

How a Small Team Can Develop Complex Systems

Challenges and Opportunities

Behzad Samadi

October 23, 2024

Designing a Complex System

Model Based Systems Engineering (MBSE)

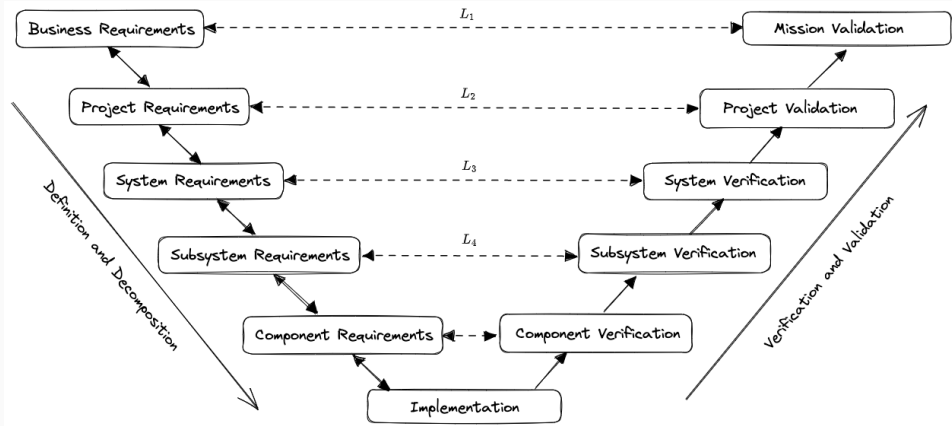


Figure 1: The V Diagram

Some references to learn more about MBSE:

- **NASA Systems Engineering Handbook**
- **Applying a Model-Based Approach to Support Requirements Analysis on the Thirty-Meter Telescope**
- **Open Model Based Engineering Environment**

Modularity

- **A modular system is made of smaller and simpler components (modules) that can be reused, replaced or reconfigured.**
- **The way components interact (middleware) keeps the system together while the components are replaced or reconfigured.**
- **Modularity helps to maintain the system's functionality over time.**
- **Modularity enable teams of engineers collaborate (work in parallel) on designing and implementing a complex system.**

Some examples of modular design are:

- **Lego bricks:** You can use different bricks to create different shapes and models.
- **IKEA furniture:** You can assemble different parts to create different furniture pieces or configurations.
- **Desktop computers:** You can customize different components to create different computer specifications or performance.

Different Types of Modularity

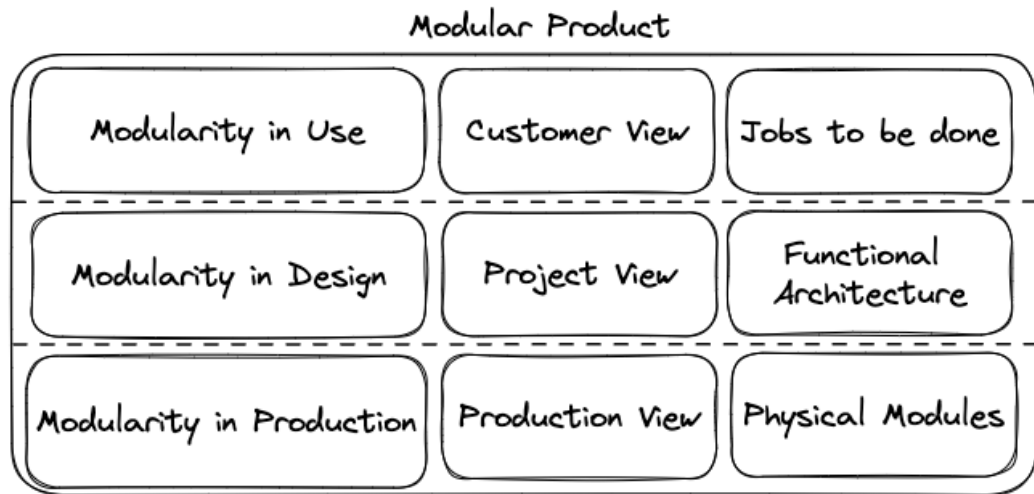


Figure 2: Types of Modularity

Software-Defined Products

- **Hardware-defined:**
 - Features of the product are static and cannot be changed.
 - To get new features, you need to buy a new product.
- **Software defined:**
 - New features will be added to the product through over-the-air updates.
 - The product can be customized for each customer using software settings.
 - New business models and types of collaboration become possible.

Open research (OECD definition): It is the practice of making “the primary outputs of publicly funded research results – publications and the research data – publicly accessible in a digital format with no or minimal restriction.”

- Publicly available
- Reuseable
- Transparent

Open Research Challenges

- **May require more time and resources**
- **May raise ethical or legal issues**
- **May face technical or cultural barriers**

Learning More About Open Research

One of the most comprehensive resources for learning about open research is The Turing Way handbook to reproducible, ethical and collaborative data science.

- **Reproducible research**
- **Project design**
- **Communicaiton**
- **Ethical research**

- **Open data**
- **Open source software**
- **Open hardware**
- **Open access publications**
- **Open notebooks**

According to The Turing Way handbook, open scholarship is a concept that extends open research further. It relates to making other aspects of scientific research open to the public, for example:

- **Open educational resources: OpenStax and edX**
- **Equity, diversity and inclusion**
- **Citizen science**

FAIR Principles for Open Data

Findable

Accessible

Interoperable

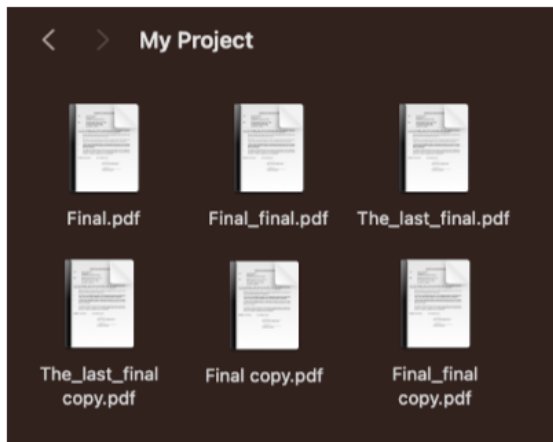
Reusable

Open-source software is an important resource for innovation:

- **It allows users to customize, modify and improve their software according to their needs**
- **It reduces costs, time and resources for software development**
- **It prevents vendor lock-in and promotes interoperability**
- **It fosters innovation, collaboration and learning among developers**

Tracking Project History

Stop doing this even if you are working alone!



Version Control

- **Tracking how a set of files changes over time**
- **Going back back to previous versions of any file or the whole project easily**
- **Collaborating with others on a project without interfering with their work on the same files**
- **Making branches to try out different changes (such as new features or bug fixes) separately from the main branch**
- **Saving a specific version of your work as a tag or a release for later use**

- **Git repositories:** public or private
- **GitHub Pages:** make a website with your files
- **GitHub Actions:** automate your tasks
- **GitHub Projects:** create plans and monitor your progress
- **GitHub Discussions:** a forum for collaborative communication
- **GitHub Wiki:** a knowledge base for your project

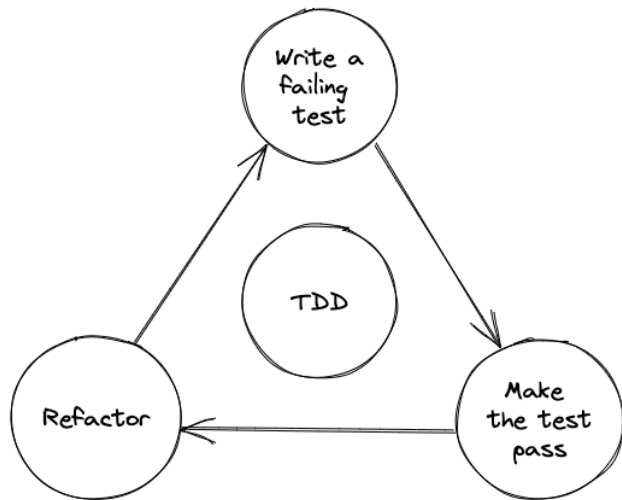
Software Development Challenges

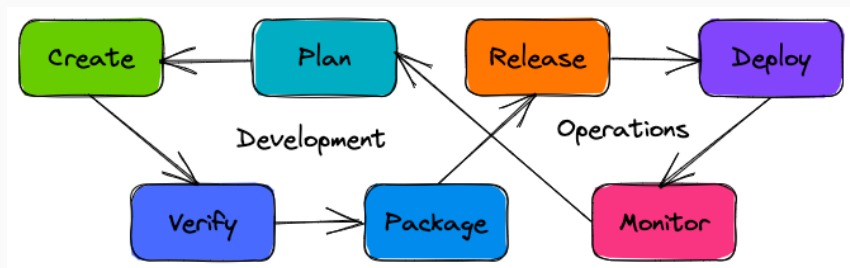
- **Where can I find the latest version of this feature?**
- **It used to work.**
- **It works on my computer.**
- **What are the requirements of the project?**
- **What are the short term objectives and key results of the project?**

Software binders:

- **Source code**
- **Dependencies and configuration files:** `pyproject.toml`
- **Continuous integration job description: build and test pipelines**
- **Continuous deployment job description: deployment pipelines**

Test-Driven Development





GitHub is a Search Engine for Open-Source Software

- **Search function:** `in:name`, `in:description`, `in:readme`
- **Topics:** labels that help you discover projects
- **Explore page:** curated collections of projects
- **External resources:**
 - **Awesome lists:** curated lists of awesome resources for various topics
 - **CodeTriage:** helps you find open source projects that need your help
 - **Libraries.io:** monitors over 3 million open source libraries across 37 different package managers

Take Action: Solve a Real-World Problem

1. **Choose a Topic:** Pick a challenge to address.
2. **Research:** Study existing solutions and requirements.
3. **Create a Repository:** Set up a project on GitHub with code and documentation.
4. **Implement:** Build a solution using best practices like modularity and TDD.
5. **Add a Simulation (if possible):** Validate your solution.
6. **Share:** Make your project open-source for collaboration.

Start Small, Think Big, Make a Difference!