

4. Service-Oriented Modeling

Jiří Vokřínek

Agent Technology Center
Department of Cybernetics

Faculty of Electrical Engineering, Czech Technical University in Prague

vokrinek@agents.felk.cvut.cz

<http://agent.felk.cvut.cz>

SOA Principles

- Service-oriented architecture
- Service-oriented analysis and design
- Service-oriented modeling
- Service-oriented computing
- Service-oriented programming
- ... separation of concerns to services

Service-oriented Modeling

- Designing and specifying service-oriented business systems within a service-oriented architecture
- Includes a modeling language understandable by both business and technical people
- Comprehensive view of analysis, design, and architecture of 'Software Entities' in an organization
- Encourages viewing software entities as 'assets' referred as 'services'

Service-oriented Modeling

- Service-Oriented Modeling and Architecture (SOMA) by IBM in 2004
 - Targets service-oriented analyses and design (SOAD) – focus to service components and flows
 - Extends traditional object-oriented and component-oriented analyses and design
 - Three phases: identification, specification, and realization (+ implement, deploy, manage)
 - Domain decomposition, goal-service modeling

Service-oriented Modeling

- Service-Oriented Modeling Framework (SOMF)
- Michael Bell¹
- Sparx Enterprise Architect modeling platform²
- Modeling language for software development
- Can be employed to design any application
(application-level or enterprise-level, local or distributed,
business or technological)

¹ <http://www.modelingconcepts.com>

² <http://www.sparxsystems.com.au/>

SOMF

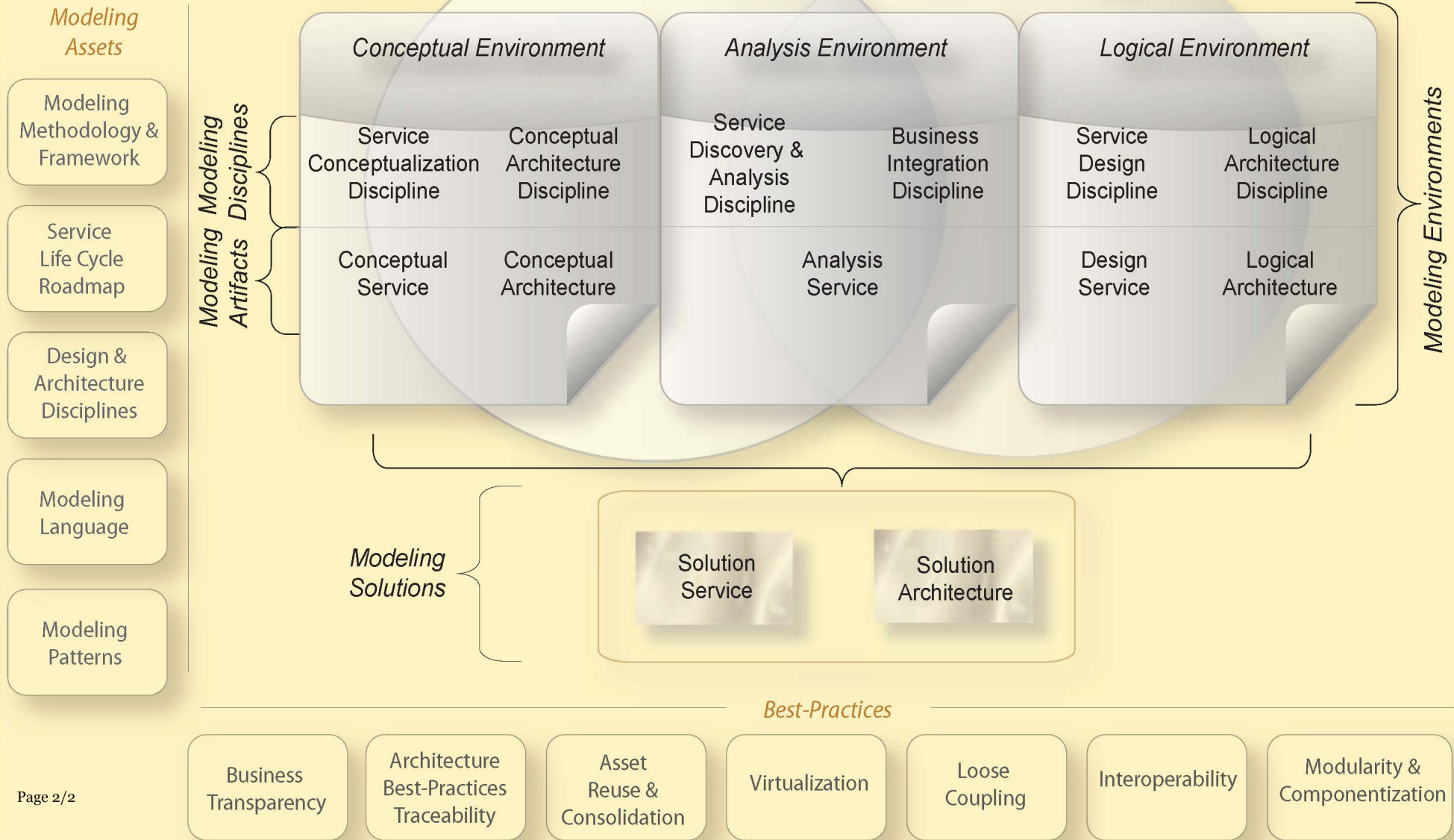
- Methodology for service-oriented development (lifecycle management and modeling)
- Intuitiveness of implementation and simplicity of usage
- Number of modeling practices, environments, disciplines, and artifacts
- <http://www.modelingconcepts.com/pages/download.htm>

SOMF

- Not based on any particular programming language, nor constrained to any implementation technology (e.g. Web Services)
- Model-driven analysis, design and architectural disciplines
- Software lifecycle and service portfolio management practices
- An easy to use notation for modeling the “used-to-be”, “as-is”, and “to-be” states of the enterprise service catalog

Service-Oriented Modeling Framework (SOMF)

Service-Oriented Modeling Practices



SOMF for Software Development

- Service-Oriented Conceptualization
- Conceptual Architecture
- Service-Oriented Discovery and Analysis
- Service-Oriented Business Integration
- Service-Oriented Design
- Logical Architecture

SOMF Modeling Language

● Analysis Model

- Service-Oriented Analysis Proposition Diagram
- Service-Oriented Logical Relationship Diagram

● Design Model

- Service-Oriented Business Integration Diagram
- Service-Oriented Logical Composition Diagram
- Service-Oriented Transaction Diagram

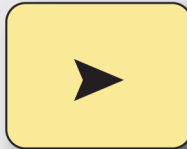
SOMF Modeling Language

● Architecture Model

- Service-Oriented Conceptual Architecture Diagram
- Service-Oriented Utilization Diagram - Logical Architecture
- Service-Oriented Transaction Directory Diagram - Logical Architecture

SOMF Modeling Patterns

SOMF PATTERNS



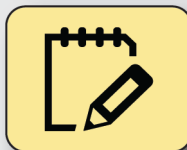
Discovery and Analysis Road Map Patterns



Service Identification Patterns



Service Categorization Patterns



Contextual Analysis and Modeling Patterns



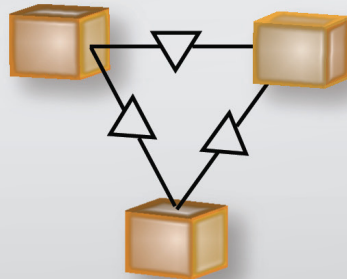
Structural Analysis and Modeling Patterns



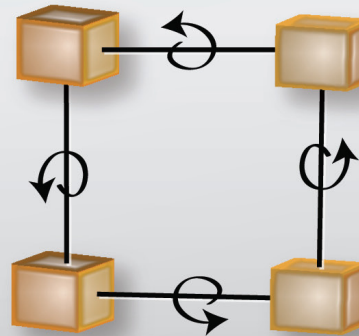
SOMF Modeling Styles

SOMF Styles

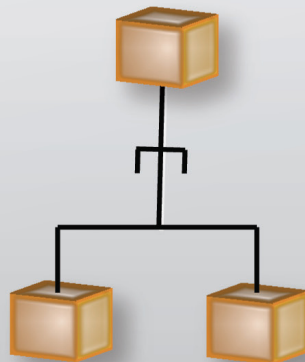
Network



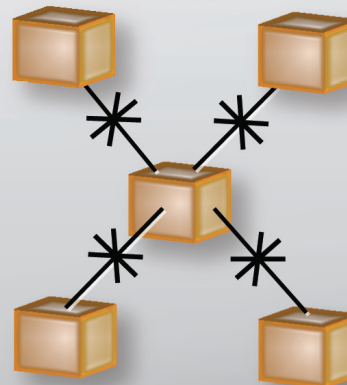
Circular



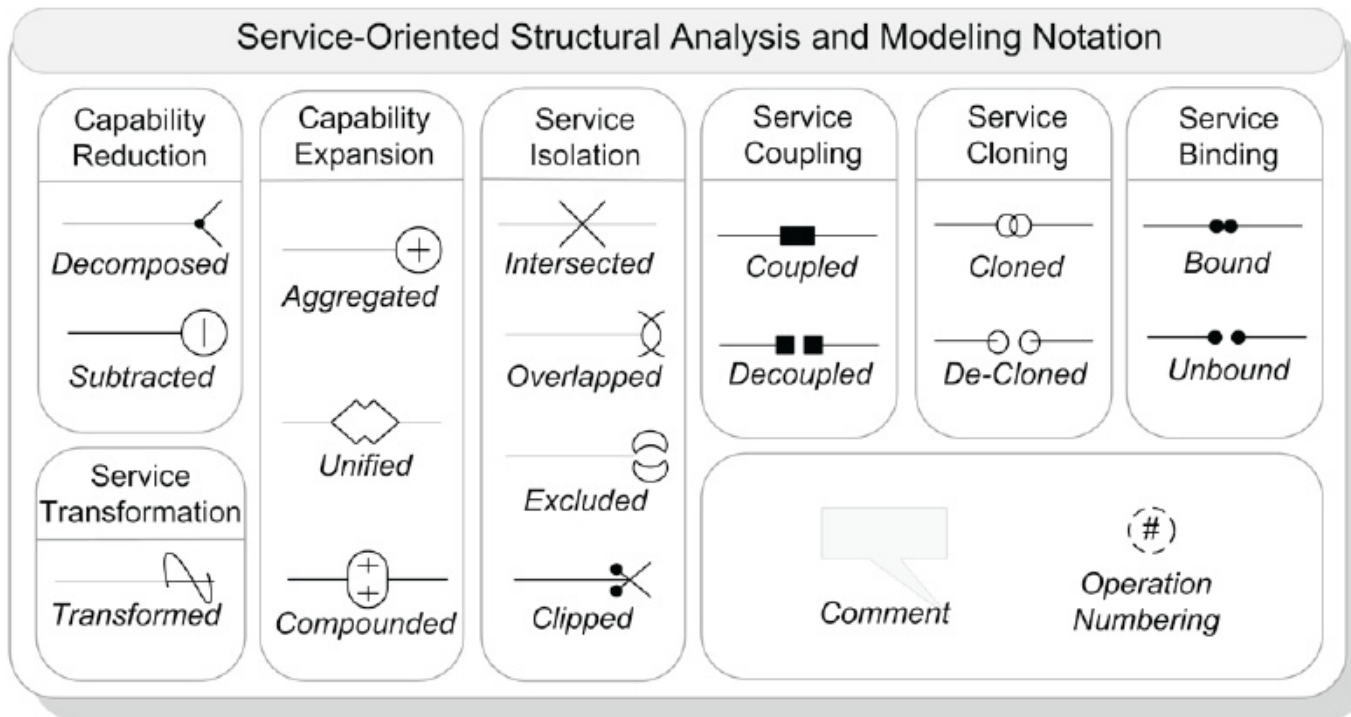
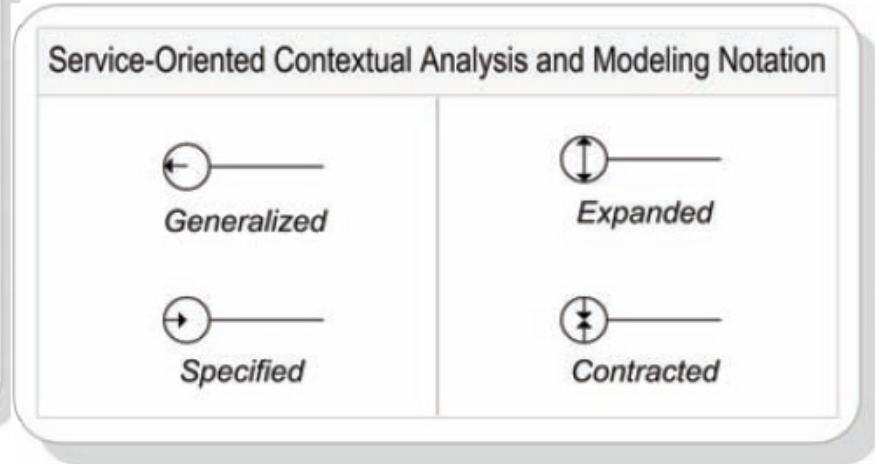
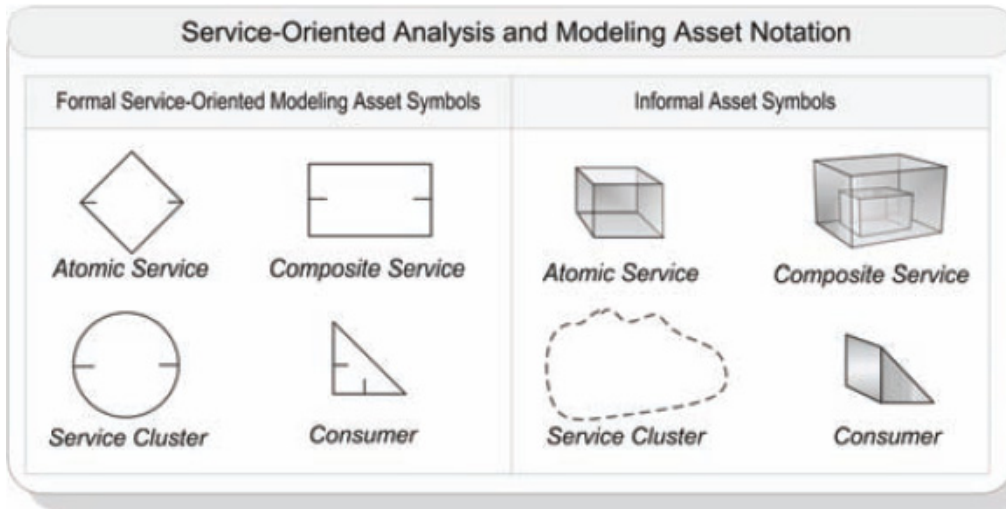
Hierarchical



Star



SOMF Notation



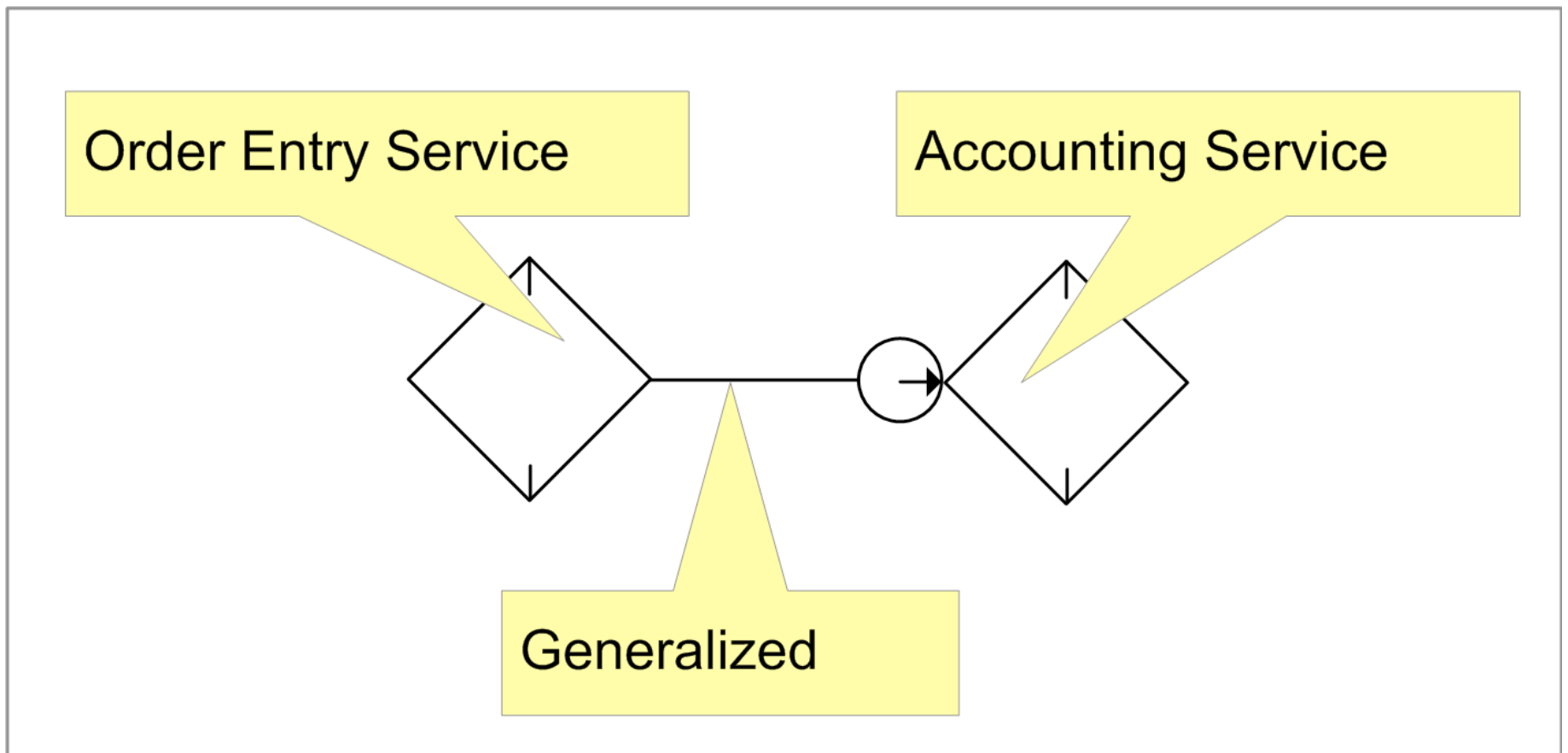
Motivation: How to Discover and Analyze services for Granularity, Reusability, Interoperability Success, etc...

Contextual Modeling

- Simple way to describe the capabilities of a software component
- Describes service functionality, name, specialty, and role
- “Art” of manipulating the context of a service to perfect its offerings and performance
- Generalization, Specification, Expansion, and Contraction

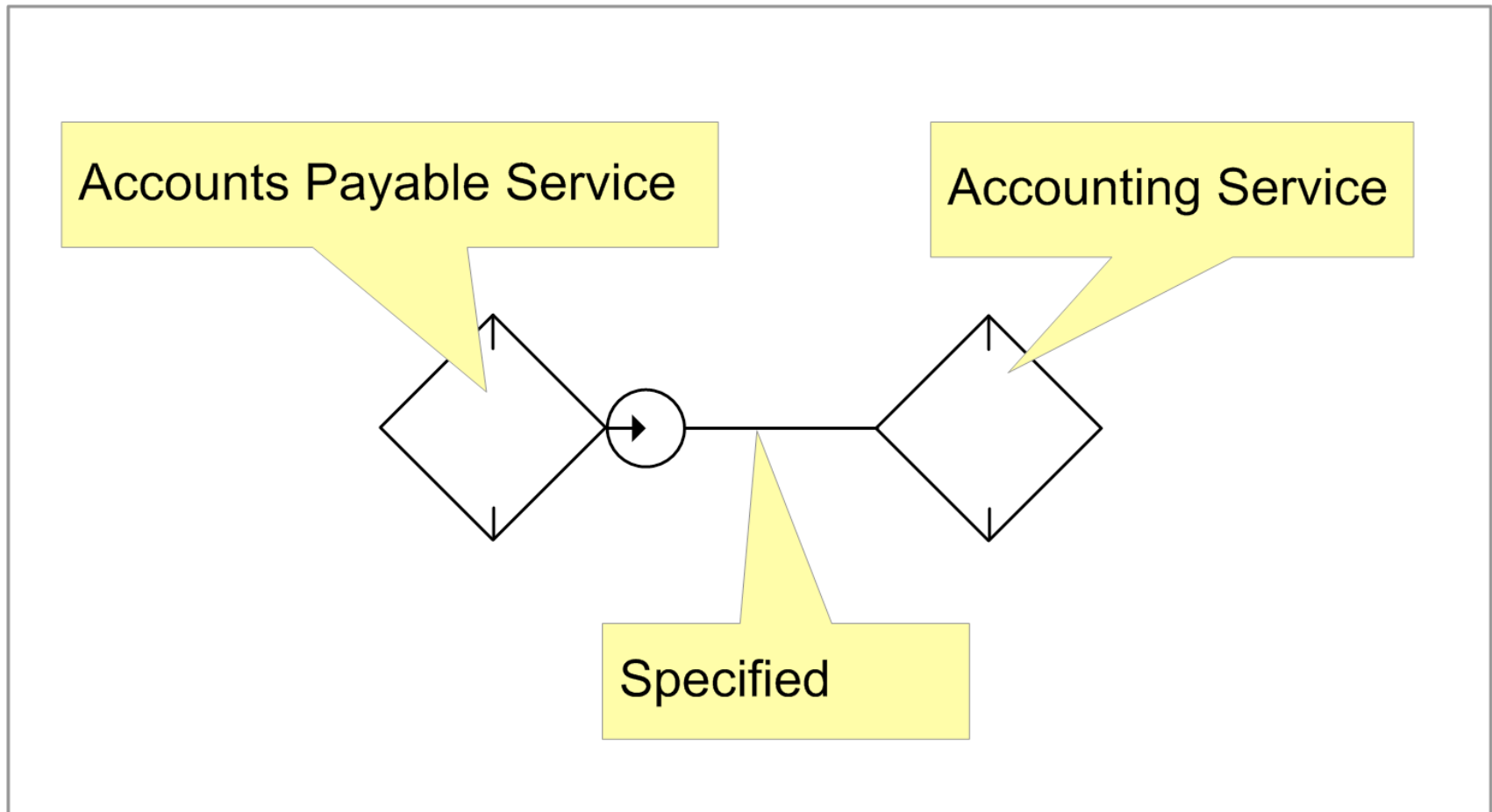
Contextual Generalization

- Raising the level of abstraction



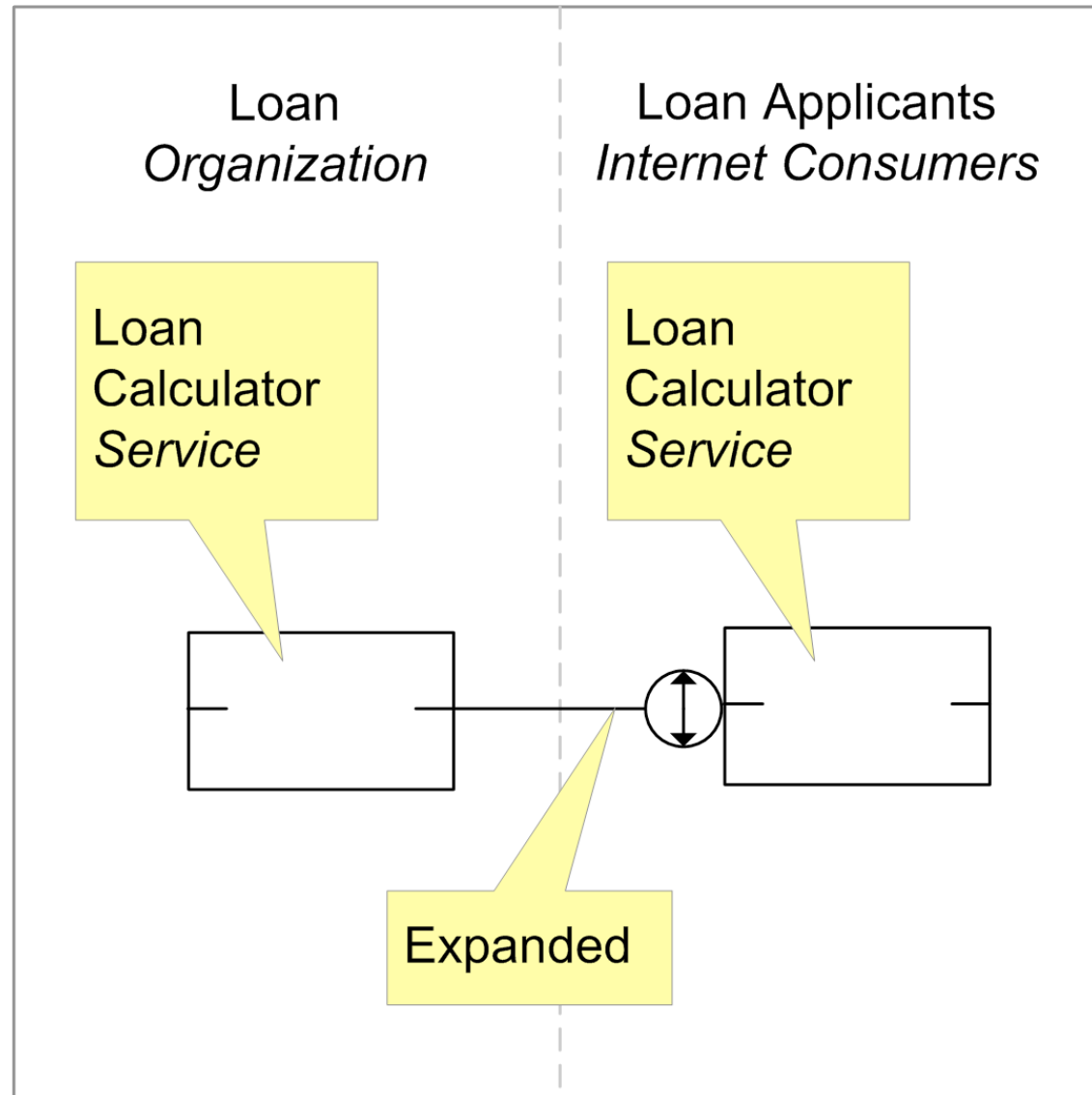
Contextual Specification

- Reducing service abstraction level (trim down the functionality)



Contextual Expansion

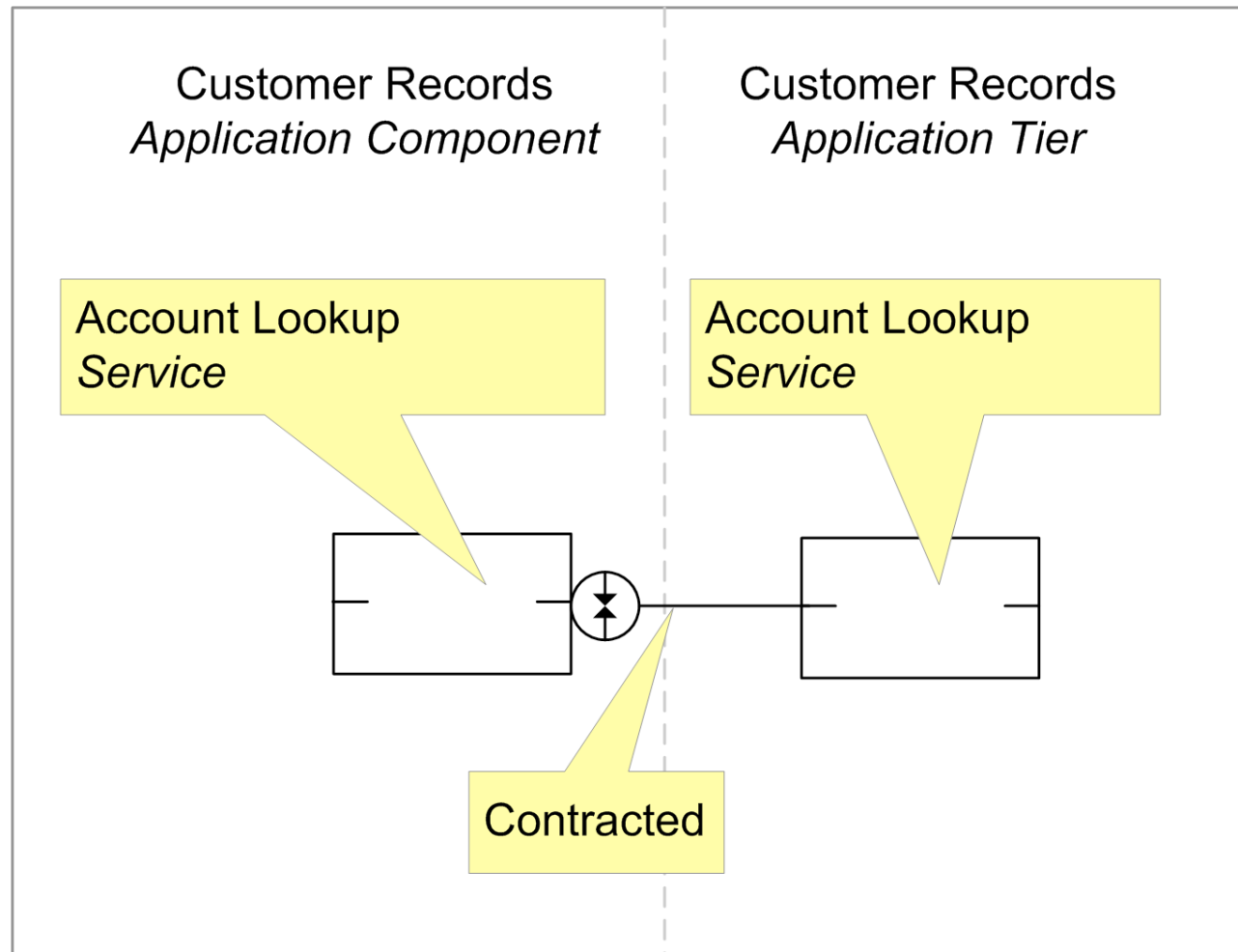
- Increasing service influence and offerings across boundaries
- increasing service's consumer base



Contextual Contraction

- Reducing a service's consumer base and decreasing its influence

- limits accessibility to the service

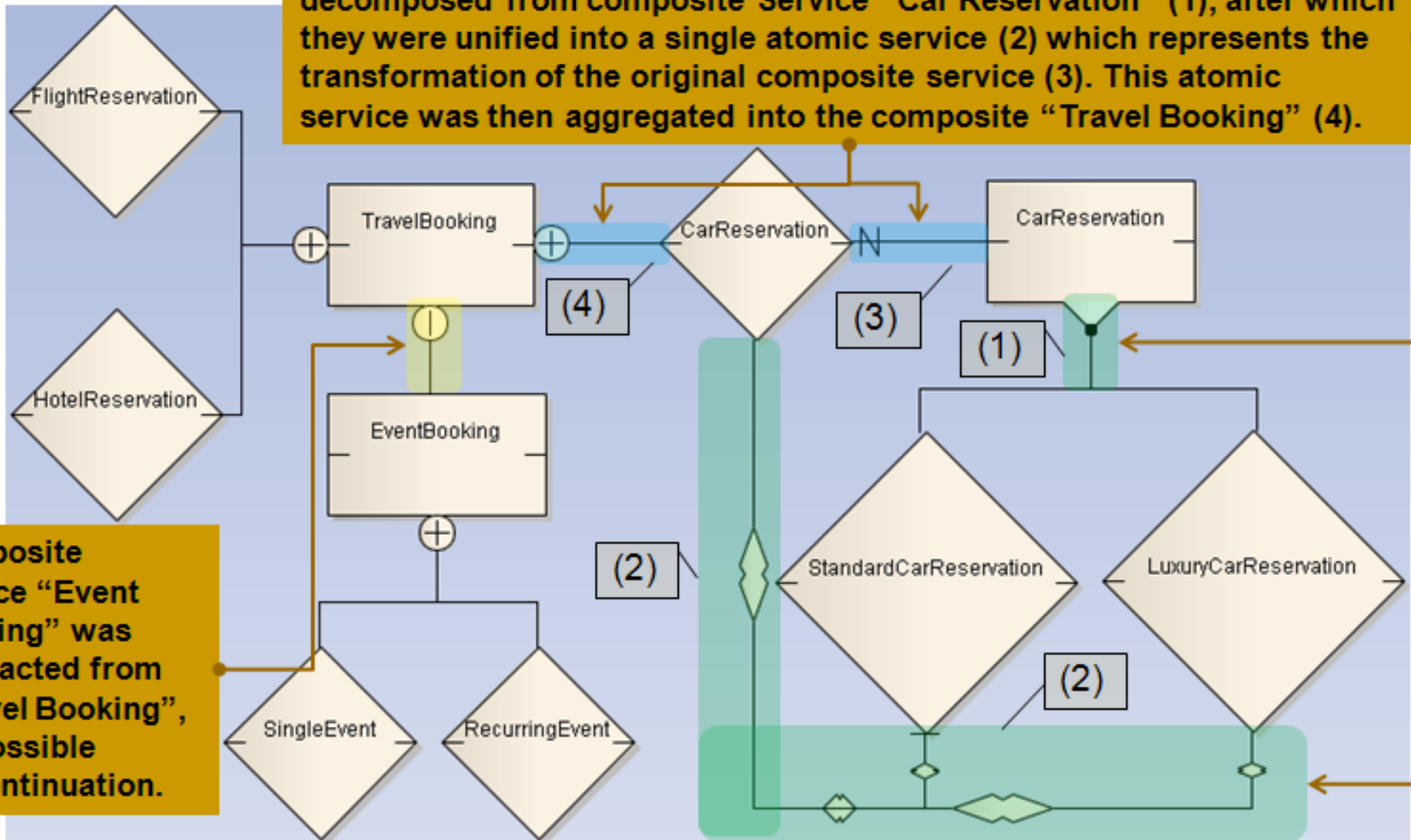


Examples of Diagrams

- Analysis Proposition Diagram
- Business Integration Diagram
- Logical Relationship Diagram
- Logical Composition Diagram
- Transaction Diagram

Analysis Proposition Diagram

Atomic services “Standard” and “Luxury Car Reservation” were decomposed from composite Service “Car Reservation” (1), after which they were unified into a single atomic service (2) which represents the transformation of the original composite service (3). This atomic service was then aggregated into the composite “Travel Booking” (4).

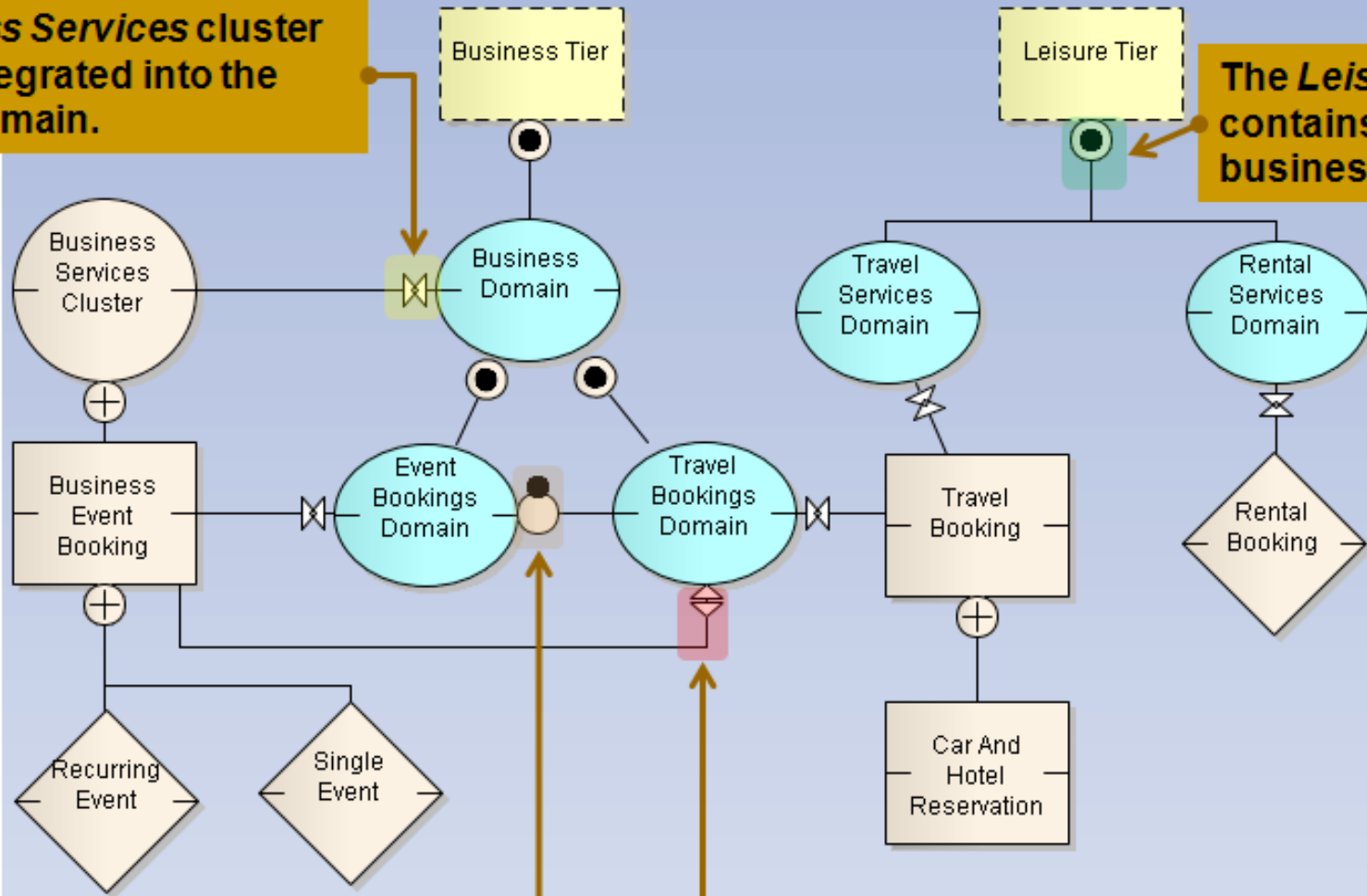


Composite service “Event Booking” was subtracted from “Travel Booking”, for possible discontinuation.

Business Integration Diagram

The *Business Services* cluster has been integrated into the *Business* domain.

Business Tiers and Business Domain elements have been color coded for visual emphasis.



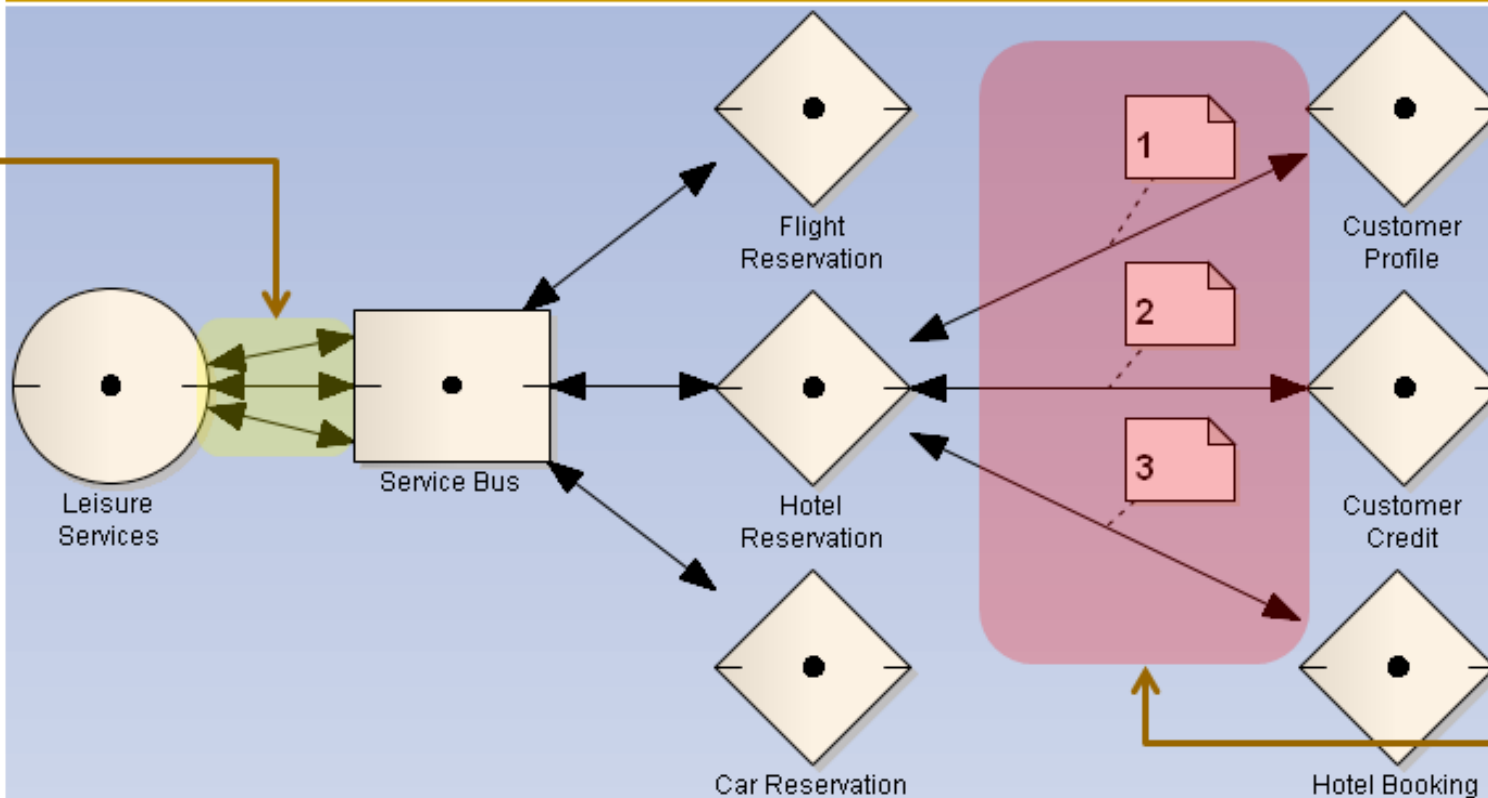
The *Leisure Tier* contains two business domains.

The *Event Bookings* business domain has been separated from the *Travel Bookings* domain.

The *Business Event Booking* composite service has been disintegrated from the *Travel Bookings* domain.

Service-Oriented Logical Design Relationship Diagram

The example on the left illustrates a “same-time”, asynchronous (non-blocking) design whereby messages can be sent & received in no particular order. To coordinate the messages an intermediary such as an ESB can be used.

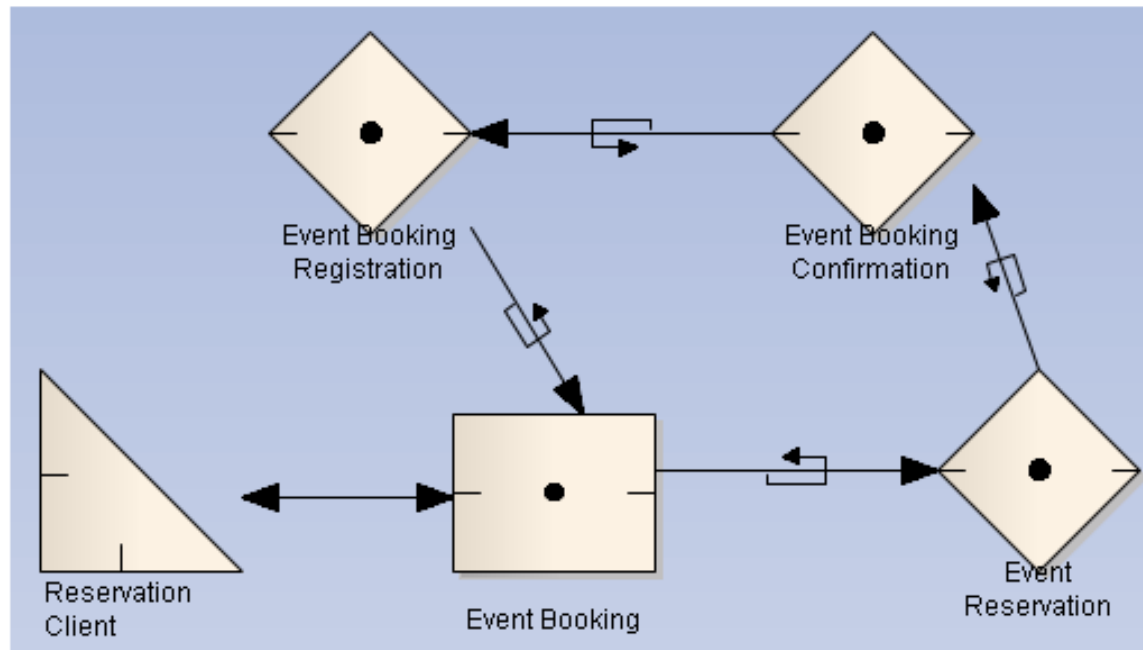


In EA, message sequencing can be documented using connector notes, as shown in this example.

The example on the right depicts an “in-order” sequential design whereby messages must be sent (and their reply received) in a particular order. This implies the calling service is blocked waiting for each response.

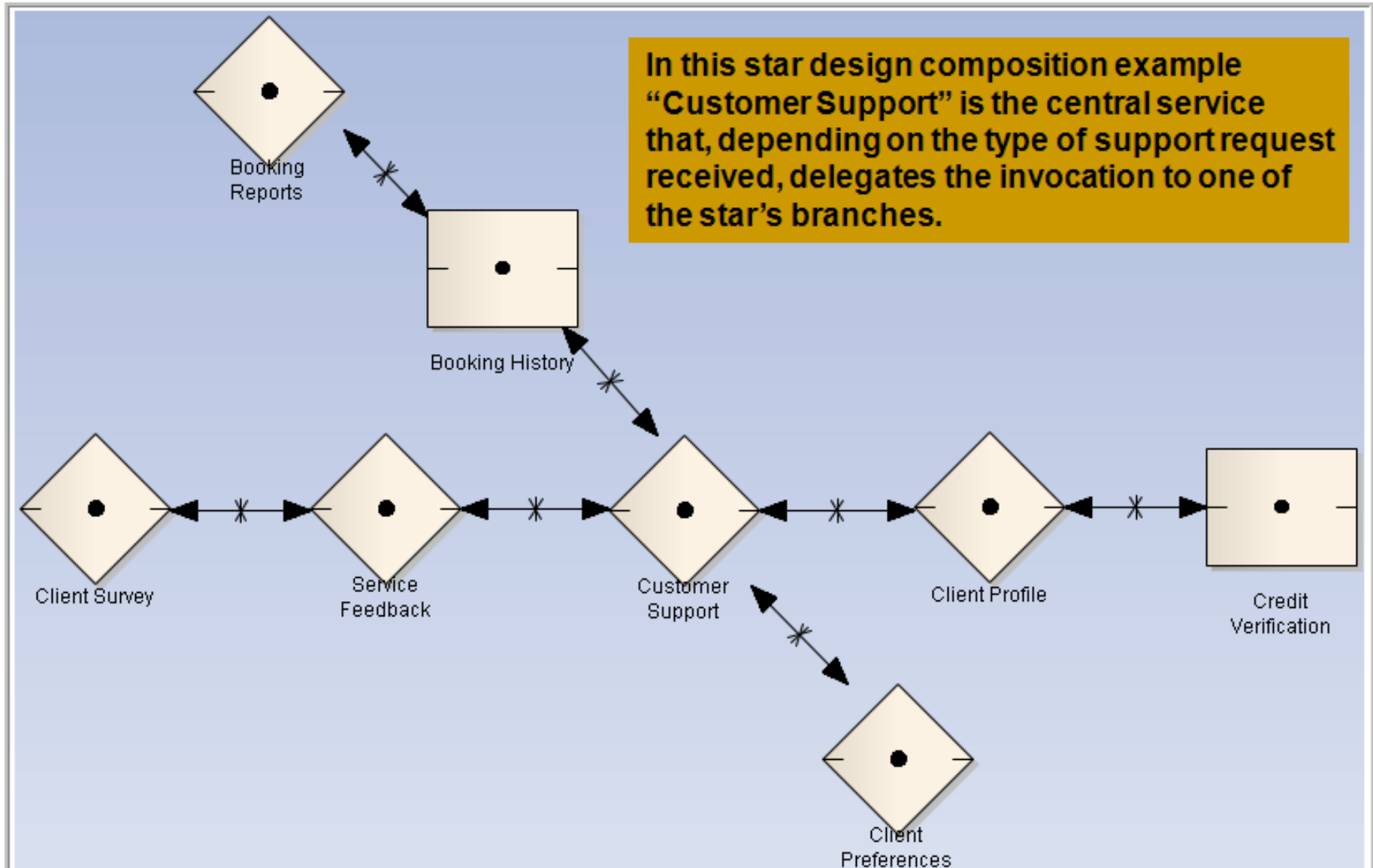
Logical Design Composition Diagram Employing the Circular Style

In this example the “Event Booking” composite service delegates the request from the consumer to the first service in the chain, “Event Reservation”. The latter, in turn, passes the request (which may be altered along the way) to the “Event Booking Confirmation” service, and so forth, until the originator receives the final message in the sequence.

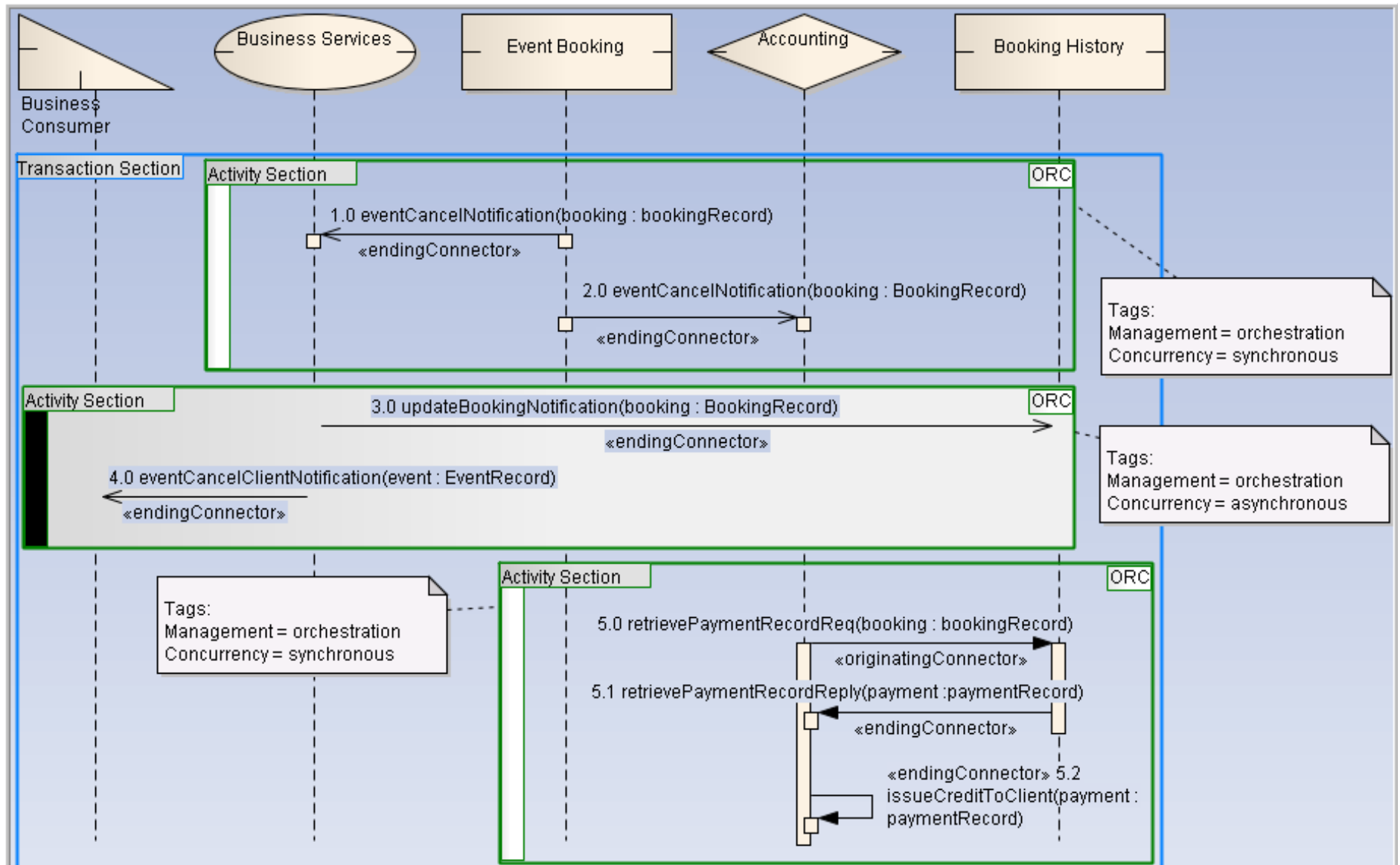


Note that in addition to the circular beam marker this style is also apparent via the unidirectional connectors of the message flow.

Logical Design Composition Diagram Employing the Star Style



Service-Oriented Transaction Diagram



SOAM in SOMF

● Service-oriented analysis modeling example

http://www.modelingconcepts.com/pdf/SOMF_ANALYSIS_MODELING.pdf



Service-Oriented Modeling Framework (SOMF) for Business & Technology

An SOA Implementation Framework

Service-Oriented Discovery & Analysis Example