



دانشگاه کردستان
University of Kurdistan
زانکۆی کوردستان

4+1 view Software Architecture

Sadegh Sulaimany
info@Bioinfotmation.ir

Software Architecture Course

Initial assessment

- › Draw a popular picture about 4+1 view of software architecture


Agenda

- › History
- › Stakeholders
- › Views
 1. Use-case
 2. Logical
 3. Implementation
 4. Process
 5. Deployment
- › Example

History

- Philippe Kruchten worked for Rational Software Corp, the preeminent seller of software development tools at the time, in 1995



Google Scholar "4+1 view" 

Articles About 4,500 results (1.24 sec)

Any time
Since 2023
Since 2022
Since 2019
Custom range...

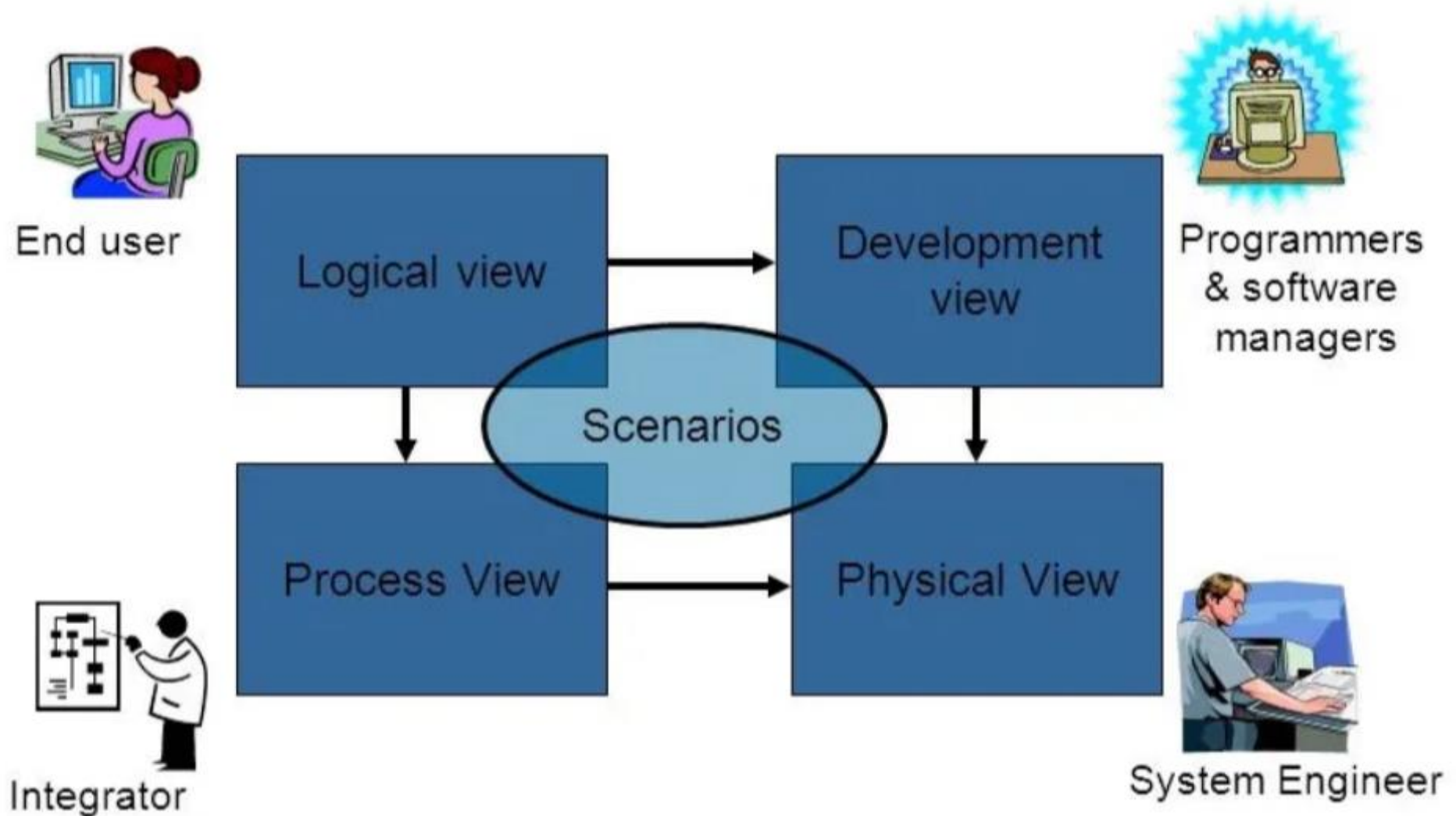
The 4+ 1 view model of architecture
[PB Kruchten - IEEE software, 1995 - ieeexplore.ieee.org](#)
The **4+1 View Model** organizes a description of a software architecture using five concurrent views, each of which addresses a specific set of concerns. Architects capture their design ...
☆ Save  Cite Cited by **4129** Related articles All 83 versions



History

- Years of experience creating large-scale complex software systems in telecommunications, aircraft, transportation, and defence had taught him that
 - › software architecture diagrams frequently failed to convey the actual system design. The meanings of the boxes, lines, and arrows were frequently unclear, and stakeholders struggled to find the information they required.
- In a paper released that year, Architectural Blueprints — The “4+1” View Model of Software Architecture,
 - › he offered a technique for organizing the description of a software architecture using a set of concurrent “views,” each addressing unique concerns for distinct stakeholders. That is how the 4+1 Architectural Model Originated.

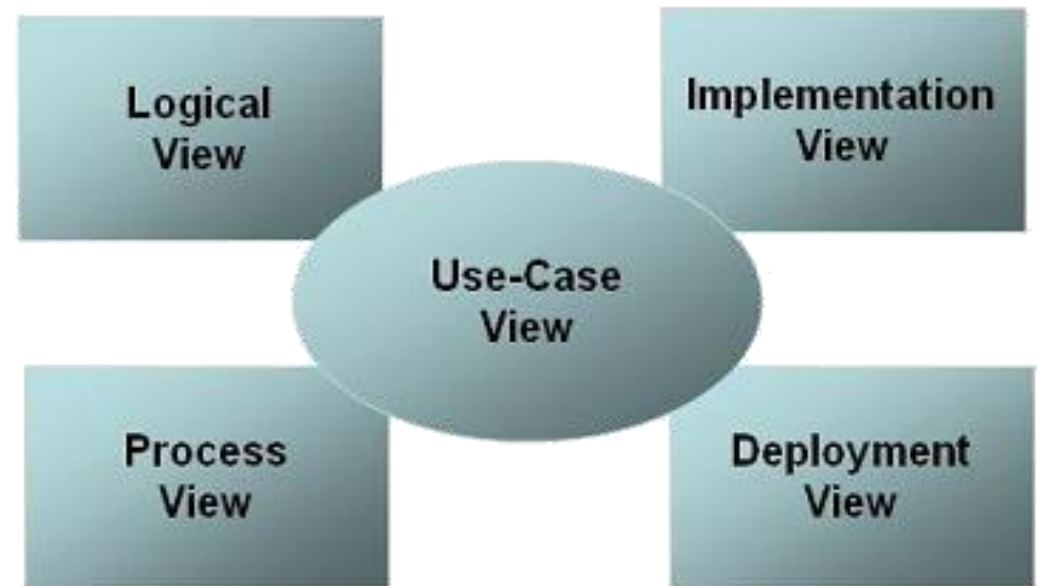
Stakeholders



4+1 views

› Use-case view:

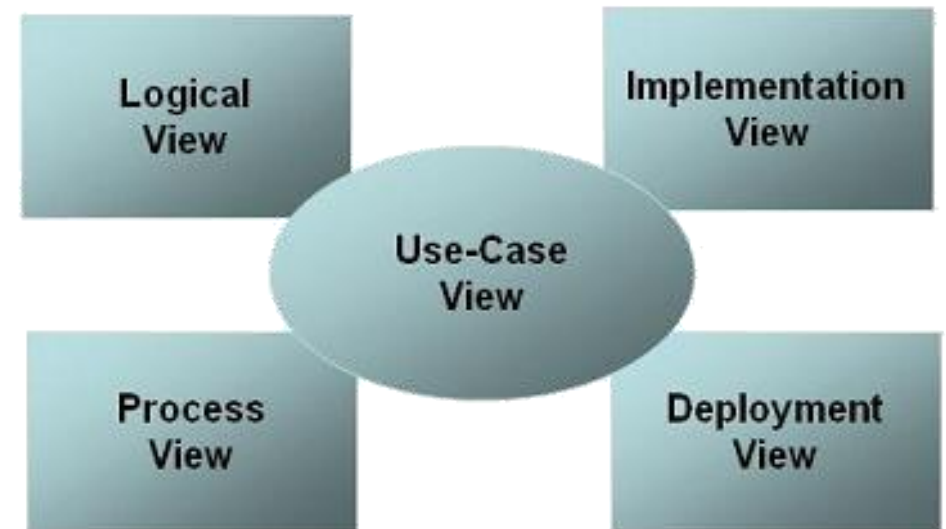
- Describes functionality of the system, its external interfaces, and its principal users.
- This view is mandatory when using the 4+1 Views, because all elements of the architecture should be derived from requirements.



4+1 views

› Logical view:

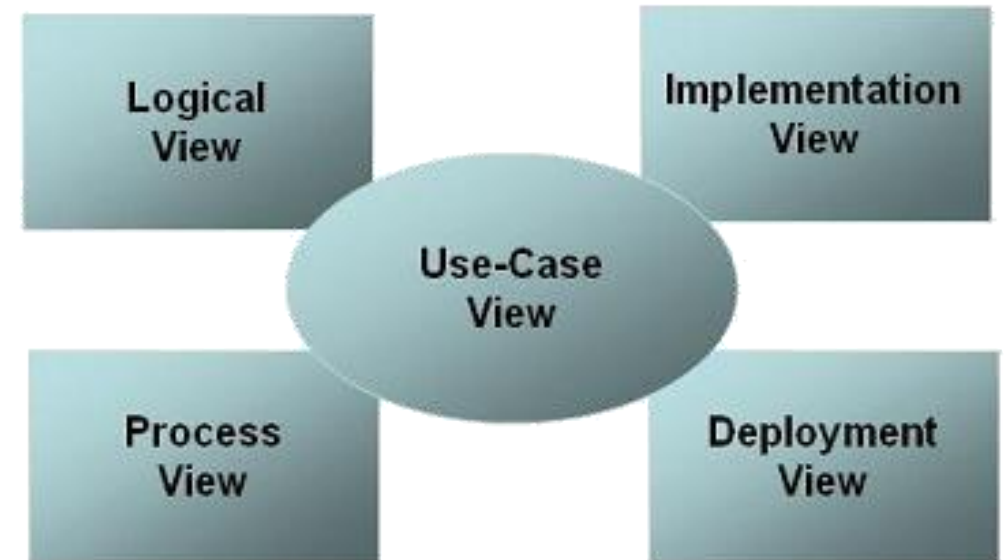
- Describes how the system is structured in terms of units of implementation.
 - › The elements are packages, classes, and interfaces.
- The relationship between elements shows dependencies, interface realizations, part-whole relationships, and so forth.



4+1 views

› Implementation view:

- Describes how development artifacts are organized in the file system. The elements are files and directories (any configuration items)
- It describes system components using the UML Component diagram
- Sometimes called
 - › Implementation view



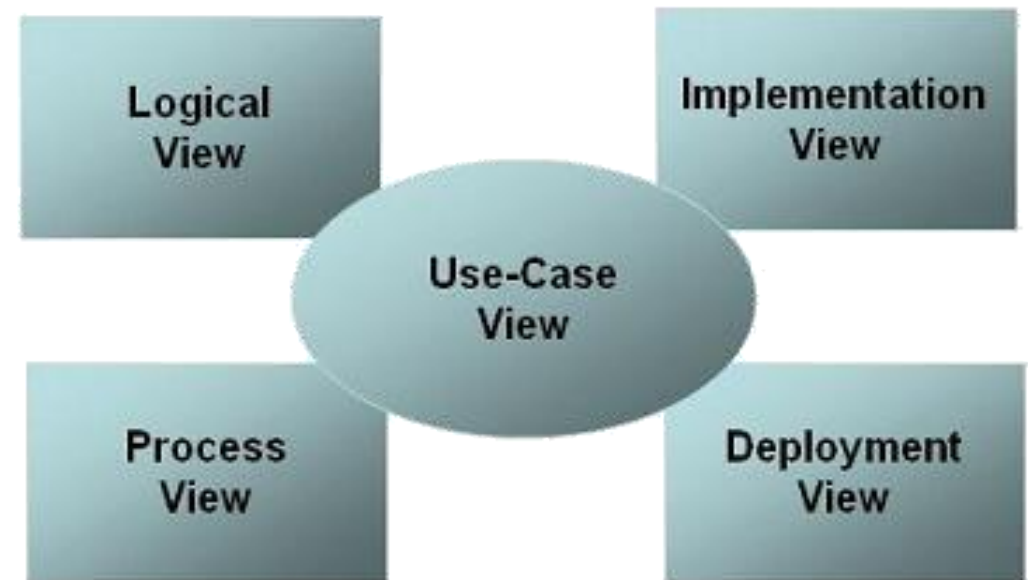
4+1 views

› Process view:

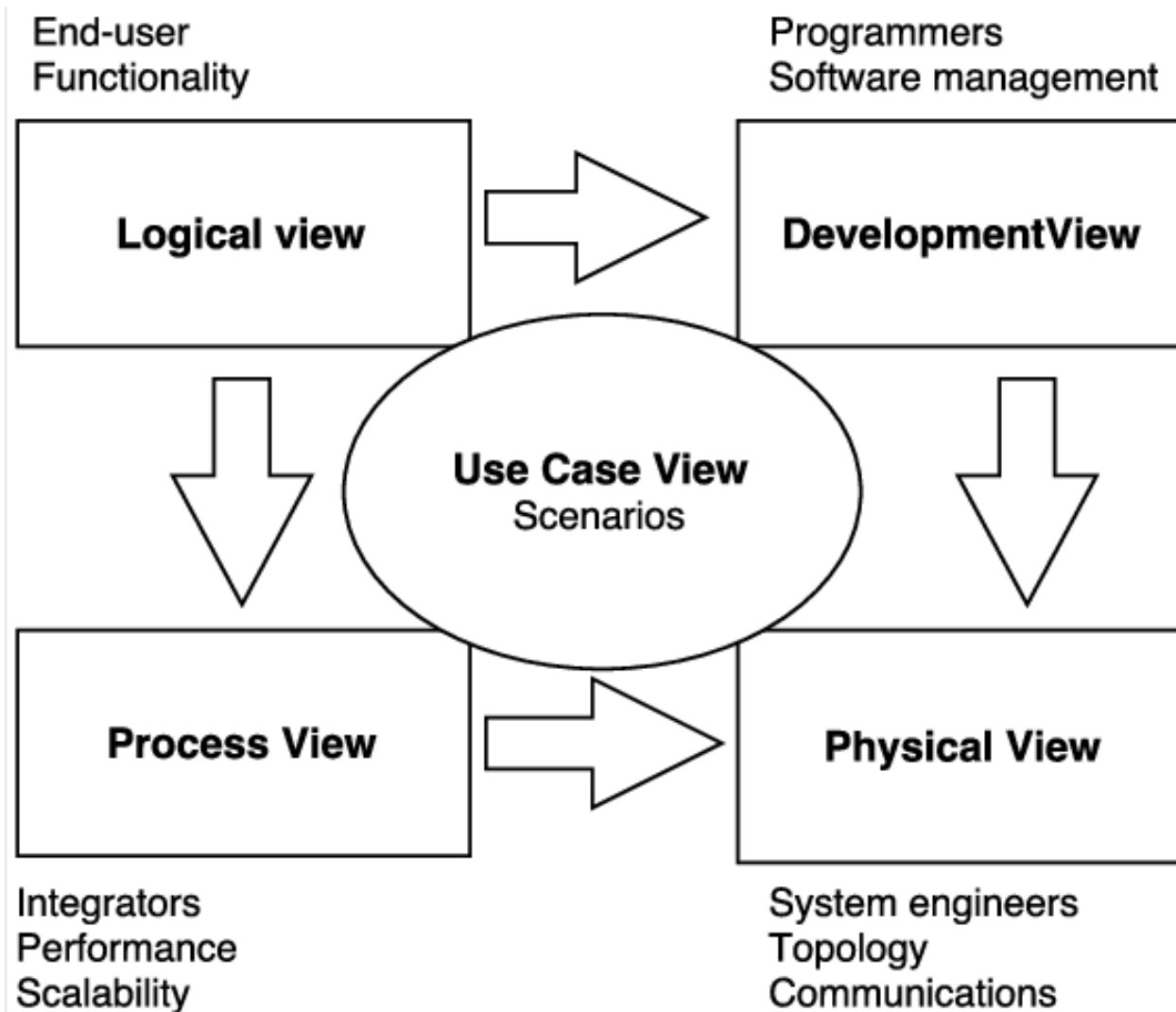
- Describes how the run-time system is structured as a set of elements that have run-time behavior and interactions.
- The elements are components that have run-time presence (processes, threads, Enterprise JavaBeans™ (EJB™), servlets, DLLs, and so on), data stores, and complex connectors, such as queues.
- Interaction between elements varies, based on technology. This view is useful for thinking about run-time system quality attributes, such as performance and reliability.

4+1 views

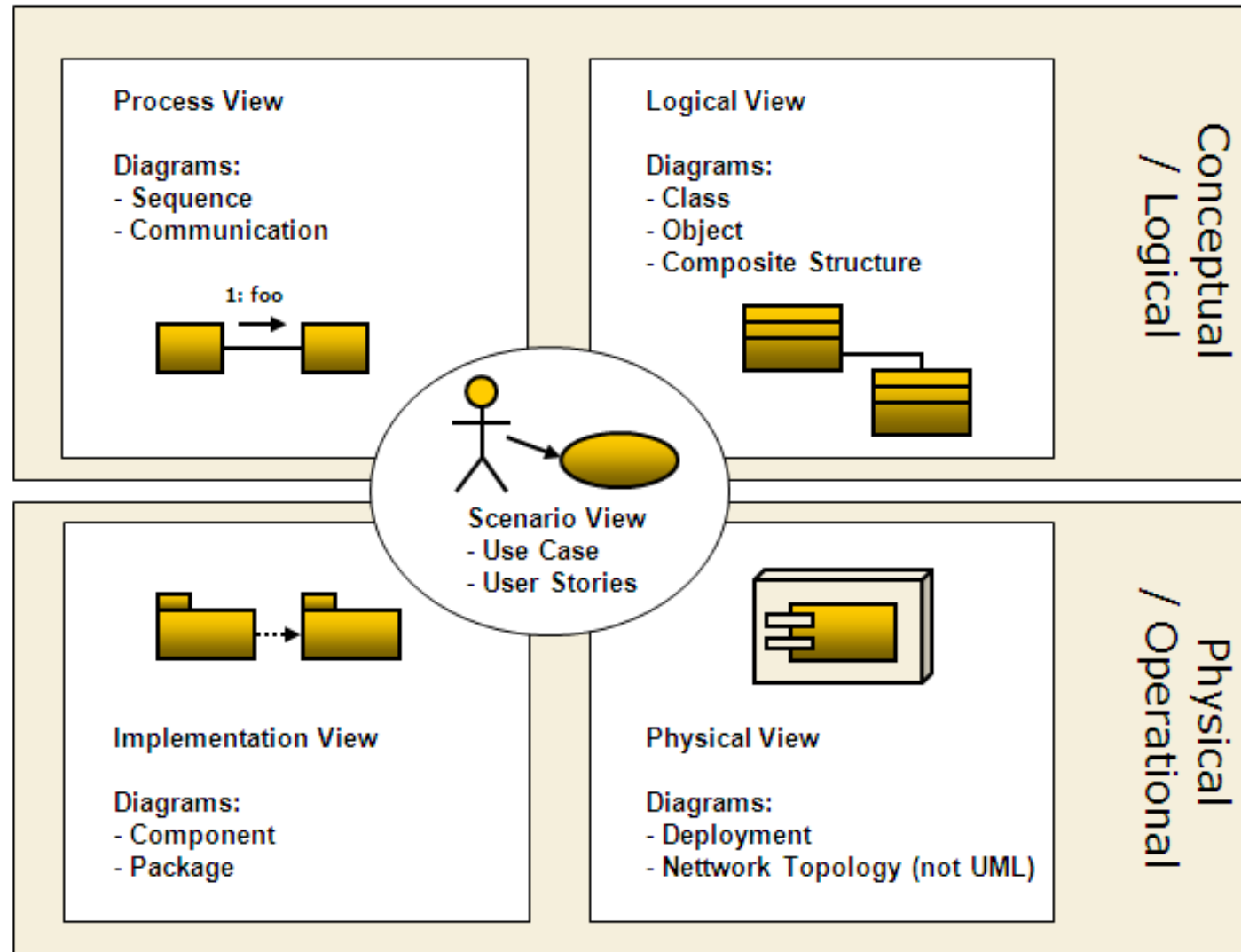
- › Deployment view:
 - Also known as physical view
 - Describe how the system is mapped to the hardware.
 - May be described with UML deployment diagram



Example



Example



Question?

Bioinformation.ir

info@Bioinformation.ir

SOFTWARE ARCHITECTURE

