



# European Component Oriented Architecture (ECOIA®) Collaboration Programme: Architecture Specification Part 7: Metamodel

BAE Ref No: IAWG-ECOIA-TR-011  
Dassault Ref No: DGT 144486-F

Issue: 6

Prepared by  
BAE Systems (Operations) Limited and Dassault Aviation

This specification is developed by BAE SYSTEMS, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE SYSTEMS, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

**Note:** *This specification represents the output of a research programme. Compliance with this specification shall not in itself relieve any person from any legal obligations imposed upon them. Product development should rely on the DefStan or BNAE publications of the ECOIA standard.*

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## Contents

<b>0</b>	<b>Introduction</b>	<b>v</b>
<b>1</b>	<b>Scope</b>	<b>1</b>
<b>2</b>	<b>Warning</b>	<b>1</b>
<b>3</b>	<b>Normative References</b>	<b>1</b>
<b>4</b>	<b>Definitions</b>	<b>2</b>
<b>5</b>	<b>Abbreviations</b>	<b>2</b>
<b>6</b>	<b>ECOA Meta Models</b>	<b>3</b>
<b>6.1</b>	<b>Abstract Metamodel</b>	<b>4</b>
<b>6.1.1</b>	<b>Overview</b>	<b>4</b>
<b>6.1.2</b>	<b>Service Definition</b>	<b>6</b>
<b>6.1.3</b>	<b>Component Definition</b>	<b>9</b>
<b>6.1.4</b>	<b>Assembly Schema</b>	<b>10</b>
<b>6.1.5</b>	<b>Component Implementation</b>	<b>11</b>
<b>6.1.6</b>	<b>Operation Links</b>	<b>14</b>
<b>6.1.7</b>	<b>Data Types</b>	<b>15</b>
<b>6.1.8</b>	<b>Deployment Schema and Logical System</b>	<b>19</b>
<b>6.1.9</b>	<b>Cross Platform View</b>	<b>21</b>
<b>6.2</b>	<b>Concrete Metamodel</b>	<b>21</b>
<b>6.2.1</b>	<b>Mapping onto Service-Component Architecture (SCA)</b>	<b>21</b>
<b>6.2.2</b>	<b>Schemas</b>	<b>24</b>
<b>6.2.3</b>	<b>Filename Conventions</b>	<b>25</b>
<b>6.2.4</b>	<b>Interim data organisation</b>	<b>27</b>
<b>7</b>	<b>Legality Rules</b>	<b>30</b>
<b>7.1</b>	<b>Types</b>	<b>30</b>
<b>7.2</b>	<b>Services</b>	<b>30</b>
<b>7.3</b>	<b>Component Definitions</b>	<b>30</b>
<b>7.4</b>	<b>Cross Platform View</b>	<b>30</b>
<b>7.5</b>	<b>Initial Assembly schema</b>	<b>30</b>
<b>7.6</b>	<b>Component Implementations</b>	<b>31</b>
<b>7.7</b>	<b>Composites</b>	<b>31</b>
<b>7.8</b>	<b>Integration</b>	<b>31</b>
<b>7.9</b>	<b>Final Assembly schema</b>	<b>32</b>
<b>7.10</b>	<b>Logical System</b>	<b>33</b>
<b>8</b>	<b>ECOA XML Schema Definitions</b>	<b>34</b>
<b>8.1</b>	<b>ecoa-sca-2.0.xsd</b>	<b>34</b>

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

8.2	<b>ecoa-sca-attributes-2.0.xsd</b>	34
8.3	<b>ecoa-sca-interface-2.0.xsd</b>	35
8.4	<b>ecoa-sca-instance-2.0.xsd</b>	37
8.5	<b>ecoa-bin-desc-2.0.xsd</b>	38
8.6	<b>ecoa-common-2.0.xsd</b>	41
8.7	<b>ecoa-deployment-2.0.xsd</b>	44
8.8	<b>ecoa-implementation-2.0.xsd</b>	50
8.9	<b>ecoa-interface-2.0.xsd</b>	68
8.10	<b>ecoa-interface-qos-2.0.xsd</b>	70
8.11	<b>ecoa-logicalsysteM-2.0.xsd</b>	74
8.12	<b>ecoa-types-2.0.xsd</b>	79
8.13	<b>ecoa-project-2.0.xsd</b>	85
8.14	<b>ecoa-cross-platforms-view-2.0.xsd</b>	87
8.15	<b>ecoa-uid-2.0.xsd</b>	90
8.16	<b>sca-1.1-cd06-subset-2.0.xsd</b>	91
8.17	<b>sca-contribution-1.1-cd06-subset-2.0.xsd</b>	92
8.18	<b>sca-core-1.1-cd06-subset-2.0.xsd</b>	93
8.19	<b>sca-implementation-composite-1.1-cd06-subset-2.0.xsd</b>	99
9	<b>Specifications of the SCA Subset</b>	100
9.1	<b>sca-1.1-cd06.xsd</b>	100
9.2	<b>sca-contribution-1.1-cd06.xsd</b>	101
9.3	<b>sca-core-1.1-cd06.xsd</b>	101

## Figures

<b>Figure 1</b>	<b>ECOA Meta Models</b>	<b>3</b>
<b>Figure 2</b>	<b>Overview of Metamodel</b>	<b>4</b>
<b>Figure 3</b>	<b>ServiceDefinition metamodel</b>	<b>6</b>
<b>Figure 4</b>	<b>ServiceQoS metamodel</b>	<b>7</b>
<b>Figure 5</b>	<b>ComponentDefinition metamodel</b>	<b>9</b>
<b>Figure 6</b>	<b>AssemblySchema and ServiceLink metamodel</b>	<b>10</b>
<b>Figure 7</b>	<b>ComponentImplementation metamodel</b>	<b>11</b>
<b>Figure 8</b>	<b>OperationLink metamodel</b>	<b>14</b>
<b>Figure 9</b>	<b>Data Type definition</b>	<b>16</b>
<b>Figure 10</b>	<b>Supported Data Types</b>	<b>17</b>
<b>Figure 11</b>	<b>Records and VariantRecords</b>	<b>18</b>
<b>Figure 12</b>	<b>Deployment Schema</b>	<b>19</b>

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

<b>Figure 13 Log Policy Definition</b>	<b>20</b>
<b>Figure 14 Directories</b>	<b>27</b>

## **Tables**

<b>Table 1 Specific QoS attributes on operations</b>	<b>7</b>
<b>Table 2 XPath Expressions</b>	<b>22</b>
<b>Table 3 Relations between the ECOA abstract metamodel and the SCA Assembly model</b>	<b>23</b>
<b>Table 4 ECOA Defined Schemas</b>	<b>24</b>
<b>Table 5 ECOA Standard Filenames</b>	<b>25</b>
<b>Table 6 Model Data Organisation</b>	<b>28</b>

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 0 Introduction

This Architecture Specification provides the specification for creating ECOA<sup>®</sup>-based systems. It describes the standardised programming interfaces and data-model that allow a developer to construct an ECOA<sup>®</sup>-based system. The details of the other documents comprising the rest of this Architecture Specification can be found in Section 3.

This document is Part 7 of the Architecture Specification, and contains the Metamodel and XML schema definitions for an ECOA<sup>®</sup> system.

The document is structured as follows:

- Section 6 describes the ECOA<sup>®</sup> metamodel;
- Section 7 provides legality rules to follow when writing ECOA<sup>®</sup> XML files;
- Section 8 details the schemas;
- Section 9 specifies the SCA subset selected for ECOA<sup>®</sup>.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 1 Scope

This Architecture Specification specifies a uniform method for design, development and integration of software systems using a component oriented approach.

## 2 Warning

This specification represents the output of a research programme. Compliance with this specification shall not in itself relieve any person from any legal obligations imposed upon them. Product development should rely on the DefStan or BNAE publications of the ECOA standard.

## 3 Normative References

Architecture Specification Part 1	IAWG-ECO-TR-001 / DGT 144474 Issue 6 Architecture Specification Part 1 – Concepts
Architecture Specification Part 2	IAWG-ECO-TR-012 / DGT 144487 Issue 6 Architecture Specification Part 2 – Definitions
Architecture Specification Part 3	IAWG-ECO-TR-007 / DGT 144482 Issue 6 Architecture Specification Part 3 – Mechanisms
Architecture Specification Part 4	IAWG-ECO-TR-010 / DGT 144485 Issue 6 Architecture Specification Part 4 – Software Interface
Architecture Specification Part 5	IAWG-ECO-TR-008 / DGT 144483 Issue 6 Architecture Specification Part 5 – High Level Platform Requirements
Architecture Specification Part 6	IAWG-ECO-TR-006 / DGT 144481 Issue 6 Architecture Specification Part 6 – ECOA <sup>®</sup> Logical Interface
Architecture Specification Part 7	IAWG-ECO-TR-011 / DGT 144486 Issue 6 Architecture Specification Part 7 – Metamodel
Architecture Specification Part 8	IAWG-ECO-TR-004 / DGT 144477 Issue 6 Architecture Specification Part 8 – C Language Binding
Architecture Specification Part 9	IAWG-ECO-TR-005 / DGT 144478 Issue 6 Architecture Specification Part 9 – C++ Language Binding
Architecture Specification Part 10	IAWG-ECO-TR-003 / DGT 144476 Issue 6 Architecture Specification Part 10 – Ada Language Binding

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

ISO/IEC 8652:1995(E)  
with COR.1:2000

Ada95 Reference Manual  
Issue 1

ISO/IEC 9899:1999(E)

Programming Languages – C

ISO/IEC 14882:2003(E)

Programming Languages C++

SPARK\_LRM

The SPADE Ada Kernel (including RavenSPARK) Issue 7.3

SCA 1.1

OASIS Service Component Architecture Version 1.1

## 4 Definitions

For the purpose of this standard, the definitions given in Architecture Specification Part 2 apply.

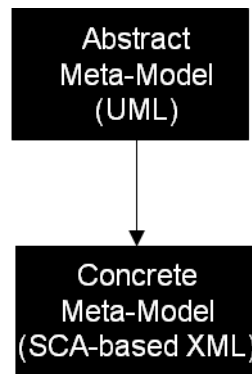
## 5 Abbreviations

API	Application Programming Interface
ARINC	Aeronautical Radio, Incorporated
ASCII	American Standard Code for Information Interchange
CPU	Central Processing Unit
ECOА	European Component Oriented Architecture. ECOА <sup>®</sup> is a registered trademark.
ELI	ECOА <sup>®</sup> Logical Interface
FIFO	First In, First Out
ID	Identifier
IT	Information Technology
OS	Operating System
PINFO	Persistent Information
QoS	Quality of Service
SOA	Service-oriented Architecture
TF	Time Frame
UML	Unified Modelling Language
XML	eXtensible Markup Language
XSD	XML Schema Definition

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 6 ECOA Meta Models



**Figure 1 ECOA Meta Models**

The structure of an ECOA system has been specified as an abstract metamodel which describes the ECOA system's data elements and their relationships. UML entity-relationship diagrams have been used to present the model information which can be found in Section 6.1.

Ultimately the requirements to exchange components, and automatically instantiate systems from them, require a precisely-specified and machine-readable version of the model. This is known as the concrete metamodel and the implementation is based on the open standard Service Component Architecture (SCA). Section 6.2 defines the concrete metamodel.

It is envisaged that the ECOA implementers will ultimately develop tool support that enables ECOA information to be captured in high-level design tools that support, for example, UML. However, the SCA-based concrete metamodel will remain the standard for exchange of information (e.g. between component suppliers and system integrators).

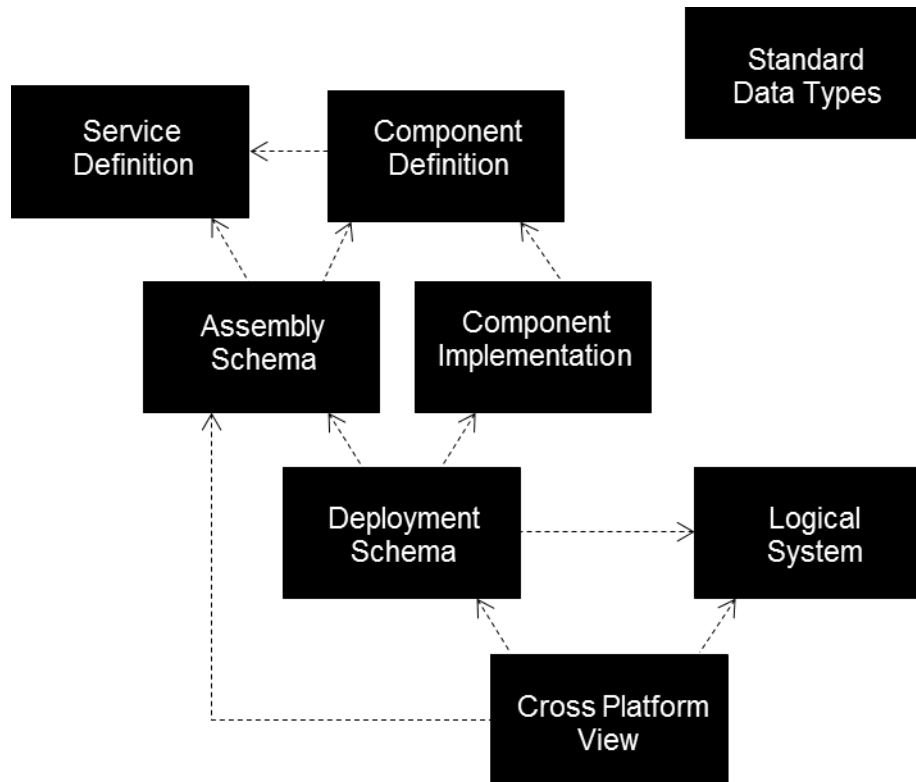
---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



## 6.1 Abstract Metamodel

### 6.1.1 Overview



**Figure 2 Overview of Metamodel**

The following sections detail the ECOA abstract metamodel. They aim to provide a definition of all the concepts and objects that need to be formalized to describe an ECOA system. This abstract metamodel is designed using a set of self-sufficient views of a UML model; each view of the metamodel is describing one given concept. An overview of this model is shown in Figure 2 which describes overarching dependencies between main ECOA concepts.

An Assembly Schema (see section 6.1.4) describes the structure of the ECOA system, or the structure of a Composite Component within the ECOA system, in a hardware independent fashion. It does this by describing service links between the Application Software Components, some of which may be regrouped into Composite Components. These are uni-directional connections between one service provided by an Application Software Component or Composite Component and another service required by an Application Software Component or Composite Component. Both services share the same full Service Definition and have a compatible Quality of Service (QoS). The service is offered or required as a whole, where a client has access to the whole set of service operations of the Service Definition.

An ECOA system is fully formalized within one or more Deployment Schemas (see section 6.1.8) which detail how the Application Software Components of the Assembly Schemas are deployed on a Logical System: it specifies how modules of each Application Software Component are mapped onto logical

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

processor nodes. A Logical System (see also section 6.1.8) describes a logical set of hardware computing resources and their connections.

A Cross Platform View (see section 6.1.9) describes ELI interactions between Composites of the ECOA system deployed onto different Logical Computing Platforms, in a way which is independent of the Assembly Schemas that describe those Composites. A Cross Platform View forms part of the ICD between ECOA Platforms.

The client defines an expected QoS for each required service (service-level and operation-level), and the provider defines an actual QoS for each provided service (service-level and operation-level). The expected and actual QoS need to be compatible for a service link to be established between both services. As a service is a collection of operations, a service link is, at technical level, implemented by a collection of module-level operation links between component modules. These operation links fulfil the rules of direction and multiplicity implied by the service link.

The Component Definition (Section 6.1.3) captures the interface of an Application Software Component and is formalized in terms of the services that are required and the services that are provided by an Application Software Component.

The interface of an Application Service, called Service Definition, is described as a set of operation signatures (see section 6.1.2).

Each Data Type (see section 6.1.7) used at the Service Definition or Component Definition level must be described.

A Component Instance is the software instantiation within the ECOA System of a given Component Implementation. A Component Implementation is the software realization of a Component Definition to which it conforms.

A Component Implementation (see section 6.1.5) is described within the ECOA System in terms of:

- Its Component Definition, which is (in SOA terms) the “Component Contract” to which it conforms,
- Its internal design, which comprises Modules, their Operations, and the Operation Links connecting them.

The concept of Module is defined (section 6.1.5) as a software entity implementing a given part of the ECOA Component Implementation. Operations in one Module may interact with Operations in another module via standard ECOA mechanisms.

Within a Component Implementation, Modules are linked together, at operation level, and are linked to the operations of the Services of the Component Implementation using Operation Links (see section 6.1.6). The Modules are the software entities that have to be deployed in the Deployment Schema.

The rules specified in Section 6.2.1.1 apply to the names for operations, component implementations and module implementations.

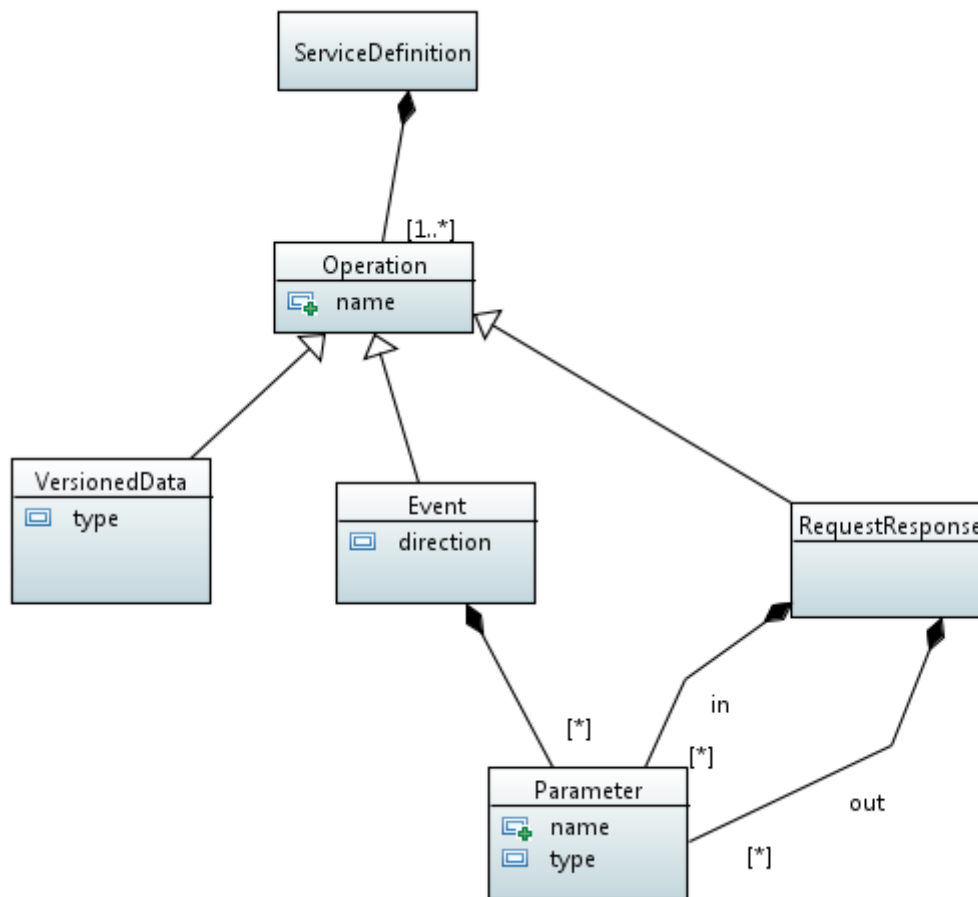
The abstract metamodel is described with UML class diagrams which use the following conventions:

- The default multiplicity for any link between two entities is 1.
- A grey-filled class indicates a reference to another class diagram in which the mentioned entity is described.
- An open arrow with a dot at the apex from one entity A to another entity B denotes a reference from A to B.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 6.1.2 Service Definition



**Figure 3 ServiceDefinition metamodel**

A **ServiceDefinition** is a set of **Operations** and **QoS Specifications**. An **Operation** is either, a **VersionedData Operation**, an **Event Operation** or a **RequestResponse Operation**. A ServiceDefinition contains at least one Operation.

Service Operation names shall be unique in a ServiceDefinition.

For a VersionedData Operation in a ServiceDefinition, the data is published by the Application Software Component that provides the service.

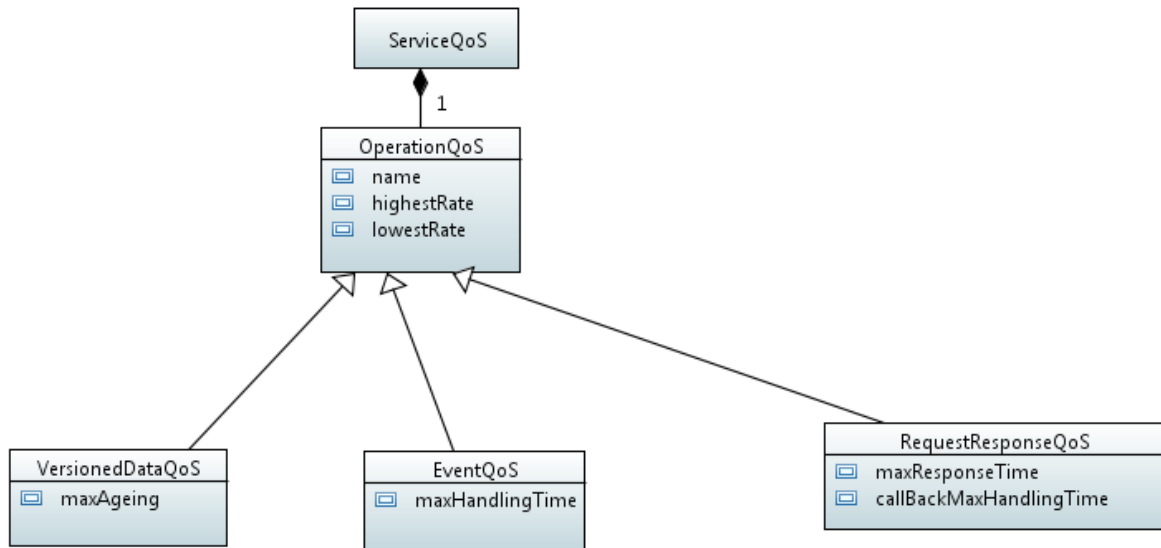
An Event Operation in a ServiceDefinition has a direction: either received\_by\_provider or sent\_by\_provider.

A RequestResponse Operation in a ServiceDefinition is initiated by the Application Software Component that requires the service.

Each Service Operation may be qualified by specific QoS attributes given hereafter. Unit is in seconds (s).

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



**Figure 4 ServiceQoS metamodel**

These attributes are requirements at component level and provide hypothesis for the internal design of the component. Internal design choices are under the responsibility of the Component Supplier while the Service Operation QoS attributes are initially defined by the System Designer.

**Table 1 Specific QoS attributes on operations**

	Provided service	Required service
<b>Event</b>	<b>IN (Received by Provider)</b> <b>MaxHandlingTime</b> = maximum duration between event receipt and end of related processing	
	<b>OUT (Sent by Provider)</b> <b>MaxHandlingTime</b> = specifies an intent on receivers for maximum duration between event receipt and end of related processing	
<b>Request-Response</b>	<b>IN (request_received)</b> <b>MaxResponseTime</b> = maximum duration between request receipt and response sent	<b>OUT (request_sent)</b> <b>MaxResponseTime</b> = maximum duration between request sent and response receipt In case of an asynchronous request-response, <b>callBackMaxHandlingTime</b> = maximum duration between response receipt and end of related processing

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

	Provided service	Required service
<b>Versioned Data</b>	<p><b>Published</b></p> <p><b>MaxAgeing</b> = maximum duration between data production (from the source) and the end of the writing process + (Worst MaxAgeing of all the consumed versioned data used for producing the data)</p>	<p><b>Consumed</b></p> <p><b>MaxAgeing</b> = maximum duration between data production (from the source) and the end of the reading process + (Worst MaxAgeing of all the consumed versioned data used for producing the data)</p> <p><b>NotificationMaxHandlingTime</b> = maximum duration between versioned data update notification receipt and end of related processing</p>

Data ageing will take into account all the transformations processed through the components chain by analysing components behaviours. In other words, data ageing will effectively be cumulative across a functional chain in order to make sense for endpoint consumers of the versioned data.

In addition to these attributes specific to each type of operation, two common QoS attributes are specified and applicable for each operation (independent of its type): the HighestRate and the LowestRate. They are based on the notion of slipping rate.

A slipping rate is defined with the help of two numbers:

- A time frame TF which is a time duration,
- A number N of operation calls which occur during the time frame.

The slipping rate expresses that, at any time, during the previous time frame (interval of time between the present time and the present time minus the time frame), exactly N operation calls have occurred.

As application of that notion, the HighestRate specifies the maximum number of occurrences of the operation within a specified time frame. If the number of occurrences is 1, the time frame corresponds to the minimum inter-arrival time between operations. The HighestRate helps to size input queues and to define processing activation laws within the component.

The LowestRate specifies the minimum number of occurrences of the operation within a specified time frame. If the number of occurrences is 1, the time frame corresponds to the maximum inter-arrival time between operations. The LowestRate helps to define the minimal processing activation laws within the component.

It is recommended to use the same time frame to express the HighestRate and the LowestRate to ease comparison between both.

For data, these rates express refreshment period requirements.

For an output event, an occurring rate can be defined to compare the receivers accepted input rates: this allows consistency checking between production and consumption rates.

For R/R replies, it is assumed that the R/R reply follows the same laws as the R/R request.

A comment can be added to describe each operation.

A single Module Operation may invoke many Container Operations during its execution, which introduces dependencies between the rates identified for those operations.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

All QoS attributes on operations are currently optional.

A ServiceDefinition also includes service-level specifications of Quality of Service (QoS) parameters (such as encryption level). These QoS specifications are used when matching up provided and required ServiceDefinition variants when creating ServiceLinks (section 6.1.6).

### 6.1.3 Component Definition

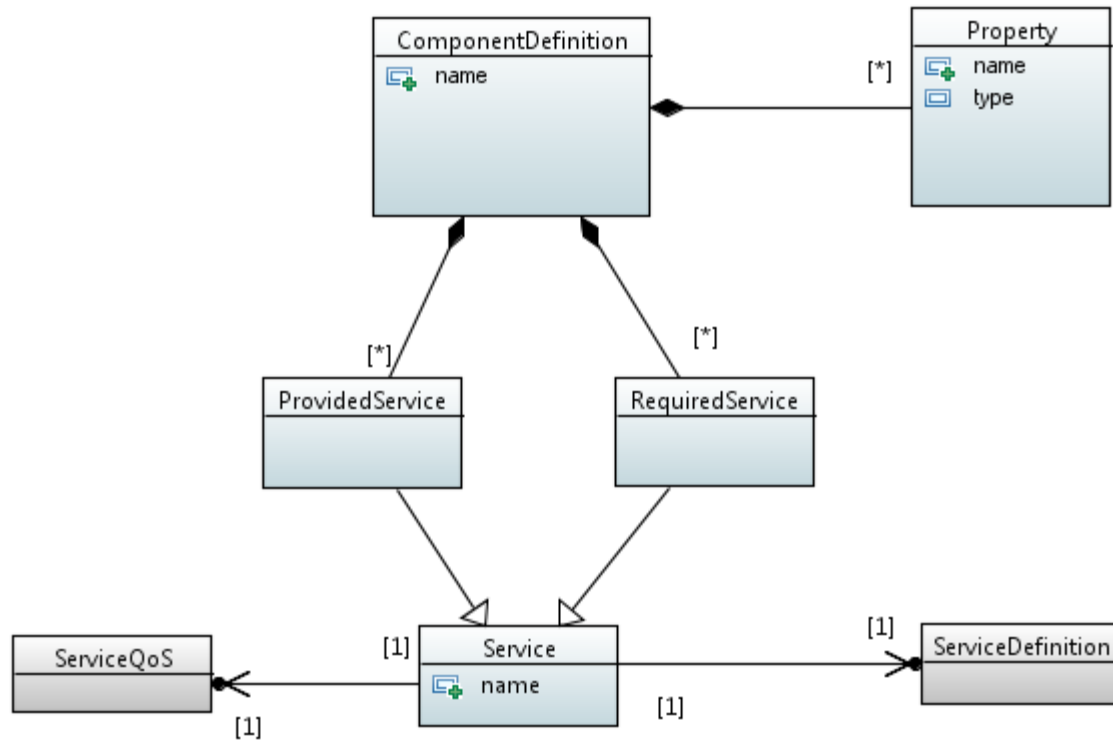


Figure 5 ComponentDefinition metamodel

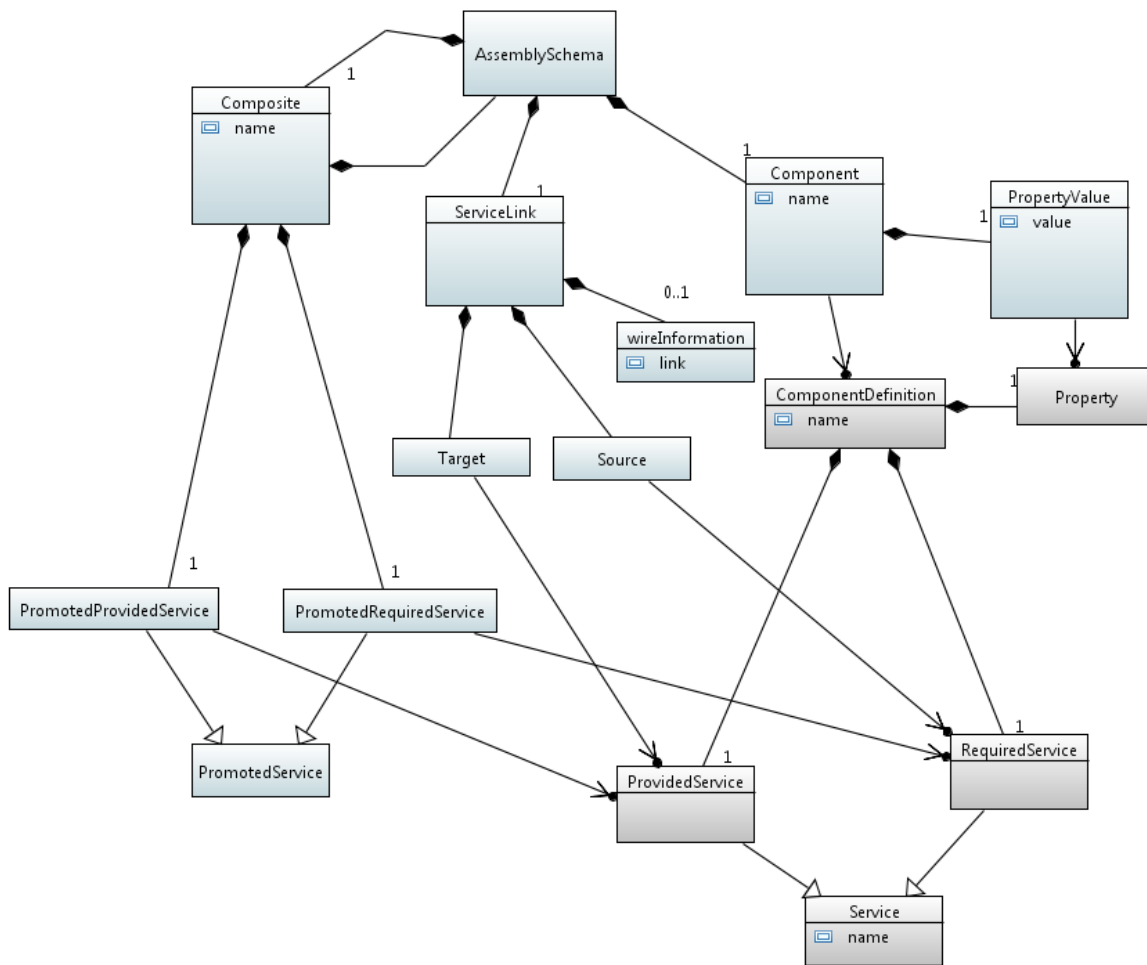
A **ComponentDefinition** is a set of **Properties**, **ProvidedServices** and **RequiredServices**. A ProvidedService or a RequiredService references a ServiceDefinition shared by ComponentDefinitions.

A ComponentDefinition must contain at least either a ProvidedService or a RequiredService.

A Property is declared in ComponentDefinition for each Public PINFO accessible by instances of that ComponentDefinition.

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 6.1.4 Assembly Schema



**Figure 6 AssemblySchema and ServiceLink metamodel**

An **AssemblySchema** describes the structure of an ECOA system, or the structure of a Composite Component within the ECOA system, independently of its physical deployment on hardware platforms. Note that grey in figure above means that the greyed element is more precisely defined in another subsection. Different Assembly Schemas may be used in order to describe different viewpoints on the ECOA system. It is possible to create “flat” Assembly Schemas which only contain ASCs, or hierarchical Assembly Schemas which describe Composite Components, those Composite Components being described by their own Assembly Schema.

An Assembly is made of **Composites**, **Components** and **ServiceLinks**.

A Composite looks like a component; its definition is provided by a dedicated AssemblySchema and promotion links between its provided or required services and the provided or required services of its internal components. Promotion links are a logical concept to master complexity through hierarchical assembly schemas. They have no existence at IT or technical levels: the assembly schema actually deployed is the one containing only components.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

A Component instantiates a ComponentDefinition in a given system. It has a set of instantiation parameters known as Properties defined at ComponentDefinition level.

A ServiceLink connects Components together via provided and required service references. Each ServiceLink connects one ProvidedServiceReference (target) to one RequiredServiceReference (source) each of which refers to a Component and to a RequiredService or ProvidedService of this component's ComponentDefinition (targets and sources are "technical" objects which are introduced to model ternary associations, without introducing the specific UML notation).

External information may be associated to a ServiceLink through the sub-element **wireInformation** and its attribute **link** which identifies an external resource (file, etc.).

Each Public PINFO Filename Association is declared in the AssemblySchema through a Property value associated with a filename. This file provides the Public PINFO data. The filename includes the relative path to the Pinfo sub-directory of "5-Integration" directory.

Each Component in the AssemblySchema resolves each PINFO-related Property of its ComponentDefinition, by referencing the corresponding AssemblySchema level Property.

### 6.1.5 Component Implementation

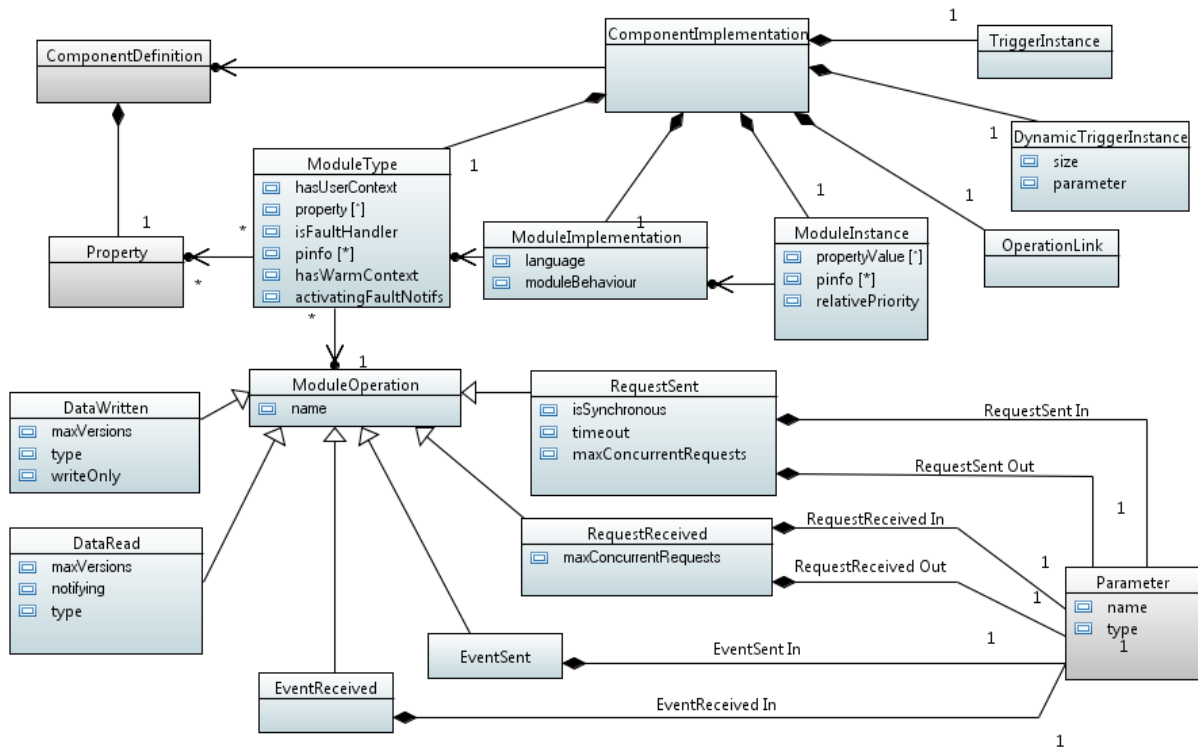


Figure 7 ComponentImplementation metamodel

A software realization of a ComponentDefinition is described by a **ComponentImplementation**.

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



A ComponentImplementation gives information to secure the integration of Application Software Components in a system, possibly sharing execution platforms, and to make early verification of the system possible.

A ComponentImplementation is made of **ModuleTypes**, **ModuleInstances**, **TriggerInstances**, **DynamicTriggerInstances** and **OperationLinks**.

A ModuleImplementation corresponds to a piece of software implementing in a certain programming language a given part of the Component Implementation that must be executable in a single thread (no parallelism, no internal synchronisation). A ModuleInstance corresponds to the instantiation of a given **ModuleImplementation**.

A ModuleType defines the interface of a ModuleImplementation in terms of ModuleOperations at module-level or in terms of properties. These operations correspond to the same exchange mechanisms used by Application Software Components (i.e. data, event; request-response), but have a direction (reflecting the module's point of view), and have additional attributes over and above the service defined ones.

Each DataRead operation that indicates the module is a reader of a versioned data relies on the following attributes:

- **maxVersions**: the maximum number of versions that the module may access in read mode,
- **notifying**: if 'true', a callback is generated by the platform tooling and is called by the container each time it is aware of a data update.

Each RequestReceived operation which is the entry-point called on the receiving of a request relies on the following attribute:

- **maxConcurrentRequests**: the maximum number of R/R IDs that the module may retain for that entry-point before sending the associated replies. If this number is reached, additional R/R cannot be retained and are discarded by the Container.

Each RequestSent operation which indicates the module may send a request towards another module relies on the following attributes

- **isSynchronous**: if true, the R/R is synchronous: the call blocks the calling module until the receiving of the response or the expiration of the timeout.
- **timeout**: the maximum time spent waiting for a reply  
When **isSynchronous** is true, it is the maximum time during which the module is blocked waiting for a reply. If the timeout is set to -1, the R/R is an indefinite blocking call.  
When **isSynchronous** is false, it is the maximum time before the Container sends a NO\_RESPONSE to the module. If the timeout is set to -1, the Container will only send a response to the module if there is a reply.

Each DataWritten operation that indicates the module is the writer of a given versioned data relies on the following attributes:

- **maxVersions**: the maximum of versions that the module may access in read-write mode.
- **writeOnly**: the possibility for a writer of a versioned data to have a "writeOnly" copy of the data, thereby avoiding the platform to initialize the local copy of the data with the current value.

The operations' names will appear in the module's container API. Each operation name shall be unique for a given moduleType definition. By annotating a module with the attribute **IsFaultHandler**, the module is considered as a Fault Handler and it may support fault management capabilities such as infrastructure error notifications or recovery actions, provided that the ECOA Platform supports Fault Handlers being implemented as ASCs (see Architecture Specification Part 5).

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

Attributes **hasUserContext**, **hasWarmContext** and **isFaultHandler** associated to a ModuleType define if the associated ModuleImplementation may use or not a User Context or a Warm Context and if the Module Instance can be an ECOA Fault Handler.

A ModuleInstance corresponds to an instance of a ModuleImplementation which itself is of a defined ModuleType. A ModuleInstance has its own internal state. The container activates the Module Instance as long as there are incoming activating operations for it.

The notion of ModuleInstance provides the ability to instantiate, a number of times, the same software code in multiple execution contexts (e.g. different execution node) inside an ECOA Application Software Component.

A TriggerInstance is similar to a ModuleInstance, except that it is dedicated to producing periodic events: it has no module type; it does not need to be implemented, as the periodic events will be generated automatically by the infrastructure.

A DynamicTriggerInstance is a trigger that generates non periodic events. The delay between the generation of two events can be dynamically set at runtime. As for the TriggerInstance, the DynamicTriggerInstance is generated by the infrastructure.

A relativePriority attribute is specified for each Module/Trigger/Dynamic Trigger Instance to allow a Component Implementer to define the relative priorities of Module/Trigger/Dynamic Trigger Instances within their implementation. This may be used by the System Integrator to guide the analysis and deployment of a Component Instance into a system.

Each Property at ComponentDefinition level may be referenced by one or more ModuleTypes of the ComponentImplementation.

The internal structure of a given ComponentImplementation must be specified in terms of OperationLinks.

A moduleType may contain an optional **pinfo** item for declaring PINFO accessible to instances of that moduleType. When being defined, the **pinfo** item is made of several **publicPinfo** and **privatePinfo** sub-items:

- Each publicPinfo and privatePinfo has a name.
- There is one publicPinfo attribute defined per Public PINFO to be accessed by instances of that moduleType.
- There is one privatePinfo attribute defined per Private PINFO to be accessed by instances of that moduleType.

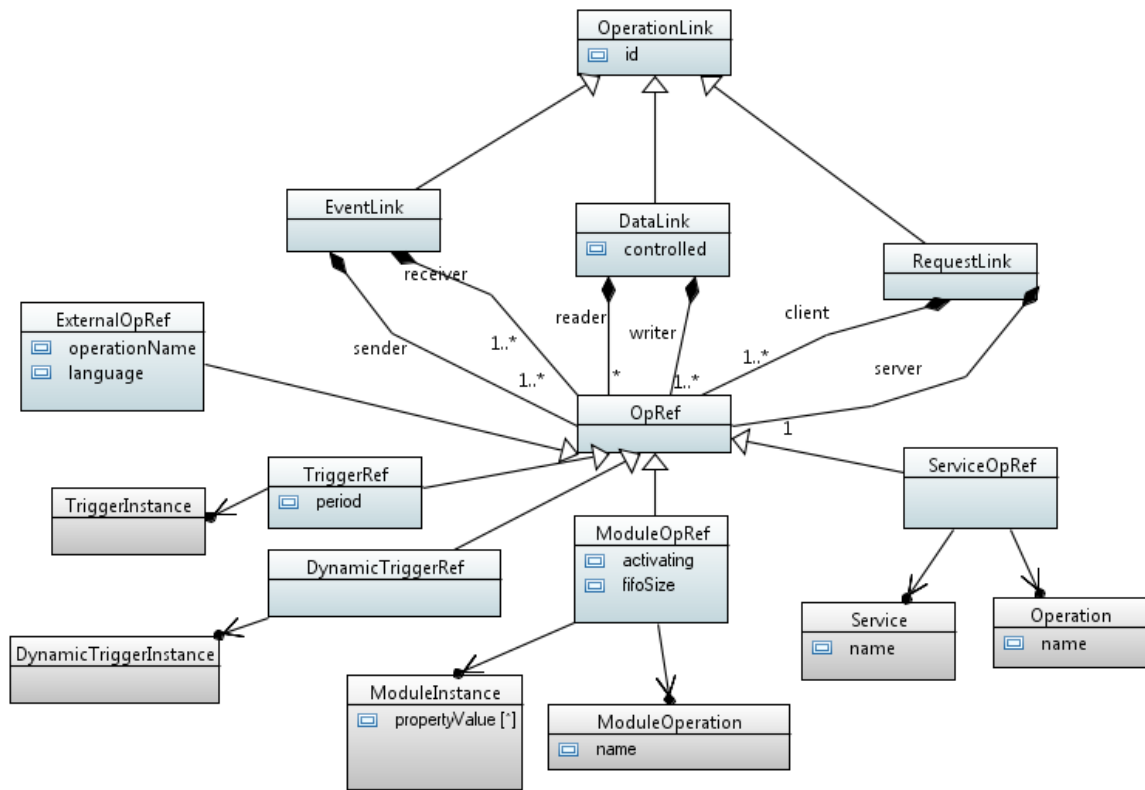
A moduleInstance resolves publicPinfo and privatePinfo declared at moduleType level as follows:

- Each publicPinfo is associated with one of the Properties declared at ComponentDefinition for Public PINFO.
- Each privatePinfo is associated with a filename. This file provides the Private PINFO data. The filename includes the relative path to the Pinfo sub-directory of the <name\_of\_component\_implementation> directory.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 6.1.6 Operation Links



**Figure 8 OperationLink metamodel**

The **OperationLinks** describe the interactions/synchronisations within the same Application Software Component. An OperationLink is a “star-like” connection linking “internal” (module-level) operations, service operations, Triggers and Dynamic Triggers. Inter-Module interactions are specified using **DataLink**, **EventLink** or **RequestLink**, depending on the kind of module operations that are linked. These three kinds of links are oriented and have different possible multiplicities:

- A DataLink may have n writers and p readers (a unique data-writer is recommended but not mandatory),
- A RequestLink has one server and p possible clients,
- An EventLink can have n possible senders and p possible receivers of the event.

An internal module operation is referred to by a **ModuleOpRef**, which refers to a ModuleInstance and one of its ModuleOperations. Attributes are associated to a ModuleOpRef to define:

- if the operation is activating or not (activating),
- and the maximum number of waiting operation calls for this operation (fifoSize) – waiting means the operation calls have not been taken out of the infrastructure.

Internally to the component, an operation of the service is referred to by a **ServiceOpRef**, which refers to a ProvidedService or RequiredService, and one of the Operations of its ServiceDefinition.

The purpose of the TriggerInstance is to define a periodic event generator internally to the component scope: the TriggerInstance will act as the sender of the event, at the specified period. The generator is

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

handled by the container (e.g. an OS watchdog or an auto-generated invisible module which sends an event). This avoids the creation of event generation components which will break the inversion of control principle, as they will need to access to the OS to generate periodic events. This system allows the creation of several flows of periodic events in a synchronised way (if all events come from the same TriggerInstance), or in a non-synchronised way (if they come from different TriggerInstances). It also allows combining a periodic source of events with other, non-periodic sources.

The purpose of the DynamicTriggerInstance is to define a one-shot event generator internally to the component scope: the DynamicTriggerInstance will act as a sender of a valued event, within a given delay specified at runtime. The principle is to receive an event, named “in” event hereafter, and to send an associated event, named “out” event at a specific time. The first parameter of the “in” event is the delay. Other parameters can be any of the ECOA types. Multiple occurrences of the same event can be queued waiting for the delays to expire. A « reset » operation can purge all waiting event occurrences.

The same EventSent operation of a module can be part of different ModuleOperationLinks at the same time. All other module operations (RequestReceived, EventReceived, DataRead, DataWritten and RequestSent operations) can only be in one OperationLink.

Each DataLink is associated to a Data that it represents and that is shared within the Application Software Component. Each DataLink has an attribute, **controlled**, which indicates if concurrent access to the Data is controlled by the ECOA Infrastructure or not.

When a DataLink is tagged as controlled, it may connect several writers that can be component-internal writes or references, as well as several readers that can be component-internal reads or services. The ECOA Infrastructure ensures that any writer or reader can access to a local copy of the Data, this copy being the most recent value accessible on the platform among those produced by writers declared in the DataLink.

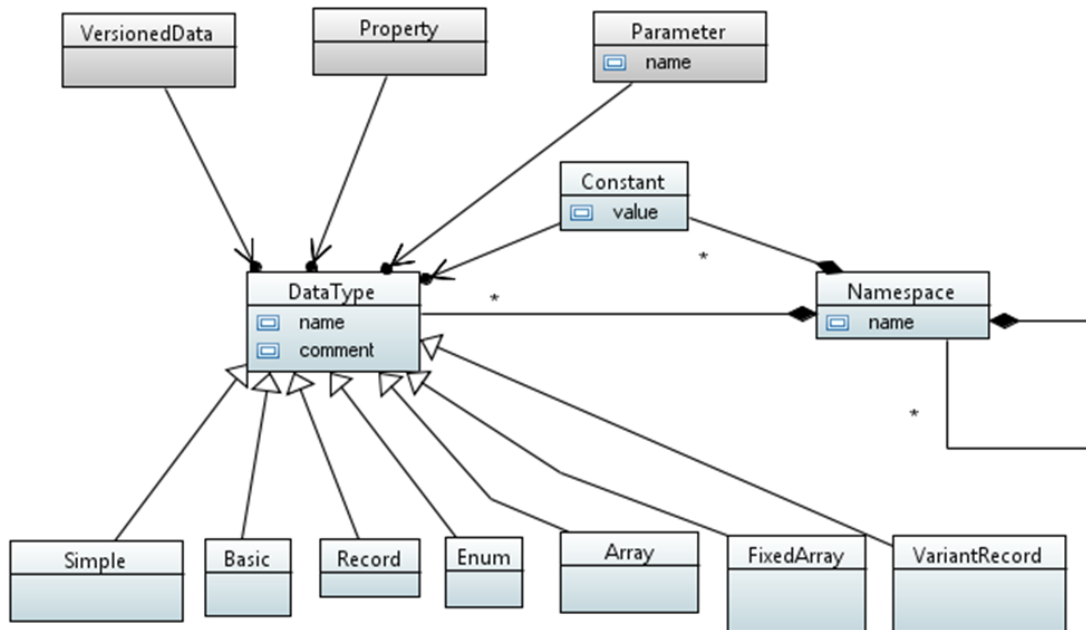
When a DataLink is tagged as not controlled, it may connect several writers that can only be component-internal writes, as well as several readers that can only be component-internal reads. The local data repository is accessed directly by all readers and writers declared in the Datalink, thereby concurrent access between Module Instances must be managed at application level under the responsibility of the Component Supplier.

### 6.1.7 Data Types

Data types are “portable types” and are only used to describe information transmitted on wires between components and operation links between modules. By using these types, information can be then serialized for transmission with the help of the ELI (Architecture Specification Part 6). The way they are physically bound to a given processor is left to the platform provider based on language bindings (Architecture Specification Part 4).

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



**Figure 9 DataType definition**

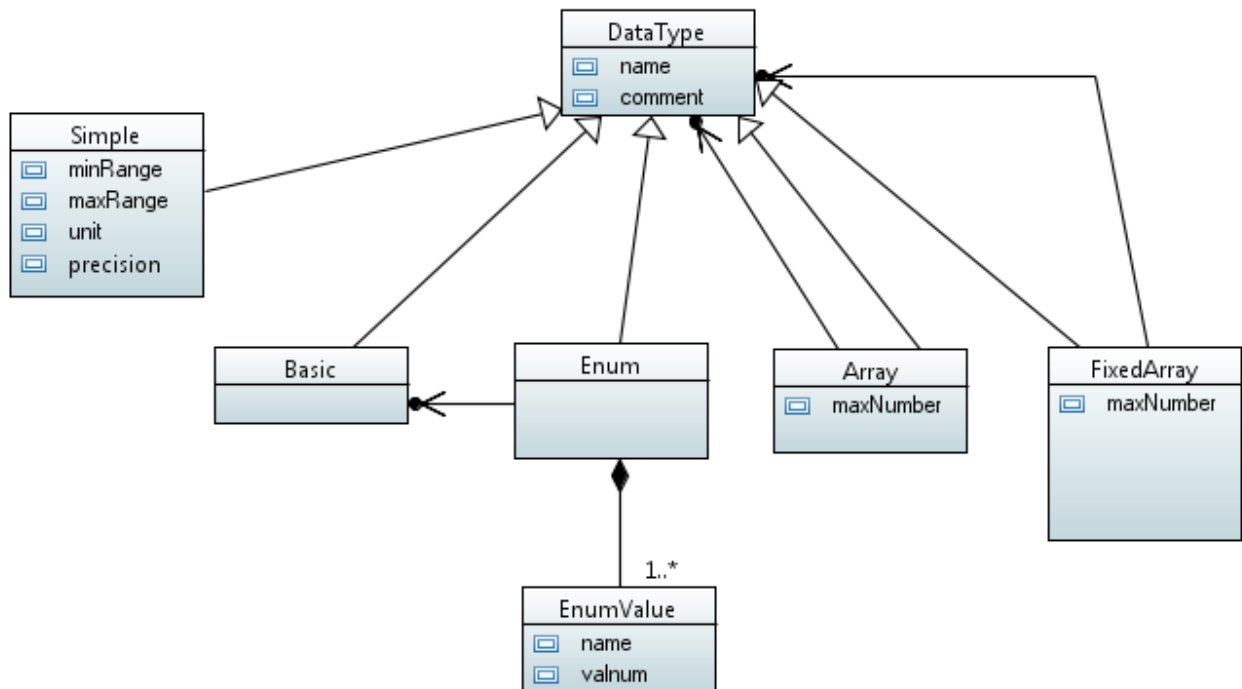
A **DataType** is a language-neutral type definition. It is used as a shared definition, to help define ServiceDefinitions: it is referenced by VersionedData, and by Parameters of Events and RequestResponses. It is also used to type Properties. A comment can describe the DataType.

A DataType definition describes the **NameSpace** in which it is located. A NameSpace is composed of DataTypes and NameSpaces, the different types are described in Figure 10 and Figure 11.

A **Constant** is a remarkable integer, floating-point or single-character value, identified with a name and located in a given NameSpace. A Constant may be of any of the Simple or Basic DataTypes except boolean8 Basic type and Simple types based on boolean8.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



**Figure 10 Supported Data Types**

A **Basic** type belongs to a fixed list of predefined types<sup>1</sup>: boolean8, char8, byte, int8, int16, int32, int64<sup>2</sup>, uint8, uint16, uint32, uint64, float32, double64. 8-bit characters are encoded in ASCII. For boolean8, the value 1 means TRUE while the value 0 FALSE. float32 and double64 are IEEE754 compatible.

A **Simple** type is defined to give a meaningful name to a basic or another simple type. It can define range limits, a precision and a unit. Each limit can be a literal numeric, a single-character or a reference to a symbolic constant. The unit is functional and expressed as a string (e.g. 'second'). The precision is functional and expressed as a literal numeric. The precision is functionally consistent with the unit.

An **Enum** type shall be a basic integer type, or a simple type derived from a basic integer type and defines the list of authorized values, **EnumValue**, each of which has a symbolic name. Each value can be a literal numeric, or a reference to a symbolic constant.

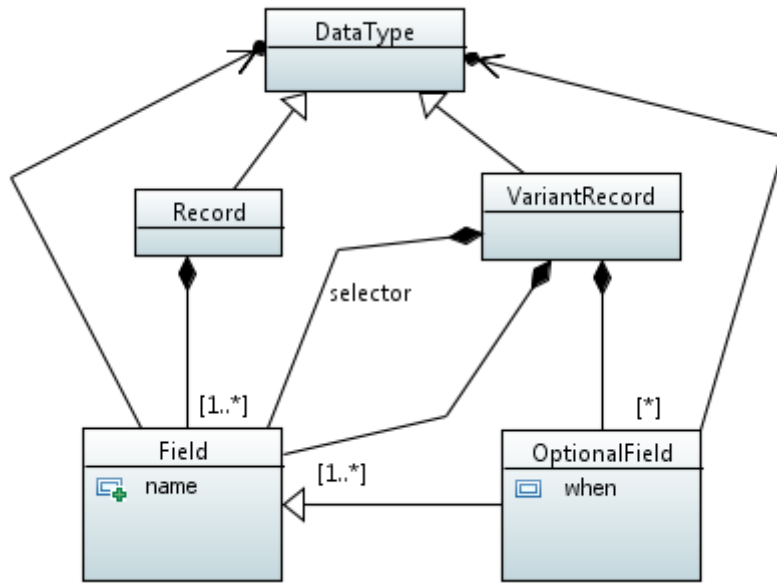
An **Array** defines a variable-capacity array, whose maximum capacity is fixed. All elements are of the same type. The maximum capacity can be a literal numeric, or a reference to a symbolic constant.

A **FixedArray** defines a fixed-capacity array. All elements are of the same type. The capacity can be a literal numeric, or a reference to a symbolic constant.

<sup>1</sup> The list of available types may be extended in the future as requirements evolve. For example, fixed-point types may be required.

<sup>2</sup> 64bit types may not be supported on every ECOA platform.

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



**Figure 11 Records and VariantRecords**

A **Record** is a structure with named **Fields**, of any type.

A **VariantRecord** is like a Record, with a special field called the selector (of boolean, integer or enum type). Some of the fields, **OptionalField**, of a VariantRecord are optional: they are valued only when the selector has a certain value (given by the attribute “when”).

Nested types are not allowed; i.e. it is not possible to define local types specific to a given field. All types used at field level must be defined prior to the record definition.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 6.1.8 Deployment Schema and Logical System

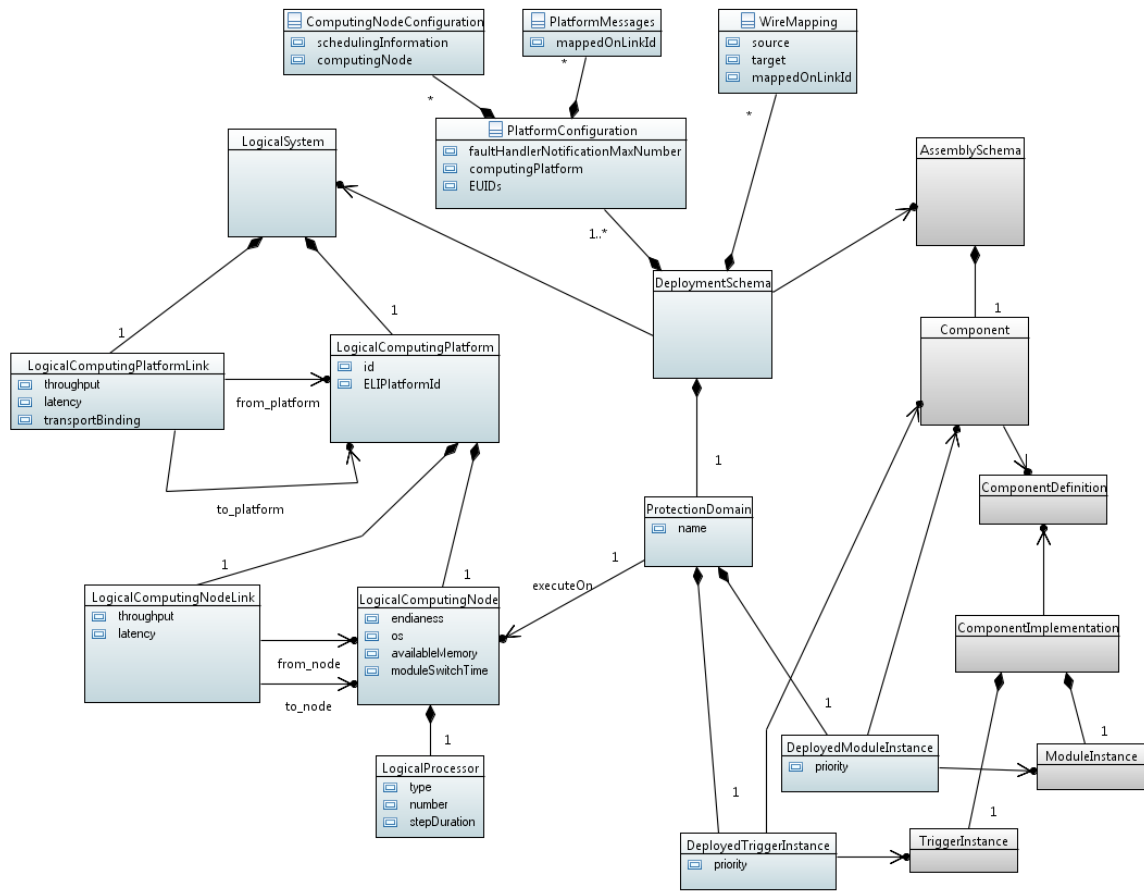


Figure 12 Deployment Schema

A **Deployment Schema** refers to an **Assembly Schema**.

It contains the mapping of **ProtectionDomains** on **LogicalComputingNodes**.

A **ProtectionDomain** offers spatial isolation (memory protection), and possibly also temporal isolation (e.g. ARINC 653 partition scheduling), on a given **LogicalComputingNode**. It corresponds to the concept of process or partition, depending on the OS used.

Each **ProtectionDomain** hosts a number of **ModuleInstances** (which are referenced by the **DeployedModuleInstance** objects).

ECO defines the **Priority** in **DeployedModule/DeployedTriggerInstances**, which is calculated based on **relativePriority** of **ModuleInstance/TriggerInstance/DynamicTriggerInstance** and the operation rate (taking account of the response times of operations).

A **LogicalComputingNode** allows early verification of the performance of a system by providing an idealised model of a set of processors. This ideal processing resource is parameterised by a number of key attributes such as computing step, memory capacity, module switch time and number/standard of processors.

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



For each LogicalComputingNode, the faultHandlerNotificationMaxNumber attribute is the number of notifications (whatever their type among those listed hereafter) that a Module Container shall be able to handle at any time. The notification types being considered are fault handler notifications. This notification type is relevant depending on the Module “isFaultHandler” attribute.

The initial model of a logical computing node chosen at this stage of the architecture definition is as a symmetric multiprocessor hosting one single OS image. It contains sets of **LogicalProcessors**; these may be heterogeneous if the OS provides an abstract interface. LogicalComputingNodes are linked together through **LogicalComputingNodeLinks** and they constitute a **LogicalComputingPlatform**. LogicalComputingPlatforms may then be linked together through **LogicalComputingPlatformLinks** and they constitute a **LogicalSystem**.

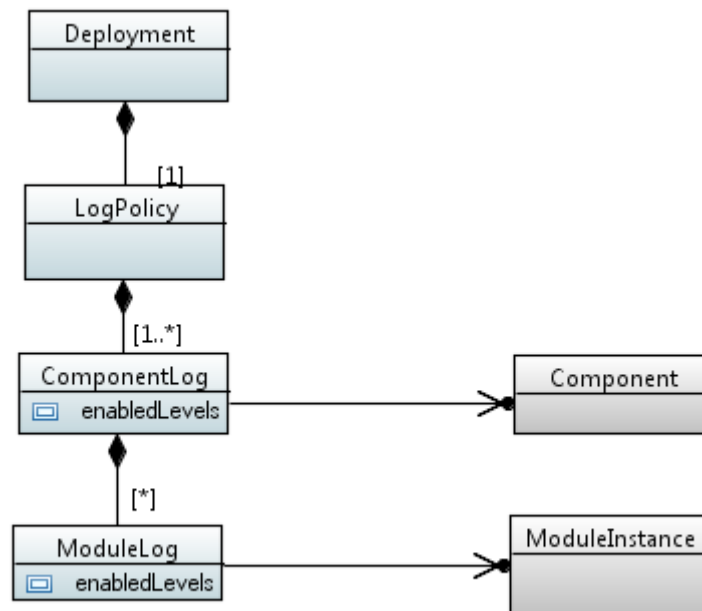
The **ID** attribute of the **PlatformConfiguration** allows declaring the name of the file which contains IDs to be used for sending ELI messages over the LogicalComputingPlatformLinks.

The mapping of component level wires onto LogicalComputingPlatformLinks (i.e. for communications between ECOA Platforms) and the mapping of component level wires onto LogicalComputingNodeLinks (i.e. for communications within an ECOA Platform) are declared in the Deployment Schema.

The mapping of platform management messages onto LogicalComputingPlatformLinks is declared in the Deployment Schema.

The transport binding (ex : UDP binding) associated to each LogicalComputingPlatformLink is declared in the Logical System.

The mapping from logical computing nodes to actual physical processors (or cores) is not defined. It may be that it is not the same as the mapping from LogicalComputingNodes to LogicalProcessors. For example one physical processor or core might be used instead of multiple LogicalProcessors or vice-versa. The mapping of logical links onto actual physical buses is not also addressed. These mappings are provided through specific artefacts supplied by the platform provider.



**Figure 13 Log Policy Definition**

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

### 6.1.9 Cross Platform View

A **Cross Platform View** describes interactions between Composites within an ECOA system that are deployed onto multiple ECOA platforms. A Cross Platform View references an **Assembly Schema** which describes the assembly of these Composites which interact via ServiceLinks. Each of these Composites being the abstraction of an Assembly Schema which will be fully deployed onto one of the Logical Computing Platforms defined in the Logical System. ServiceDefinitions are referenced by ProvidedServices and RequiredServices of Composites in the Assembly Schema referenced by the Cross Platform View.

There may be several Cross Platform Views within an ECOA System depending on the level of visibility required in each part of the system.

A Cross Platform View describes the mapping of the Composites of the referenced Assembly Schema onto Logical Computing Platforms of the Logical System.

A Cross Platform View describes the mapping between the ServiceLinks featured in the referenced Assembly Schema and the LogicalComputingPlatformLinks described in the Logical System.

A Cross Platform View describes the binding between the LogicalComputingPlatformLinks described in the Logical System and resource files containing ELI Service Operation Identifiers (used within the ELI messages, as specified in Architecture Specification Part 6).

## 6.2 Concrete Metamodel

### 6.2.1 Mapping onto Service-Component Architecture (SCA)

The Service Component Architecture (SCA) is a standardised model for building applications and (software) systems using a Service-Oriented Architecture (SOA), developed by a set of industry partners. Initially developed as an industrial collaboration, this open standard is now reaching maturity and is maintained by the OASIS (<http://www.oasis-open.org/>) organisation. Using some SCA concepts and implementations avoids unnecessary re-implementation and potentially leverages existing tool support.

This section describes the translation of the abstract metamodel described in the previous section onto an XML metamodel based on the SCA assembly model. In fact, the XML metamodel re-describes all ECOA artefacts already described by the abstract metamodel but in a way usable by software tooling.

#### 6.2.1.1 Rules on XML writing

Certain rules need to be followed to ensure that the XML is consistent and correct. The following rules may be checked through normal validation requirements of the XML relative to its XSD schema or by specific tooling.

- Information names used within XML files are case sensitive. If the name of one item is used many times, character strings used for that name shall use the same case sensitivity.
- The parsing of XML files is done in one pass; i.e. items need to be defined before they are used. For example, the type for a field in a structure shall be defined before the definition of the structure.
- Each component implementation name must be unique within the assembly schema.
- Each component instance name must be unique within the assembly schema.
- Each module instance name must be unique within the component implementation.
- Each module implementation name must be unique within the protection domain hosting it.
- Each operation name must be unique within each module definition.
- Each operation name must be unique within each service definition.
- Each parameter name must be unique within each operation definition.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

- Operation, operation parameter and module names must follow the naming conventions for identifiers used in the most common programming languages: a name being a sequence of letters, figures and underscores, beginning with a letter. Typically, “-“ must not be used in ECOA names.
- Operation, operation parameter and module names must not clash with reserved words of the target language.
- The order of Operation Parameters in the Component Implementation must match the order declared in the Service Definition.

### 6.2.1.2 XPath Syntax

The syntax used to identify an element is the XPath one. XPath uses path expressions to select nodes in an XML document. The node is selected by following a path or steps. The most useful path expressions are listed in Table 2.

**Table 2 XPath Expressions**

Expression	Description
Nodename	Selects all child nodes of the named node
@	Selects attributes of the current node

NOTE: More information can be found at [http://www.w3schools.com/xpath/xpath\\_syntax.asp](http://www.w3schools.com/xpath/xpath_syntax.asp).

Nodename in XPath should be a NCName (a name which does not contain colon character) or a QName (prefix:localName where prefix is defined as a reference to a namespace elsewhere).

To avoid a prefix definition we add a new syntax: {namespace}localName where 'namespace' is equal to 'ecoa-sca' in Table 3.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

### 6.2.1.3 ECOA to SCA Mapping

**Table 3 Relations between the ECOA abstract metamodel and the SCA Assembly model**

ECOA Abstract item	SCA item
ServiceDefinition and service-level operations	Interface extension See Section 8.9 (ecoa-interface-2.0.xsd)
ComponentDefinition	componentType See Section 8.18 (sca-core-1.1-cd06-subset-2.0.xsd)
Property	Property
Property / @name	property/@name
Property / @type	property/{ecoa-sca}type property/@type="xsd:string" <sup>3</sup>
ProvidedService	Service
ProvidedService/@name	service/@name
ProvidedService / @ServiceDefinition	service/{ecoa-sca}interface/@syntax
RequiredService	Reference
RequiredService / @name	reference/@name
RequiredService / @ServiceDefinition	reference / {ecoa-sca}interface / @syntax
ComponentImplementation, module artefacts, module-level operations and promotion links	Implementation extension See Section 8.8 (ecoa-implementation-2.0.xsd)
Component	Component
Component / @name	component/@name
Component / @ComponentDefinitionRef	component/{ecoa-sca}instance/ @componentType
Component / @ComponentImplementationRef	component/{ecoa-sca}instance/ {ecoa-sca}implementation/@path
ServiceLink	Wire
ServiceLink / @ProvidedServiceRef	wire / @target
ServiceLink / @RequiredServiceRef	wire / @source
Composite	Composite
AssemblySchema	Composite See Section 8.4 (ecoa-sca-instance-2.0.xsd)
DeploymentSchema	Refinement of a composite. Computing nodes are described in separate XML files.

<sup>3</sup> SCA uses xsd types to type component properties but ECOA has defined its own data typing model. As the assembly schema associates ECOA concepts to SCA concepts, the way to associate the ECOA type of the property to the SCA type is to use systematically the generic "xsd:string".

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

ECO Abstract item	SCA item
Data types	Specific description See Section 8.12 (ecoa-types-2.0.xsd)

## 6.2.2 Schemas

The ECOA concrete metamodel references the following files produced by the OASIS organisation. Currently the ECOA metamodel is defined against version 1.1 of the SCA ([SCA 1.1]).

SCA (sca- 1.1-cd06.xsd) which is available from:

<http://docs.oasis-open.org/opencsa/sca-assembly/sca-1.1-cd06.xsd>

SCA core (sca-core-1.1-cd06.xsd) which is available from:

<http://docs.oasis-open.org/opencsa/sca-assembly/sca-core-1.1-cd06.xsd>

SCA contributions (sca-contribution-1.1-cd06.xsd) which is available from:

<http://docs.oasis-open.org/opencsa/sca-assembly/sca-contribution-1.1-cd06.xsd>

The ECOA metamodel only refers to a subset of the SCA concepts (see 6.2.1.3). the section describes in extension the selected subset. It is so possible to comment out unused XSD entries in SCA schemas to validate the ECOA XML files. Filtered versions of SCA schemas are available; their names are suffixed with '-subset'.

Table 4 summarizes the ECOA schemas, which are presented in full in Section 8.

**Table 4 ECOA Defined Schemas**

Filename	Description	Section
ecoa-sca-2.0.xsd	Required for compatibility with SCA. Defines SCA extension schemas.	8.1
ecoa-sca-attributes-2.0.xsd	Required for compatibility with SCA. Defines SCA attribute extensions.	8.2
ecoa-sca-interface-2.0.xsd	Describes reference to service definition at component level within the assembly	8.3
ecoa-sca-instance-2.0.xsd	Describes reference to component implementation description at component level within the assembly	8.4
ecoa-bin-desc-2.0.xsd	Defines the links between module implementations and binary objects.	8.5
ecoa-common-2.0.xsd	Declares the use of a library of data types.	8.6
ecoa-deployment-2.0.xsd	Defines how Modules are mapped onto a logical architecture (ie. protection domains and processing nodes)	8.7
ecoa-implementation-2.0.xsd	Describes all the information needed to integrate the software implementation of an ECOA component in an ECOA system.	8.8

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

Filename	Description	Section
ecoa-interface-2.0.xsd	Describes an ECOA service, including a set of operations.	8.9
ecoa-interface-qos-2.0.xsd	Describes the provided and required quality of service associated with a component definition.	8.10
ecoa-logicalsyste-2.0.xsd	Describes a logical computing architecture consisting of computing nodes and protection domains connected by a network. This architecture description is intended to support early verification.	8.11
ecoa-types-2.0.xsd	Describes the syntax for defining ECOA types constructed from the ECOA basic types.	8.12
ecoa-project-2.0.xsd	Describes directories used for one given ECOA application	8.13
ecoa-cross-platforms-view-2.0.xsd	Describes the cross platform view	8.14
ecoa-uid-2.0.xsd	Describes associations between IDs and wires.	8.15
sca-1.1-cd06-subset-2.0.xsd	Selected subset of Service Component Architecture Schema Version 1.1	8.16
sca-contribution-1.1-cd06-subset-2.0.xsd	Selected subset of Service Component Architecture Contribution Schema Version 1.1	8.17
sca-core-1.1-cd06-subset-2.0.xsd	Selected subset of Service Component Architecture Core Schema Version 1.1	8.18
sca-implementation-composite-1.1-cd06-subset-2.0.xsd	Selected subset of Service Component Architecture Core Schema Version 1.1	8.19

ECOA Schemas and XML files are fully compliant with the W3C XML Standards. They are validated with the following files:

- XMLSchema.xsd that describes the Schema for XML Schemas. Origin of the file used for the purpose of ECOA: <http://www.w3c.org/2001/XMLSchema.xsd>
- xml.xsd that describes the XML namespace, in a form suitable for import by other schema documents. Origin of the file used for the purpose of ECOA: <http://www.w3c.org/2001/xml.xsd>

### 6.2.3 Filename Conventions

Table 5 specifies standard filenames for the different instances of the ECOA concrete metamodel. It also defines the main XSD file associated to the kind of file and it can be used as an entry point within the concrete metamodel.

**Table 5 ECOA Standard Filenames**

	ECOA Standard Filename	Comments	XSD
Project definition	#name#.project.xml	General information about one ECOA application - usage is optional	ecoa-project-2.0.xsd

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

	<b>ECO Standard Filename</b>	<b>Comments</b>	<b>XSD</b>
Type definitions	#name#.types.xml	Data types used by operations within service or module definitions	ecoa-types-2.0.xsd
Service definition	#name#.interface.xml	List of service operations Name required for conformance to SCA	ecoa-interface-2.0.xsd
Component definition	#name#.componentType	List of services provided and required by the component and its properties. Name required for conformance to SCA	sca-1.1-cd06-subset-2.0.xsd
Service QoS definition	#name#.interface.qos.xml	Service operation-level QoS	ecoa-interface-qos-2.0.xsd
Component implementation	#name#.impl.xml	Description of component architecture: modules, triggers, module operation links, etc	ecoa-implementation-2.0.xsd
Initial assembly schema	#name#.composite	Application architecture connecting component instances through wires. Decorrelated from any implementation Name required for conformance to SCA	sca-1.1-cd06-subset-2.0.xsd
Final assembly schema	#name#.impl.composite	This file adds to #name#.composite pointers to component implementations. Name required for conformance to SCA	sca-1.1-cd06-subset-2.0.xsd
Deployment schema	#name#.deployment.xml	Mapping of modules onto computing nodes	ecoa-deployment-2.0.xsd
Mapping onto binary files	#name#.bin-desc.xml bin-desc.xml	Mapping of logical module implementation names onto actual physical binary files Useful for packaging	ecoa-bin-desc-2.0.xsd
Logical System	#name#.logical-system.xml	Description of the computing platforms: computing nodes, links between them and performance characteristics.	ecoa-logicalsistem-2.0.xsd
ID associations	#name#.ids.xml	Associations between IDs and wires.	ecoa-uid-2.0.xsd

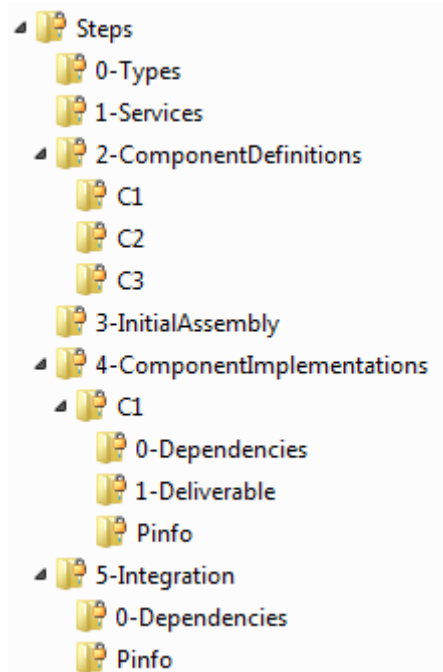
Logical names #name# used in filenames shall only contain characters a-z, A-Z, 0-9 and ‘\_’ since they are mapped onto software languages (see binding parts).

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 6.2.4 Interim data organisation

All data describing the example are organized into files and directories.

Figure 14 shows an intermediate organisation used during early stages of the programme. This organisation might then evolved based on the optional ecoa-project file in future stages.



**Figure 14 Directories**

Data types used for every definition are defined by “\*\*\*.types.xml” located in the directory named “0-Types”.

Service definitions are defined by “\*\*\*.interface.xml” located in the directory named “1-Services”.

Each Component Definition is described by a “\*\*\*.componentType” file located in a sub-directory of the directory named “2-ComponentDefinitions”. The name of the sub-directory is the name of the component definition itself. For each Component Definition, “\*\*\*.interface-qos.xml” files describe the initial QoS expected for each service provided or required by an instance of this component definition.

The initial Assembly Schema is defined by a “\*\*\*.composite” file located in the directory named “3-InitialAssembly”.

Each Component Implementation is described by a “\*\*\*.impl.xml” file located in a sub-directory of the directory named “4-ComponentImplementations”. The name of the sub-directory is the name of the component implementation itself. The component supplier may also overload the expected QoS with a new QoS; however, the new QoS shall be compatible with the expected one (e.g. an overloaded data maxageing can be less than the expected one). The file bin-desc.xml describes the list of binary objects associated to modules. The sub-directory of the component implementation contains a “Pinfo” sub-directory for storing private PINFO. The “Pinfo” sub-directory can be broken down into lower sub-directories for organizing PINFO.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



The directory “5-Integration” describes associations and mappings of software onto a logical system. The logical system is described by the file “\*\*\*.logical-system.xml”: it defines logical computing nodes and logical links between them. The association between the component instances and the component implementations is described by the “\*\*\*.impl.composite”. The grouping of modules into protection domains and the mapping of protection domains onto logical computing nodes is described by the file “\*\*\*.deployment.xml”. The computingPlatform and computingNode attributes of the element executeOn in the deployment XML file shall match id attributes of logicalComputingPlatform element and one of its logicalComputingNode child elements in the logical-system XML file. The values for these attributes are free character strings. It is not required to use fixed prefixes. The actual deployment (fine grain deployment) is described by platform-specific files and shall be documented by the platform provider. These files are not described in this example. The file “sca-contribution.xml” is only there for compatibility with the SCA standard. The directory “5-Integration” contains a “Pinfo” sub-directory for storing public PINFO. The “Pinfo” sub-directory can be broken down into lower sub-directories for organizing PINFO. The cross platform view is defined by a “\*\*\*.cross\_platform\_view.xml” file. Association between IDs and wires of the cross platform view are defined by one or more “\*\*\*.ids.xml” files. Association between IDs and wires of the final assembly (“\*\*\*.impl.composite”) are defined by one or more “\*\*\*.ids.xml” files. These “\*\*\*.ids.xml” files are referenced in wiremapping information described by the file “\*\*\*.deployment.xml”.

Table 6 summarizes text above.

**Table 6 Model Data Organisation**

Directory	Sub-directory 1	Sub-directory 2	Sub-directory N	Files
0-Types	N/A	N/A	N/A	***.types.xml
1-Services	N/A	N/A	N/A	***.interface.xml
2-ComponentDefinitions	<name_of_component_definition>	N/A	N/A	<name_of_component>.componentType ***.interface.qos.xml  By example: required_<service_name>.interface.qos.xml and provided_<service_name>.interface.qos.xml
3-InitialAssembly	N/A	N/A	N/A	***.composite
4-ComponentImplementations	<name_of_implementation>			<name_of_implementation>.impl.xml ***.interface.qos.xml (e.g. new_required_<service_name>.interface.qos.xml) bin-desc.xml Binary files (e.g. *.o or *.dll)
		0-Dependencies	N/A	Data type, service and component definitions if “0-Types”, “1-Services” and “2-ComponentDefinitions” directories are not available.
		1-Deliverable	N/A	Zipped file of the upper directory
		Pinfo		Sub-directory for storing Private PINFO
		<name_of_subdirectory>		Optional Sub-directories for organizing Private PINFO
5-Integration	N/A			***.impl.composite ***.logical-system.xml ***.deployment.xml sca-contribution.xml ***.cross_platform_view.xml ***.ids.xml

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

Directory	Sub-directory 1	Sub-directory 2	Sub-directory N	Files
	0-Dependencies	N/A	N/A	Set of directories containing component implementations if 4-ComponentImplementations is not available
	Pinfo			Sub-directory for storing Public PINFO
		<name_of_subdirectory>	<name_of_subdirectory>	Optional Sub-directories for organizing Public PINFO

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 7 Legality Rules

This section defines rules to ensure coherency and consistency of ECOA XML files as well as their compatibility with the underlying SCA technology.

### 7.1 Types

[XML-TYP-1]

DataTypes may have identical names as long as they are declared in different NameSpaces.

[XML-TYP-2]

DataTypes may overload basic ECOA types.

The ECOA Infrastructure shall lookup DataTypes in the local namespace first, then in the ECOA namespace if no match is found in local namespace.

### 7.2 Services

Empty section

### 7.3 Component Definitions

Empty section

### 7.4 Cross Platform View

[XML-CPV-1]

All Composites and their ServiceLinks referenced in a Cross Platform View shall be described into an Assembly Schema.

[XML-CPV-2]

The Logical Computing Platforms referenced in a Cross Platform View shall be described in the Logical System.

[XML-CPV-3]

The LogicalComputingPlatformLinks referenced in a Cross Platform View shall be described in a Logical System.

### 7.5 Initial Assembly schema

[XML-AS-1]

Each component property defined in an assembly schema shall use the type xsd:string for its attribute "type".

The actual ECOA type of the property is defined by the mandatory attribute "@{ecoa-sca}type".

[XML-AS-2]

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

To be valid, an assembly schema does not require that all service instances are connected with service links.

In other words, provided services or required services do not necessarily need to be connected as long as the expected behaviour of the involved components allows it.

[XML-AS-3]

If connected, a required service (reference) shall be connected to at most one provided service.

However, a provided service may be connected to several required services.

## 7.6 Component Implementations

[XML-CI-1]

Every service operation of every service provided by a component shall be connected to at least one module operation provided by one of the component modules through an operation link.

In other words, a component implementation shall provide at least one entry point for each service operation of the services it provides.

[XML-CI-2]

It is possible for a module operation (i.e RequestReceived, EventReceived) not to be connected through an operation link.

It is possible for a container operation (i.e RequestSent, DataRead, DataWritten, EventSent) not to be connected through an operation link.

*Note: see Architecture Specification Part 4 for information about the Infrastructure behaviour in these cases. This may allow reusing module implementations in different contexts.*

## 7.7 Composites

[XML-CO-1]

A provided service shall be promoted only once.

A required service shall be promoted only once.

The same reference promotion may promote several required services.

In case of conflict when flattening a hierarchical assembly schema, names of component instances may be prefixed by the composite name.

## 7.8 Integration

[XML-INT-1]

Any ELI message sent by a Logical Computing Platform shall be tagged with an ID value according to ELI interactions described in the Cross Platform View and may be defined according to the example in Architecture Specification Part 6.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

[XML-INT-2]

ID files used at IT and technical levels for generating Platform Integration Code on one Logical Computing Platform shall associate IDs to Component level interactions as per the Final Assembly.

*I.e: IDs defined at Composite level must be “flattened” or “allocated” to Component level interactions. See also Architecture Specification Part 6.*

[XML-INT-3]

For a given logical computing platform link, ID values must be uniquely defined in the Cross Platform View.

[XML-INT-4]

For a given logical computing platform link, the same ID value can be allocated to more than one Component level interaction in the final assembly, only if these interactions have the same Destination Component Instance, Destination Service Instance and ServiceOperationName.

[XML-INT-5]

The same ID values may be used in different logical computing platform links.

## 7.9 Final Assembly schema

[XML-ASF-1]

The Final Assembly Schema (i.e. the one used at IT and technical levels for generating Platform Integration Code on one Logical Computing Platform) must be a single composite containing a “flat” assembly of components (i.e. without nested composites nor promotion links).

[XML-ASF-2]

For a given Logical Computing Platform, the Final Assembly schema shall describe the final assembly of all ASCs deployed onto that Logical Computing Platform.

*I.e: these ASCs correspond to the Composite of the Cross Platform View which is deployed onto that Logical Computing Platform*

[XML-ASF-3]

For a given Logical Computing Platform, the Final Assembly schema may describe one ASC for each Composite of the Cross Platform View which is deployed onto another Logical Computing Platform.

*Note: such a configuration allows taking Cross Platform View information into account in the Final Assembly schema, and limits the scope of the Final Assembly schema to what is strictly needed to generate Platform Integration Code for one Logical Computing Platform.*

[XML-ASF-4]

For a given Logical Computing Platform, the Final Assembly schema may describe the final assemblies of ASCs deployed onto other Logical Computing Platforms.

*I.e: these ASCs correspond to the Composites of the Cross Platform View which are not deployed onto that Logical Computing Platform. As long as [XML-INT-1] is respected, it is possible to use a*

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an ‘as is’ basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

*Final Assembly schema which contains the “flat” view of all ASCs deployed on all Logical Computing Platforms, which is more information than what is strictly needed to generate Platform Integration Code for one Logical Computing Platform.*

## **7.10 Logical System**

[XML-LS-1]

Name Ids used for Logical Computing Platform Links shall be different from name Ids used for Logical Computing Node Links.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

## 8 ECOA XML Schema Definitions

### 8.1 ecoa-sca-2.0.xsd

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:ecoa-
sca="http://www.ecoa.technology/sca-extension-2.0"
  xmlns:sca="http://docs.oasis-open.org/ns/opencsa/sca/200912"
  elementFormDefault="qualified"
targetNamespace="http://www.ecoa.technology/sca-extension-2.0">
  <import namespace="http://docs.oasis-open.org/ns/opencsa/sca/200912"
    schemaLocation="sca-core-1.1-cd06-subset-2.0.xsd"/>
  <include schemaLocation="extensions/ecoa-sca-instance-2.0.xsd"/>
  <include schemaLocation="extensions/ecoa-sca-interface-2.0.xsd"/>
  <include schemaLocation="ecoa-sca-attributes-2.0.xsd"/>
</schema>
```

### 8.2 ecoa-sca-attributes-2.0.xsd

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" xmlns:ecoa-
sca="http://www.ecoa.technology/sca-extension-2.0"
targetNamespace="http://www.ecoa.technology/sca-extension-2.0">

  <attribute name="type" type="xs:string"/>

  <attribute name="deployment" type="xs:string"/>

  <complexType name="wireInformation">
    <annotation>
      <documentation>Any additional wire information parameters required
        by the system integrator (non-defined by ECOA) e.g.
        CIA (Confidentiality, Integrity and Availability)
      </documentation>
    </annotation>
    <attribute name="link" type="anyURI">
      <annotation>
        <documentation>Link to any external file containing parameters
        </documentation>
      </annotation>
    </attribute>
  </complexType>
```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```
</schema>
```

### 8.3 ecoa-sca-interface-2.0.xsd

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/interface-2.0"
  xmlns:tns="http://www.ecoa.technology/interface-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/interface-2.0">
  <xsd:include schemaLocation="ecoa-common-2.0.xsd"/>
  <xsd:element name="serviceDefinition" type="ServiceDefinition"/>
  <xsd:complexType name="ServiceDefinition">
    <xsd:annotation>
      <xsd:documentation>The definition of an ECOA service, including a
        set of
        operations.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" ref="use"/>
      <xsd:element name="operations" type="Operations">
        <xsd:unique name="operationname">
          <xsd:selector xpath="tns:data|tns:event|tns:requestresponse"/>
          <xsd:field xpath="@name"/>
        </xsd:unique>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="Operations">
    <xsd:annotation>
      <xsd:documentation>A set of named operations.</xsd:documentation>
    </xsd:annotation>
    <xsd:choice maxOccurs="unbounded" minOccurs="0">
      <xsd:element name="data" type="Data"/>
      <xsd:element name="event" type="Event">
        <xsd:unique name="eventparameter">
          <xsd:selector xpath="tns:input"/>
          <xsd:field xpath="@name"/>
        </xsd:unique>
      </xsd:element>
      <xsd:element name="requestresponse" type="RequestResponse">
        <xsd:unique name="requestparameter">
```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



```

        <xsd:selector xpath="tns:input|tns:output"/>
        <xsd:field xpath="@name"/>
    </xsd:unique>
</xsd:element>
</xsd:choice>
</xsd:complexType>
<xsd:complexType abstract="true" name="Operation">
    <xsd:attribute name="name" type="NameId" use="required"/>
    <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="Data">
    <xsd:annotation>
        <xsd:documentation>Use of the "versioned data" exchange mechanism.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="Operation">
            <xsd:attribute name="type" type="TypeQName" use="required"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="Event">
    <xsd:annotation>
        <xsd:documentation>Use of the "event" exchange mechanism.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="Operation">
            <xsd:sequence>
                <xsd:element maxOccurs="unbounded" minOccurs="0"
                    name="input" type="Parameter"/>
            </xsd:sequence>
            <xsd:attribute name="direction" type="E_EventDirection"
                use="required"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="RequestResponse">
    <xsd:annotation>
        <xsd:documentation>Use of the "request-response" exchange
            mechanism.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:extension base="Operation">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="input" type="Parameter"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="output" type="Parameter"/>
  </xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:simpleType name="E_EventDirection">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="SENT_BY_PROVIDER"/>
    <xsd:enumeration value="RECEIVED_BY_PROVIDER"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="Parameter">
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="type" type="TypeQName" use="required"/>
</xsd:complexType>
</xsd:schema>

```

#### 8.4 ecoa-sca-instance-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:ecoa="http://www.ecoa.technology/sca-extension-2.0"
  xmlns:sca="http://docs.oasis-open.org/ns/opencsa/sca/200912"
  elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/sca-extension-2.0">

  <!-- This file contains the extension of the abstract sca:implementation
    into a concrete ecoa:instance. It allows ecoa:instance elements to be used
    in XML files wherever sca:implementation is being allowed in SCA XSDs -->

  <import namespace="http://docs.oasis-open.org/ns/opencsa/sca/200912"
    schemaLocation="../../../sca-core-1.1-cd06-subset-2.0.xsd"/>

  <element name="instance" substitutionGroup="sca:implementation"
    type="ecoa:Instance"/>

  <complexType name="Instance">
    <complexContent>
      <extension base="sca:Implementation">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    <sequence>
      <element maxOccurs="1" minOccurs="0" name="implementation">
        <complexType>
          <attribute name="name" type="string" use="required"/>
        </complexType>
      </element>
    </sequence>
    <attribute name="componentType" type="anyURI" use="required"/>
    <attribute name="version" type="string" use="optional"/>
  </extension>
</complexContent>
</complexType>

</schema>

```

## 8.5 ecoa-bin-desc-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/bin-desc-2.0"
  xmlns:tns="http://www.ecoa.technology/bin-desc-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/bin-desc-2.0">

  <xsd:include schemaLocation="ecoa-common-2.0.xsd"/>
  <xsd:element name="binDesc" type="BinDesc"/>

  <xsd:complexType name="BinDesc">
    <xsd:annotation>
      <xsd:documentation>Links between module implementations and binary
        objects
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="processorTarget" type="ProcessorTarget"/>
      <xsd:element maxOccurs="unbounded" name="binaryModule"
        type="BinaryModule"/>
    </xsd:sequence>
    <!-- the following attribute points to a logical name -->
    <xsd:attribute name="componentImplementation" type="NameId"
      use="required"/>
    <xsd:attribute name="insertionPolicy" type="xsd:anyURI"
      use="optional">
    <xsd:annotation>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    <xsd:documentation>Link to an external table containing
        insertion policy data of the binary ECOA component
    </xsd:documentation>
</xsd:annotation>
</xsd:attribute>
</xsd:complexType>

<xsd:complexType name="ProcessorTarget">
    <xsd:annotation>
        <xsd:documentation>"Identification of the processor for which
            modules have been compiled"
        </xsd:documentation>
    </xsd:annotation>
    <xsd:attribute name="type" type="xsd:string" use="required"/>
</xsd:complexType>

<xsd:complexType name="BinaryModule">
    <xsd:annotation>
        <xsd:documentation>Technical description of the binary module
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="0"
            name="binaryDependency" type="BinaryDependency"/>
    </xsd:sequence>
    <xsd:attribute name="reference" type="xsd:NCName" use="required">
        <xsd:annotation>
            <xsd:documentation>Name of the module implementation
            </xsd:documentation>
        </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="object" type="xsd:anyURI" use="required">
        <xsd:annotation>
            <xsd:documentation>Filename of the binary implementing the
                referenced module. The filename may contain path information
                relative to the bin-desc location e.g. "binaries/module.o"
                relates a file 'module.o' located in a subdirectory 'binaries'
                of the directory containing the bin-desc file. Separators are
                '/'
            </xsd:documentation>
        </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="userContextSize" type="HexOrDecValue"
        use="required">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:annotation>
  <xsd:documentation>Size in bytes of the module user context
</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
<xsd:attribute name="warmStartContextSize" type="HexOrDecValue"
  use="required">
  <xsd:annotation>
    <xsd:documentation>Size in bytes of the module warm start
      context
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="stackSize" type="HexOrDecValue"
  use="required">
  <xsd:annotation>
    <xsd:documentation>maximum size in bytes of the stack used by
      any module entry point (including all sub-function calls)
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="heapSize" type="HexOrDecValue"
  use="required">
  <xsd:annotation>
    <xsd:documentation>maximum size in bytes of the heap (memory
      dynamically allocated by the module binary itself: malloc or
      object instances)
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="checksum" type="HexOrDecValue"
  use="required">
  <xsd:annotation>
    <xsd:documentation>Checksum of the binary</xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>

<xsd:complexType name="BinaryDependency">
  <xsd:annotation>
    <xsd:documentation>binary dependency that needs to be linked
      with
      the initial object
    </xsd:documentation>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

</xsd:annotation>
<xsd:attribute name="object" type="xsd:anyURI" use="required">
  <xsd:annotation>
    <xsd:documentation>Filename of the binary implementing the
      referenced dependency
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="checksum" type="HexOrDecValue"
  use="required">
  <xsd:annotation>
    <xsd:documentation>Checksum of the binary dependency
  </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>
</xsd:schema>

```

## 8.6 ecoa-common-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <!-- The following regexps define what is allowed/forbidden for each kind
    of names used in ECOA. They must take into account the constraints of
    different
    programming languages and development environments supported by ECOA
    (characters
    allowed in file names, identifiers, etc.) NOTE: XML character classes (\i,
    \c, etc.) are intentionally avoided, because of the complexity of their
    definition. -->

  <!-- Name of a library containing data types -->
  <!-- Note: The '.' character is used to structure libraries into hierarchical
    namespaces (like Java packages). -->
  <xsd:simpleType name="LibraryName">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="[A-Za-z][A-Za-z0-9_\.]*"/>
    </xsd:restriction>
  </xsd:simpleType>

  <!-- Name that can be used as an identifier in ECOA models and in the source
    code of ECOA components -->

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<!-- Note: Names starting with '_' are excluded from ECOA models. -->
<xsd:simpleType name="NameId">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[A-Za-z][A-Za-z0-9_]*"/>
  </xsd:restriction>
</xsd:simpleType>

<!-- Name of a data type inside a library -->
<xsd:simpleType name="TypeName">
  <xsd:restriction base="NameId">
  </xsd:restriction>
</xsd:simpleType>

<!-- Name of a type, possibly prefixed by the name of the library that
      defines it. -->
<!-- The prefix may be omitted only for predefined types. -->
<!-- A type T defined in a library L will be denoted "L:T". -->
<xsd:simpleType name="TypeQName">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="([A-Za-z][A-Za-z0-9_\.\.]*:)?[A-Za-z][A-Za-z0-9_]*"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:element name="use">

  <xsd:annotation>
    <xsd:documentation>Declares the use of a library of data types. A
      type T defined in a library L will be denoted "L:T".
    </xsd:documentation>
  </xsd:annotation>

  <xsd:complexType>
    <xsd:attribute name="library" type="LibraryName"
      use="required"/>
  </xsd:complexType>
</xsd:element>

<xsd:simpleType name="HexOrDecValue">
  <xsd:annotation>
    <xsd:documentation> This type shall be used as the base type for
      any XML schema attribute or element that contains a hex-encoded
      binary value or a decimal-coded binary value.
      This hex-encoded binary value contains the mandatory string 0x
      followed by a finite-length sequence of characters 0-9 and a-f.
    </xsd:documentation>
  </xsd:annotation>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:documentation>
</xsd:annotation>
<xsd:restriction base="xsd:string">
  <xsd:pattern value="0x[0-9A-Fa-f]+|[1-9][0-9]*|0"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="ProgrammingLanguage">
  <xsd:annotation>
    <xsd:documentation>Programming languages supported by ECOA
      bindings
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="C"/>
    <xsd:enumeration value="C++"/>
    <xsd:enumeration value="Ada"/>
    <xsd:enumeration value="Java"/>
    <xsd:enumeration value="HI_Ada"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="TimeDuration">
  <xsd:annotation>
    <xsd:documentation>The unit is in seconds</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:double">
    <xsd:minInclusive value="0.0"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="Steps">
  <xsd:annotation>
    <xsd:documentation>A number of logical steps
      This allows to size
      durations in an abstract way.
      See stepDuration in the logical
      system
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:double">
    <xsd:minInclusive value="0.0"/>
  </xsd:restriction>
</xsd:simpleType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



```
</xsd:schema>
```

## 8.7 ecoa-deployment-2.0.xsd

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/deployment-2.0"
  xmlns:tns="http://www.ecoa.technology/deployment-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/deployment-2.0">
  <xsd:include schemaLocation="ecoa-common-2.0.xsd" />

  <xsd:element name="deployment" type="Deployment">
    <xsd:key name="execnamekey">
      <xsd:selector xpath="tns:protectionDomain" />
      <xsd:field xpath="@name" />
    </xsd:key>
    <xsd:key name="deployedModuleInstanceKey">
      <xsd:selector xpath="tns:protectionDomain/tns:deployedModuleInstance" />
      <xsd:field xpath="@componentName" />
      <xsd:field xpath="@moduleInstanceName" />
    </xsd:key>
    <xsd:key name="deployedTriggerInstanceKey">
      <xsd:selector xpath="tns:protectionDomain/tns:deployedTriggerInstance" />
      <xsd:field xpath="@componentName" />
      <xsd:field xpath="@triggerInstanceName" />
    </xsd:key>
    <xsd:unique name="componentLogKey">
      <xsd:selector xpath="tns:logPolicy/tns:componentLog" />
      <xsd:field xpath="@instanceName" />
    </xsd:unique>
    <xsd:key name="computingPlatformConfigurationKey">
      <xsd:selector xpath="tns:computingPlatformConfiguration" />
      <xsd:field xpath="@name" />
    </xsd:key>
    <xsd:key name="wireMappingKey">
      <xsd:selector xpath="tns:wireMapping" />
      <xsd:field xpath="@source" />
      <xsd:field xpath="@target" />
      <xsd:field xpath="@mappedOnLinkId" />
    </xsd:key>
  </xsd:element>
<xsd:complexType name="Deployment">
```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:sequence>
  <xsd:element maxOccurs="unbounded" name="protectionDomain"
    type="ProtectionDomain" />
  <xsd:element maxOccurs="unbounded" minOccurs="0" name="logPolicy"
    type="LogPolicy" />
  <xsd:element maxOccurs="unbounded" minOccurs="1"
    name="platformConfiguration" type="PlatformConfiguration">
    <xsd:annotation>
      <xsd:documentation>Defines platform-wide settings
    </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element maxOccurs="unbounded" minOccurs="0" name="wireMapping"
    type="WireMapping">
    <xsd:annotation>
      <xsd:documentation>Defines a mapping between a wire and a
        computing platform link
    </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:sequence>
<xsd:attribute name="finalAssembly" type="NameId" use="required">
  <xsd:annotation>
    <xsd:documentation>Name of the composite referenced by this
      deployment
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="logicalSystem" type="NameId" use="required">
  <xsd:annotation>
    <xsd:documentation>Name of the logical system this deployment is
      made on
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>
<xsd:complexType name="ProtectionDomain">
  <xsd:annotation>
    <xsd:documentation>Defines an OS executable, offering memory (and
      possibly also temporal) protection
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="executeOn">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:complexType>
  <xsd:attribute name="computingNode" type="NameId" use="required" />
  <xsd:attribute name="computingPlatform" type="NameId"
    use="required">
    <xsd:annotation>
      <xsd:documentation>Id of a logical system.
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:choice maxOccurs="unbounded" minOccurs="0">
  <xsd:element maxOccurs="unbounded" minOccurs="0"
    name="deployedModuleInstance">
    <xsd:complexType>
      <xsd:attribute name="componentName" type="NameId" use="required" />
      <xsd:attribute name="moduleInstanceName" type="NameId"
        use="required" />
      <xsd:attribute name="modulePriority" type="ModulePriority"
        use="required">
        <xsd:annotation>
          <xsd:documentation>Abstract module priority that can be
            used by the platform to map the module on an
            actual OS priority
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
    </xsd:complexType>
  </xsd:element>
  <xsd:element maxOccurs="unbounded" minOccurs="0"
    name="deployedTriggerInstance">
    <xsd:complexType>
      <xsd:attribute name="componentName" type="NameId" use="required" />
      <xsd:attribute name="triggerInstanceName" type="NameId"
        use="required" />
      <xsd:attribute name="triggerPriority" type="ModulePriority"
        use="required">
        <xsd:annotation>
          <xsd:documentation>Abstract trigger priority that can be
            used by the platform to map the trigger on an actual
            OS priority
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
    </xsd:complexType>
  </xsd:element>
</xsd:choice>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        </xsd:complexType>
    </xsd:element>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name="name" type="NameId" use="required" />
</xsd:complexType>

<xsd:simpleType name="ModulePriority">
    <xsd:restriction base="xsd:decimal">
        <xsd:minInclusive value="0" />
        <xsd:maxInclusive value="255" />
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="LogPolicy">
    <xsd:annotation>
        <xsd:documentation>Defines the log policy for deployed components
            and modules
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="0" name="componentLog"
            type="ComponentLog">
            <xsd:key name="moduleIdKey">
                <xsd:selector xpath="./tns:moduleLog" />
                <xsd:field xpath="@instanceName" />
            </xsd:key>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ComponentLog">
    <xsd:annotation>
        <xsd:documentation>Defines default level of logging for a given
            component
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="0" name="moduleLog"
            type="ModuleLog"/>
    </xsd:sequence>
    <xsd:attribute name="instanceName" type="xsd:string" use="required" />
    <xsd:attribute name="enabledLevels" type="xsd:string" use="required" />
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:complexType name="ModuleLog">
  <xsd:annotation>
    <xsd:documentation>Defines level of logging for a deployed module
      instance
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="instanceName" type="xsd:string" use="required" />
  <xsd:attribute name="enabledLevels" type="xsd:string" use="required" />
</xsd:complexType>

<xsd:complexType name="PlatformConfiguration">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="computingNodeConfiguration" type="ComputingNodeConfiguration">
      <xsd:annotation>
        <xsd:documentation>Defines the computing node level
          configuration
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="platformMessages" type="PlatformMessages">
      <xsd:annotation>
        <xsd:documentation>Define on which the platform
          domain messages are mapped
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="computingPlatform" type="NameId" use="required">
    <xsd:annotation>
      <xsd:documentation>Id of a logical system.</xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="faultHandlerNotificationMaxNumber" type="FifoSize"
    use="optional" default="8">
    <xsd:annotation>
      <xsd:documentation>Defines the number of fault handler
        notifications that a Module Container shall be able
        to handle at any time. These notifications are
        relevant depending on the Module "isFaultHandler"
        attribute.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="EUIDs" type="xsd:anyURI" use="optional">
    <xsd:annotation>
      <xsd:documentation>Specific EUIDS file associated to one
        given peer of the link
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>

<xsd:simpleType name="FifoSize">
  <xsd:restriction base="xsd:decimal">
    <xsd:minInclusive value="1" />
    <xsd:maxInclusive value="255" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="ComputingNodeConfiguration">
  <xsd:attribute name="computingNode" type="NameId" use="required">
    <xsd:annotation>
      <xsd:documentation>Id of a logical computing node
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="schedulingInformation" type="xsd:anyURI"
    use="optional">
    <xsd:annotation>
      <xsd:documentation>Link to any external file containing any
        additional scheduling parameters required by the system
        integrator (non-defined by ECOA)
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>

<xsd:complexType name="PlatformMessages">
  <xsd:attribute name="mappedOnLinkId" type="NameId" use="required">
    <xsd:annotation>
      <xsd:documentation>Refers to the inter-platforms link
        on which the platform-level management messages
        are mapped. The link is defined in
        the logical system.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>

<xsd:complexType name="WireMapping">
  <xsd:annotation>
    <xsd:documentation>Defines a mapping between a wire and a
      computing platform link
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="source" type="xsd:anyURI" use="required">
    <xsd:annotation>
      <xsd:documentation>Wire Source
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="target" type="xsd:anyURI" use="required">
    <xsd:annotation>
      <xsd:documentation>Wire Target
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="mappedOnLinkId" type="NameId" use="required">
    <xsd:annotation>
      <xsd:documentation>Refers to the inter-nodes
        or the inter-platforms link
        on which the wire is mapped. The link is
        is defined at platform description level
        in the logical system.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>

</xsd:schema>

```

## 8.8 ecoa-implementation-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/implementation-2.0"
  xmlns:tns="http://www.ecoa.technology/implementation-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/implementation-2.0">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:include schemaLocation="ecoa-common-2.0.xsd"/>
<xsd:element name="componentImplementation" type="ComponentImplementation">
  <!-- keys: name unicity constraints -->
  <xsd:key name="moduleTypekey">
    <xsd:selector xpath="tns:moduleType"/>
    <xsd:field xpath="@name"/>
  </xsd:key>
  <xsd:key name="moduleImplementationkey">
    <xsd:selector xpath="tns:moduleImplementation"/>
    <xsd:field xpath="@name"/>
  </xsd:key>
  <xsd:key name="moduleInstancekey">
    <xsd:selector xpath="tns:moduleInstance"/>
    <xsd:field xpath="@name"/>
  </xsd:key>
  <xsd:key name="triggerInstancekey">
    <xsd:selector xpath="tns:triggerInstance"/>
    <xsd:field xpath="@name"/>
  </xsd:key>
  <xsd:key name="dynamicTriggerInstancekey">
    <xsd:selector xpath="tns:dynamicTriggerInstance"/>
    <xsd:field xpath="@name"/>
  </xsd:key>
  <!-- triggers, dynamicTriggers and ordinary modules must have distinct
  names -->
  <xsd:key name="moduleOrTriggerInstancekey">
    <xsd:selector
xpath="tns:moduleInstance|tns:triggerInstance|tns:dynamicTriggerInstance"/>
    <xsd:field xpath="@name"/>
  </xsd:key>
  <!-- The same operation shall appear only one time if present in the
  element clients -->
  <xsd:key name="moduleInstanceClientRequestLinkkey">
    <xsd:selector xpath="tns:requestLink/tns:clients/tns:moduleInstance"/>
    <xsd:field xpath="@instanceName"/>
    <xsd:field xpath="@operationName"/>
  </xsd:key>
  <xsd:key name="serviceClientRequestLinkkey">
    <xsd:selector xpath="tns:requestLink/tns:clients/tns:service"/>
    <xsd:field xpath="@instanceName"/>
    <xsd:field xpath="@operationName"/>
  </xsd:key>
  <!-- The same operation shall appear only once if present in the element

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



```

servers -->
<xsd:key name="moduleInstanceServerRequestLinkkey">
  <xsd:selector xpath="tns:requestLink/tns:server/tns:moduleInstance"/>
  <xsd:field xpath="@instanceName"/>
  <xsd:field xpath="@operationName"/>
</xsd:key>
<xsd:key name="moduleInstanceReceiversEventLinkkey">
  <xsd:selector xpath="tns:eventLink/tns:receivers/tns:moduleInstance"/>
  <xsd:field xpath="@instanceName"/>
  <xsd:field xpath="@operationName"/>
</xsd:key>
<xsd:key name="moduleInstanceReadersDataLinkkey">
  <xsd:selector xpath="tns:dataLink/tns:readers/tns:moduleInstance"/>
  <xsd:field xpath="@instanceName"/>
  <xsd:field xpath="@operationName"/>
</xsd:key>
<!-- keyrefs: constraints that a reference refers to a name defined in
a key -->
<xsd:keyref name="moduleInstancekeyRef" refer="moduleInstancekey">
  <xsd:selector xpath="*/*/tns:moduleInstance"/>
  <xsd:field xpath="@instanceName"/>
</xsd:keyref>
<xsd:keyref name="triggerInstancekeyRef" refer="triggerInstancekey">
  <xsd:selector xpath="tns:eventLink/tns:trigger"/>
  <xsd:field xpath="@triggerInstance"/>
</xsd:keyref>
<xsd:keyref name="dynamicTriggerInstancekeyRef"
refer="dynamicTriggerInstancekey">
  <xsd:selector xpath="tns:eventLink/*/tns:dynamicTrigger"/>
  <xsd:field xpath="@instanceName"/>
</xsd:keyref>
<xsd:keyref name="moduleImplementation_to_moduleType"
refer="moduleTypekey">
  <xsd:selector xpath="tns:moduleImplementation"/>
  <xsd:field xpath="@moduleType"/>
</xsd:keyref>
<xsd:keyref name="moduleInstance_to_moduleImplementation"
refer="moduleImplementationkey">
  <xsd:selector xpath="tns:moduleInstance"/>
  <xsd:field xpath="@implementationName"/>
</xsd:keyref>
<xsd:key name="pinfoname">
  <xsd:selector

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

xpath="tns:moduleType/tns:pinfo/tns:publicPinfo|tns:moduleType/tns:pinfo/tns:privatePinfo"/>
  <xsd:field xpath="@name"/>
</xsd:key>
<xsd:keyref name="pinfovalue_to_pinfo" refer="pinfoName">
  <xsd:selector
xpath="tns:moduleInstance/tns:pinfo/tns:publicPinfo|tns:moduleInstance/tns:pinfo/tns:privatePinfo"/>
  <xsd:field xpath="@name"/>
</xsd:keyref>
</xsd:element>
<xsd:complexType name="ComponentImplementation">
  <xsd:annotation>
    <xsd:documentation>
      Describes all the information needed to
      integrate the software implementation of an ECOA
      component in an
      ECOA system.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0" ref="use"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="service"
      type="ServiceQoS"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="reference" type="ServiceQoS"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="moduleType" type="ModuleType"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="moduleImplementation" type="ModuleImplementation"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="moduleInstance" type="ModuleInstance"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="triggerInstance" type="TriggerInstance"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="dynamicTriggerInstance" type="DynamicTriggerInstance"/>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element maxOccurs="unbounded" minOccurs="0"
        name="dataLink" type="DataLink"/>
      <xsd:element maxOccurs="unbounded" minOccurs="0"
        name="eventLink" type="EventLink"/>
      <xsd:element maxOccurs="unbounded" minOccurs="0"
        name="requestLink" type="RequestLink"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:choice>
  </xsd:sequence>
  <xsd:attribute name="componentDefinition" type="NameId"
    use="required"/>
</xsd:complexType>
<xsd:complexType name="ServiceQoS">
  <xsd:annotation>
    <xsd:documentation>To define a new QoS for a provided or required
      service
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="newQoS" type="xsd:anyURI" use="required"/>
</xsd:complexType>
<xsd:complexType name="ModuleType">
  <xsd:annotation>
    <xsd:documentation>Describes a single-threaded ECOA module,
      implemented as software, contributing to the implementation of
      an ECOA component.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element minOccurs="0" name="properties">
      <xsd:annotation>
        <xsd:documentation>Set of module properties. The value of each
          module property is set at design time.
        </xsd:documentation>
      </xsd:annotation>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element maxOccurs="unbounded" name="property"
            type="Parameter">
            <xsd:annotation>
              <xsd:documentation>The value of each module property is
                set at design time at instance definition level.
              </xsd:documentation>
            </xsd:annotation>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
      <xsd:unique name="propertyname">
        <xsd:selector xpath="tns:property"/>
        <xsd:field xpath="@name"/>
      </xsd:unique>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

</xsd:element>
<xsd:element minOccurs="0" name="pinfo">
  <xsd:annotation>
    <xsd:documentation>Set of pinfo used by the module
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" maxOccurs="unbounded"
        name="publicPinfo" type="PublicPinfo"/>
      <xsd:element minOccurs="0" maxOccurs="unbounded"
        name="privatePinfo" type="PrivatePinfo"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="operations">
  <xsd:complexType>
    <xsd:choice maxOccurs="unbounded">
      <xsd:element minOccurs="0" name="dataWritten">
        <xsd:annotation>
          <xsd:documentation>Read+Write access to a versioned
            data if writeonly=false. Write only access to a
            versioned data if writeonly=true.
            Note: the writeonly attribute is ignored by the
            Infrastructure if controlled=false on the dataLink.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="VersionedData">
              <xsd:attribute default="false" name="writeOnly"
                type="xsd:boolean" use="optional"/>
            </xsd:extension>
          </xsd:complexContent>
        </xsd:complexType>
      </xsd:element>
      <xsd:element minOccurs="0" name="dataRead">
        <xsd:annotation>
          <xsd:documentation>Read-only access to a versioned data.
          </xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:complexContent>
            <xsd:extension base="VersionedData">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        <xsd:attribute default="false" name="notifying"
            type="xsd:boolean" use="optional"/>
    </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element minOccurs="0" name="eventSent"
    type="Event">
    <xsd:unique name="eventparameter_sent">
        <xsd:selector xpath="tns:input"/>
        <xsd:field xpath="@name"/>
    </xsd:unique>
</xsd:element>
<xsd:element minOccurs="0" name="eventReceived">
    <xsd:complexType>
        <xsd:complexContent>
            <xsd:extension base="Event"/>
        </xsd:complexContent>
    </xsd:complexType>
    <xsd:unique name="eventparameter_received">
        <xsd:selector xpath="tns:input"/>
        <xsd:field xpath="@name"/>
    </xsd:unique>
</xsd:element>
<xsd:element minOccurs="0" name="requestSent">
    <xsd:complexType>
        <xsd:complexContent>
            <xsd:extension base="RequestResponse">
                <xsd:attribute name="timeout" type="xsd:double"
                    use="required">
                    <xsd:annotation>
                        <xsd:documentation>Timeout value to
                            unblock/inform respectively a
                            synchronous/asynchronous RR
                            If the value is
                            negative, the timeout
                            is infinite.
                        </xsd:documentation>
                    </xsd:annotation>
                </xsd:attribute>
                <xsd:attribute name="isSynchronous"
                    type="xsd:boolean" use="required"/>
                <xsd:attribute name="maxConcurrentRequests"
                    type="xsd:positiveInteger" use="optional" default="10">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        <xsd:annotation>
            <xsd:documentation>Max number of concurrent
                requests that the module may handle for the
                related container call.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:unique name="requestparameter_req">
    <xsd:selector xpath="tns:input|tns:output"/>
    <xsd:field xpath="@name"/>
</xsd:unique>
</xsd:element>
<xsd:element minOccurs="0" name="requestReceived">
    <xsd:complexType>
        <xsd:complexContent>
            <xsd:extension base="RequestResponse">
                <xsd:attribute default="10"
                    name="maxConcurrentRequests" type="xsd:positiveInteger"
                    use="optional">
                    <xsd:annotation>
                        <xsd:documentation>Max number of concurrent
                            responses that the module may handle for the
                            related entry-point, regardless of incoming
                            requestLinks related to that entry-point.
                        </xsd:documentation>
                    </xsd:annotation>
                </xsd:attribute>
            </xsd:extension>
        </xsd:complexContent>
    </xsd:complexType>
    <xsd:unique name="requestparameter_pro">
        <xsd:selector xpath="tns:input|tns:output"/>
        <xsd:field xpath="@name"/>
    </xsd:unique>
</xsd:element>
</xsd:choice>
</xsd:complexType>
<xsd:key name="operationkey">
    <xsd:selector xpath="tns:*"/>
    <xsd:field xpath="@name"/>
</xsd:key>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:element>
</xsd:sequence>
<xsd:attribute name="name" type="NameId" use="required"/>
<xsd:attribute default="true" name="hasUserContext"
  type="xsd:boolean" use="optional">
  <xsd:annotation>
    <xsd:documentation>To indicate if the module relies on a user
      context
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute default="true" name="hasWarmStartContext"
  type="xsd:boolean" use="optional">
  <xsd:annotation>
    <xsd:documentation>To indicate if the module relies on a warm
      start context
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute default="false" name="isFaultHandler"
  type="xsd:boolean" use="optional">
  <xsd:annotation>
    <xsd:documentation>To indicate if the module is a Fault
      Handler
      or not and to generate fault handling API.
      To enable the
      generation, the platform has to support
      this kind of Fault
      Handler deployment. See Platform
      Procurement Requirements.
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute default="true" name="activatingFaultNotifs"
  type="xsd:boolean" use="optional">
  <xsd:annotation>
    <xsd:documentation>Does the reception of fault notifications
      cause the activation of the receiver Fault Handler (only if
      the Fault Handler is implemented as an ECOA component) ?
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>
<xsd:complexType name="Event">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:sequence>
  <xsd:element maxOccurs="unbounded" minOccurs="0" name="input"
    type="Parameter"/>
</xsd:sequence>
<xsd:attribute name="name" type="NameId" use="required"/>
</xsd:complexType>
<xsd:complexType name="RequestResponse">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="input"
      type="Parameter"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="output"
      type="Parameter"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="NameId" use="required"/>
</xsd:complexType>
<xsd:complexType name="VersionedData">
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="type" type="TypeQName" use="required">
    <xsd:annotation>
      <xsd:documentation>Type stored by the versioned data.
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute default="1" name="maxVersions" type="xsd:positiveInteger"
    use="optional">
    <xsd:annotation>
      <xsd:documentation>Max number of versions accessed at the same
        time.
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>
<xsd:complexType name="Parameter">
  <xsd:annotation>
    <xsd:documentation>A parameter a an operation (Event,
      RequestResponse or VersionedData)
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="type" type="TypeQName" use="required"/>
</xsd:complexType>
<xsd:complexType name="PublicPinfo">
  <xsd:annotation>
    <xsd:documentation>Logical name of a public pinfo used by a module
  </xsd:documentation>
  </xsd:annotation>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



```

    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="name" type="NameId" use="required"/>
</xsd:complexType>
<xsd:complexType name="PrivatePinfo">
  <xsd:annotation>
    <xsd:documentation>Logical name of a private pinfo used by a
      module
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="name" type="NameId" use="required"/>
</xsd:complexType>
<xsd:complexType name="ModuleImplementation">
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="language" type="ProgrammingLanguage"
    use="required">
    <xsd:annotation>
      <xsd:documentation>Programming language</xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="moduleType" type="NameId" use="required"/>
</xsd:complexType>
<xsd:complexType name="Instance">
  <xsd:annotation>
    <xsd:documentation/>
  </xsd:annotation>
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="moduleBehaviour" type="xsd:anyURI"
    use="optional"/>
  <xsd:attribute name="relativePriority" type="RelativePriority"
    use="required">
    <xsd:annotation>
      <xsd:documentation>
        Relative priority of this module instance to
        others module instances of the same component instance
        to help
        to distinguish them when allocating actual
        priorities at
        deployment level
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:complexType name="ModuleInstance">
  <xsd:annotation>
    <xsd:documentation>Describes an instance of a Module (having its
      own internal state).
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Instance">
      <xsd:sequence>
        <xsd:element maxOccurs="1" minOccurs="0"
          name="propertyValues" type="PropertyValues"/>
        <xsd:element maxOccurs="1" minOccurs="0" name="pinfo">
          <xsd:annotation>
            <xsd:documentation>Set of pinfo used by the module
          </xsd:documentation>
          </xsd:annotation>
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element minOccurs="0" maxOccurs="unbounded"
                name="publicPinfo" type="PinfoValue"/>
              <xsd:element minOccurs="0" maxOccurs="unbounded"
                name="privatePinfo" type="PinfoValue"/>
            </xsd:sequence>
          </xsd:complexType>
          <xsd:unique name="pinfovaluename">
            <xsd:selector xpath="tns:publicPinfo|tns:privatePinfo"/>
            <xsd:field xpath="@name"/>
          </xsd:unique>
        </xsd:element>
      </xsd:sequence>
      <xsd:attribute name="implementationName" type="NameId"
        use="required"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:simpleType name="RelativePriority">
  <xsd:restriction base="xsd:nonNegativeInteger">
    <xsd:minInclusive value="0"/>
    <xsd:maxInclusive value="255"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="PropertyValues">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:annotation>
  <xsd:documentation>set of module property values
</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
  <xsd:element maxOccurs="unbounded" name="propertyValue"
    type="PropertyValue">
  </xsd:element>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="PropertyValue">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="name" type="xsd:string" use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="PinfoValue">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute name="name" type="xsd:string" use="required"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="DataLink">
  <xsd:annotation>
    <xsd:documentation>Link between DATA operations.
  </xsd:documentation>
</xsd:annotation>
  <xsd:sequence>
    <xsd:element name="writers">
      <xsd:complexType>
        <xsd:sequence maxOccurs="unbounded">
          <xsd:choice>
            <xsd:element name="reference" type="OpRef"/>
            <xsd:element name="moduleInstance" type="OpRef"/>
          </xsd:choice>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="readers" minOccurs="0" maxOccurs="1">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:complexType>
  <xsd:sequence maxOccurs="unbounded">
    <xsd:choice>
      <xsd:element name="service" type="OpRef"/>
      <xsd:element name="moduleInstance" type="OpRefActivatableFifo"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="id" type="xsd:int" use="optional"/>
<xsd:attribute name="controlled" type="xsd:boolean"
  use="optional" default="true">
  <xsd:annotation>
    <xsd:documentation>Boolean flag to indicate if the Versioned
      Data access are controlled by the Infrastructure.
      If true, each concurrent write accesses to its own copy
      of the data and readers are ensured that the copy they
      access is stable until the release of the VD handle.
      Otherwise, if false, any module getting a handle may
      directly access the local data repository (as no copy is
      made by the Infrastructure).
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>
<xsd:complexType name="EventLink">
  <xsd:annotation>
    <xsd:documentation>Link between EVENT operations.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element minOccurs="0" name="senders">
      <xsd:complexType>
        <xsd:sequence maxOccurs="unbounded">
          <xsd:choice>
            <xsd:element name="service" type="OpRef"/>
            <xsd:element name="reference" type="OpRef"/>
            <xsd:element name="moduleInstance" type="OpRef"/>
            <xsd:element name="trigger" type="OpRef_Trigger"/>
            <xsd:element name="dynamicTrigger" type="OpRef"/>
            <xsd:element name="external" type="OpRef_External"/>
          </xsd:choice>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:complexType>
</xsd:element>
<xsd:element name="receivers">
  <xsd:complexType>
    <xsd:sequence maxOccurs="unbounded">
      <xsd:choice>
        <xsd:element name="service" type="OpRef"/>
        <xsd:element name="reference" type="OpRef"/>
        <xsd:element name="moduleInstance" type="OpRefActivatableFifo"/>
        <xsd:element name="dynamicTrigger" type="OpRefActivatingFifo"/>
      </xsd:choice>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="id" type="xsd:int" use="optional"/>
</xsd:complexType>
<xsd:complexType name="RequestLink">
  <xsd:annotation>
    <xsd:documentation>Link between RR operations. Must have exactly
      one server. Can have many clients.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="clients">
      <xsd:complexType>
        <xsd:sequence maxOccurs="unbounded">
          <xsd:choice>
            <xsd:element name="service" type="OpRef"/>
            <xsd:element name="moduleInstance" type="OpRefActivatable">
              <xsd:annotation>
                <xsd:documentation>Note: attribute 'activating'
                  concerns the response, and is applicable to
                  asynchronous RR operations only.
                </xsd:documentation>
              </xsd:annotation>
            </xsd:element>
          </xsd:choice>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="server">
      <xsd:complexType>
        <xsd:choice>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:element name="reference" type="OpRef"/>
<xsd:element name="moduleInstance" type="OpRefActivatableFifo">
  <xsd:annotation>
    <xsd:documentation>Note: optional attributes concern the
      request
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="id" type="xsd:int" use="optional"/>
</xsd:complexType>
<xsd:complexType name="OpRef">
  <xsd:attribute name="instanceName" type="NameId" use="required">
    <xsd:annotation>
      <xsd:documentation>Reference to a module instance, a service, or
        a reference
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="operationName" type="NameId" use="required"/>
</xsd:complexType>
<xsd:complexType name="OpRefActivatable">
  <xsd:complexContent>
    <xsd:extension base="OpRef">
      <xsd:attribute default="true" name="activating"
        type="xsd:boolean" use="optional">
        <xsd:annotation>
          <xsd:documentation>Does the reception of the event/data/rr
            cause the activation of the receiver module ?
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="OpRefActivatableFifo">
  <xsd:complexContent>
    <xsd:extension base="OpRefActivatable">
      <xsd:attribute default="8" name="fifoSize"
        type="xsd:positiveInteger" use="optional">
        <xsd:annotation>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:documentation>Max number of incoming operations that
    can be
    stored in the receiver module's FIFO queue for that
    particular operation link, before the activation
    of the
    corresponding entrypoint.
    There is one fifoSize per
    operation link on the receiver side.
    If this max number is
    exceeded, new incoming operations are trashed.
</xsd:documentation>
</xsd:annotation>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="OpRefActivatingFifo">
  <xsd:complexContent>
    <xsd:extension base="OpRef">
      <xsd:attribute default="8" name="fifoSize"
        type="xsd:positiveInteger" use="optional">
        <xsd:documentation>Max number of incoming operations that
          can be stored in the receiver module's FIFO queue for that
          particular operation link, before the activation
          of the
          corresponding entrypoint.
          There is one fifoSize per
          operation link on the receiver
          side.
          If this max number is
          exceeded, new incoming operations
          are discarded.
          These
          operations are activating.
        </xsd:documentation>
      </xsd:documentation>
    </xsd:attribute>
  </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="OpRef_Trigger">
  <xsd:attribute name="instanceName" type="NameId" use="required"/>
  <xsd:attribute name="period" type="TimeDuration" use="required">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    <xsd:annotation>
      <xsd:documentation>period in seconds</xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>
<xsd:complexType name="OpRef_External">
  <xsd:annotation>
    <xsd:documentation>Reference used for asynchronous notification
      coming the legacy code (driver component)
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="operationName" type="NameId" use="required"/>
  <xsd:attribute name="language" type="ProgrammingLanguage"
    use="required">
    <xsd:annotation>
      <xsd:documentation>Programming language in which the external
        API will be generated for the non-ECO A SW part of the driver
        component
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>
<xsd:complexType name="TriggerInstance">
  <xsd:complexContent>
    <xsd:extension base="Instance">
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="DynamicTriggerInstance">
  <xsd:complexContent>
    <xsd:extension base="Instance">
      <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="0"
          name="parameter" type="Parameter"/>
      </xsd:sequence>
      <xsd:attribute default="1" name="size"
        type="xsd:positiveInteger" use="optional">
        <xsd:annotation>
          <xsd:documentation>Max number of events waiting for delay
            expiration in the trigger
          </xsd:documentation>
        </xsd:annotation>
      </xsd:attribute>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



```

    </xsd:complexContent>
  </xsd:complexType>
</xsd:schema>

```

## 8.9 ecoa-interface-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/interface-2.0"
  xmlns:tns="http://www.ecoa.technology/interface-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/interface-2.0">
  <xsd:include schemaLocation="ecoa-common-2.0.xsd"/>
  <xsd:element name="serviceDefinition" type="ServiceDefinition"/>
  <xsd:complexType name="ServiceDefinition">
    <xsd:annotation>
      <xsd:documentation>The definition of an ECOA service, including a
        set of
        operations.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0" ref="use"/>
      <xsd:element name="operations" type="Operations">
        <xsd:unique name="operationname">
          <xsd:selector xpath="tns:data|tns:event|tns:requestresponse"/>
          <xsd:field xpath="@name"/>
        </xsd:unique>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="Operations">
    <xsd:annotation>
      <xsd:documentation>A set of named operations.</xsd:documentation>
    </xsd:annotation>
    <xsd:choice maxOccurs="unbounded" minOccurs="0">
      <xsd:element name="data" type="Data"/>
      <xsd:element name="event" type="Event">
        <xsd:unique name="eventparameter">
          <xsd:selector xpath="tns:input"/>
          <xsd:field xpath="@name"/>
        </xsd:unique>
      </xsd:element>
      <xsd:element name="requestresponse" type="RequestResponse"/>
    </xsd:choice>
  </xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        <xsd:unique name="requestparameter">
            <xsd:selector xpath="tns:input|tns:output"/>
            <xsd:field xpath="@name"/>
        </xsd:unique>
    </xsd:element>
</xsd:choice>
</xsd:complexType>
<xsd:complexType abstract="true" name="Operation">
    <xsd:attribute name="name" type="NameId" use="required"/>
    <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="Data">
    <xsd:annotation>
        <xsd:documentation>Use of the "versioned data" exchange mechanism.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="Operation">
            <xsd:attribute name="type" type="TypeQName" use="required"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="Event">
    <xsd:annotation>
        <xsd:documentation>Use of the "event" exchange mechanism.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
        <xsd:extension base="Operation">
            <xsd:sequence>
                <xsd:element maxOccurs="unbounded" minOccurs="0"
                    name="input" type="Parameter"/>
            </xsd:sequence>
            <xsd:attribute name="direction" type="E_EventDirection"
                use="required"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="RequestResponse">
    <xsd:annotation>
        <xsd:documentation>Use of the "request-response" exchange
            mechanism.
        </xsd:documentation>
    </xsd:annotation>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:complexContent>
  <xsd:extension base="Operation">
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="0"
        name="input" type="Parameter"/>
      <xsd:element maxOccurs="unbounded" minOccurs="0"
        name="output" type="Parameter"/>
    </xsd:sequence>
  </xsd:extension>
</xsd:complexContent>
</xsd:complexType>
<xsd:simpleType name="E_EventDirection">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="SENT_BY_PROVIDER"/>
    <xsd:enumeration value="RECEIVED_BY_PROVIDER"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="Parameter">
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="type" type="TypeQName" use="required"/>
</xsd:complexType>
</xsd:schema>

```

## 8.10 ecoa-interface-qos-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/interface-qos-2.0"
  xmlns:tns="http://www.ecoa.technology/interface-qos-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/interface-qos-2.0">
  <xsd:include schemaLocation="ecoa-common-2.0.xsd"/>
  <xsd:element name="serviceInstanceQoS" type="ServiceInstanceQoS"/>
  <xsd:complexType name="ServiceInstanceQoS">
    <xsd:annotation>
      <xsd:documentation>The definition of an ECOA service, including a
        set of operations.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence maxOccurs="1" minOccurs="1">
      <xsd:element name="operations" type="Operations"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="Operations">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:annotation>
  <xsd:documentation>A set of named operations.</xsd:documentation>
</xsd:annotation>
<xsd:choice maxOccurs="unbounded" minOccurs="1">
  <xsd:element name="data" type="Data"/>
  <xsd:element name="event" type="Event"/>
  <xsd:element name="requestresponse" type="RequestResponse"/>
</xsd:choice>
</xsd:complexType>
<xsd:complexType name="Data">
  <xsd:annotation>
    <xsd:documentation>Use of the "versioned data" exchange
      mechanism.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element maxOccurs="1" minOccurs="0" name="highestRate"
      type="OperationRate">
      <xsd:annotation>
        <xsd:documentation>Max number of occurrences within a
          reference time
          frame
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element maxOccurs="1" minOccurs="0" name="lowestRate"
      type="OperationRate">
      <xsd:annotation>
        <xsd:documentation>Min number of occurrences within a
          reference time frame
        </xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="maxAgeing" type="TimeDuration"
    use="optional">
    <xsd:annotation>
      <xsd:documentation>Operation Provided : max duration between
        Data production (from the source) and the end of writing
        process.
        Operation Required : max duration between Data
        production
        (from the source) and the end of reading process.
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    Unit is second.
  </xsd:documentation>
</xsd:annotation>
</xsd:attribute>
<xsd:attribute name="notificationMaxHandlingTime"
  type="TimeDuration" use="optional">
  <xsd:annotation>
    <xsd:documentation>Notifying data case: maxHandlingTime for
      notification event. Unit is second.
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>
<xsd:complexType name="Event">
  <xsd:annotation>
    <xsd:documentation>Use of the "event" exchange mechanism.
  </xsd:documentation>
</xsd:annotation>
<xsd:sequence>
  <xsd:element maxOccurs="1" minOccurs="0" name="highestRate"
    type="OperationRate">
    <xsd:annotation>
      <xsd:documentation>Max number of occurrences within a
        reference time frame
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element maxOccurs="1" minOccurs="0" name="lowestRate"
    type="OperationRate">
    <xsd:annotation>
      <xsd:documentation>Min number of occurrences within a
        reference time frame
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
</xsd:sequence>
<xsd:attribute name="name" type="NameId" use="required"/>
<xsd:attribute name="maxHandlingTime" type="TimeDuration"
  use="optional">
  <xsd:annotation>
    <xsd:documentation>Event Sent : specifies an intent on receivers
      for maximal duration between Event Reception and end of
      related processing
      Event Received : maximal duration between

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        Event Received and end of related processing.
        Unit is second.
    </xsd:documentation>
</xsd:annotation>
</xsd:attribute>
</xsd:complexType>
<xsd:complexType name="RequestResponse">
    <xsd:annotation>
        <xsd:documentation>Use of the "request-reply" exchange mechanism.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element maxOccurs="1" minOccurs="0" name="highestRate"
            type="OperationRate">
            <xsd:annotation>
                <xsd:documentation>Max number of occurrences within a
                    reference time frame
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
        <xsd:element maxOccurs="1" minOccurs="0" name="lowestRate"
            type="OperationRate">
            <xsd:annotation>
                <xsd:documentation>Min number of occurrences within a
                    reference time frame
                </xsd:documentation>
            </xsd:annotation>
        </xsd:element>
    </xsd:sequence>
    <xsd:attribute name="name" type="NameId" use="required"/>
    <xsd:attribute name="maxResponseTime" type="TimeDuration"
        use="optional">
        <xsd:annotation>
            <xsd:documentation>Operation Provided : maximal duration between
                Request Reception and Callback Sent
                Operation Required : maximal duration between Request Sent
                and Callback reception.
                Unit is second.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:attribute>
    <xsd:attribute name="callbackMaxHandlingTime" type="TimeDuration"
        use="optional">
        <xsd:annotation>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    <xsd:documentation>maxHandlingTime to execute the callback
    entry-point.
    Unit is second.
  </xsd:documentation>
</xsd:annotation>
</xsd:attribute>
</xsd:complexType>
<xsd:complexType name="OperationRate">
  <xsd:attribute name="numberOfOccurrences" type="xsd:decimal"
    use="optional">
    <xsd:documentation>Min or max number of operations occurring
    during a specified duration
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="timeFrame" type="TimeDuration"
    use="optional">
    <xsd:documentation>Equal to min or max inter-arrival time when
    NumberOfOccurrences value is 1.
    In other cases, specifies a sizing duration for operations
    bursts.
    Unit is second.
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
</xsd:complexType>
</xsd:schema>

```

## 8.11 ecoa-logicalsysteM-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/logicalsysteM-2.0"
  xmlns:tns="http://www.ecoa.technology/logicalsysteM-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.ecoa.technology/logicalsysteM-2.0">
  <xsd:include schemaLocation="ecoa-common-2.0.xsd" />

  <xsd:element name="logicalSystem" type="LogicalSystem">
    <xsd:key name="platformIdKey">
      <xsd:selector xpath="./logicalComputingPlatform" />
      <xsd:field xpath="@id" />
    </xsd:key>
  </xsd:element>
</xsd:schema>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:unique name="ELIPlatformIdKey">
  <xsd:selector xpath="./logicalComputingPlatform" />
  <xsd:field xpath="@ELIPlatformId" />
</xsd:unique>
</xsd:element>
<xsd:complexType name="LogicalSystem">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" name="logicalComputingPlatform"
      type="LogicalComputingPlatform">
      <xsd:key name="nodeIdKey">
        <xsd:selector xpath="./logicalComputingNode" />
        <xsd:field xpath="@id" />
      </xsd:key>
    </xsd:element>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="logicalComputingPlatformLinks"
      type="LogicalComputingPlatformLinks">
      <xsd:unique name="computingPlatformLinkIdKey">
        <xsd:selector xpath="./link" />
        <xsd:field xpath="@id" />
      </xsd:unique>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="id" type="NameId" use="required" />
</xsd:complexType>

<xsd:complexType name="LogicalComputingPlatform">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="logicalComputingNode">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="endianess">
            <xsd:complexType>
              <xsd:attribute name="type" use="required">
                <xsd:simpleType>
                  <xsd:restriction base="xsd:string">
                    <xsd:enumeration value="BIG" />
                    <xsd:enumeration value="LITTLE" />
                  </xsd:restriction>
                </xsd:simpleType>
              </xsd:attribute>
            </xsd:complexType>
          </xsd:element>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



```

<xsd:element maxOccurs="unbounded" name="logicalProcessors">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="stepDuration">
        <xsd:complexType>
          <xsd:attribute name="nanoSeconds" type="xsd:integer"
            use="required" />
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
    <xsd:attribute name="type" type="xsd:string" use="required" />
    <xsd:attribute name="number" type="xsd:integer"
      use="required" />
  </xsd:complexType>
</xsd:element>
<xsd:element name="os">
  <xsd:complexType>
    <xsd:attribute name="name" use="required">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="fastos" />
          <xsd:enumeration value="linux" />
          <xsd:enumeration value="ima-integrity" />
          <xsd:enumeration value="ims-vxworks" />
          <xsd:enumeration value="integrity" />
          <xsd:enumeration value="pikeos" />
          <xsd:enumeration value="rtems" />
          <xsd:enumeration value="vxworks" />
          <xsd:enumeration value="vxworks-arinc653" />
          <xsd:enumeration value="vxworks-cert" />
          <xsd:enumeration value="windows" />
          <xsd:enumeration value="zephyr" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="version" type="xsd:string"
      use="optional" />
  </xsd:complexType>
</xsd:element>
<xsd:element name="availableMemory">
  <xsd:complexType>
    <xsd:attribute name="gigaBytes" type="xsd:integer"
      use="required" />
  </xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:element>
    <xsd:element name="moduleSwitchTime">
      <xsd:complexType>
        <xsd:attribute name="microSeconds" type="xsd:integer"
          use="required" />
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:string" use="required" />
</xsd:complexType>
</xsd:element>
<xsd:element maxOccurs="unbounded" minOccurs="0"
  name="logicalComputingNodeLinks" type="LogicalComputingNodeLinks">
  <xsd:unique name="nodeLinkIdKey">
    <xsd:selector xpath="./link" />
    <xsd:field xpath="@id" />
    <xsd:field xpath="@to" />
    <xsd:field xpath="@from" />
  </xsd:unique>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="id" type="NameId" use="required" />
<xsd:attribute name="ELIPlatformId" type="xsd:unsignedInt"
  use="optional">
  <xsd:annotation>
    <xsd:documentation>
      Define the Id to be used as Logical ELI Platform ID
      in the ELI generic header
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>

<xsd:complexType name="LogicalComputingNodeLinks">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" name="link">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="throughput" minOccurs="0" maxOccurs="1">
            <xsd:complexType>
              <xsd:attribute name="megaBytesPerSecond" type="xsd:integer"
                use="required" />
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:element>
    <xsd:element name="latency" minOccurs="0" maxOccurs="1">
      <xsd:complexType>
        <xsd:attribute name="microSeconds" type="xsd:integer"
          use="required" />
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="id" type="NameId" />
  <xsd:attribute name="to" type="xsd:string" use="required" />
  <xsd:attribute name="from" type="xsd:string" use="required" />
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="LogicalComputingPlatformLinks">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" name="link">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="throughput" minOccurs="0" maxOccurs="1">
            <xsd:complexType>
              <xsd:attribute name="megaBytesPerSecond" type="xsd:integer"
                use="required" />
            </xsd:complexType>
          </xsd:element>
          <xsd:element name="latency" minOccurs="0" maxOccurs="1">
            <xsd:complexType>
              <xsd:attribute name="microSeconds" type="xsd:integer"
                use="required" />
            </xsd:complexType>
          </xsd:element>
          <xsd:element name="transportBinding" minOccurs="0"
            maxOccurs="1">
            <xsd:annotation>
              <xsd:documentation>
                Describe on which transport protocol the logical link is
associated
                to
              </xsd:documentation>
            </xsd:annotation>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        <xsd:attribute name="protocol" type="xsd:string"
            use="required" />
        <xsd:attribute name="parameters" type="xsd:anyURI"
            use="required" />
    </xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="id" type="NameId" use="required" />
<xsd:attribute name="to" type="xsd:string" use="required" />
<xsd:attribute name="from" type="xsd:string" use="required" />
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:schema>

```

## 8.12 ecoa-types-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/types-2.0"
    xmlns:tns="http://www.ecoa.technology/types-2.0"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="qualified"
    targetNamespace="http://www.ecoa.technology/types-2.0">
    <xsd:include schemaLocation="ecoa-common-2.0.xsd"/>
    <xsd:simpleType name="E_basic">
        <xsd:annotation>
            <xsd:documentation>ECOA basic types</xsd:documentation>
        </xsd:annotation>
        <xsd:restriction base="xsd:string">
            <xsd:enumeration value="boolean8"/>
            <xsd:enumeration value="int8"/>
            <xsd:enumeration value="int16"/>
            <xsd:enumeration value="int32"/>
            <xsd:enumeration value="int64"/>
            <xsd:enumeration value="uint8"/>
            <xsd:enumeration value="uint16"/>
            <xsd:enumeration value="uint32"/>
            <xsd:enumeration value="uint64"/>
            <xsd:enumeration value="char8"/>
            <xsd:enumeration value="byte"/>
            <xsd:enumeration value="float32"/>
            <xsd:enumeration value="double64"/>
        </xsd:restriction>
    </xsd:simpleType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="Simple">
  <xsd:annotation>
    <xsd:documentation>A type based on a predefined type (simple or
      E_basic) with a name, min/max constraints, and a unit.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="type" type="TypeQName" use="required"/>
  <xsd:attribute name="name" type="TypeName" use="required"/>
  <xsd:attribute name="minRange" type="ConstantReferenceOrValue"
    use="optional"/>
  <xsd:attribute name="maxRange" type="ConstantReferenceOrValue"
    use="optional"/>
  <xsd:attribute name="unit" type="xsd:string" use="optional">
    <xsd:annotation>
      <xsd:documentation>Use of International System units is
        recommended.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="precision" type="ConstantReferenceOrValue"
    use="optional">
    <xsd:annotation>
      <xsd:documentation>Precision of values
    </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="Constant">
  <xsd:annotation>
    <xsd:documentation>Constant definition</xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="name" type="TypeName" use="required"/>
  <xsd:attribute name="type" type="TypeQName" use="required"/>
  <xsd:attribute name="value" type="ConstantReferenceOrValue"
    use="required"/>
  <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:simpleType name="ConstantValue">
  <xsd:annotation>
    <xsd:documentation>Type allowing inputs of decimal values and
      character values

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    </xsd:documentation>
  </xsd:annotation>
  <xsd:union
    memberTypes="xsd:double xsd:integer CharConstantType
HexaCharConstantType"/>
</xsd:simpleType>
<xsd:simpleType name="CharConstantType">
  <xsd:annotation>
    <xsd:documentation>Character Constant Type</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="\p{IsBasicLatin}"/>
    <xsd:length value="1"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="HexaCharConstantType">
  <xsd:annotation>
    <xsd:documentation>This type shall be used as the base type for
any XML schema attribute or element that contains a hex-encoded
char value.
This hex-encoded binary value contains the mandatory
string 0x followed by a sequence of 1 or 2 characters 0-9 and
a-f.
</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="0x[0-9A-Fa-f][0-9A-Fa-f]|0x[0-9A-Fa-f]"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="ConstantReferenceOrValue">
  <xsd:annotation>
    <xsd:documentation>Use of a constant reference or of any constant
value.
Note: the ConstantValue type is replaced by its flattened
union members.
</xsd:documentation>
  </xsd:annotation>
  <xsd:union
    memberTypes="ConstantReference xsd:double xsd:integer CharConstantType
HexaCharConstantType"/>
</xsd:simpleType>
<xsd:simpleType name="ConstantReference">
  <xsd:annotation>
    <xsd:documentation>Constant reference
  </xsd:documentation>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

</xsd:annotation>
<xsd:restriction base="xsd:string">
  <xsd:pattern value="%([A-Za-z][A-Za-z0-9_\.]*)?[A-Za-z][A-Za-z0-9_]*%"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="ConstantReferenceOrPositiveIntegerValue">
  <xsd:annotation>
    <xsd:documentation>Use of a constant or of a positive integer
      value.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:pattern
      value="%([A-Za-z][A-Za-z0-9_\.]*)?[A-Za-z][A-Za-z0-9_]*|[0-9]+"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="ConstantReferenceOrIntegerValue">
  <xsd:annotation>
    <xsd:documentation>Use of a constant or of an integer value.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:pattern
      value="%([A-Za-z][A-Za-z0-9_\.]*)?[A-Za-z][A-Za-z0-9_]*|(\+|-)?[0-9]*"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="Enum">
  <xsd:annotation>
    <xsd:documentation>Enumerated type</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" name="value" type="EnumValue"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="TypeName" use="required"/>
  <xsd:attribute name="type" type="TypeQName" use="required"/>
  <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="EnumValue">
  <xsd:annotation>
    <xsd:documentation>A possible value of an enumerated type
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="name" type="TypeName" use="required"/>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:attribute name="valnum" type="ConstantReferenceOrIntegerValue"
  use="optional"/>
<xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="FixedArray">
  <xsd:annotation>
    <xsd:documentation>Fixed-size array (size is always equal to max
      capacity)
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="itemType" type="TypeQName" use="required"/>
  <xsd:attribute name="maxNumber"
    type="ConstantReferenceOrPositiveIntegerValue" use="required"/>
  <xsd:attribute name="name" type="TypeName" use="required"/>
  <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="Array">
  <xsd:annotation>
    <xsd:documentation>Variable-size (bounded capacity) array
    </xsd:documentation>
  </xsd:annotation>
  <xsd:attribute name="itemType" type="TypeQName" use="required"/>
  <xsd:attribute name="maxNumber"
    type="ConstantReferenceOrPositiveIntegerValue" use="required"/>
  <xsd:attribute name="name" type="TypeName" use="required"/>
  <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="Record">
  <xsd:annotation>
    <xsd:documentation>A record with named fields (Ada record, C
      struct)
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" name="field" type="Field"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="TypeName" use="required"/>
  <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="Field">
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="type" type="TypeQName" use="required"/>
  <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



```

<xsd:complexType name="VariantRecord">
  <xsd:annotation>
    <xsd:documentation>A record with variable parts: each "union"
      exists only if the selector has the value given by the "when"
      attribute.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0" name="field"
      type="Field"/>
    <xsd:element maxOccurs="unbounded" name="union" type="Union"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="TypeName" use="required"/>
  <xsd:attribute name="selectName" type="NameId" use="required"/>
  <xsd:attribute name="selectType" type="TypeQName" use="required"/>
  <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="Union">
  <xsd:attribute name="name" type="NameId" use="required"/>
  <xsd:attribute name="type" type="TypeQName" use="required"/>
  <xsd:attribute name="when" type="xsd:string" use="required"/>
  <xsd:attribute name="comment" type="xsd:string" use="optional"/>
</xsd:complexType>
<xsd:complexType name="DataTypes">
  <xsd:annotation>
    <xsd:documentation>A set of data type definitions
    </xsd:documentation>
  </xsd:annotation>
  <xsd:choice maxOccurs="unbounded" minOccurs="0">
    <xsd:element name="simple" type="Simple"/>
    <xsd:element name="record" type="Record">
      <xsd:unique name="field">
        <xsd:selector xpath="tns:field"/>
        <xsd:field xpath="@name"/>
      </xsd:unique>
    </xsd:element>
    <xsd:element name="constant" type="Constant"/>
    <xsd:element name="variantRecord" type="VariantRecord">
      <xsd:unique name="fieldunion">
        <xsd:selector xpath="tns:field|tns:union"/>
        <xsd:field xpath="@name"/>
      </xsd:unique>
    </xsd:element>
    <xsd:element name="array" type="Array"/>
  </xsd:choice>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:element name="fixedArray" type="FixedArray"/>
<xsd:element name="enum" type="Enum">
  <xsd:unique name="value">
    <xsd:selector xpath="tns:value"/>
    <xsd:field xpath="@name"/>
  </xsd:unique>
  <xsd:unique name="valnum">
    <xsd:selector xpath="tns:value"/>
    <xsd:field xpath="@valnum"/>
  </xsd:unique>
</xsd:element>
</xsd:choice>
</xsd:complexType>
<xsd:complexType name="Library">
  <xsd:annotation>
    <xsd:documentation>A set of data types in a library
  </xsd:documentation>
</xsd:annotation>
<xsd:sequence>
  <xsd:element maxOccurs="unbounded" minOccurs="0" ref="use"/>
  <xsd:element name="types" type="DataTypes">
    <xsd:unique name="typename">
      <xsd:selector xpath="*"/>
      <xsd:field xpath="@name"/>
    </xsd:unique>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:element name="library" type="Library"/>
</xsd:schema>

```

### 8.13 ecoa-project-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/project-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/project-2.0">
  <xsd:include schemaLocation="ecoa-common-2.0.xsd"/>

  <xsd:element name="ECOAProject" type="EcoaProject"/>

  <xsd:complexType name="EcoaProject">
    <xsd:annotation>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:documentation>
  Describes a whole ECOA project
</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
  <xsd:choice maxOccurs="unbounded" minOccurs="0">
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="serviceDefinitions" type="Files"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="componentDefinitions" type="Files"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="types" type="Files"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="initialAssembly" type="xsd:anyURI"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="componentImplementations" type="Files"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="logicalSystem" type="xsd:anyURI"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="crossPlatformsView" type="xsd:anyURI"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="deploymentSchema" type="xsd:anyURI"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="outputDirectory" type="xsd:anyURI"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="implementationAssembly" type="xsd:anyURI"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="EUIDs" type="ELI_EUIDs"/>
  </xsd:choice>
</xsd:sequence>
  <xsd:attribute name="name" type="xsd:string" use="required"/>
</xsd:complexType>

<xsd:complexType name="Files">
  <xsd:annotation>
    <xsd:documentation>List of files</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" name="file" type="xsd:anyURI"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ELI_EUIDs">
  <xsd:annotation>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    <xsd:documentation>List of bindings</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="1" name="EUID"
      type="xsd:anyURI"/>
  </xsd:sequence>
</xsd:complexType>

</xsd:schema>

```

## 8.14 ecoa-cross-platforms-view-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/cross-platforms-view-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/cross-platforms-view-2.0">
  <xsd:include schemaLocation="ecoa-common-2.0.xsd"/>

  <xsd:element name="view" type="View"/>

  <xsd:complexType name="View">
    <xsd:annotation>
      <xsd:documentation>
        Describes how composites are mapped onto
        platforms, how wires
        are mapped onto logical links and how IDs
        are bound to logical links
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:choice maxOccurs="unbounded" minOccurs="0">
        <xsd:element maxOccurs="unbounded" minOccurs="0"
          name="composite" type="Composite">
          <xsd:annotation>
            <xsd:documentation>Defines where a composite is executed
          </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
        <xsd:element maxOccurs="unbounded" minOccurs="0"
          name="wireMapping" type="WireMapping">
          <xsd:annotation>
            <xsd:documentation>Defines the mapping of a wires onto a
              logical
              platform link
            </xsd:documentation>
          </xsd:annotation>
        </xsd:element>
      </xsd:choice>
    </xsd:sequence>
  </xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
<xsd:element maxOccurs="unbounded" minOccurs="0"
    name="euidsBinding" type="EUIDsBinding">
    <xsd:annotation>
        <xsd:documentation>Defines the binding of EUIDs onto a
            logical platform link
        </xsd:documentation>
    </xsd:annotation>
</xsd:element>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="assembly" type="NameId" use="required">
    <xsd:annotation>
        <xsd:documentation>Name of the system composite referenced by
            this view
        </xsd:documentation>
    </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="logicalSystem" type="NameId" use="required">
    <xsd:annotation>
        <xsd:documentation>Name of the logical system this deployment is
            made on
        </xsd:documentation>
    </xsd:annotation>
</xsd:attribute>
</xsd:complexType>

<xsd:complexType name="Composite">
    <xsd:attribute name="name" type="NameId" use="required"/>
    <xsd:attribute name="deployedOnComputingPlatform"
        type="NameId" use="required">
        <xsd:annotation>
            <xsd:documentation>Name of a logical platform
        </xsd:documentation>
        </xsd:annotation>
    </xsd:attribute>
</xsd:complexType>

<xsd:complexType name="WireMapping">
    <xsd:attribute name="source" type="xsd:anyURI" use="required">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<xsd:annotation>
  <xsd:documentation>
    wire source
  </xsd:documentation>
</xsd:annotation>
</xsd:attribute>
<xsd:attribute name="target" type="xsd:anyURI" use="required">
  <xsd:annotation>
    <xsd:documentation>
      wire target
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="mappedOnLinkId" type="NameId"
  use="required">
  <xsd:annotation>
    <xsd:documentation>
      reference an inter-platform link
      identified in
      the logical system
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>

<xsd:complexType name="EUIDsBinding">
  <xsd:sequence>
    <xsd:element maxOccurs="unbounded" minOccurs="0"
      name="specificBinding">
      <xsd:annotation>
        <xsd:documentation>
          specific EUIDS file associated to one given
          peer of the link
        </xsd:documentation>
      </xsd:annotation>
      <xsd:complexType>
        <xsd:attribute name="EUIDs" type="xsd:anyURI"
          use="required">
          <xsd:annotation>
            <xsd:documentation>
              specific EUIDS file associated to one
              given peer of the link
            </xsd:documentation>
          </xsd:annotation>
        </xsd:complexType>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

</xsd:attribute>
<xsd:attribute name="boundToComputingPlatform"
  type="NameId" use="required">
  <xsd:annotation>
    <xsd:documentation>Name of a logical platform
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="EUIDs" type="xsd:anyURI" use="required">
  <xsd:annotation>
    <xsd:documentation>
      Bind an EUIDS file to a given logical
      computing platform link
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
<xsd:attribute name="boundToLinkId" type="NameId" use="required">
  <xsd:annotation>
    <xsd:documentation>
      Reference an inter-platform link
      identified in
      the logical system
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
</xsd:complexType>
</xsd:schema>

```

## 8.15 ecoa-uid-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="http://www.ecoa.technology/uid-2.0"
  xmlns:tns="http://www.ecoa.technology/uid-2.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  targetNamespace="http://www.ecoa.technology/uid-2.0">

  <xsd:element name="ID_map" type="ID_map">
    <!-- each "key" attribute shall be unique -->
    <xsd:key name="key">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        <xsd:selector xpath="tns:ID"/>
        <xsd:field xpath="@key"/>
    </xsd:key>
</xsd:element>

<xsd:complexType name="ID_map">
    <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="0" name="ID"
            type="ID"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ID">
    <xsd:attribute name="key" type="xsd:string" use="required"/>
    <xsd:attribute name="value" type="xsd:int" use="required"/>
</xsd:complexType>

</xsd:schema>

```

## 8.16 sca-1.1-cd06-subset-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!-- Copyright (C) OASIS (R) 2005,2010. All Rights Reserved. OASIS trademark, IPR
and other policies apply. -->
<!-- This file is a derivative work of the original OASIS XSD file: -->
<!-- sca-1.1-cd06.xsd -->
<!-- In that sense, ECOA is not fully compliant with the OASIS SCA
specification. -->
<!-- This file is provided to help users to check more easily their ECOA XML
files. -->
<!-- Other verification means may be used. -->

<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:sca="http://docs.oasis-
open.org/ns/opencsa/sca/200912"
    targetNamespace="http://docs.oasis-open.org/ns/opencsa/sca/200912">

    <include schemaLocation="sca-implementation-composite-1.1-cd06-subset-
2.0.xsd"/>
    <include schemaLocation="sca-core-1.1-cd06-subset-2.0.xsd"/>

    <include schemaLocation="sca-contribution-1.1-cd06-subset-2.0.xsd"/>

</schema>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



## 8.17 sca-contribution-1.1-cd06-subset-2.0.xsd

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!-- Copyright(C) OASIS(R) 2005,2010. All Rights Reserved. OASIS trademark, IPR
and other policies apply. -->
<!-- This file is a derivative work of the original OASIS XSD file: -->
<!-- sca-contribution-1.1-cd06.xsd -->
<!-- In that sense, ECOA is not fully compliant with the OASIS SCA
specification. -->
<!-- This file is provided to help users to check more easily their ECOA XML
files. -->
<!-- Other verification means may be used. -->

<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:ecoa="http://www.ecoa.technology/sca-extension-2.0"
xmlns:sca="http://docs.oasis-open.org/ns/opencsa/sca/200912"
  elementFormDefault="qualified" targetNamespace="http://docs.oasis-
open.org/ns/opencsa/sca/200912">

  <import namespace="http://www.ecoa.technology/sca-extension-2.0"
schemaLocation="ecoa-sca-2.0.xsd"/>

  <include schemaLocation="sca-core-1.1-cd06-subset-2.0.xsd"/>

  <!-- Contribution -->
  <element name="contribution" type="sca:ContributionType"/>
  <complexType name="ContributionType">
    <complexContent>
      <extension base="sca:CommonExtensionBase">
        <sequence>
          <element maxOccurs="unbounded" minOccurs="0" name="deployable"
type="sca:DeployableType"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>

  <!-- Deployable -->
  <complexType name="DeployableType">
    <complexContent>
      <extension base="sca:CommonExtensionBase">
        <sequence>
          <any maxOccurs="unbounded" minOccurs="0" namespace="##other"
processContents="lax"/>
        </sequence>
        <attribute ref="ecoa:deployment"/>
      </extension>
    </complexContent>
  </complexType>
</schema>
```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        <attribute name="composite" type="QName" use="required"/>
    </extension>
</complexContent>
</complexType>

</schema>

```

## 8.18 sca-core-1.1-cd06-subset-2.0.xsd

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!-- Copyright(C) OASIS(R) 2005,2010. All Rights Reserved. OASIS trademark, IPR
and other
    policies apply. -->
<!-- This file is a derivative work of the original OASIS XSD file: -->
<!-- sca-core-1.1-cd06.xsd -->
<!-- In that sense, ECOA is not fully compliant with the OASIS SCA
specification. -->
<!-- This file is provided to help users to check more easily their ECOA XML
files. -->
<!-- Consequently, this file also features some ECOA specific attributes added
in some places into
SCA complex types, in order to allow automated XML validation. However, even
without these custom XSD
attributes ECOA XML files can be validated against the original OASIS sca-core-
1.1-cd06.xsd -->
<!-- Other verification means may be used. -->

<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:ecoa="http://www.ecoa.technology/sca-extension-2.0"
xmlns:jxb="http://java.sun.com/xml/ns/jaxb"
    xmlns:sca="http://docs.oasis-open.org/ns/opencsa/sca/200912"
elementFormDefault="qualified" jxb:version="1.0"
    targetNamespace="http://docs.oasis-open.org/ns/opencsa/sca/200912">

    <import namespace="http://www.w3.org/XML/1998/namespace"
schemaLocation="../../../xml/xml.xsd"/>

    <!-- Workaround to allow within Eclipse the validation of .composite -->
    <import namespace="http://www.ecoa.technology/sca-extension-2.0"
schemaLocation="ecoa-sca-2.0.xsd"/>

    <!-- Common extension base for SCA definitions -->
    <complexType name="CommonExtensionBase">
        <sequence>
            <element maxOccurs="unbounded" minOccurs="0" ref="sca:documentation"/>
        </sequence>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<!-- Note that the <anyAttribute namespace="##other" processContents="lax"/>
originally present in the original OASIS sca-core-1.1-cd06.xsd file is not
contained in this subset file for ECOA. However, when replacing this subset
file with the original sca-core-1.1-cd06.xsd file, the presence of the
<anyAttribute namespace="##other" processContents="lax"/> in the
CommonExtensionBase complex type allows validating ECOA XML files which
feature ECOA custom attributes in some places. This guarantees the
compatibility between ECOA XML files and SCA. -->
</complexType>

<element name="documentation" type="sca:Documentation"/>
<complexType mixed="true" name="Documentation">
  <sequence>
    <any maxOccurs="unbounded" minOccurs="0" namespace="##other"
processContents="lax"/>
  </sequence>
  <attribute ref="xml:lang"/>
</complexType>
<!-- Component Type -->
<element name="componentType" type="sca:ComponentType"/>
<complexType name="ComponentType">
  <complexContent>
    <extension base="sca:CommonExtensionBase">
      <sequence>
        <choice maxOccurs="unbounded" minOccurs="0">
          <element name="service">
            <complexType>
              <complexContent>
                <restriction base="sca:ComponentService">
                  <sequence>
                    <sequence>
                      <!-- Note: when replacing this
sca-core-1.1-cd06-subset.xsd file with the original
OASIS sca-core-1.1-cd06.xsd file, which contains
sca:interface in place of the following ecoa:interface,
ECOA XMLs are still validated against SCA thanks to
the ecoa-sca-interface-2.0.xsd which extends the
abstract sca:interface with ecoa:interface.
The following explicit substitution is done here
only to help XSD validation for ECOA. -->
                      <element ref="ecoa:interface"/>
                    </sequence>
                  </sequence>
                </restriction>
              </complexContent>
            </complexType>
          </element>
        </choice>
      </sequence>
    </extension>
  </complexContent>
</complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        </complexContent>
    </complexType>
</element>
<element name="reference" type="sca:ComponentTypeReference"/>
<element name="property" type="sca:Property"/>
</choice>
</sequence>
</extension>
</complexContent>
</complexType>
<!-- Composite -->
<element name="composite" type="sca:Composite"/>
<complexType name="Composite">
    <complexContent>
        <extension base="sca:CommonExtensionBase">
            <sequence>
                <choice maxOccurs="unbounded" minOccurs="0">
                    <element name="service" type="sca:Service"/>
                    <element name="property" type="sca:Property"/>
                    <element name="component" type="sca:Component"/>
                    <element name="reference" type="sca:Reference"/>
                    <element name="wire" type="sca:Wire"/>
                </choice>
            </sequence>
            <attribute name="name" type="NCName" use="required"/>
            <attribute name="targetNamespace" type="anyURI" use="required"/>
        </extension>
    </complexContent>
</complexType>
<!-- Contract base type for Service, Reference -->
<complexType abstract="true" name="Contract">
    <complexContent>
        <extension base="sca:CommonExtensionBase">
            <sequence>
                <element minOccurs="0" ref="ecoa:interface"/>
            </sequence>
            <attribute name="name" type="NCName" use="required"/>
        </extension>
    </complexContent>
</complexType>
<!-- Service -->
<complexType name="Service">
    <complexContent>
        <extension base="sca:Contract">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        <attribute name="promote" type="anyURI" use="required"/>
    </extension>
</complexContent>
</complexType>
<!-- Interface -->
<element abstract="true" name="interface" type="sca:Interface"/>
<complexType abstract="true" name="Interface">
    <complexContent>
        <extension base="sca:CommonExtensionBase"/>
    </complexContent>
</complexType>
<!-- Reference -->
<complexType name="Reference">
    <complexContent>
        <extension base="sca:Contract">
            <attribute name="multiplicity" type="sca:Multiplicity" use="required"/>
            <!-- The promote attribute is a standard SCA attribute used on a
                composite reference, in order to provide the list of component
                references which are promoted by this composite reference -->
            <attribute name="promote" type="sca:listOfAnyURIs" use="required"/>
        </extension>
    </complexContent>
</complexType>
<!-- Property -->
<complexType mixed="true" name="SCAPropertyBase">
    <sequence>
        <any maxOccurs="unbounded" minOccurs="0" namespace="##any"
processContents="lax"/>
        <!-- NOT an extension point; This any exists to accept the element-based
            or complex type property i.e. no element-based extension point under
            "sca:property" -->
    </sequence>
    <attribute name="name" type="NCName" use="required"/>
    <attribute name="type" type="QName" use="optional"/>
    <anyAttribute namespace="##other" processContents="lax"/>
</complexType>
<complexType mixed="true" name="Property">
    <complexContent mixed="true">
        <extension base="sca:SCAPropertyBase">
            <attribute default="false" name="mustSupply" type="boolean"
use="optional"/>
        </extension>
    </complexContent>
</complexType>
<complexType mixed="true" name="PropertyValue">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

<complexContent mixed="true">
  <extension base="sca:SCAPropertyBase">
    <attribute name="source" type="string" use="optional"/>
    <attribute name="file" type="anyURI" use="optional"/>
  </extension>
</complexContent>
</complexType>
<!-- WireFormat Type -->
<element abstract="true" name="wireFormat" type="sca:WireFormatType"/>
<complexType abstract="true" name="WireFormatType">
  <!-- <anyAttribute namespace="##other" processContents="lax"/> -->
</complexType>
<!-- Component -->
<complexType name="Component">
  <complexContent>
    <extension base="sca:CommonExtensionBase">
      <sequence>
        <element ref="sca:implementation"/>
        <choice maxOccurs="unbounded" minOccurs="0">
          <element name="service" type="sca:ComponentService"/>
          <element name="reference" type="sca:ComponentReference"/>
          <element name="property" type="sca:PropertyValue"/>
          <!-- <element ref="sca:requires"/> -->
          <!-- <element ref="sca:policySetAttachment"/> -->
        </choice>
        <!-- <element ref="sca:extensions" minOccurs="0" maxOccurs="1" /> -->
      </sequence>
      <attribute name="name" type="NCName" use="required"/>
    </extension>
    <!-- <attribute name="autowire" type="boolean" use="optional"/> -->
    <!-- <attribute name="requires" type="sca:listOfQNames" -->
    <!-- use="optional"/> -->
    <!-- <attribute name="policySets" type="sca:listOfQNames" -->
    <!-- use="optional"/> -->
  </complexContent>
</complexType>
<!-- Component Service -->
<complexType name="ComponentService">
  <complexContent>
    <extension base="sca:Contract"/>
  </complexContent>
</complexType>
<!-- Component Reference -->
<complexType name="ComponentReference">

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

    <complexContent>
      <extension base="sca:Contract">
        <attribute default="1..1" name="multiplicity" type="sca:Multiplicity"
use="optional"/>
      </extension>
    </complexContent>
  </complexType>
  <!-- Component Type Reference -->
  <complexType name="ComponentTypeReference">
    <complexContent>
      <restriction base="sca:ComponentReference">
        <sequence>
          <element maxOccurs="unbounded" minOccurs="0"
ref="sca:documentation"/>
          <element ref="ecoa:interface"/>
        </sequence>
        <attribute name="name" type="NCName" use="required"/>
        <attribute default="1..1" name="multiplicity" type="sca:Multiplicity"
use="optional"/>
      </restriction>
    </complexContent>
  </complexType>
  <!-- Implementation -->
  <element abstract="true" name="implementation" type="sca:Implementation"/>
  <complexType abstract="true" name="Implementation">
    <complexContent>
      <extension base="sca:CommonExtensionBase"/>
    </complexContent>
  </complexType>
  <!-- Implementation Type -->
  <element name="implementationType" type="sca:ImplementationType"/>
  <complexType name="ImplementationType">
    <complexContent>
      <extension base="sca:CommonExtensionBase">
        <attribute name="type" type="QName" use="required"/>
      </extension>
    </complexContent>
  </complexType>
  <!-- Wire -->
  <complexType name="Wire">
    <complexContent>
      <extension base="sca:CommonExtensionBase">
        <sequence>
          <any maxOccurs="unbounded" minOccurs="0" namespace="##other"
processContents="lax"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>

```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

```

        </sequence>
        <attribute name="source" type="anyURI" use="required"/>
        <attribute name="target" type="anyURI" use="required"/>
    </extension>
</complexContent>
</complexType>
<!-- Extensions element -->
<element name="extensions">
    <complexType>
        <sequence>
            <any maxOccurs="unbounded" minOccurs="1" namespace="##other"
processContents="lax"/>
        </sequence>
    </complexType>
</element>
<!-- Value type definition for property values -->
<element name="value" type="sca:ValueType"/>
<complexType mixed="true" name="ValueType">
    <sequence>
        <any maxOccurs="unbounded" minOccurs="0" namespace="##any"
processContents="lax"/>
    </sequence>
    <anyAttribute namespace="##any" processContents="lax"/>
</complexType>
<!-- Miscellaneous simple type definitions -->
<simpleType name="Multiplicity">
    <restriction base="string">
        <enumeration value="0..1"/>
        <enumeration value="1..1"/>
        <enumeration value="0..n"/>
        <enumeration value="1..n"/>
    </restriction>
</simpleType>
<simpleType name="listOfQNames">
    <list itemType="QName"/>
</simpleType>
<simpleType name="listOfAnyURIs">
    <list itemType="anyURI"/>
</simpleType>
</schema>

```

## 8.19 sca-implementation-composite-1.1-cd06-subset-2.0.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
```

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.



```

<!-- Copyright(C) OASIS(R) 2005,2010. All Rights Reserved.
      OASIS trademark, IPR and other policies apply. -->
<!-- This file is a derivative work of the original OASIS XSD file: -->
<!-- sca-implementation-composite-1.1-cd06.xsd -->
<!-- In that sense, ECOA is not fully compliant with the OASIS SCA
specification. -->
<!-- This file is provided to help users to check more easily their ECOA XML
files. -->
<!-- Other verification means may be used. -->
<schema xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:sca="http://docs.oasis-open.org/ns/opencsa/sca/200912"
  targetNamespace="http://docs.oasis-open.org/ns/opencsa/sca/200912"
  elementFormDefault="qualified">

  <include schemaLocation="sca-core-1.1-cd06-subset-2.0.xsd"/>

  <!-- Composite Implementation -->
  <element name="implementation.composite" type="sca:SCAImplementation"
    substitutionGroup="sca:implementation"/>
  <complexType name="SCAImplementation">
    <complexContent>
      <extension base="sca:Implementation">
        <sequence>
          <any namespace="##other" processContents="lax" minOccurs="0"
            maxOccurs="unbounded"/>
        </sequence>
        <attribute name="name" type="QName" use="required"/>
      </extension>
    </complexContent>
  </complexType>

</schema>

```

## 9 Specifications of the SCA Subset

This section describes for each file of the SCA Assembly metamodel (see 6.2.2 and ref. SCA 1.1) the elements used by ECOA.

Tooling may either explicitly check they are present in any ECOA XML file or use XSD subsets provided in the ECOA schemas pack (sca-1.1-cd06-subset.xsd, sca-core-1.1-cd06-subset.xsd and sca-contribution-1.1-cd06-subset.xsd).

### 9.1 sca-1.1-cd06.xsd

Only the include elements related to "[sca-core-1.1-cd06.xsd](#)" and "[sca-contribution-1.1-cd06.xsd](#)" are used.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

All other elements are not used.

## 9.2 sca-contribution-1.1-cd06.xsd

The complexType "ContributionType" is used apart the referenced elements "sca:importBase", "sca:exportBase" and "sca:extensions".

The complexType "DeployableType" is used as it is.

It can be extended with the attribute "ecoa:deployment".

The element "importBase" and the associated complexType "Import" are not used.

The element "import" and the associated complexType "ImportType" are not used.

The element "exportBase" and the associated complexType "Export" are not used.

The element "export" and the associated complexType "ExportType" are not used.

## 9.3 sca-core-1.1-cd06.xsd

The schema "sca-policy-1.1-cd04.xsd" is not used and does not need to be included.

For allowing XML validation with tools such as Eclipse, it may be helpful to import the schema "ecoa-sca-2.0.xsd".

The complexType "CommonExtensionBase" is used apart its generic attribute element "anyAttribute" which can be excluded. However, this generic attribute guarantees the compatibility between ECOA XML files and the original OASIS schema "sca-core-1.1-cd06.xsd", with regard to extensions created in the complexType "Wire" and complexType "Reference".

The element "documentation" and its associated complexType "Documentation" are used as defined in the original file.

The element "componentType" is used as defined in the original file.

The complexType "ComponentType" is used but :

- Its optional element "service" can be restricted to elements of type "ecoa:interface",
- The referred element "sca:extensions" is not used.

The element "composite" is used as defined in the original file.

The complexType "Composite" is used apart:

- the referenced elements "sca:include",
- the optional referenced elements "sca:requires" and "sca:policySetAttachment",
- the element "any",
- the attributes "local", "autowire", "requires" and "policySets".

The complexType "Contract" is used apart:

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

- The referenced elements "sca:binding", "sca:callback", "sca:requires", "sca:policySetAttachment" and "sca:extensions",
- The attributes "required" and "policySets".

The referenced element "sca:interface" can be replaced by the referenced element "ecoa:interface". This explicit substitution is done only to help XSD validation for ECOA. ECOA XML files are compliant with SCA, thanks to the the "ecoa-sca-interface-2.0.xsd" which extends the abstract "sca:interface" through "ecoa:interface".

The complexType "Service" is used as defined in the original file.

The element "interface" is used as defined in the original file.

The complexType "Interface" is used apart:

- The optional referenced elements "sca:requires" and "sca:policySetAttachment",
- The attributes "remotable", "requires" and "policySets".

The complexType "Reference" is used apart its attributes "target" and "wiredByImpl".

The complexType "SCAPropertyBase" is used apart its attributes "element", "many", and "value".

The complexType "Property" is used as defined in the original file.

The complexType "PropertyValue" is used as defined in the original file.

The element "binding" and its associated complexType "Binding" are not used.

The element "bindingType" and its associated complexType "BindingType" are not used.

The element "wireFormat" is used as defined in the original file.

The complexType "WireFormatType" is used apart its generic attribute element "anyAttribute".

The element "operationSelector" and its associated complexType "OperationSelectorType" are not used.

The element "callback" and its associated complexType "Callback" are not used.

The complexType "Component" is used apart:

- Its optional referenced elements "sca:requires" and "sca:policySetAttachment",
- Its referenced elements "sca:extensions"
- Its attributes "autowire", "requires" and "policySets".

The complexType "ComponentService" is used as in the original file.

The complexType "ComponentReference" is used apart its attributes "autowire", "target", "wiredByImpl" and "nonOverridable".

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.

The complexType `ComponentTypeReference` is used apart:

- Its referenced elements "sca:binding", "sca:callback", "sca:requires", "sca:policySetAttachment" and "sca:extensions"
- Its attributes "autowire", "wiredByImpl", "requires" and "policySets"
- Its generic attribute element "anyAttribute".

Its referenced element `sca:interface` can be replaced by the referenced element `ecoa:interface`.

The element `implementation` is used as defined in the original file.

The complexType `Implementation` is used apart:

- Its optional referenced elements `sca:requires` and `sca:policySetAttachment`
- Its attributes `requires` and `policySets`

The element `implementationType` is used as defined in the original file.

The complexType `Implementation` is used apart:

- Its sequence of optional anonymous elements (`any`),
- Its attributes `alwaysProvides` and `mayProvide`

The complexType `wire` is used apart its attribute `replace`.

The element `include` and its associated complexType `Include` are not used.

The element `extensions` is used as defined in the original file.

The general attribute `requires` is not used.

The general attribute `callback` is not used.

The element `value` and its associated complexType `ValueType` are used as defined in the original file.

The simpleType `Multiplicity` is used as defined in the original file.

The simpleType `OverrideOptions` is not used.

The simpleTypes `listOfQNames` and `listOfAnyURIs` are used as they are defined in the original file.

The simpleType `CreateResource` is not used.

---

This specification is developed by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd and the copyright is owned by BAE Systems (Operations) Limited, Dassault Aviation, Bull SAS, Thales Systèmes Aéroportés, GE Aviation Systems Limited, General Dynamics United Kingdom Limited and Leonardo MW Ltd. The information set out in this document is provided solely on an 'as is' basis and co-developers of this specification make no warranties expressed or implied, including no warranties as to completeness, accuracy or fitness for purpose, with respect to any of the information.