocation to sub-systems will allocate requirements to those sub-systems.

The RMFA is based on a meta-model which describes a schema of the model entities and their relationships between them (Fig.2). Each of these entities is described as follows:

- Generic missile System of Interest (SOI): Describes the SOI which contains functions which represent a superset of functionalities of which a sub set can be allocated to missile products. The SOI also contains a set of ports which represents the superset of external services;
- Use cases: Represent the goals of the generic missile;
- Functional chains: Sequenced functions which provide the system with well-defined sub capabilities;
- Role-based actors: A functional role-based actor (e.g. Launcher, Fig.3) is responsible for a single functional activity within the system. Functional role-based actors allow for project specific missile models to define their physical actors through inheriting role-based actors from RMFA. (See Role-base actor section for more details);
- Black-box functions: A superset of functions that describe the missile ‘high-level’ behavior. Black-box functions would map to the same level as traditionally described in a missile Technical Requirements Specification (TRS);
- White-Box functions: A superset of functions that describe the missile ‘equipment level’ behavior. White-Box functions would map to the same level as traditionally described in equipment TRS’s;
- Functional nodes: Functional nodes represent a grouping of logically related behavior. Functional nodes are identified in order to maximize cohesion, minimize coupling, minimize connectivity and promote reuse. Functional nodes are allocated to missile equipment;
- Port: Associated with the Generic Missile SOI, Role-based Actors and Functional Nodes. Ports provide a ‘gateway’ into entities provided or required interfaces;

- Commodity: A term used to describe data, information and physical entities, such as power. Commodities are passed across interfaces and processed by functions;
- Interface: Contains events which allow interactions;
- Event: Is the interface message;
- Service: A collection of interfaces.

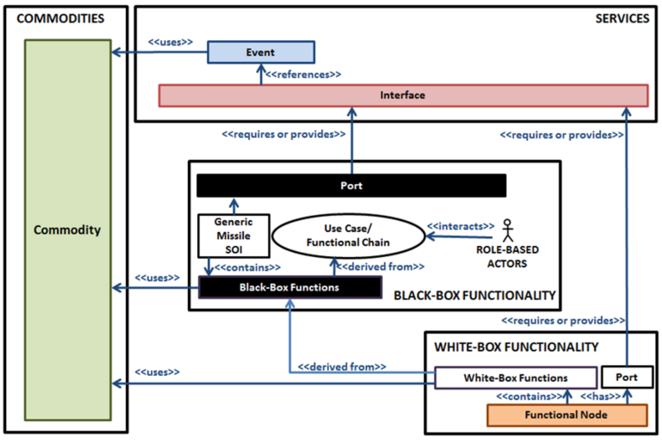

Fig. 2. RFMA meta-model

A. Role-based actors

The RMFA model contains a set of around 30 functional role-based actors which can be used to create project-specific actors for a missile project through the use of the object-oriented technique of inheritance. A functional role-based actor is responsible for a single functional atomic activity within the system, allowing for a flexible solution which can be created quickly and be easily maintained. Projects that adopt the RMFA and use its functional role-based actors will be able to compare capability and functionality between them, harnessing reuse where possible across the solution space.

A quick view of the Generalization relationship allows the interfaces described within the role based actors to reside on the launcher (Fig.3), it shows two variants of a launcher (actor, in blue), a standard "Launcher" and an "Export Launcher". In this instance the Export Launcher capability is a subset of the standard Launcher. The Export Launcher has the capability to provide moment arm and alignment data through role-based actors (in lilac). The standard Launcher inherits the Export Launcher, thus it also can provide both alignment and moment arm data.

To complete the example, Fig.4 shows how the Export Launcher inherits from the functional actors all the functional interfaces (the same applies for commodities).

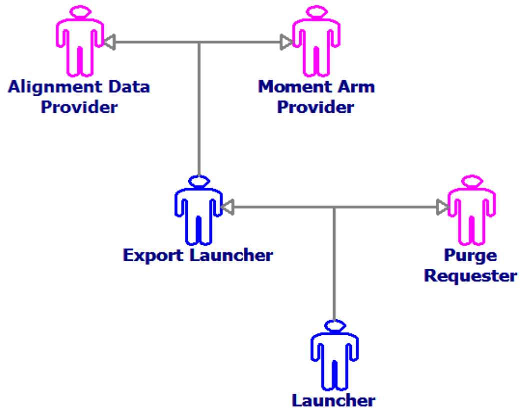

Fig. 3. Launcher & Export Launcher role inheritance

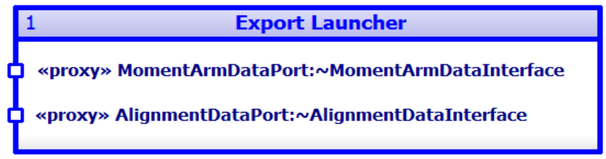

Fig. 4. Actor block showing interface ports

B. Missile Use Cases

The RMFA use cases describe the RMFA goals and boundary of the generic missile. The goals span across the full-lifecycle to include both operational (e.g. effect Target) and non-operational (e.g. dispose System) use cases. The use cases have been developed as per industry standard, through detailing the following for each use case:

- Use case goal: Elaborates the use case name and describes the goal of the use case;
- Primary actor: Describes the actor that initiates the interaction with the generic missile to a achieve a goal;
- Pre-condition: The conditions that should be met before the use case can commence;
- Trigger: The use case initiator;
- Perceived functionality: Describes the superset of functionality that the missile performs to answer the use case goal. Alternative scenarios ("rainy day scenarios") are included at this stage.
- Success post-conditions: The condition the missile is in after the use case has finished successfully.

Note that for operational use cases, the perceived functionality is described based upon the high-level functionality defined within RMFA Functional Chains.

C. Functional Chains

RMFA has been developed through functional chains analysis, where a functional chain sequences together functions which form a well-defined sub capability.

A set of 10 functional chains have been agreed across the international community. Each functional chain is composed of approximately ten functions, to capture the system functional behavior at both missile and equipment levels. During white box analysis, each function in the functional chains will usually be allocated to missile equipment (missile sub-systems).

Such functional analysis aims to identify the functional & non-functional requirements attached to the functions and then will support the sub-systems requirements elicitation once the functions have been allocated to a specific sub-system.

The system architect is encouraged to represent the link between functional chains and use cases (e.g. allocation matrix, Fig.5) and their relationship with time by associating functional chains to missile phases for each use case. These representations are very useful during complex systems development, allowing understanding of the system and its functionalities across multiple specialist teams.

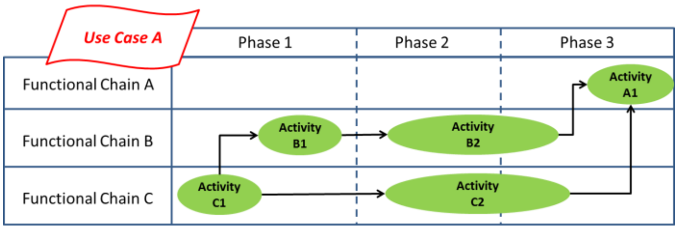

Fig. 5. Example of association between functional chains and missile phases for one use case

D. Missile phases

Missile operational lifecycle can be split in several time-related phases. A taxonomy of missile phases is proposed as well as phase diagrams patterns.

VI. RMFA Use and 'Tuning' to Specific Project

A project-specific missile architecture model contains 3 layers, context, functional and physical architectures to represent the different types of analysis performed. (Fig.6 shows an excerpt). The context architecture describes the missile as a system of interest where external interfaces and high-level functionality is analyzed. This is often referred to as black-box analysis. The functional architecture layer decomposes the high-level system functions into smaller parts which can be allocated as a whole to the physical architecture. The physical architecture describes how the functions within the functional architecture can be allocated to missile equipment where a further allocation of non-functional requirements can be added.

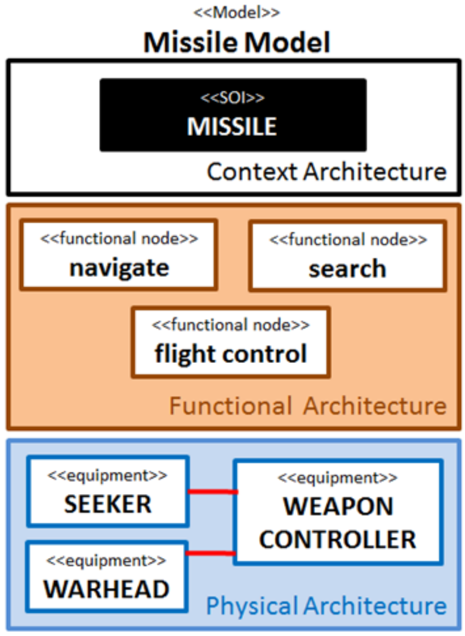

Fig. 6. Missile Architecture Landscape example

The RMFA is layered and as per the project specific missile architecture landscape and contains both a context and functional architecture (Fig.7). The context architecture contains a Reference Missile System of Interest which comprises of a superset of functionality and ports in which a subset of these would be referenced and used within a project-specific architecture. The functional architecture contains a super-set of functional nodes which could again be referenced and used within a project-specific missile architecture. The RMFA also contains a collection of functional interfaces and commodities which can be specialised and tuned according to the missile needs.

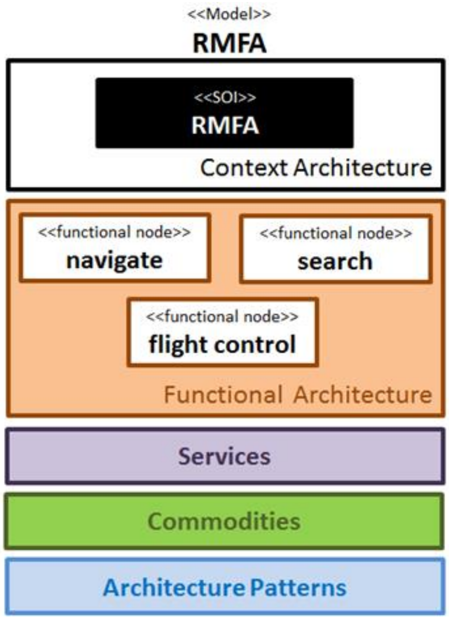

Fig. 7. RMFA Architecture Landscape

RMFA acts like a missile architects toolkit where functionality, commodities and services are referenced into the project-specific architecture through the SysML concept of inheritance (Fig.8, where project-specific missile context architecture is created from RMFA).

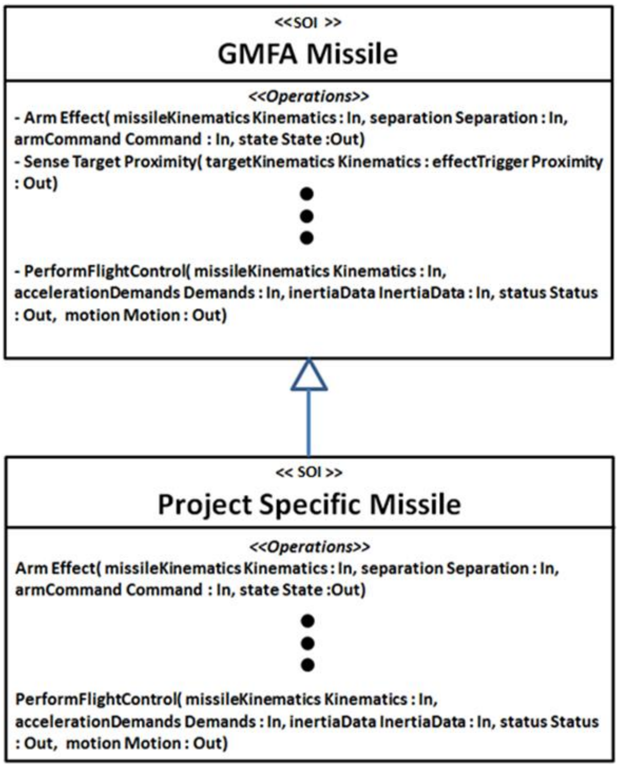

Fig. 8. Inheriting high-level missile functionality from RMFA

RMFA has been designed to be applicable across the MBDA portfolio solution space. The RMFA has been developed in conjunction with our current developments and considerations have been put in place to allow for scalable solution allowing for future products.

RMFA has been designed to allow functionality to be reused directly by projects and to allow functionality to be tailored through the tuning of the function parameters. We can see the process of how functions, both at black and white box levels can be 'tuned' through specialisation and populated accordingly to project needs:

STAGE 1: Describes the original function within RMFA. This example shows the Check Kinematic Criteria function which checks Kinematic Data against a Kinematic Criteria giving a pass or fail result (Fig.9).

STAGE 2: The project references the function and commodities from RMFA from its dedicated missile model. Any commodities that need to be 'tuned' are specialized using a generalization relationship. The example shows the Kinematic Criteria has been specialized where a Project Specific Kinematic Criteria commodity has been created (Fig.10).

STAGE 3: The project completes the tuning by populating the detail of any specialized commodity. In the example the Project Specific Kinematic Criteria is populated with an acceleration constraint (Fig.11).

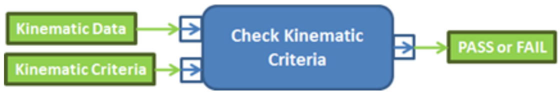

Fig. 9. STAGE 1 – 'Tuning' RMFA Functionality

Fig. 10. STAGE 2 – 'Tuning' RMFA Functionality

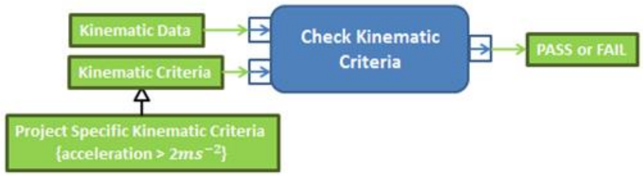

Fig. 11. STAGE 3 – 'Tuning' GMFA Functionality

If the required functionality is not available within RMFA, projects are able to create their own. The new functionality will be used to update RMFA and allow other projects in the MBDA portfolio to reference it.

VII. CONCLUSIONS

The Reference Functional Architecture currently under development in MBDA, has proven to be a valuable asset and has been implemented with success on a number of current projects at various stages of the lifecycle. Its development has been achieved through reverse engineering out current product architectures and missile system design practitioners contribution from across the business. The RMFA is a framework and guide for system architects, allowing them to reuse and tailor, through specialization of artefacts to create a project-specific missile. The RMFA will provide an open, modular, robust, interoperable and cost-efficient reference for future developments while protecting classified and restricted information through a layered approach. In addition, the RMFA has proved to be an excellent asset for knowledge management.

ACKNOWLEDGMENT

This paper summarizes the outcomes that MBDA's international team has produced, the authors would like to thank the entire team.

Dissemination of MBSE in MBDA Italy is made possible thank to Carlo Colandrea and Tonino Genito.

REFERENCES

[1] "Deploying MBSE into an International Company" – Andy Howells - IBM Innovate 2012

[2] "MBDA and our use of MBSE" – Ian Clark - INCOSE UK MBSE WG Meeting 2016

[3] "DoDAF Architecture Framework Version 2.02" - https://dodcio.defense.gov/

[4] "MBDA Extendible Command & Control" - Christian Di Biagio, Pietro Piccirilli et al. - 4th international Conference in Software Engineering for Defense Applications 2015

[5] "Agile process for a Model Based System Engineering (MBSE) design" - Christian Di Biagio and Pietro Piccirilli – 4th international Conference in Software Engineering for Defense Applications 2015

[6] "A Model-Based Safety and Dependability Methodology for Missile Safety Engineering" - Dieter Fasol, Burkhard Münker and Peter Bunus - 33rd International System Safety Conference 2015

[7] "Computer interaction specification example" – Daniel Simmons, Judith Astwood, Kerry Tatlock and Chris Vance – conference on Contemporary Ergonomics and Human Factors 2012

[8] "Methods & Tools for constrained system architecting" Jean-Luc Voirin – INCOSE International Symposium 2014