System Analysis and Design
UML

Sixth Edition

Alan Dennis, Barbara Wixom, Roberta M. Roth

Chapter 1

The Systems Analyst and Information Systems Development

Objectives

- Explain the role played in IS development by the systems analyst.
- Describe the fundamental systems development life cycle and its four phases.
- Explain how organizations identify IS development projects.
- Explain the importance of linking the IS to business needs.
- Be able to create a system request.
- Describe technical, economic, and organizational feasibility assessment.
- Be able to perform a feasibility analysis.

Copyright ©2022 John Wiley & Sons, Inc.

Introduction

- The systems development life cycle (SDLC) is the process of determining how an information system (IS) can support business needs, designing the system, building it, and delivering it to users
- The systems analyst plays a key role in the SDLC, analyzing the business situation, identifying opportunities for improvements, and designing an IS to implement the improvements
- The primary goal of the system analyst is to create value for the organization, which for most companies means increasing profits
- Systems analysts do things and challenge the current way that an organization works.

Copyright ©2022 John Wiley & Sons, Inc.

Introduction Continued

- Large systems development projects are particularly susceptible to failure
- An analysis of very large development projects conducted by the Standish Group found that for projects that exceed $100 million in labor costs, only 2% are successful
  - Several major factors were delays in decision making and a high workforce turnover during the project

Copyright ©2022 John Wiley & Sons, Inc.

The Systems Analyst

- The systems analyst works closely with all project team members so that the team develops the right system in an effective way
- Systems analysts must understand how to apply technology to solve business problems
- Systems analysts serve as change agents who identify the organizational improvements needed, design systems to implement those changes, and train and motivate others to use the systems

Copyright ©2022 John Wiley & Sons, Inc.

Systems Analyst Skills

1. Analysts must have the ***technical skills*** to understand the organization's existing technical environment, the new system's technology foundation, and the way in which both can be fit into an integrated technical solution
2. ***Business skills*** are required to understand how IT can be applied to business processes and to ensure that IT delivers real business value
3. Analysts are continuous problem solvers at both the project and the organizational level, and they put their ***analytical skills*** to the test regularly

Copyright ©2022 John Wiley & Sons, Inc.

Systems Analyst Skills Continued

4. Analysts need strong ***interpersonal skills*** to communicate effectively, one-on-one with users and business managers, with programmers and other technical specialists, and with people from outsourcing firms and vendor organizations
5. Analysts need to ***manage*** people with whom they work, and they must manage the pressure and risks associated with unclear situations
6. Analysts must deal fairly, honestly, and ***ethically*** with other project team members, managers, and system users

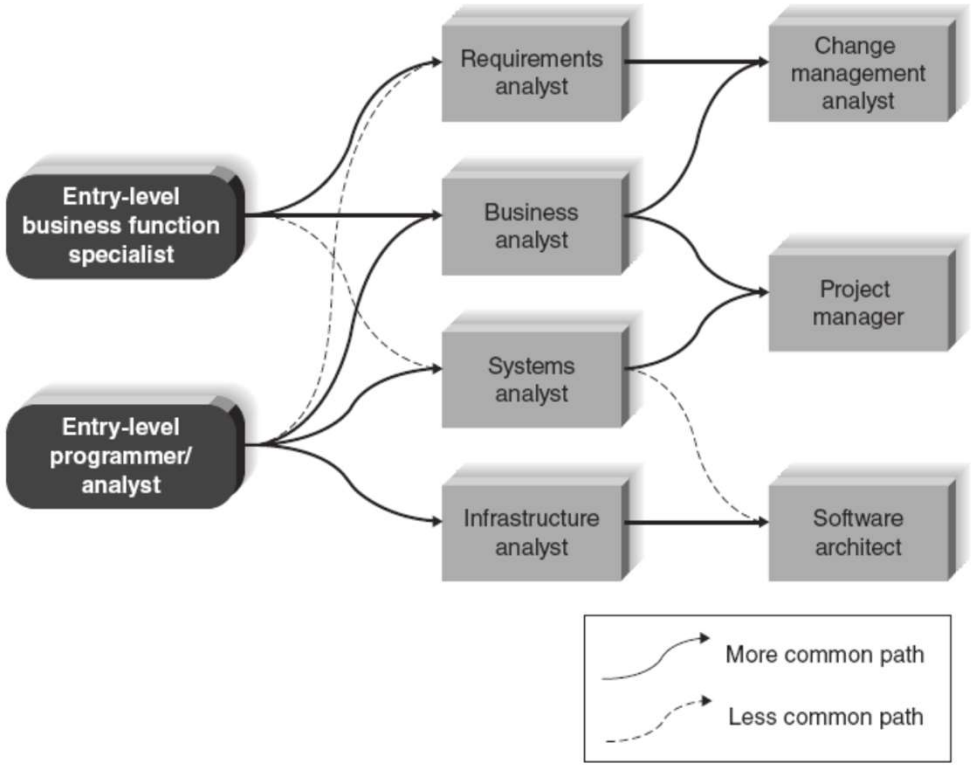
Copyright ©2022 John Wiley & Sons, Inc.

Systems Analyst Roles

- The ***systems analyst*** role focuses on the IS issues surrounding the system
- The ***business analyst*** role focuses on the business issues surrounding the system
- The ***requirements analyst*** role focuses on eliciting the requirements from the stakeholders associated with the new system
- The ***infrastructure analyst*** role focuses on technical issues surrounding the ways the system will interact with the organization's technical infrastructure

Copyright ©2022 John Wiley & Sons, Inc.

Career Paths for System Developers



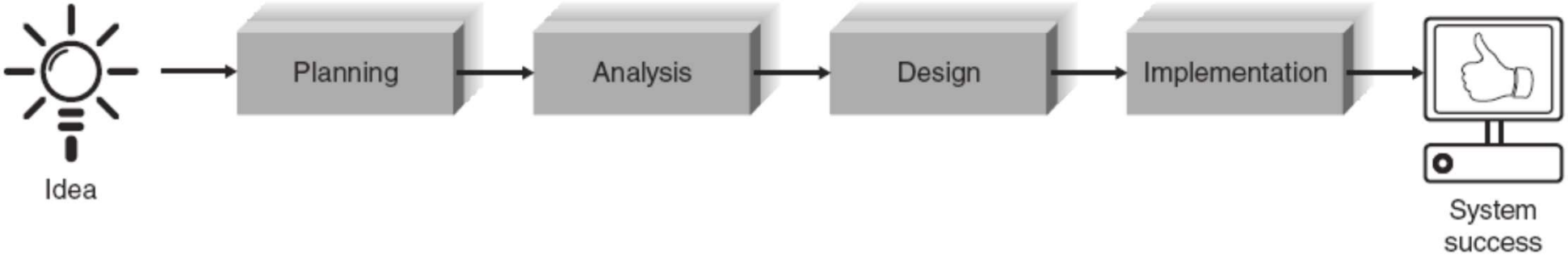
Copyright ©2022 John Wiley & Sons, Inc.

Systems Analyst Roles Continued

- The ***software architect*** takes a holistic view of the organization's entire IT environment and guides application design decisions within that context
- The ***change management*** analyst role focuses on the people and management issues surrounding the system installation
- The ***project manager*** role ensures that the project is completed on time and within budget and that the system delivers the expected value to the organization

Copyright ©2022 John Wiley & Sons, Inc.

The Systems Development Life Cycle



Copyright ©2022 John Wiley & Sons, Inc.

Planning Phase

- Project initiation
  - Prepare system request
  - Perform preliminary feasibility analysis
- Set up the project
  - Project plan, including work plan and staffing plan

Copyright ©2022 John Wiley & Sons, Inc.

Analysis Phase

- Determine analysis strategy
  - Study existing system and its problems
- Collect and analyze requirements
  - Develop new system concept
  - Describe new system with analysis models
- Prepare and present system proposal
  - Summarize results of the Analysis Phase
  - Go/No Go decision made by sponsor and steering committee

Copyright ©2022 John Wiley & Sons, Inc.

Design Phase

- Determine design strategy
  - Build / buy / outsource
- Design system components
  - Architecture, interface, database, programs
  - Assemble design elements into system specification
- Present to steering committee
  - Go /no go decision before entering final phase

Copyright ©2022 John Wiley & Sons, Inc.

Implementation Phase

- System construction
  - Programming and testing
- System installation
  - Training
  - Conversion to new system
- On-going system support

Copyright ©2022 John Wiley & Sons, Inc.

Project Identification and Initiation

- Fulfill a business need
  - Enable a business initiative or strategy
  - Support a merger/acquisition
  - Fix a “point of pain”
  - Utilize a new technology
- Outgrowth of business process management (BPM)

Copyright ©2022 John Wiley & Sons, Inc.

How many methods are there to develop a lunch making machine?

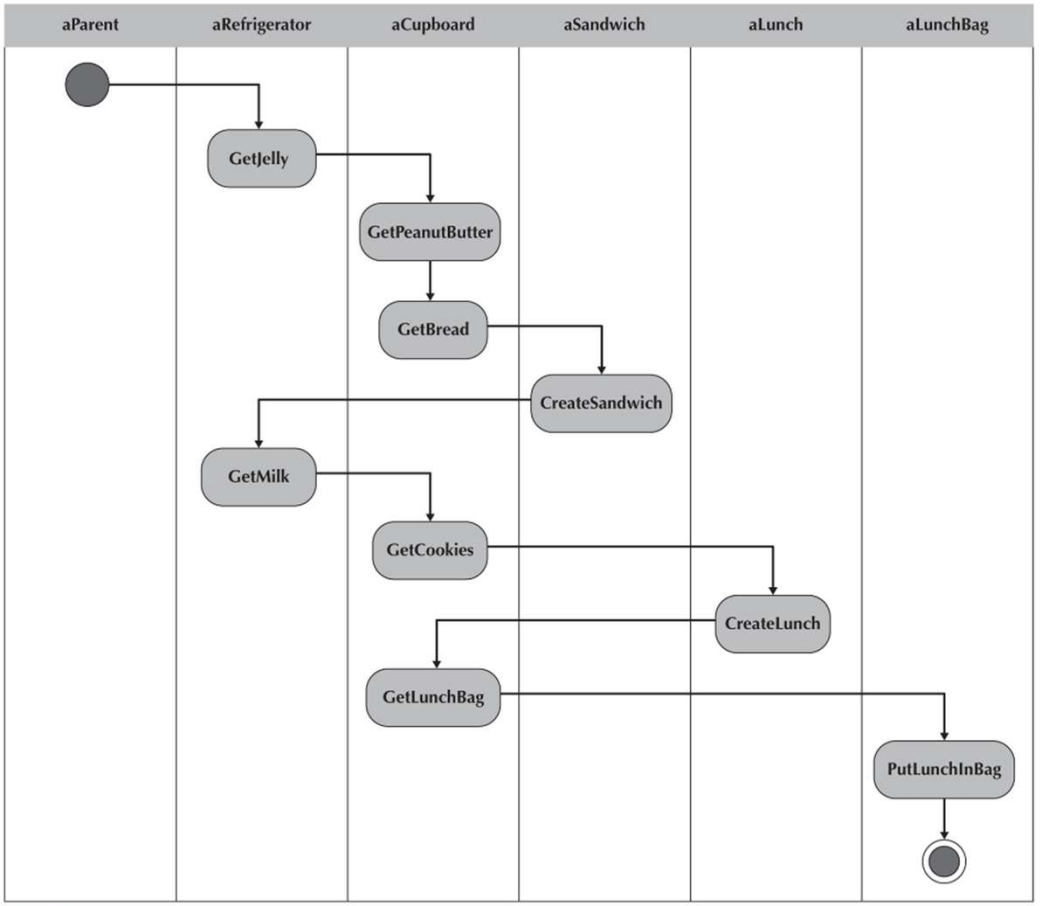


FIGURE 1-1 A Simple Behavioral Model for Making a Simple Lunch

Copyright ©2022 John Wiley & Sons, Inc.

Structured Analysis

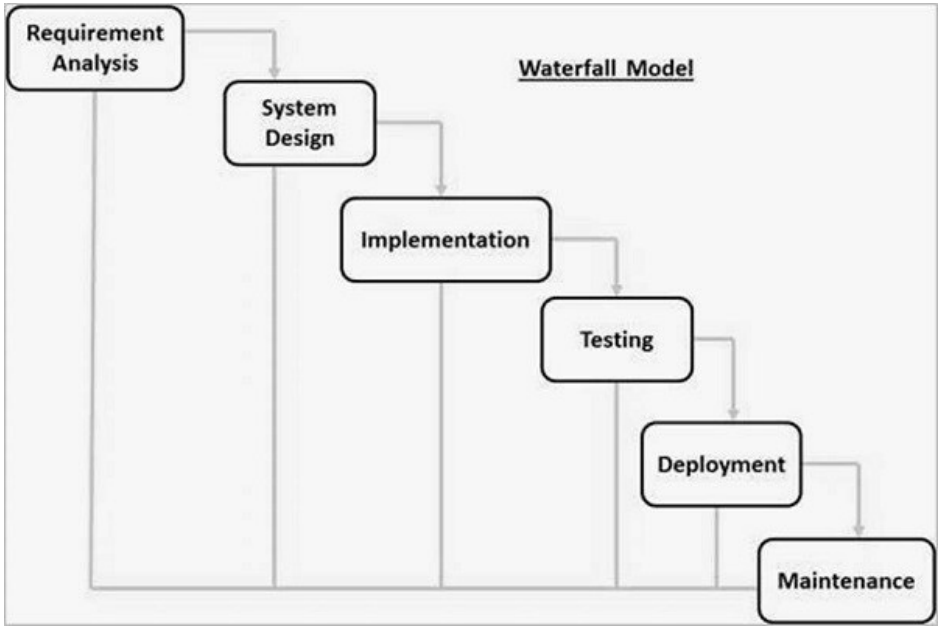
- Structured design methodologies adopt a formal step-by-step approach to the SDLC that moves logically from one phase to the next.

Copyright ©2022 John Wiley & Sons, Inc.

Structured Analysis

Structured design methodologies adopt a formal step-by-step approach to the SDLC that moves logically from one phase to the next.

- Waterfall
  - Development moves forward from phase to phase in the same manner as a waterfall.
  - Analysts and users proceed in sequence from one phase to the next.
  - Key deliverables for each phase are typically very long (often hundreds of pages in length) and are presented to the project sponsor for approval as the project moves from phase to phase.
  - Upon sponsor approval next phase begins, and not before
  - Although it is possible to go backward in the SDLC (e.g., from design back to analysis), it is extremely difficult (imagine yourself as a salmon trying to swim upstream against a waterfall).



Copyright ©2022 John Wiley & Sons, Inc.

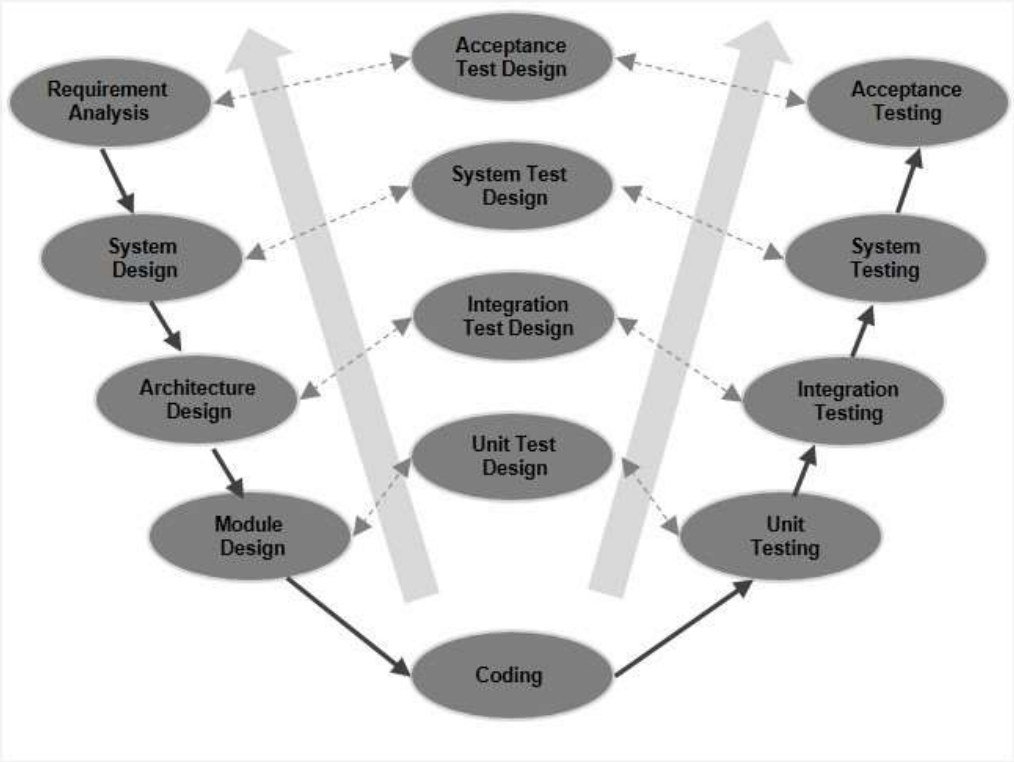
Structured Analysis

- Waterfall
  - Advantages
    - Requirements well defined before development
    - Minimizes changes to requirements during the SDLC.
  - Disadvantages
    - Design MUST be complete before programming
    - Long delays between Planning and actual development of any sort. Time To Market takes a hit

Copyright ©2022 John Wiley & Sons, Inc.

Structured Analysis

- V-Model
  - Extension of the waterfall model
  - Based on the association of a testing phase for each corresponding development stage.
  - Every single phase in the development cycle, has a directly associated testing phase.
  - This is a highly-disciplined model and the next phase starts only after completion of the previous phase.

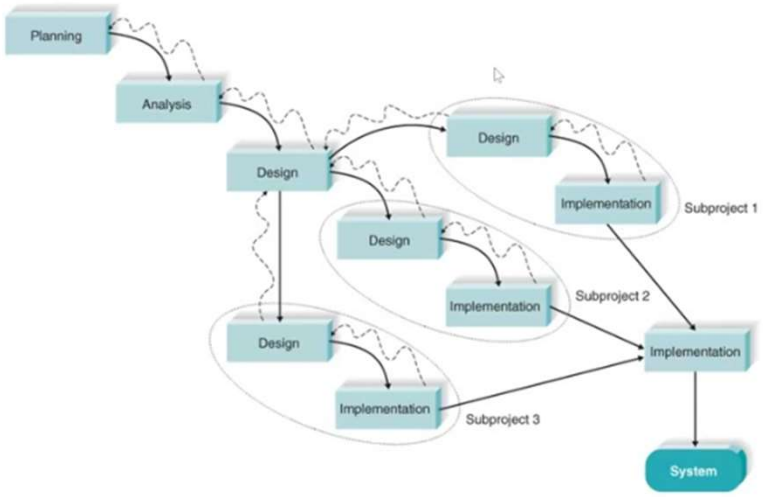


Copyright ©2022 John Wiley & Sons, Inc.

Structured Analysis

Parallel Development Methodology

- Subdivide the project into subprojects that can be worked on at the same time.
- Reduce the overall project length



- Parallel Development
  - Define high level system design
  - Divide into sub systems and develop in parallel.
  - Integrate once all sub systems complete
  - Advantage
    - Reduce time to market
  - Disadvantage
    - Independence and mutual exclusion of sub systems key. But dependencies necessarily exist. When one subsystem changes, the others need to as well. Lots of churn.

Copyright ©2022 John Wiley & Sons, Inc.

Rapid Application Develop

- RAD-based methodologies attempt to address both weaknesses of structured design methodologies by adjusting the SDLC phases to get some part of the system developed quickly and into the hands of the users.
- In this way, the users can better understand the system and suggest revisions that bring the system closer to what is needed
- Special techniques and computer tools to speed up the analysis, design, and implementation phases
- Computer-aided software engineering (CASE) tools,
- Joint application design (JAD) sessions
- Fourth-generation or visual programming languages that simplify and speed up programming
- Code generators that automatically produce programs from design specifications.

Copyright ©2022 John Wiley & Sons, Inc.

Rapid Application Develop

- Adverse Affects
  - Use of the tools and techniques that can improve the speed and quality may result in user expectations of what is possible can change dramatically.
  - As a user better understands the information technology (IT), the systems requirements tend to expand.
  - This was less of a problem when using methodologies that spent a lot of time thoroughly documenting requirements.

Copyright ©2022 John Wiley & Sons, Inc.

Rapid Application Develop

- Phased Development
  - Breaks an overall system into a series of versions that are developed sequentially.
  - The analysis phase identifies the overall system concept
  - Stakeholders categorize the requirements into a series of versions.
  - The most important and fundamental requirements are bundled into the first version of the system.
  - The analysis phase then leads into design and implementation—but only with the set of requirements identified for version 1.
  - Once version 1 is implemented, work begins on version 2.
  - Additional analysis is performed based on the previously identified requirements and combined with new ideas and issues that arose from the users' experience with version 1.
  - Repeat for version 3, etc.

Copyright ©2022 John Wiley & Sons, Inc.

Rapid Application Develop

- Prototyping
  - Not really known what the end system is really supposed to be.
  - Analysis, design, and implementation phases are concurrently performed.
  - All three phases are performed repeatedly in a cycle until the system is completed.
  - The basics of analysis and design are performed, and work immediately begins on a system prototype, a quick-and-dirty program that provides a minimal amount of features.
  - The first prototype is usually the first part of the system that is used.
  - Stakeholder feedback is used to reanalyze, redesign, and reimplement a second prototype, which provides a few more features.
  - This process continues in a cycle until the analysts, users, and sponsor agree that the prototype provides enough functionality to be installed and used in the organization. After the prototype (now called the “system”) is installed, refinement occurs until it is accepted as the new system.

Copyright ©2022 John Wiley & Sons, Inc.

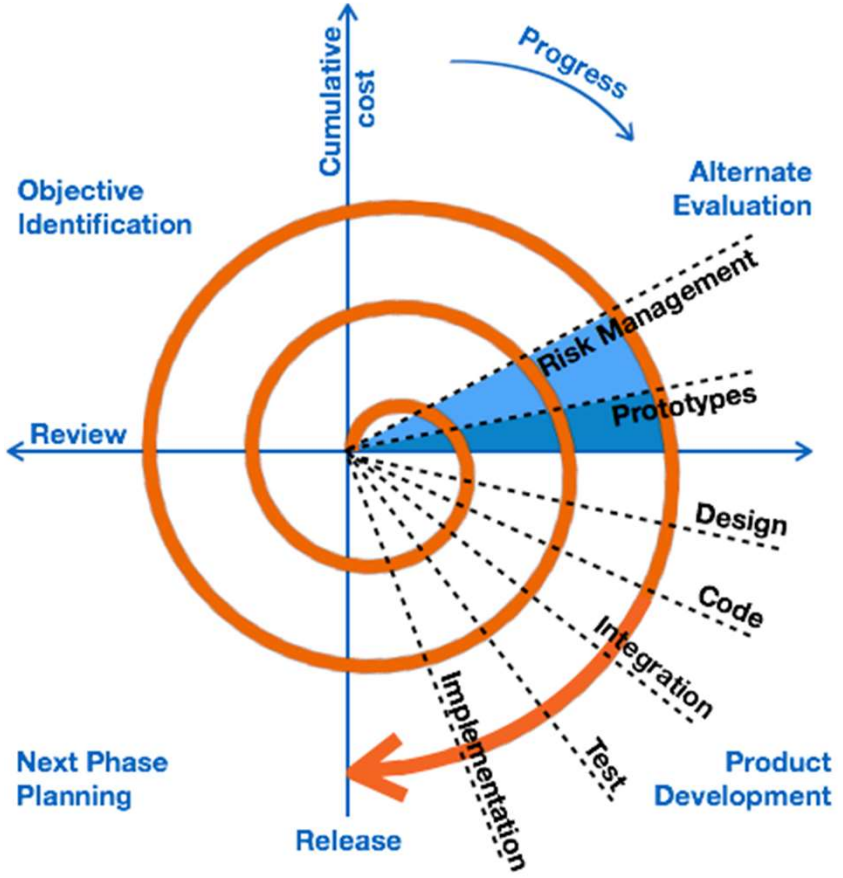
Rapid Application Develop

- Prototyping
  - Advantages
    - Very quickly provides a system with which the users can interact, even if it is not ready for widespread organizational use at first
    - Reassures users that the project team is working on the system (there are no long delays in which the users see little progress)
    - Prototyping helps to more quickly refine real requirements.
  - Disadvantage
    - Its fast-paced system releases challenge attempts to conduct careful, methodical analysis.
    - Often the prototype undergoes such significant changes that many initial design decisions become poor ones.
    - This can cause problems in the development of complex systems because fundamental issues and problems are not recognized until well into the development process.

Copyright ©2022 John Wiley & Sons, Inc.

Rapid Application Develop

- Spiral
  - Focused on Risk Management
  - Combines the idea of iterative development with the systematic, controlled aspects of the waterfall model.
  - It allows incremental releases of the product



Copyright ©2022 John Wiley & Sons, Inc.

Rapid Application Develop

- Spiral
  - Advantages
    - Changing requirements can be accommodated.
    - Allows extensive use of prototypes.
    - Requirements can be captured more accurately.
    - Users see the system early.
    - Development can be divided into smaller parts and the risky parts can be developed earlier which helps in better risk management.
  - Disadvantages
    - Management is more complex.
    - End of the project may not be known early.
    - Not suitable for small or low risk projects and could be expensive for small projects.
    - Process is complex
    - Spiral may go on indefinitely.
    - Large number of intermediate stages requires excessive documentation.

Copyright ©2022 John Wiley & Sons, Inc.

More Reading

- https://www.tutorialspoint.com/sdlc/index.htm

Copyright ©2022 John Wiley & Sons, Inc.

Structured Analysis

- Structured design methodologies adopt a formal step-by-step approach to the SDLC that moves logically from one phase to the next.

Copyright ©2022 John Wiley & Sons, Inc.

Object Oriented Analysis and Design

- Use RAD-based sequence of SDLC phases but attempt to balance the emphasis between process and data by focusing the analysis of problems on objects that contain both data and processes.

Copyright ©2022 John Wiley & Sons, Inc.

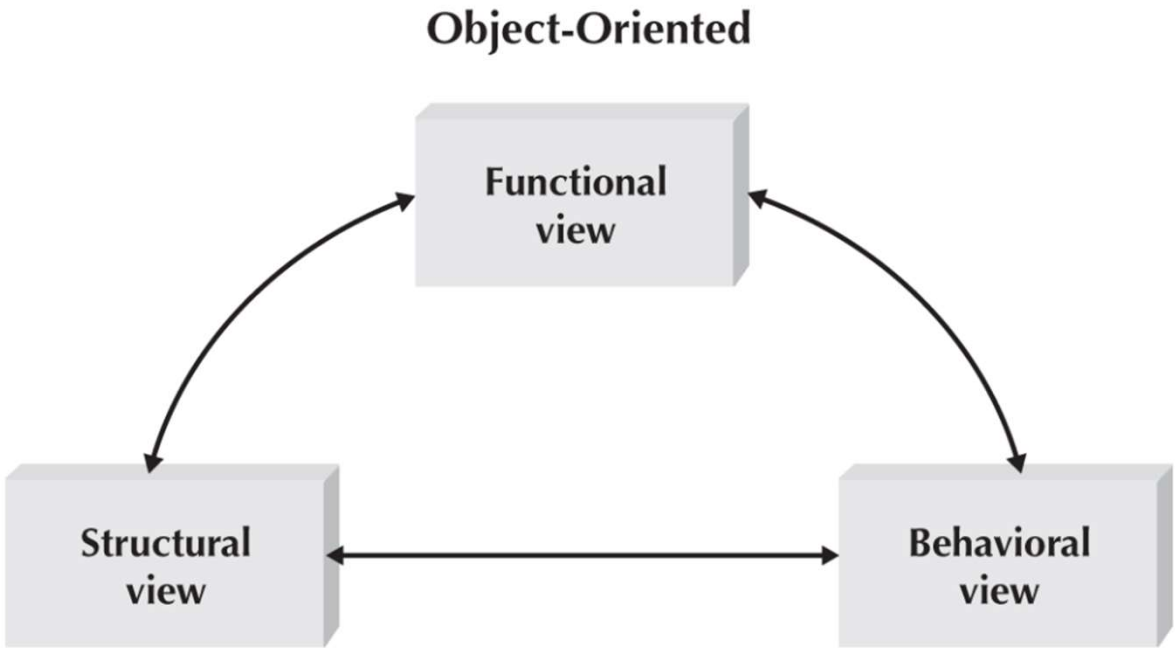
Object Oriented Analysis and Design

- Use Case Driven – Focus on Uses of the system, versus desired functions of the system. Subtle, but different enough.
  - According to the creators of the Unified Modeling Language (UML), Grady Booch, Ivar Jacobson, and James Rumbaugh,5 any modern object-oriented approach to developing information systems must be use-case driven, architecture-centric, and iterative and incremental.
- Architecture Centric – The underlying software architecture of the evolving system specification drives the specification, construction, and documentation of the system
  - Functional
  - Structural (Static)
  - Behavioral (Dynamic)
- Iterative and Incremental – specify, analyze, design, implement, repeat; in baby steps.

Copyright ©2022 John Wiley & Sons, Inc.

Object Oriented Analysis and Design

- Functional View – Use Cases
- Structural View – Architecture and Design Diagrams
- Behavioral View - Activity and Sequence Diagrams



Copyright ©2022 John Wiley & Sons, Inc.

Agile Development

- This is a process methodology, not a development methodology
- Any of the above develop approaches can fit within the Agile paradigm.
  - OOAD just supports it best.
- Agile is defined by the Agile Manifesto, and the 12 principles.

Individuals and interactions **over** processes and tools
Working software **over** comprehensive documentation
Customer collaboration **over** contract negotiation
Responding to change **over** following a plan

Copyright ©2022 John Wiley & Sons, Inc.

Agile Development – 12 Principles

- Software is delivered early and continuously through the development process, satisfying the customer.
- Changing requirements are embraced regardless of when they occur in the development process.
- Working software is delivered frequently to the customer.
- Customers and developers work together to solve the business problem.
- Motivated individuals create solutions; provide them the tools and environment they need, and trust them to deliver.
- Face-to-face communication within the development team is the most efficient and effective method of gathering requirements.
- The primary measure of progress is working, executing software.
- Both customers and developers should work at a pace that is sustainable. That is, the level of work could be maintained indefinitely without any worker burnout.
- Agility is heightened through attention to both technical excellence and good design.
- Simplicity, the avoidance of unnecessary work, is essential.
- Self-organizing teams develop the best architectures, requirements, and designs.
- Development teams regularly reflect on how to improve their development processes.

Copyright ©2022 John Wiley & Sons, Inc.

Agile Critics

- Much of the actual information systems may be development offshore, outsourced, and/or subcontracted. Given agile development methodologies requiring co-location of the team, this seems to be a very unrealistic assumption.
- If agile development is not carefully managed, ***and by definition it is not***, the development process can devolve into a prototyping approach that essentially becomes a “programmers gone wild” environment where programmers attempt to hack together solutions.
- Based on the lack of actual documentation created during the development of the software, issues are raised regarding the auditability of the systems being created. Without sufficient documentation, neither the system nor the systems-development process can be assured.
- Can deliver large mission-critical systems???

Copyright ©2022 John Wiley & Sons, Inc.

Extreme Programming

- Based on communication, simplicity, feedback, and courage.
  - Whenever courage is a basis, you may be doomed to failure
- Very small sets of functionality developed with little to no documentation, and no concrete vision of the final product.
- May work for very small, non safety or mission critical pieces of software.

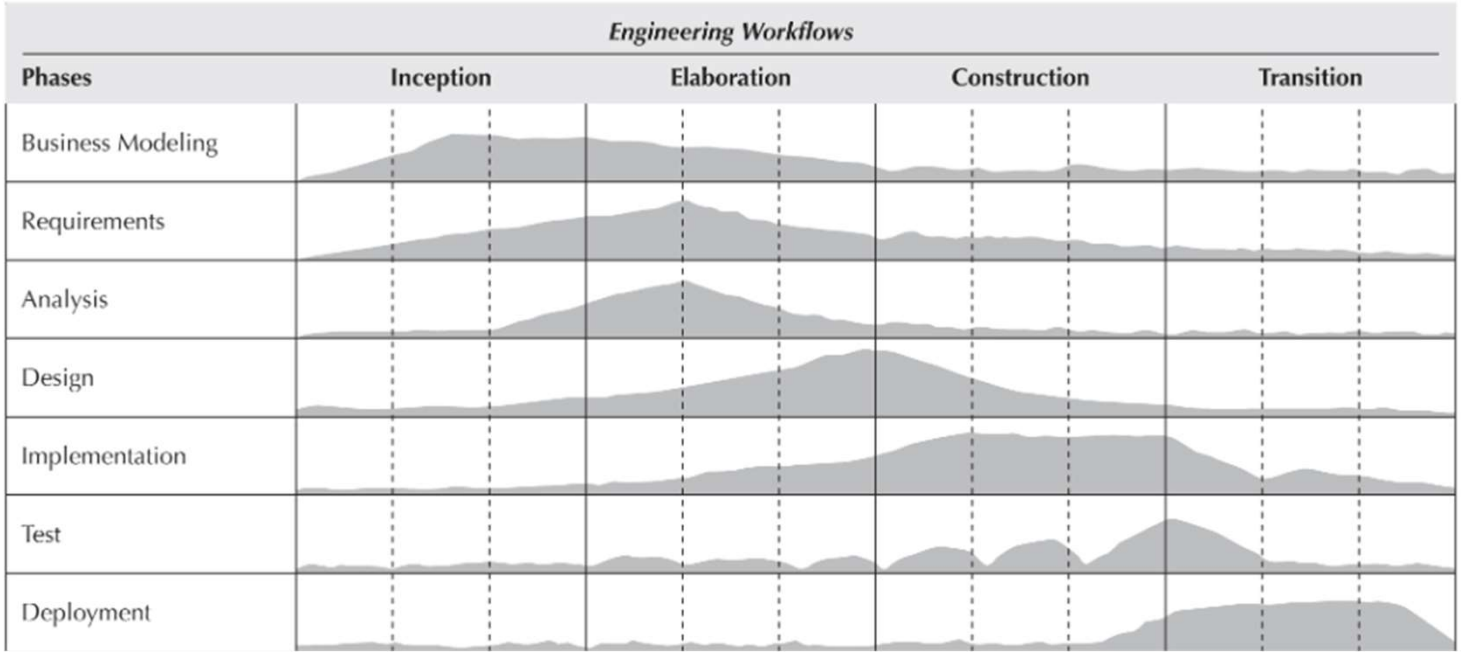
Copyright ©2022 John Wiley & Sons, Inc.

SCRUM

- In rugby, a scrum is used to restart a game.
- In a nutshell, the creators of the Scrum method believe that no matter how much you plan, as soon as the software begins to be developed, chaos breaks out and the plans go out the window.
- The best you can do is to react to where the proverbial rugby ball squirts out. You then sprint with the ball until the next scrum.
- In the case of the Scrum methodology, a sprint is defined as a time-boxed event, during which a small piece of the product is produced.
- At the end of the sprint, a demonstratable product is delivered to the customer.
- Much more later….

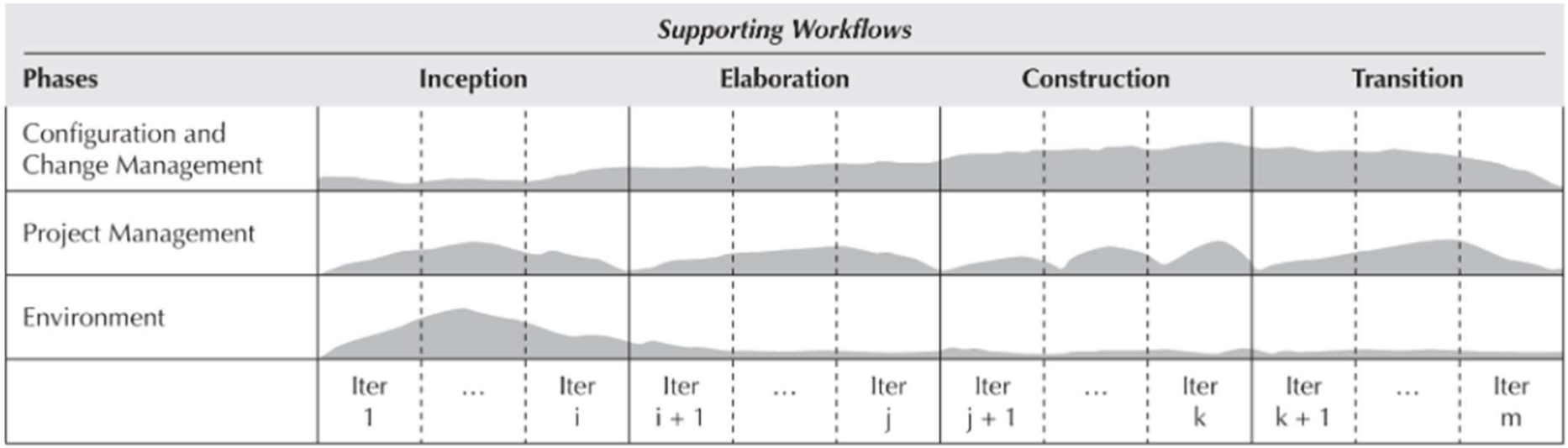
Copyright ©2022 John Wiley & Sons, Inc.

Unified Process



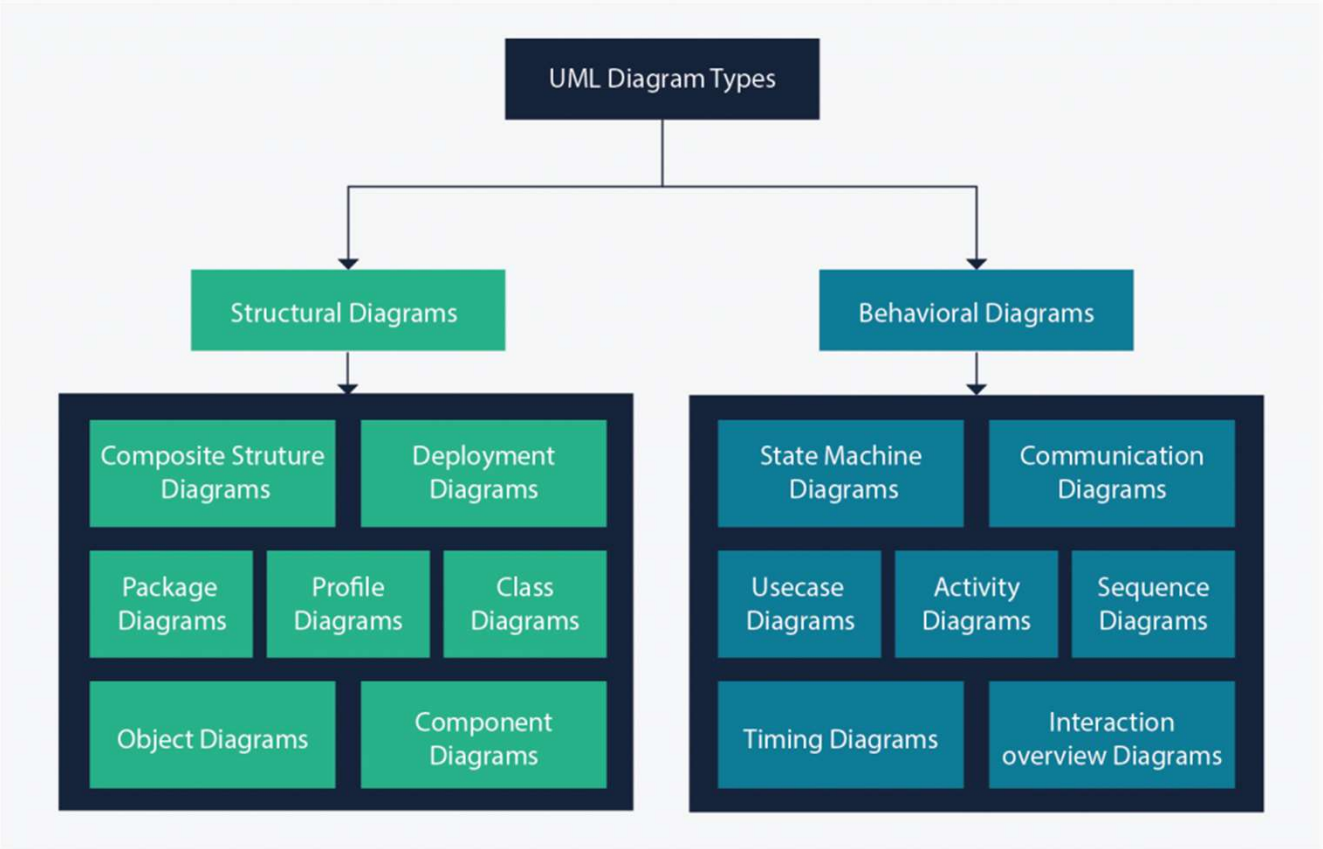
Copyright ©2022 John Wiley & Sons, Inc.

Unified Process



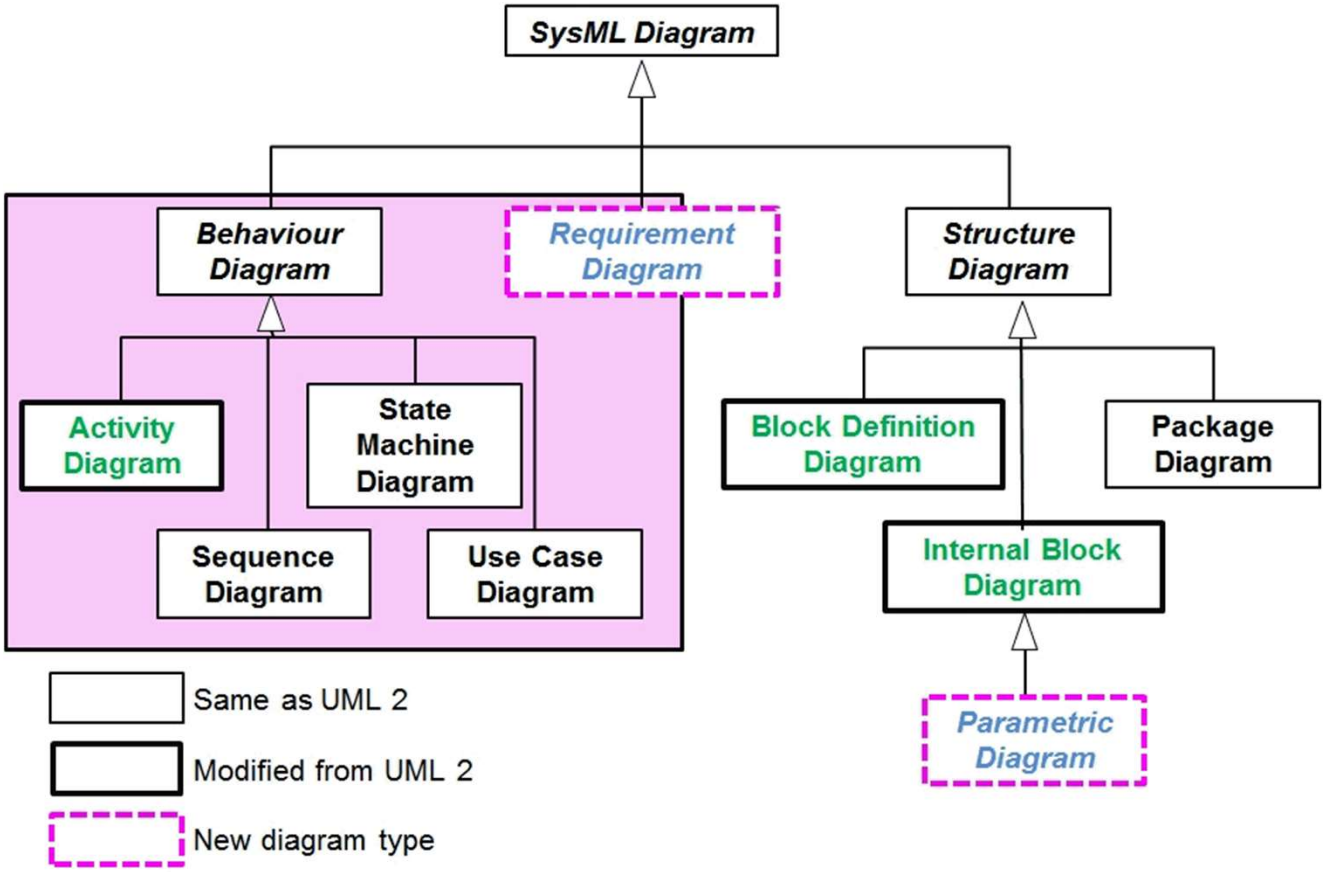
Copyright ©2022 John Wiley & Sons, Inc.

UML Diagrams



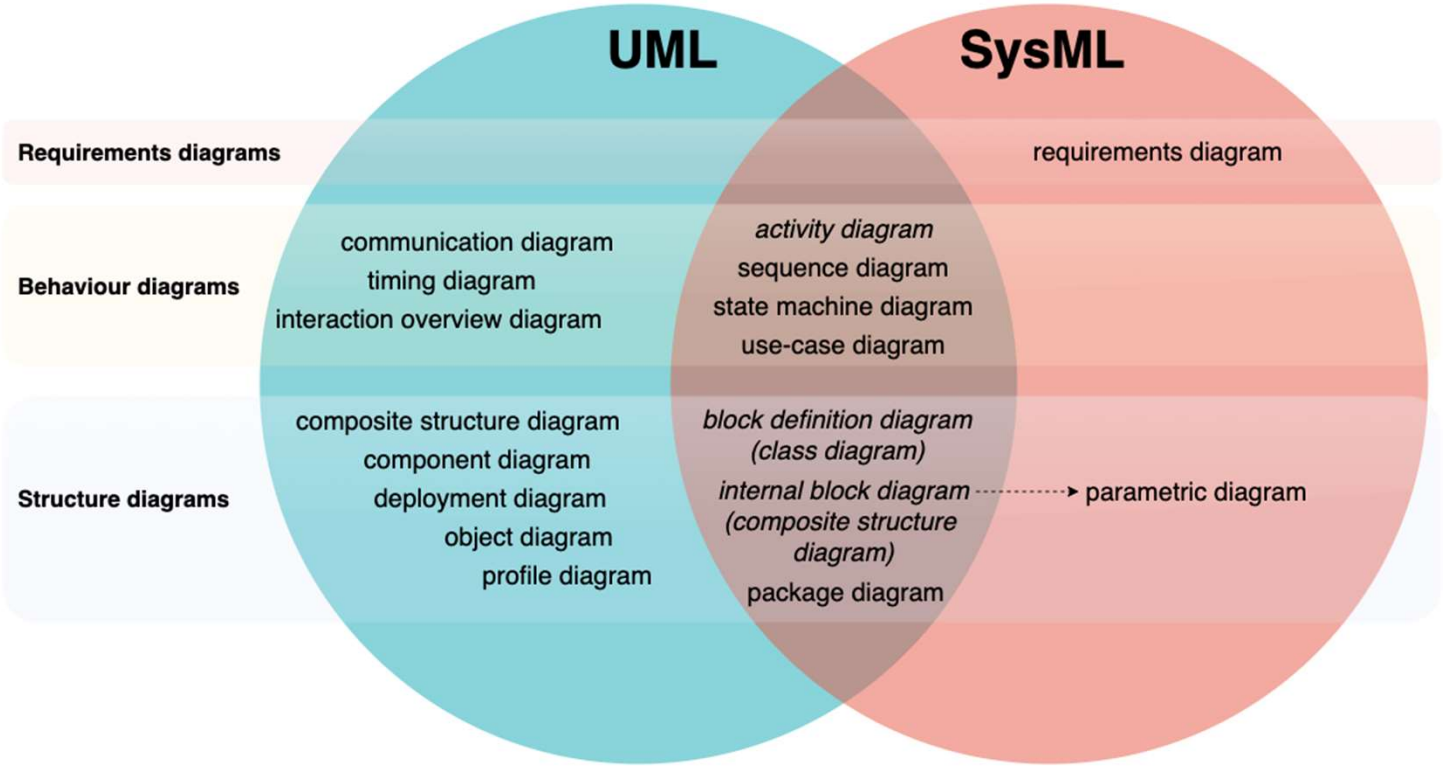
Copyright ©2022 John Wiley & Sons, Inc.

SysML Diagrams



Copyright ©2022 John Wiley & Sons, Inc.

UML/SysML Diagram Relationships



Copyright ©2022 John Wiley & Sons, Inc.