



# Session 1: Introduction to SOA



# หัวข้อนำเสนอ

- การพัฒนาระบบซอฟต์แวร์แบบองค์รวม
- Service เบื้องต้น
- Architecture เบื้องต้น
- SOA เบื้องต้น
- สรุป



# ลองค้นคำว่า "SOA"

**SOA - ค้นหาคำใน Google - Mozilla Firefox**  
http://www.google.co.th/search?hl=th&q=SOA&btnG=%E0%68%84%E0%B9%85  
SOA  
ผลการค้นหา 1 - 10 รายการจากประมาณ 48,300,000 สำหรับคำว่า SOA. (0.03 วินาที)  
Service-oriented architecture - Wikipedia, the free encyclopedia  
SOA can also be regarded as a style of information systems architecture that ... SOA  
Practitioners Guide: Why **Services-Oriented Architecture?** provides a ...  
en.wikipedia.org/wiki/Service-oriented\_architecture - 77k - [หน้าหลักเกี่ยวกับ](#) - [หน้าที่คล้ายกัน](#)  
SOA  
An educational, research and Canada who primarily  
www.soa.org/ - 20k - 8 s  
webservices.xml.com  
Service-Oriented Archi  
loose coupling between i  
www.xml.com/pub/a/ws/  
SOA Watch  
Lead organization workin  
www.soaw.org/ - 71k - 11

**"Web Services" - Google Search - Mozilla Firefox**  
http://www.google.co.th/search?hl=en&q=%22Web+Services%22&btnG=Google+Search  
"Web Services"  
Search: the web pages from Thailand  
Results 1 - 10 of about 275,000,000 for "Web Services". (0.29 seconds)  
Web Services  
W3C's overview of news, groups, documents and specifications, events and recommended reading.  
www.w3.org/2002/ws/ - 30k - [Cached](#) - [Similar pages](#)  
Web Services Description Language (WSDL)  
The W3C note is a specification for describing network services as a set of endpoints operating on messages containing either document-oriented or ...  
www.w3.org/TR/wsdl - 131k - [Cached](#) - [Similar pages](#)  
Web service - Wikipedia, the free encyclopedia  
For example, WS-I only recognizes **Web services** in the context of these ... The first **Web services** tools were focused on RPC, and as a result this style is ...  
en.wikipedia.org/wiki/Web\_service - 28k - [Cached](#) - [Similar pages](#)  
Sponsored Links  
Buy and Sell **Web Services**  
Integrate multiple **Web services** into your Website. Free Trials  
www.strikeiron.com  
Oracle SOA Suite  
Comprehensive. Hot-Pluggable. Free download.  
www.Oracle.com/SOA  
OSS Fast **Web Services**  
Binary SOAP - High Performance.  
C/S, J2EE, Evaluations available.



# ข้อสงสัยเกี่ยวกับ SOA

- SOA คืออะไร?
- ทำไมต้องใช้ SOA?
- ใช้ SOA เมื่อไร?
- พัฒนา SOA อย่างไร?
- ไม่ใช่ SOA ได้ไหม?
- เริ่มต้นตอนนี้ เข้าไปหรือไม่?
- จะเปลี่ยนอีกไหมนี้?
- จะหาข้อมูลเกี่ยวกับ SOA ได้ที่ไหน?





# ประโยชน์ของ SOA

- SOA ได้รับการกล่าวถึงว่า
  - สามารถทำให้การพัฒนาระบบเป็นไปแบบองค์รวม (Integration) ด้วยค่าใช้จ่ายที่คุ้มค่า
  - สามารถทำให้ระบบซอฟต์แวร์เดิมที่มีให้บริการแก่ระบบซอฟต์แวร์อื่นๆ ได้ (Reusability)
  - สามารถปรับเปลี่ยน Implementation ของบริการได้โดยไม่กระทบกับผู้ใช้บริการอื่นๆ
  - สามารถทำให้ระบบซอฟต์แวร์เดิมแบบ Legacy นำกลับมาใช้ร่วมกันแบบองค์รวมได้ เกิดความคุ้มค่าในการลงทุน



# การพัฒนาาระบบซอฟต์แวร์แบบองค์รวม



# วิวัฒนาการของการพัฒนาระบบซอฟต์แวร์

- Structural
- Object Oriented
- Component Based
- Service Oriented



# ความสำเร็จในการพัฒนาระบบซอฟต์แวร์

- Build the proven assets (Components/Services)
- Reuse the proven assets
- Integrate the proven assets to obtain new solution



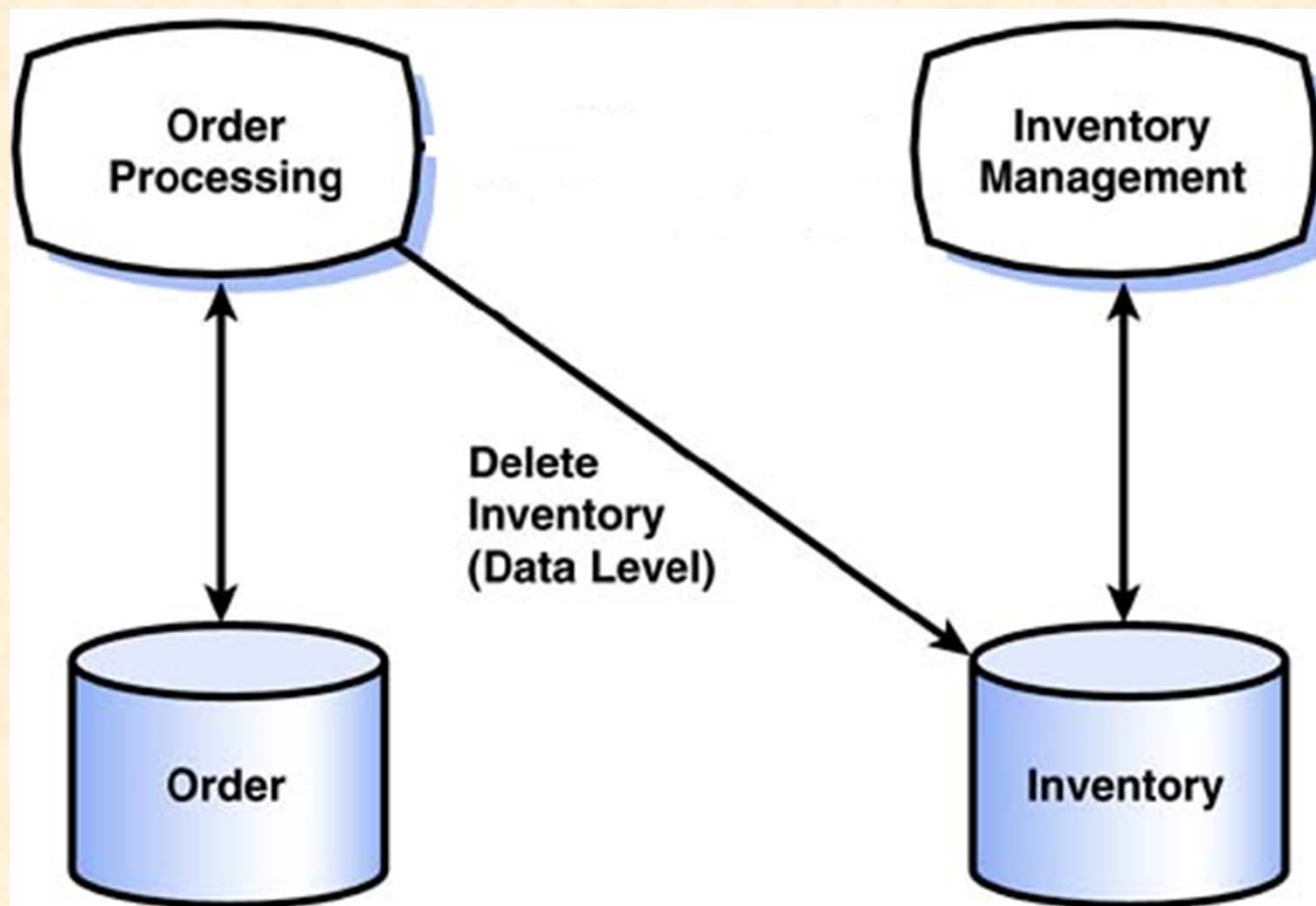


# การรวมระบบซอฟต์แวร์แบบองค์รวม

- การรวมแบบ Data-Level Integration
- การรวมแบบ Application-Level Integration

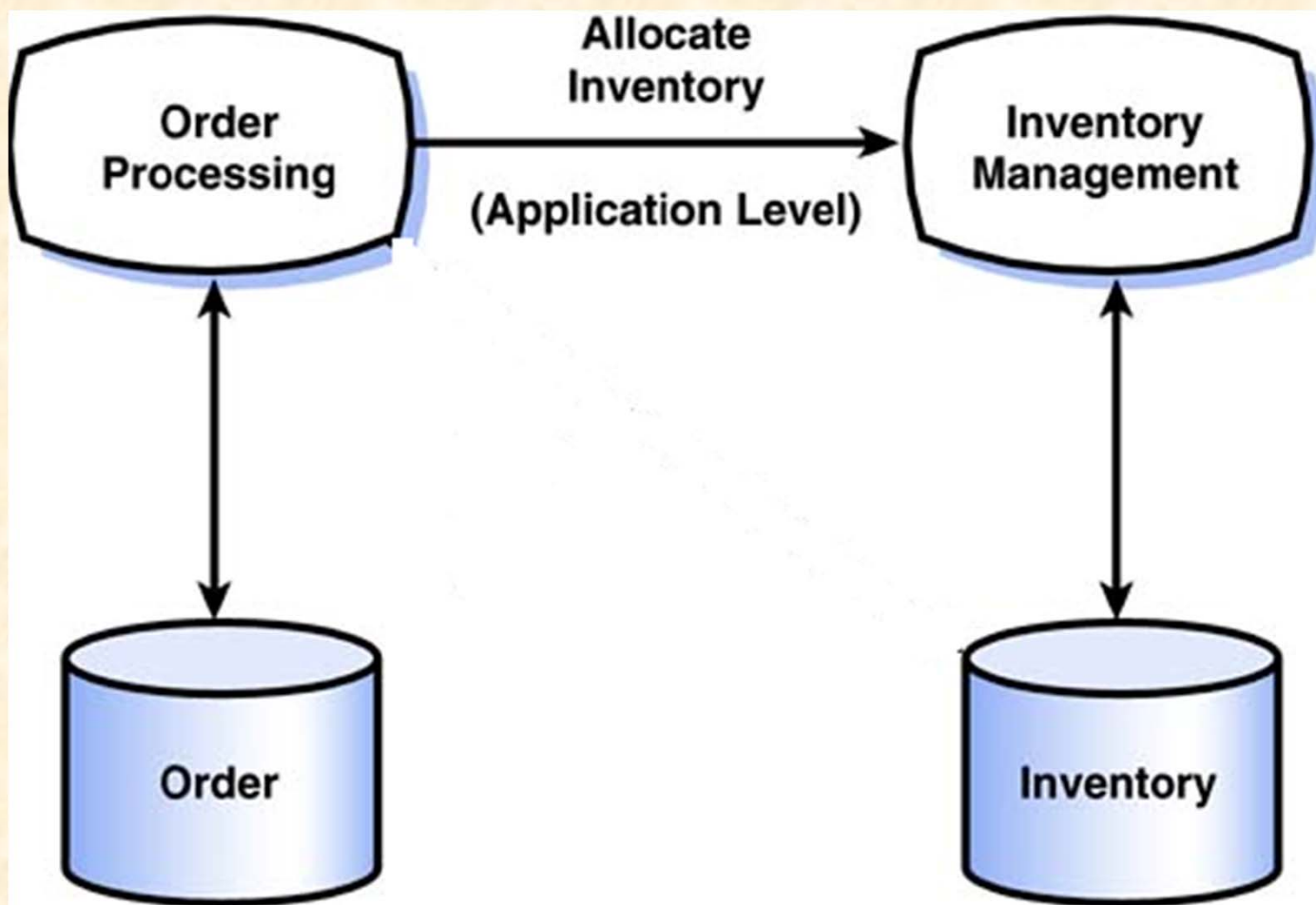


# การรวมระบบซอฟต์แวร์แบบ Data-Level Integration





# การรวมระบบซอฟต์แวร์แบบ Application-Level Integration





# Service เบื้องต้น





# Service คืออะไร

- “A Service is a thing that fulfills a purpose. A Service is a worker employed to achieve a specific end goal for a requester”



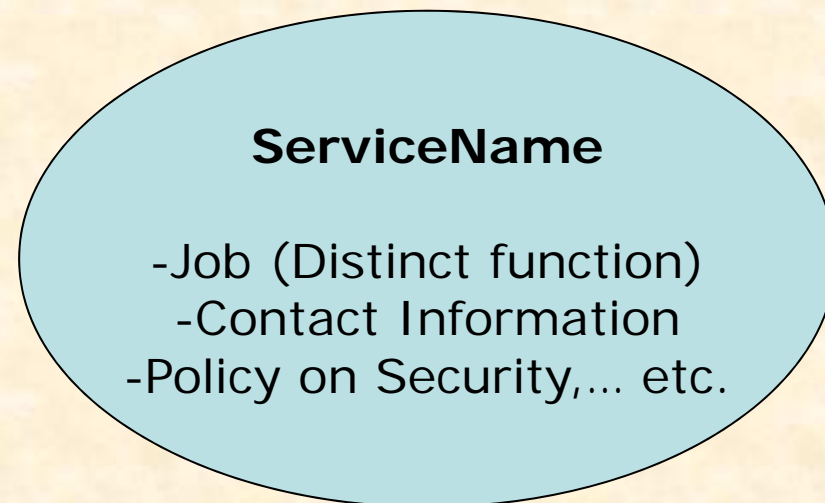
# Service Granularity

- ขนาดของ Service กำหนดด้วย Level of Granularity
  - Small in Scope (to retrieve information)
  - Large in Scope (to perform business process)



# Service

- A Service is an abstract resource that has
  - Name
  - Job (a single distinct business function/process)
  - Job Tasks
  - Contact Information
  - Policy on security
  - Etc.





# Service Collaboration

- Service ที่มีอยู่จะทำงานแบบ Collaboration ดังนี้
  - **Orchestration**  
คือมี Primary Service เป็นผู้นำเรียกใช้ Simple Services อื่นๆ ตามลำดับ อย่างลงตัว (Sequence of actions ฝังอยู่ใน Primary Service)
  - **Business Interaction**  
คือมี Coordination Mechanism ที่ทำหน้าที่จัดการ Sequence of actions ของ Long-lived Services โดยไม่ต้องฝังไว้ใน Service ใด (Coordination Mechanism อาจจะเป็น Business Process Execution Engine, Work Flow Engine, หรือ ESB)
  - **Interception**  
คือมี Intermediary Services กลางที่คอยช่วยรับ/ส่งระหว่างทางจนไปถึงผู้รับปลายทาง (Service ปลายทาง) โดยที่ผู้ส่งตัวจริง (Requesting Entity) และผู้รับปลายทางไม่ทราบว่ามี Intermediary Services เหล่านี้





# Service ทำหน้าที่แบบใดบ้าง

- Request/Reply
  - เรียกใช้และตอบกลับทันที
- Worker
  - ทำงาน Function ที่มีการเปลี่ยนสถานะของข้อมูลจริง (Changing State of the thing)
- Monitor
  - ทำหน้าที่ Observe และรายงานผลตามกฎที่ได้รับ ไม่ทำอะไรเกินกว่านั้น
- Agent
  - ทำหน้าที่ Observe และเมื่อทราบผลก็มีการกระทำต่อเนื่องจากผลที่ได้รับเลย
- Intermediary
  - ทำหน้าที่รับ/ส่ง ข้อความไปมา โดยทำหน้าที่สำคัญระหว่างทางแบบ Value-added เช่น Translation การตรวจสอบความปลอดภัย เป็นต้น
- Aggregator
  - รวบรวมผลลัพธ์จาก Services ต่างๆ เพื่อทำหน้าที่ประสานระหว่างกลุ่ม Services ที่มีอยู่มากมาย
- Process
  - เป็น Long-running Service ทำหน้าที่ติดต่อกับ Services ที่จำเป็นในการทำงานของ Business Process หนึ่งๆ ให้สำเร็จอย่างแท้จริง

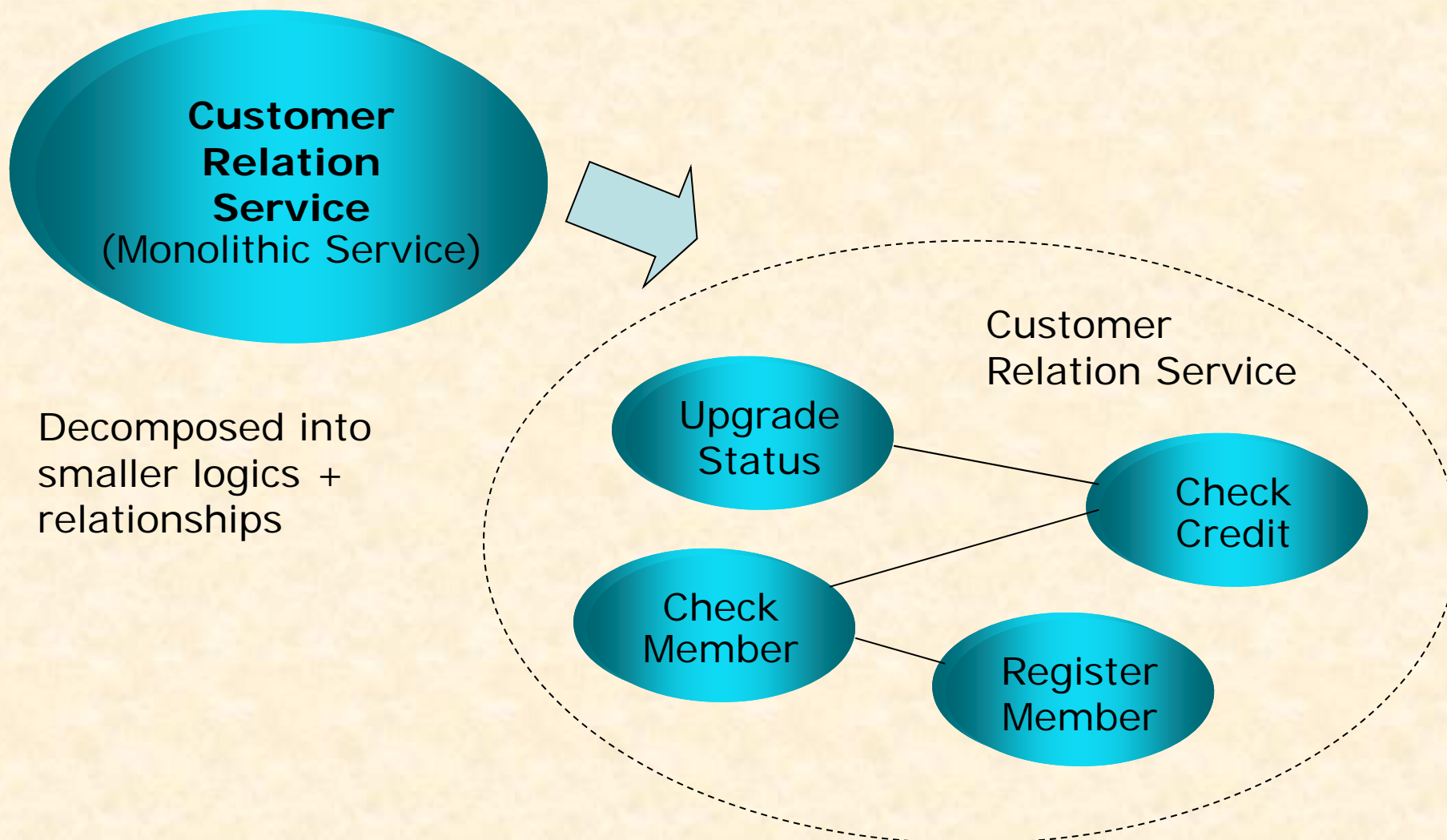


# ตัวอย่าง Service

- Services are reusable components that represent business tasks
  - Customer lookup
  - Account lookup
  - Credit card validation
  - Credit check
  - Hotel reservatio
  - Interest calculation
- Services can be
  - Globally distributed across organizations
  - Reconfigured into new business processes

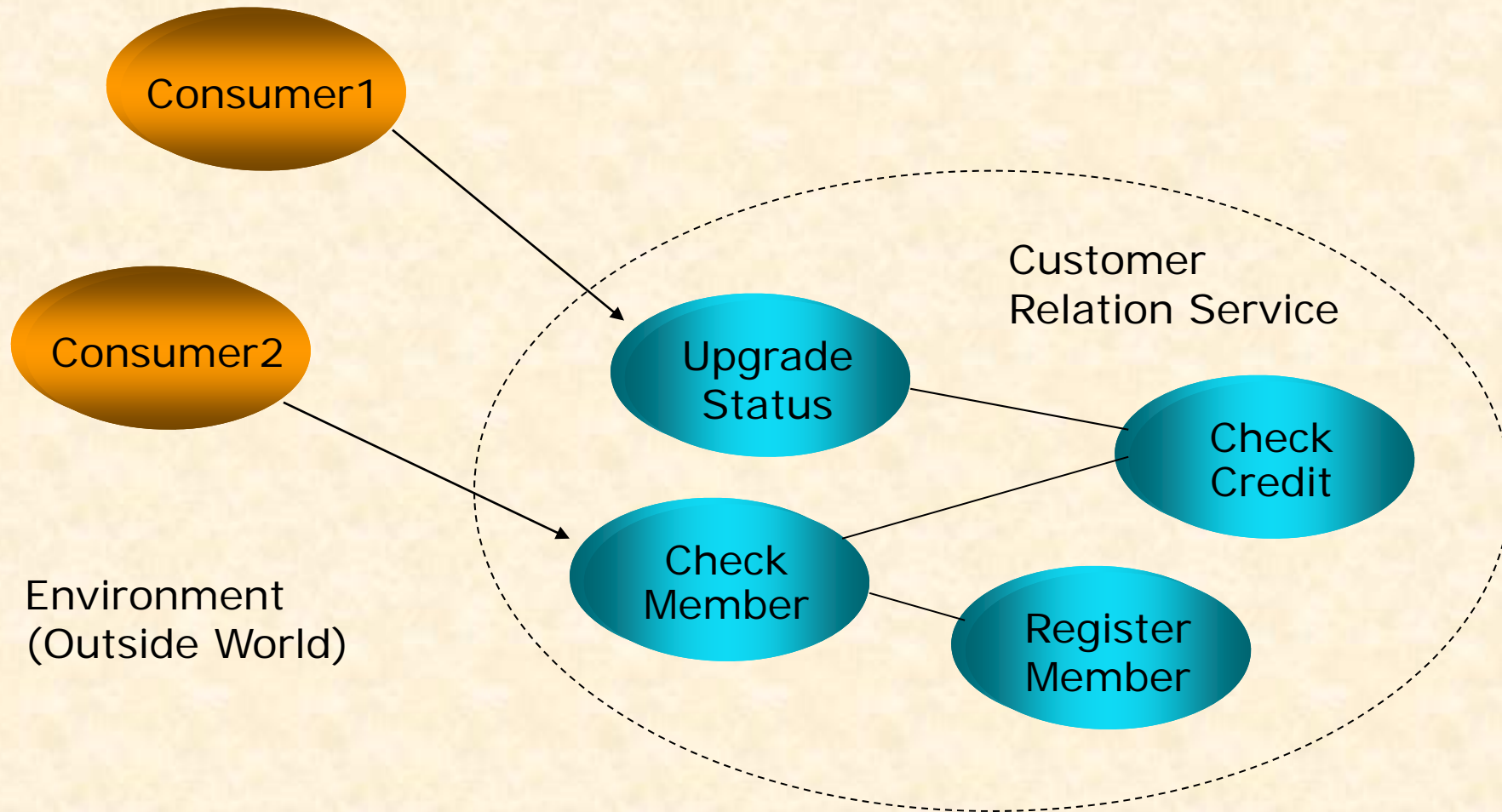


# Monolithic Service (Runtime)





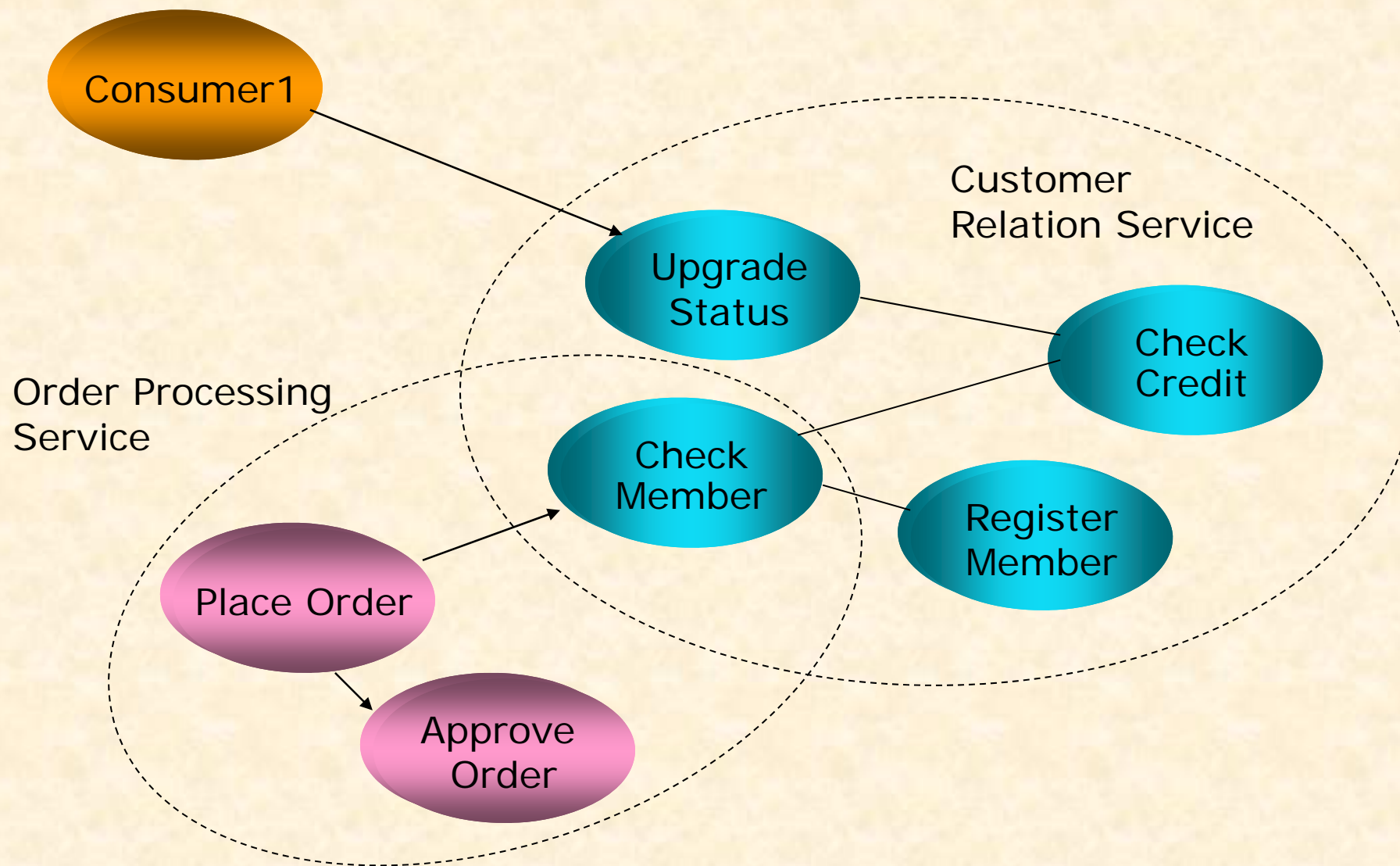
# Service and Its Environment





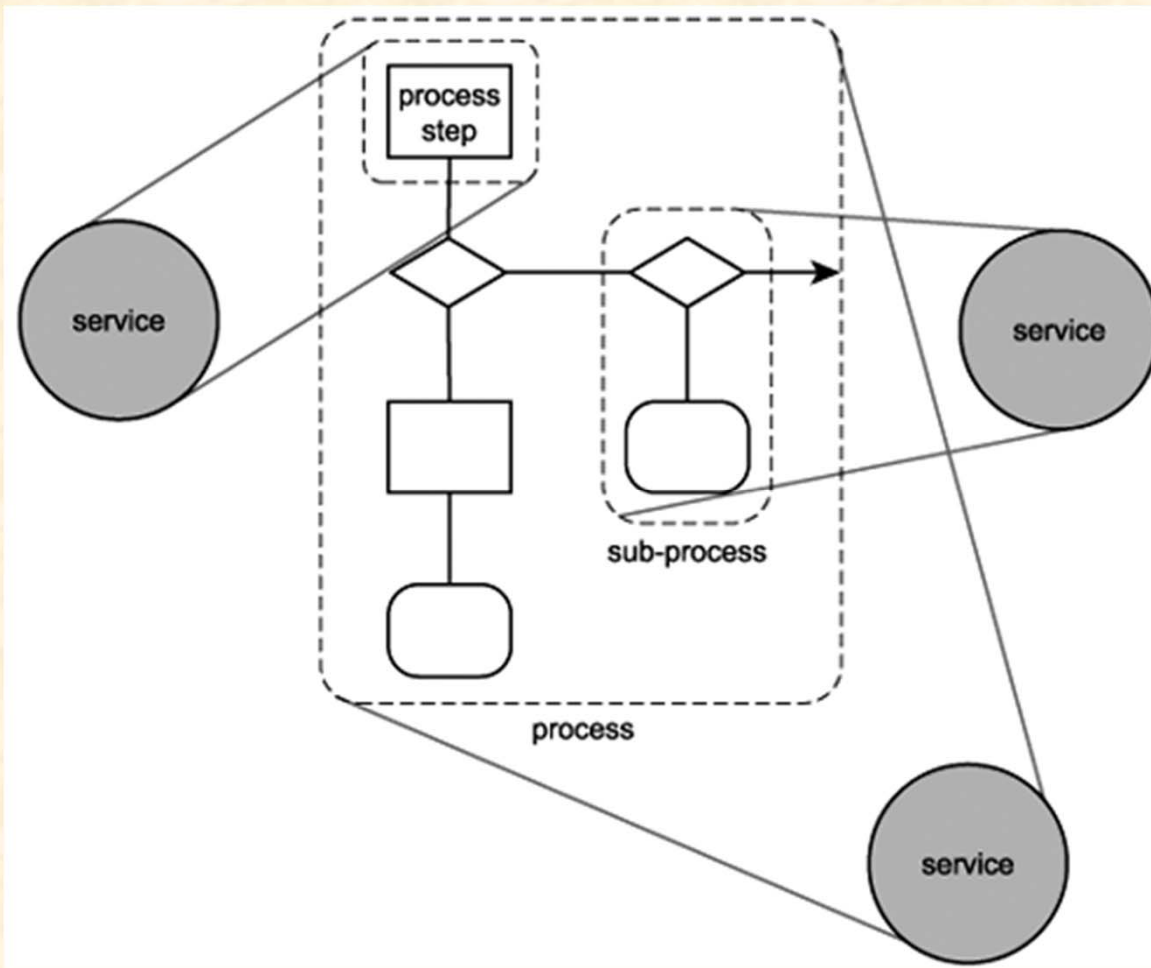


# Service is Reusable



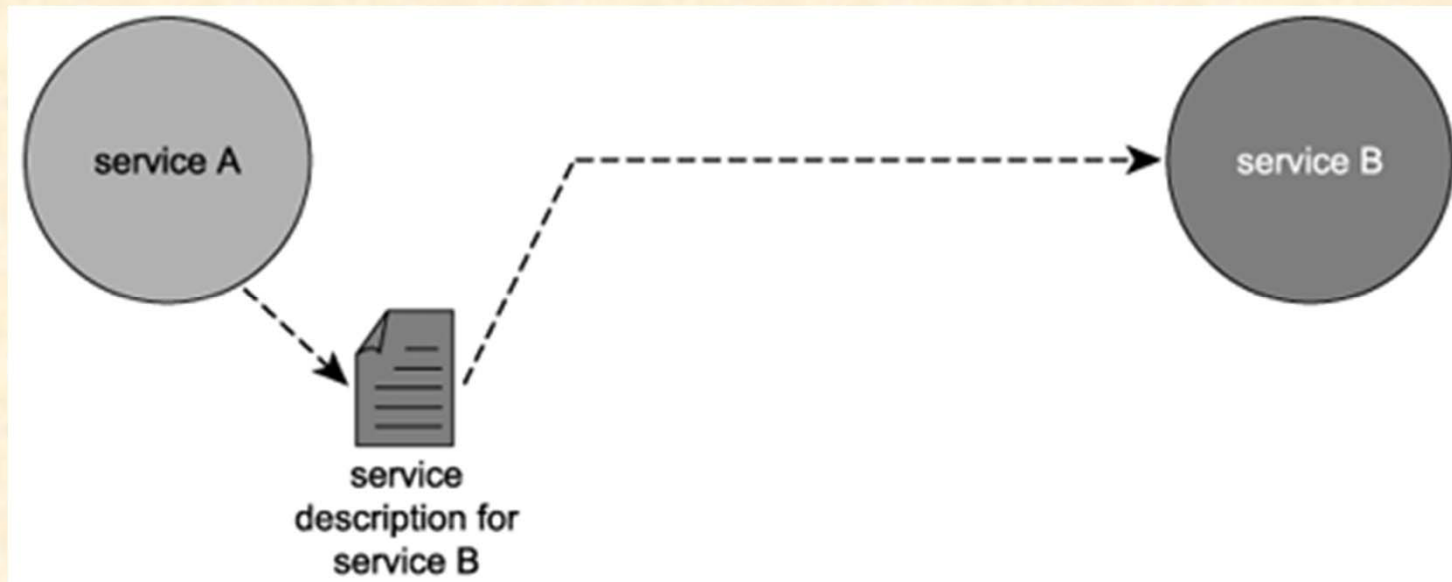


# How Service Encapsulate Logics



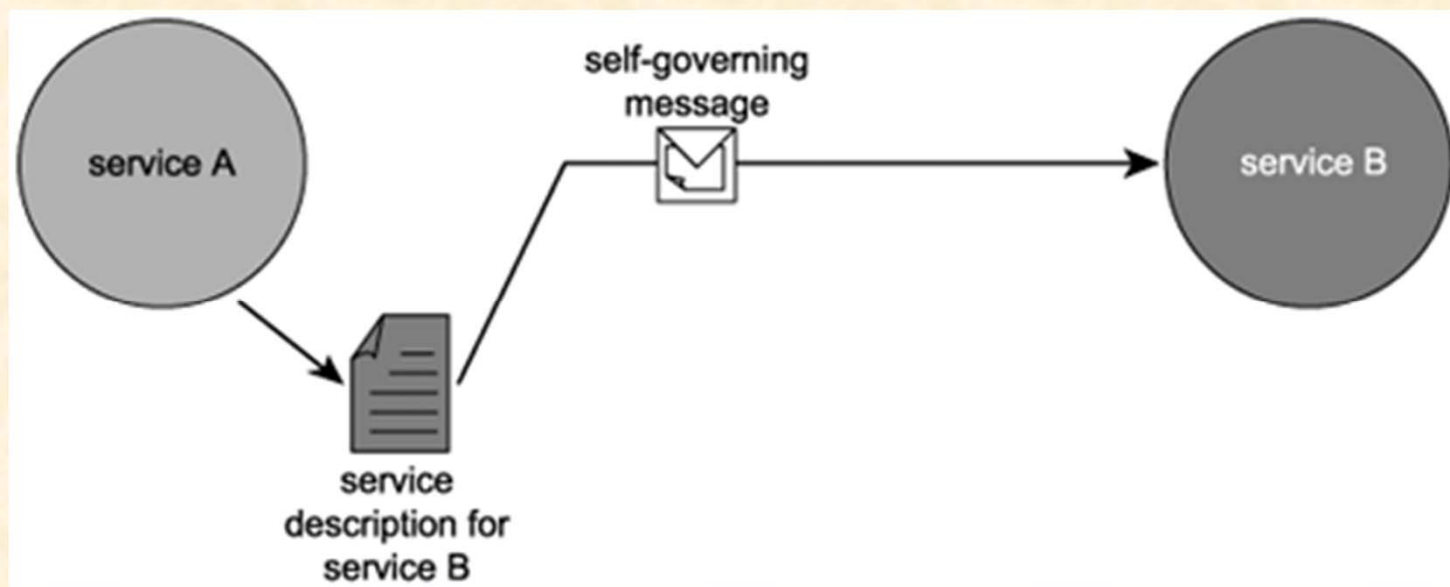


# How Service Relates



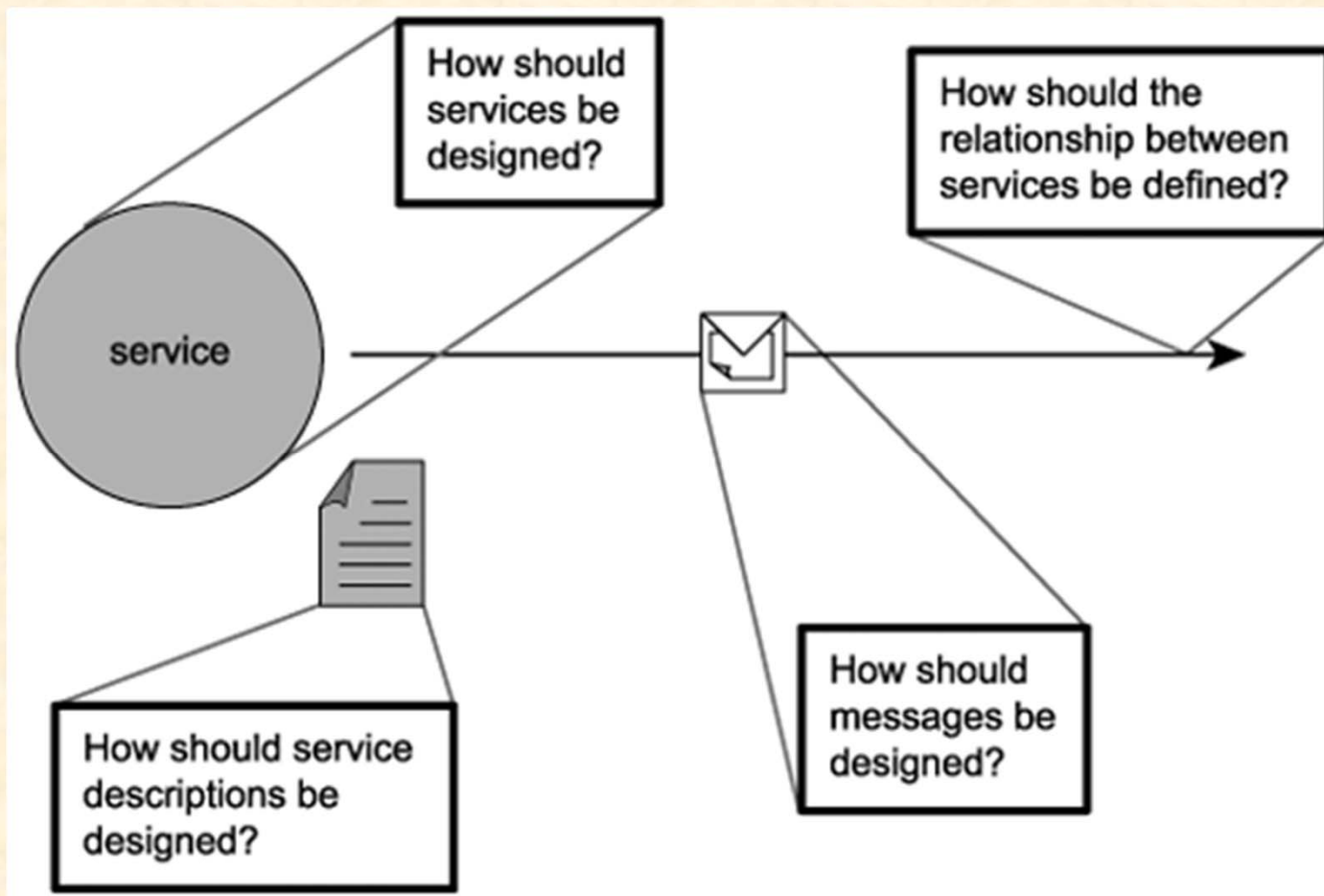


# How Services Communicate





# How Services are designed







# Architecture เบื้องต้น



# Architecture คืออะไร

- IEEE 1471-2000  
“Software architecture is the fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution”



# SOA เบื้องต้น



# SOA คืออะไร

- Service-oriented architecture is a way of designing systems that enables
  - Cost-efficiency
  - Agility
  - Adaptability
  - Leverage of legacy investments





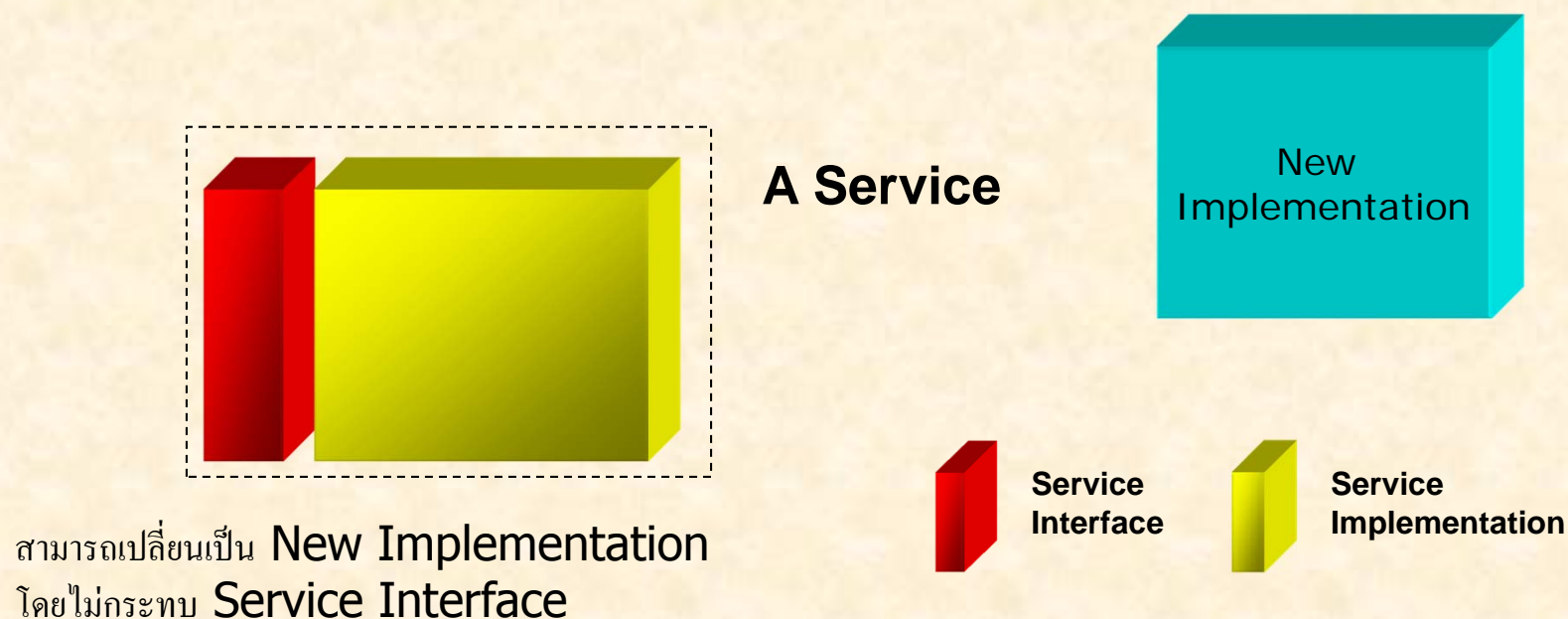
# ส่วนประกอบของ SOA

- SOA เป็นสถาปัตยกรรมระบบที่ประกอบด้วย
  - Loosely Coupled Services
  - Described by platform-agnostic interfaces
  - that can be discovered and invoked dynamically



# A Loosely Coupled Service

- Loosely Coupled หมายถึง Service Implementation แยกออกจาก Service Interface

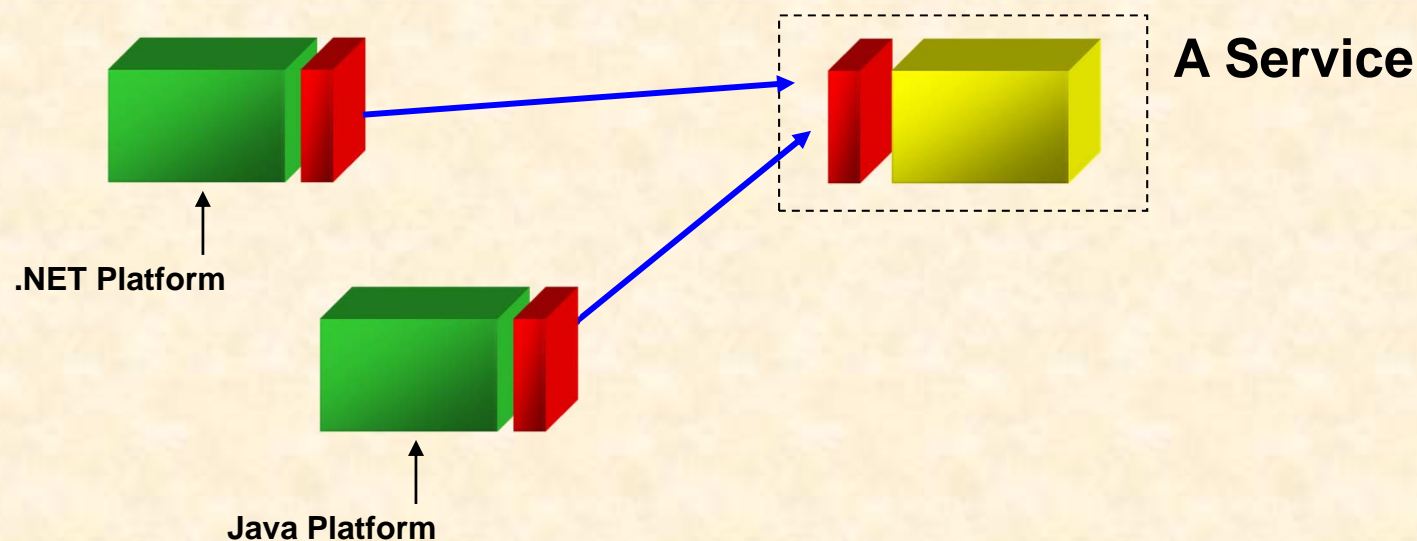




# Platform-Agnostic Interface

- Client ที่มี Platform ที่ต่างกันสามารถมาใช้บริการ Service ได้โดยผ่านทาง Interface ชนิดนี้

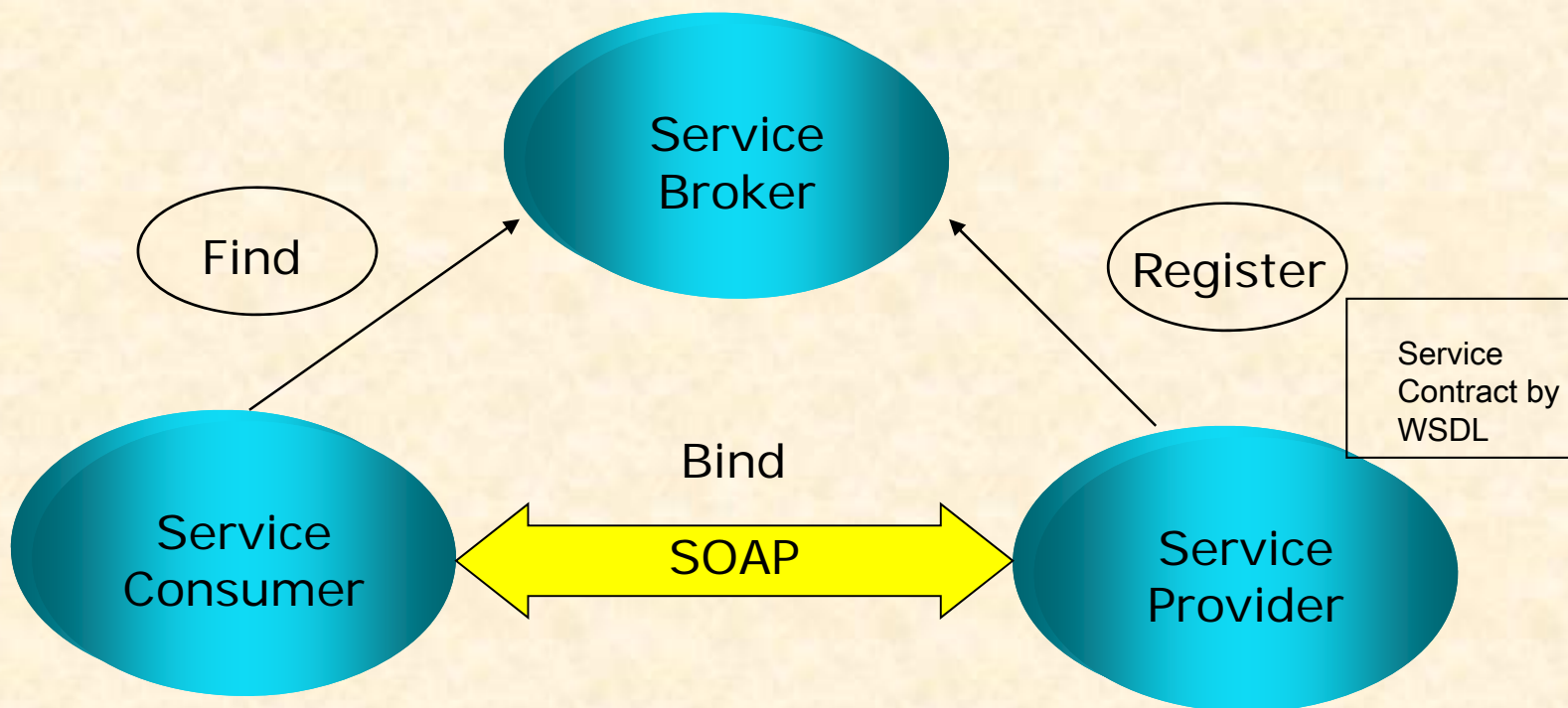
## Service Consumer





# Service can be Discovered

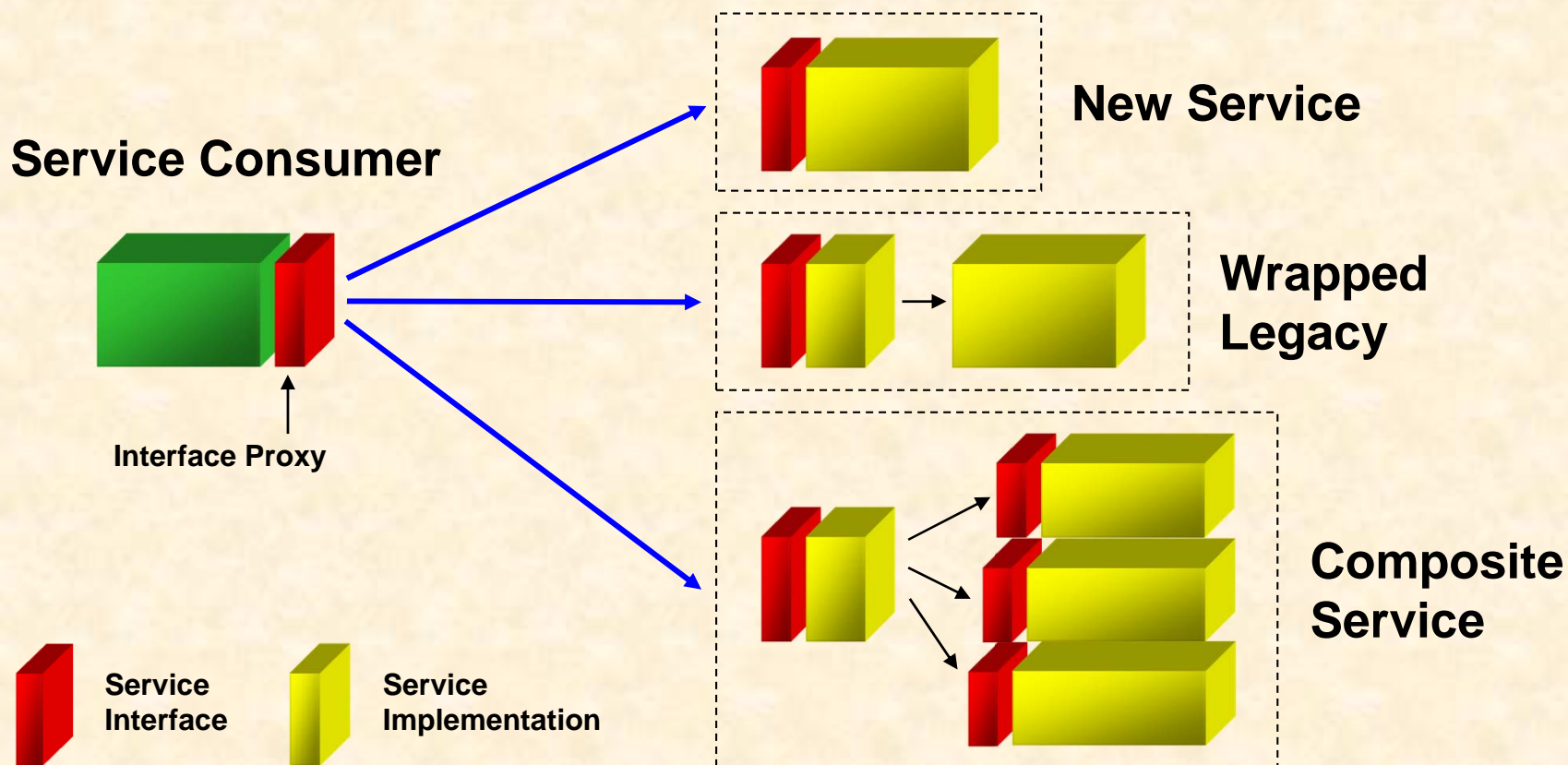
- Service Consumer ไม่ต้องรู้จัก Service Provider มาก่อน เหมือนกับ Client-Server ทั่วไป





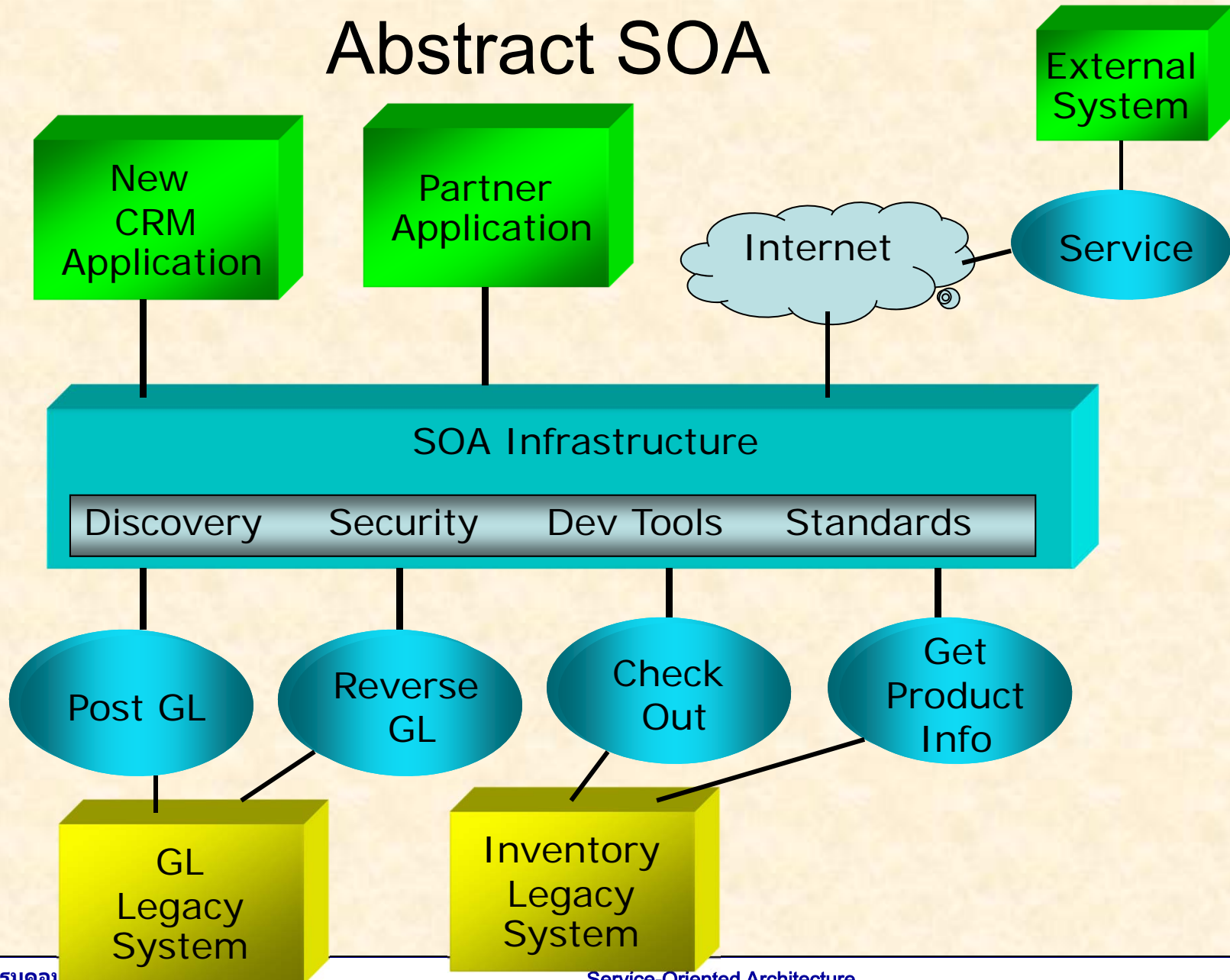


# SOA ประสานการทำงานกับ Legacy System





# Abstract SOA





# สรุป

- Service Oriented Architecture เป็นสถาปัตยกรรมของการออกแบบระบบประเภทหนึ่งที่เราคุ้นเคยกันมากระยะหนึ่งแล้ว
- SOA เน้นการรวมระบบซอฟต์แวร์เข้าแบบองค์รวม โดยให้มีการกำหนด Service และ Collaborate ระหว่างกันและกันเท่านั้นในระดับ Application-Level
- มีเทคโนโลยีในการพัฒนา SOA หลากหลายแบบที่ควรเลือกให้เหมาะสม