



# The EARS Knowledge Library

Tuesday, 11 December 2018

## Table of contents

- Introduction to EARS
- The content of the EARS Knowledge Library
- Next steps with the library



## What is EARS?

- EARS: Easy Approach to Requirements Syntax
- Created by Rolls-Royce and Intel
- To tackle the main issues detected in natural language requirements:
  - Ambiguity
  - Vagueness
  - Complexity
  - Omission
  - Duplication
  - Wordiness
  - Inappropriate implementation
  - Untestability

- More info at: [https://www.researchgate.net/publication/224079416\\_Easy\\_approach\\_to\\_requirements\\_syntax\\_EARS](https://www.researchgate.net/publication/224079416_Easy_approach_to_requirements_syntax_EARS)

# What is EARS?

- EARS: Easy Approach to Requirements Syntax
- The means proposed by EARS to tackle these issues is the identification of a series of syntax to write requirements
- These syntax are:
  - Generic requirement
  - Ubiquitous requirement
  - Event-driven requirement
  - Unwanted behaviours
  - State-driven requirements
  - Optional features
  - Complex

# What is EARS?

- › EARS: Easy Approach to Requirements Syntax
  - › Generic requirement:
    - › <Precondition> <Trigger> The <system name> shall <system response>
  - › Ubiquitous requirement
    - › The <system name> shall <system response>
  - › Event-driven requirement
    - › WHEN <trigger> <optional precondition> the <system name> shall <system response>
  - › Unwanted behaviours
    - › IF <unwanted condition or event>, THEN the <system name> shall <system response>
  - › State-driven requirements
    - › WHILE <system state>, the <system name> shall <system response>
  - › Optional features
    - › WHERE <feature is included>, the <system name> shall <system response>
  - › Complex: a combination of the other types of requirements

# What is a Knowledge Library

- › A combination of Knowledge items,
  - › of different nature,
  - › at different levels of abstraction
- › Representing a specific business domain or **area of knowledge**
- › With the aim of improving the way projects are managed, including:
  - › the promotion of the principle: **quality** *right the first time*,
  - › enabling semantic search portals to archive and retrieve assets,
  - › thus providing tools to **reuse** assets at different level,
  - › and reducing **time** to market,
  - › improving the way engineers generate (**author**) new assets,
  - › enhancing the way items are inspected and **verified**,
  - › Enabling real **interoperability** mechanisms and services,
  - › reducing **time** to elaborate documents, systems and projects

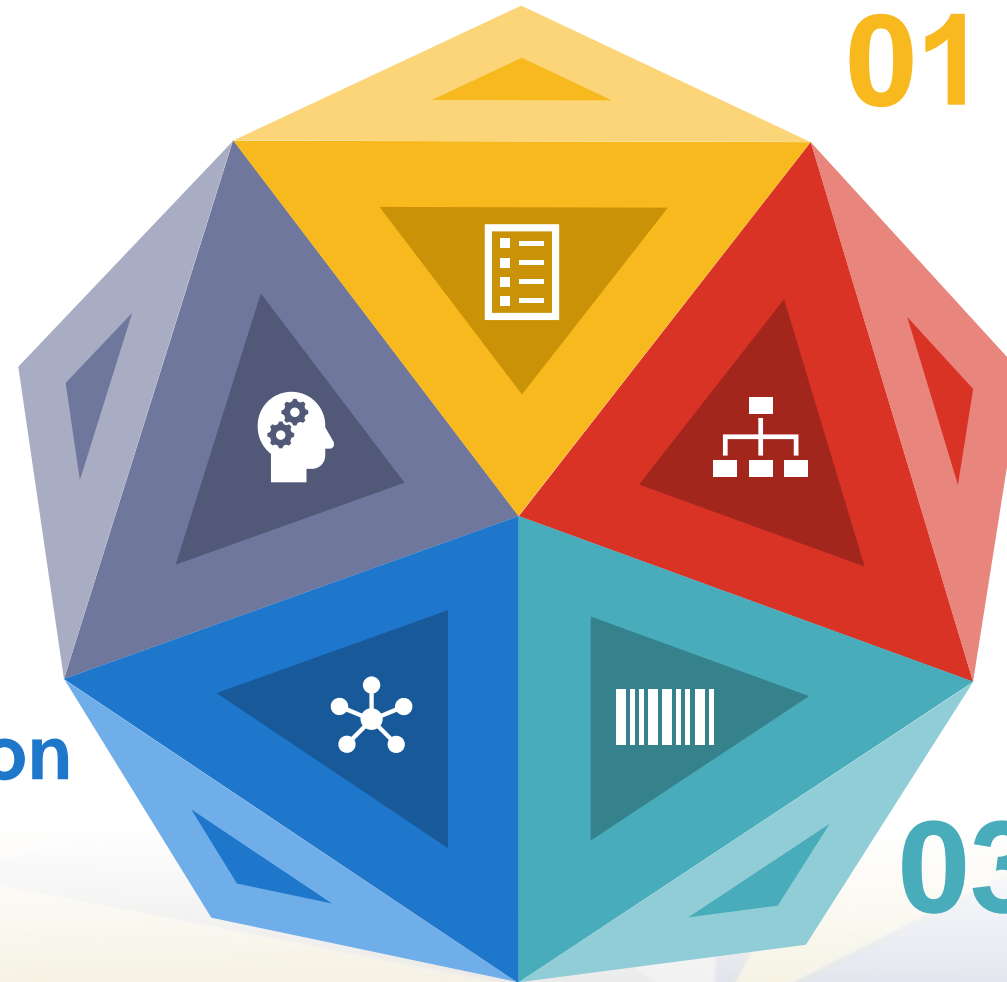


Knowledge Libraries



# What is a Knowledge Library

## Knowledge Libraries



01

### Vocabulary/Glossary

Controlled Organizational and Project Vocabulary for a common understanding among stakeholders

02

### SCM/Architectures

Capture the system architectures represented in views and models. Establish relationships among system and system elements, and among other system entities. Classifying information by meaning, nature...

03

### Patterns

Representing a set of agreed-upon templates (grammars) to create and maintain consistent textual artifacts

05 Reasoning

A combination of rules, and actions to infer information from valuable assets and to control the behavioural part of the knowledge library

04 Formalization

Representation of assets semantic through SRL – System Representation Language

# Example of Knowledge Library

## Vocabulary

Aircraft

Ground segment

System

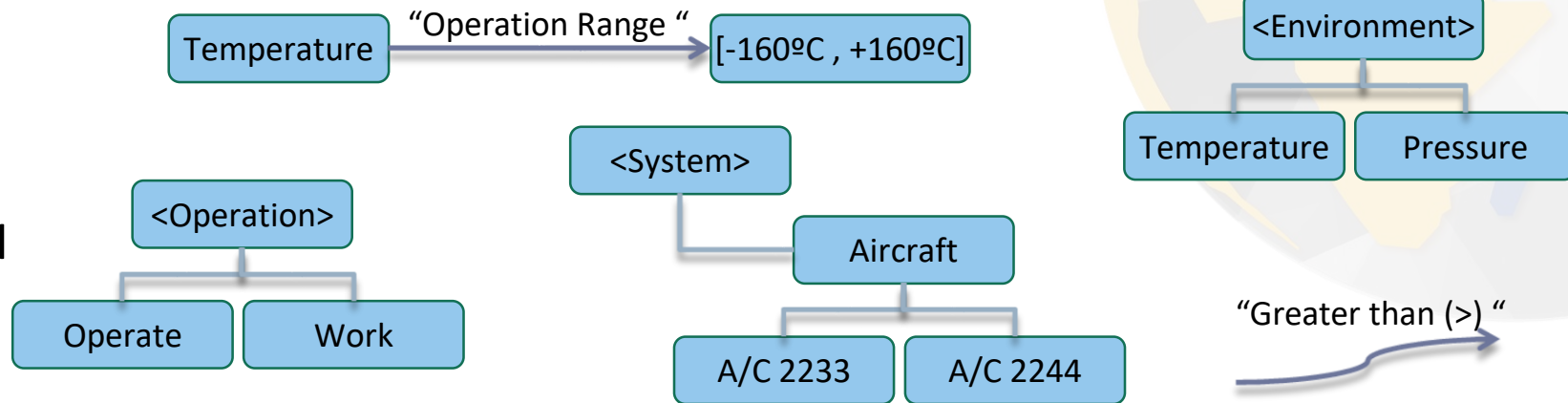
Operate

Temperature

Environment

Pressure

## Architectures - Conceptual model



## Patterns

<System>

Shall

<Operation>

At

Minimum

<Environment>

Of

NUMBER

MEASUREMENT UNIT

## Formalization

The aircraft shall be able to operate at a minimum temperature of -170° C

Temperature

"Greater than (>)"

-170

°C

## Reasoning

If NUMBER

Lower than (<) -160° °C

Or NUMBER

Greater than (>) +160° °C

→





# The EARS Knowledge Library

- › The content of the Knowledge Library is the following:
  - › The common purpose vocabulary extracted from the EARS examples and documentation
  - › A taxonomy of types of requirements following the types of EARS
  - › A set of patterns for each type following the examples provided by the EARS guides
  - › A set of small reusable patterns (as building blocks) used in the main 6 patterns

# The EARS Knowledge Library

- The content of the Knowledge Library is the following:

Pattern - Syntax

Pattern fields:

Identifier: 997 Name: System name

Original example: the system

Current example: the system

Syntax:

ARTICLE + <SYSTEM> NOUN +

the ? system

The System

ARTICLE GENDER: N/A NUMBER: INVARIANT

<SYSTEM> NOUN GENDER: N/A NUMBER: INVARIANT

Show advanced options Show advanced options

Finish Cancel

Pattern - Syntax

Pattern fields:

Identifier: 1,001 Name: Entity

Original example: at least the request light ac system

Current example: at least the request light ac system

Syntax:

[Quantifier] + [Adjective / Verb in P.P.] + <ENTITY> +

or

[Entity: <Entity> <Parameter>]

or

[Entity: <parameter> of the <System Element>]

at least ? the ? request ? light ? ac system

At least The Request Light Ac system

ADVERB GENDER: N/A NUMBER: INVARIANT

ARTICLE GENDER: N/A NUMBER: INVARIANT

PERSON: GERUND VERB GENDER: N/A NUMBER: INVARIANT

ADJECTIVE GENDER: N/A NUMBER: INVARIANT

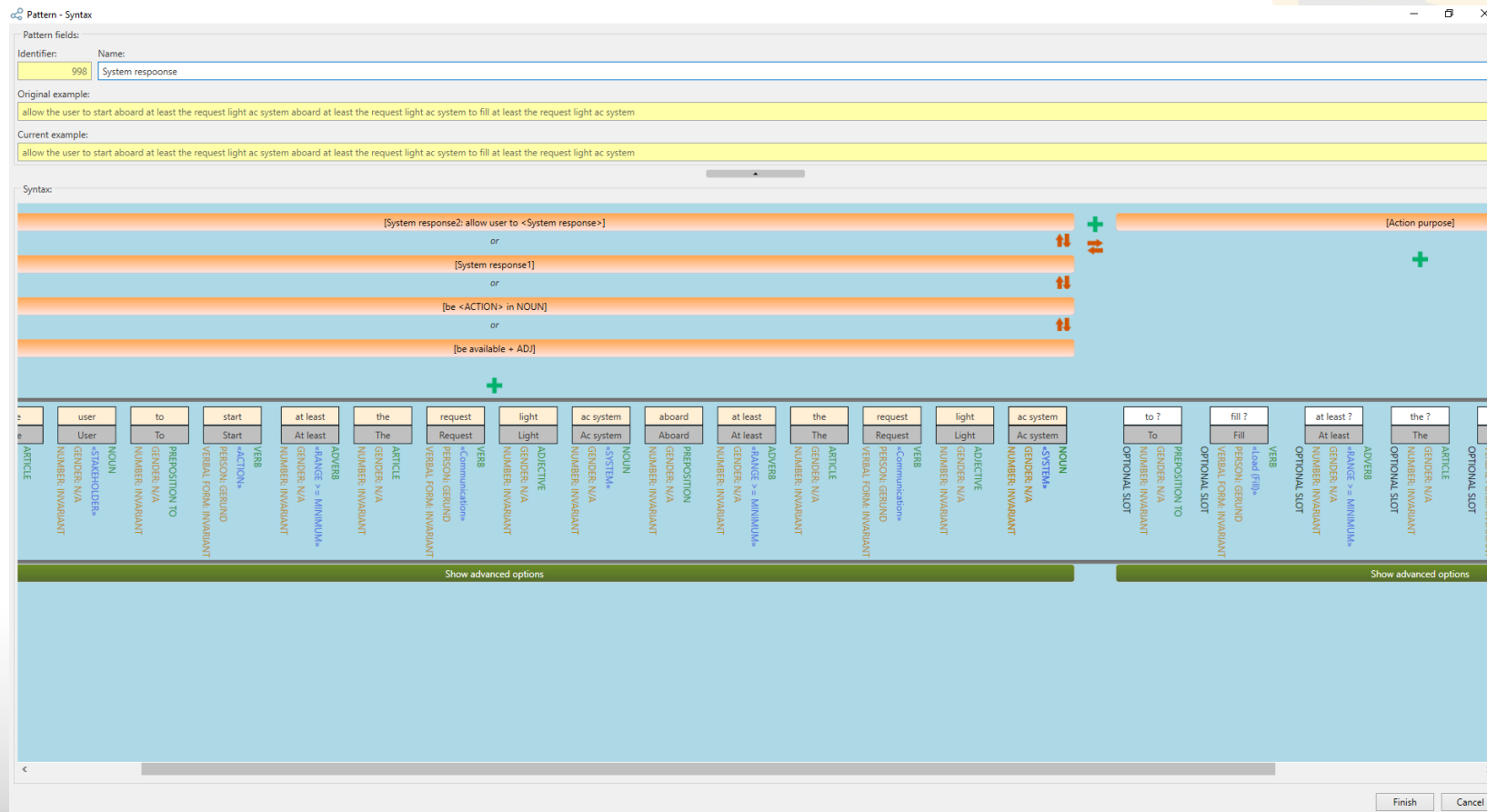
<SYSTEM> NOUN GENDER: N/A NUMBER: INVARIANT

Show advanced options Show advanced options

Finish Cancel

# The EARS Knowledge Library

- The content of the Knowledge Library is the following:

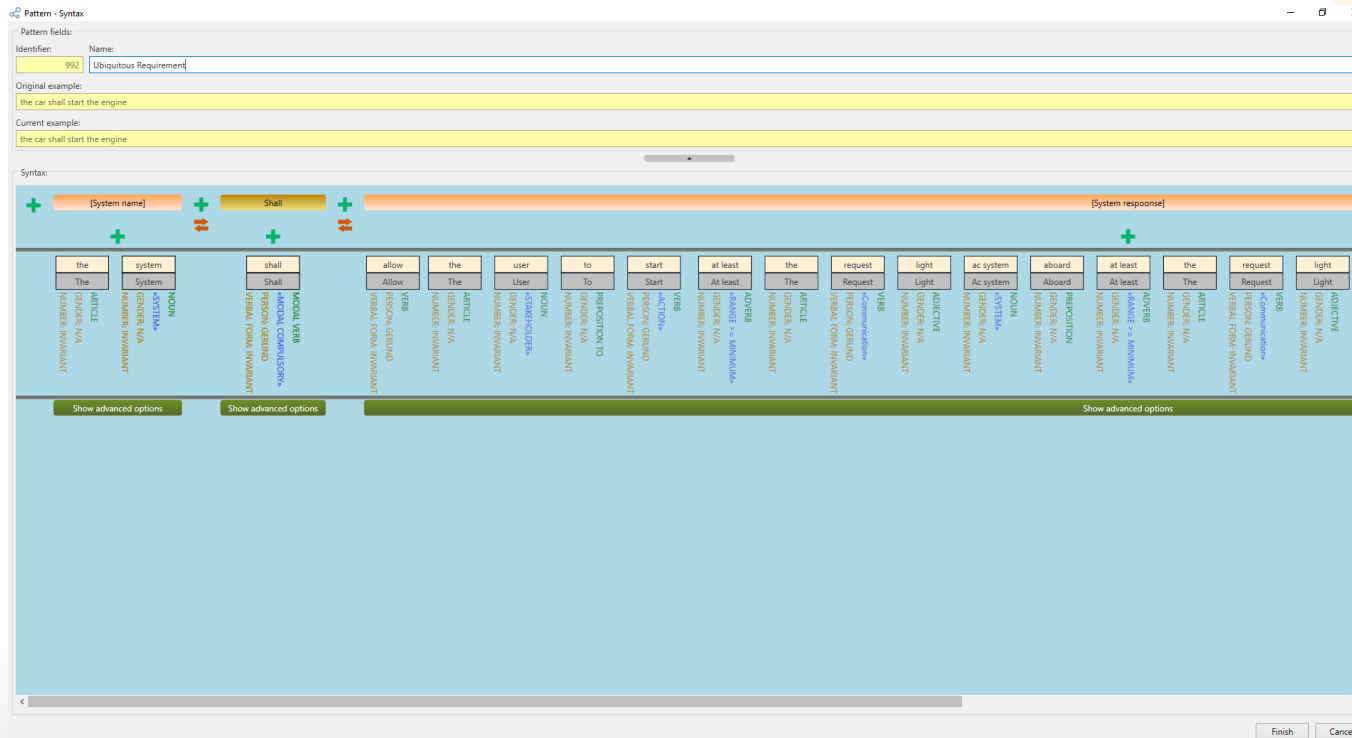


The screenshot shows a software interface titled "Pattern - Syntax". It contains the following fields and content:

- Identifier:** 998
- Name:** System response
- Original example:** allow the user to start aboard at least the request light ac system aboard at least the request light ac system to fill at least the request light ac system
- Current example:** allow the user to start aboard at least the request light ac system aboard at least the request light ac system to fill at least the request light ac system
- Syntax:**
  - [System response2: allow user to <System response>]
  - or
  - [System response]
  - or
  - [be <ACTION> in NOUN]
  - or
  - [be available + ADJ]
- Word Bank:** A list of words with their grammatical categories below them:
  - user (User, NOUN)
  - to (To, PREPOSITION TO)
  - start (Start, VERB)
  - at least (At least, ADVERB)
  - the (The, ARTICLE)
  - request (Request, VERB)
  - light (Light, ADJECTIVE)
  - ac system (Ac system, NOUN)
  - aboard (Aboard, PREPOSITION)
  - at least (At least, ADVERB)
  - the (The, ARTICLE)
  - request (Request, VERB)
  - light (Light, ADJECTIVE)
  - ac system (Ac system, NOUN)
  - to ? (To, PREPOSITION TO)
  - fill ? (Fill, VERB)
  - at least ? (At least, ADVERB)
  - the ? (The, ARTICLE)
- Buttons:** "Show advanced options" (two instances), "Finish", and "Cancel".

# The EARS Knowledge Library

- The content of the Knowledge Library is the following:



Pattern - Syntax

Identifier: Name: 992 Ubiquitous Requirement

Original example: the car shall start the engine

Current example: the car shall start the engine

Syntax:

[System name] Shall [System response]

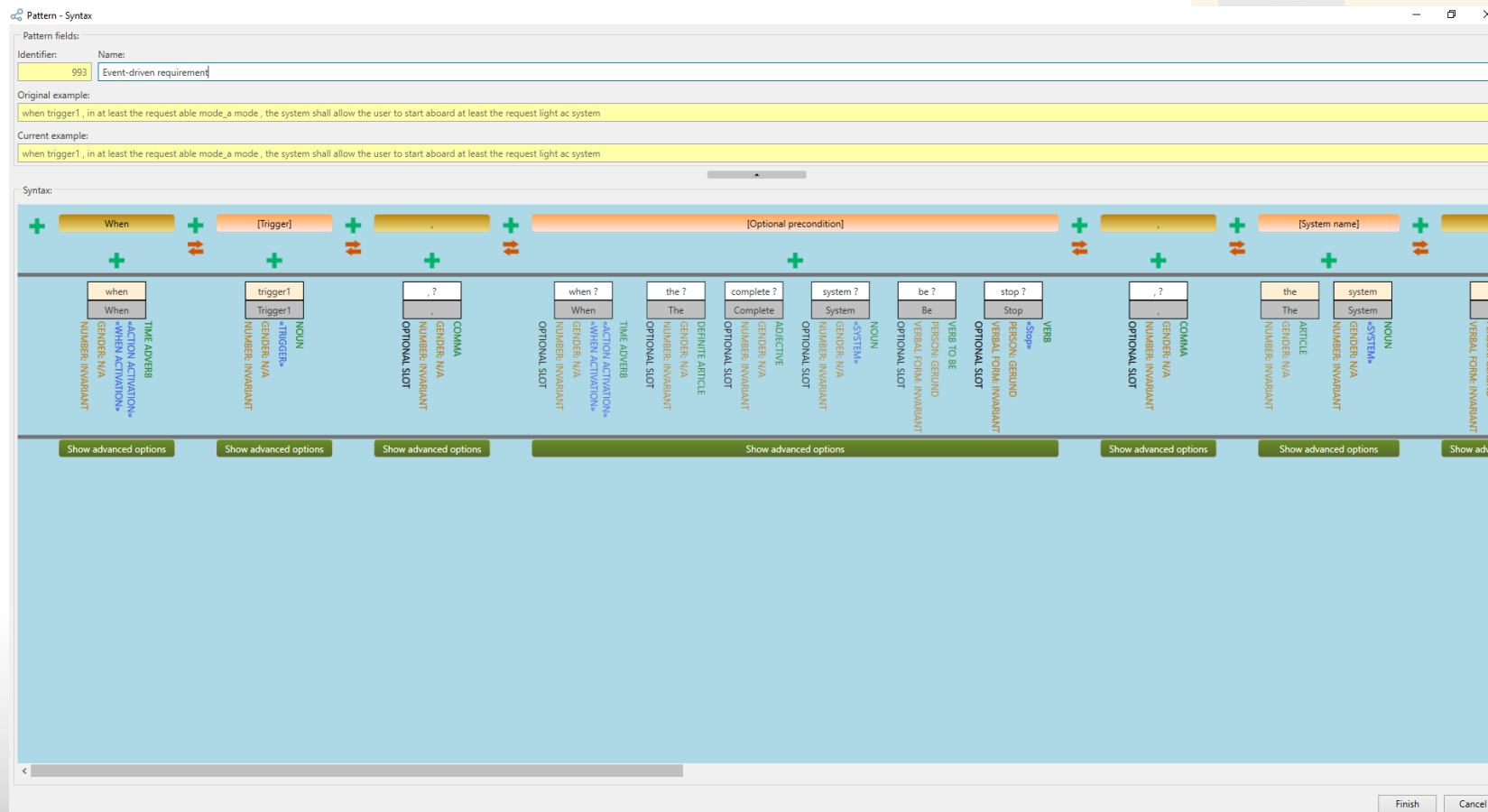
the	system	shall	allow	the	user	to	start	at least	the	request	light	ac system	aboard	at least	the	request	light
The	System	Shall	Allow	The	User	To	Start	At least	The	Request	Light	Ac system	Aboard	At least	The	Request	Light
ARTICLE	SYSTEM	MODAL_VRB	VERB	ARTICLE	PERSON_NVA	PREPOSITION_TO	ACTON_V	ADVERB	ARTICLE	PERSON_GERUND	ADJECTIVE	SYSTEM	PREPOSITION	ADVERB	ARTICLE	PERSON_GERUND	ADJECTIVE
NUMBERS_INVARIANT	NUMBERS_INVARIANT	PERSON_GERUND	PERSON_GERUND	NUMBERS_INVARIANT	NUMBERS_INVARIANT	NUMBERS_INVARIANT	NUMBERS_INVARIANT	NUMBERS_INVARIANT	NUMBERS_INVARIANT	PERSON_GERUND	NUMBERS_INVARIANT	NUMBERS_INVARIANT	NUMBERS_INVARIANT	NUMBERS_INVARIANT	NUMBERS_INVARIANT	PERSON_GERUND	NUMBERS_INVARIANT

Show advanced options

Finish Cancel

# The EARS Knowledge Library

- The content of the Knowledge Library is the following:



The screenshot shows a software interface titled "Pattern - Syntax". It displays a requirement pattern and its syntactic breakdown.

**Pattern fields:**

- Identifier: 993
- Name: Event-driven requirement

**Original example:**  
when trigger1, in at least the request able mode\_a mode, the system shall allow the user to start aboard at least the request light ac system

**Current example:**  
when trigger1, in at least the request able mode\_a mode, the system shall allow the user to start aboard at least the request light ac system

**Syntax:**

The syntax is visualized as a sequence of elements with their grammatical roles and options:

- When** (TIME ADVERB, optional)
- [Trigger]** (TRIGGER, optional)
- ,** (COMMA, optional)
- [Optional precondition]** (optional)
- ,** (COMMA, optional)
- [System name]** (SYSTEM, optional)
- shall** (MODAL VERB, mandatory)
- allow** (VERB, mandatory)
- the user** (PERSON, mandatory)
- to start aboard** (VERB, mandatory)
- at least the request light ac system** (PERSON, mandatory)

Each element has a "Show advanced options" button below it.

# The EARS Knowledge Library

- The content of the Knowledge Library is the following:

Pattern - Syntax

Pattern fields:

Identifier: 994 Name: Unwanted Behavior Requirement

Original example: if in at least the request able mode\_a mode , Then the car shall activate the engine

Current example: if in at least the request able mode\_a mode , Then the car shall activate the engine

Syntax:

«IF ACTIVATION» CONECTOR REQUIREMENT/CONDITION + [Unwanted condition or event] + , + Then + [System name] + «MODAL COMPULSORY» MODAL VERB +

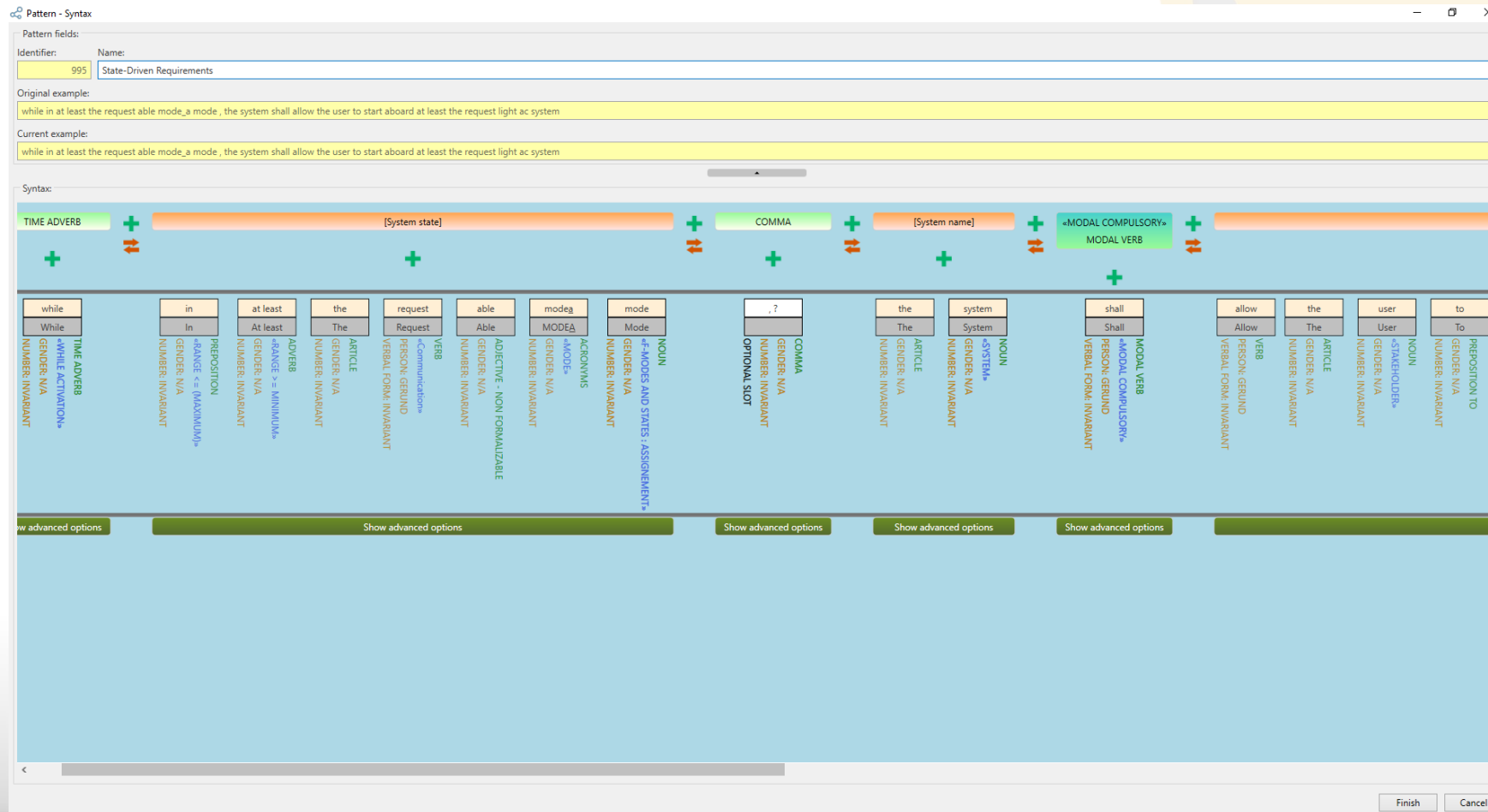
if	when	the	complete	system	be	stop	, ?	then	the	system	shall	allow	the
If	When	The	Complete	System	Be	Stop	, ?	Then	The	System	Shall	Allow	The
«IF ACTIVATION» GENDER: N/A NUMBER: INVARIANT	«WHEN ACTIVATION» GENDER: N/A NUMBER: INVARIANT	DEFINITE ARTICLE GENDER: N/A NUMBER: INVARIANT	ADJECTIVE GENDER: N/A NUMBER: INVARIANT	«SYSTEM» GENDER: N/A NUMBER: INVARIANT	VERB TO BE PERSON: GERUND VERBAL FORM: INVARIANT	«STOP» PERSON: GERUND VERBAL FORM: INVARIANT	OPTIONAL SLOT GENDER: N/A NUMBER: INVARIANT	TIME ADVERB GENDER: N/A NUMBER: INVARIANT	ARTICLE GENDER: N/A NUMBER: INVARIANT	«SYSTEM» GENDER: N/A NUMBER: INVARIANT	«MODAL COMPULSORY» PERSON: GERUND VERBAL FORM: INVARIANT	VERB PERSON: GERUND VERBAL FORM: INVARIANT	ARTICLE GENDER: N/A NUMBER: INVARIANT

Show advanced options

Finish Cancel

# The EARS Knowledge Library

- The content of the Knowledge Library is the following:



The screenshot shows a software interface titled "Pattern - Syntax". It displays a requirement pattern and its syntactic breakdown. The pattern is: "while in at least the request able mode\_a mode , the system shall allow the user to start aboard at least the request light ac system".

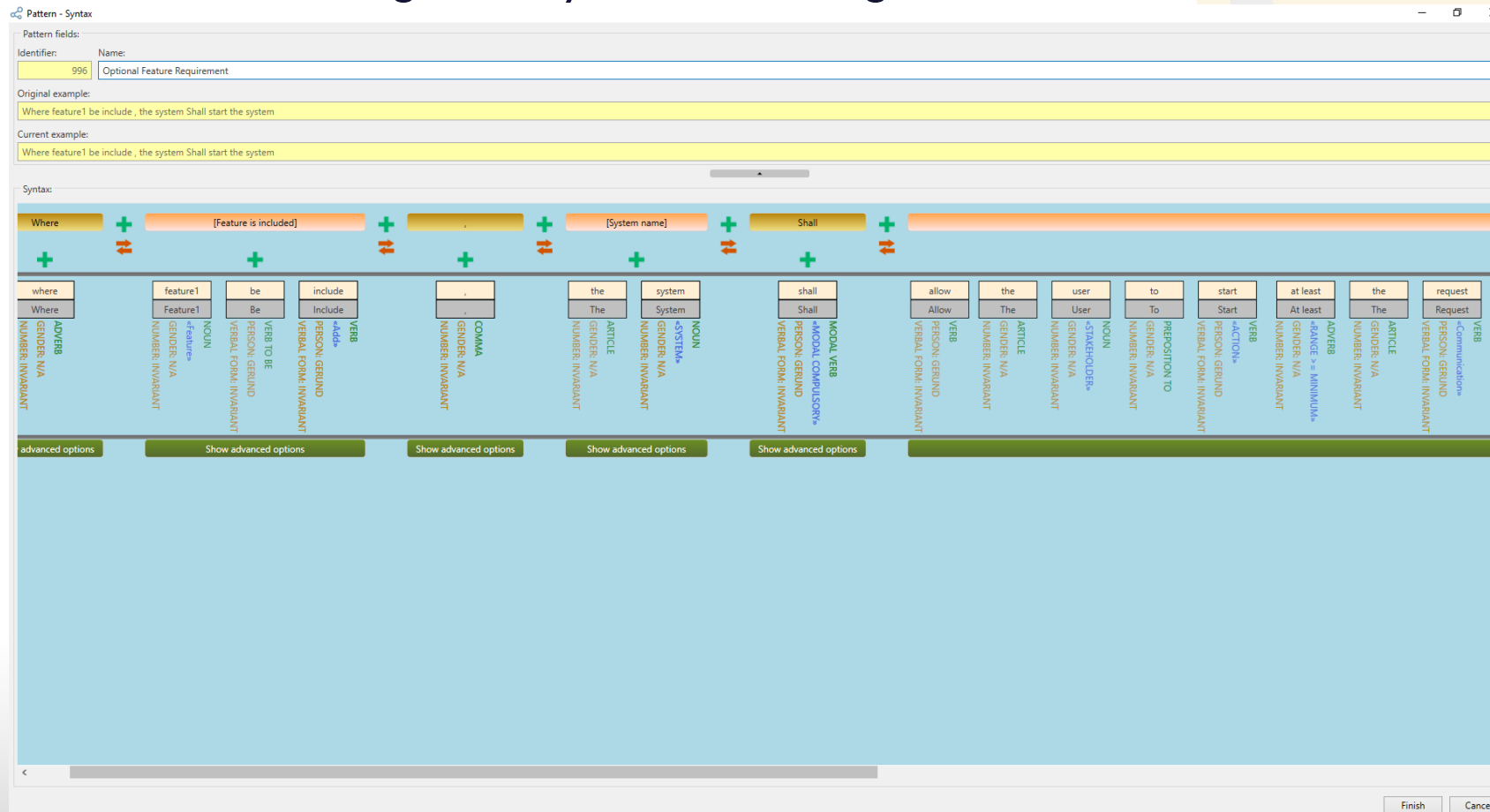
The syntactic breakdown is as follows:

Pattern Field	Example	Category
[System state]	while	TIME ADVERB
[System name]	in	PREPOSITION
[MODAL COMPULSORY]	at least	ADVERB
[MODAL VERB]	the	ARTICLE
[MODAL VERB]	request	VERB
[MODAL VERB]	able	ADJECTIVE - NON-FORMALIZABLE
[MODAL VERB]	mode_a	ACRONYM
[MODAL VERB]	mode	NOUN
[MODAL VERB]	, ?	COMMA
[MODAL VERB]	the	ARTICLE
[MODAL VERB]	system	NOUN
[MODAL VERB]	shall	MODAL VERB
[MODAL VERB]	allow	VERB
[MODAL VERB]	the	ARTICLE
[MODAL VERB]	user	NOUN
[MODAL VERB]	to	PREPOSITION TO

Each syntactic element is shown with its grammatical category and a list of possible words. Below the breakdown, there are several "Show advanced options" buttons.

# The EARS Knowledge Library

- The content of the Knowledge Library is the following:



The screenshot shows a software interface for defining and analyzing requirement patterns. The window title is "Pattern - Syntax".

**Pattern fields:**

- Identifier: 996
- Name: Optional Feature Requirement

**Original example:**  
Where feature1 be include , the system Shall start the system

**Current example:**  
Where feature1 be include , the system Shall start the system

**Syntax:**

The syntax is visualized as a sequence of tokens with their grammatical roles and parts of speech:

Token	Role	Part of Speech
Where	+	ADVVERB
[Feature is included]	+	VERB
,	+	COMMA
[System name]	+	NOUN
Shall	+	MODAL VERB
allow	+	VERB
the	+	ARTICLE
user	+	NOUN
to	+	PREPOSITION TO
start	+	VERB
at least	+	ADVERB
the	+	ARTICLE
request	+	VERB

Below the syntax visualization, there are buttons for "Show advanced options" for each token.





the

**REUSE**

company

