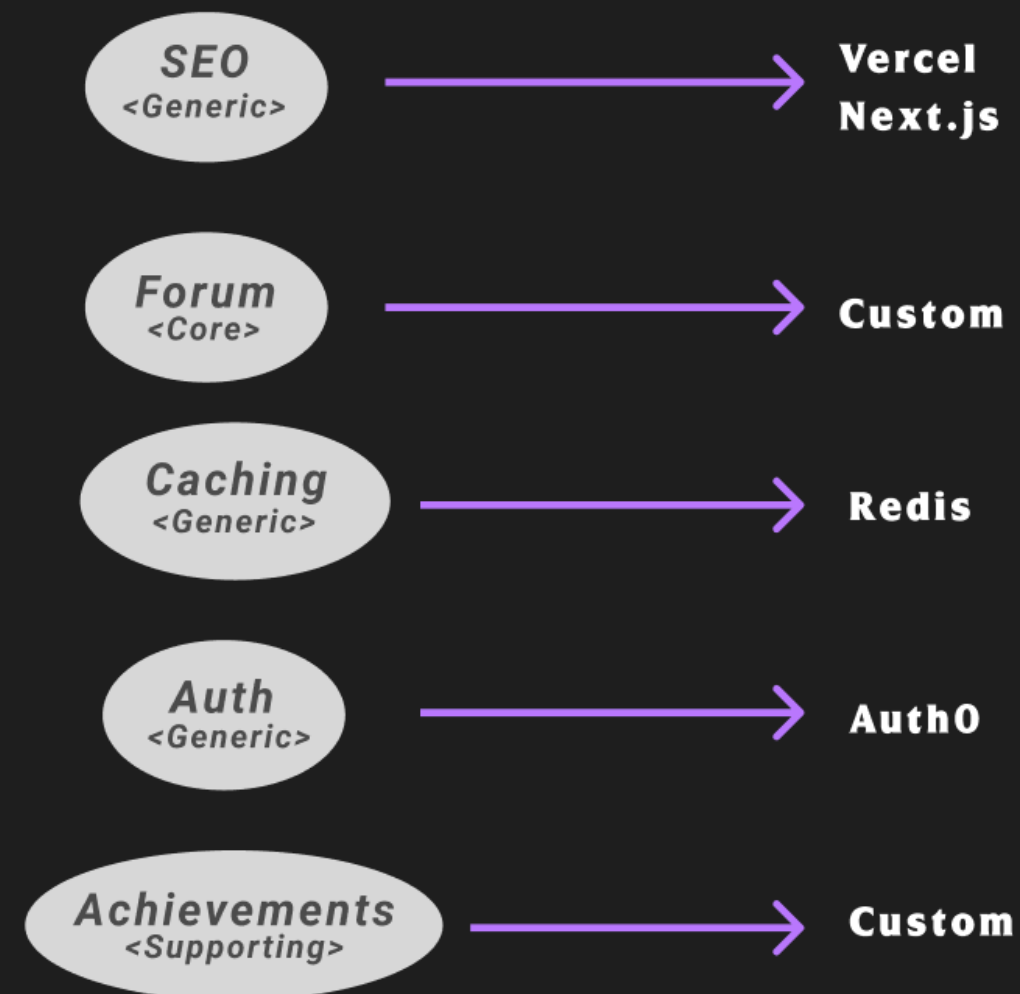




Theoretical Essential: Strategic Design



Theoretical Essential

Strategic Design

Strategic design is making smart decisions about large scale architectural components.

But there's more to architecture than just libraries, tools & frameworks

How do you know which components to use?

How do you know what to build yourself?

How do you integrate the component together?

What do you do when there isn't a component you need?

All developers need to know how to do 4 things:

1. Stereotype components
2. Develop components
3. Integrate components
4. Prioritize components

The Responsibility-Driven Design Process

Start

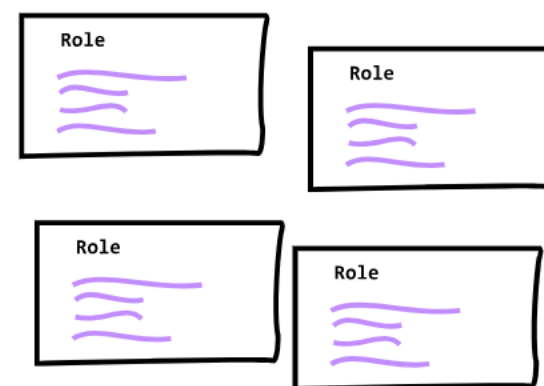


Responsibilities

Extract the responsibilities from the requirements

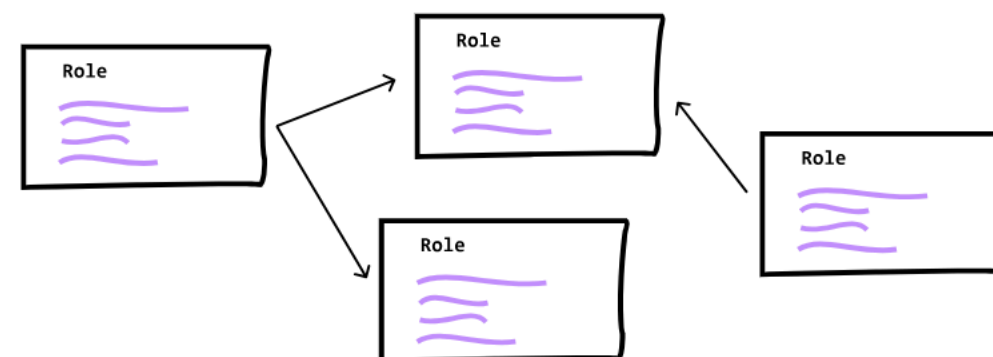
Roles

Assign responsibilities to roles



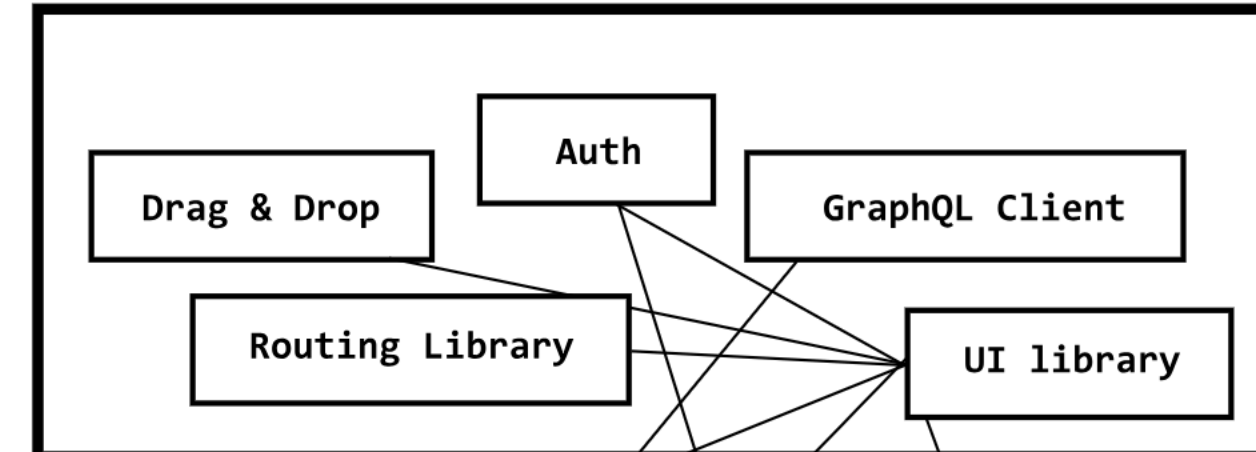
Collaborations

Design collaborations between roles



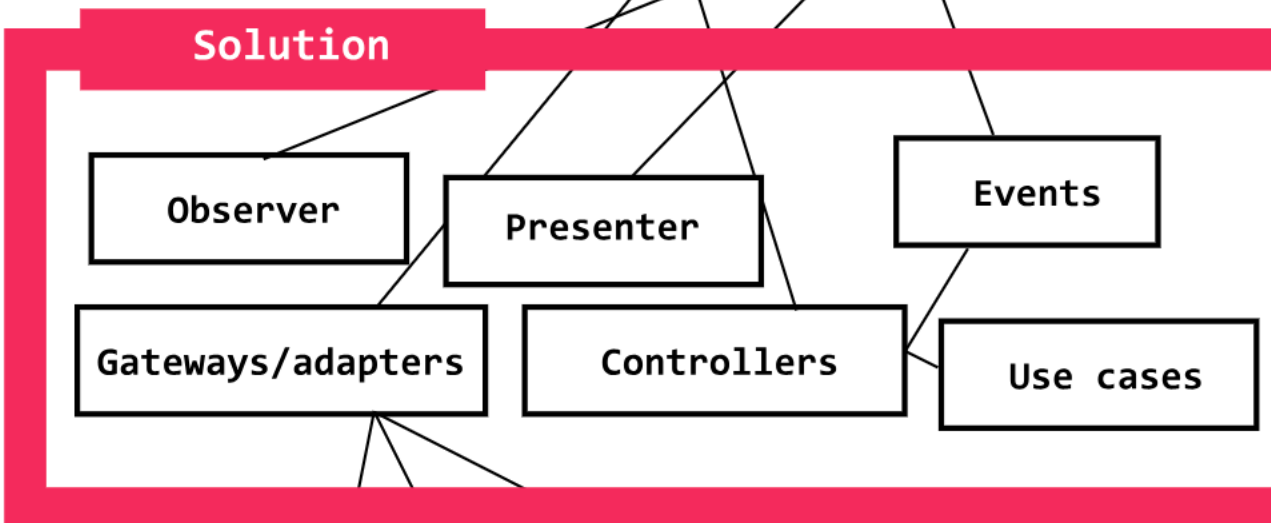
OOP (Object-oriented programming)

Infra
Integrate these

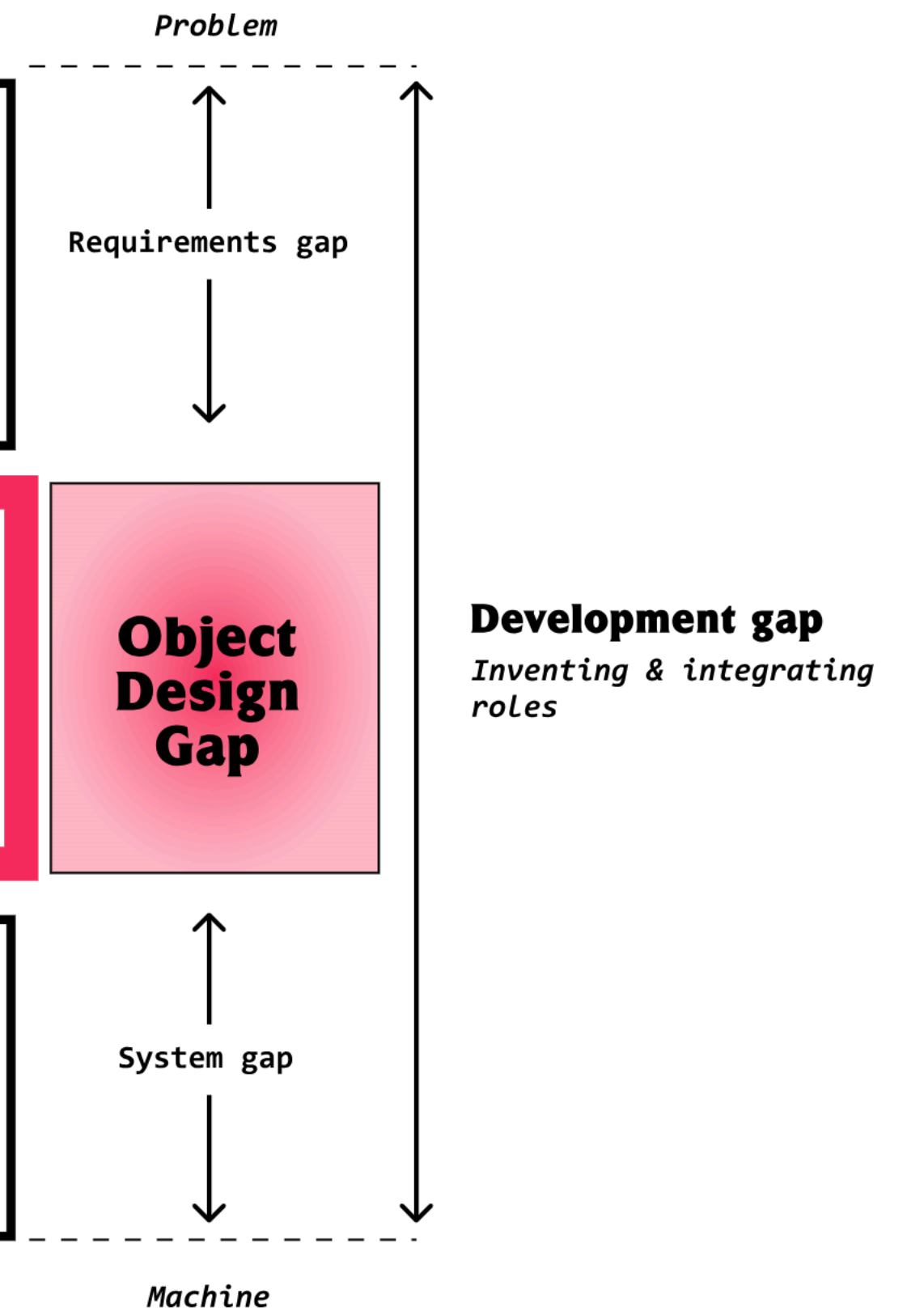
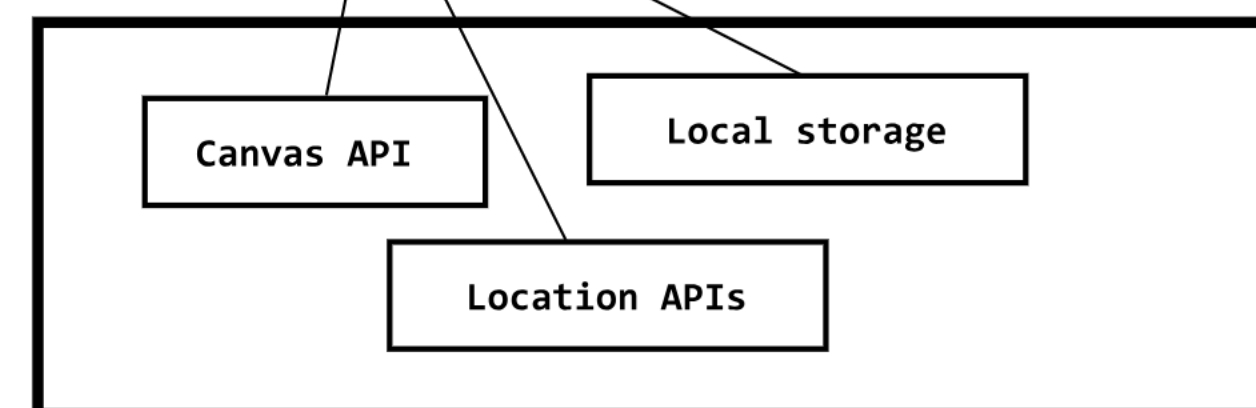


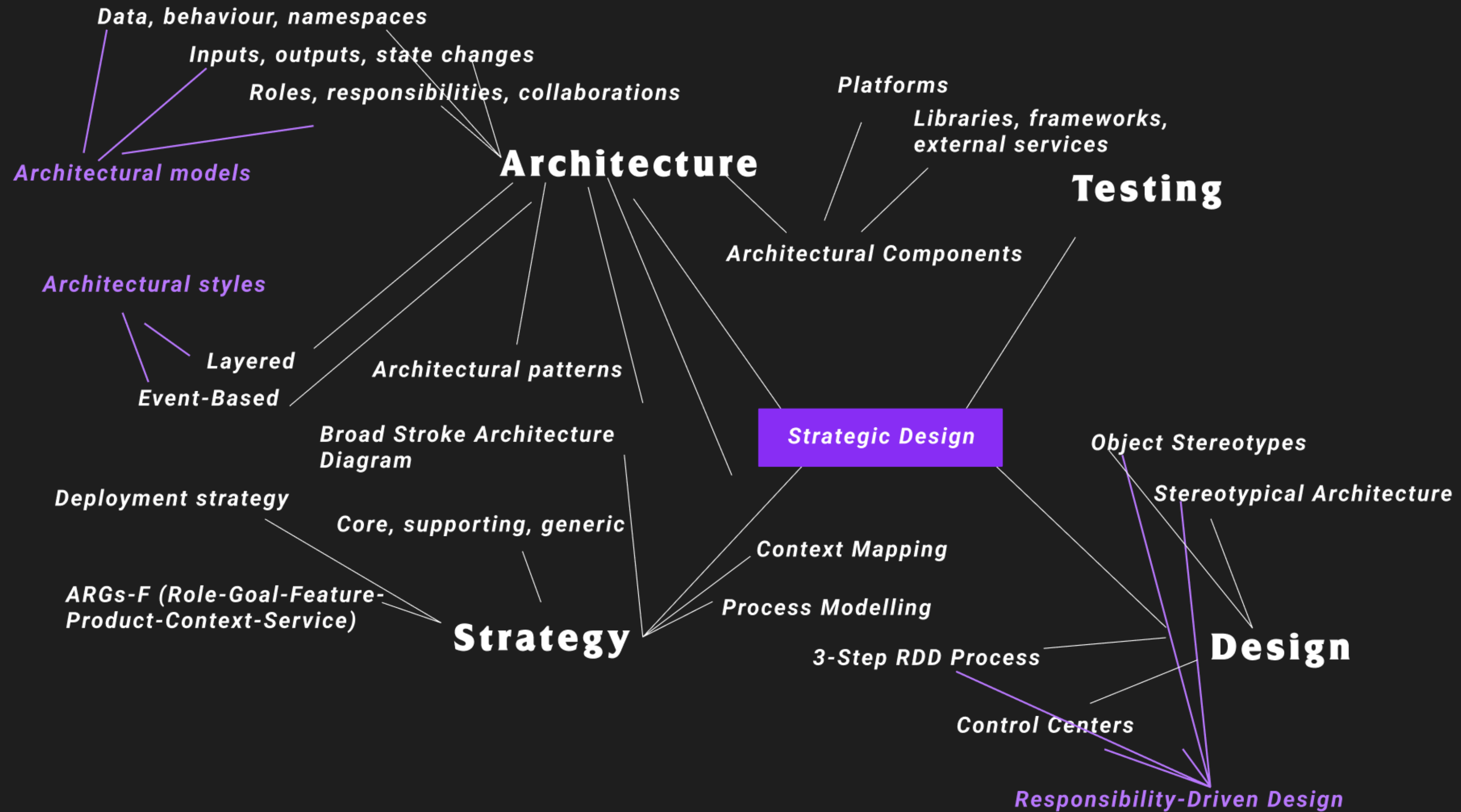
Custom objects (app & domain)

Invent these!



Infra (system)
Integrate these





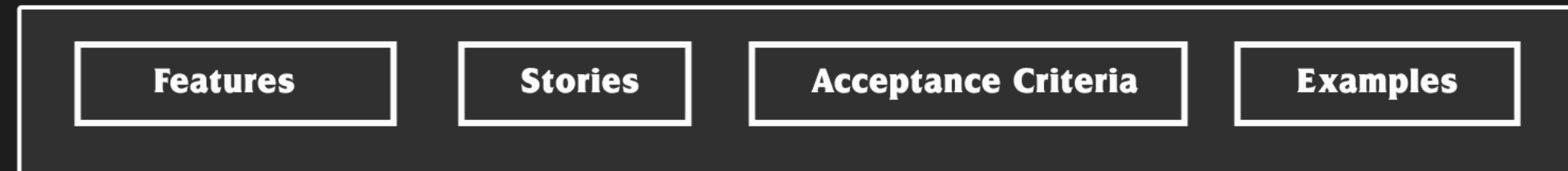
The Theoretical Guesses Points The “Who-What-What”

from the customer

Problem *Who are we helping? What are we trying achieve?
What will they be able to do?*



Solution *How can we help? What will we build? What are the scenarios we need to build? Can we come up with some concrete examples?*



Architecture *What does the system need to be able to do? How should it “be”? What are the responsibilities that need to be handled? Which architectural components will play those roles?*



To the Physical

Where we'll learn more