Artifact design methodology (simplified)

BEST PRACTICES | DRAFT | V.0.2

Daniel Alvarez 15 (104 days ago)

Introduction

This page describes a ***simplified**** version of a methodology for designing artifacts that aim to solve specific technical problems. This abstract view of the methodology was inspired by the Design Science Research (DSR) initially positioned by Hevner in [1-2], and in particular by its information- and software-oriented version described by Wieringa in [3].

*** Disclaimer**

The word "*simplified*" implies here that the intention is not to replace or reinvent the Design Science Research (DSR) methodology already established in academic research. Instead, this interpretation adopts the fundamental components of artifact design from DSR that can serve as a guideline to frame multiple collaborative projects within a unified specific structure. Please follow this simplified methodology for designing artifacts within the COVESA alliance. If you require a research-oriented design, please refer to the corresponding literature on DSR or similar scientific frameworks.

Design Science Research in a nutshell

DSR is used in fields like information systems and engineering to solve real-world problems by creating innovative solutions. Think of it as a way of applying a guideline (i.e., a structured process) to design something new or to answer open questions about an already existing design. In DSR, the innovative solution is generally referred to as an ***artifact***. In a nutshell, DSR is like an organized way of inventing cool stuff to solve practical problems!

In order for the design to be useful and valid, one has to define (ideally upfront) the fundamental components: ***Problem***, ***Goal(s)***, ***Requirements***, and the ***Artifact type***. Doing so will automatically determine the scope for the necessary work.

Artifact types

The term "*artifact*" may sound too abstract and ambiguous at first. However, it is only a general way to cover multiple possibilities under the same methodology. In practice, this word is replaced with a specific artifact type. Nevertheless, using its generic form is helpful at the initial stages of a project, where the solution's particular shape has yet to be defined. As the list of specific artifacts could be extensive, one can directly link any piece of technology as a subcategory of eight parent types, as described in [4]:

Artifact type	Explanation	Examples
System design	Description of a structure, a process, or some interaction.	• Software architecture • Database schema • Business process diagram
Method	Activities that are *performed by people* in order to accomplish a particular task or solve a problem.	• A/B testing method • Design thinking process • Kanban agile method
Language or Notation	Symbols, rules, or conventions used to represent information or instructions as a formal abstraction of reality.	• A data model of a particular domain (e.g., Vehicle Signal Specification) • A controlled vocabulary (e.g., a taxonomy of vehicle types) • An object-oriented programming

Algorithm	Description of a *machine-executable* step-by-step procedure or set of rules designed to solve a problem or perform a specific task.	• Sorting algorithm • Encryption algorithm • Data mining algorithm
Guideline	Suggestion regarding behaviour in a particular situation.	• "If a vehicle data model has to be used in specific applications, use the parser to output the desired standard format and adapt the result to match the required schema."
Requirements	Statements about a system. They constraint the operational conditions.	• "In a connected vehicle architecture, none safety-critical data points can be publicly shared." • "In a vehicle data architecture, the standards X must be supported."
Pattern	Reusable elements of a design.	• Asynchronous message queue • Service-oriented architecture
Metric	Any model that can be used to evaluate aspects of system design.	• Latency • Business case

Fundamental components of an artifact design

Problem

Specific issue or challenge that requires a solution or improvement.

Goal

Ultimate objective(s) that a solution aims to achieve, typically formed by the stakeholders' desires. In the context of COVESA, the goal of an artifact is inherit from the general COVESA goals defined as an alliance. In other words, each artifact will represent (minor or major) steps towards an ultimate goal.

Requirements

Criteria and specifications that the artifact must meet to address the identified problem and achieve the set goals. Typically presented as functional and non-functional.

- **Functional**: Specific functions, tasks, or actions that the designed artifact must perform to proof utility.
- **Non-functional**: Specific qualities or characteristics that the artifact must have. They represent the constraints under which the design must operate.

Artifact

Represents the tangible outcome of a design that aims to solve the problem and fulfils the specified requirements and goals.

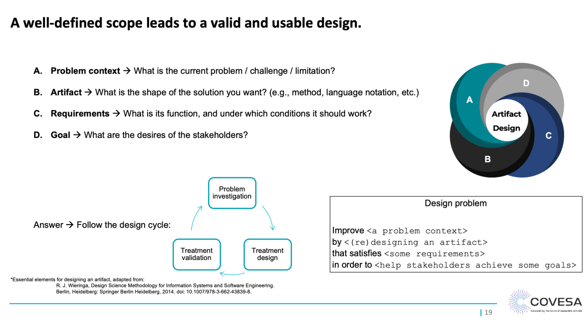
Well-defined design problems

The fundamental components of an artifact design form together a clear and well-scoped summary. Following the template suggested in [3], each workflow at COVESA must target only artefacts with a complete set of design components as follows:

Formulating the design problem

Improve (or solve) a ***<problem>***
by designing an ***<artifact>***
that satisfies ***<requirements>***
in order to achieve ***<goal(s)>***

Bear in mind that specific details of the fundamental components might be unclear at the starting phase of a project. Nevertheless, as the work advances, they must be updated according to the development and aligned with the official releases of an artifact.



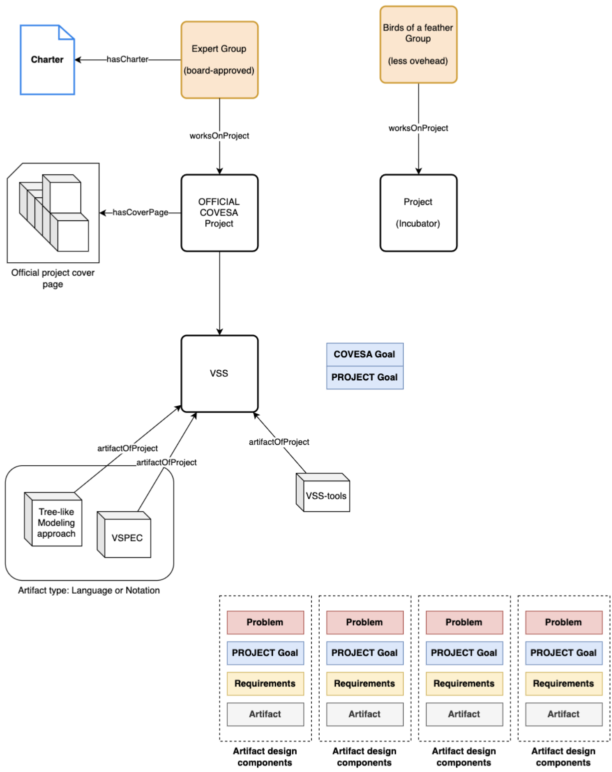
---

*Idea as of 21.11.2023*Idea as of 21.11.2023Guideline

DRAFT

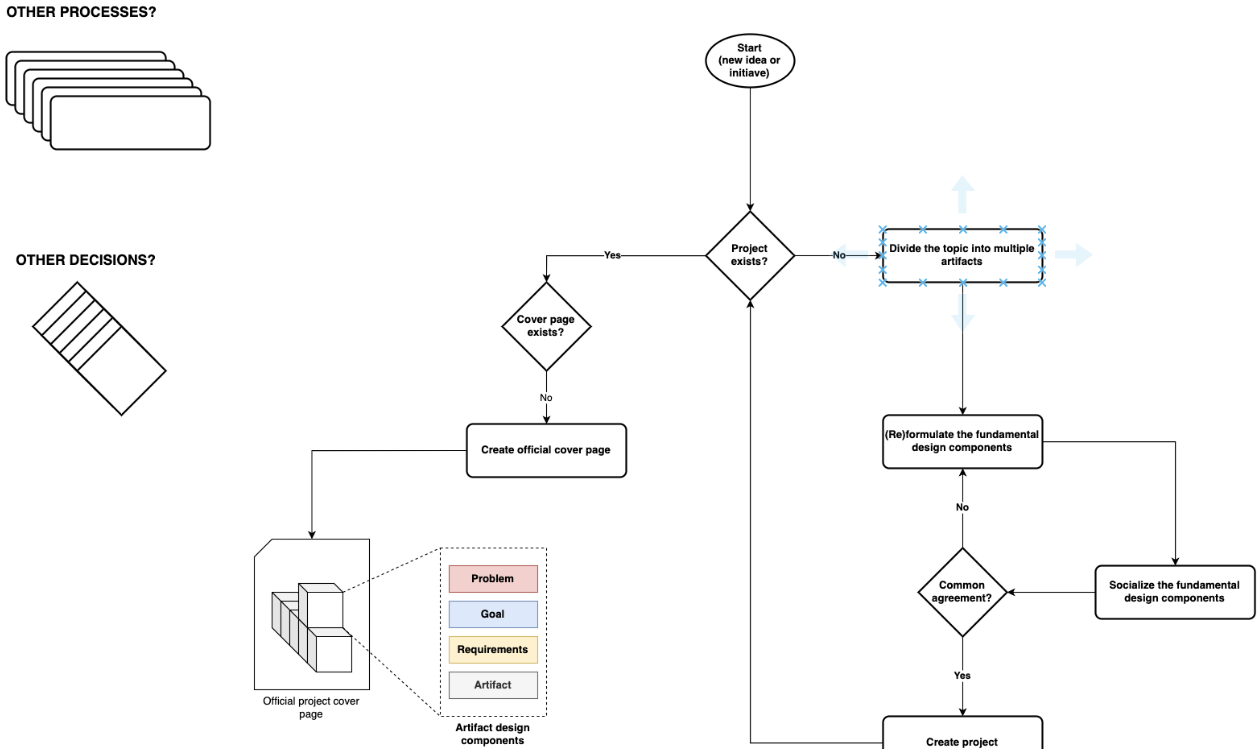
Overall COVESA structure

*Idea as of 21.11.2023



Decision flow

*Idea as of 21.11.2023



Resources

- COVESA project cover page template PLANNED
- COVESA individual artifact template PLANNED

Example: VSS fundamental design components

The existing work related to the Vehicle Signal Specification (VSS) was taken as an example on how one can frame existing COVESA projects in terms of this simplified design methodology. Please, follow this link to access the example.

Click here to see the example: VSS fundamental design components

---

References

[1]
A. Hevner, S. March, J. Park, and S. Ram, ***"Design science in information systems research,"*** *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004.

[2]
S. Gregor and A. Hevner, ***"Positioning and presenting design science research for maximum impact,"*** *MIS quarterly*, pp. 337–355, 2013.

[3]
R. J. Wieringa, ***Design Science Methodology for Information Systems and Software Engineering***. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014. doi: 10.1007/978-3-662-43839-8.

[4]
P. Offermann, S. Blom, M. Schönherr, and U. Bub, ***"Artifact Types in Information Systems Design Science – A Literature Review,"*** in *Global Perspectives on Design Science Research*, vol. 6105, R. Winter, J. L. Zhao, and S. Aier, Eds., in Lecture Notes in Computer Science, vol. 6105. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 77–92. doi: 10.1007/978-3-642-13335-0_6.