

Service-Oriented Architecture

Definitions – Discussions

Dirk Draheim
draheim@acm.org

Service

At the core of an SOA is a service. A service is a coarse-grained, discoverable, and self-contained software entity that interacts with applications and other services through a loosely coupled, often asynchronous, message-based communication model.

Brown, Johnston, Kelly. Using Service-Oriented Architecture and Component-Based Development to Build Web Service Applications. Santa Clara, CA: Rational Software Corporation, 2002.

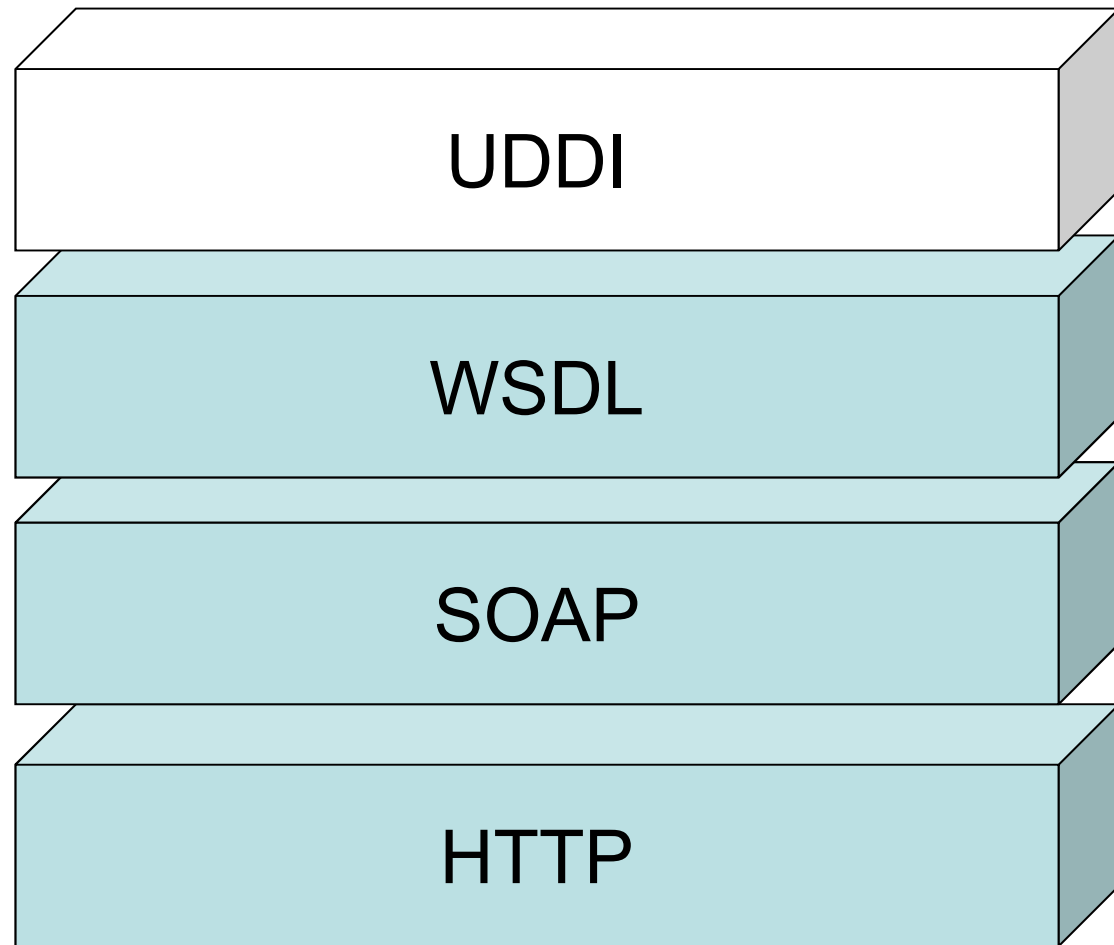
SOA

An SOA is a collection of services with well-defined interfaces and a shared communications model. A system or application is designed and implemented to make use of these services. This developed capability may itself provide services within the overall SOA. [There are...] several options for incorporating services:

- 1. Service interfaces are added to existing enterprise information systems for applications to use, while these systems remain unchanged for internal users.*
- 2. Service-specific code is written to provide functionality for applications to use.*
- 3. Services written by third parties and deployed elsewhere are used within applications.*

Lewis, Morris, O'Brien, Smith, SMART: The Service-Oriented Migration and Reuse Technique, Technical Note CMU/SEI-2005-TN-029, SEI, September 2005

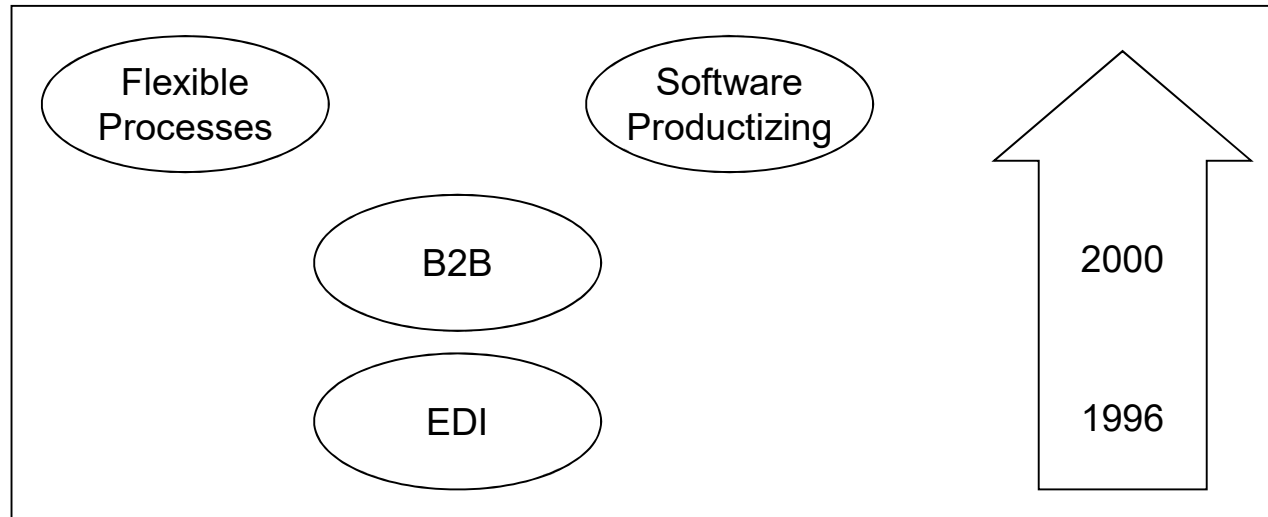
Web Services



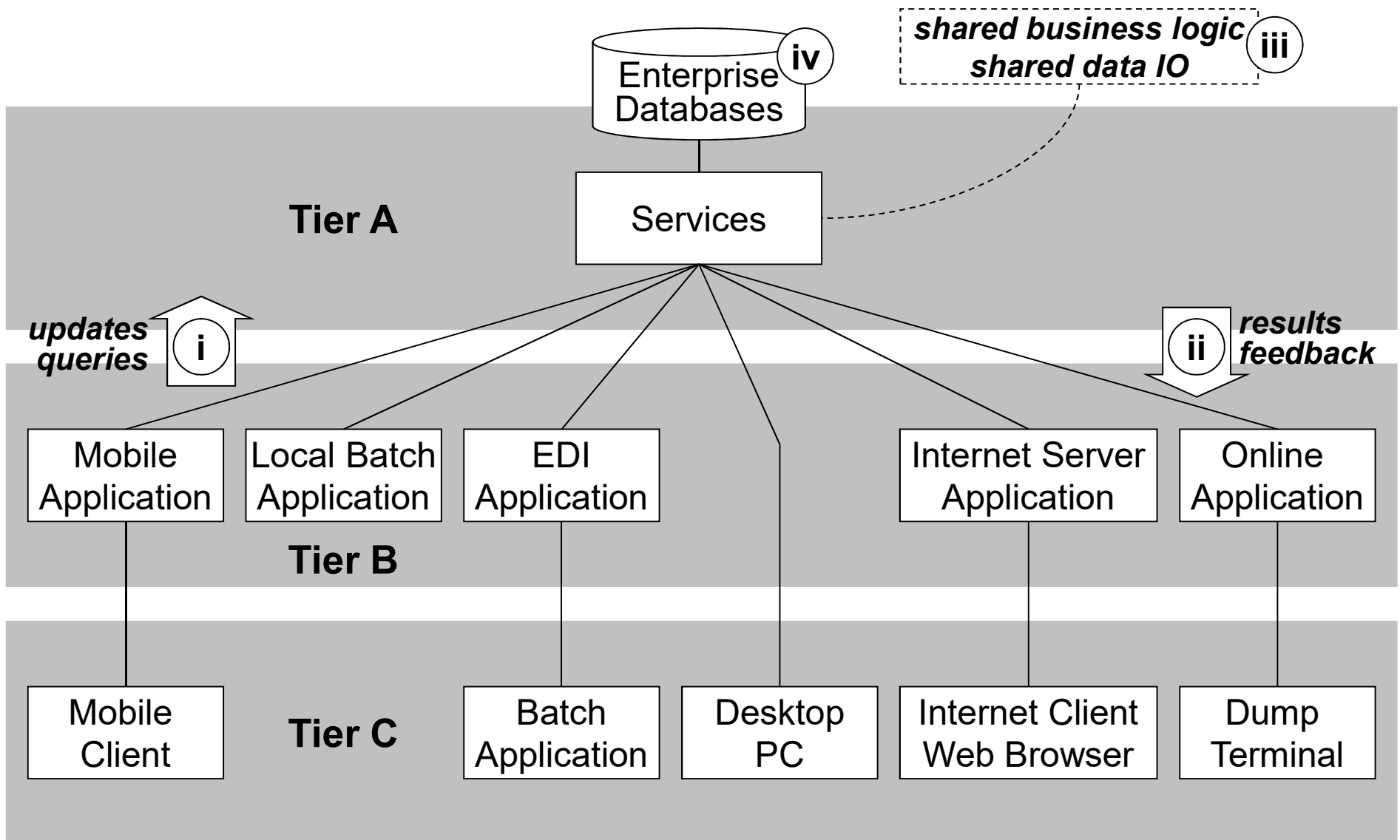
Enterprise Application Integration

	<i>Hub and Spoke Architecture</i>	<i>Distributed Architecture</i>
<i>Loose Coupling</i>	DB	Enterprise Service Bus
<i>Tight Coupling</i>	Application Server	Message- Oriented Middleware

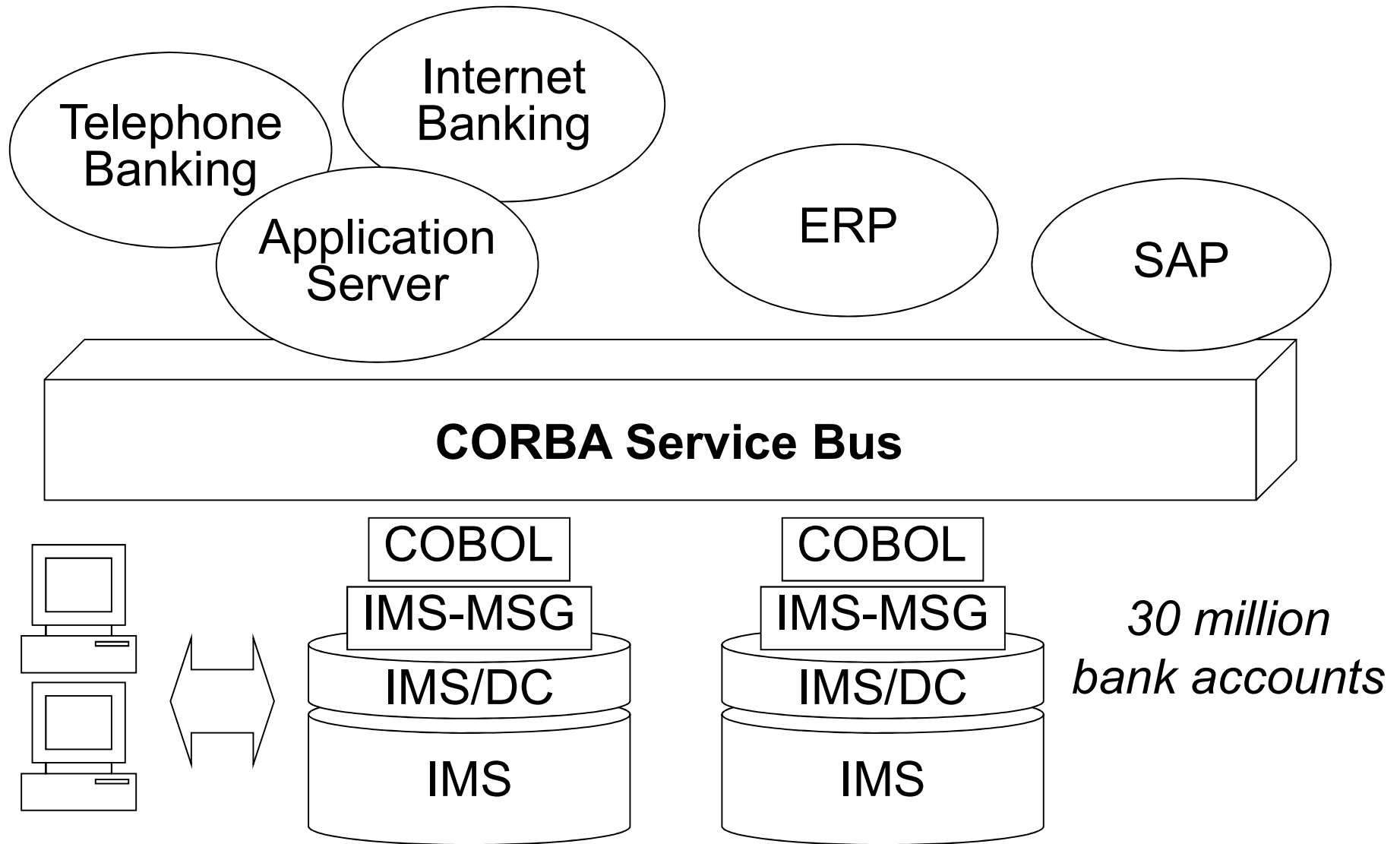
The Evolution of SOA Paradigms and Visions



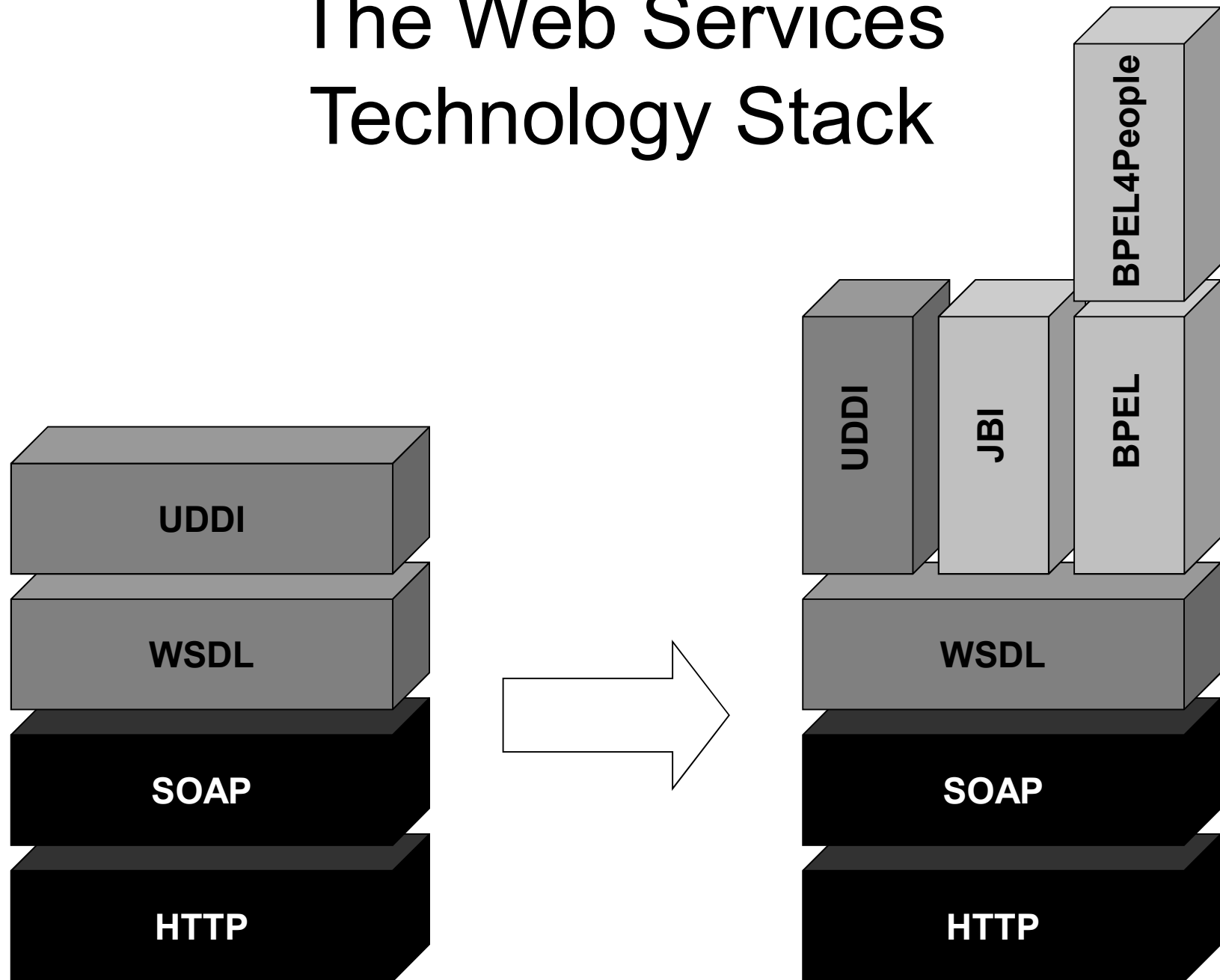
Gartner Group Tier Terminology for SOA



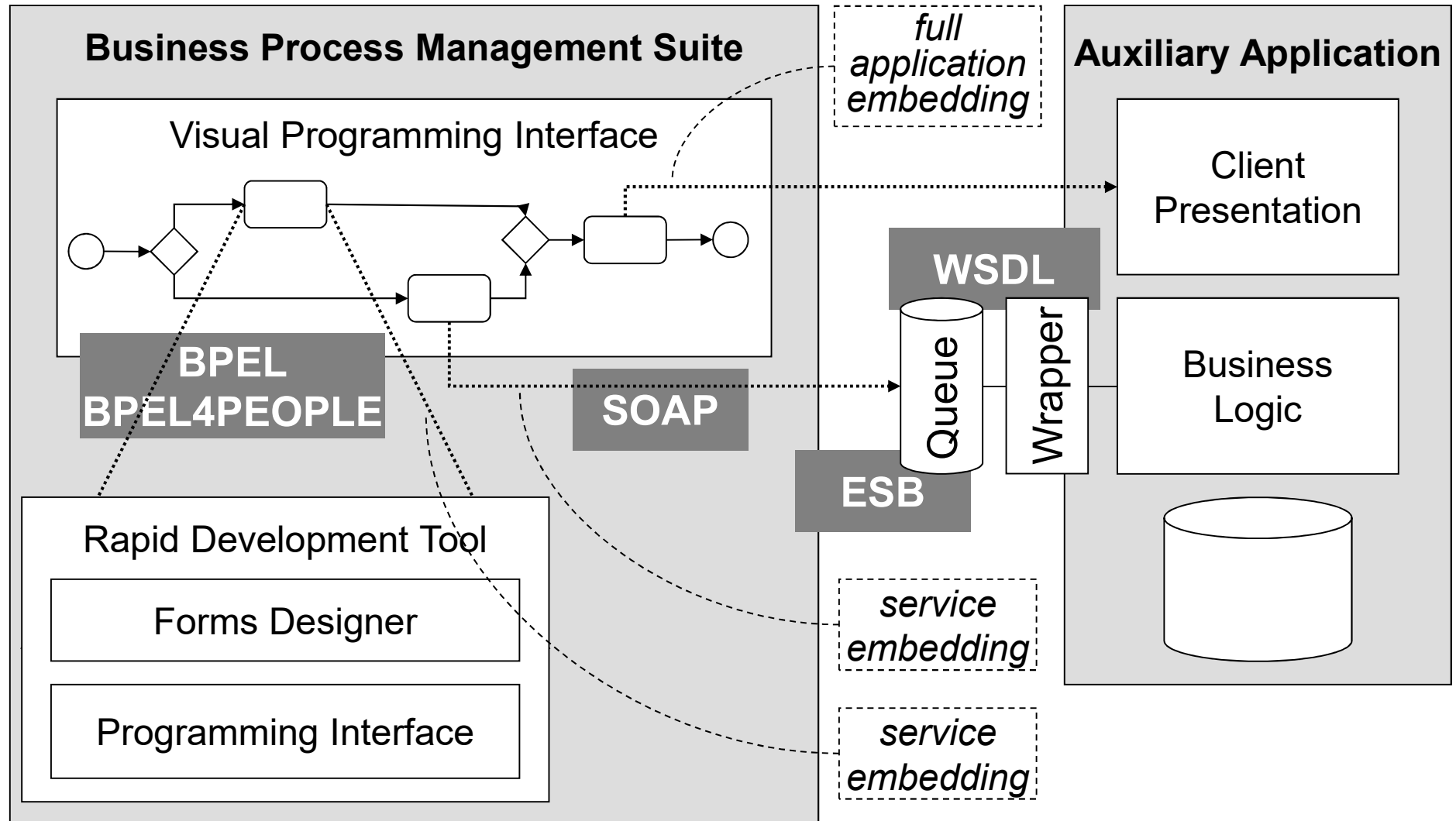
GAD: CORBA-Service-Bus



The Web Services Technology Stack



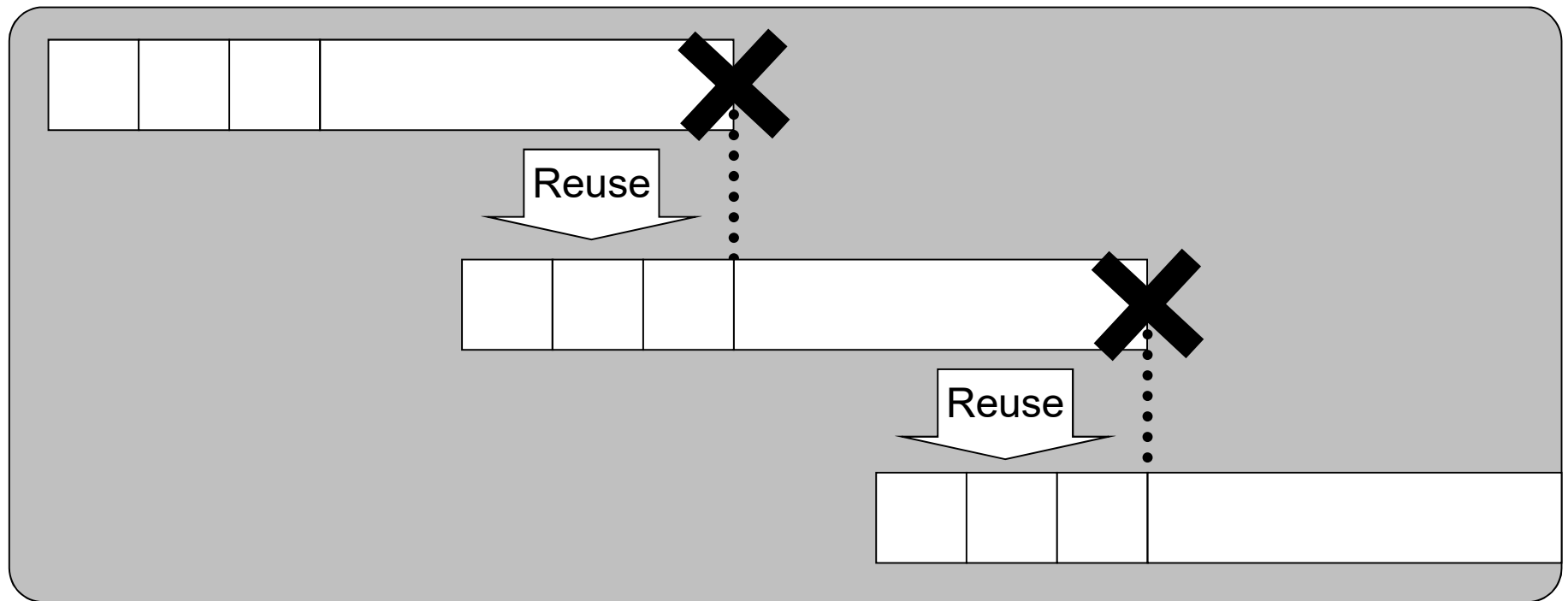
Web Services Technologies for Business Process Management



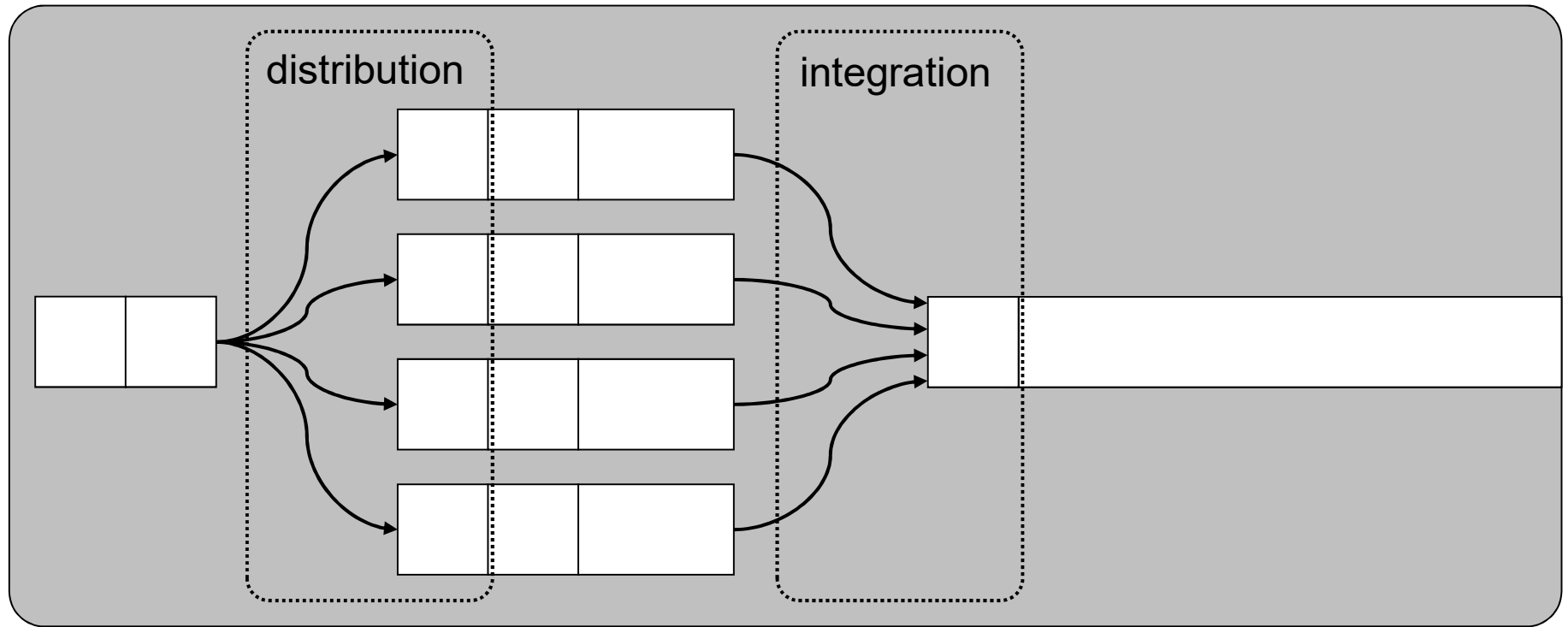
Silo Software System Development

Project A				
Project B				
Project C				

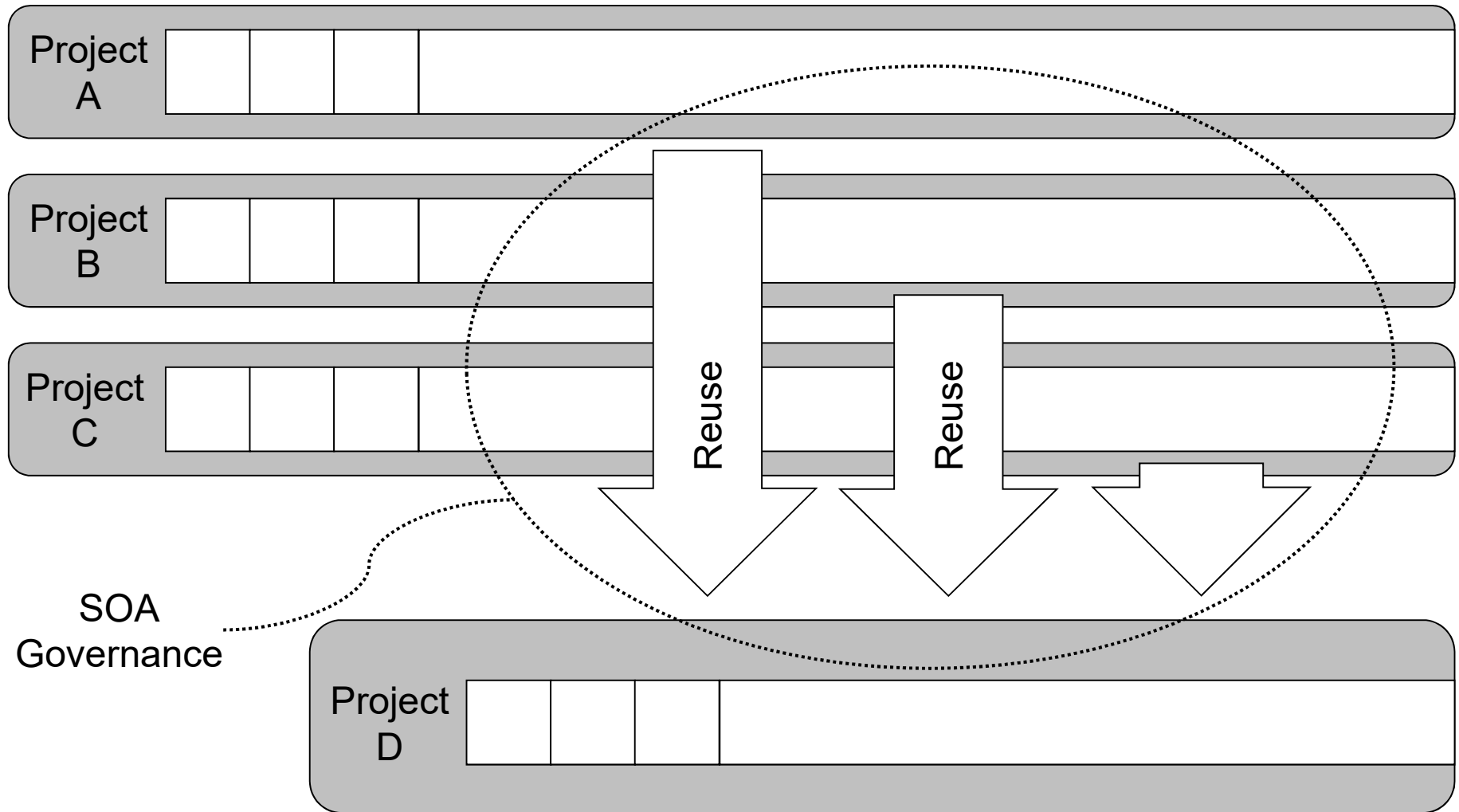
Iterative Development of a Silo Software System



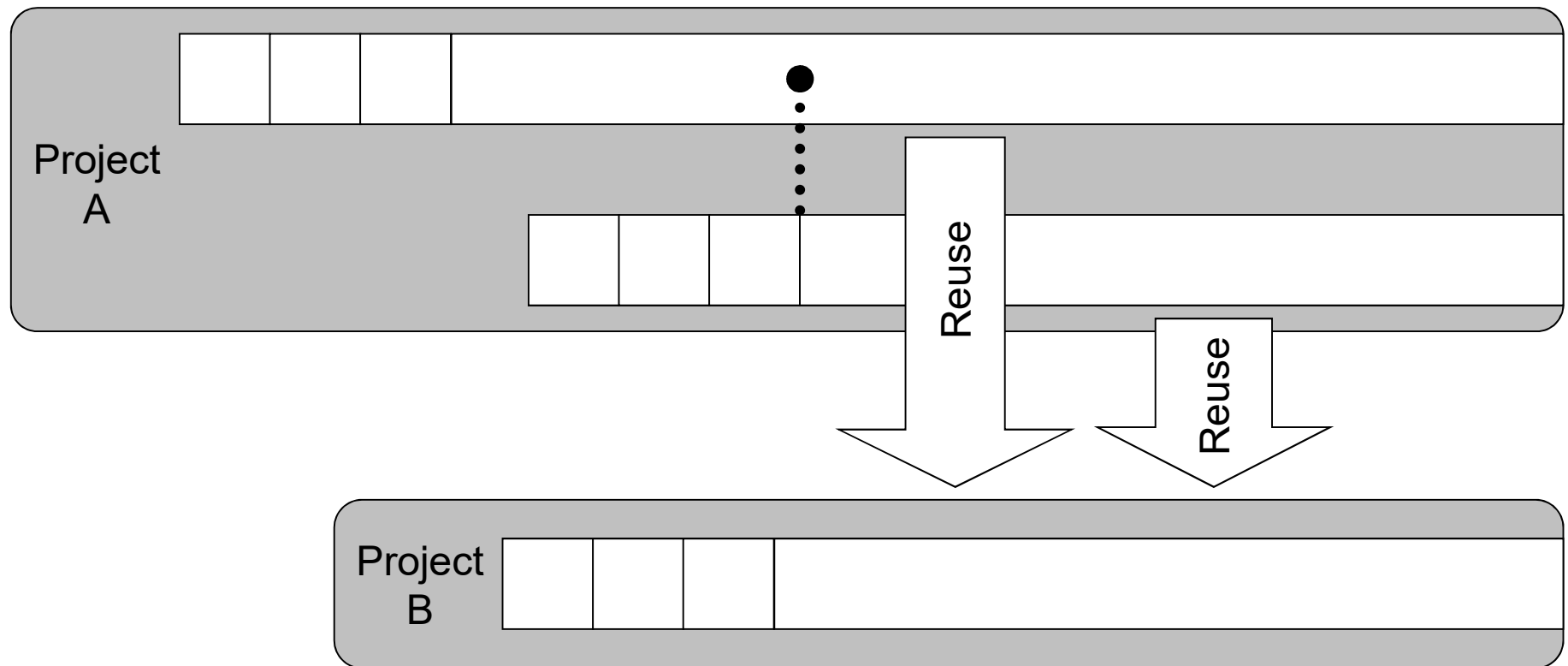
Division of a Project into Subprojects



Software Reuse across Project Boundaries



Software Reuse from a Maintained Software Product



SOA Governance as Ubiquitous Reuse

