

OCD I

Modeling

CS577a

Fall 2001

Modeling

Why Model?

- What makes computers useful?

Can faithfully represent a conceptual system in a particular context outside of real time/space.

That is....

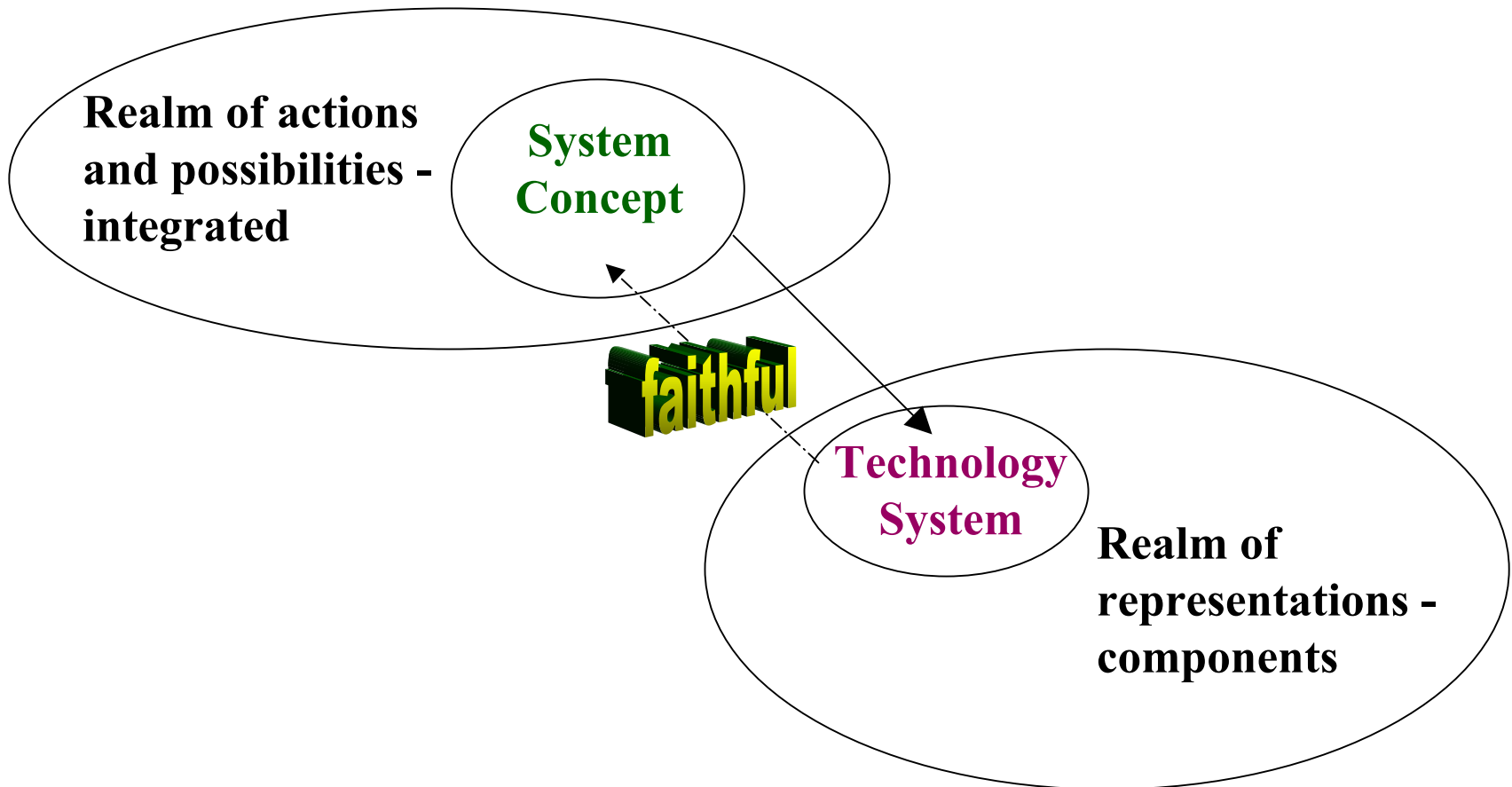
Computers support software models.

Software implementations are representations (models) of real-world conceptual systems.

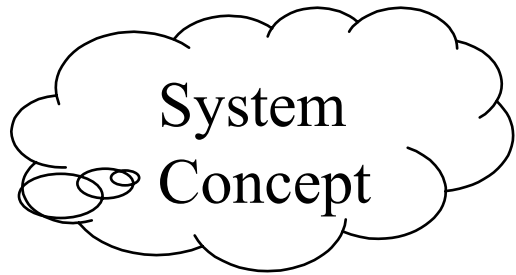
Realms

- Systems start as people conceived ideas
- The task of development is to represent concepts with technology
- Development engineering processes move a concept from the *Realm of Actions* (concepts) to the *Realm of Representations* (technology)

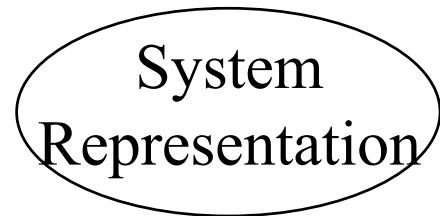
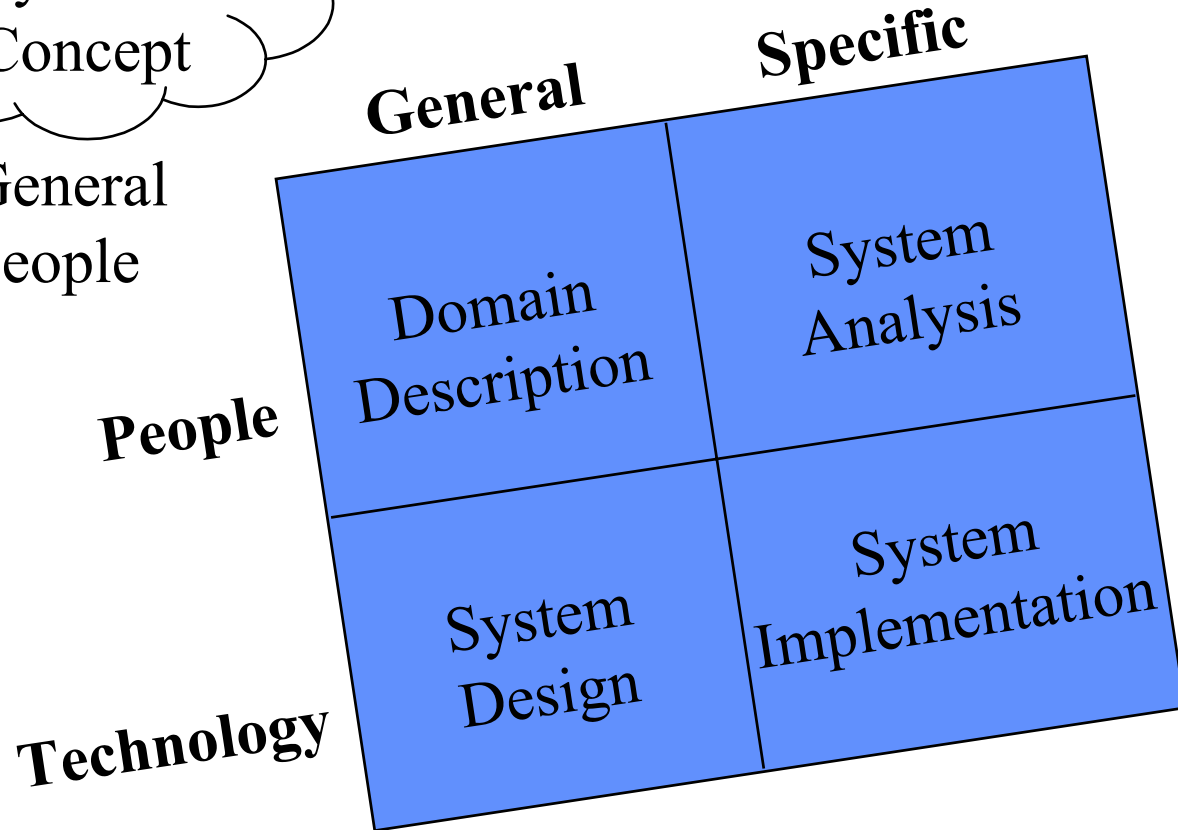
Realm of Actions to Realm of Representations



Concept to representation gap (product models)



- General
- People



- Technology
- Specific

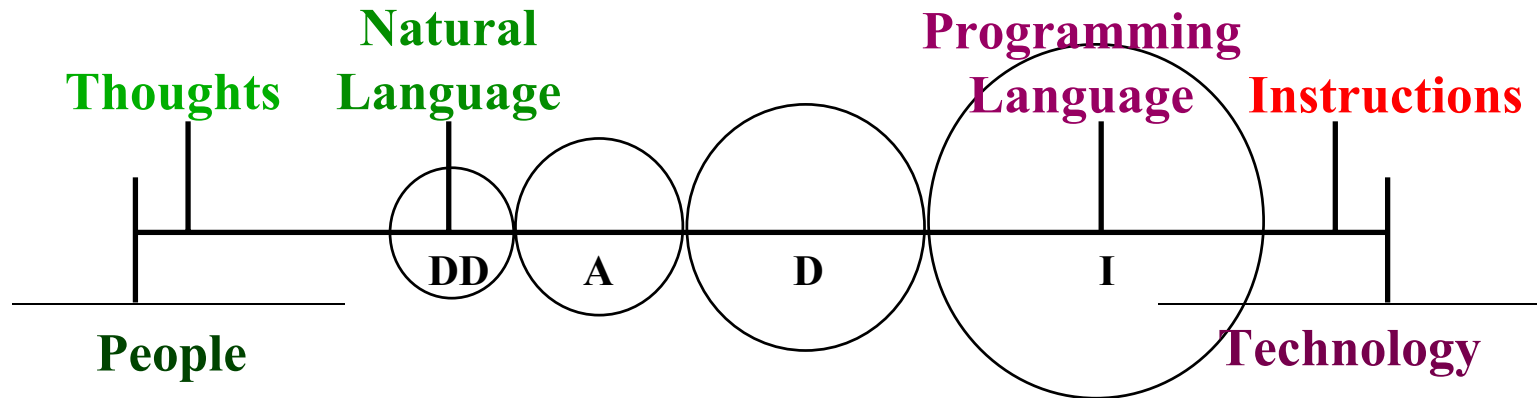
Computer Based Models

- Computers can be used to build models
- Computer models can
 - faithfully represent a conceptual system in a particular context outside of real time/space.
 - support software models.
 - support software implementations that model (real-world) conceptual systems.

People Based Models

- People
 - also build models
 - use language to abstract everyday experiences
 - model systems differently than technology
- A good methodology tries to resolve these differences between people and computers

People-Technology Gap



- Development must build bridges from people to technology
- These layers translate between various audiences and languages
- Subsequent stages increase information content and complexity

People vs. Computers

People

- Non-linear
- Abstract
- Continuous
- Context Sensitive
- Active
- Creative
- Inconsistent
- Need Donuts

Computers

- Linear
- Concrete
- Discrete
- Context Free
- Passive
- Logical
- Consistent
- Need Batteries

People vs. Computers (cont.)

People

- Semantic
- Highly Parallel (small tasks)
- Approximate (PAC, PACE)
- Learn
- Do as they want
- Make Choices
- Flexible (usually)
- Desires power
- Informal

Computers

- Syntactic
- Sequential
- Exact
- Represent
- Do as they are told
- Compute
- Rigid
- Needs power
- Formal

Model Audiences

- Different project audiences view a system concept in different ways (non-technical vs. technical)
- The final project's constituents usually do not want disruptive revolution
- The software engineer should try to evolve new structures from the existing successes within the project domain and users

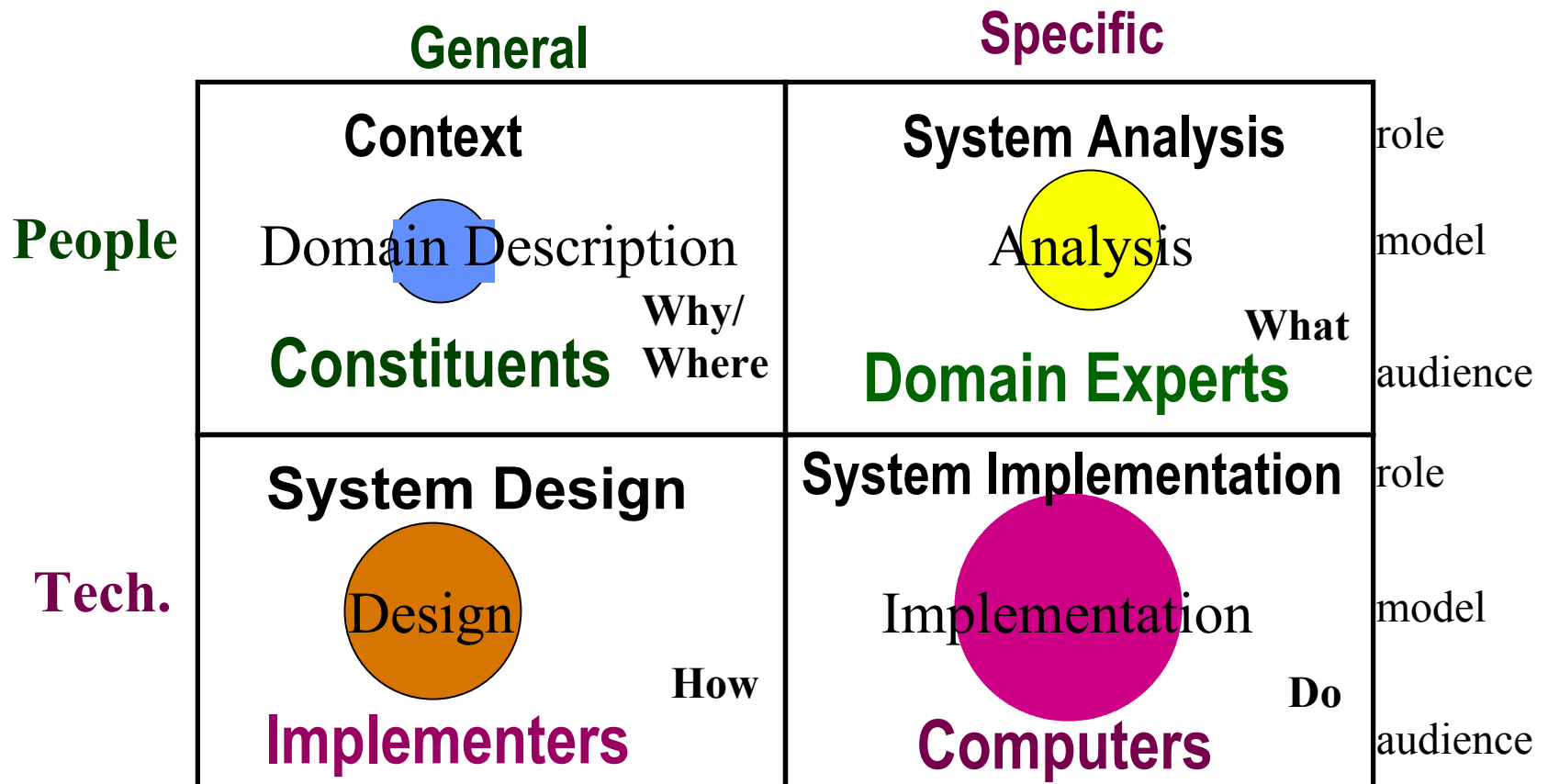
Project Roles

- 4 different types of people/audiences involved in a project

	Conceptual	Practical
User	Customers and Domain Experts	Analysts
Creator	Designers	Implementers

Audience Matrix

Each role works with a particular model and audience

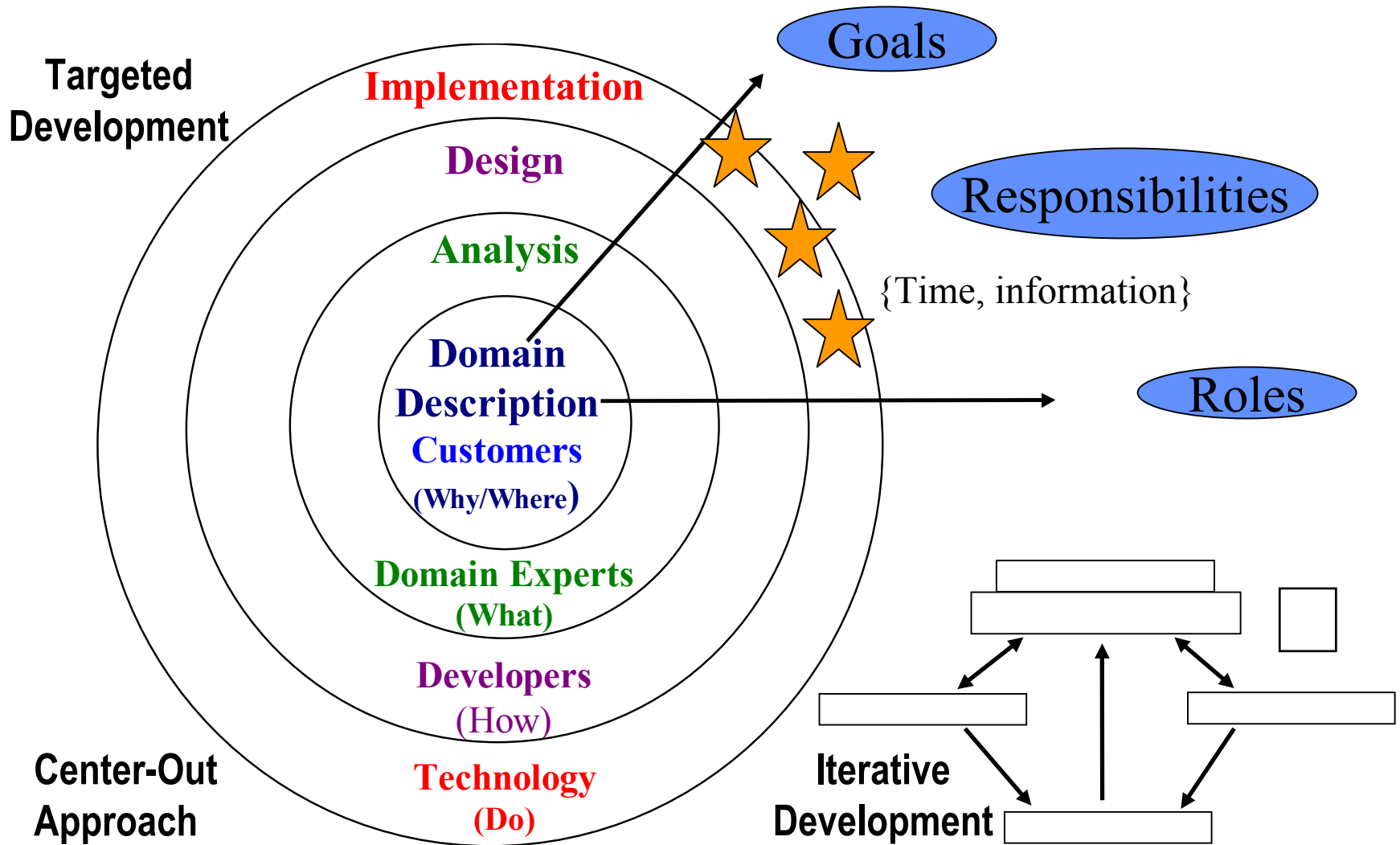


Model Layers

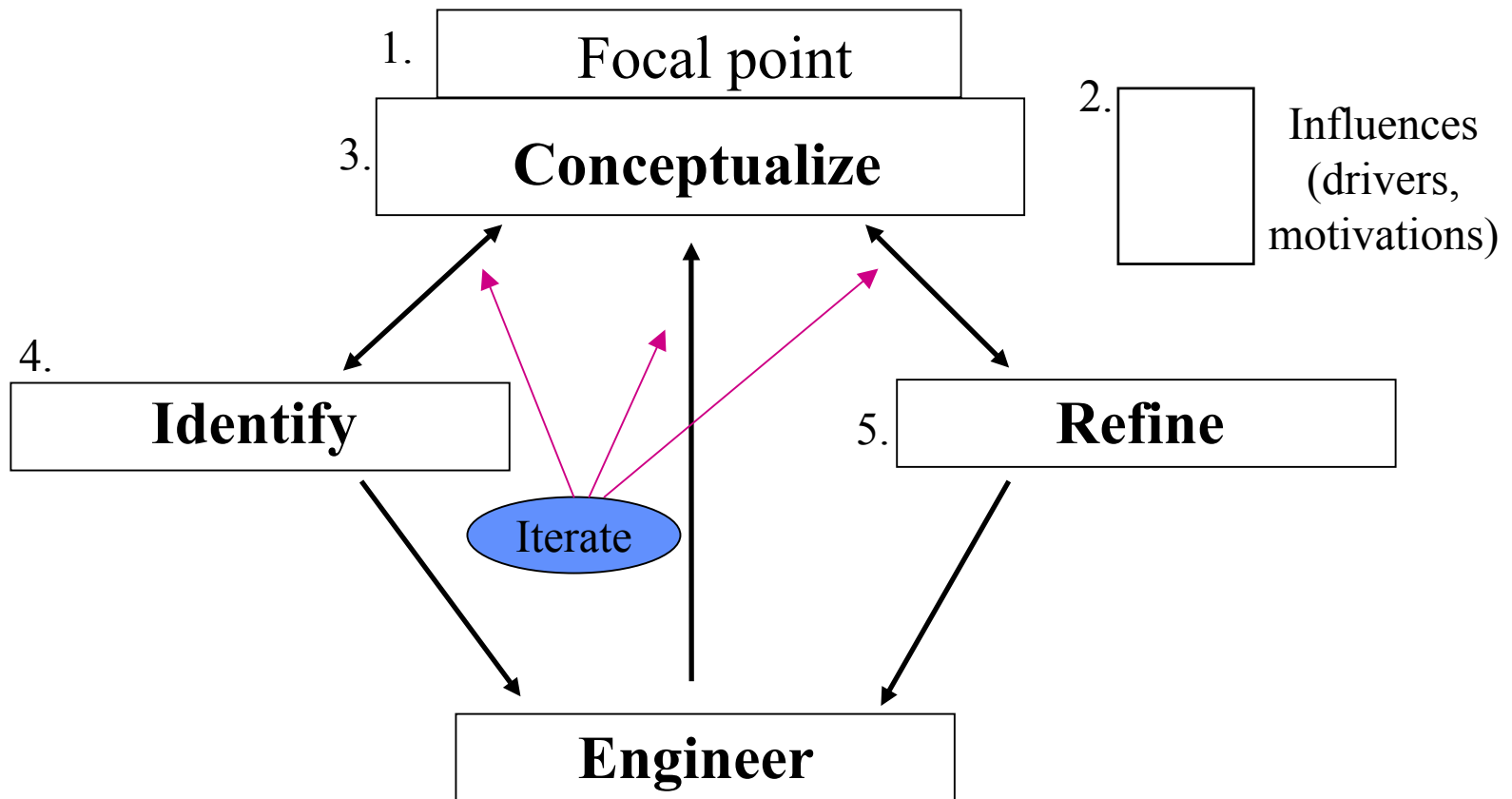
Product Model Layers

- MBASE builds product models in stages
- Each stage refines models from previous stages
- This approach provides openings for a project to change details without driving the project off-track
- Models allow us to hide complexity and focus on key aspects

General Center-Out Development



Iterative Development



Model Clashes: Faithfulness

- **Consistency:**
 - (I) A constraint set forth by one model element does not violate a constraint elsewhere (e.g. inter-model clashes such Waterfall and IKIWISI)
 - (II) The same name is never used to denote two different model elements (e.g. domain-entities and system-components, [internal and external] increments)
 - (III) Two different names are never used to refer to the same model element (e.g increment and build)
- **Soundness:** all elements (including the product itself) deduced from a model satisfy all models
 - **Ex. All requirements must be consistently represented in the software**
 - **Ex. Requirements are eventually functionally decomposed into operations (operations can only be carried out by objects)**
 - **Ex. Gantt/Pert charts must represent the process strategy**
- **Coverage:** all model elements map to other model or non-model elements
 - **Ex. All operations must eventually be mapped to objects**
 - **Ex. All level of service requirements must have a verification activity**

Why Faithful Systems

- Faithfulness - A good software system should reflect the systems original intent and the context in which the system was conceived
- No matter how complex, an unfaithful system can destabilize a project
- Faithful systems are intuitive, predictable, and may be more robust and scalable

Faithfulness

- Is the key to building good, reliable systems
- Acknowledges that computer structures are different than the concepts they represent
- Defines the semantics for the final implemented structure via the original concept it represents
- May not always be achievable, but MBASE tries to achieve faithfulness at least locally

Modeling Framework

- Models help manage complexity and reflect the real world
- Without a proper framework, model clashes can cripple a development effort
- Many modeling techniques exist, but to be effective, it is essential to manage communication between various modeling stages of large software development efforts

Modeling versus Documenting

- MBASE describes models, how to build them, integrate them, and ways to communicate them.
- Documentation is a necessary consequence of modeling
 - Why? What are models... how are they used...
 - Important observation: documenting solidifies a model
- Communication is an essential part of the collaborative development approach
- Avoid documenting for documentation's sake!
 - Everything you document should be the result of modeling activity
 - Everything you document should have value and meaning to stakeholders (think about risk here)

The models are the documentation!!!

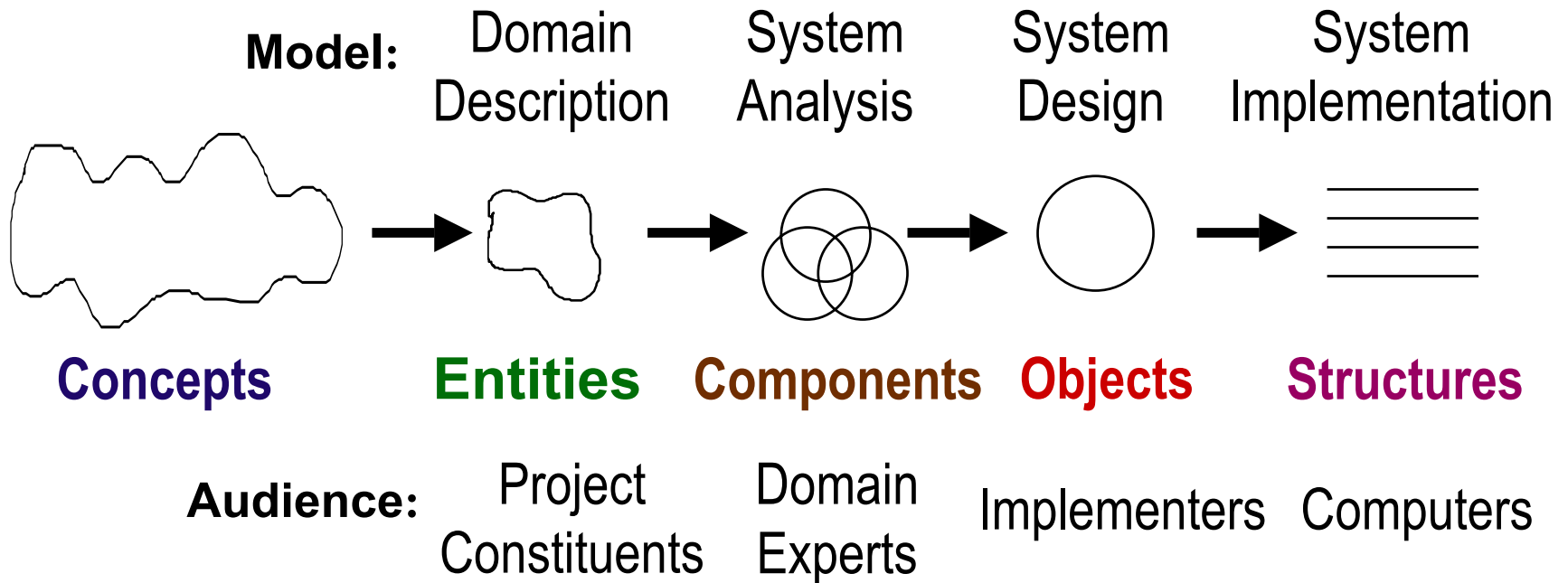
The documents are the models!!!

Evolution

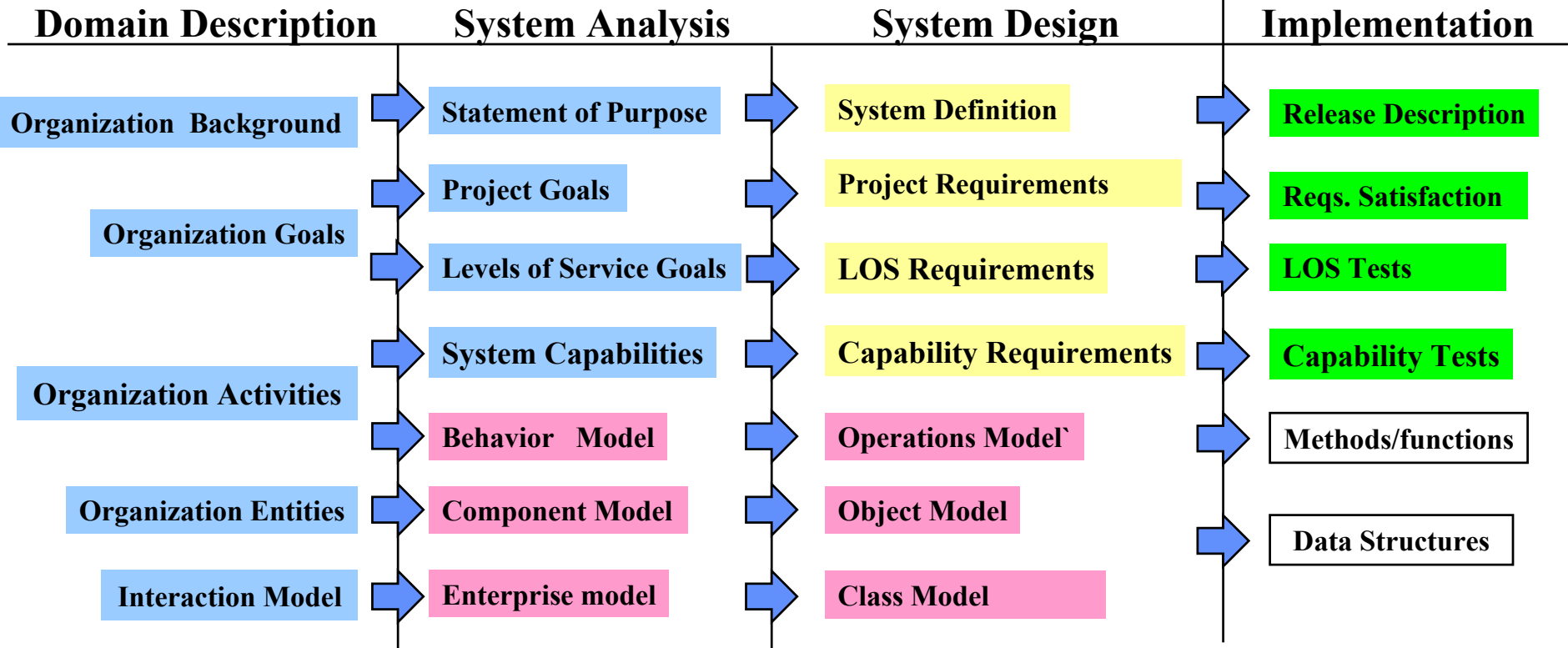
Evolutionary Modeling

- Different types of audiences move a software project from conception to implementation (customer to software)
- Breaking the evolution into natural phases helps manage complexity
- Avoiding jumps to implementation can free up a project to have a more effective overall architecture

OOAD Evolution



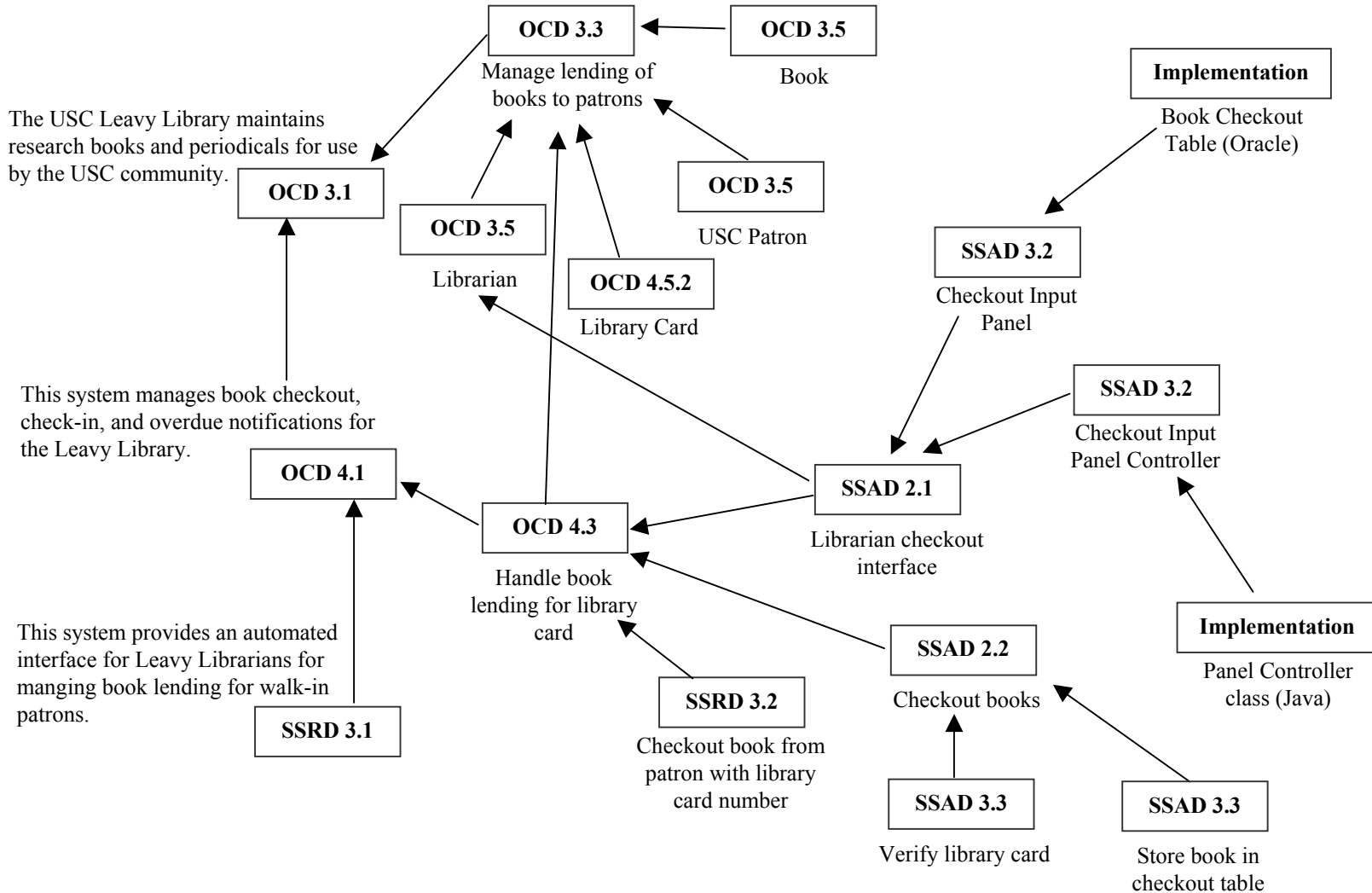
Coverage/Traceability of MBASE Product Models*



- Operational Concept Description (OCD)
- System and Software Requirements Definition (SSRD)
- System and Software Architecture Description (SSAD)
- Construction, Transition, Support (CTS)
- External to MBASE

* Does not include all MBASE models

Example Trace Map



The OCD Models

Purpose of OCD

- **Describe the overall context of the system** to be developed, why it's being built, what exists now, and where the project is starting from
- **Describe to the stakeholders** of the system to be developed (“developed” is meant to include such terms as “enhanced”, “updated”, “re-engineered”, "automated"), how the system will work in practice once it is deployed
- **Enable the operational stakeholders to evolve** knowledgeably from their current operational concept to the new operational concept, and to collaboratively adapt the operational concept as developments arise, to make clear the value of developing the new system
- **Establish goals and other success criteria**, establish basis of value assessment (for use in FRD Business Case)
- **Prepare for evolution:** how to transition from current to new operational concept, what future changes may be needed.

OCD High-Level Dependencies

WinWin Negotiations Give:

- System Capabilities
- Project Goals and Constraints
- Levels of Service
- Terms (CDL) for the domain description

OCD Yields:

- Project, System and Level of service Reqs for SSRD
- Domain Description and Initial Analysis for SSAD
- Stakeholder and Organizational Responsibilities for LCP
- Business Case Analysis parameters for FRD

OCD Completion Criteria

- LCO
 - Top- level system objectives and scope
 - Operational concept
 - Shared vision and context for stakeholders
- LCA
 - Elaboration of system objectives and scope
 - Critical scenarios described
 - Complete Traceability between capabilities, activities, project goals, levels of service and organization goals.

OCD Modeling Resources

- Electronic Process Guide (EPG)
- MBASE Deliverables Guidelines
- Active Template for OCD
- Best Practices
 - Examples from previous projects

Domain Description

Domain Descriptions

- Specify and justify:
 - **Why** the system is being built
 - **What** overall organization goals and activities the project will support and be responsible for when deployed
 - **Where** the project is starting from
 - What is there already available to build from, and what else is missing and needed.
 - What context do we build in and for

Context Setting

- Methodologies help a project meet a list of pre-formed desires
- A domain description helps keep these desires in harmony with actual activities and goals of the organization
- The domain description captures subtle “context sensitive” notions that are invisible to “context free” technology

Domain Description

- Serves as the starting point of a project
- Uses the organization's goals to help describe the entities and operations within the current organization relevant to the project
- Sets the boundary for what is relevant and what is not relevant
- Helps with later decision making as more information and details emerge

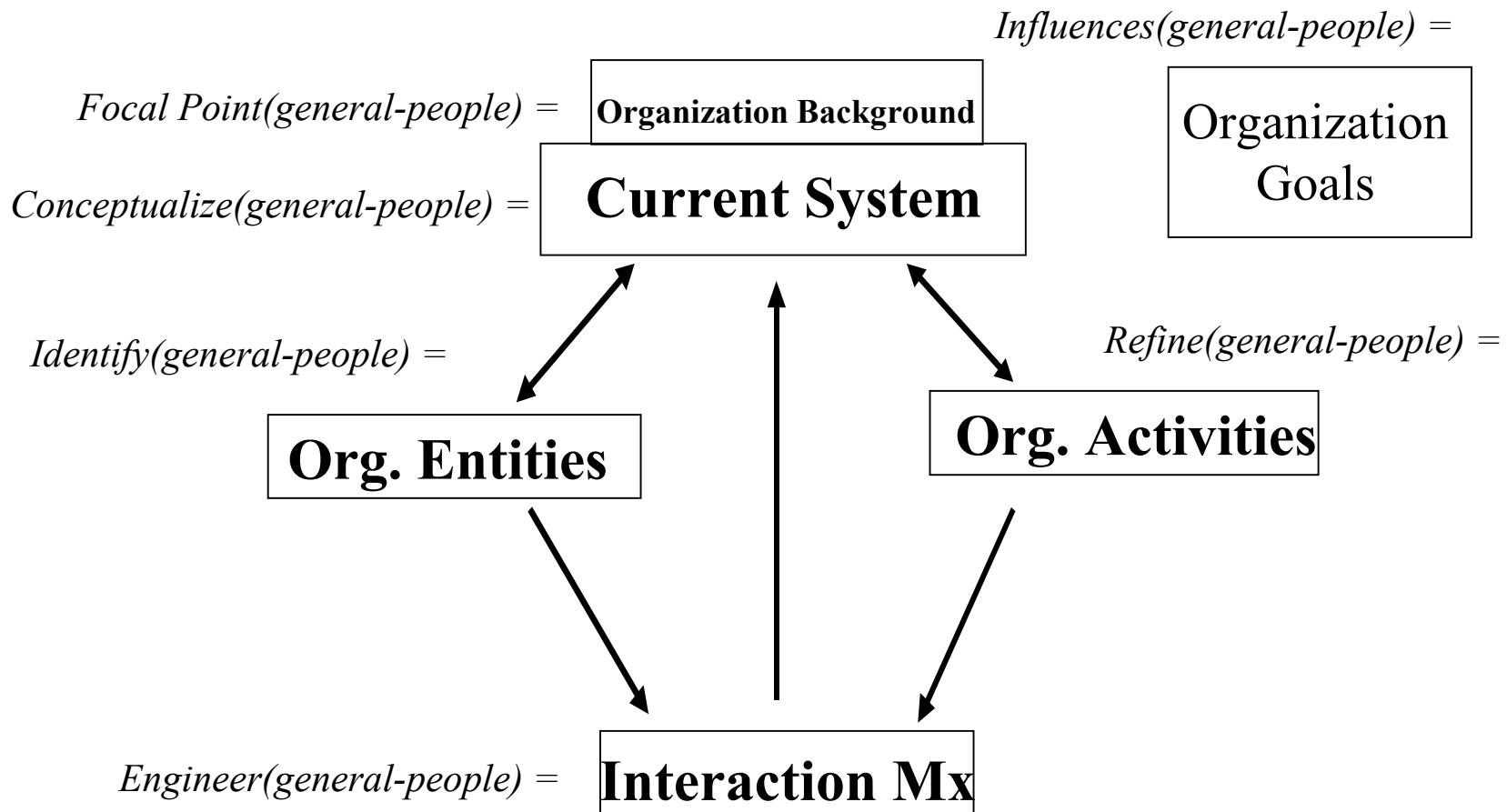
Motivating a Project

- A domain description provides a natural starting point for a project, but specifying agreed terminology to describe the existing organizations architecture
- A strong domain description takes extra time, but helps guide a project
- Without a solid domain description, projects can become less intuitive, unmotivated and drift off-track

Domain Description Purpose

- Create a concise, but not precise, description of the basic organization for which the project will be built
- Establish the context within which the project will both be developed and will operate
- Determine what is or is not relevant to the project
- Provide a familiar starting point

Domain Description Models



OCD 3.1 Organization Background

(Product Model)

- Good organization's background description. Only talks about existing organization, not in context with proposed system. It should give organization description in a large context, not too specific .

Example from team 6-1 Butler Stack Locator CS3156 Fall 1999:

3.1 Organization Background

The Butler stacks houses two million volumes of the University's collections in the humanities, with particular strengths in history (including government documents and social science materials published before 1974), literature, philosophy and religion, in addition to one of country's most extensive collections of materials pertinent to the study of Greco-Roman antiquity. The stacks has 12 levels with the entrance on the third floor of Butler near the Circulation Desk. The purpose of the Butler Library is to provide access to a variety of resources for patrons of Columbia University.

(from <http://www.columbia.edu/cu/libraries/indiv/under/tour/butlertour.html>)

OCD 3.2 Organization Goals

(Success Model)

3.2 Organization Goals

1. Ease the workload on the patrons and staff affiliated with Butler Library by providing user-friendly and useful assistance via computers to facilitate the searching of books in the library stacks

M: number of people who still seek assistance in finding a book with the BSL

R: the BSL wants to offer a plausible alternative to searching to searching for a book totally on one's own or through the assistance of the Management

2. Increase efficiency for the management through the creation of a system that facilitates categorization and grouping of books

M: time it takes to re-shelve and relocate books with the proposed system

R: maximizing efficiency allows the Management to concentrate their energies on other matters

3. Increase quality of service by allowing the management to focus their time and energies on those patrons that really need help

M: number of patrons who still seek assistance with the BSL

R: increased quality of service will book public image of the Butler Library Management

OCD 3.3 Organization Activities (Product Model)

3.3 Organization Activity Model

The current organizational activities of Butler Library are relatively simple. The system relies heavily on the limited activities of the Staff, since no computer system has been designed to alleviate Staff from performing these mundane tasks. The following is a description of the current activities of the Butler Library.

1. Search for Collection Materials
 1. User Searches for the Book.
 1. User uses CLIO or Card Catalog to find the call number, the size, and the collection type of the book.
 2. Inexperienced User asks questions on how to search for book.
 3. User refers to list of call numbers and stack levels called the Stack Directory.
 4. User proceeds to corresponding stack level and reads off a floor plan present at the elevator on each level to find the general location of the book.
 5. User scans through each range for correct call number intervals.
 2. User iterates through the above procedure
 3. User cannot find book
 1. User fills out formal request for a Staff to locate the book.
 2. User requests assistance from Staff (informal request)

OCD 3.4 Description of Current System

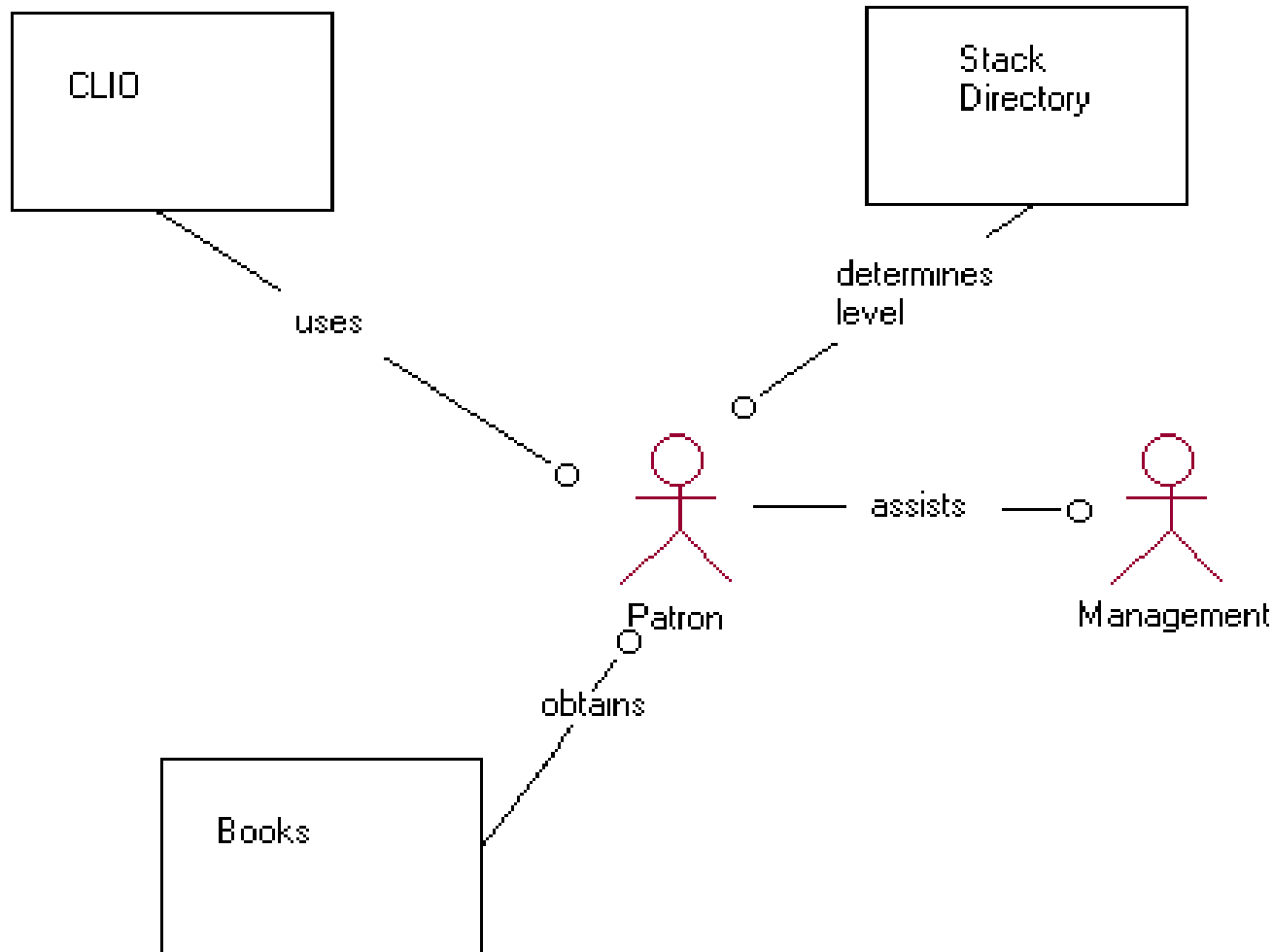
(Product Model)

Good description of operation of current system. The description is associated with *block* diagrams.

3.4 Description of Current System

- Patron
 - The patron determines the call number of the desired book through the Columbia's online catalog, CLIO.
 - The patron will then approach the circulation desk to acquire a copy of the stack directory (a document that maps the call number prefix with the corresponding stack level), with which he/she will determine the stack level of the book.
 - The patron will then proceed to the destination level and reference with the map that is posted on each stack level.
 - With this information, the patron is ready to approach the aisle that contains the desired book.

OCD 3.4 Description of Current System (Product Model)



OCD 3.5 Entity Model: Possible Entities (Product Model)

3.5 Entity Model

- Entities of the proposed system:
 - Patron
 - Management
 - CLIO
 - Books
 - Butler Library
 - Stack Level

E-01

Identifier - E-01

Description - a user affiliated with Columbia University who utilizes the resources of Butler Library

Name - Patron

Properties - Student, Staff, Employee, or any affiliated person of Columbia University.

Activities

- Uses CLIO
- seeks assistance and checks out books from Management
- uses books as resources
- affiliated with Columbia University

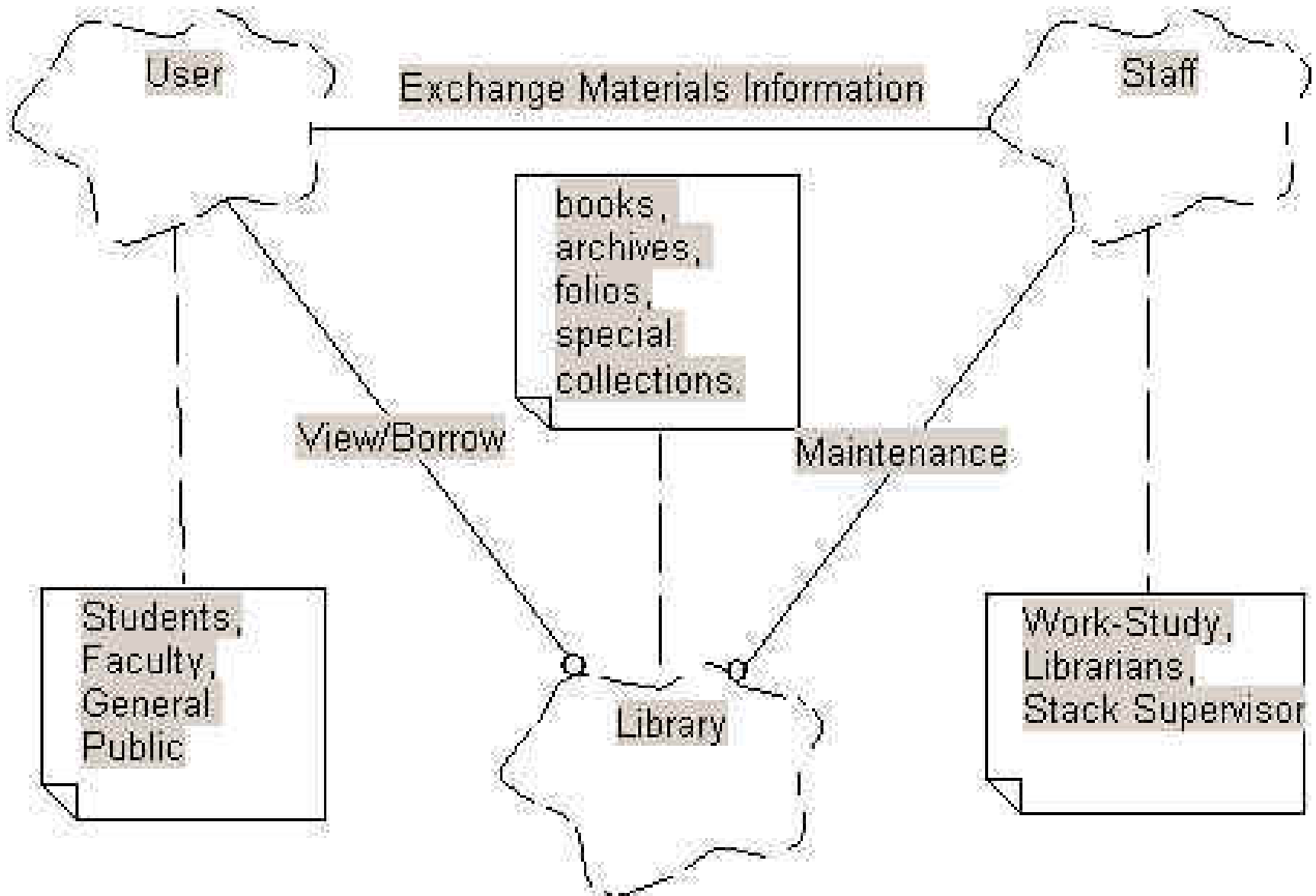
Connections – ENT-01

OCD 3.5 Entity Model: Entity Specifications (Product Model)

Example from team 16-1 CS3156 Fall 1999

Identifier	E-01
Description	People who utilize the library system and the services that the Butler Stacks provide. This includes faculty, students, researchers and the general public.
Name	User
Properties	<ol style="list-style-type: none">1. Name2. Level of Priority (Student or Faculty)
Activities	<ol style="list-style-type: none">1. Inquires about stack materials2. Requests stack materials3. Views stack materials4. Borrows stack materials
Connections	<ol style="list-style-type: none">1. Users receive information about stack materials from Staff2. Users borrow stack materials from the Library

OCD 3.5 Entity Model Diagram



OCD 3.6 Interaction Model (Product Model)

	<i>Patron</i>	<i>Management</i>	<i>Stack Level</i>	<i>Book</i>	<i>CLIO</i>	<i>Stack Directory</i>
<i>Uses CLIO</i>	X	X			X	
<i>Seeks Assistance</i>	X					
<i>Uses Stack Directory</i>	X	X				X
<i>Gets Stack Directory</i>	X					X
<i>Gives Stack Directory</i>		X				X
<i>Goes to Stack Level</i>	X	X	X			
<i>Finds Book</i>	X	X		X		
<i>Relocates Book</i>		X		X		
<i>Searches for Book</i>	X	X				

OCD 3.7 Current System Shortfalls (Product Model)

- Good description of current system shortfalls. It should not give advantage or improvements will be done by proposed system.

3.7 Current System Shortfalls

1. User

1. Users have to go through several look-up tables, which makes book searching a very time-consuming task. User has to search stack directory, locate range on stack map, and search through range numbers in order to locate a specified book. If all else fails, the User must go to Staff for assistance.
2. Not all the call numbers are in the same format so inexperienced users can get easily confused with the various collections available.
3. User is not directly provided with a map. The sole map for each stack level is located at the entrance to the level, so it is difficult for Users to find their way around the stacks.