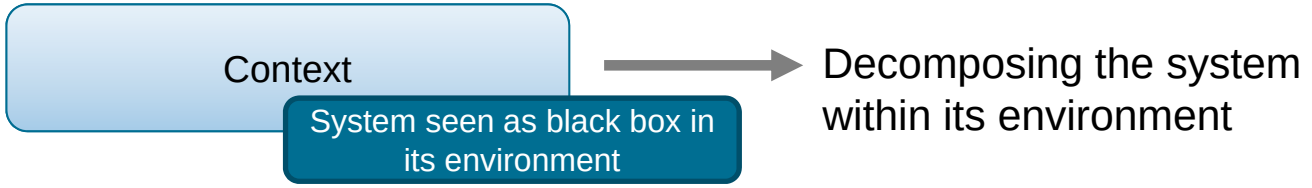
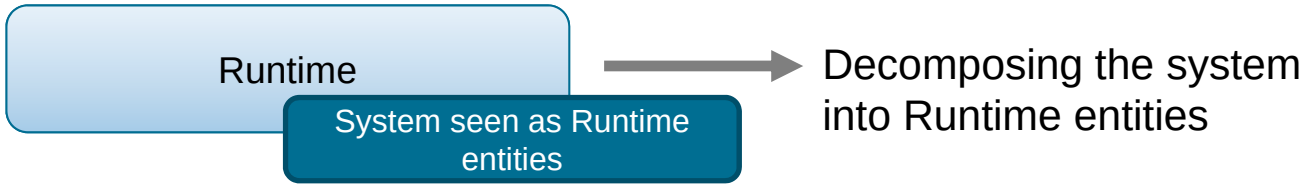
Decomposition Dimensions

- **Decomposition Dimensions**
  address the characteristics of a software system at different **abstraction levels**:

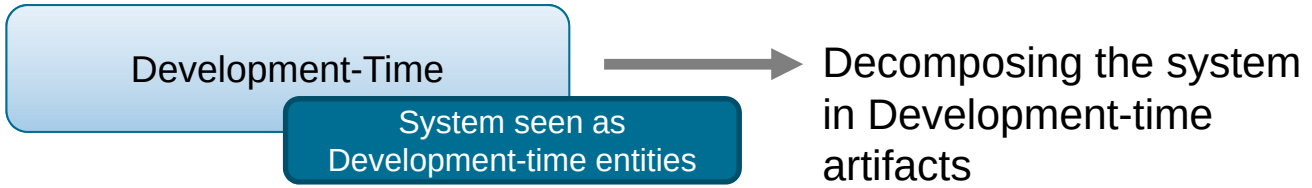
How will the system interact with its environment?



How will the system work at runtime?

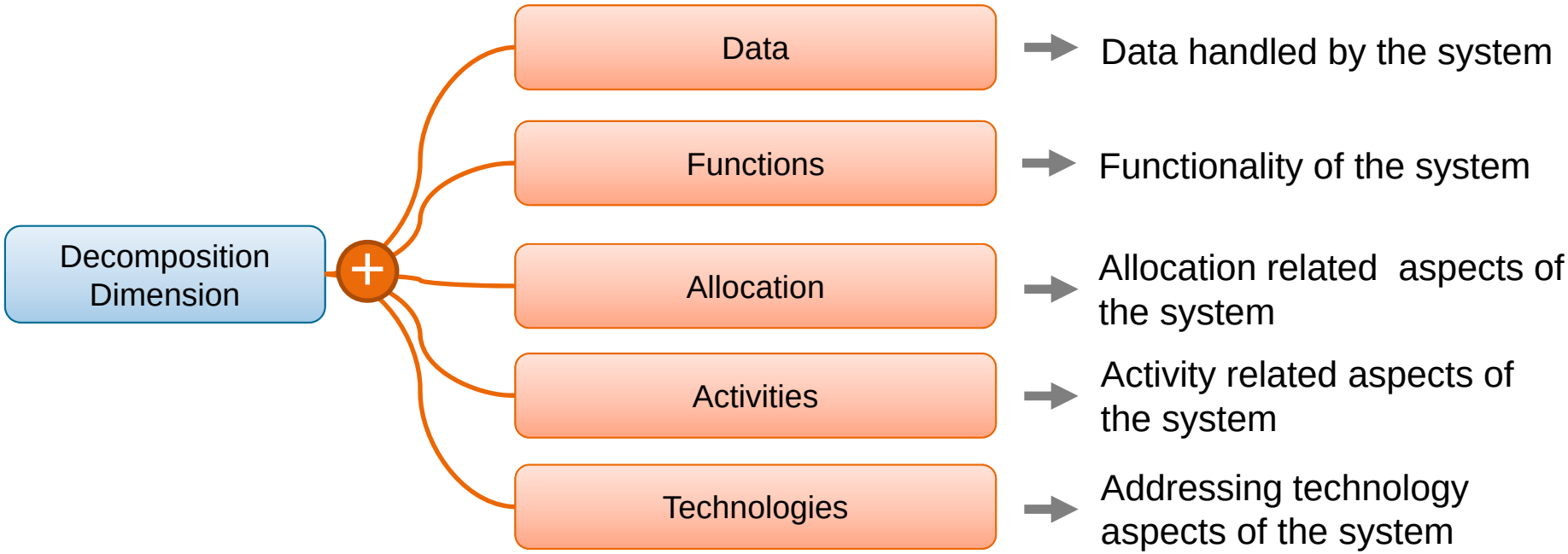


How will the system be implemented?



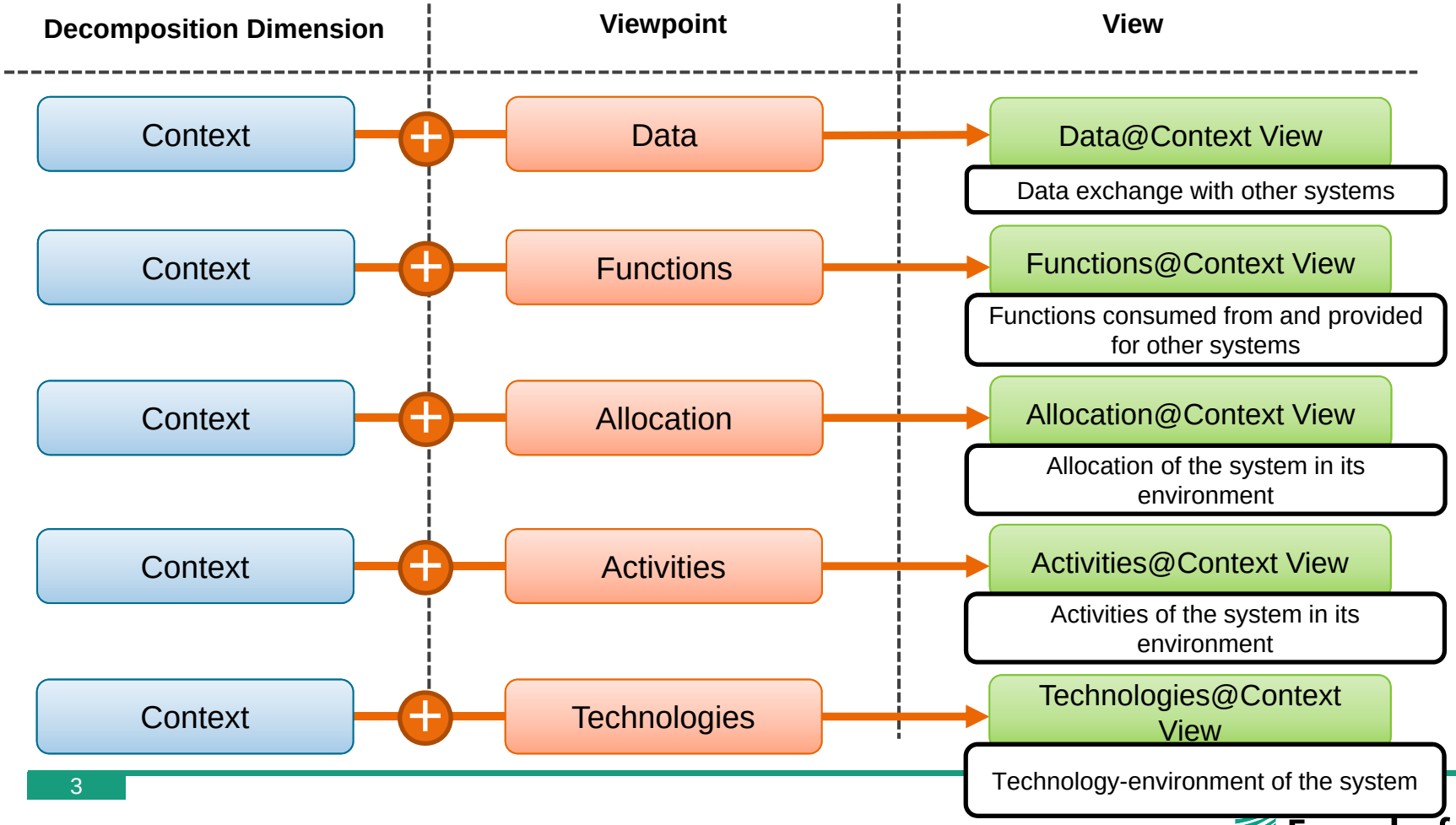
Viewpoints

- **Viewpoints**
  refine and specify the decomposition dimensions
  by addressing **different aspects** of each **dimension**:

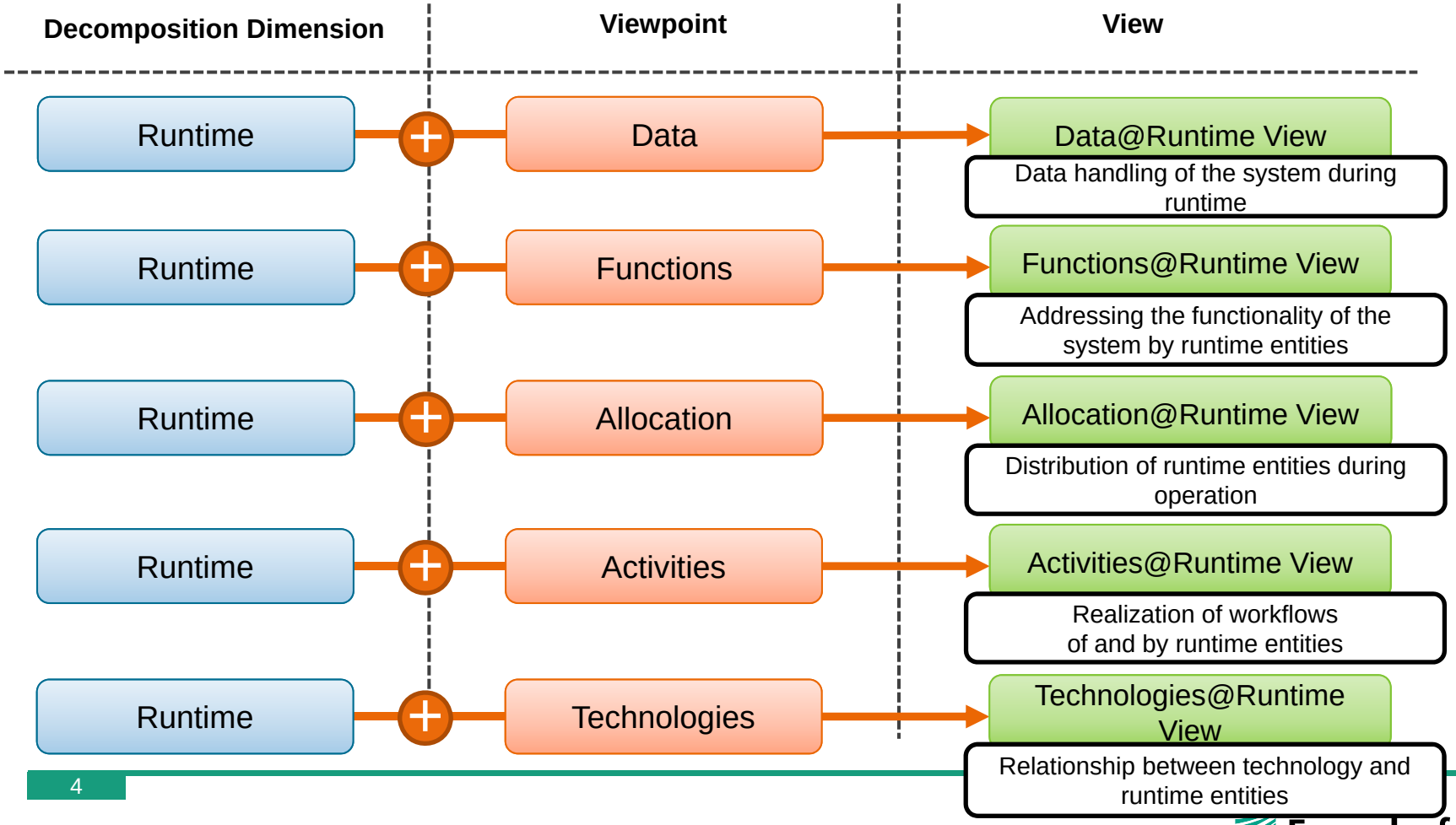


Viewpoints applied to the Context dimension (System = black box)

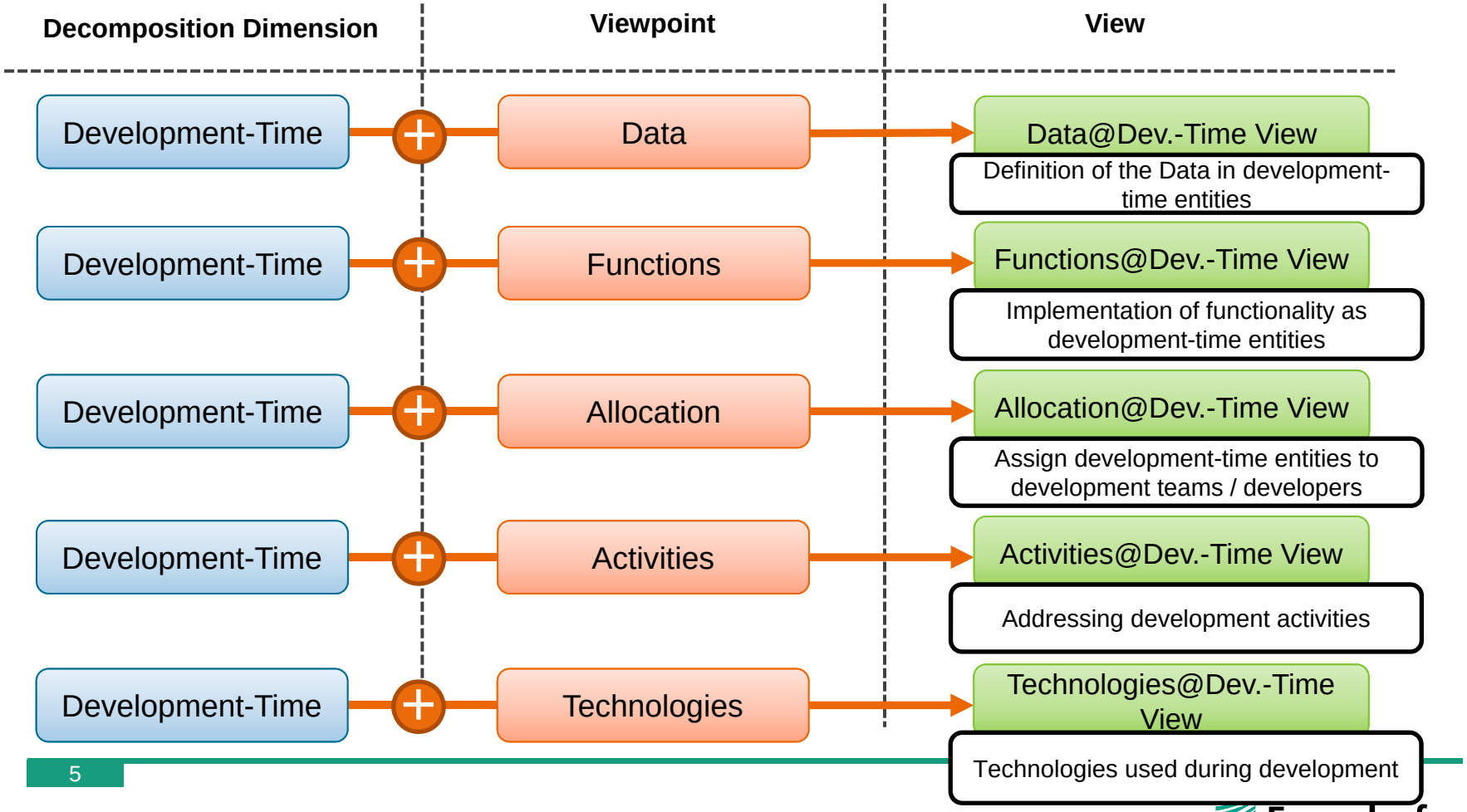
- Combining Dimensions and Viewpoints lead to a specific **Views**



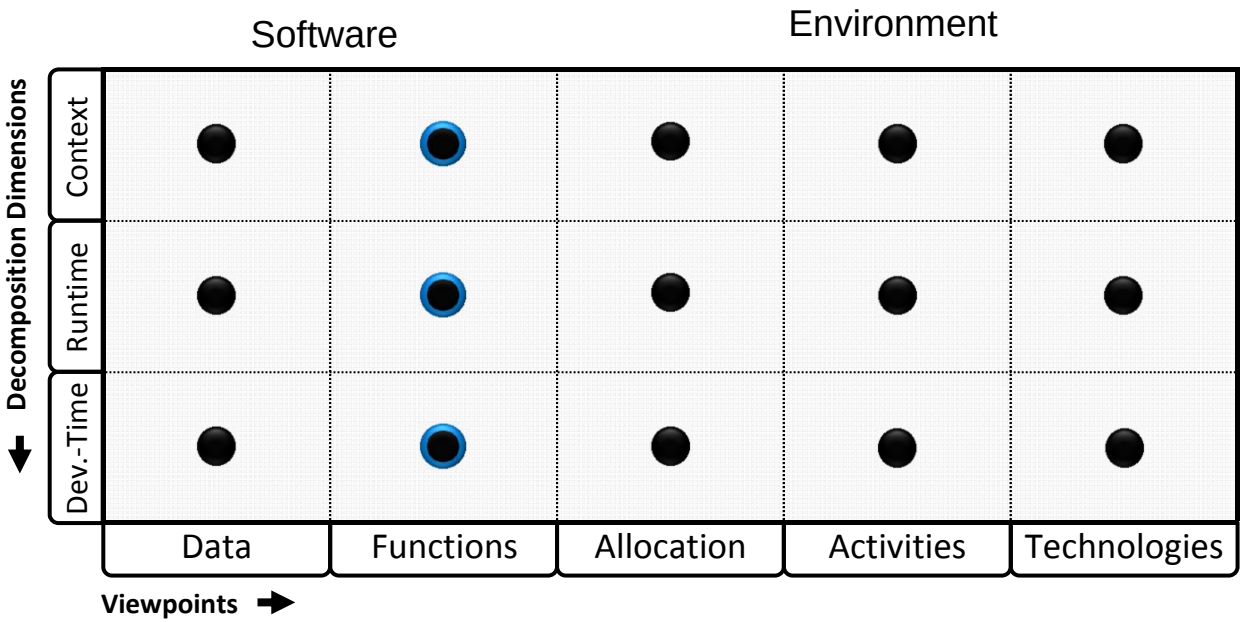
Viewpoints applied to the Runtime dimension (System = runtime entities)



Viewpoints applied to the Development-time dimension (System = development-time entities)



Architecture Decomposition
Software and Environment



The “Hello World Corporation”

partially based on material of @ Prof. Dr. Peter Knauber, HS Mannheim

Stakeholders
Internal and External

Marcus Marketing

Carla Customer

Arnold Architect

Mike Manager

Paul Projectleader

Daniel Developer

Quincy Quality

Fraunhofer IESE

Hello World V1

partially based on material of @ Prof. Dr. Peter Knauber, HS Mannheim

V1: Requirements

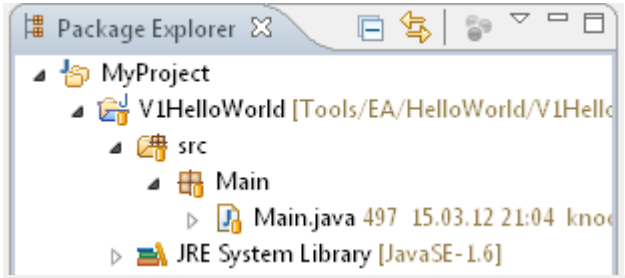
Mike Manager

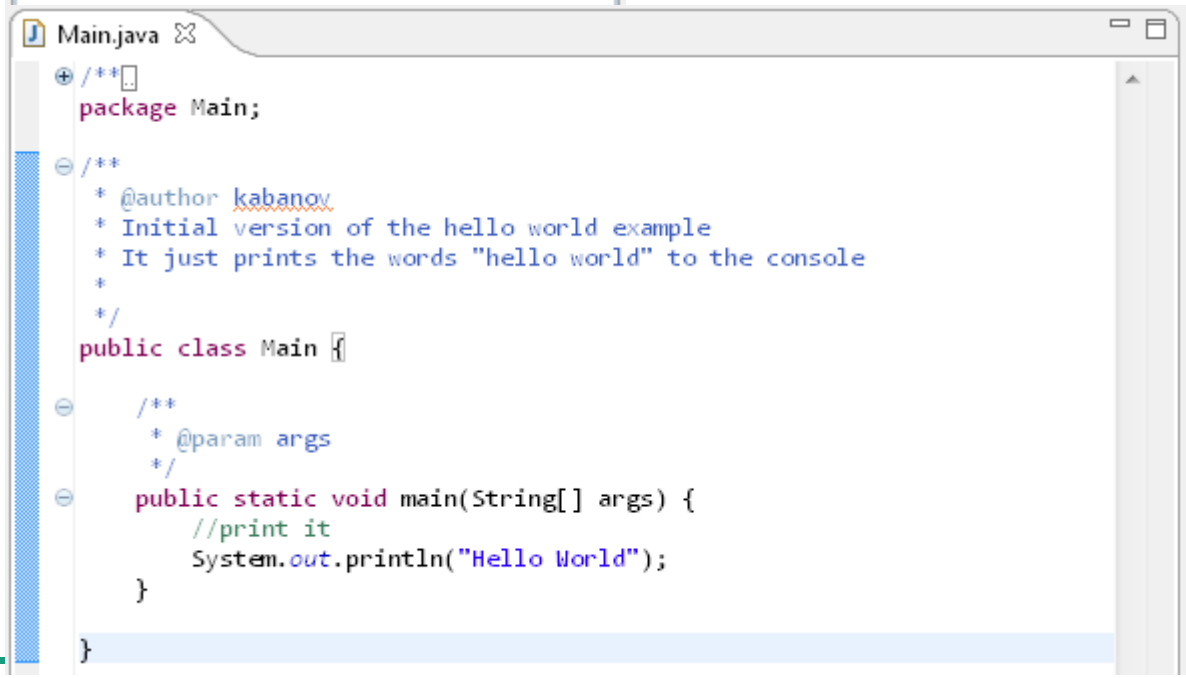
- I have a **great** idea!
  - We build a new product
  - It prints “Hello World”!
  - Nobody else has such an application
  - Let’s start developing it!
  - → We will make a lot of **business**!

V1: Source Code

Daniel Developer

- Well-implemented, prints “Hello World”, Good Job!





V1: Well-designed? Adequate Architecture?

Arnold Architect

→ Discuss with your neighbor

- What architectural decisions were made?
  - on what level of abstraction?
    - → Context (Ctx), RunTime (RT), DevelopmentTime (DT)
  - about which aspects?
    - → Data, Functions, Allocation, Activities, Technologies
  - in perspective of which quality attributes?
    - → So-called “-ilities”,
      e.g., performance, maintainability, reliability, security, extensibility, …

V1: Architecture
Decision 01: Hello World Function (Functions@Ctx, @DT)

Arnold Architect

- Architecture driver
  - → Business goal
- Rationales
  - → Achieve the key functional requirement to print hello world
  - → Module “Main”
- Consequences
  - → Developer responsible for implementing module “Main”
- Alternatives
  - Classes “Hello” & “World”
  - …

V1: Architecture
Decision 02: Technology Java (Technology@Ctx)

Arnold Architect

- Architecture driver
  - → Business goal
- Rationales
  - → All developers are able to implement and understand source code written in Java
- Consequences
  - → Java is required skill for new developers

- Alternatives
  - C/C++
  - Ruby on Rails
  - Perl
  - Cobol
  - Assembler
  - …

V1: Architecture
Decision 03: Technology IDE Eclipse (Technology@DT)

Arnold Architect

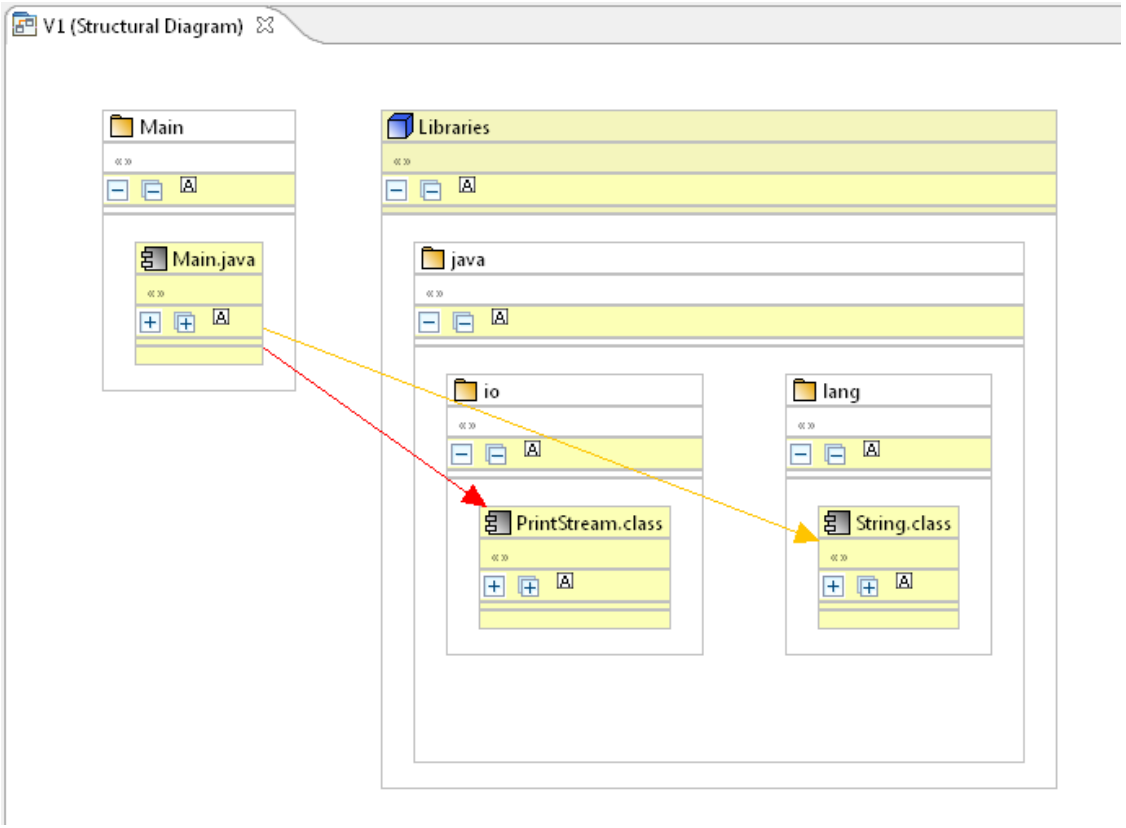
- Architecture driver
  - → Business goal
- Rationales
  - → We have made good experiences with using the Eclipse IDE for developing Java applications
- Consequences
  - → Eclipse reference installation (pre-configured plug-ins) is required at development time

- Alternatives
  - NetBeans
  - IntelliJ
  - NotePad
  - Emacs
  - …

V1: Quality Assurance

Quincy Quality

- Well-Realized, Good Quality!



Fraunhofer IESE

Architecture Decomposition
Software and Environment