M1 L4
Service-Oriented Architecture and Concepts

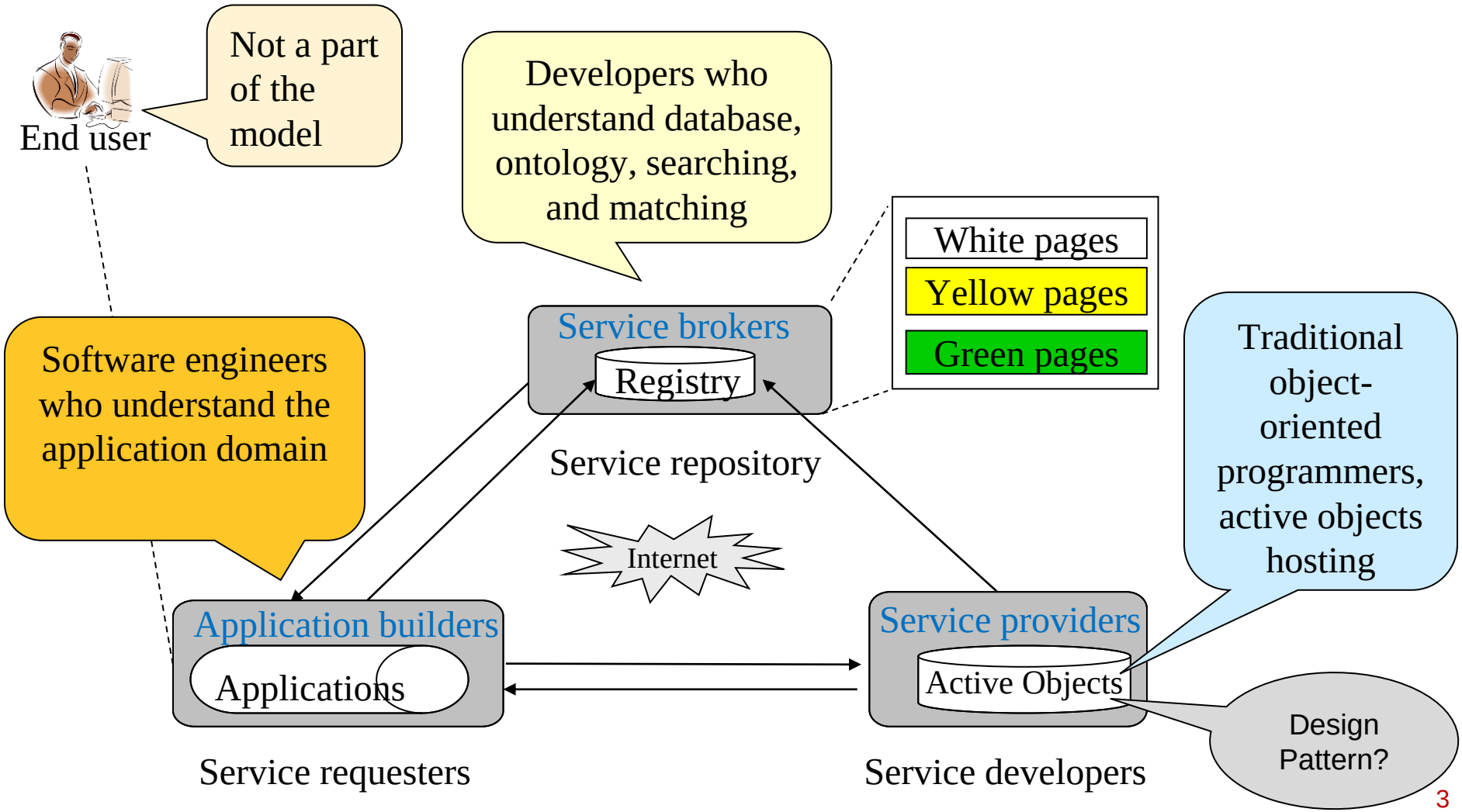
Ira A. Fulton Schools of Engineering
Arizona State University

Lecture Overview

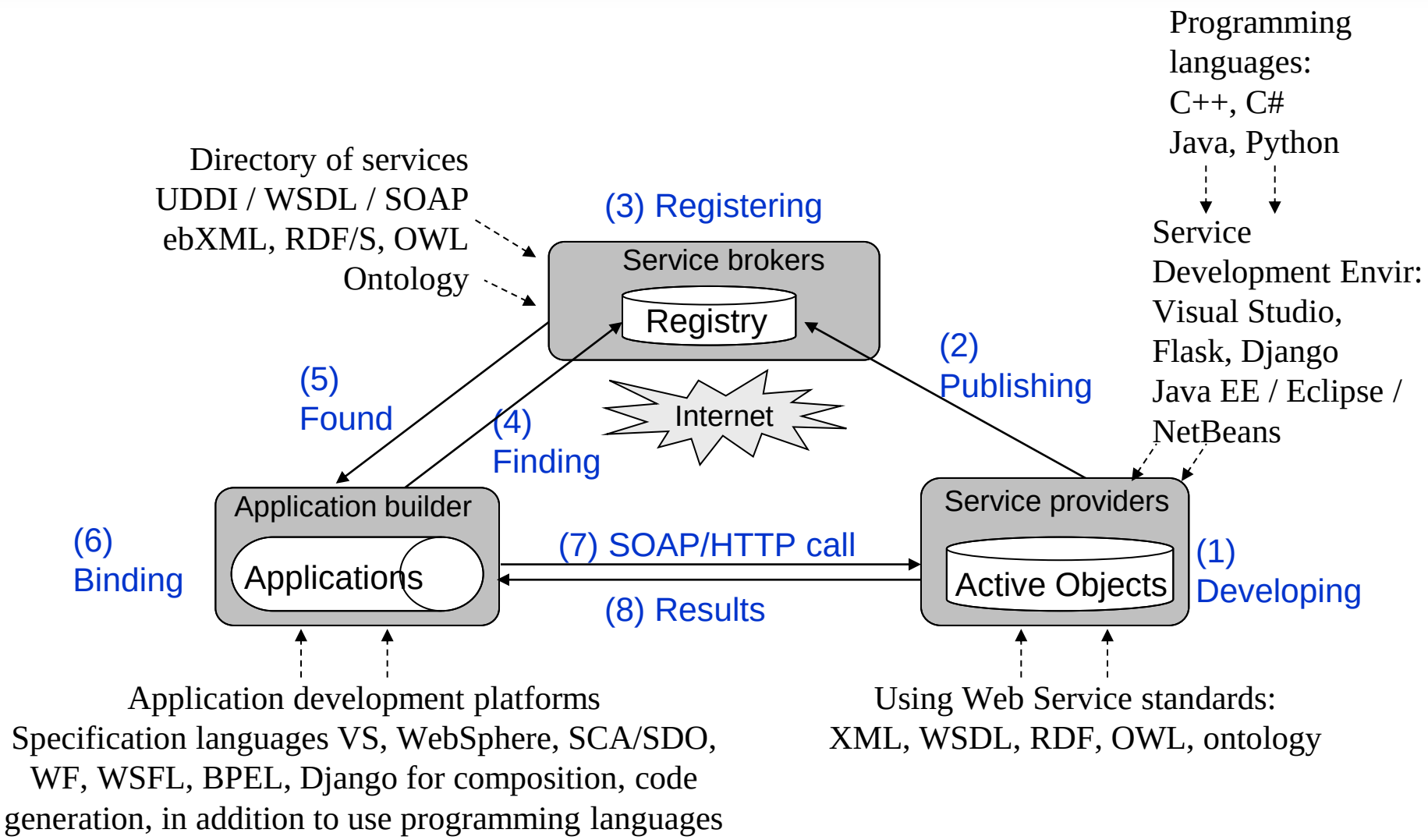
- **Service-Oriented Architecture and Concepts**
  - Three-Party Model of Service-Oriented computing
  - SOA, SOC, SOD, Web services
  - Development Cycle

Distributed Development: Separation of Responsibility

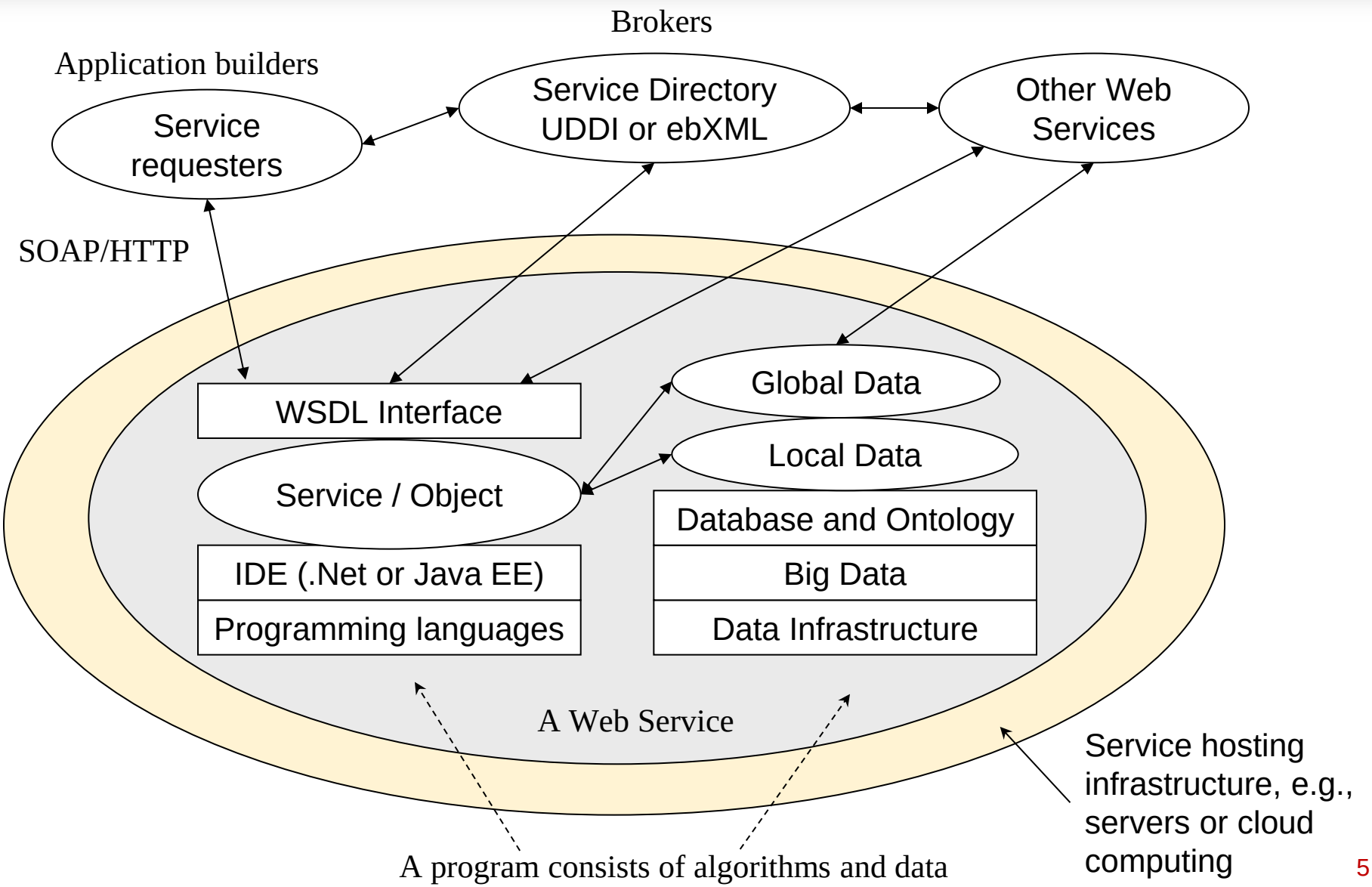
The Three-Party Model of Service-Oriented Software Development



A Scenario of the SO Software Development and Tools



Web Services in Web Applications



Definitions and Terminologies

- A service is the interface between the producer and the consumer.
- From the service provider's point of view, a service is a function module that is well-defined, self-contained, and does not depend on the context or state of other functions. A service is often implemented by an active object.
  - Services can be newly developed modules or just wrapped around existing (legacy) software to give them new interfaces.
- From the application builder's point of view, a service is a unit of work done by a service provider to achieve desired end results for a service consumer (an application builder, not an end user).
- A service normally does not have the human user's interface. Instead, it provides loosely coupled Application Programming Interface (API), with standard interface, so that a service can be discovered and invoked by a computer program.
- For human users (end users) to use services, a (graphic) user interface needs to be added – forming a (Web) application.

Definitions and Terminologies (contd.)

- Service-Oriented Architecture (SOA): Software consisting of a collection of loosely coupled and platform-independent services that communicate with each other through standard interfaces. SOA does not concern developing operational software.

Architecture Pattern

- Service-Oriented Computing (SOC) refers to the paradigm that represents computation in SOA. A level deeper than SOA, incl. algorithms and data structures.

Computing Pattern

- SOA and SOC are often used interchangeably.
- Web Services are services accessible over the Web. As an architecture, it refers to Web-based SOA and a set of enabling Web technologies, including XML, JSON, SOAP, WSDL, HTTP, UDDI, and ebXML. Two types of Web services:
  - SOAP/WSDL Services (heavy duty services)
  - RESTful Services (lightweight or micro services)

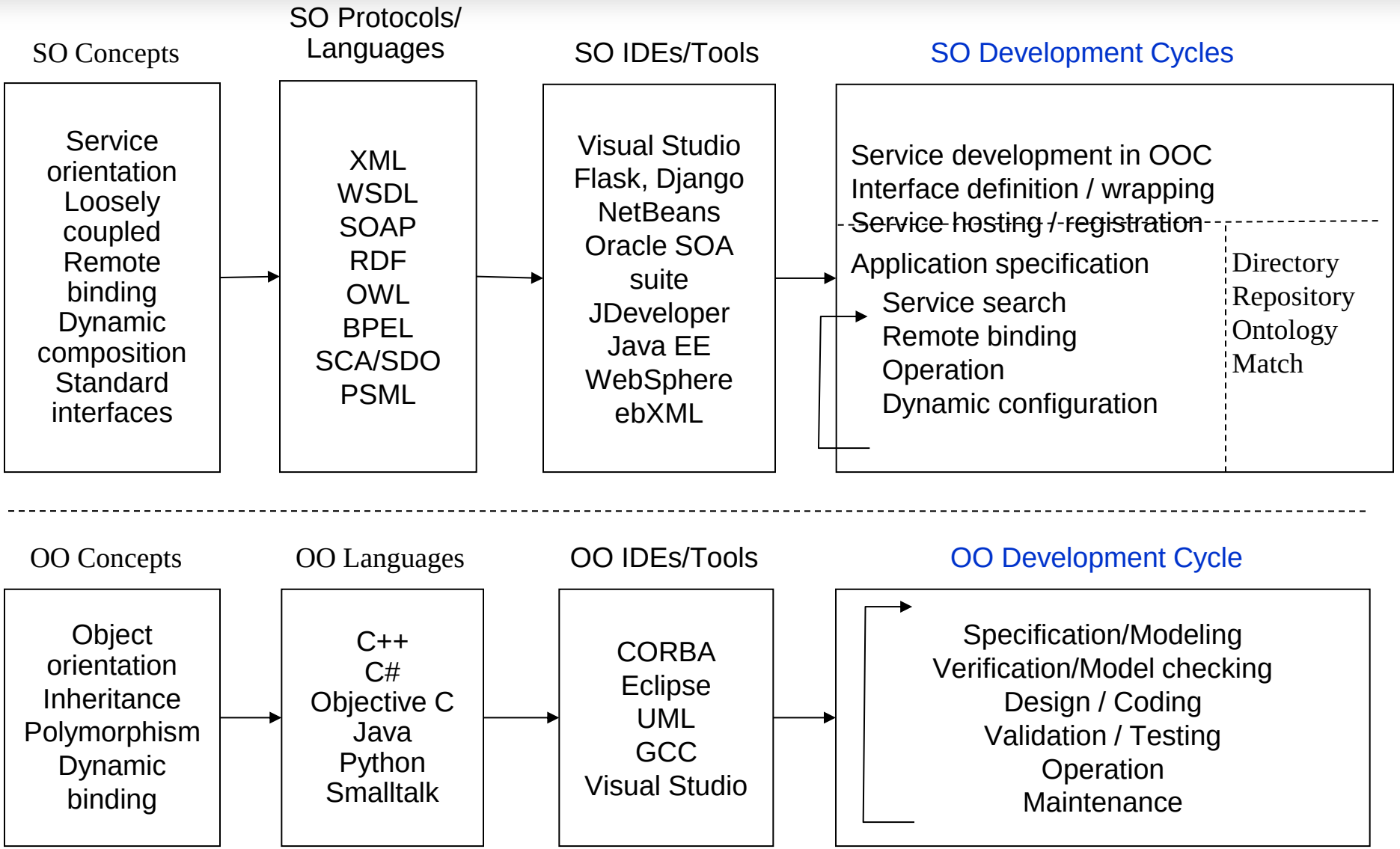
Definitions and Terminologies (contd.)

This course studies SOD

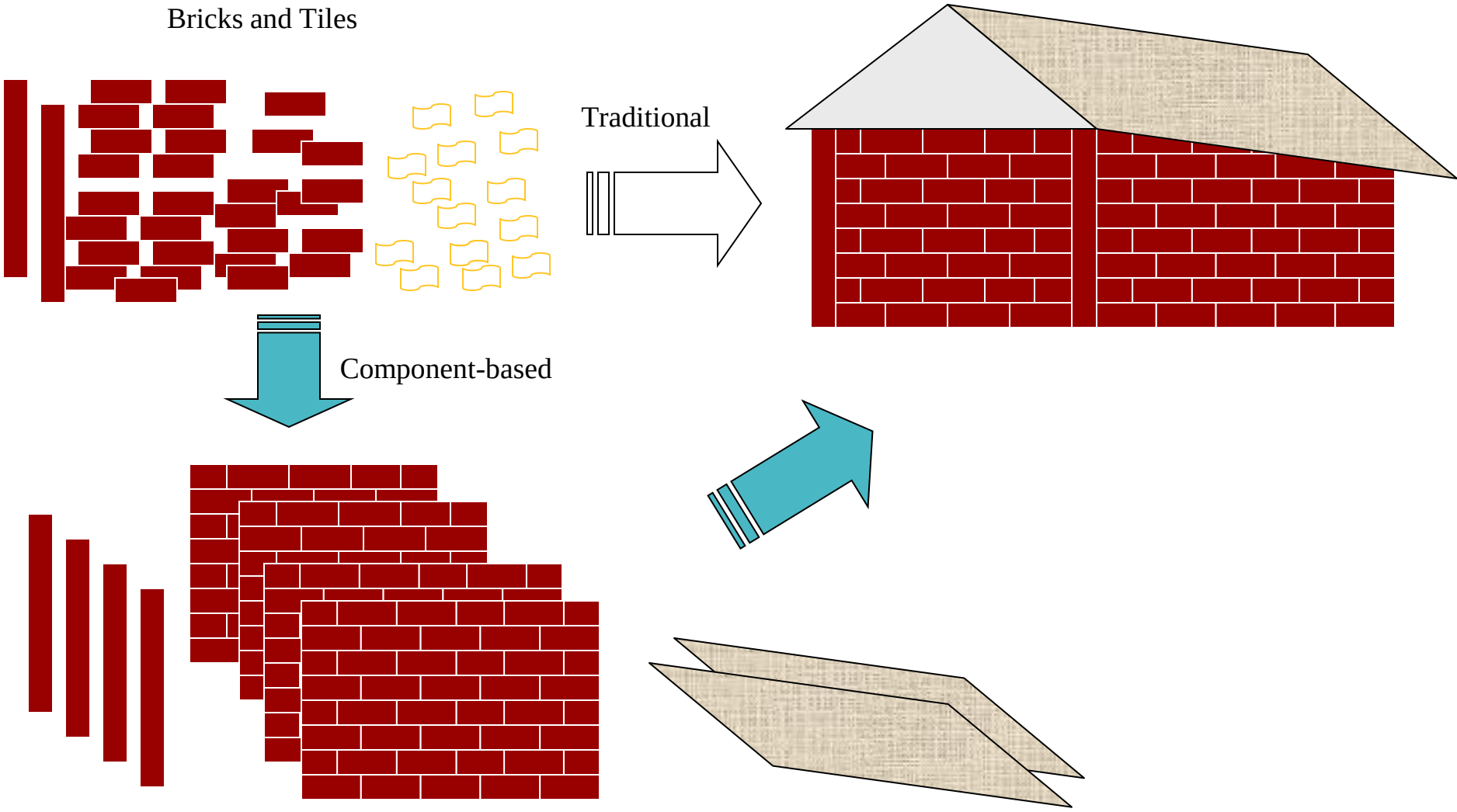
Service-Oriented Development (SOD)

- concerns the entire software **development cycle** based on SOA concepts and SOC paradigm.
- involves current technologies and tools to effectively produce operational software:
  - Development environments: VS, Java EE, Flask, Django, NetBeans, WebSphere, etc.;
  - Databases, SQL, Oracle, BD, XML DB, etc.;
  - Deployment and hosting environment: Server, cloud: SoD, AWS Cloud, Google Cloud, MS Azure, etc.

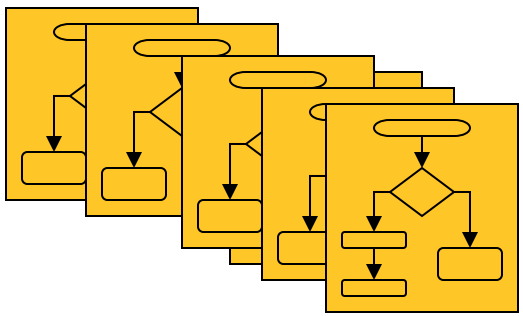
OO Development versus SO Development



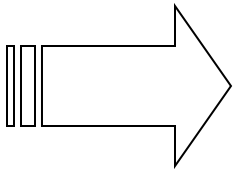
Component-Based Development

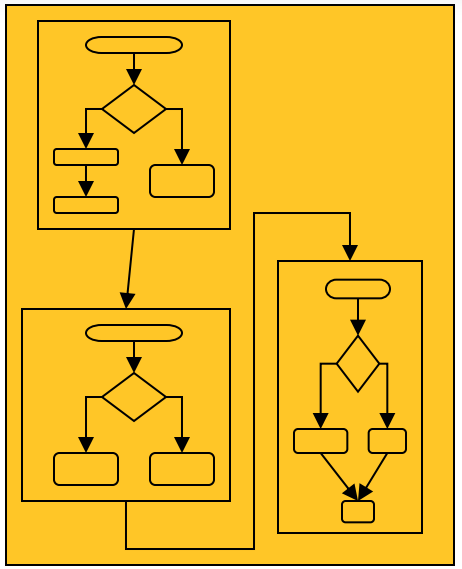


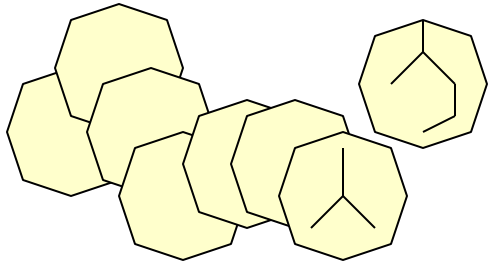
Component-Based Software Development



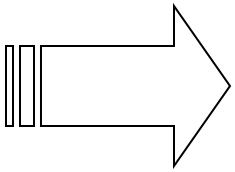
Organization X:
Component library

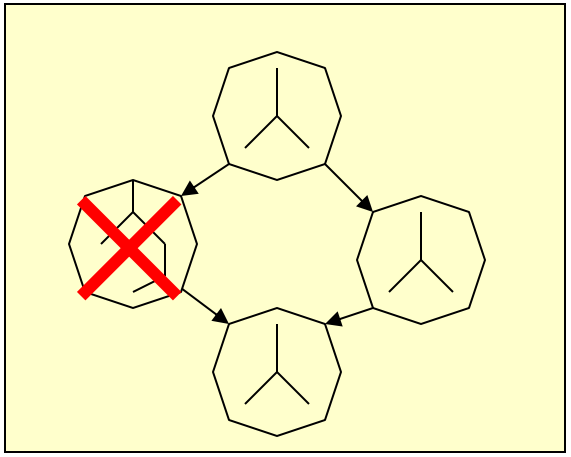




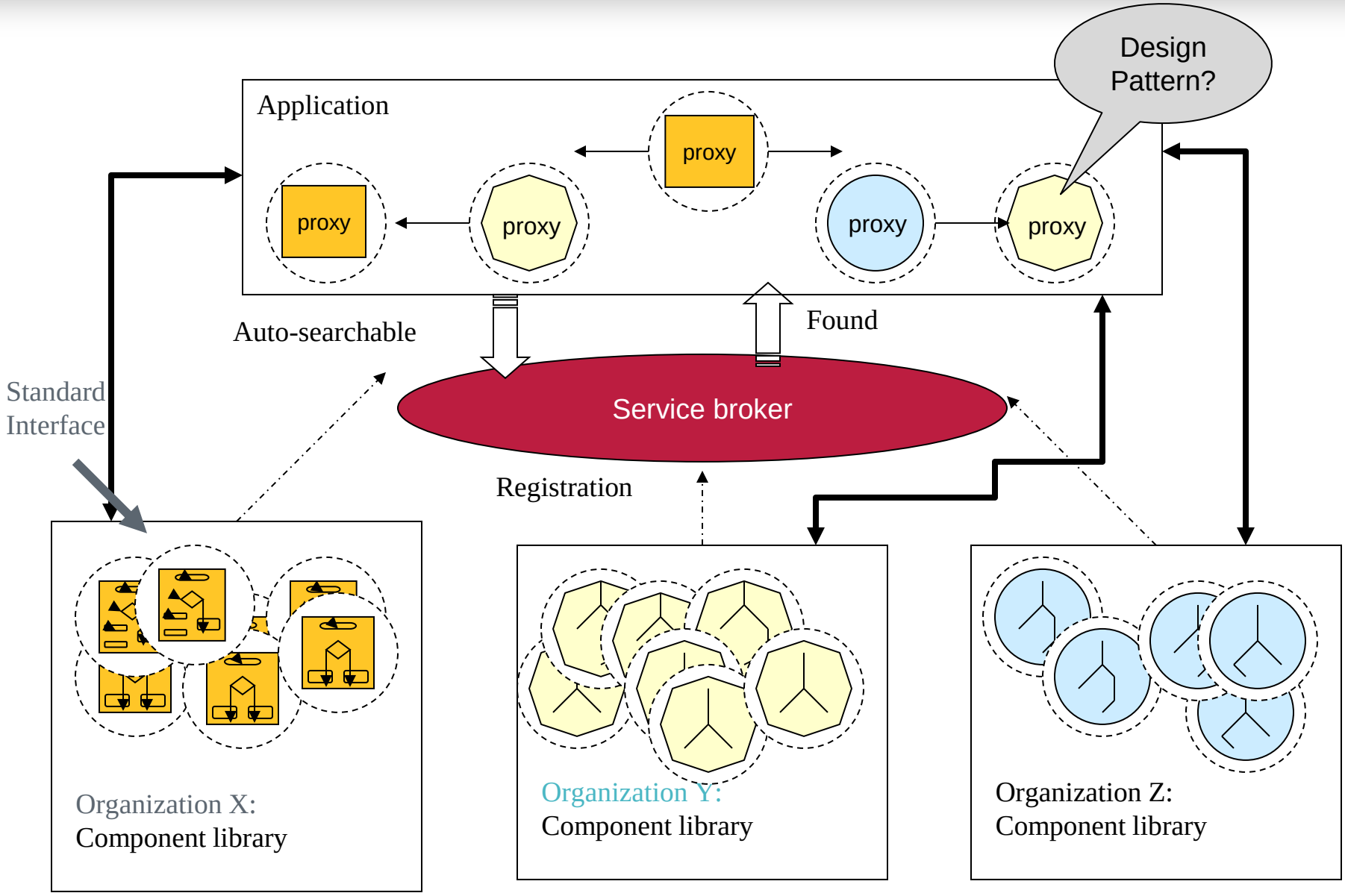


Organization Y:
Component library





Service-Oriented Software Development



Summery: Key SOA, SOC & SOD Concepts

- Component-based composition with open standards and protocols
- Remote objects, binding, and remote invocation
- Autonomous services with platform-independency: Using proxy, instead of code integration
- Loosely coupling using message-based communication and asynchronous communication is supported
- Web-based repository of internet-searchable and reusable services
- Automatic discovering and binding, based on collaboration negotiation, dynamic re-composition, and ontology-based reasoning.
- Separation of development: service provider, broker, client


Ira A. Fulton Schools of
Engineering
Arizona State University