Wyyzzk, Inc.

Product Family and Program of Projects Architectures

Version 5.0

Lesson Goal & Objectives

- Understand the purpose and nature of product family and program architectures.
- Upon completion of the lesson, the participant will be able to:
  - Describe the issues and constraints of a product family architecture
  - Describe the issues and constraints of a program of projects architecture

Lesson Outline

- Product Family (Line-of-Business, Product Line) Architecture
  - What is product family architecture?
  - Variability
  - Commonality
  - Versioning & Bug Fixes
  - Use Cases and Other Requirements
  - Documentation
- Program (programme) of Projects Architecture
  - What is program of projects architecture?
  - Patterns
- Constraints and issues
- Impact on Project Architecture

Product Family Architecture

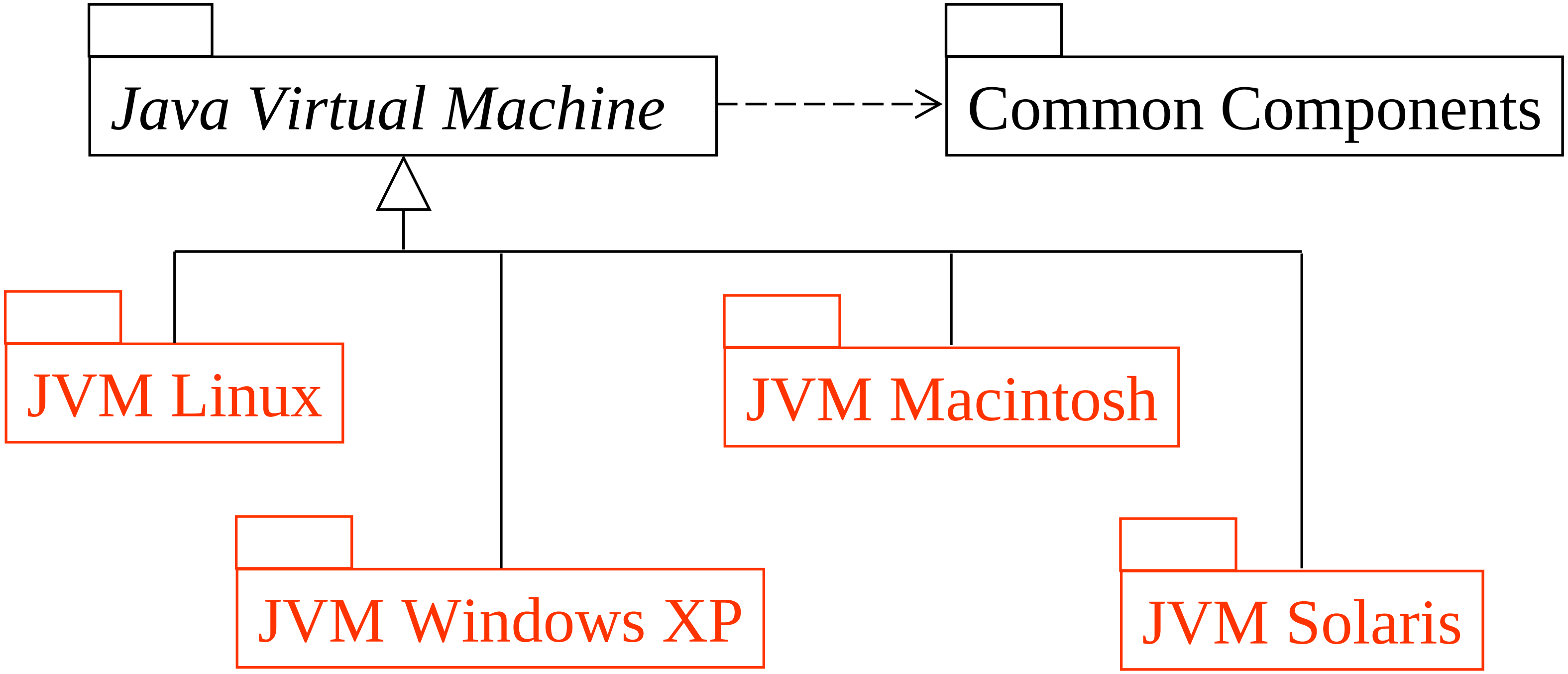
- This concerns the architecture of a set of applications that are related because they are variations on the same product or set of products
- This is sometimes called line-of-business architecture or product line architecture

Product Family Architecture

- Variability
  - Platform
  - Edition or feature set
  - Customized code
- Commonality
  - Framework
  - Base functionality
  - Common functions, events, services, agents
  - Data or file format
- Versioning & Bug fixes

Wyyzzk, Inc.

Platform



Java Virtual Machine

Common Components

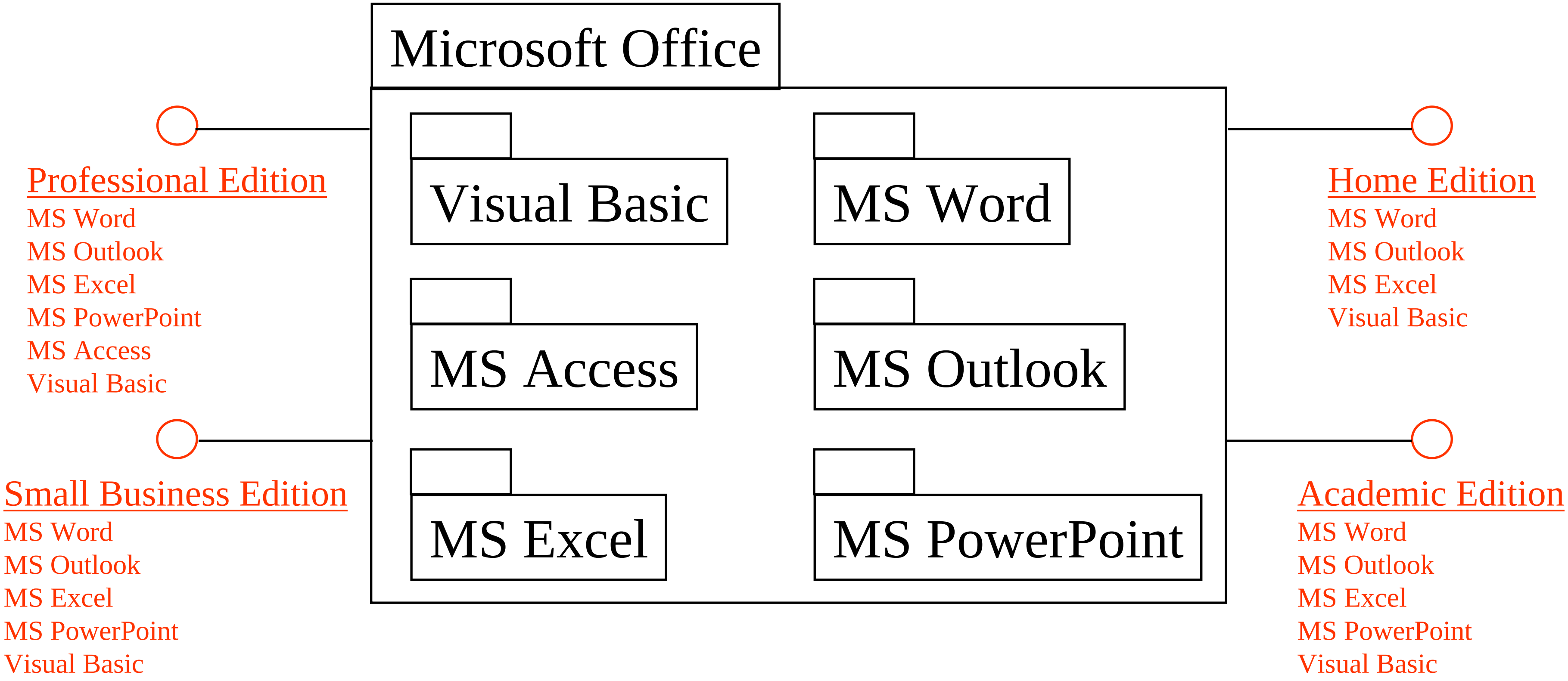
JVM Linux

JVM Macintosh

JVM Windows XP

JVM Solaris

Edition



Microsoft Office

Visual Basic

MS Word

MS Access

MS Outlook

MS Excel

MS PowerPoint

Professional Edition

MS Word
MS Outlook
MS Excel
MS PowerPoint
MS Access
Visual Basic

Home Edition

MS Word
MS Outlook
MS Excel
Visual Basic

Small Business Edition

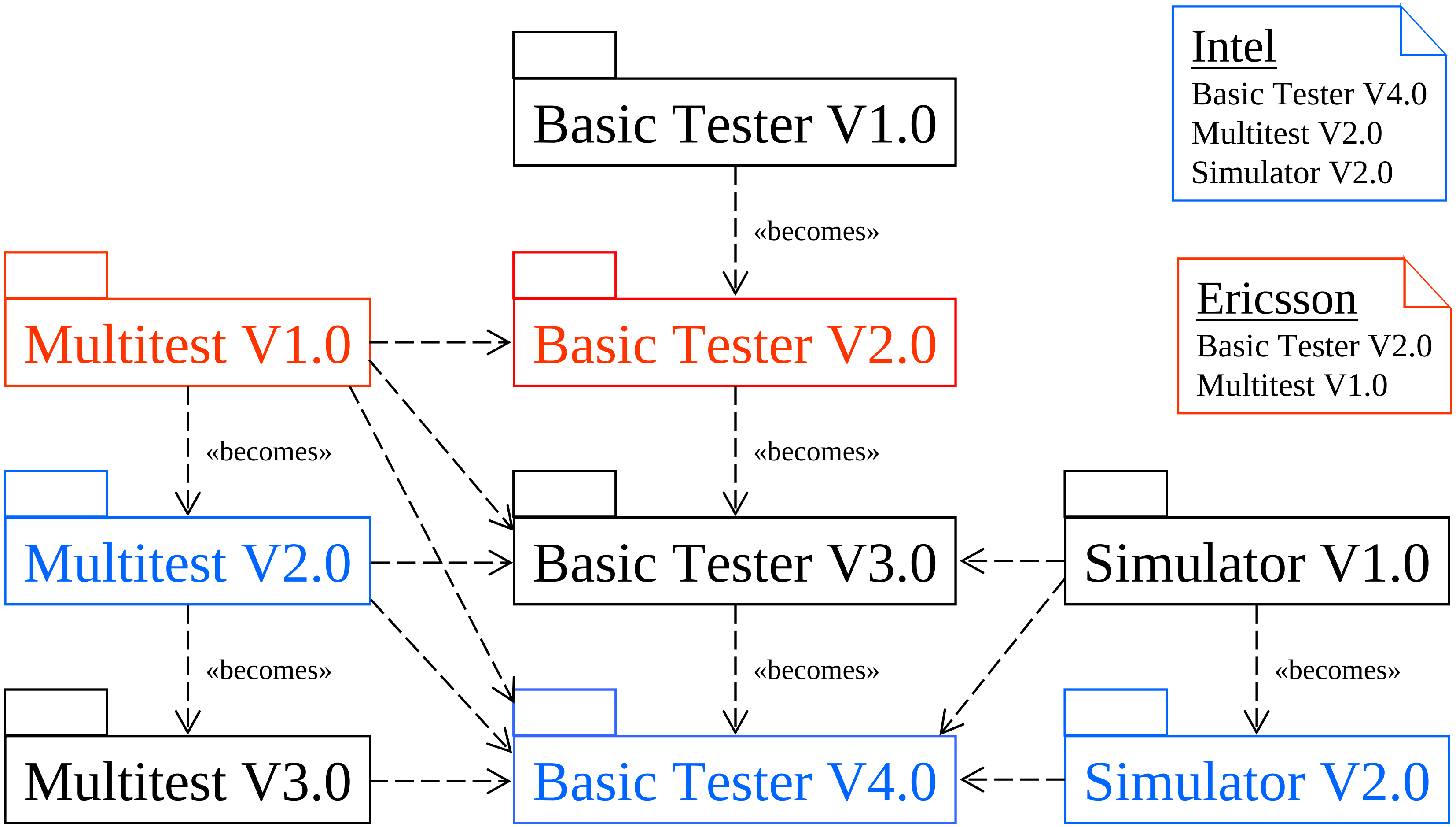
MS Word
MS Outlook
MS Excel
MS PowerPoint
Visual Basic

Academic Edition

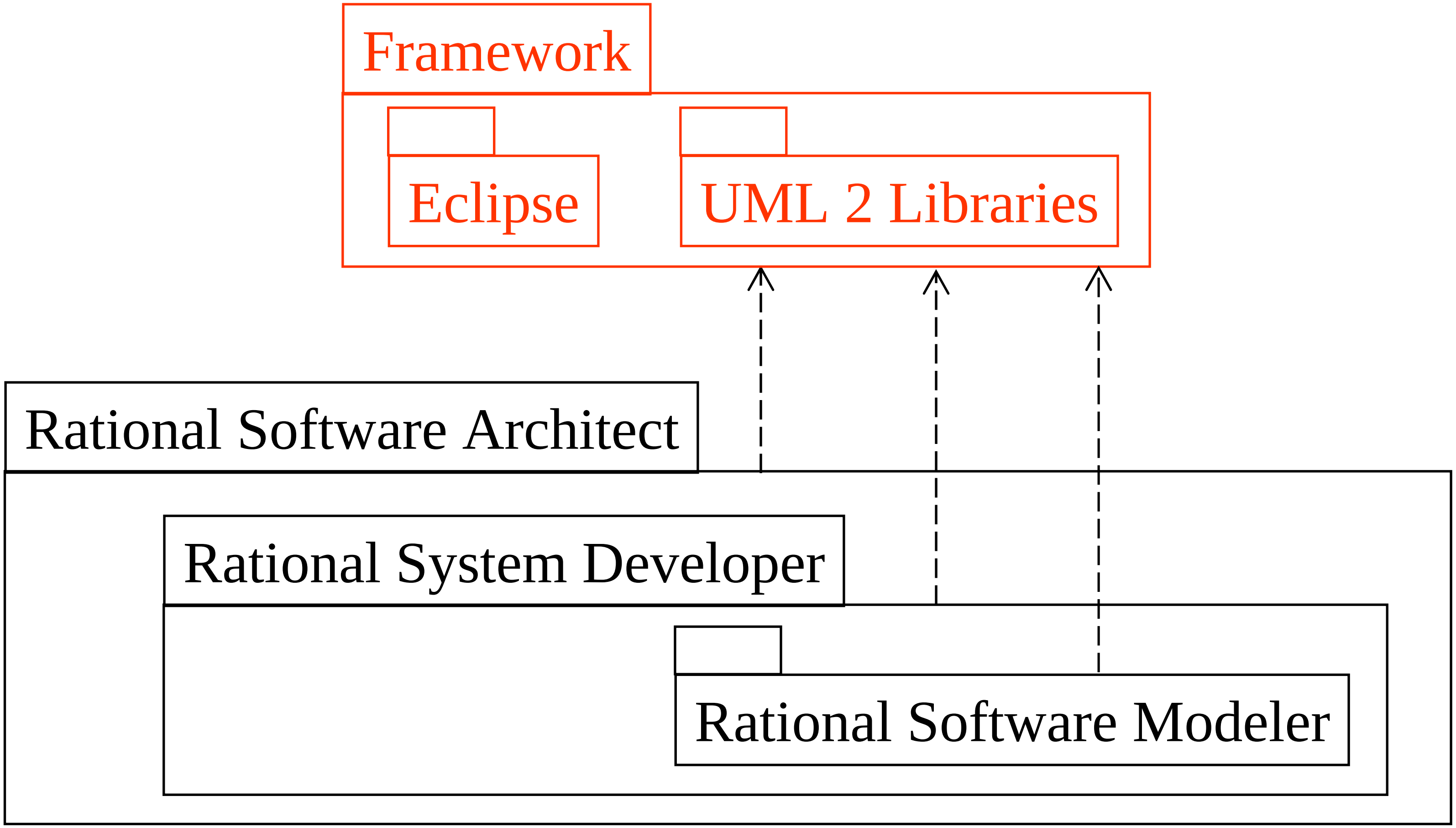
MS Word
MS Outlook
MS Excel
MS PowerPoint
Visual Basic

Wyyzzk, Inc.

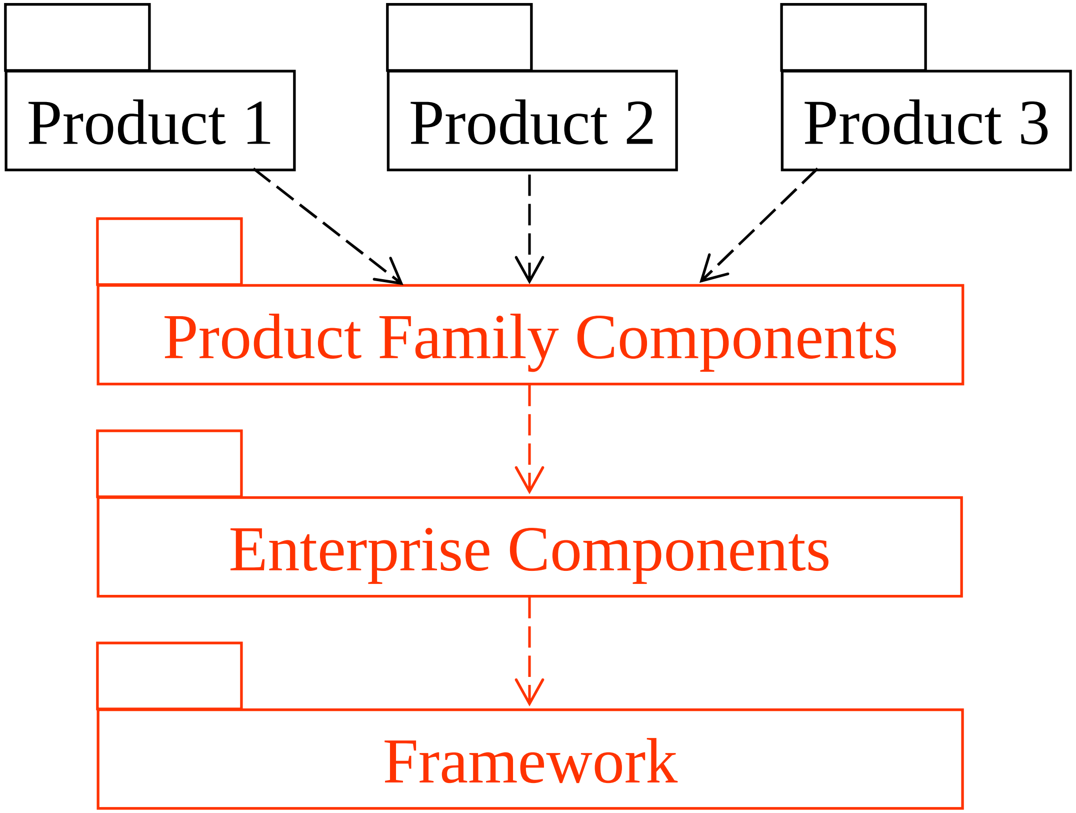
Customized Code



Framework

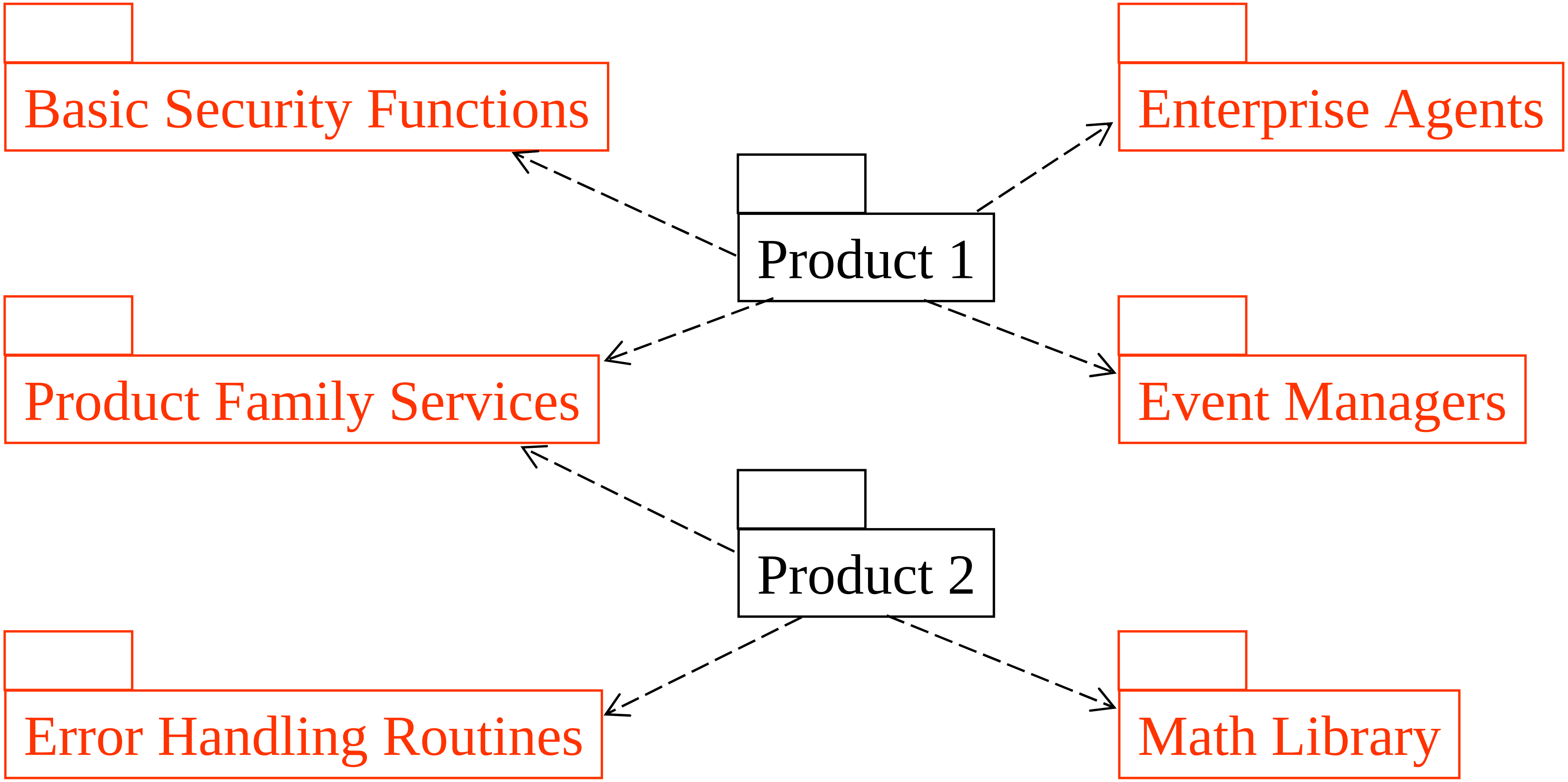

Framework
Eclipse
UML 2 Libraries
Rational Software Architect
Rational System Developer
Rational Software Modeler

Base Functionality



Wyyzzk, Inc.

Libraries



Data & File Format

- Shared data model
- Same file format
- Ease of maintenance
- Ease of upgrades

Versioning & Bug Fixes

- Bug found in one version of one product
  - What if found in old version and already fixed in later versions
  - What if found in newest version of product
  - What if in all products
- How many old versions to maintain
  - Loss of customers if upgrade is required
  - Issue of new features in old product versions
- Version control software

Use Cases & other Requirements

- Shared between products in the family
- Specialized for different products
- Can be used to find common areas and variable areas of the software

Documentation

- UML Package diagram
  - Show subsystems, packages, libraries
  - Show common and specialized elements
- UML Class diagram
  - As needed to show important elements of the different subsystems
- UML Use Case Diagram
  - Show use cases for product family and specialized use cases
- ER Diagrams
  - Show the data model
- MS Word
  - Use Case Specifications
  - Constraints, Regulatory Requirements, Guidelines

Break

- As a class, review the given Product Family Architecture.

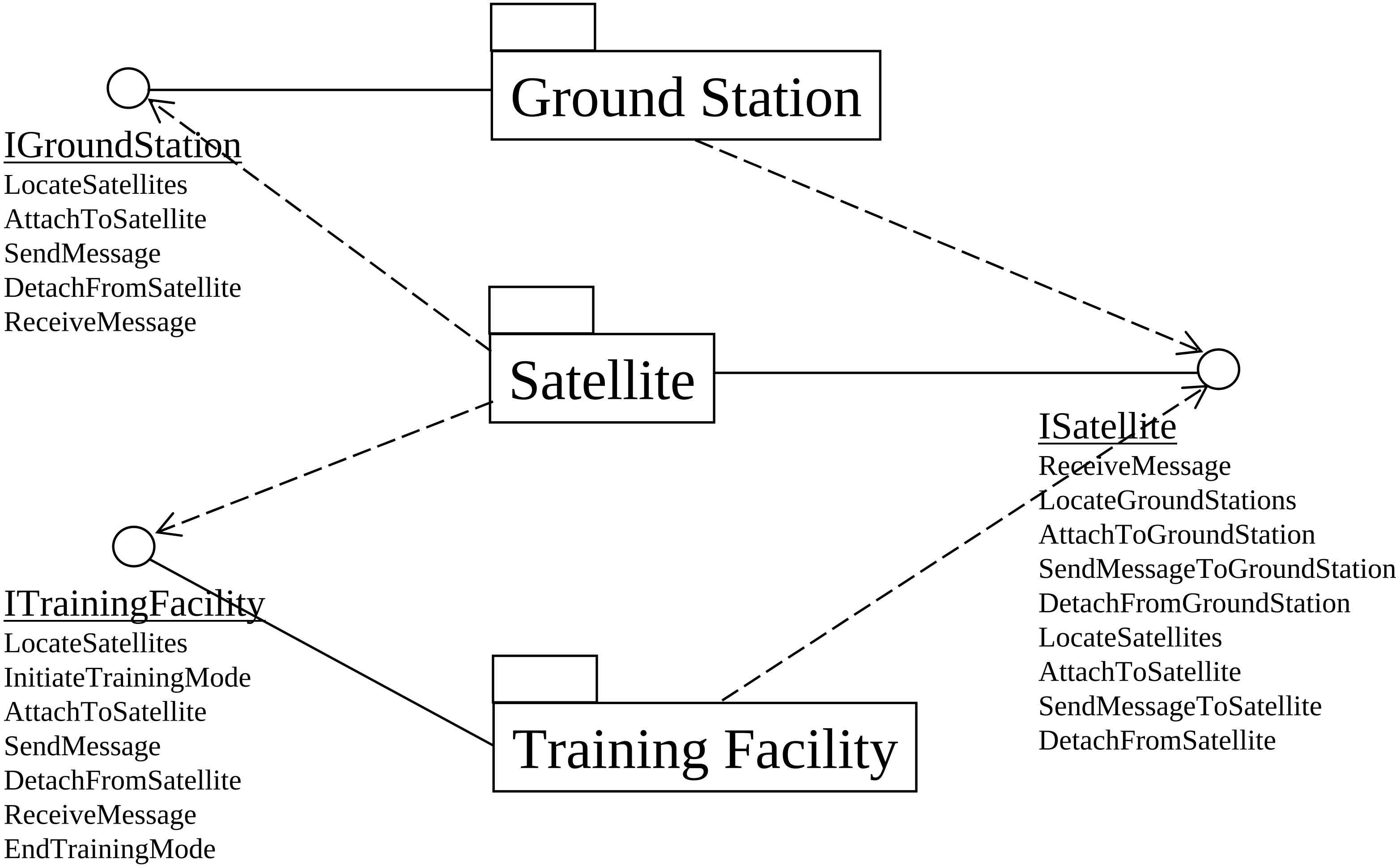
Program of Projects Architecture

- This concerns the architecture of a set of applications that are related because they are parts of a much larger application
- This is as much an architecture of the management of the projects as it is software architecture
- Common in defense industry, becoming necessary in IT

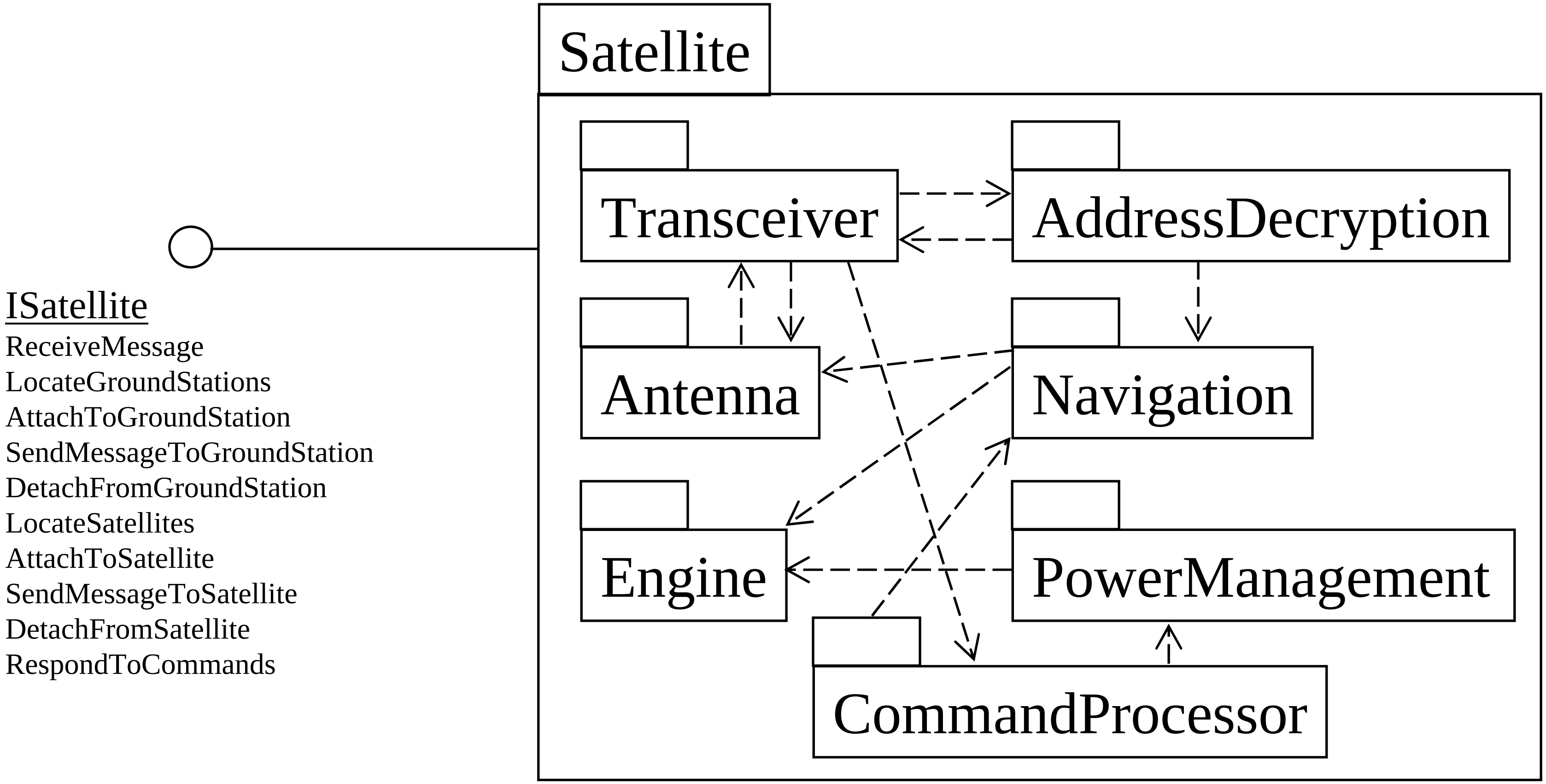
Program of Projects Architecture

- Define the overall project
- Create a high-level architecture
  - Components
  - Relationships
  - Assignment of requirements/use cases to components
- Make each component a project

Program of Projects Architecture



Program of Projects Architecture



Satellite

Transceiver

AddressDecryption

Antenna

Navigation

Engine

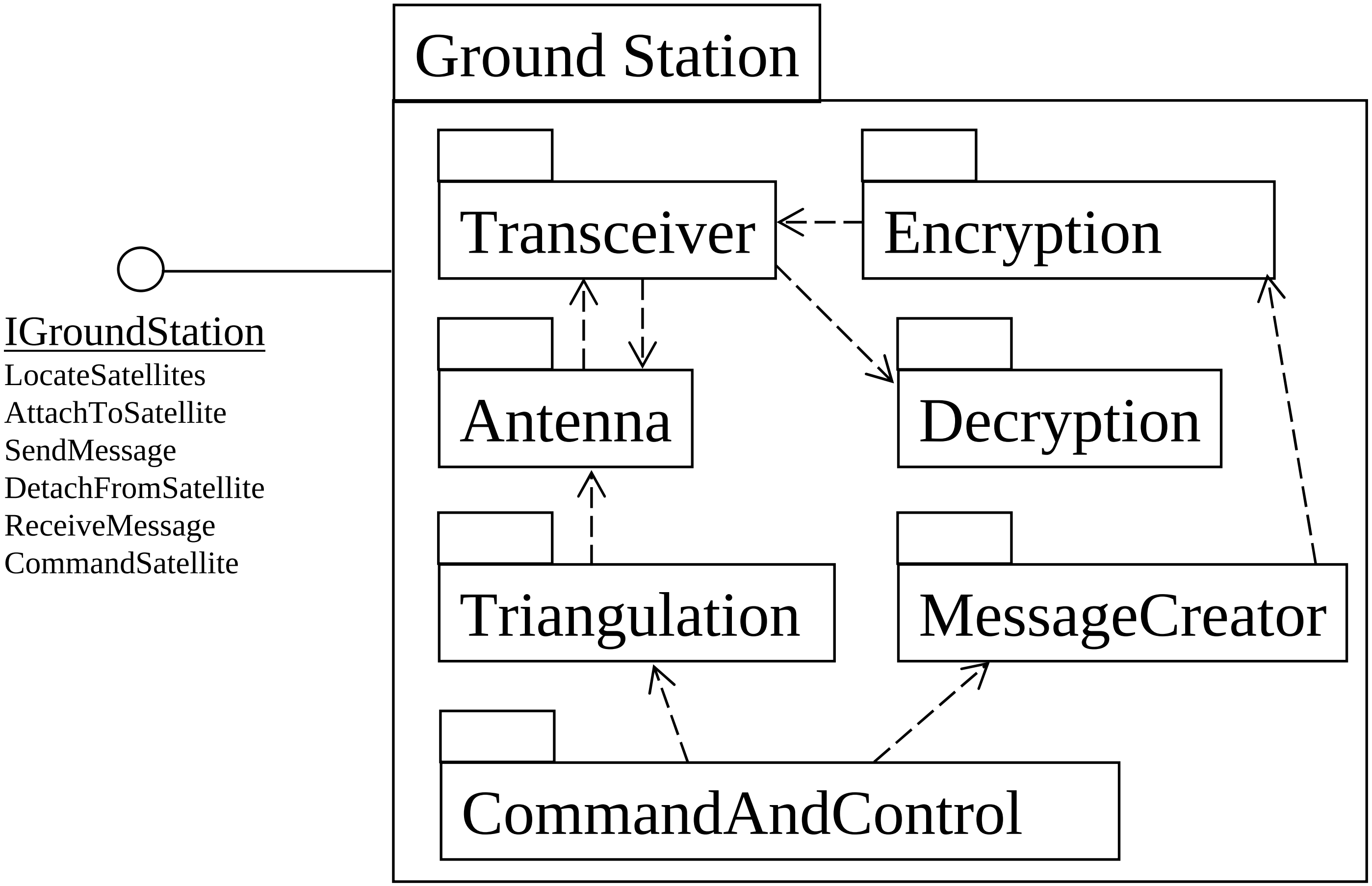
PowerManagement

CommandProcessor

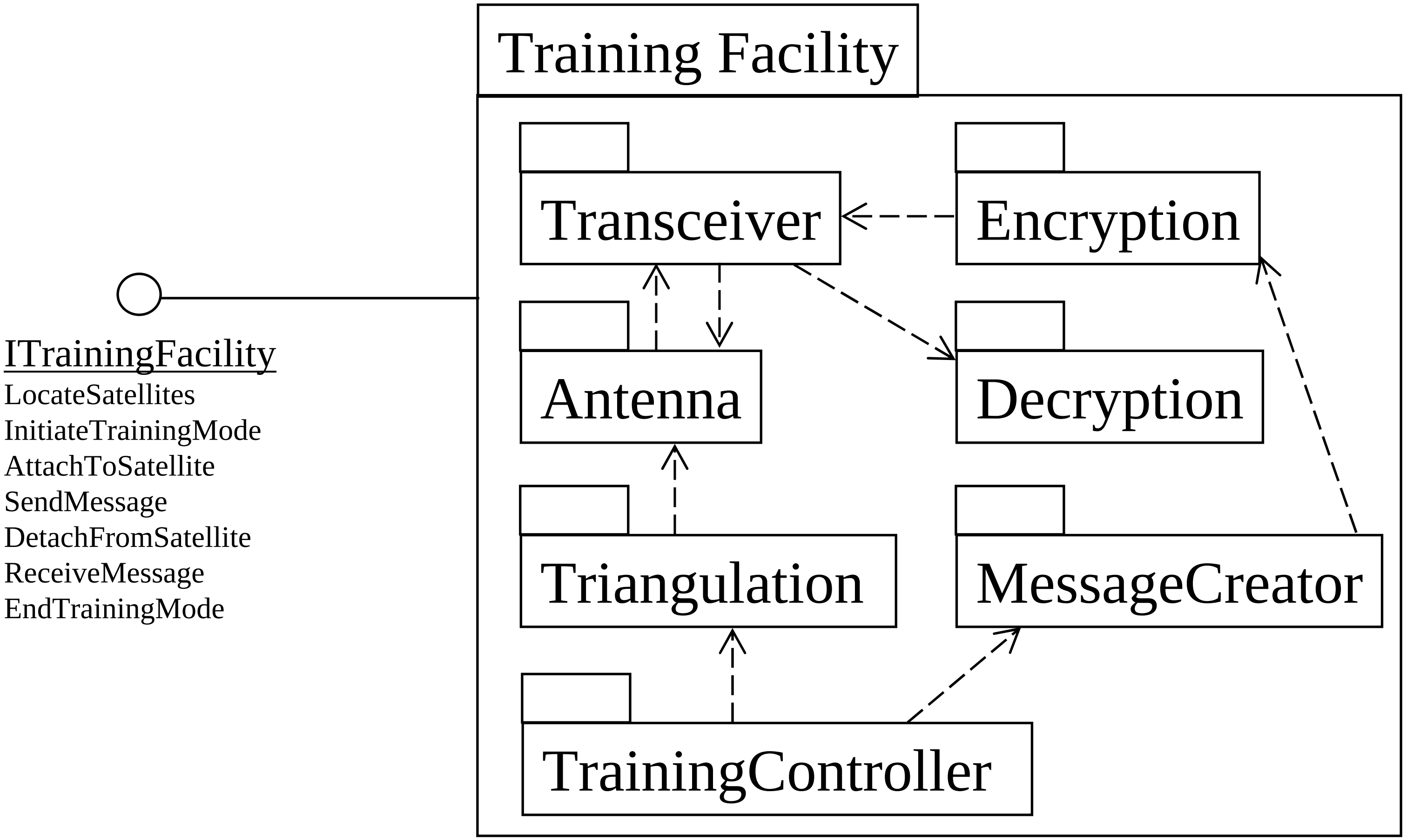
ISatellite

ReceiveMessage
LocateGroundStations
AttachToGroundStation
SendMessageToGroundStation
DetachFromGroundStation
LocateSatellites
AttachToSatellite
SendMessageToSatellite
DetachFromSatellite
RespondToCommands

Program of Projects Architecture



Program of Projects Architecture



ITrainingFacility

- LocateSatellites
- InitiateTrainingMode
- AttachToSatellite
- SendMessage
- DetachFromSatellite
- ReceiveMessage
- EndTrainingMode

Data & File Format

- Coordinate shared information between components
- Typically one component (project team) is the “owner” of the format of information

Shared libraries

- Shared libraries can be a good thing on a program
- Must be developed early, thoroughly tested, then NOT CHANGED
  - Changes propagate throughout the whole program

Use Cases & other Requirements

- Divided among components (project teams)
- Requirement changes may need to be negotiated between project teams

Documentation

- UML Package diagram
  - Show subsystems, packages, libraries
  - Show common and specialized elements
- UML Class diagram
  - As needed to show important elements of the different subsystems
- UML Use Case Diagram
  - Show use cases for program and for each component
- ER Diagrams
  - Show the data model
- MS Word
  - Use Case Specifications
  - Constraints, Regulatory Requirements, Guidelines

Break

- As a class, review the given Program Architecture.

Constraints and Issues

- Regulatory
  - Sarbanes/Oxley
  - HIPPA
  - Privacy laws
- Security
  - Limited access – physical, electronic
  - Encryption – data, communications
- Policy
  - Required performance
  - Required uptime
  - Fault tolerance
  - Budget for new hardware or software
  - Technical support of systems
  - Centralized vs distributed
  - Black-out periods for new releases

Constraints and Issues

- Licensing
  - Servers
  - Usage compliance
- Maintenance
  - Status monitoring
  - Mirroring & backups
  - Scheduled down-time for hardware/software updates
  - Run-time updates
  - Allocation of maintenance costs
- Technical Support
  - User queries
  - Problem resolution (possibly over many systems)

Program or Product Family Group

- Responsible for setting, documenting, and enforcing all program or product family level policies for projects within the program or product family
- Oversee project teams to verify that project architecture does not violate program or product family architecture
- Determine when program or product family architecture needs to change and how to change it
- Oversee all changes to program or product family architecture

Impact on project architecture

- Requirements of program or product family architecture are also project requirements
- Need to negotiate problem areas with other project teams within a product family or program, in addition to negotiating problems with Enterprise level groups
  - Corporate Security
  - Technical support (for users)
  - Maintenance group
  - Accounting
  - Regulatory groups
  - Enterprise architecture group
  - Database architecture group

Impact on Project Architecture

- Patterns selected at the program or product family level have to be followed at the project level
  - For example, if the program uses SOA, then your project will be designed around services
- Constraints and regulatory requirements set at the program or product family level have to be followed at the project level
  - For example, privacy laws require personal information to be encrypted. If your project in any way uses personal information, you will have to deal with decryption and encryption, and possibly only certain people on your team will have access to that data.

Impact on Project Architecture

- Because of the need to have more people involved with your project, your schedule will be longer
  - Other project teams, program or product family groups, Enterprise architects, security people, regulatory agencies, and so on
- In the defense industry, you can add the need for clearances to the process
  - It is possible that team members have different clearances and need to know, and that you will have to put processes in place to ensure everyone has the appropriate access

Consequences

- Perception of too much overhead
- Lack of these structures generally leads to project failure
  - Too many meetings or too few
  - Too many people making decisions or too few
  - Lack of understanding of complete project

Summary

- A program or product family architecture concerns applications that are related because they are part of a program or because they are variations on the same product
- In both architectures, consider what is common among the various projects and what is different when creating the architecture
- Base the architecture on the kind of anticipated changes in the program or product family
- Constraints, issues, and policies set at the program or product family level will effect your project architecture
- You may find that many groups will be involved in your project. For example, enterprise architecture, corporate security, regulatory agencies, technical support, maintenance, and database architecture, as well as program or product family group and other project teams within the same program or product family.