4+1 View Model of Software Architecture

- Problem
- Solution
- 4+1 view model
  - Logical view
  - Process view
  - Development view
  - Physical view
  - Scenarios
- The Iterative process

Problem

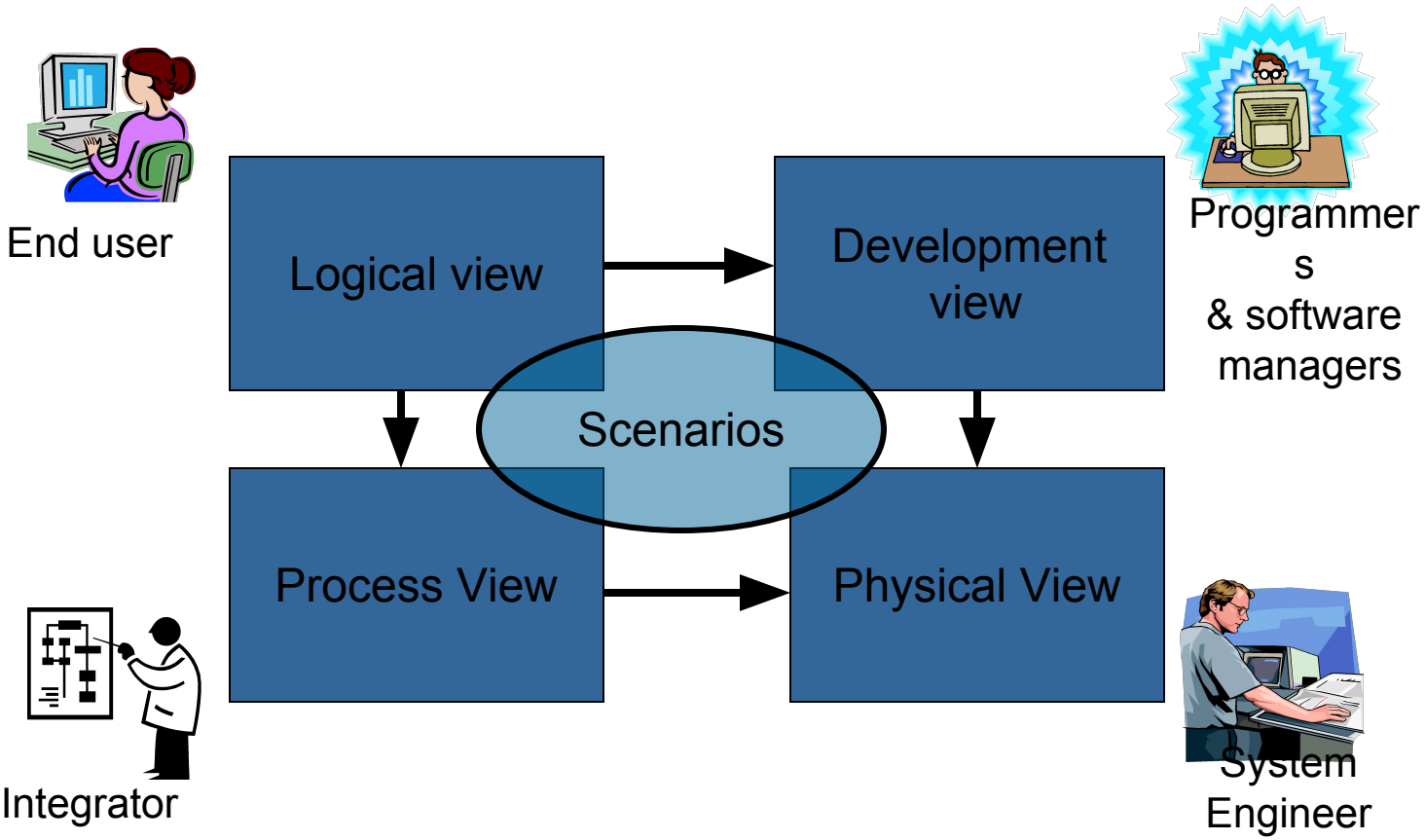
- Arch. documents over-emphasize an aspect of development (i.e. team organization) or do not address the concerns of all stakeholders
- Various stakeholders of software system: end-user, developers, system engineers, project managers
- Software engineers struggled to represent more on one blueprint, and so arch. documents contain complex diagrams.

Solution

- Using several concurrent *views* or *perspectives*, with different notations each one addressing one specific set for concerns

- “4+1” view model presented to address large and challenging architectures

4+1 View Model of Architecture



Logical View

(Object-oriented Decomposition)

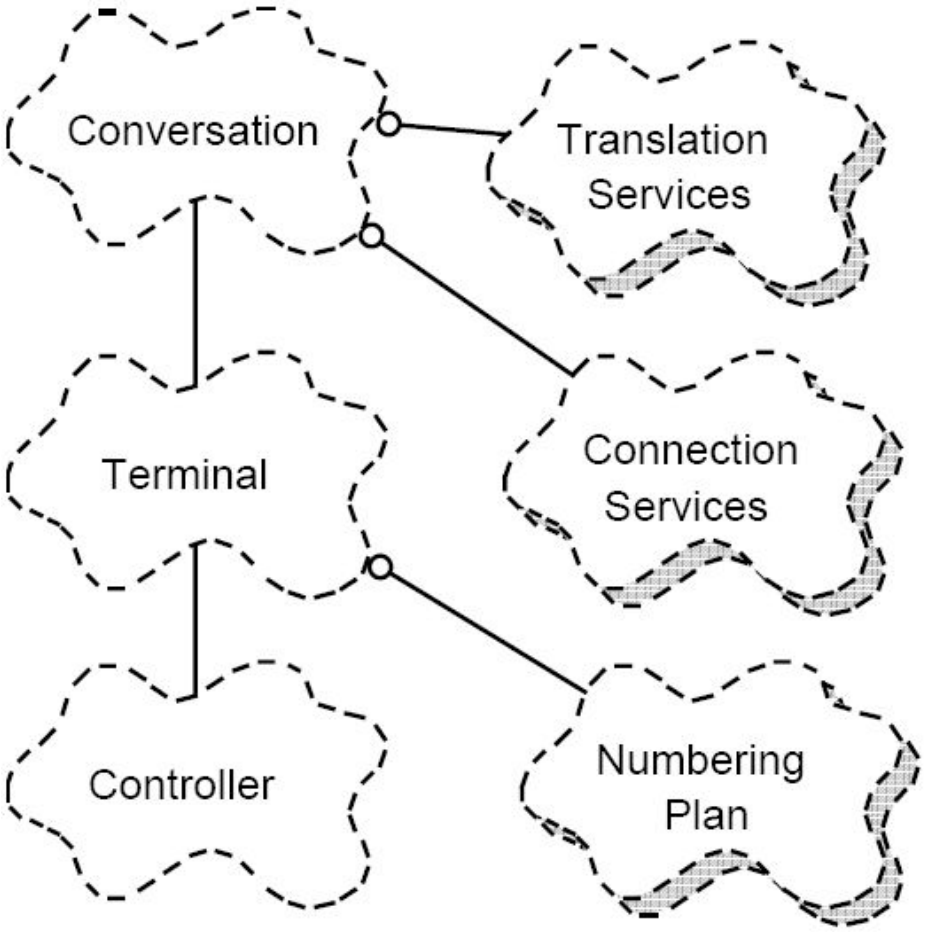
Viewer: End-user

considers: Functional requirements- What the system should provide in terms of services to its users.

Notation: The Booch notation (OMT) (object and dynamic models)

Tool: Rational Rose

Logical view Example



Process View

(The process decomposition)

viewer: Integrators

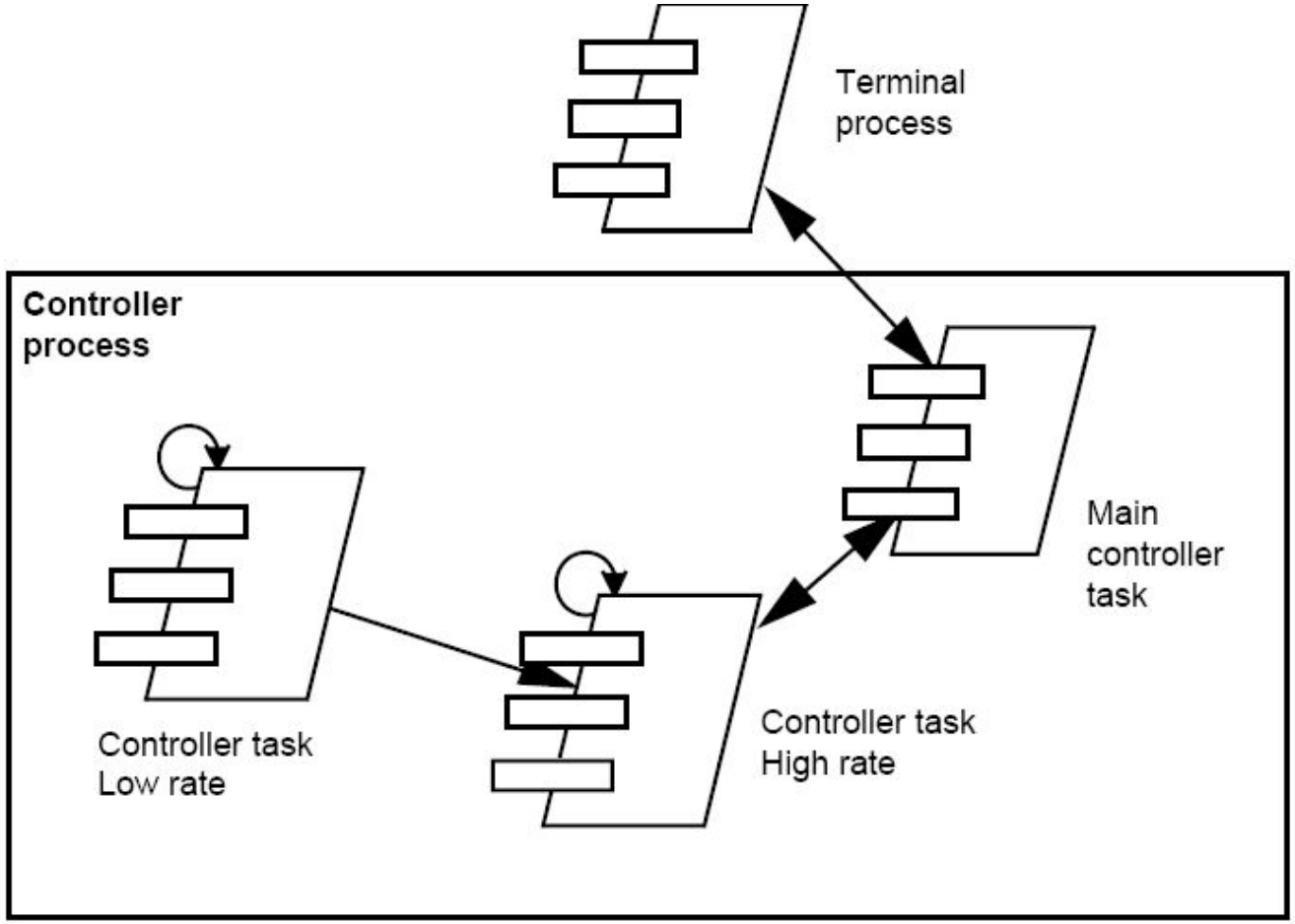
considers: Non - functional requirements (concurrency, performance, scalability)

style: Several styles would fit in this view (Garlan and Shaw ‘s Architecture styles)

Process view (cont.)

- Uses multiple levels of abstractions, a logical network of processes at the highest level
- A process is a grouping of tasks that form an executable unit:
  - Major Tasks: Arch. relevant tasks
  - Minor Tasks: Helper tasks. (Buffering)

Process View example



Development View

(Subsystem decomposition)

Basis of a line of product

Viewer: Programmers and Software Managers

considers: software module organization (Hierarchy of layers, software management, reuse, constraints of tools)

Style: layered style

Notation: the Booch notation (module, subsystem, layer)

Physical View

(Mapping the software to the Hardware)

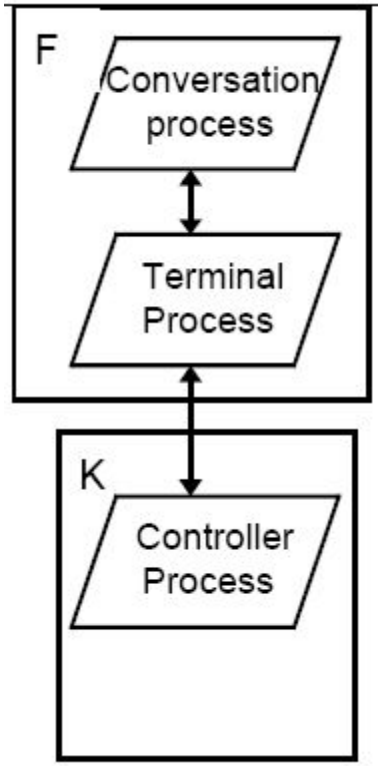
Viewer: System Engineers

Considers: Non-functional req. regarding to underlying hardware (Topology, Communication)

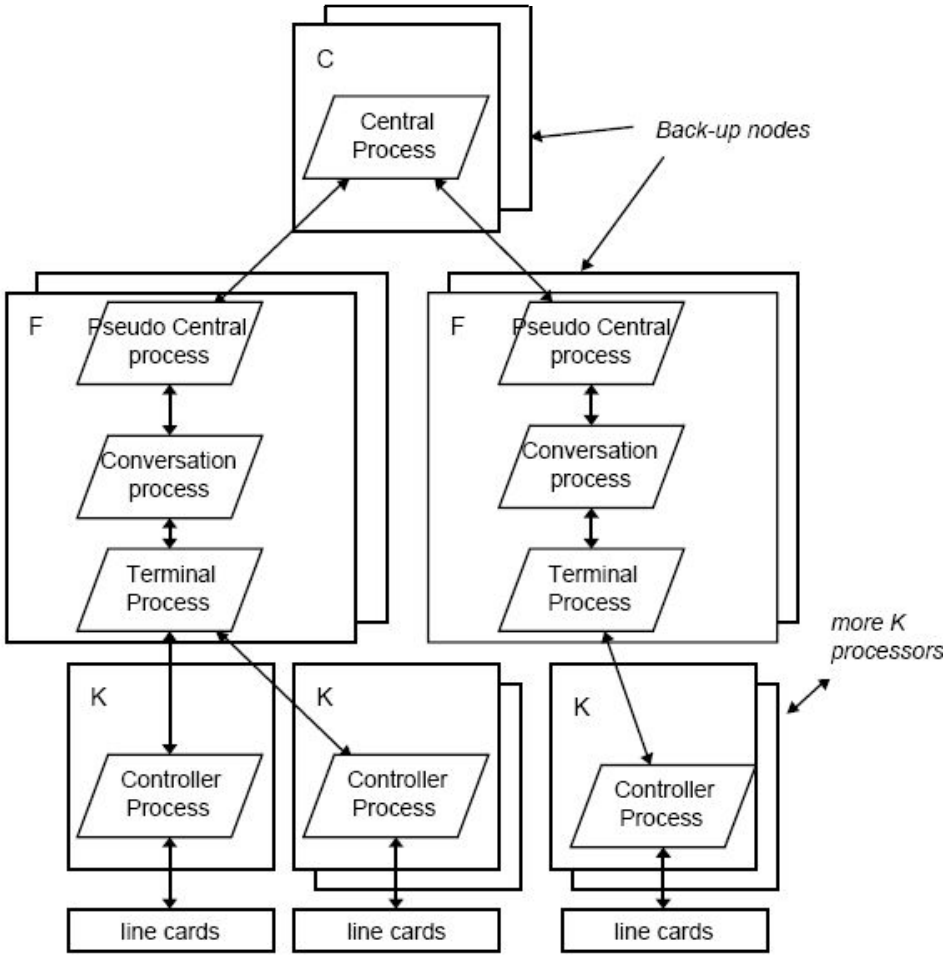
Notation: May have several forms and may Tightly connected to the process view

- There may be two architecture:
  - Test and development
  - deployment

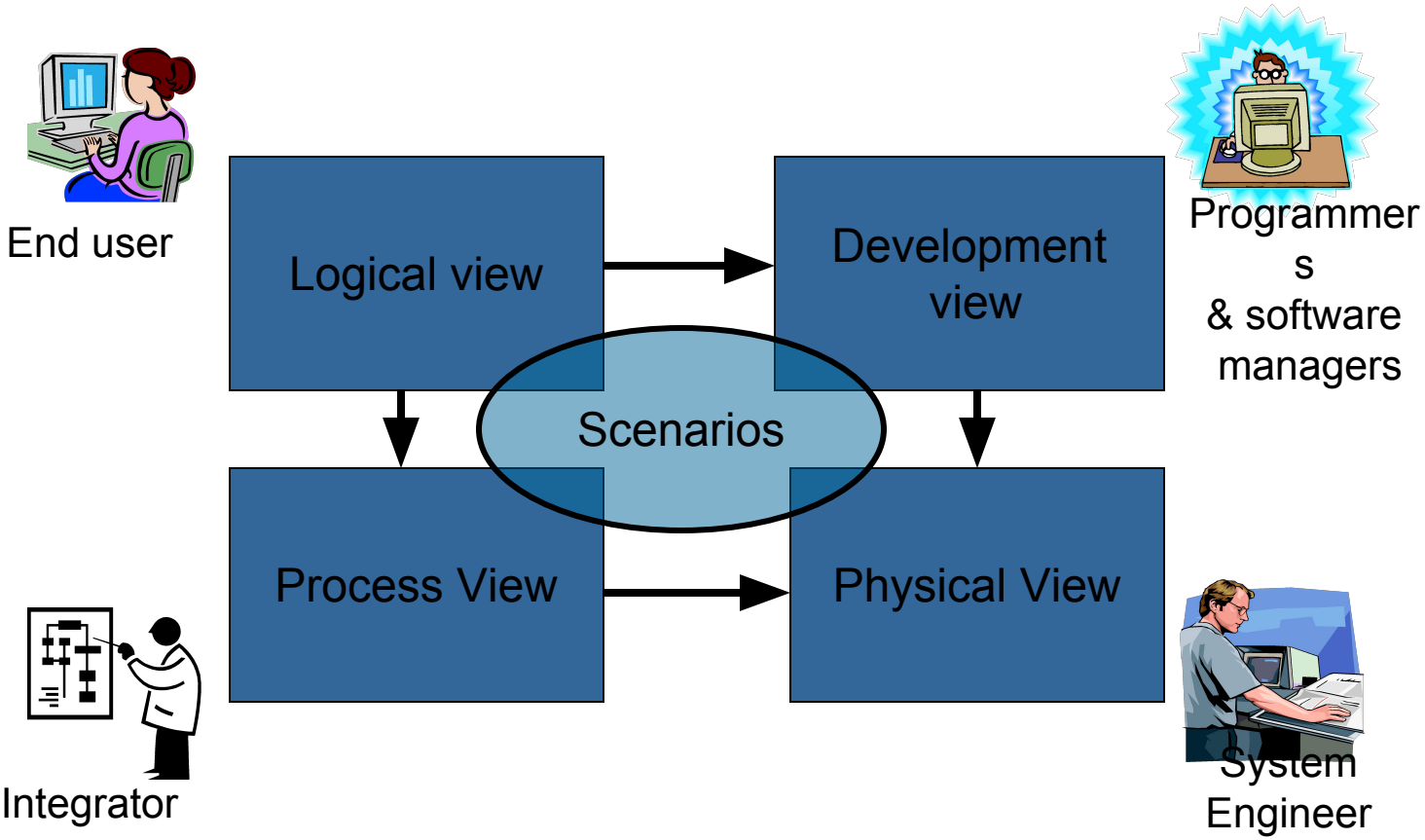
Physical view example



Physical view example (cont.)



4+1 View Model of Architecture



Scenarios

(Putting it all together)

Viewer: All users of other views and Evaluators.

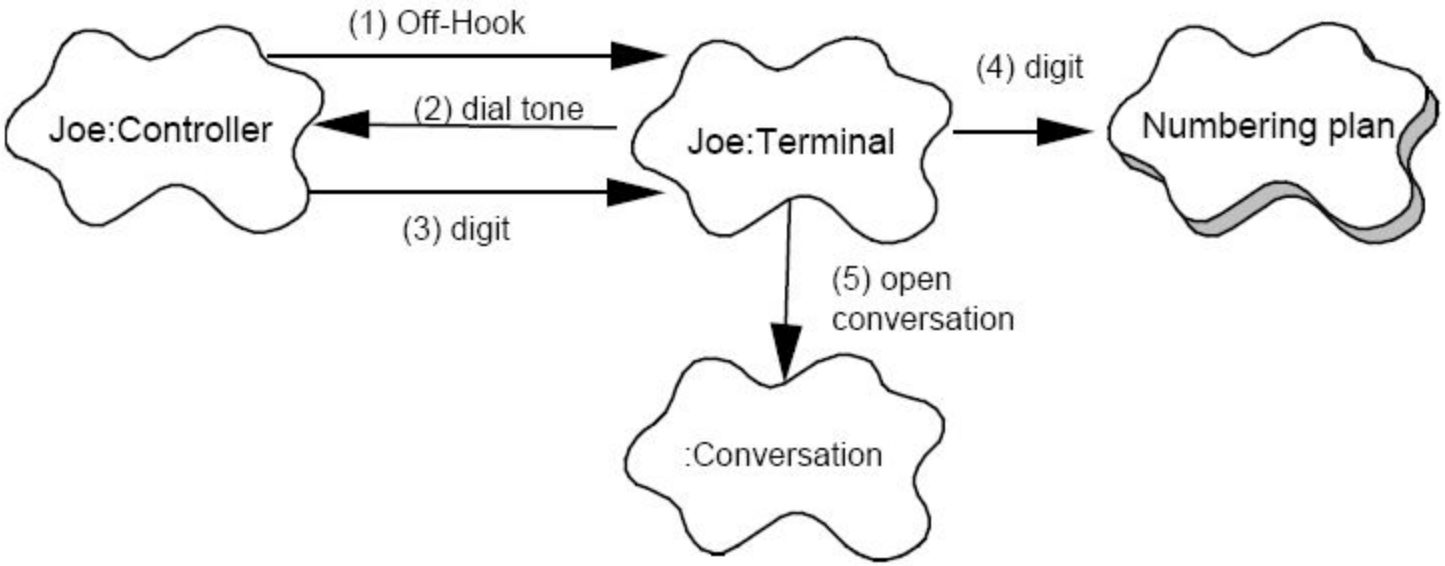
Considers: System consistency, validity

Notation: almost similar to logical view

Tool: Rational Rose

- Help illustrate and validate the document
- Help Architect during the architecture design

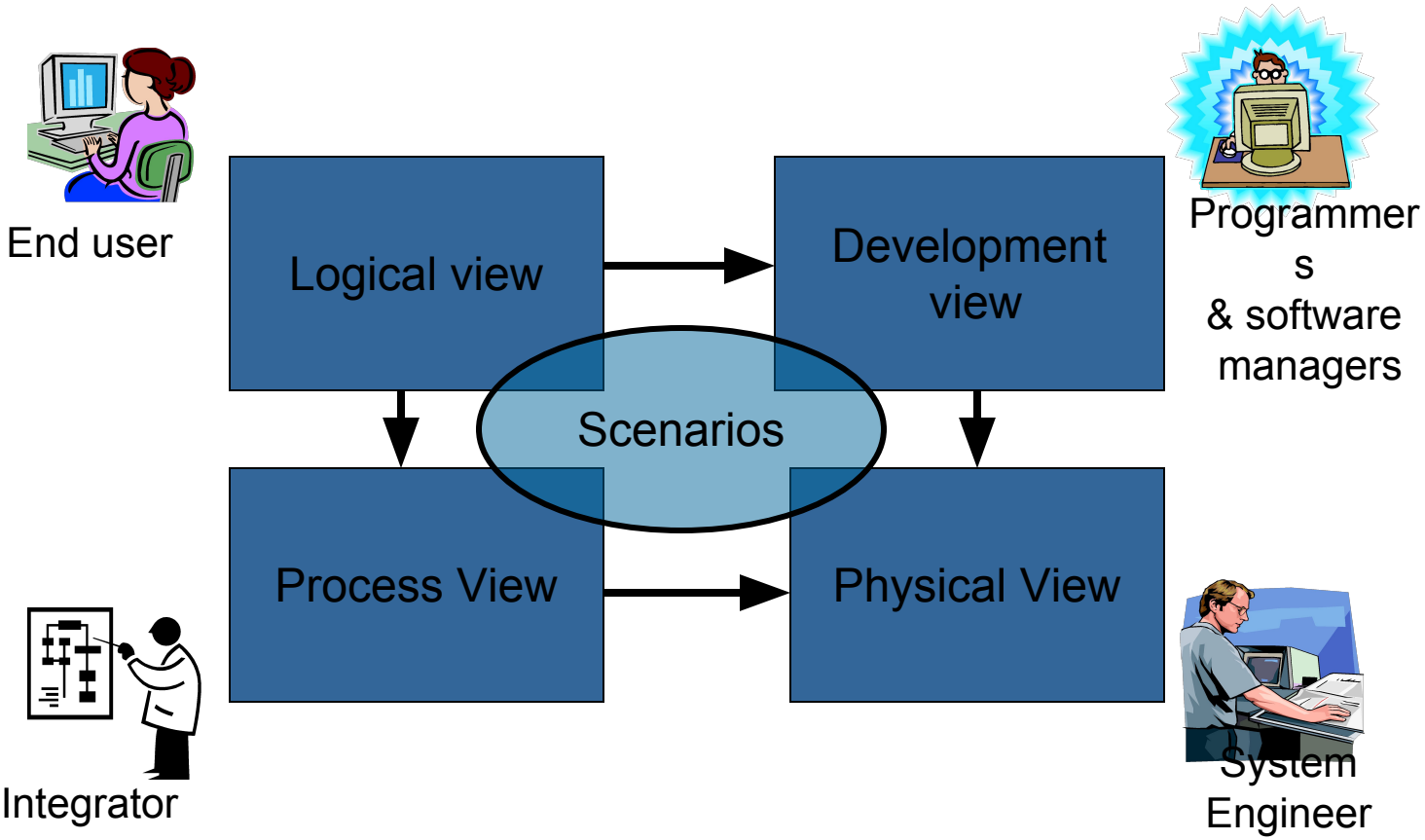
Scenario example



Correspondence between views

- Views are interconnected.
- Start with Logical view (Req. Doc) and Move to Development or Process view and then finally go to Physical view.

4+1 View Model of Architecture



From logical to Process view

- Two strategy to analyse level of ***concurrency***:
  - Inside-out: starting from Logical structure
  - Outside-in: starting from physical structure

From Logical to development

- They are very close, but the larger the project, the greater the distance
- Grouping to subsystems based on:
  - Classes
  - Class packages
  - Line of codes
  - Team organization

The Iterative process

- Not all software arch. Need all views.
- A scenario-driven approach to develop the system
- Documentation:
  - Software architecture document
  - Software design guidelines