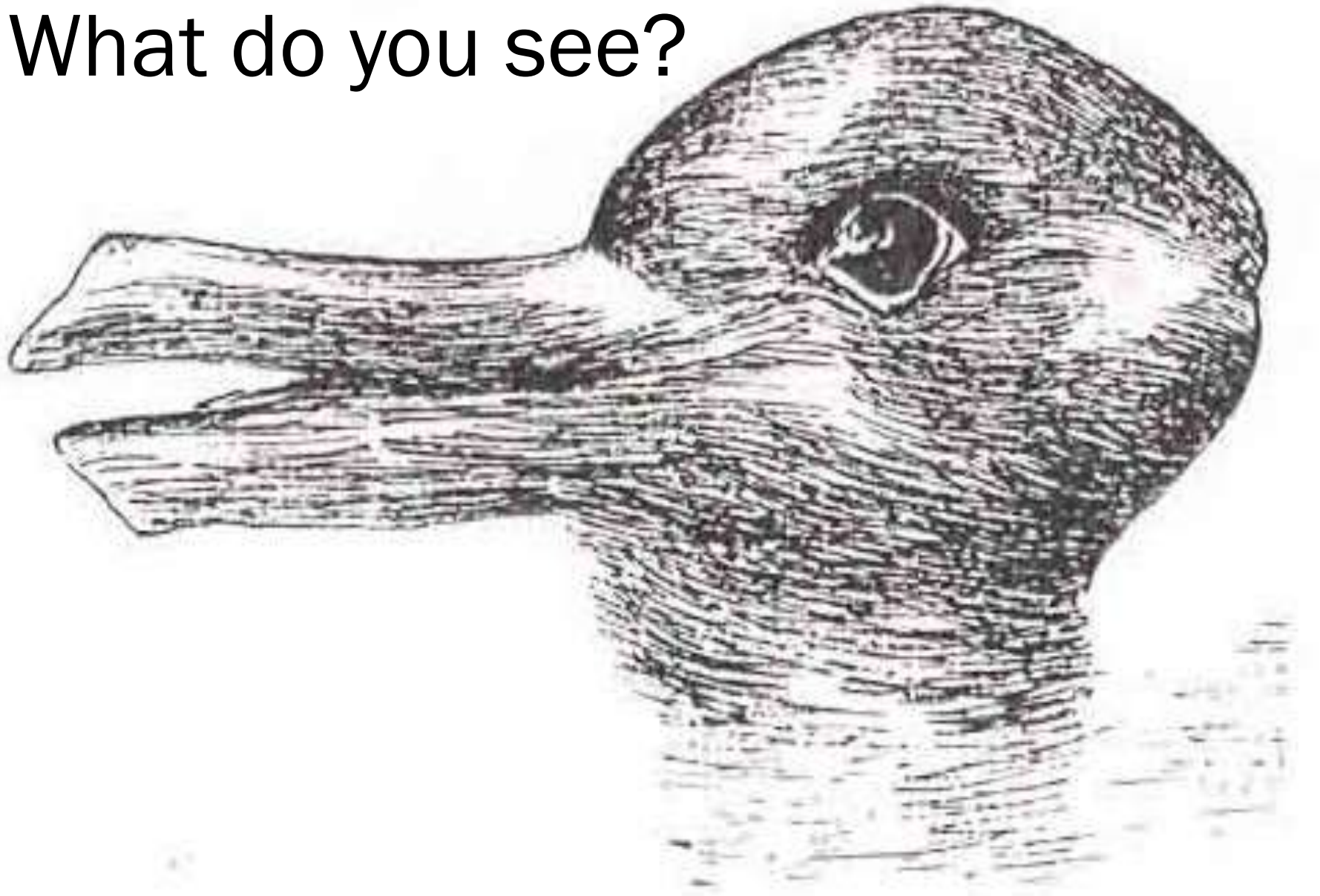


**Paradigm shift
from project-based to product based
Requirements Management**

What do you see?





Where do the requirements belong?

Product?

Project?

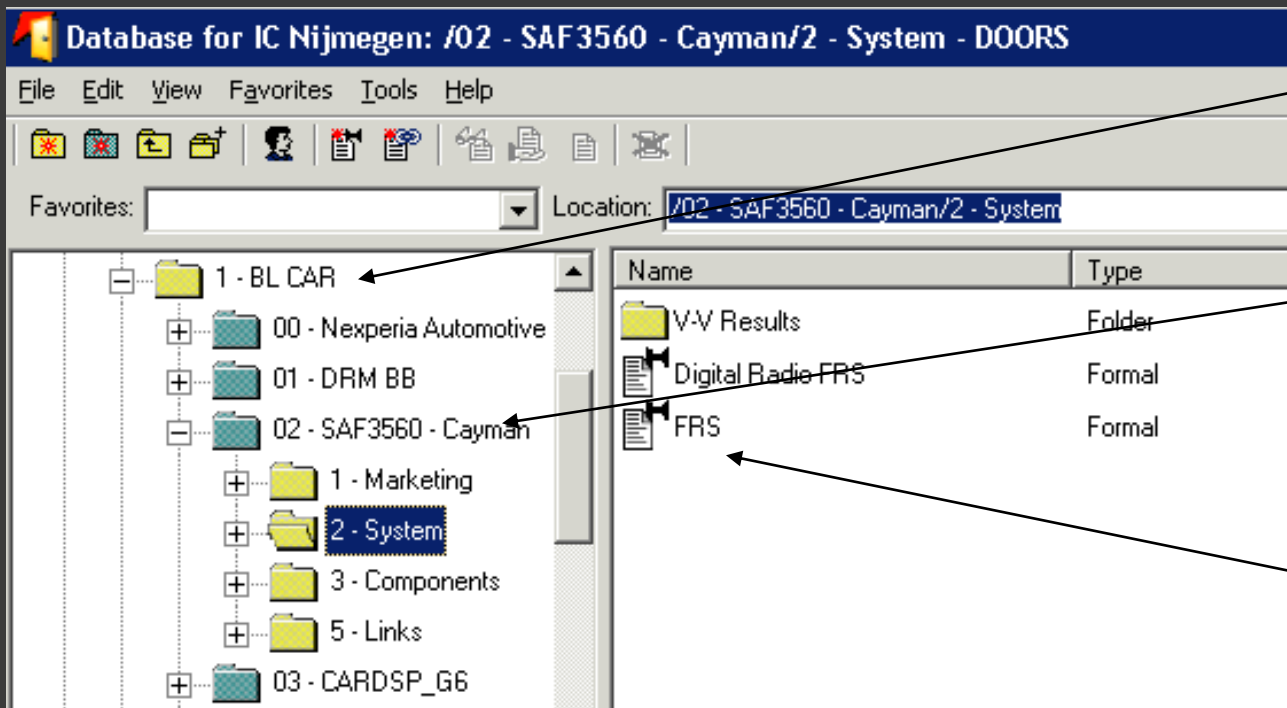
Requirements management

- ① Project-based
- ① Product-based

Project-based requirements

- ⦿ Development project == DOORS project.
- ⦿ Product and its components are only visible through a development project.
- ⦿ The traceability information is also set up within the scope of a development project.

Representation in DOORS



Development projects in a Business Line

project delivers an IC

Product specifications of an IC on the system level.

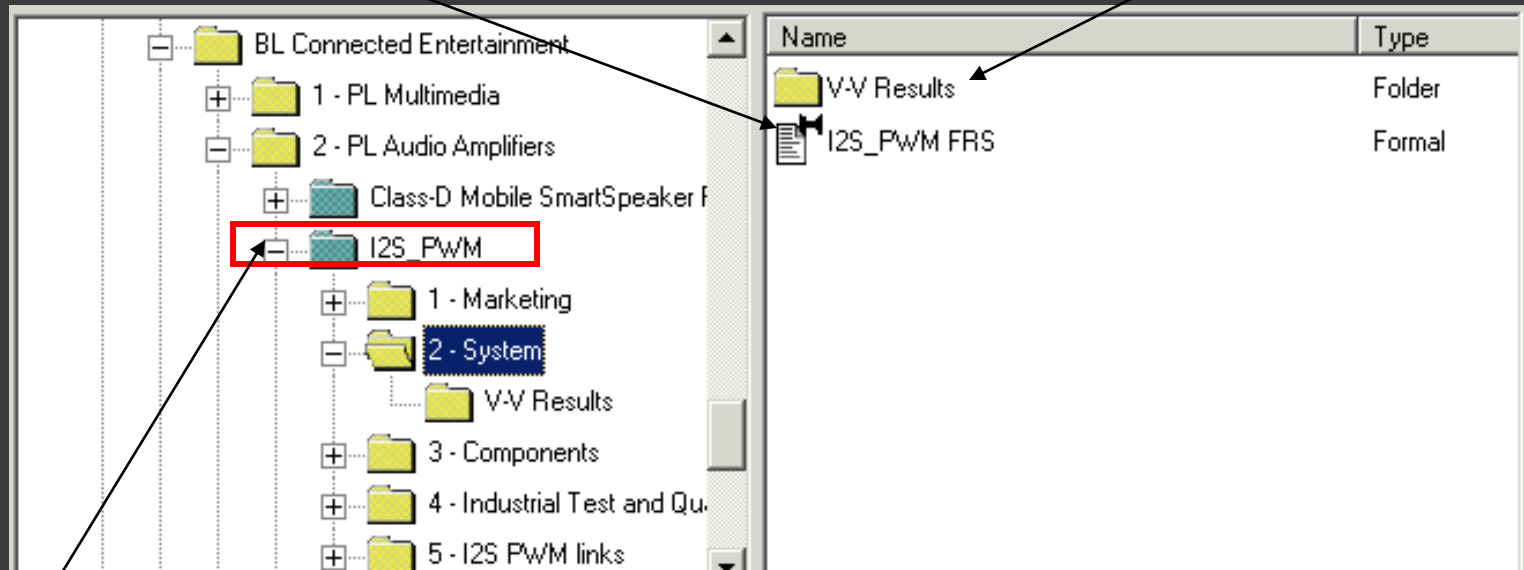
DOORS implementation

- ⦿ Requirements for a product gathered in the context of a project.
- ⦿ Each product or product component has a dedicated folder.
- ⦿ Links are stored in link modules within project scope):
- ⦿ Every product/product component requirements module describes the product from the black box perspective.

DOORS implementation

(1) Requirements module

(2/3) Test Specification & Test Results



Development Project

Link modules

Project-based requirements

⦿ Advantages

- Simple to understand,
- Easy to implement in DOORS,
- Good overview and structure for the project.

⦿ Disadvantages

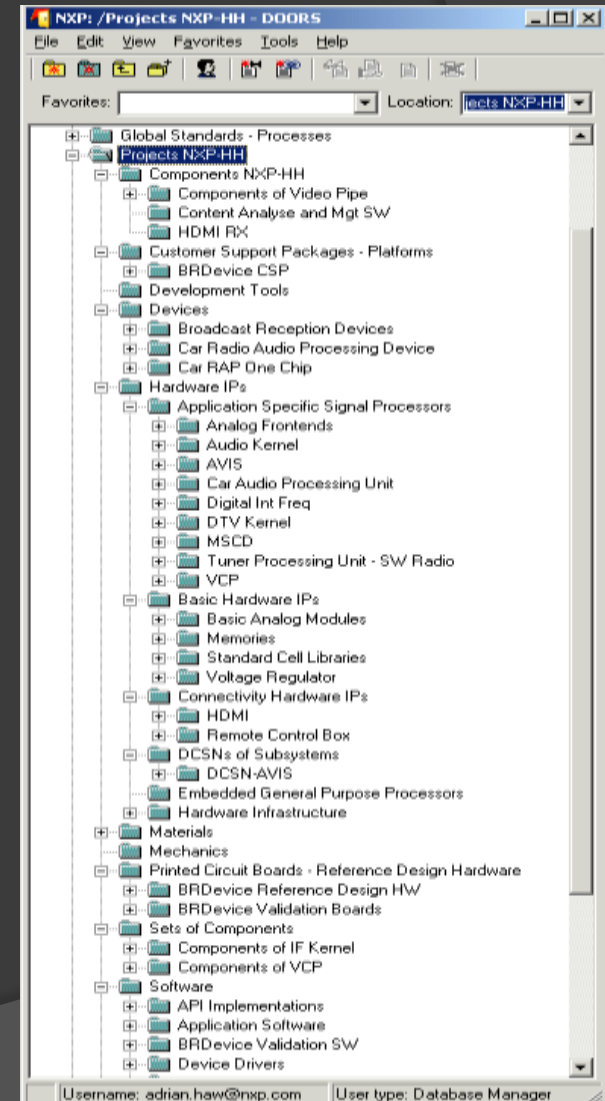
- Product visibility is poor,
- Reuse is difficult if not impossible.

Product-based requirements

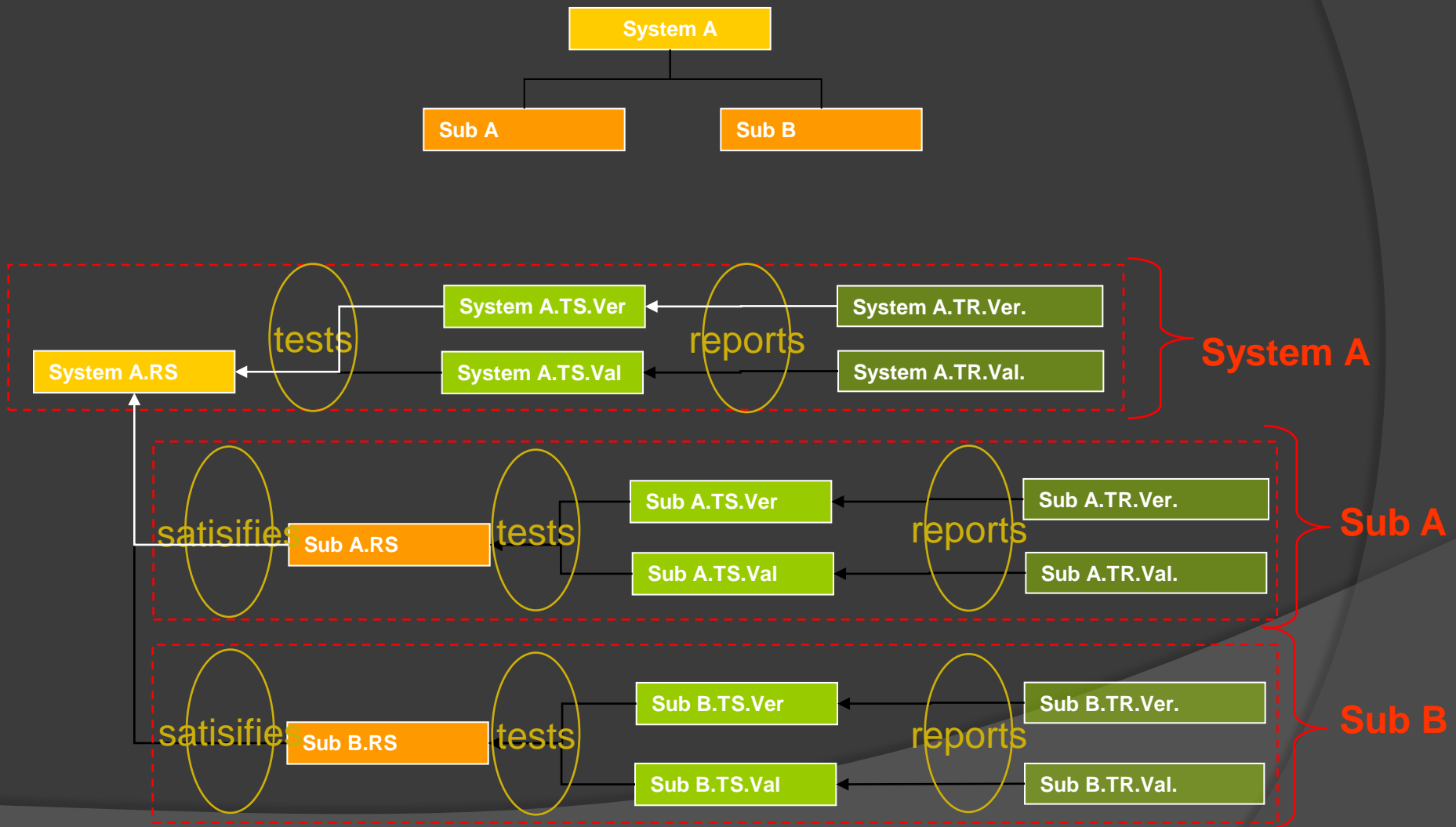
- The requirements are grouped per product/ product component instead of per project

Product-based requirements

- Product/ Product component == DOORS project.
- Development projects are not visible in DOORS.
- The traceability information between a product and its components is clearly defined within a product.



DOORS implementation



Product-based requirements

- ⦿ Advantages

- Makes the reuse of requirements easy,
- Faster requirements development.

- ⦿ Disadvantages

- Requires certain maturity in requirements writing,
- Can be perceived as difficult.



SOLUTION 1

SOLUTION 3

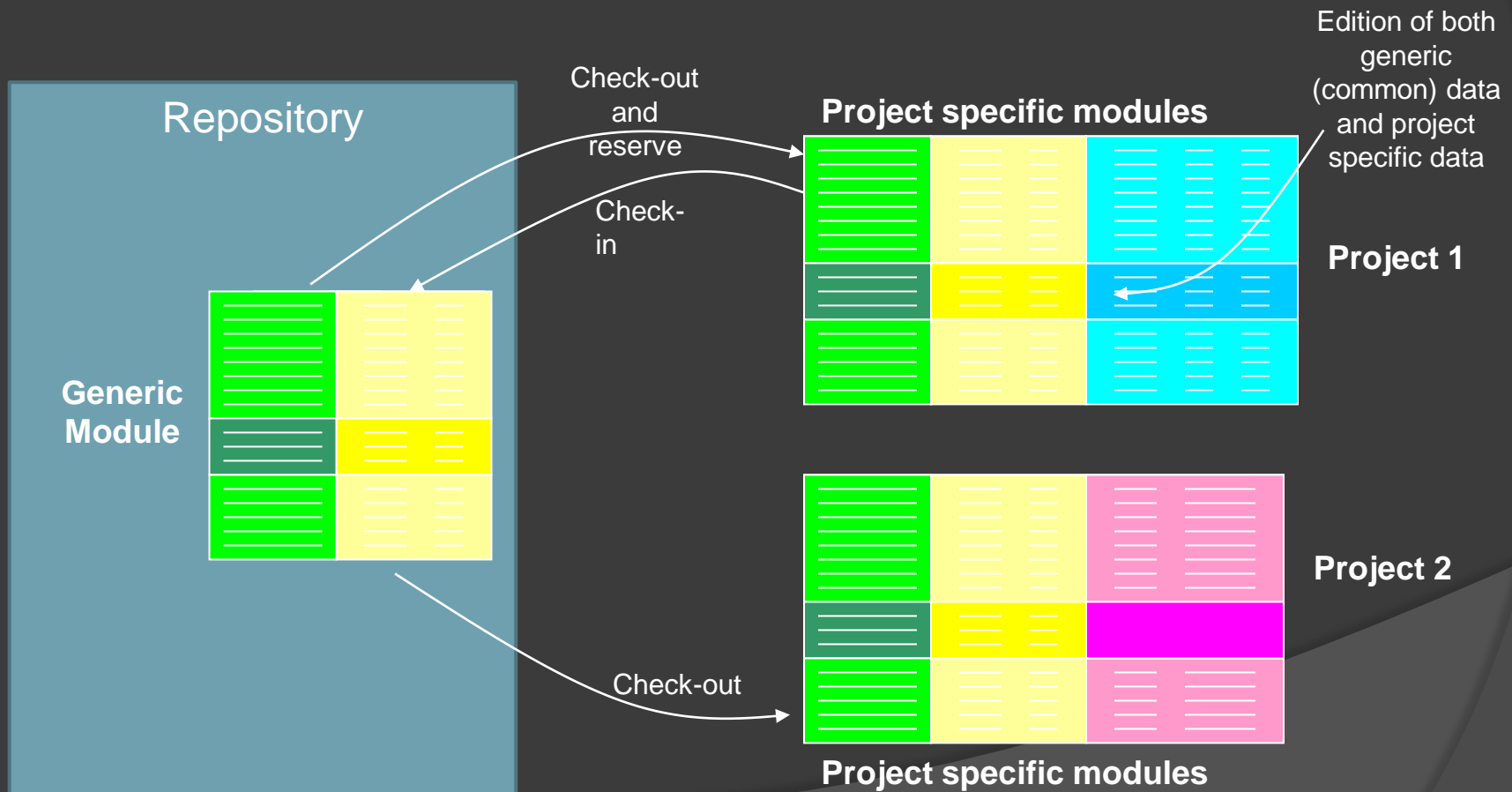
SOLUTION 2

Repositories

Combination of product-based and project-based requirements.

- ① Develop requirements product-based,
- ② Use requirements project-based.

Repository - principles



Repository-based requirements

⦿ Advantages

- Combines the simplicity of project-based requirements with product-based basis,
- Facilitates reuse,
- Allows faster requirements development,
- Off-the-shelf solutions available

⦿ Disadvantages

- Clear responsibilities over content of the repositories is necessary,
- Requires heavy DXL scripting if an off-the-shelf solution not used.

Conclusions

- ① There are different ways to work with requirements.
- ① Organizations have to evaluate their way of working and choose the requirements approach which fits their needs.
- ① Transition from the project-based to repository-based or product-based way of working is possible.

Thank you
Questions?