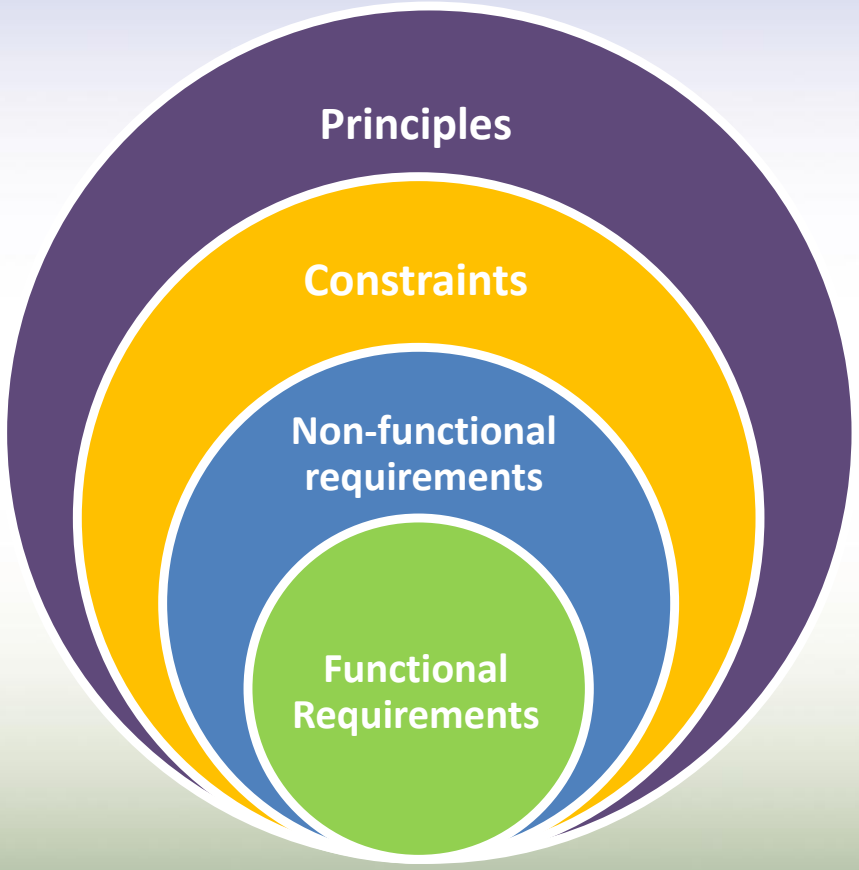
Agile architecture sketches «4C» approach

Sergey Denisov
Data Architect & Modeler
Salym Petroleum Development
11.03.2014

AGENDA

- Context
- Problem
- Methodology/approach
- Implementation
- What is next?

DESIGNING SOFTWARE



PROBLEMS

• SA HLD documents in current format is not useful.
It takes much time and power, is coming elder before finished.

•There is no single “materialized” view on solution as a whole.

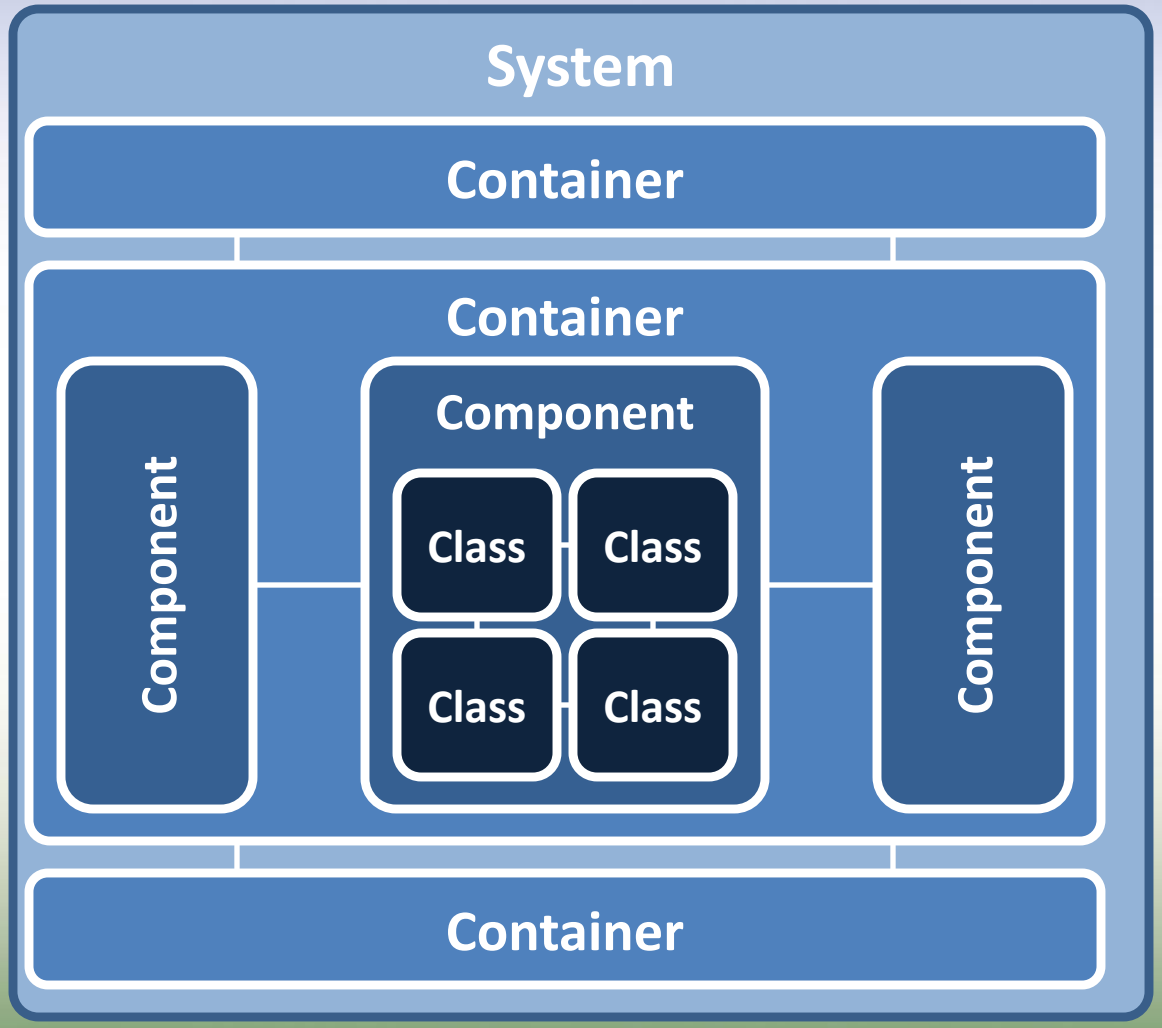
•We have troubles in communication of business requirements and architecture decisions: what and how should we build IT-solutions.

•New staff on-boarding to project is complicated and chaotic.

•Painful handover to support process and scattered support documentation.

•Trash in meta.

«4C» DIAGRAM SKETCHING



Context diagram

Container diagram

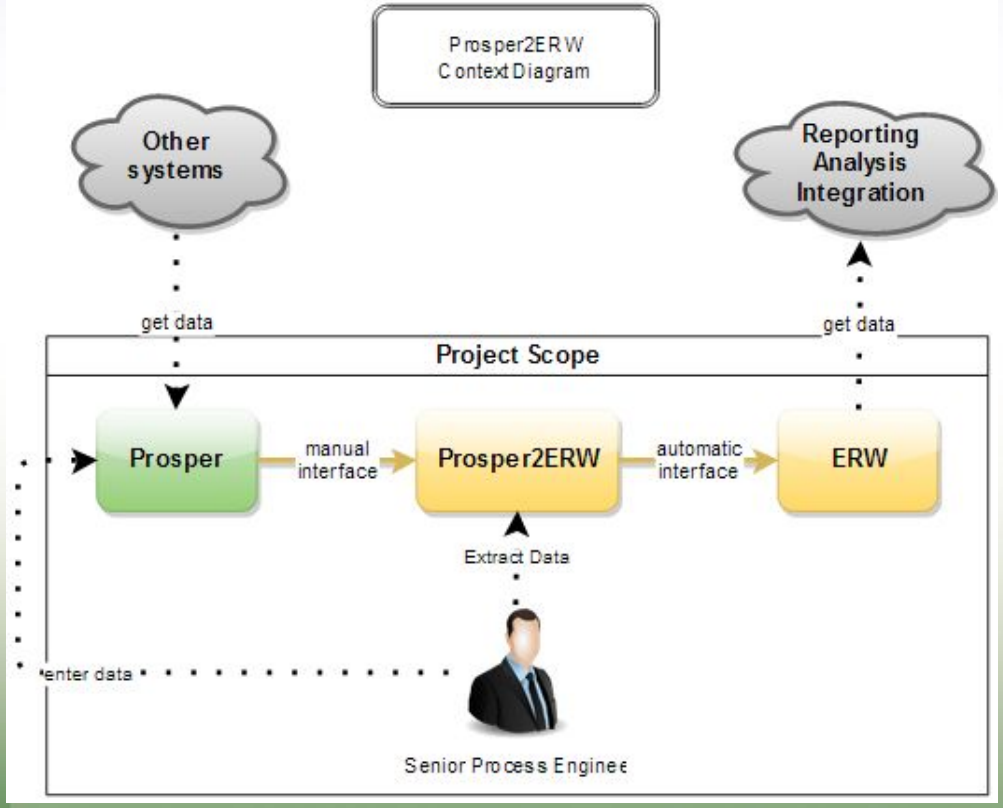
Component diagram

Class diagram

1C: CONTEXT DIAGRAM

An big picture of the system landscape:

- What is the software system that we are building?
- Who is using it?
- How does it fit in the existing IT environment?



Content:

IT System

Users, actors, roles, personas, etc

Purpose →

Motivation

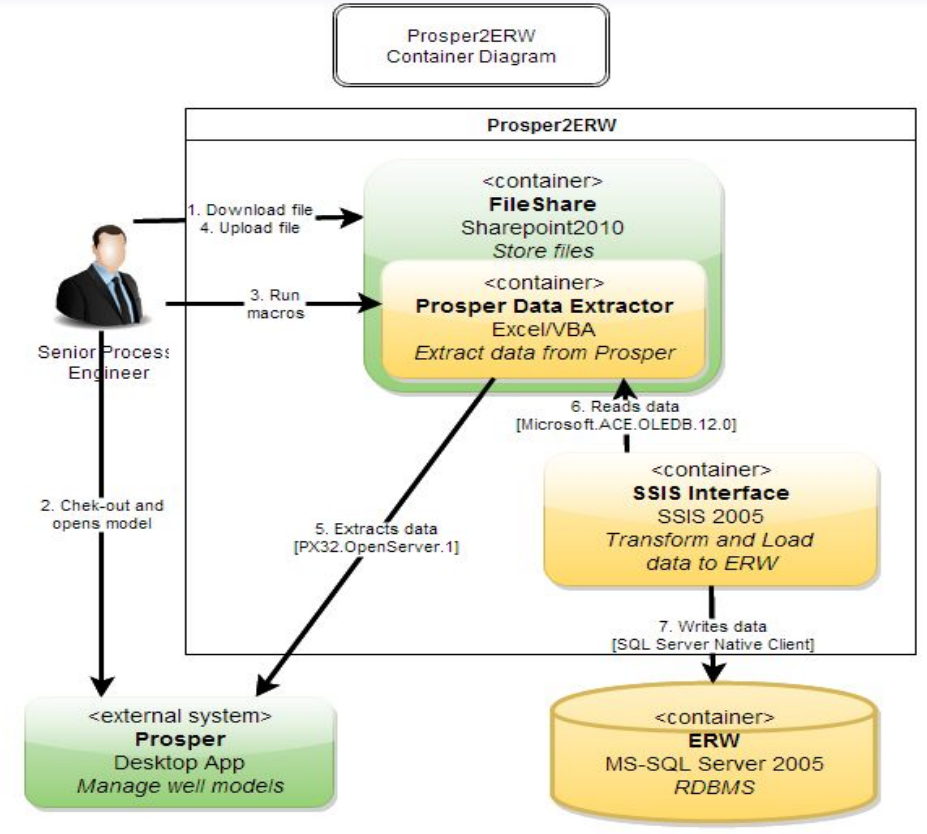
- Makes context explicit - **no assumptions.**
- What is being **added** to an existing **IT environment.**
- Starting point for **discussions** between **technical** and **non-technical** people.
- Who we need to go concerning **inter-system interfaces.**
- Not much detail: help to **set the scene**, starting point for other diagrams.

Audience

- **Technical** and **non-technical** people, **inside** and **outside** project team.

2C: CONTAINER DIAGRAM

- What is the overall shape of the software system?
- What are the high-level technology decisions?
- How are responsibilities distributed across the system?
- How do containers communicate with one another?
- Where do we need to write code to implement features?



Content:

Name
Technology
Responsibilities

Containers - logical executables or processes that make up the software system.

Purpose
Method
Style
[Protocol/port]

→ Inter-container communication
Is inter-process communication.

Motivation

- Makes the high-level **technology choices explicit.**
- Shows **relationships between containers** and how they communicate.
- Provides a **framework** in which to place **components** (components home).
- Provides the **link between** a very high-level **context** diagram and a very cluttered **component** diagram.

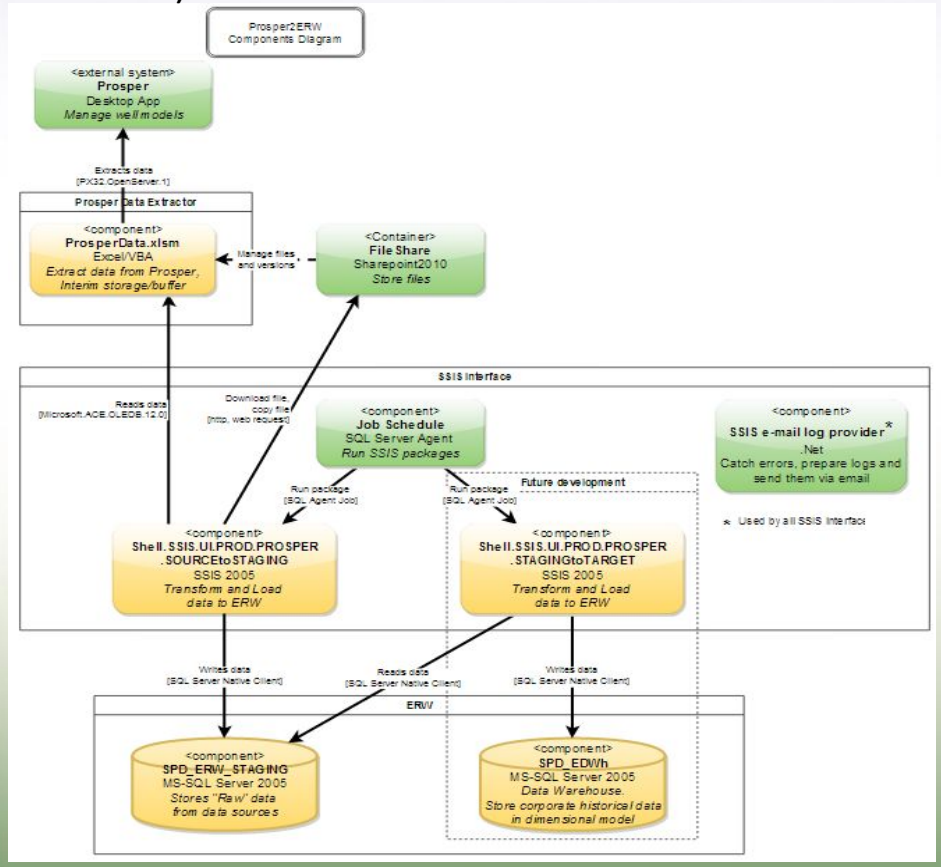
Audience

Technical people inside and outside of the project team: everybody from software **developers** through to **operational** and **support** staff.

3C: COMPONENT DIAGRAM

Zoom in and decompose each container:

- What components/services is the system made up of?
- Is it clear how the system works at a high-level?
- Do all components/services have a home (reside in a container)?



Content:

Name
Technology
Responsibilities

Components are the coarse-grained building blocks of your system

Purpose
Style

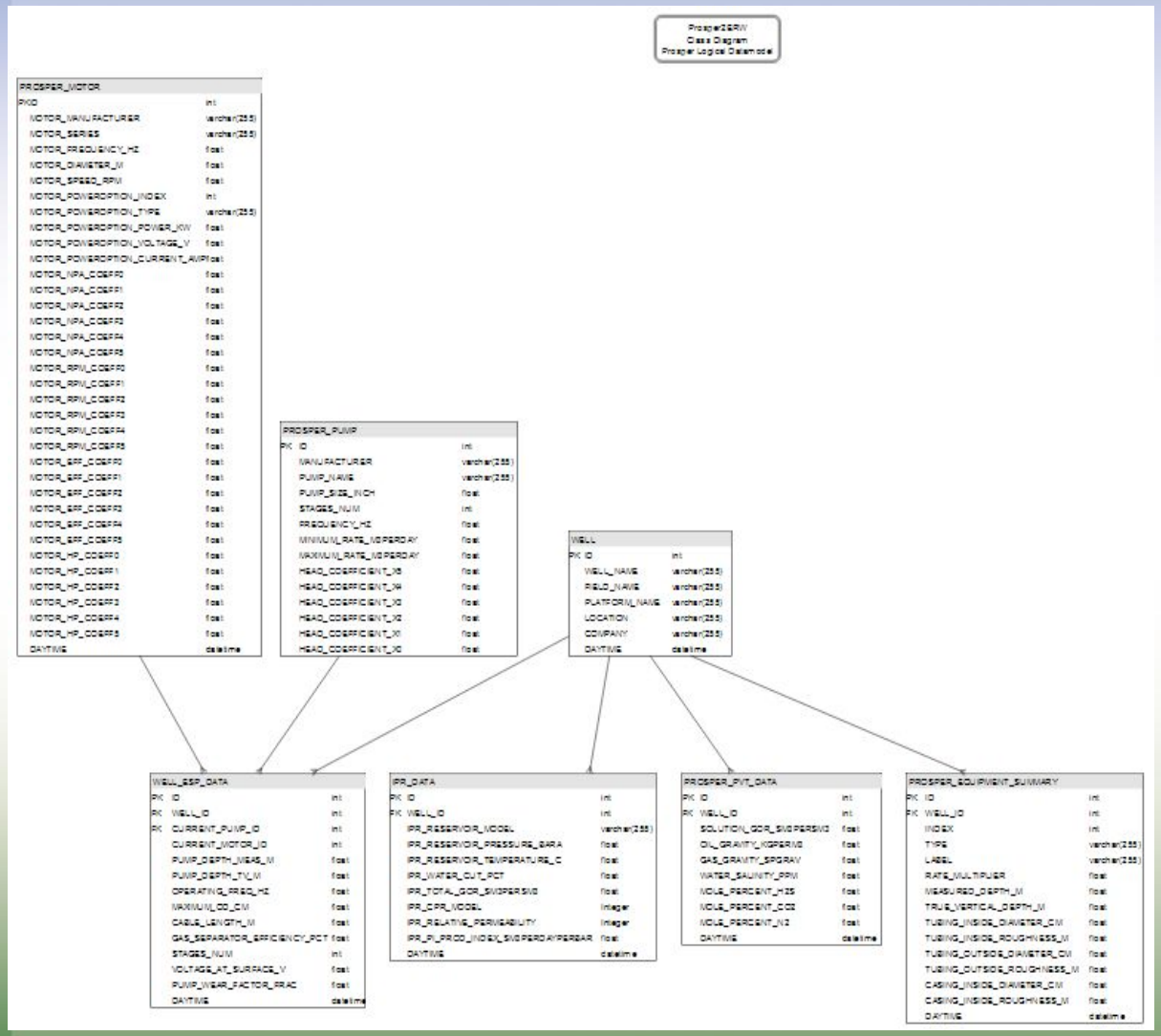
Motivation

- Shows the high-level **decomposition** of your software system into **components** with distinct **responsibilities**.
- Shows **relationships** and **dependencies** between **components**.
- Provides a **framework** for high-level software development **estimates** and how the **delivery** can be broken down (**WBS**).

Audience

Technical people within the software development team

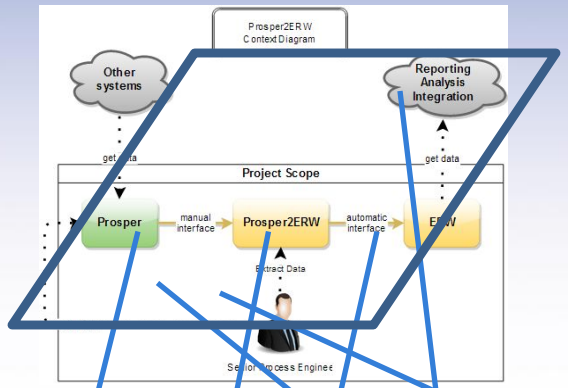
4C: CLASS DIAGRAM (optional?)

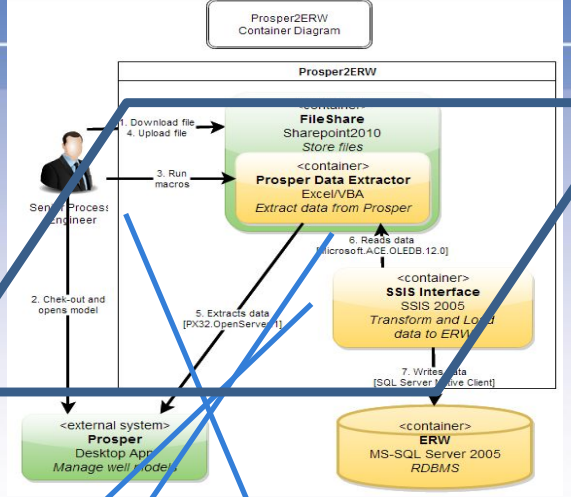


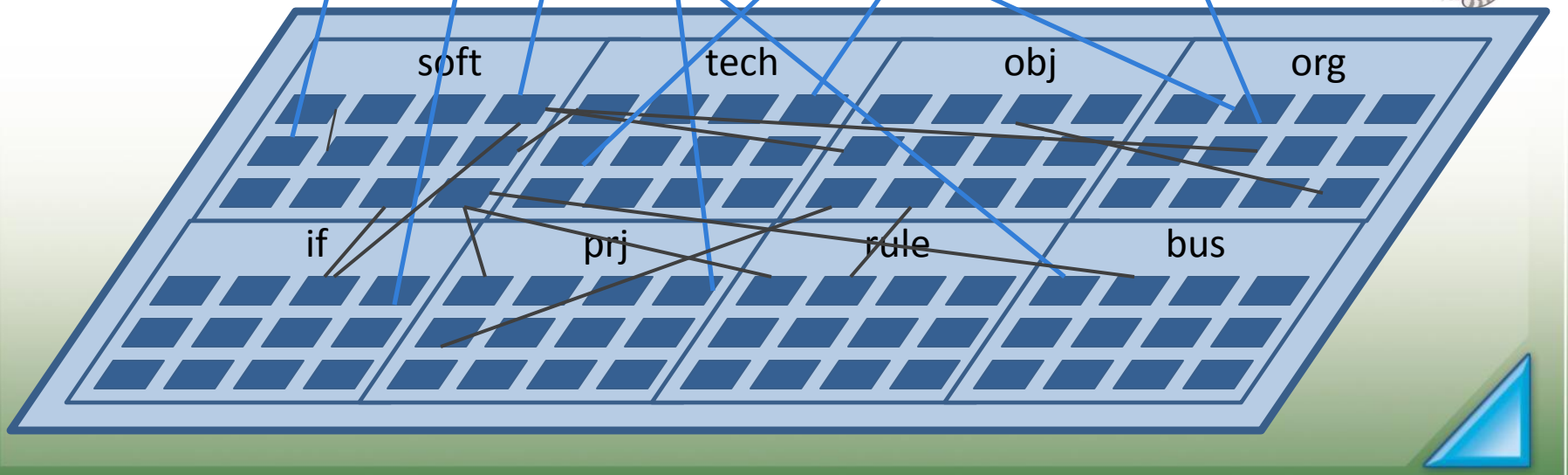
- Is a high-level **UML class diagram.**
- Explains how a particular **pattern** or **component** is **implemented.**
- Classes are the smallest building blocks of our software systems.

Instead of classic UML class-diagram we will use **Conceptual/Logical Data Model Diagram**

SA HLD ON META







IS THIS ENOUGH?

• SA HLD – is not just “word document somewhere in SP”, but power tool which help to:

- assess, collaborate and communicate BRs and technical decisions

- present high-level view on the solution and help to **navigate** throughout the solution

- provides relevant levels of abstraction for different contributors during full product life-cycle (requirements-design-development-testing-deploy-support-decomission).

• This is **not** a complete set of project/tech. documents – this is SA HLD.

(Process diagram, data-models, mapping, detailed design, Deployment diagram etc.)

WHAT IS NEXT?

For all projects:

- SA HLD should be published on meta in “4C”-format.
- Workshops Arch-PM-BA-(BUS) to collaborate requirements and high-level vision. Deliverables: C1 and C2.
- “Architecture checkbox” on ABP when C1-C4 is published on meta.