Architecture - modeling

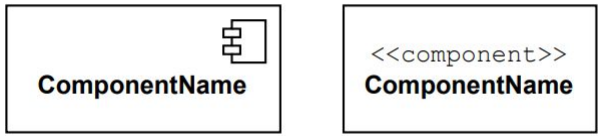
Jana Kostičová

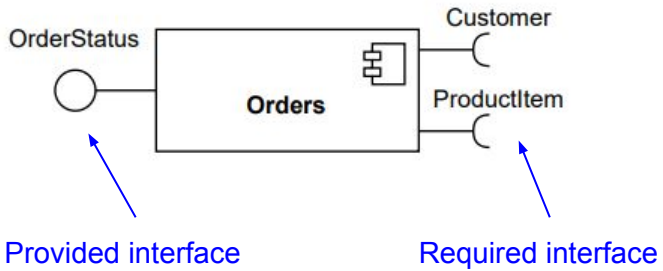
Architecture modeling - examples

- Components
  - UML Component diagrams
- Deployment
  - Deployment Diagrams Overview
  - UML Deployment Diagrams Examples
- Network architecture
  - Network Architecture Diagrams
  - AWS Cloud architecture for web hosting
  - 13 sample architectures to kickstart your Google Cloud journey

UML Component Diagram

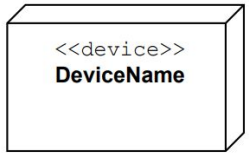
- **Component**
  - = a modular unit with well-defined interfaces [1]
  - The concept of components supports component-based architecture
  - **Notation:** Specialized class (may be nested)
- Provided and required interfaces
  - Notation: ball and socket
  - Ball-and-socket join vs dependency
- Ports
  - = interaction points
- Used to model either logical or physical structure of a system

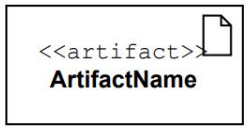




UML Deployment Diagram

- **Device**
  - = a physical computational resource with processing capability upon which artifacts may be deployed for execution
  - **Notation:** Node
- **Execution environment**
  - **Notation:** Node
- **Artifact**
  - = the specification of a physical piece of information that is used or produced by a software development process, or by deployment and operation of a system
  - **Notation:** A specialized classifier
  - **Examples:** Text document, source file, script, binary executable file, library,...





We need

1. To map artifacts to deployment targets (nodes)
2. To map components to artifacts
   - = manifestation of components by artifacts
   - Modeled by <<manifest>> dependencies / abstractions

   (Abstraction = a dependency that relates two entities that represent the same concept at different levels of abstraction)

Example mapping

Logical component	Physical component	Physical artifact	Deployment target
Orders	Orders Service	OrdersService.jar (business logic)	Application Server
Orders	Orders Service	OrdersAPI.jar (API Gateway)	Application Server
Orders	Orders static page	orders.html	Web Server
Orders	Orders table	orders table (database table)	Database Server

Network Architecture

- UML deployment diagrams could be used
  - Custom networking stereotypes
  - Issues: May not provide enough detail when representing network-level components
- Often custom format
- Some common network-level components:
  - Routers, Switches, Gateways
    - Manage the traffic flow between networks and devices
  - Firewalls and Proxies
    - Secure the network by controlling access and monitoring traffic
  - Load Balancers and CDNs
    - Optimize application delivery by distributing traffic and caching content
  - DNS servers
- Modern cloud architectures
  - Automatic scaling, elastic load balancing, CDNs, managed databases and storage solutions, security, …

→ Strong focus on non-functional requirements (scalability, data backup, data durability, response time, security, …)

References

1. OMG® Unified Modeling Language® (OMG UML®), Version 2.5.1, 2017
2. R. Červenka: UML Components
3. R. Červenka: UML Deployment
4. The component diagram (IBM)

\+ Examples (1st slide)