European Component Oriented Architecture (ECOA®) Collaboration Programme: Preliminary version of the ECOA Architecture Specification Part 5: High Level Platform Requirements

Dassault Ref No: DGT 2041084-A
Thales DMS Ref No: 69398919-035 --

Issue: 7

Prepared by
Dassault Aviation and Thales DMS

Note: *This specification is preliminary and is subject to further adjustments. Consequently, users are advised to exercise caution when relying on the information herein. No warranties are provided regarding the completeness or accuracy of the information in this preliminary version. The final version of the document will be released to reflect further improvements.*

Without prejudice to the property rights relating to the ECOA AS6 Standard, the information in this document relating to the changes envisaged for the transition from the ECOA AS6 Standard to the ECOA AS7 Standard is the intellectual property of Dassault Aviation and Thales DMS France SAS. 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.

This Page Intentionally Left Blank

This Page Intentionally Left Blank

Contents

Tables

0 Introduction

This Architecture Specification provides the specification for creating ECOA®-based software systems. It describes the standardised programming interfaces and data-model that allow a developer to construct an ECOA®-based software system. It uses terms defined in the Definitions (Architecture Specification Part 2). The details of the other documents comprising the rest of this Architecture Specification can be found in Section 3.

This document is Part 5 of the Architecture Specification, and describes the high level requirements for the conformity of platform to ECOA®.

The ECOA Standard is a modular one, which aggregates several sub-standards:

- The ECOA Architecture Specification (Core + Options),
- ECOA Bindings,
- ECOA Extensions.

This document deals with the Core as well as Options, and gives the complete list of Options currently defined in the Architecture Specification.

The purpose of this document is providing sets of requirements to help ECOA® Platform or ECOA® Reference Platform providers to build the right product and to help system integrators to check the conformance of their procurements.

The document relies on other Reference Manuals of the ECOA® Architecture Specification and refers to them. The assumption is made that any ECOA® Platform is delivered at least with a Toolset, a Version Description and a User's Manual.

The document is structured as follows:

- Section 6 gives the complete list of Options currently defined in the ECOA® Architecture Specification.
- Section 7 describes the generic high level requirements for any ECOA® Platform

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 shall rely on the BNAE publications of the ECOA standard.

3 Normative References

Architecture Specification Part 1	Dassault Ref No: DGT 2041078-A Thales DMS Ref No: 69398915-035 -- Issue 7 Architecture Specification Part 1 – Concepts
Architecture Specification Part 2	Dassault Ref No: DGT 2041081-A Thales DMS Ref No: 69398916-035 -- Issue 7 Architecture Specification Part 2 – Definitions
Architecture Specification Part 3	Dassault Ref No: DGT 2041082-A Thales DMS Ref No: 69398917-035 -- Issue 7 Architecture Specification Part 3 – Mechanisms
Architecture Specification Part 4	Dassault Ref No: DGT 2041083-A Thales DMS Ref No: 69398918-035 -- Issue 7 Architecture Specification Part 4 – Software Interface
Architecture Specification Part 5	Dassault Ref No: DGT 2041084-A Thales DMS Ref No: 69398919-035 -- Issue 7 Architecture Specification Part 5 – High Level Platform Requirements
Architecture Specification Part 6	Dassault Ref No: DGT 2041491-A Thales DMS Ref No: 69398920-035 -- Issue 7 Architecture Specification Part 6 – Options
Architecture Specification Part 7	Dassault Ref No: DGT 2041086-A Thales DMS Ref No: 69398925-035 -- Issue 7 Architecture Specification Part 7 – Metamodel

4 Definitions

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

In addition, in platform requirement, those shown below apply.

NOTE The following definitions are inspired from RFC 2119

4.1 May

Means that an item is truly optional. The item may be part of an Option of the ECOA Architecture Specification, or one platform provider may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item.

4.2 Shall

Means that the definition is an absolute requirement of the ECOA Core specification.

4.3 Should

Means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

5 Abbreviations

API	Application Programming Interface
ECOA	European Component Oriented Architecture. ECOA® is a registered trademark.
OS	Operating System
PINFO	Persistent Information
RFC	Request For Comments
RTOS	Real-Time Operating System
XML	eXtensible Markup Language

6 Options defined in the ECOA Standard

6.1 Modular structure of the Standard

As presented in [Architecture Specification Part 1], the ECOA Standard is a modular one, which aggregates several sub-standards:

- The ECOA Architecture Specification (Core + Options),
- ECOA Bindings,
- ECOA Extensions.

[Architecture Specification Part 6] highlights optional mechanisms, or "ECOA Options", in the ECOA Architecture Specification.

A platform may claim conformance to the ECOA Architecture Specification without implementing these optional mechanisms. In other words, "ECOA-compliant Platform", without any further precision, means that the platform is compliant with the ECOA Core.

In addition to the ECOA Core, a platform may implement any number of ECOA Options.

A platform may claim a fully conformance to the ECOA Architecture Specification if it provides all optional mechanisms.

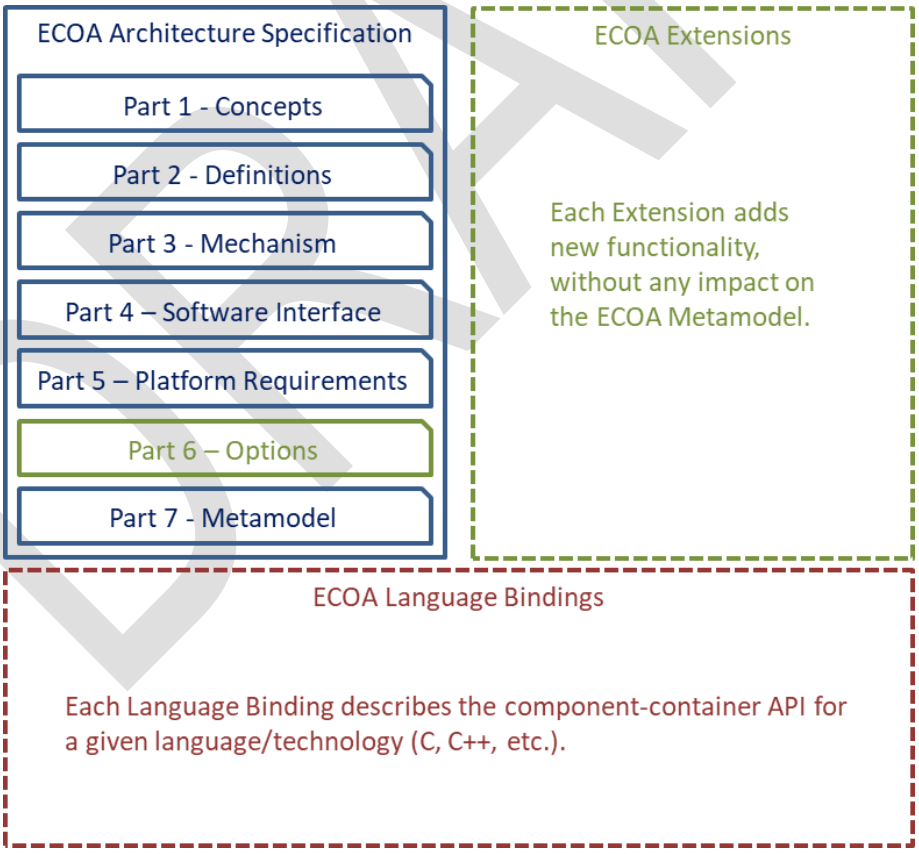

Figure 1 ECOA Standard - Core, Options, Extensions and Bindings

6.1 Naming convention for options

In a modular standard, it is important to have a well-defined:

- complete list of all options,
- unique name for each option.

The "official" name of each option follows the following rules:

- It is surrounded by square bracket [...].
- It is made of space-separated words.
- It starts with the word OPTION.
- Each word is made of uppercase letters and digits.

Example:

[OPTION A B C D]

In order to facilitate searches in documents, the exact official name of each option shall be used everywhere.

6.2 List of options

The following table gives the complete list of Options currently defined in the ECOA Standard.

Name of Option	Short Description	AS6 compatiblity
[OPTION SUPERVISION]	This option defines a kind of components ("supervisor components") that allows to control the lifecycle of components within an application.	No
[OPTION DYNAMIC TRIGGER]	This option defines a special kind of component, that is dedicated to sending an event at a specified date, on request of other components. When the option is activated, this component is fully generated by the platform.	Yes
[OPTION UINT64]	This option defines the datatype "uint64".	Yes
[OPTION INT64]	This option defines the datatype "int64".	Yes
[OPTION FAULT HANDLER]	This option allows to route errors towards the fault handler from the underlying ECOA platform at application level.	Yes
[OPTION EXTERNAL INTERFACE]	This option allows to generate an API usable by non ECOA code to send events to ECOA components.	Yes
[OPTION PINFO WRITE]	This option defines how some PINFO can be writable by components.	No
[OPTION WARM START CONTEXT]	This option defines how components can save and retrieve private context data, across the restarting of Applications.	Yes
[OPTION AUTO START EXTERNAL TASK]	This option allows to autostart the external task of an EXTERNAL component.	No

Name of Option	Short Description	AS6 compatiblity
[OPTION UTC TIME]	This option allows to have a UTC time reference common across the whole ECOA Platform.	Yes
[OPTION COMM PORTS]	This option allows an ECOA application to have external operations that can be connected to other applications through communication ports.	No

Table 1 List of Options currently defined in the ECOA Standard

7 High Level Requirements for an ECOA Platform

This section provides generic high level requirements that a platform shall satisfy to claim a conformance to the ECOA Architecture Specification and then be identified as an ECOA Platform.

Additional safety and security requirements (e.g. data integrity checks, authentication functions, determinism, level of assurance) may be specified as additional platform procurement requirements depending on the needs of each programme that uses ECOA. Such requirements are not defined by ECOA.

Id.	Requirement
	Implementation
HLR.1	The ECOA Platform **shall** conform to the Core perimeter of the Architecture Specification.
HLR.2	The ECOA Platform **may** support any number of the Options listed in section 6.2 of this document.
HLR.3	When the ECOA Platform supports an option, it **shall** implement it in its entirety.
HLR.4	The ECOA Platform **shall** provide an options support table, indicating which options are supported. An example of such a matrix is provided in Annex A of this document.
HLR.5	The ECOA Platform **shall** provide an extensions support table, indicating which Extensions are supported.
HLR.6	The ECOA Platform **shall** support at least one language binding, among the ECOA reference language bindings (C, C++, ...).
HLR.7	Any supported reference language binding **shall** be implemented in its entirety by the ECOA Platform, excluding the API that are related to options not supported by the Platform.
HLR.8	The ECOA Platform **shall** provide a bindings support table, indicating which Bindings are supported.
HLR.9	The ECOA Platform **shall** schedule ECOA Component Instances, using a scheduling policy which complies with the requirement to respect the priorities set by the System Integrator.
HLR.10	The ECOA Platform **shall** map Executables onto segregated memory spaces.

Id.	Requirement
HLR.11	The ECOA Platform **may** support the communication, managed by the Platform, between several Applications, through the use of "ports" in the deployment [OPTION COMM PORTS].
HLR.12	The ECOA Platform **should** support at least one protocol allowing the communication, managed by the Platform, between several Applications running on the same Computing Node.
HLR.13	The ECOA Platform **may** support graceful shutdown capability.
HLR.14	The ECOA Platform **may** perform container level checks of operations at runtime, based on specific (non-ECOA) requirements.
HLR.15	The ECOA Platform **may** support the configuration of which logging levels are enabled/disabled, before or during execution, globally or on an instance-by-instance basis, except for ERROR and FATAL that cannot be disabled.
HLR.16	The ECOA Platform **may** support the configuration of where logging information is directed to before or during execution (files, network, memory, etc.).
	Hardware
HLR.17	The characteristics and the limitations of the ECOA Platform **shall** be indicated by the Platform Supplier. For example: the maximum number of components that can be deployed into a single executable.
	Tooling
HLR.18	The ECOA Platform toolset **should** check the validity of ECOA XML files against the ECOA Metamodel.
HLR.19	The ECOA Platform toolset **should** check values declared in ECOA XML files relative to the type of attribute the value is associated with.
HLR.20	The ECOA Platform toolset **shall** be able to take any ECOA model as an input, provided that this model is complete, consistent, and conform to Architecture Specification Part 7, and even though unavailable optional features might be addressed. In that case, the ECOA Platform shall exit and shall report one of the missing optional features.
HLR.21	The ECOA Platform **shall** not require metadata to ensure conformity to ECOA mechanisms. Nevertheless, the ECOA Platform may offer specific features based on metadata.
HLR.22	When processing an ECOA model, the ECOA Platform **shall** set all undescribed optional attributes to their default value.
HLR.23	Software generation: Existing Component Implementations source code **should** not be replaced if it already exists.
HLR.24	Software Building: the ECOA Platform **should** respect compiling and building directives defined in Component Implementations and Deployment.

Table 2 High level requirements for an ECOA Compliant Platform

Annex A. Template for the option support table of an ECOA Platform

Name of Option	Support (YES or NO)
[OPTION SUPERVISOR COMPONENTS]	
[OPTION DYNAMIC TRIGGER MANAGER]	
[OPTION UINT64]	
[OPTION INT64]	
[OPTION FAULT HANDLER]	
[OPTION EXTERNAL INTERFACE]	
[OPTION PINFO WRITE]	
[OPTION WARM START CONTEXT]	
[OPTION AUTO START EXTERNAL TASK]	
[OPTION UTC TIME]	
[OPTION COMM PORTS]	