# e-Business Modeling
## 406.306 Management Information Systems

**Jonghun Park**

**jonghun@snu.ac.kr**

**Dept. of Industrial Engineering**

**Seoul National University**

**9/20/2007**

# models and systems

- model: an abstract representation of the real world that reduces **complexity** and represents only the **details necessary** for a specific purpose
- modelling
  - designing of software applications before coding.
  - allows developers to consider alternatives, select the best option, work out details, & achieve agreement before they start building an application
- system: a set of elements, characterized by interrelations, with a common objective
- enterprise modeling
  - follows systems approach: definition of system boundaries -> decomposition (determination of components and their interrelationships)
- reference model: representation of particular types of organizations that help to decompose organizations into subsystems and to identify critical ISs
- modelling methodology: specifies activities to be performed, roles of participants, techniques to be used and deliverables to be produced

# business modeling

- business modelling
  - a methodology for business processes
  - provide means of expressing businesses in terms of business activities and collaborative behaviour
- business model
  - descriptive
    - limits the types of objects, relationships, and properties to be identified and modeled
    - flexible
  - normative
    - forces modelers to select from a prespecified set of constructs and map the perceived system into this prespecified set
    - less flexible, but rigorous and consistent
    - e.g., UML, SCOR, BPMN

# Business Processes and Collaborations

- Collaborative processes span multiple enterprises
- Business collaboration activities
  - a predefined set of activities and processes of partners that is initiated by a partner to accomplish an explicitly shared business goal and terminated upon recognition of one of the agreed conclusions by all the involved partners
  - specified by a business process analyst as use cases, requirements and business object flow graphs that defines the choreograph of business processes
- A collaborative process
  - a process that is implicit in the behavior and interaction between multiple business entities described as having different roles
  - The activities of a collaborative process are the actions performed by the participating entities **in response to the messages** they receive from other participating entities

# business modeling with UML

- UML
  - defines a standard language and graphical notation for creating models of business and technical systems
  - defines model types that span a range from functional requirements and activity workflow models to class structure design and component diagrams
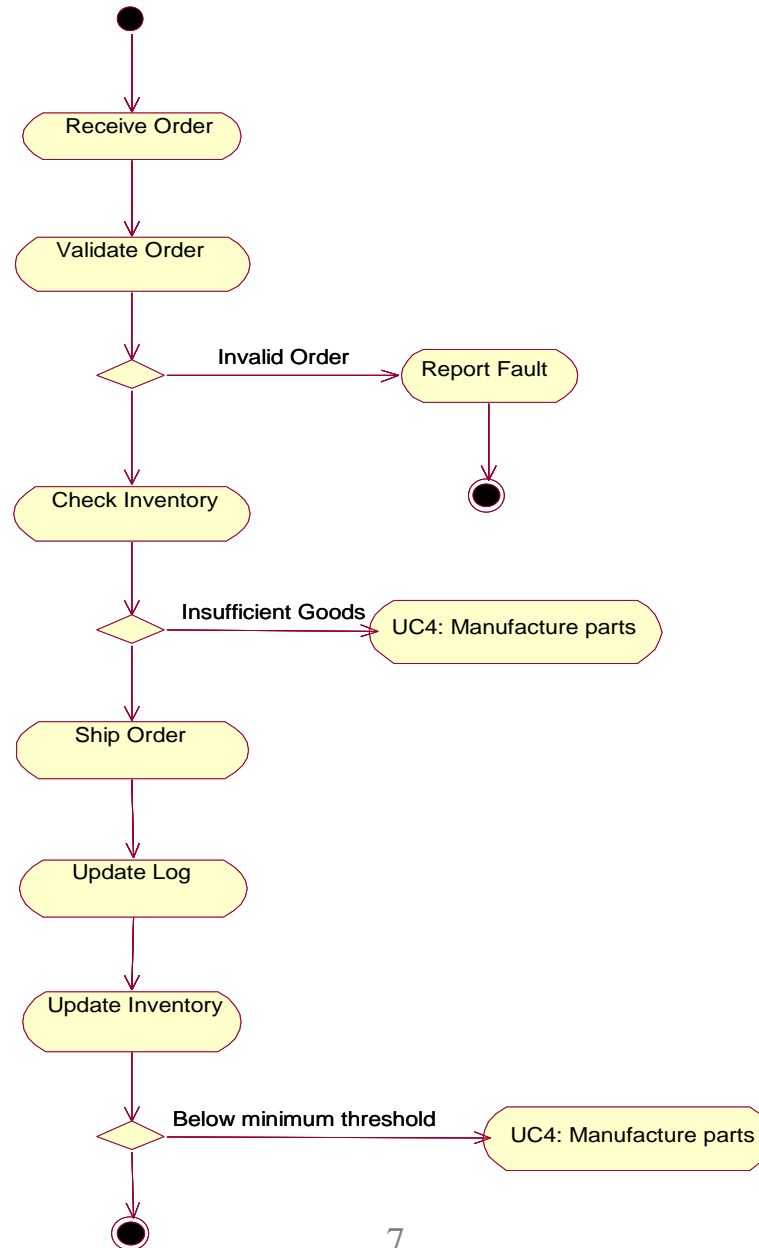
DIGITAL INTERACTIONS LAB

# UML

- class diagram: describes the static structure in a business
- use case diagram: describes the business context and shows processes in non-sequential representation
- activity diagram: describes the behavior of business workflows
- sequence diagram: descrbies the dynamic interactions between employees and the items they manipulate
- deployment diagram: shows how a system will be physically deployed in its implementation environment

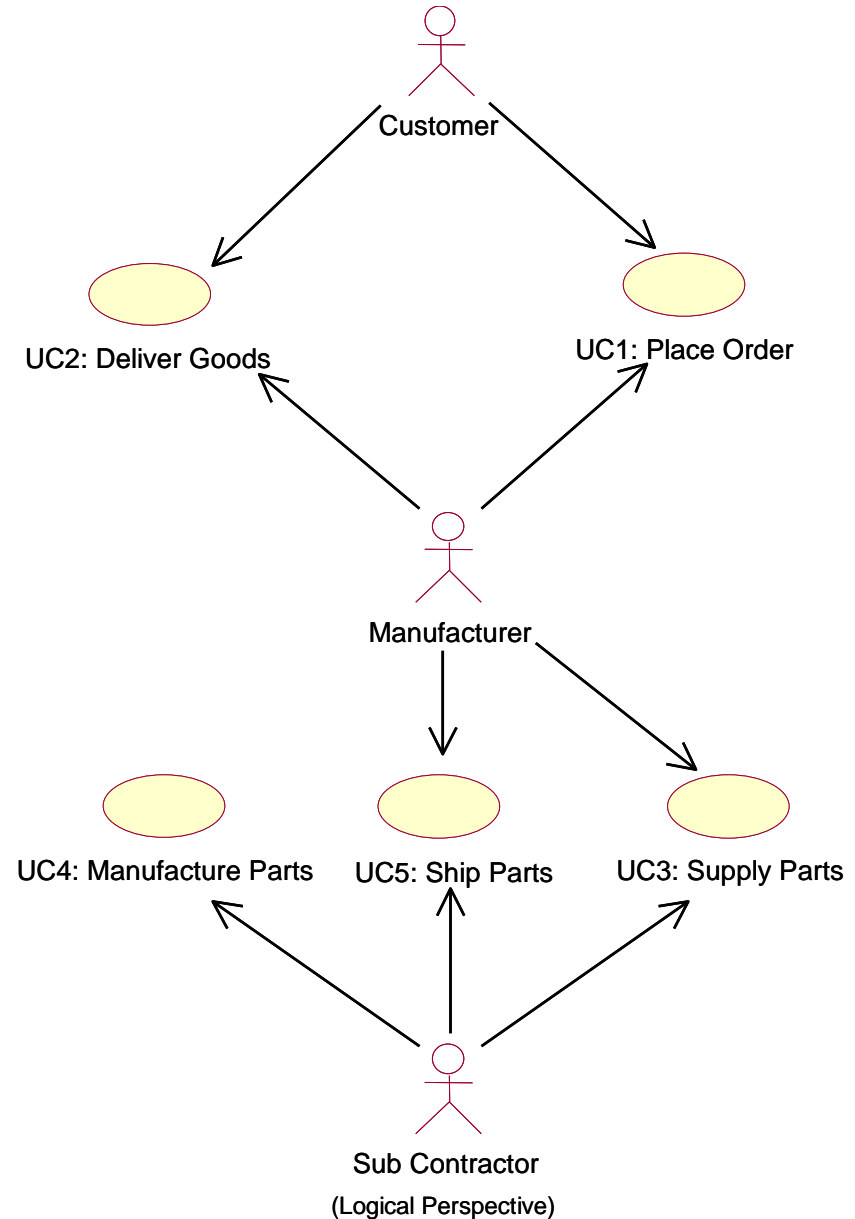# UML Class Diagram for a Computer Manufacturer
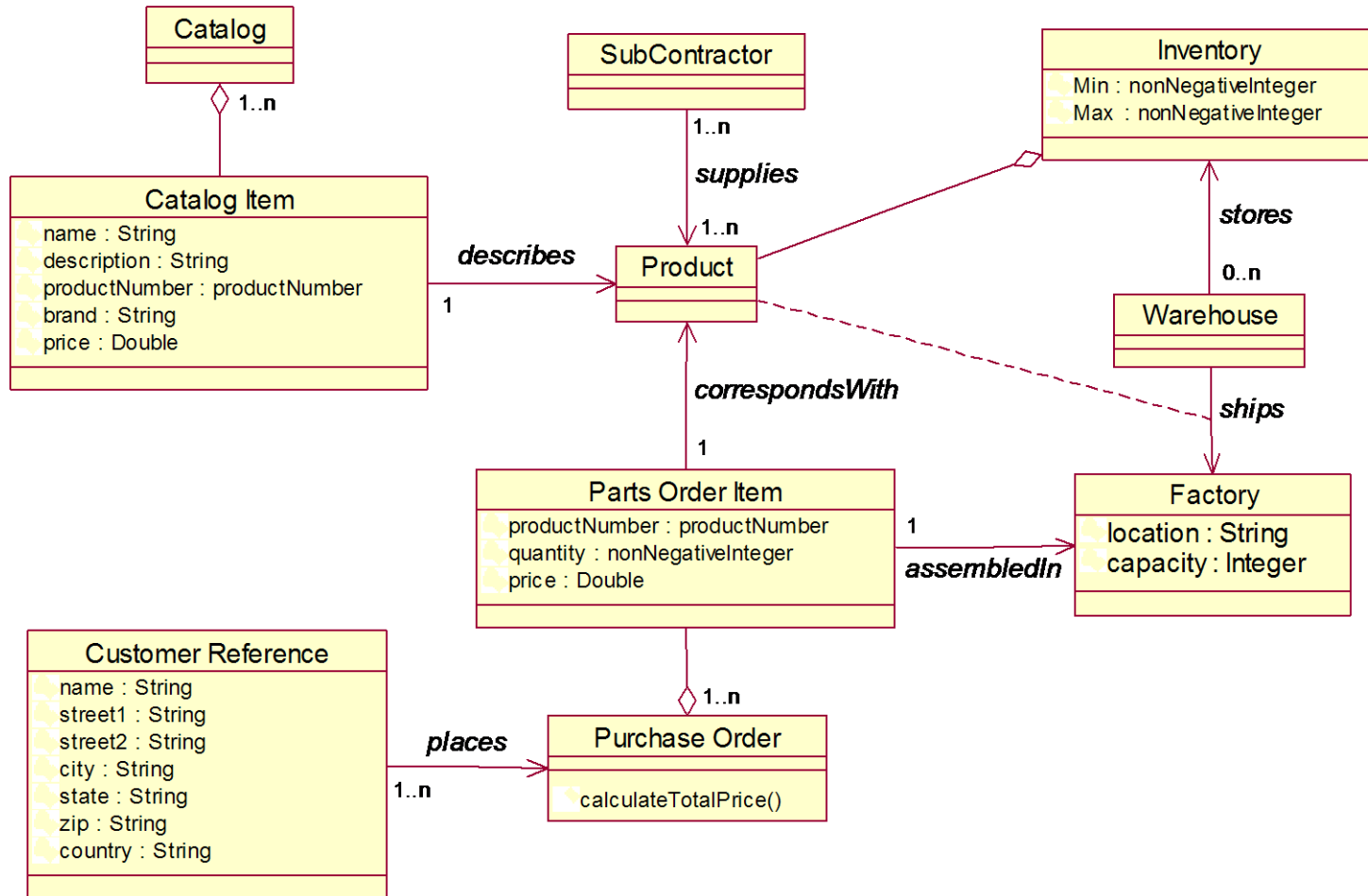


- arrow: association, diamond: aggregation, triangle: inheritance

# Activity diagram for checking inventory, shipping orders and updating the inventory
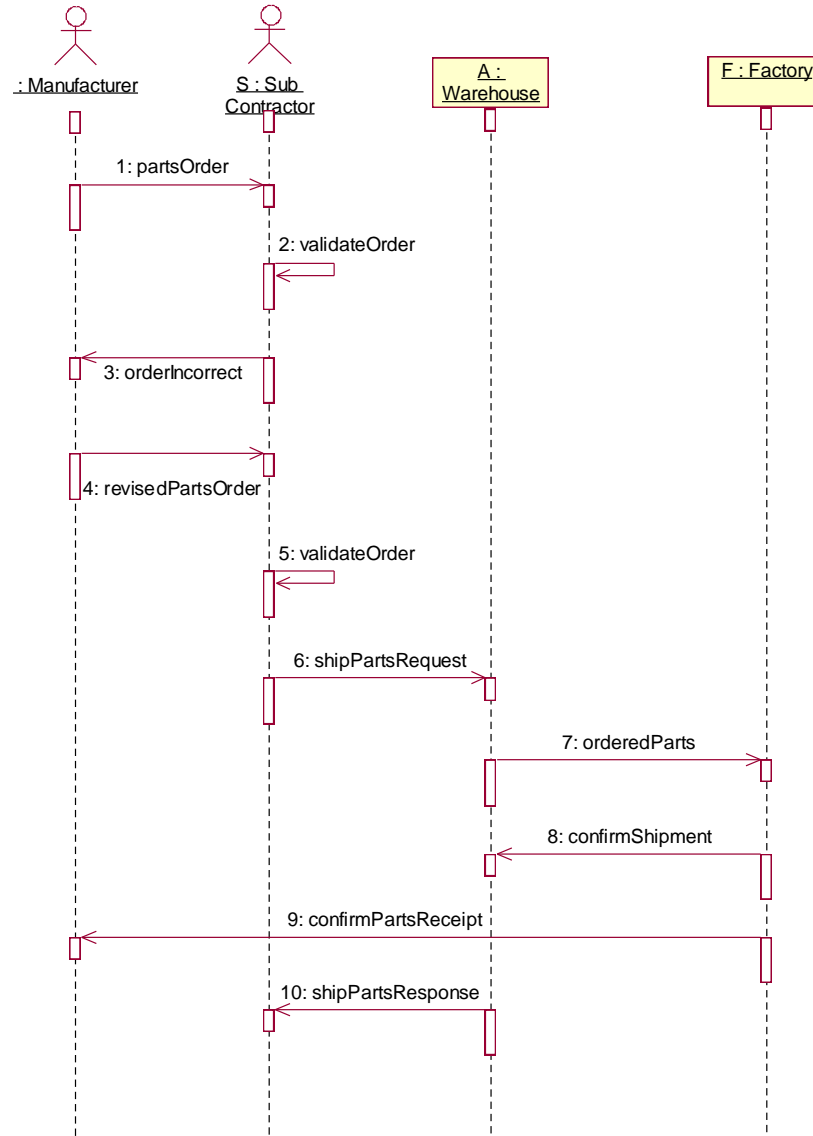
# Use case diagram of customers, manufacturers and subcontractors.



Customer

UC2: Deliver Goods

UC1: Place Order

Manufacturer

UC4: Manufacture Parts

UC5: Ship Parts

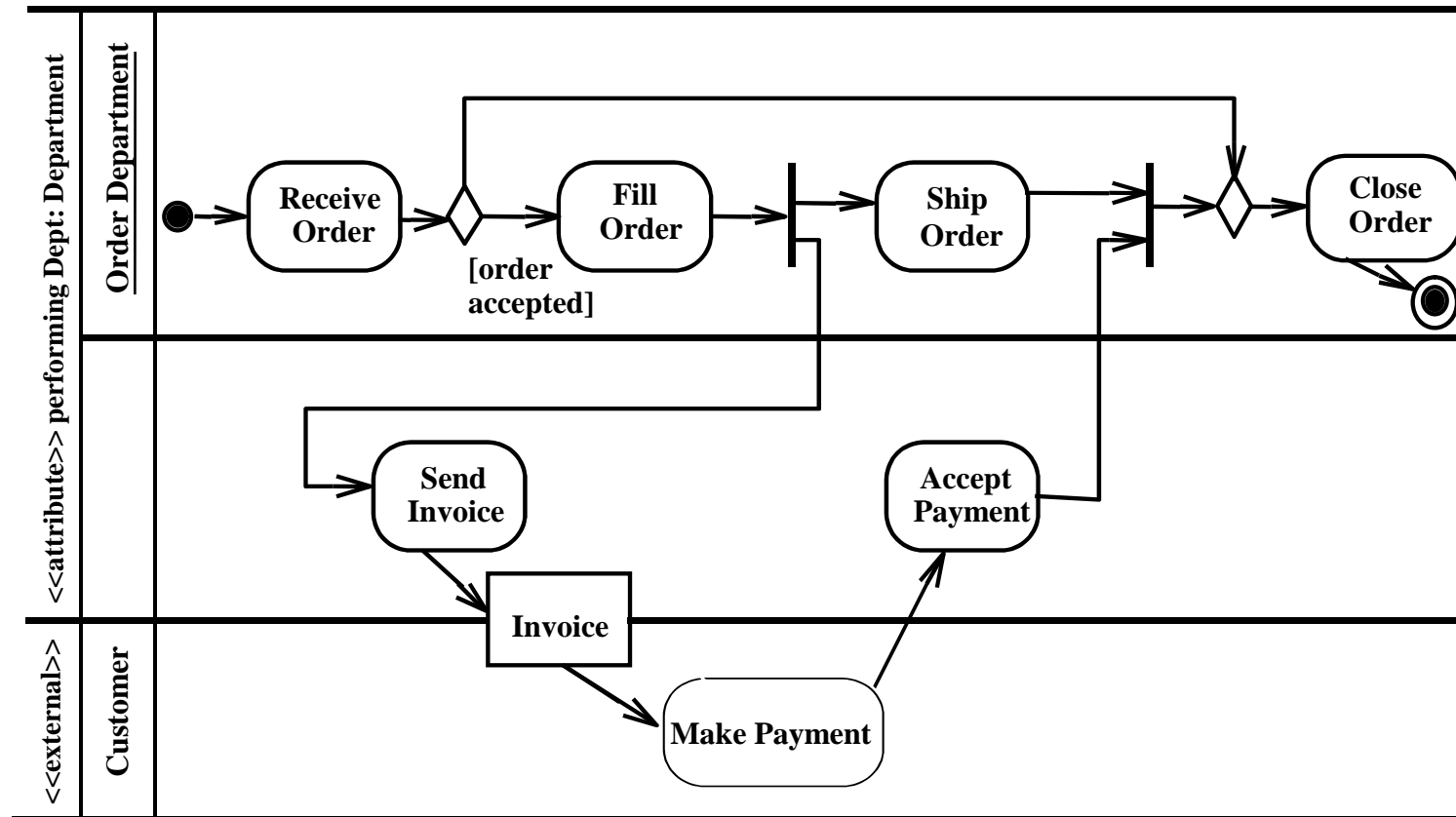UC3: Supply Parts

Sub Contractor

(Logical Perspective)

# Sequence diagram: shipment of parts between a Manufacturer and a SubContractor.

# Computer Manufacturer System Deployment Diagram
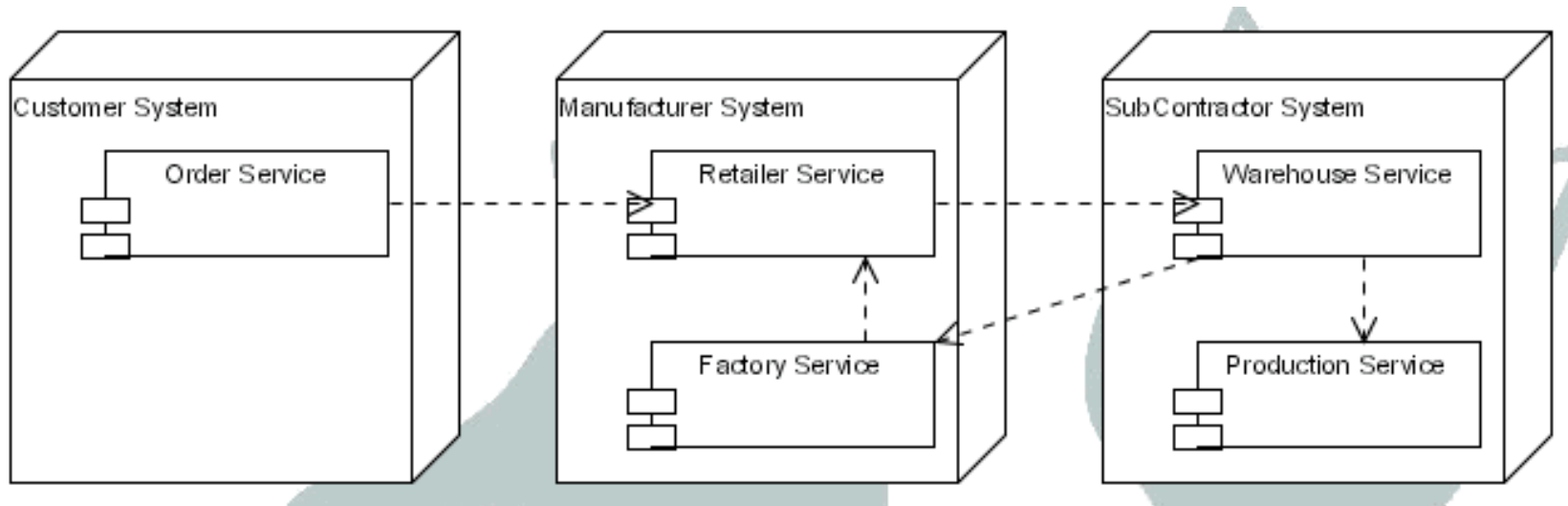
# An Activity Diagram for a Simple Order Processing Application in UML 2.0



- action node, control node, object node

# Business Process Modelling

- process-centric company: an organization whose managers conceptualize it as a set of BPs
- BP models
  - a powerful tool for structuring and formatting enterprise-related information
  - the visibility of the business rules -> makes its easier to adapt them to changing market conditions, and a shared understanding of what functions are supported by an enterprise, and how these are supported
  - allow analysts to visualize, analylze and improve processes
  - guarantee that processes will be executed in a repeatable, consistent, and preceise way

DIGITAL INTERACTIONS LAB

# Business Process Modelling Methodologies

- A business process methodology follows a series of meaningful steps to ensure consistent production and delivery of real technical implementations from high-level problem descriptions

- The methodology is **independent of the middleware platform** and **tools** used to develop the e-Business applications. The following items are necessary:

  - A **modelling language** to precisely model the business processes that e-Business application or system automates and the activities that are performed in these processes.

  - A **standardized methodology** to organize e-Business project activities needed to develop the process models.

  - **Schema languages** to model the business information that is exchanged in e-Business messaging on the basis of XML.

# The Unified Software Development Process

- The unified software development process defines a **'process framework'** for different kinds of software systems, domains, and organizations. Its general characteristics are:
  - Iterative development
  - Use-case driven
  - Requirements and change management
  - Model-based
  - Component-based
- unified SW development process life cycle
  - inception
  - elaboration
  - construction
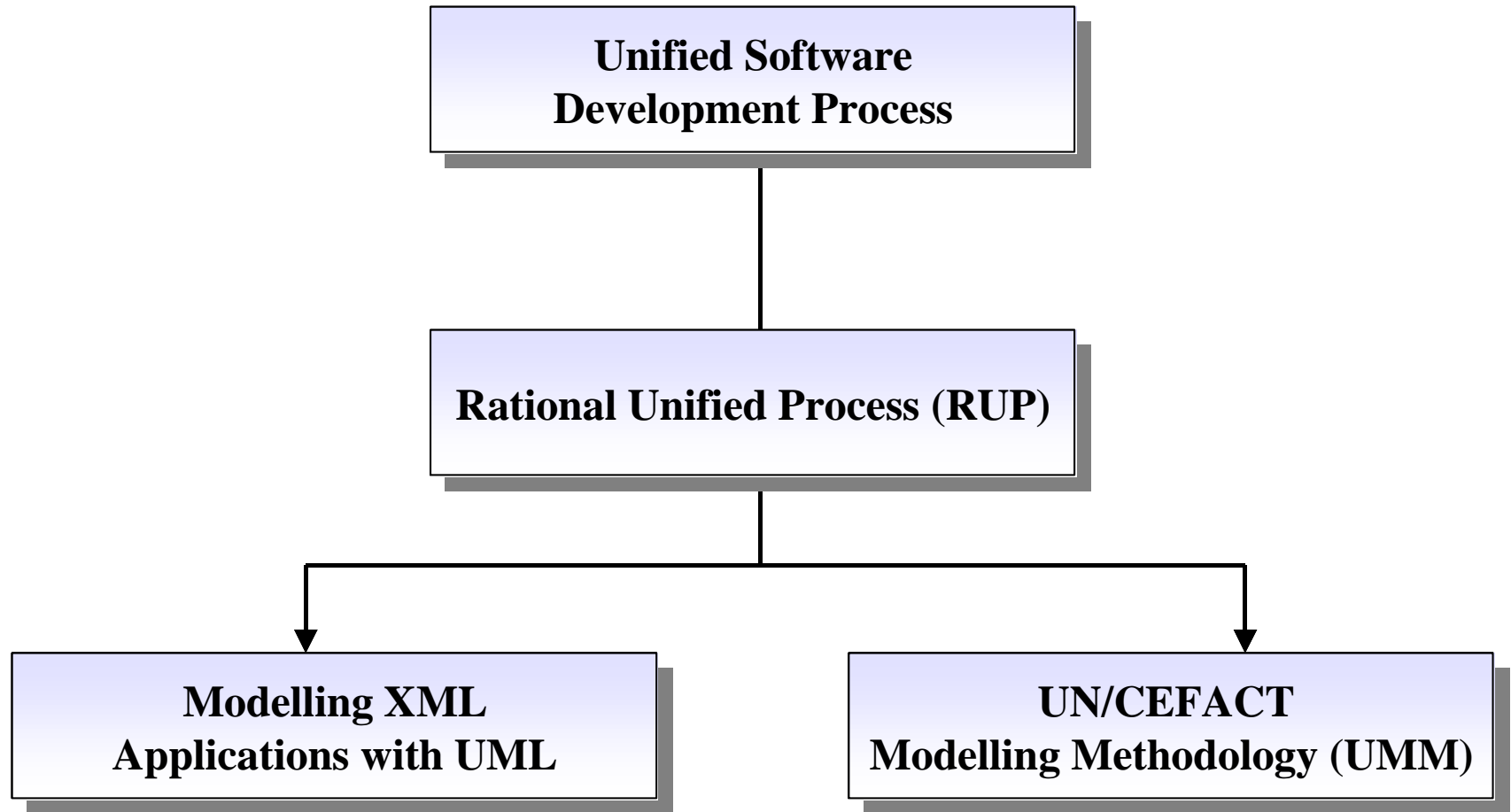  - transition

# The Rational Unified Process (RUP)

- RUP is a **configurable software development process platform** to deliver proven best practices and a configurable architecture

- based on CMM

- RUP also includes **e-business specific extensions** that provide explicit guidance in areas such as business modeling, web architectures and testing for the web

- provides extensive guidelines, templates, and examples for all critical e-development activities

- The RUP is designed and documented using the UML

- architecture has 2 dimensions

  - horizontal: represents time and shows the lifecycle aspects of the process

  - vertical: represents core process disciplines ,which logically group software engineering activities by their nature
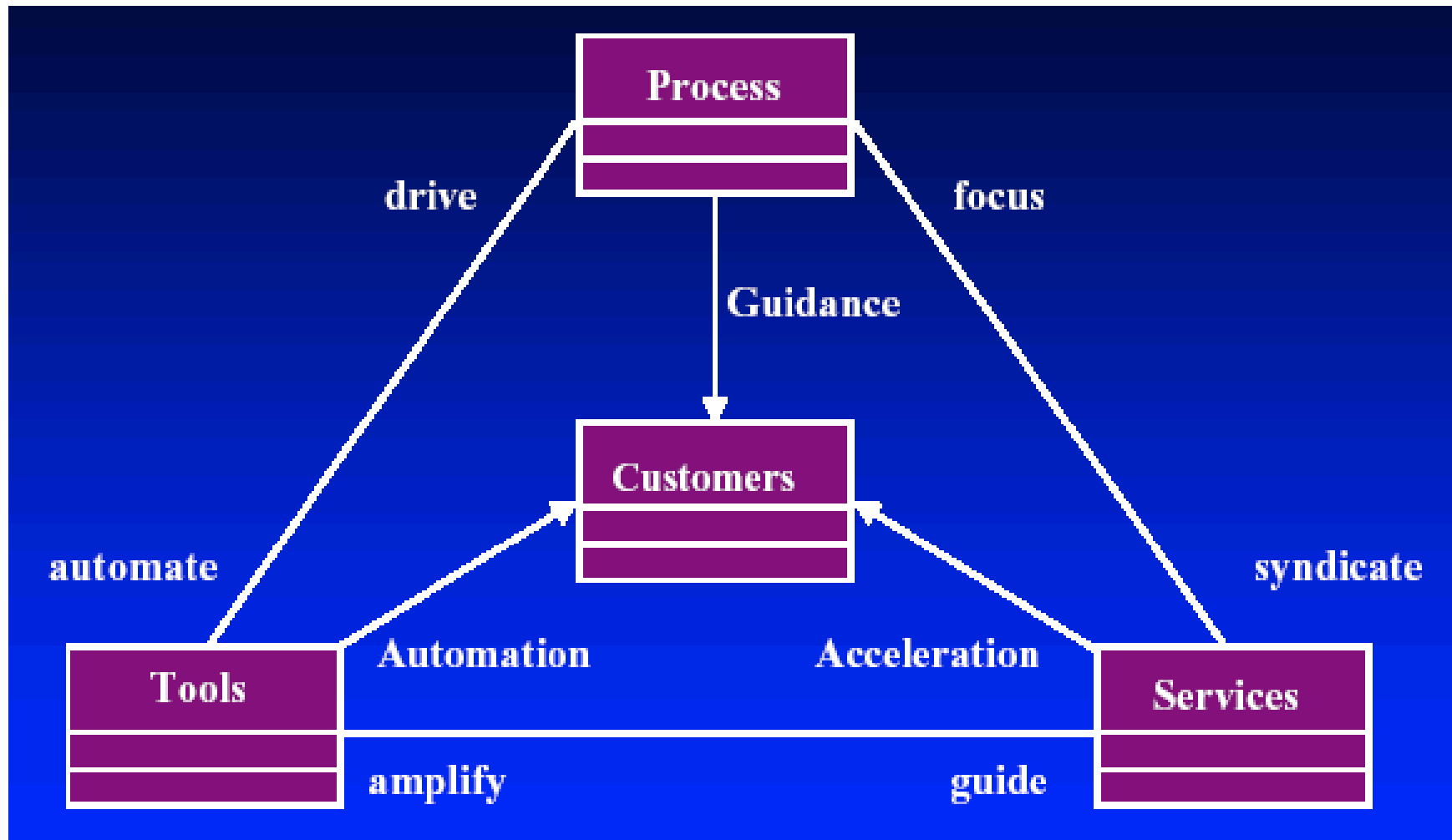
# RUP supports

- develop software iteratively
- manage requirements
- use component-based architectures
- visually model software
- verifiy software quality
- control changes to software

# Relationships Between Software Development Methodologies



Unified Software Development Process

Rational Unified Process (RUP)

Modelling XML Applications with UML

UN/CEFACT Modelling Methodology (UMM)

DIGITAL INTERACTIONS LAB

# RUP's Strategy

# Problems Addressed by RUP

- No repeatable process – results are lacking, unpredictable, and highly dependent on heroic programmers
- Software that poorly fits user needs
- Inability to deal with changing requirements
- Tedious and expensive testing procedures
- Discovery of serious flaws too late in the project
- Software that's hard to maintain and extend

# Describing processes

- The RUP uses four elements to describe processes:

  - Workers – describe a role, some people have many roles.

  - Activities – small, definable, reusable tasks that can be allocated to a single worker.

  - Artifacts – usually process deliverables, like: use cases, code, plans, test cases, test results.

  - Workflows – coordinated sequences of activities.

# UN/CEFACT modeling methodology (UMM)

- UMM has been developed as a **specialization** of a modified subset of the RUP for **e-business projects**

- UMM is an incremental business process and information construction methodology

- prescribes a specific way to perform business process and information modeling for e-business

- meta model: an explicit specification of a set of concepts and relations, which are used to define and constrain models

- UMM meta-model:  is a description of business semantics that allows  trading partners to capture the details for a specific business scenarios (a business process) using a consistent modeling methodology.

- The UMM views software engineering project as a series of sequential steps, or workflows, moving from technology independent business process modeling to technology dependent deployment.

# phases in UMM

- inception and elaboration
  - business modeling workflow
  - requirements workflow
  - analysis workflow
  - design workflow
- construction and transition
  - implementation workflow
  - test workflow
  - deployment workflow
  - configuration and change management workflow
  - environment workflow

DIGITAL INTERACTIONS LAB

# views supported by UMM metamodel

- The business domain view (BDV): partitions a business domain into business areas, process areas, and business processes.

- The business requirement view (BRV): a view of a business process model that captures the use case scenarios, inputs, outputs, constraints and system boundaries.

- The business transaction view (BTV): a view of a business model that captures the semantics of business information entities and their flow of exchange between roles as they perform business activities.

- The business service view (BSV): captures the syntax and semantics of business messages and their exchange between business services.

# The UN/CEFACT Unified Modeling Methodology
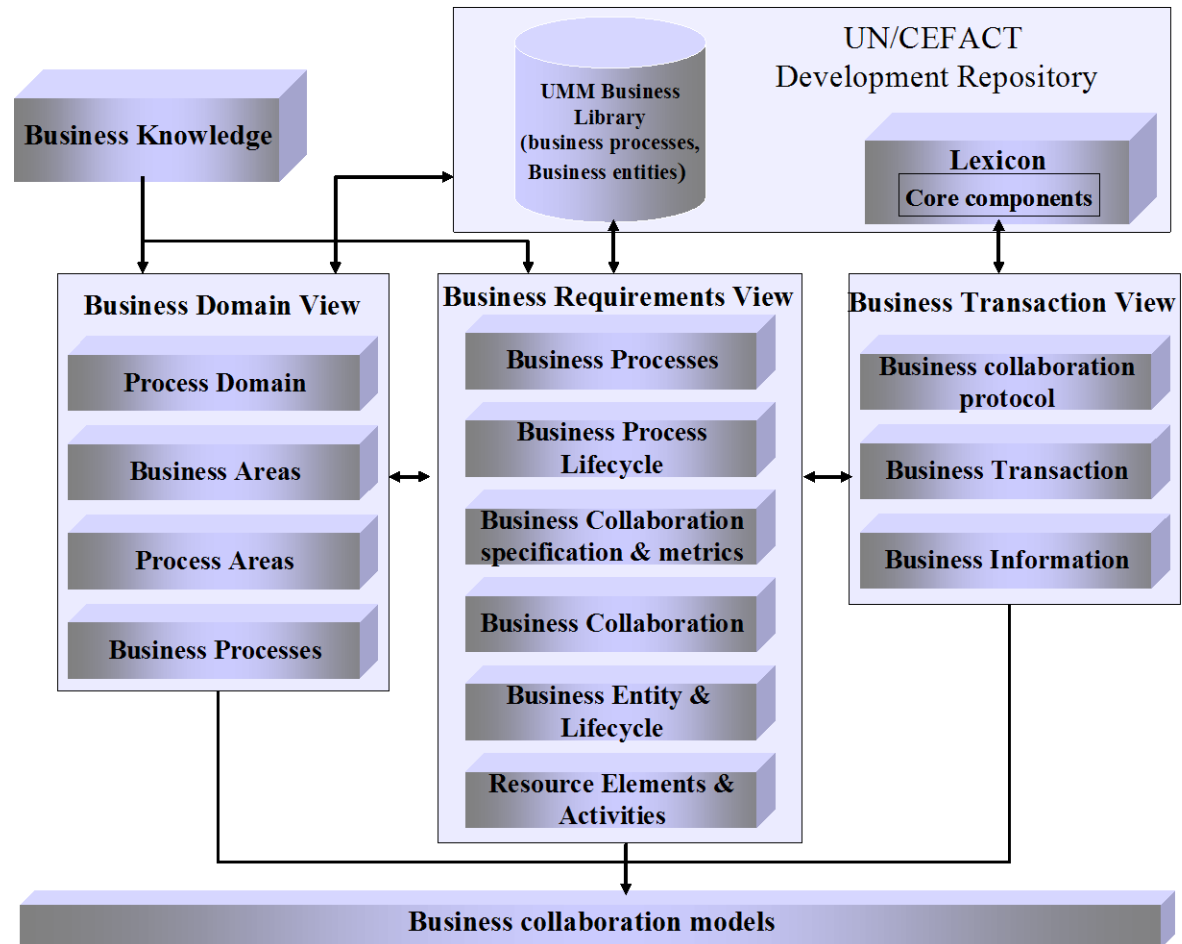
# Supply Chain Operations Reference (SCOR) model

- SCOR
  - a cross-industry, standardized, supply-chain reference model that enables companies to analyze and improve their supply-chain operations by helping them to communicate supply-chain information across the enterprise and measure performance objectively
  - gives inter-organizational SC partners a common basis for integration by providing them with a tangible framework to interact and work with
- The SCOR framework model depicts the supply-chain from a **strategic perspective**
- The core of the SCOR model
  - consists of 5 basic processes: plan, source, make, deliver, and return
  - comprises **four levels of processes** that guide supply chain members on the road to integrative process improvement
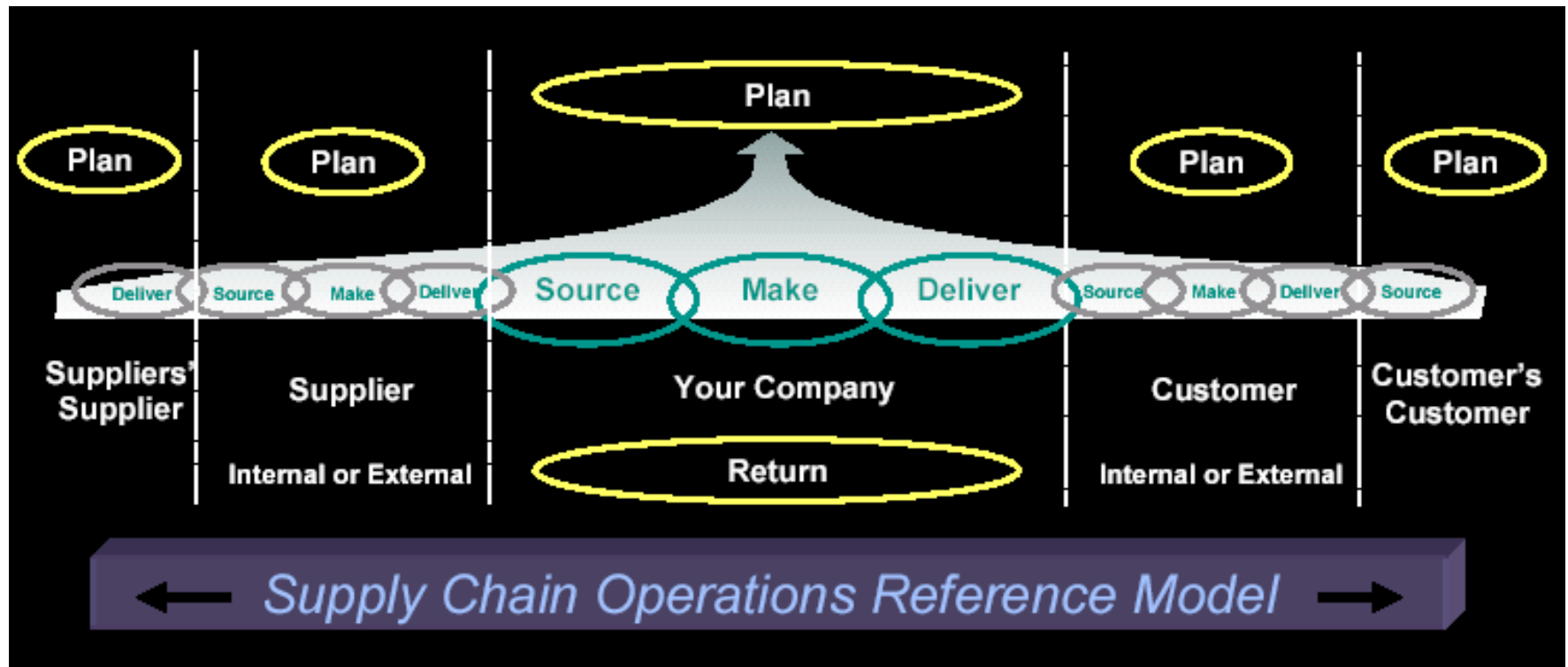- facilitates BPR, benchmarking, and best practice analysis
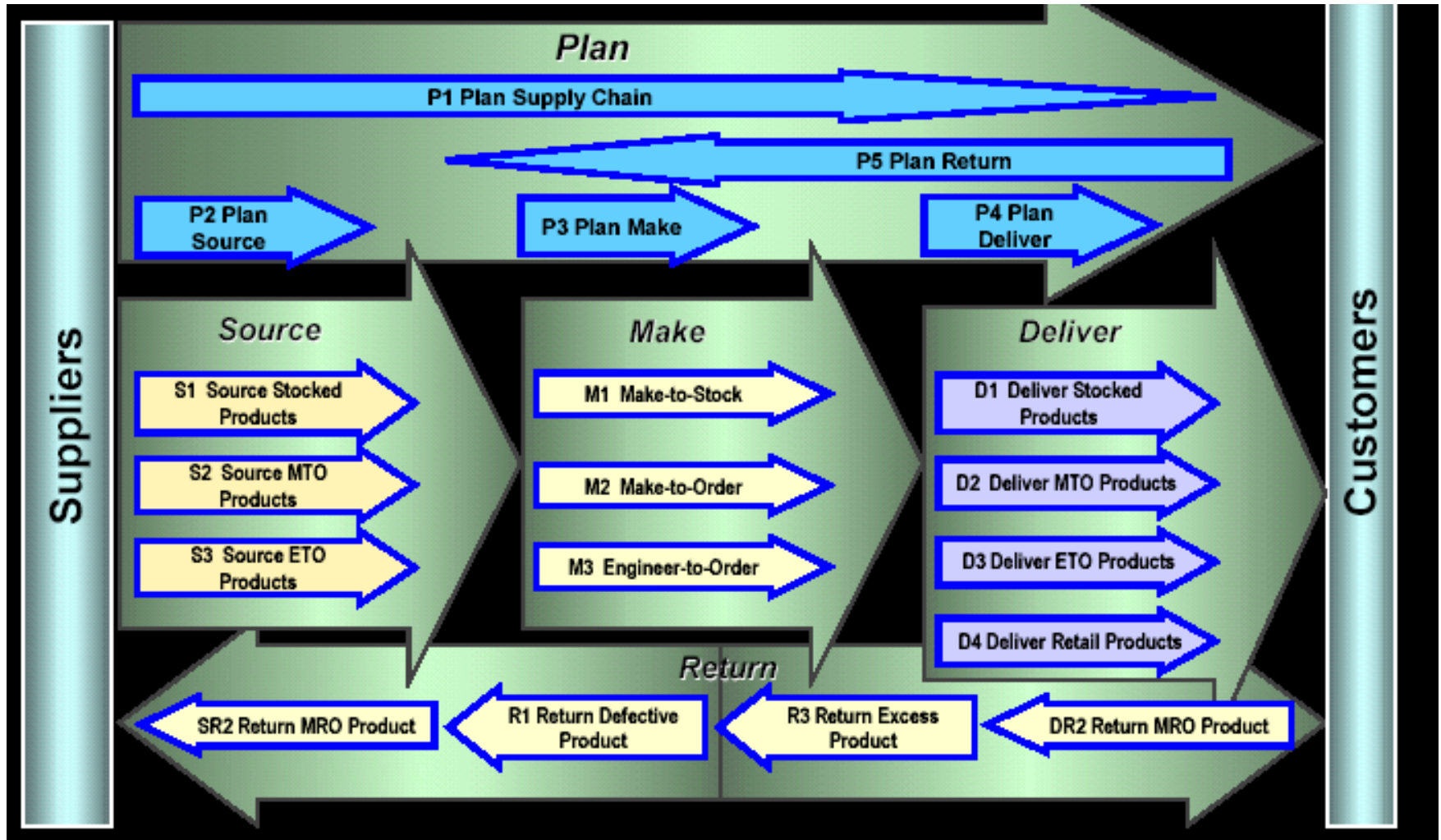
# Connecting the SCOR Process Levels
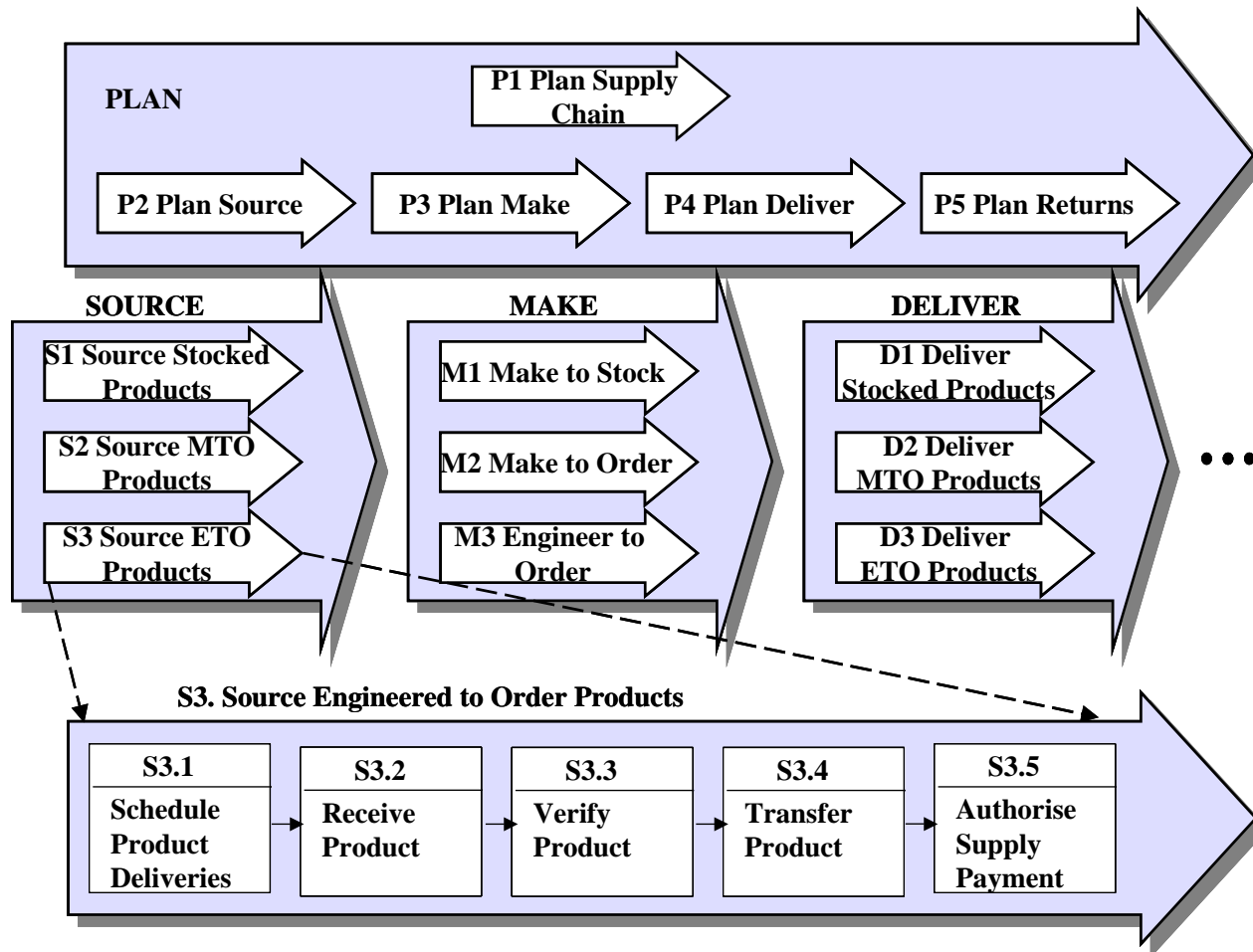
# End-to-end Business Flows

# Level 1 Processes



Supply Chain Operations Reference Model
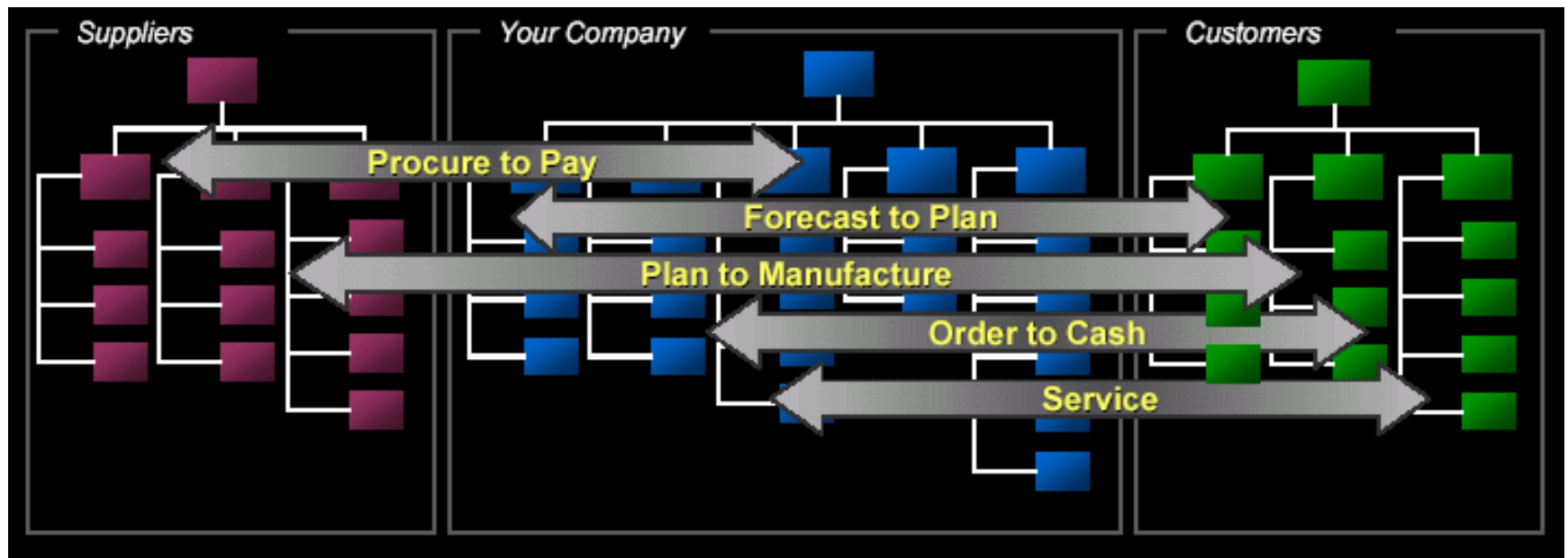
# Level 2 Processes

# Level 3 Processes

| S1 Source Stocked Product | M1 Make-to-Stock | D1 Deliver Stocked Product | R1 Return Defective Product | P2 Plan Source | ED Enable Deliver |
|---|---|---|---|---|---|
| S1.1 Schedule Product Deliveries | M1.1 Schedule Production Activities | D1.1 Process Inquiry & Quote | DR1.1 Authorize Return | P2.1 Identify, Prioritize, and Aggregate Product Requirements | ED.1 Manage Deliver Business Rules |
| S1.2 Receive Product | M1.2 Issue Product | D1.2 Receive, Enter, & Validate Order | DR1.2 Schedule Product Return | P2.2 Identify, Assess, and Aggregate Product Resources | ED.2 Assess Delivery Performance |
| S1.3 Verify Product | M1.3 Produce and Test | D1.3 Reserve Inventory and Determine Delivery Date | DR1.3 Receive Defective Product | P2.3 Balance Product Resources with Product Requirements | ED.3 Manage Deliver Information |
| S1.4 Transfer Product | M1.4 Package | D1.4 Consolidate Orders | SR1.4 Verify Defective Product | P2.4 Establish Sourcing Plans | ED.4 Manage Finished Product Inventories |
| S1.5 Authorize Supplier Payment | M1.5 Stage Product | D1.5 Plan and Build Loads | SR1.5 Disposition Defective Product | | ED.5 Manage Deliver Capital Assets |
| | M1.6 Release Product to Deliver | D1.6 Route Shipments | SR1.6 Return Replacement or Credit | | ED.6 Manage Transportation |
| | | D1.7 Select Carriers and Rate | | | ED.7 Manage |

DIGITAL INTERACTIONS LAB

# Supply Chain Best Practices

## Order Management

- Support multi-channel order capture, including complex configurations
- Improve order accuracy and visibility
- Reduce order processing time and costs

## Supply Chain Planning

- Improve forecast accuracy
- Synchronize value chain operations/increase velocity
- Minimize inventory costs while improving service levels

## Procurement

- Lower procurement transaction costs
- Lower total purchasing spend
- Increase quality of goods/ services purchased

## Manufacturing

- Reduce cost while improving quality
- Reduce cycle times
- Enable manufacturing agility: make vs. outsource

## Logistics

- Minimize cost of movement and storage
- Improve speed & accuracy of delivery commitments
- Reduce shipping errors

## Service

- Deploy lowest cost, best fit service channel
- Decrease issue resolution time
- Grow revenues via differentiated services

## Maintenance Mgmt

- Minimize asset downtime
- Reduce maintenance costs
- Improve safety and compliance

## Product Lifecycle Mgmt

- Design more competitive products
- Reduce product cost
- Accelerate time-to-market

## Supply Chain Intelligence

- Measure and monitor key performance indicators in real-time
- Compare results, view trends
- Identify exceptions and opportunities for continuous process improvement

# SCOR as a process reference model

- standard descriptions of management practices
- a framework of relationships among the standard processes
- standard metrics to measure process performance
- management practices that produce best-in-class performance
- standard alignment to features and functionality

# phases defined in SCOR methodology

- define the supply-chain process
- determine the performance of the existing supply chain
- establish supply-chain strategy, goals, and priorities
- redesign supply chain as needed
- enable the redesign and implement

# Business Process Modelling Notation

- BPMN is an attempt at a **standards based business process modelling language** that can **unambiguously** define business logic and information requirements to the extent that the resultant models are **executable**

- Business logic and information requirements are maintained in a model that is consistent with and reflects the actual EIS supporting business activities

- BPMN contains notations and semantics for capturing workflowsor sequences of activities, decision points and prerequisites, information transformation and flows, collaborations among multiple entities and actors

- based on BPML from BPMI

# business process diagram (BPD)

- defined as a series of activities connected by sequence flow, where the direction of the sequence flow arrowheads determines the order of the sequence
- is made up of
  - flow objects: events, activities, and gateways (forking, merging, joining)
  - connecting objects: sequence flows, message flows, associations
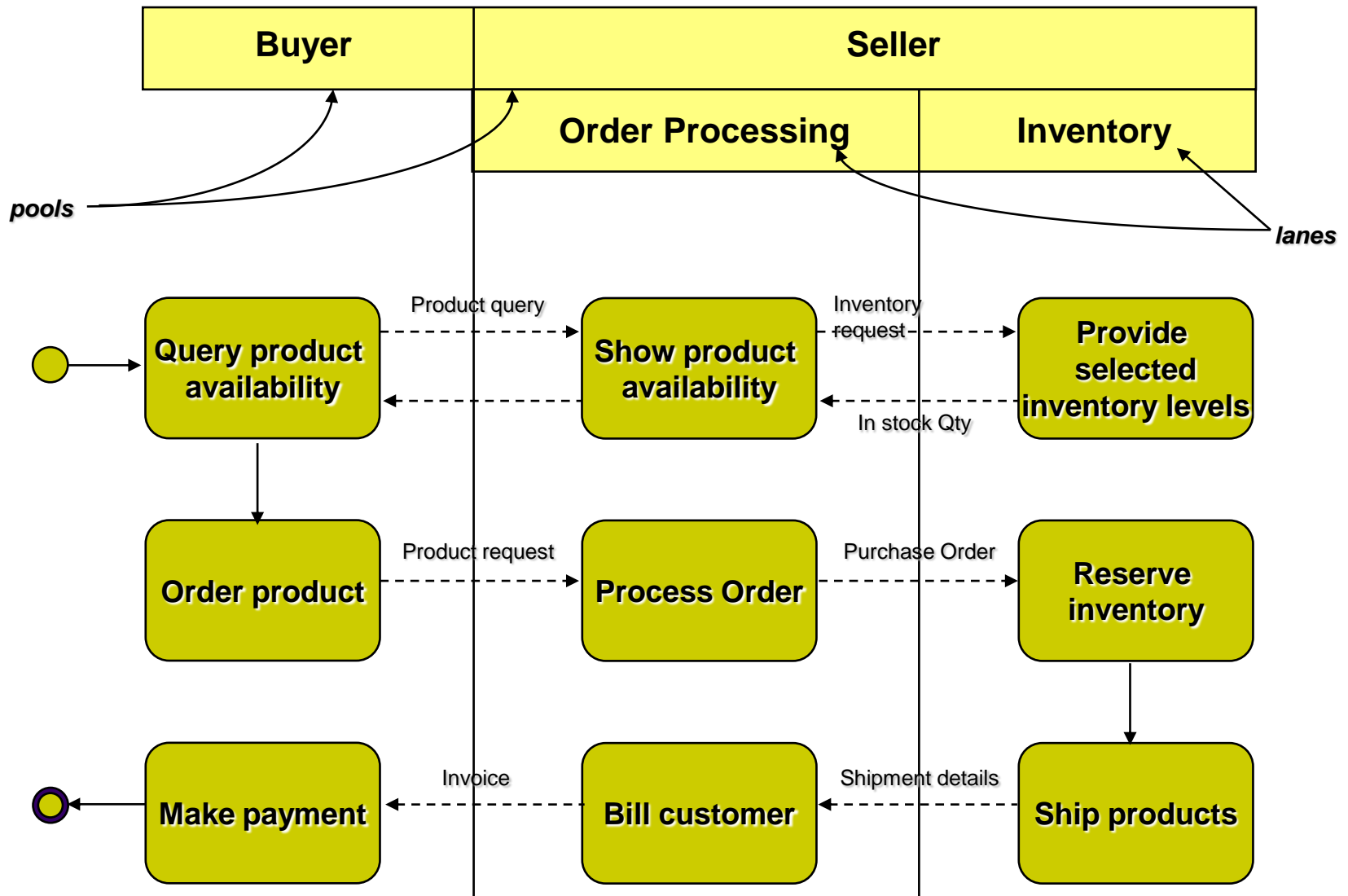  - swimlanes and artifacts

# BPMN example of a high-level business process
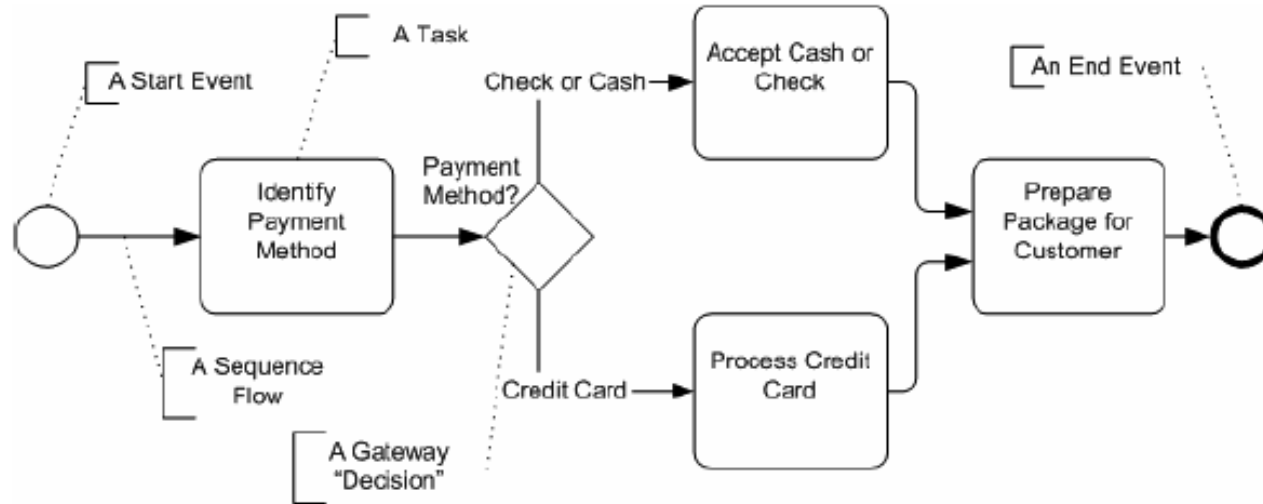


# BPMN example of a business process at high-level of precision

DIGITAL INTERACTIONS LAB
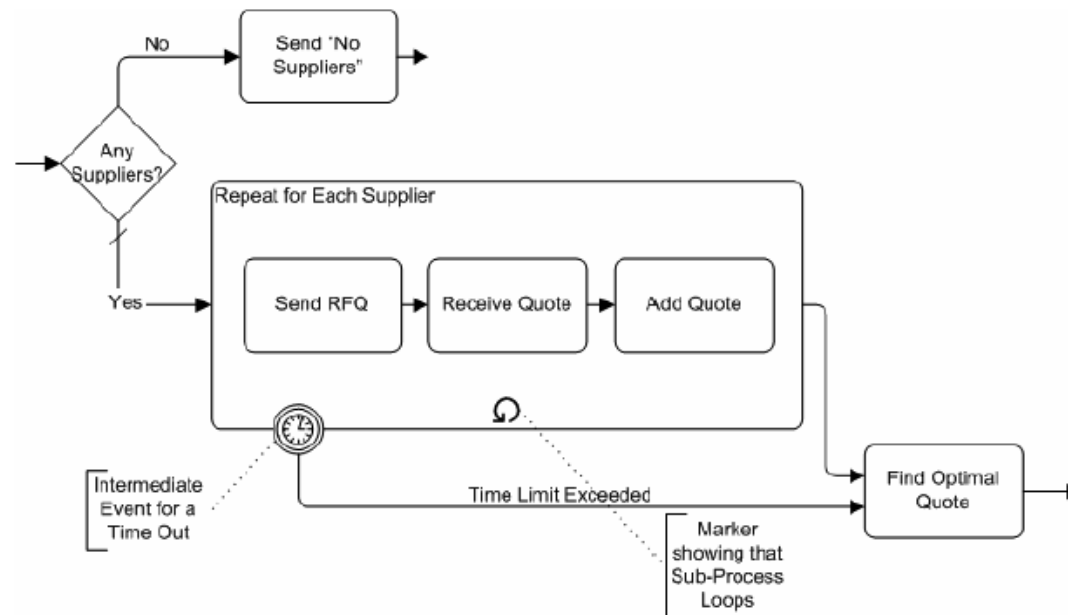
# BPMN example including two pools and two lanes

DIGITAL INTERACTIONS LAB

# Meta-models

- A meta-model is a description of a modelling language
  - A model of models
  - It defines all the concepts that can be used within a language
  - A model is constructed using the syntax defined by a meta-model
- Meta-modelling is the process of describing new modelling languages
  - Analogous to the process of defining new programming languages using Backus Naur Form (BNF).
- Requries a meta-language, a language for defining new languages
  - The BNF is an example of a meta-language – it can express a wide range of textual language.

DIGITAL INTERACTIONS LAB

# MDA (Model-Driven Architecture)

- Proposed by OMG (Object Management Group)
- To enable conceptual models that are independent of platforms and thus to make for easier interoperation of heterogeneous systems and easier portability and evolution of such systems as the underlying platforms evolve
- The lifecycle includes capturing business requirements, modeling and design, implementation and testing, configuration and deployment, management, and evolution induced by changing requirements or changing technology
- Based on three modeling capabilities
  - UML (Unified Modeling Language)
  - MOF (Meta Object Facility): enables the exchange of models among development tools and middleware
  - CWM (Common Warehouse Metamodel): standardizes the data warehouse application lifecycle

# Model Driven Architecture (MDA)

- motivated by integration and interoperability concerns at the enterprise scale and beyond

- **separates** the fundamental logic behind a specification from the specifics of the particular middleware that implements it

- allows **rapid development and delivery of new interoperability specifications** that use new deployment technologies but are based on proven, tested business models

- utilizes models and a generalized idea of architecture standards to address integration of enterprise systems in the face of evolving heterogeneous technology and business domains
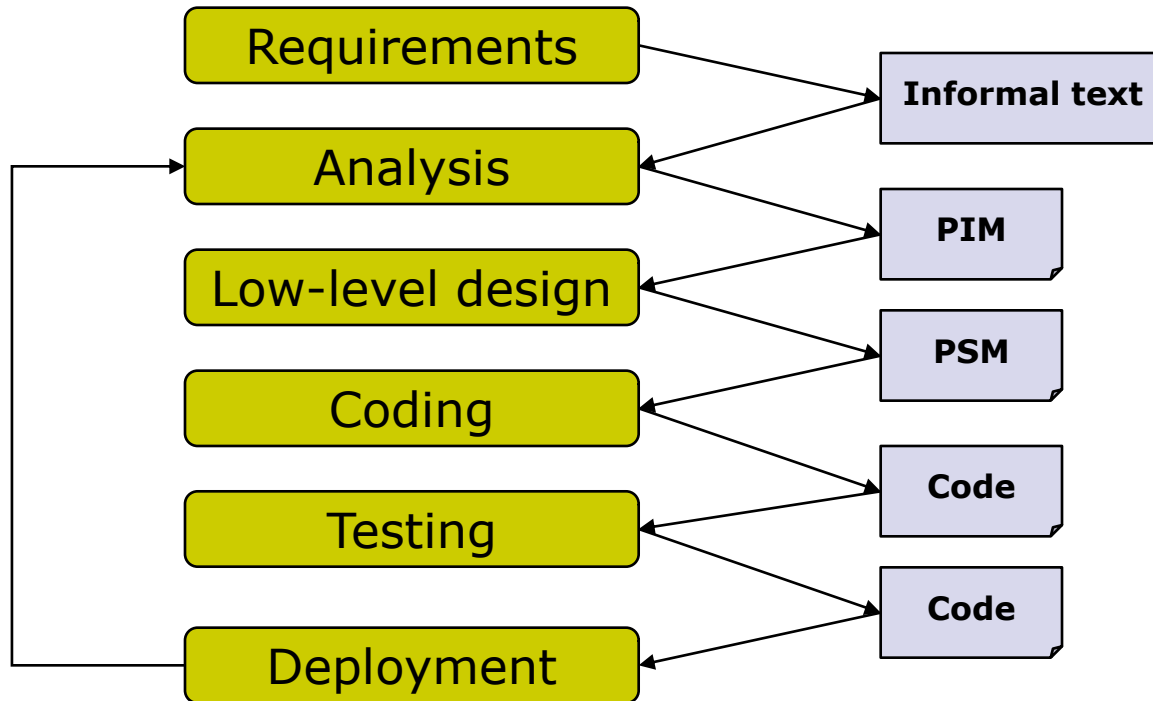
# characteristics of MDA

- portability
- cross-platform interoperability
- platform independence
- domain specificity (through domain specific models)
- productivity

DIGITAL INTERACTIONS LAB

# components of MDA

- XML metadata interchange (XMI)
  - a mapping mechanism, which expresses UML models in XML schemas, providing an XML serialization mechanism
- meta-object facility (MOF)
  - provides the standard modeling and interchange constructs that are used in the MDA
- common warehouse metamodel (CWM)
  - establishes data warehouse industry standard

# Model driven architecture

- The purpose of the MDA is to facilitate a software development process that is driven by modelling, rather than coding
- MDA involves the development of an initial Platform-Independent Model (PIM) and successive refinement through Platform-Specific Models to eventual code implementation
- At its simplest, the MDA software development lifecycle is as follows:

| Requirements | → | Informal text |
| Analysis | ← | |
| | | PIM |
| Low-level design | ← | |
| | | PSM |
| Coding | ← | |
| | | Code |
| Testing | ← | |
| | | Code |
| Deployment | ← | |

PIM is used to define an abstract specification of system – refined or transformed into a PSM that includes platform specific details – the PSM is closer to implementation code, so it is more easily implemented.

# MDA Transformations



Abstract views
(platform independent,
Written in UML2)

*Traditional mapping*

Abstract system architecture

Domain-specific
Models
(Platform specific,
Written in UML2 with
extra notation)

COM+ enterprise architecture

Safety critical architecture

Embedded System architecture

Implementation

OO: simple

.NET: COM+, ASP

CORBA

C