


Announcements

- 
- Schedule update
 - Participation Cards
 - Team Assignments
 - Assignment #1 (Course Project Milestone #1)
 - Today's Lecture
 - ▣ Architectural Views

PRODUCT ARCHITECTURES VIEWS -- DERIVATION, MEASUREMENT AND MANAGEMENT

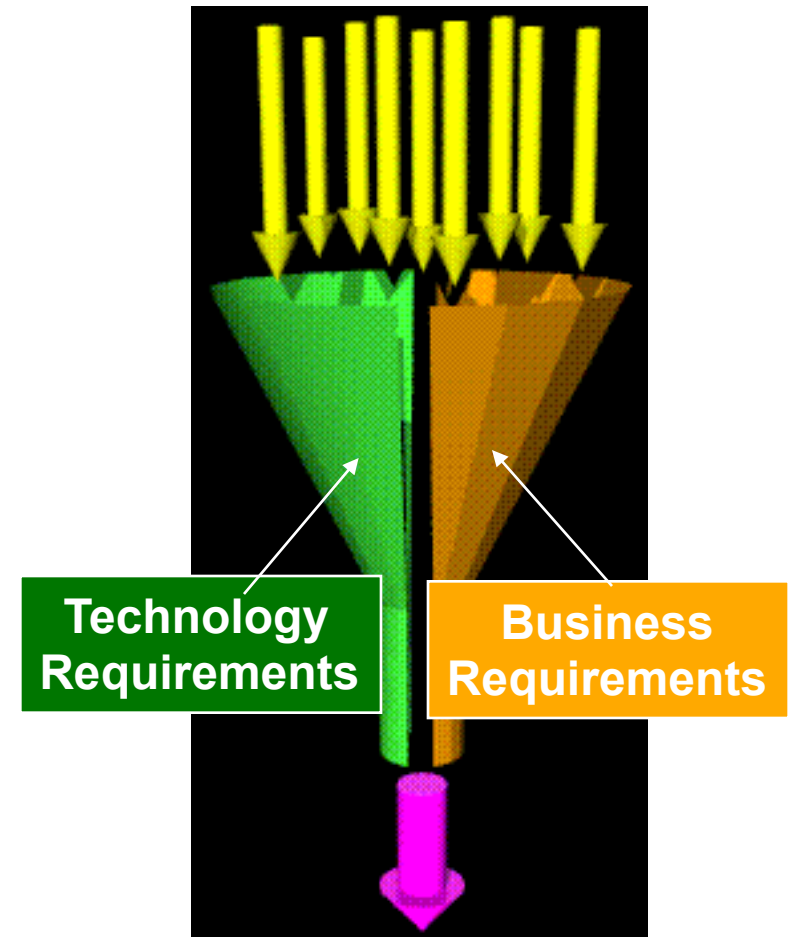
Suzanne Barber

Copyright ©2019 Suzanne Barber
Proprietary and All Rights Reserved.

Duplication or distribution without the expressed written approval of Suzanne Barber is prohibited.

The Great Divide of Requirements types drives Architectures and the separation of concerns. [Barber]

- Premise: What services must be performed/accomplished changes less often than how services are implemented
- **Business Requirements:** “What is required”
 - What services/tasks
 - What data
 - What kind of resources
- **Technology Requirements:** “How implementation is delivered”
 - technology-specific
 - Includes non-functional and installation requirements



ONE SYSTEM ARCHITECTURAL DESIGN..... MANY ARCHITECTURAL VIEWS

Different methodologies delineate architecture types (views) differently. Yet, have many similarities.

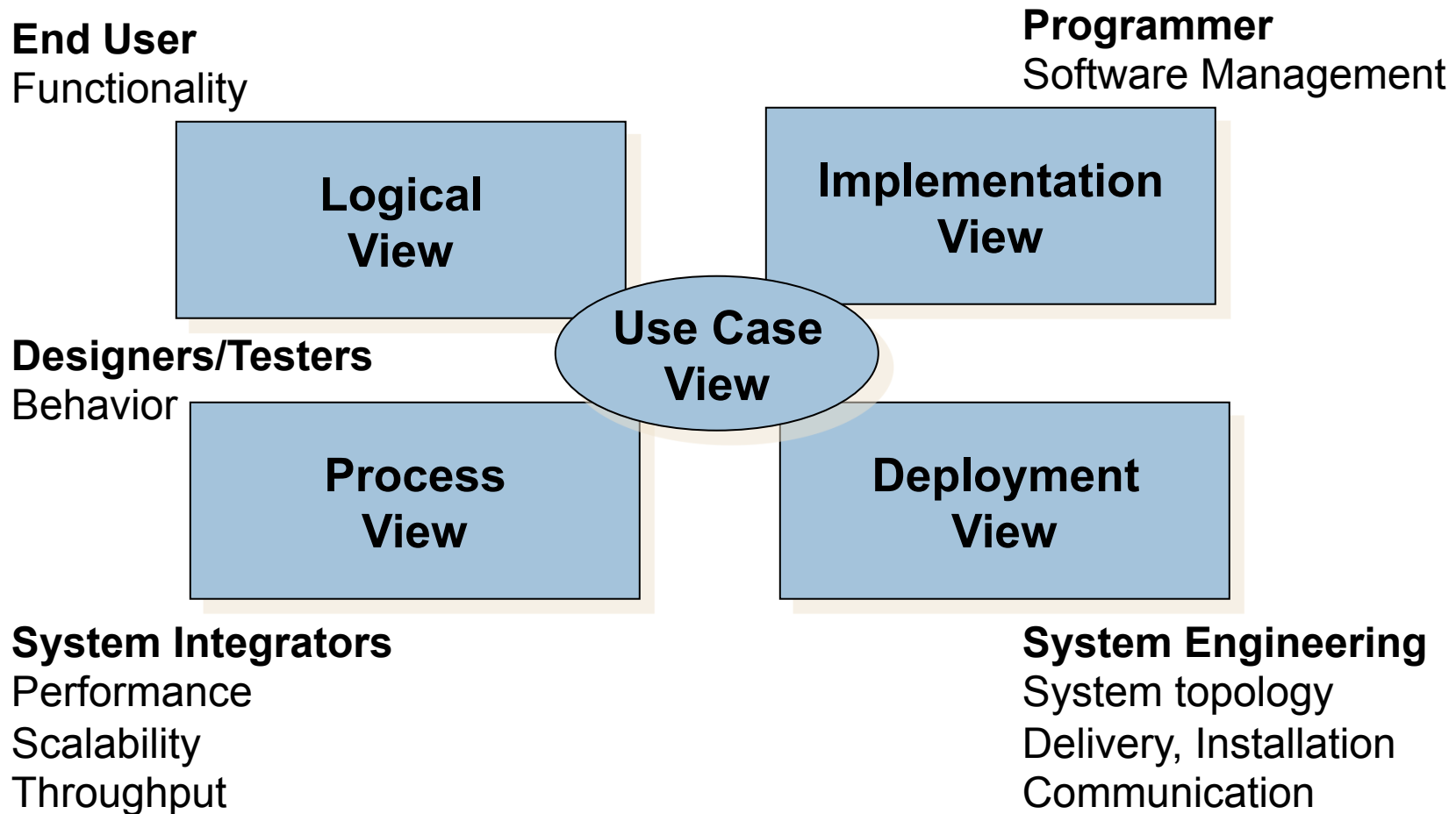


- 4+1 Architecture Views
- AWAREness 3d Architectures

Which Architectural View is most useful ? Why?

- As we review these different types of architectural views, keep notes about which views would be most useful to....
 - ▣ **You** to accomplish the work you do?
 - ▣ **Management?**
 - ▣ **Customers?**
 - ▣ **Users?**
 - ▣ **Developers? Builders?**
 - ▣ **Third party contractors?**
 - ▣ **Product Test team?**
 - ▣ **Marketing team?**

4+1 Architecture Views [Krutchen]



4+1 View of Architecture “Design” [Krutchen]

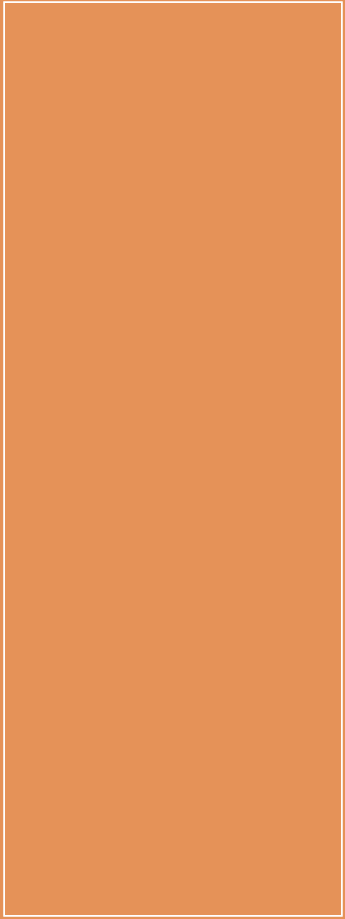
- Logical View
 - ▣ Functionality of the system
 - ▣ Abstraction of design model represents the logical structure of the system in terms of subsystems and classes (“deliverers” of functionality)
- Implementation View
 - ▣ System description relevant to implementation: (for software, that is source code, libraries, object classes, etc.)
 - ▣ Static View of implementation pieces, not how each interacts.
- Process View
 - ▣ Describes operations of the system (extremely important for systems with parallel tasks, interfaces with other systems, multi-threading).
 - ▣ View helps to identify problems such as race conditions or deadlock, throughput and performance issues.
- Deployment View
 - ▣ Allocation of the Implementation View to the Supporting Infrastructure (e.g. OS, computing platforms).
 - ▣ Not concerned with What interaction are but rather concerned with the existence of interactions and constraints when two systems meet.

4+1 View of Architecture (cont'd)

□ Use Cases *[Jacobsen, et.al.,]*

- ▣ Describe the intended system behavior as opposed to understanding the behavior of the business the system will operate within.
- ▣ Describe the System's behavior from the perspective of how the various users interact with the system to accomplish their objectives.
- ▣ Describes a sequences of actions a system performs that yields a result of value to a particular actor.
 - What the system does for the user.
- ▣ Use Case Model includes
 - All Actors of the system and all Use Cases by which Actors interact with the system.

Which views would be most useful to....

- 
- ▣ **You** to accomplish the work you do?
 - ▣ **Management?**
 - ▣ **Customers?**
 - ▣ **Users?**
 - ▣ **Developers? Builders?**
 - ▣ **Third party contractors?**
 - ▣ **Product Test team?**
 - ▣ **Marketing team?**

AWAREness System Blueprint “views” help separate concerns and conquer complexity.



Business Blueprint (Version.Release)

- BB version reflects current state of approved, verified, accepted Requirements (scenarios, functions, I/O, NonFunctional)
- BB version.release is the portion of Business Blueprint version to be satisfied at a Deployment Milestone



Solution Blueprint

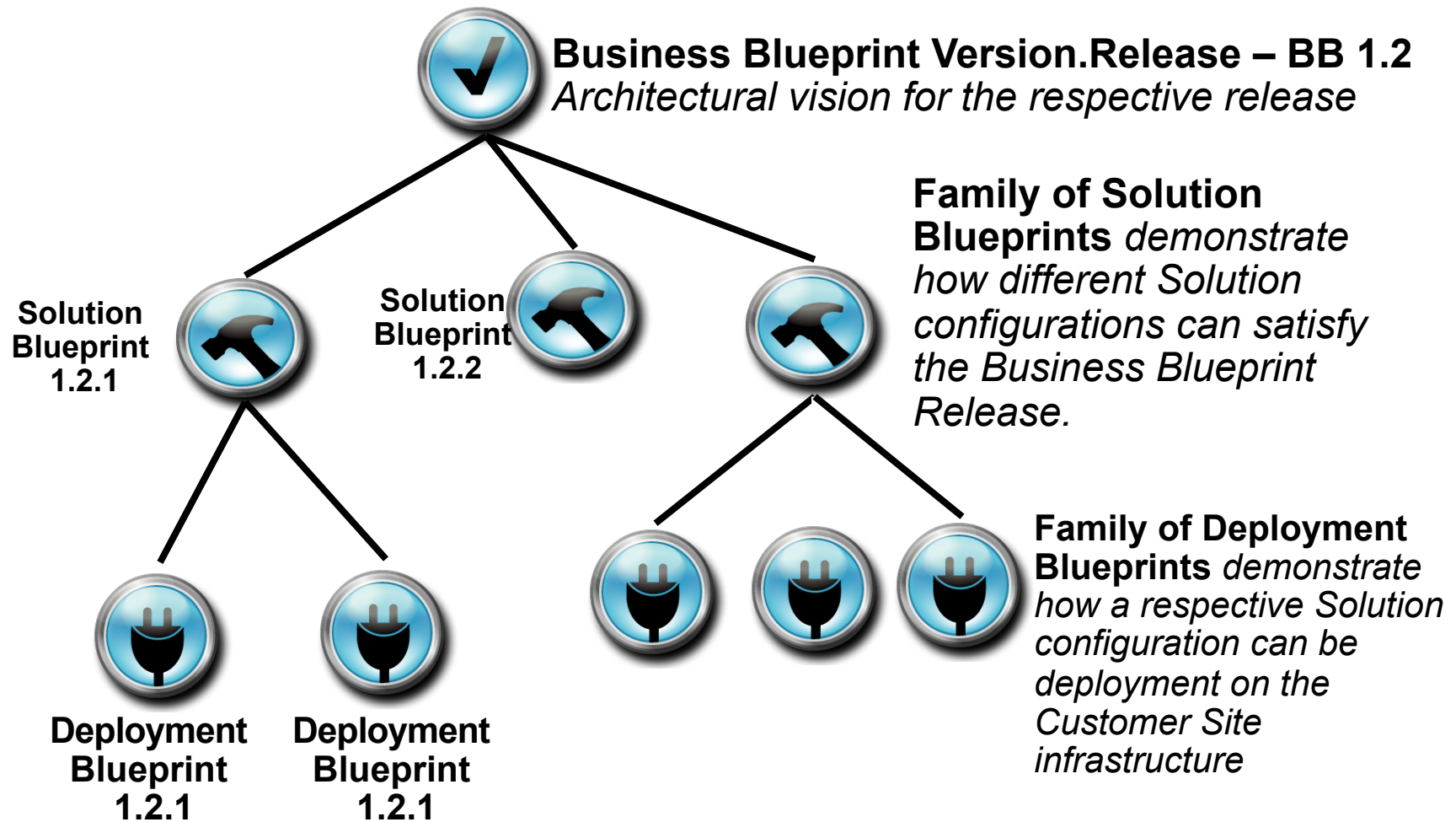
- Configuration of Solutions
- Solutions selected to comply with Business Blueprint Version.Release



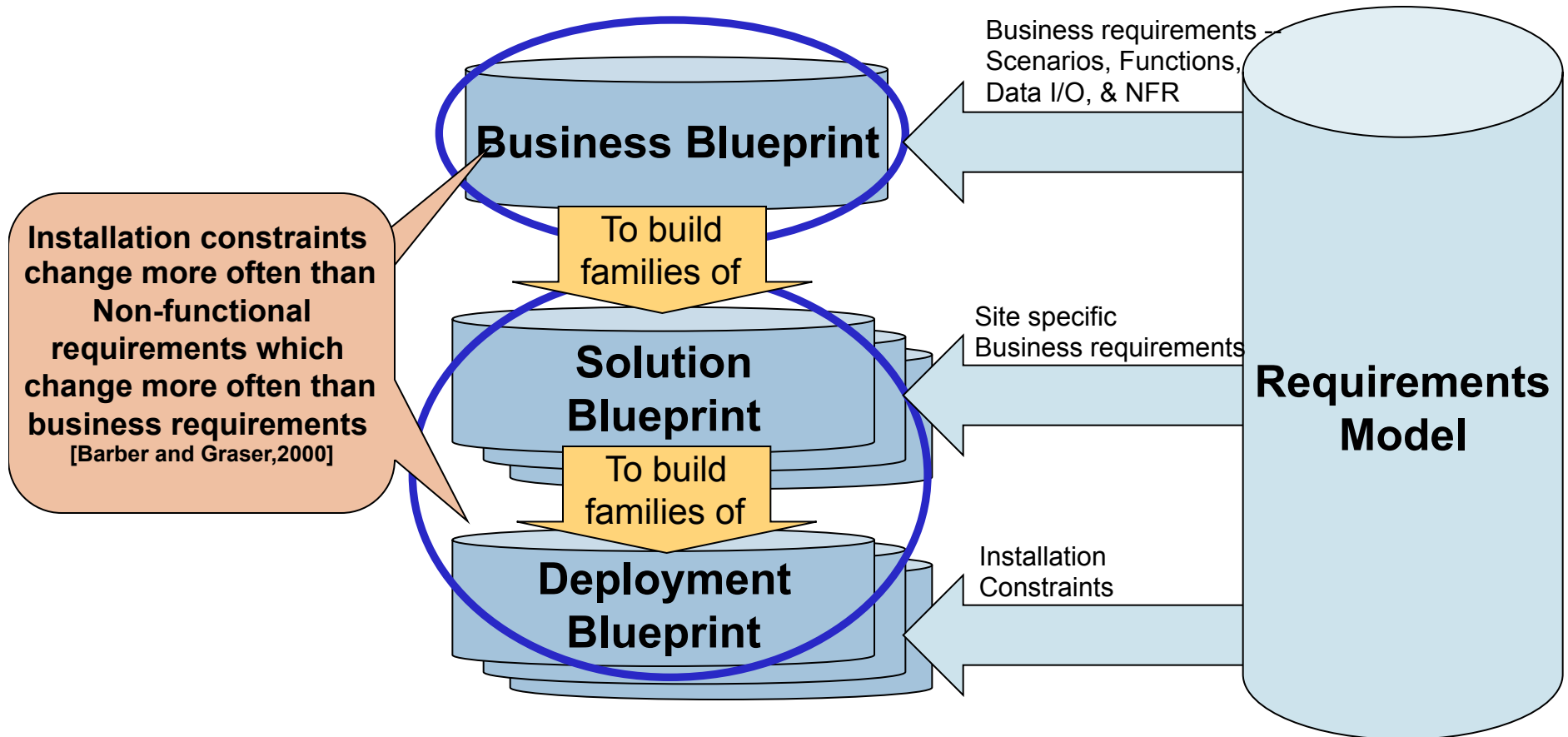
Deployment Blueprint

- Solution Blueprint installed in a potential deployment environment (solutions assigned to a physical environment, e.g. HW, SW)

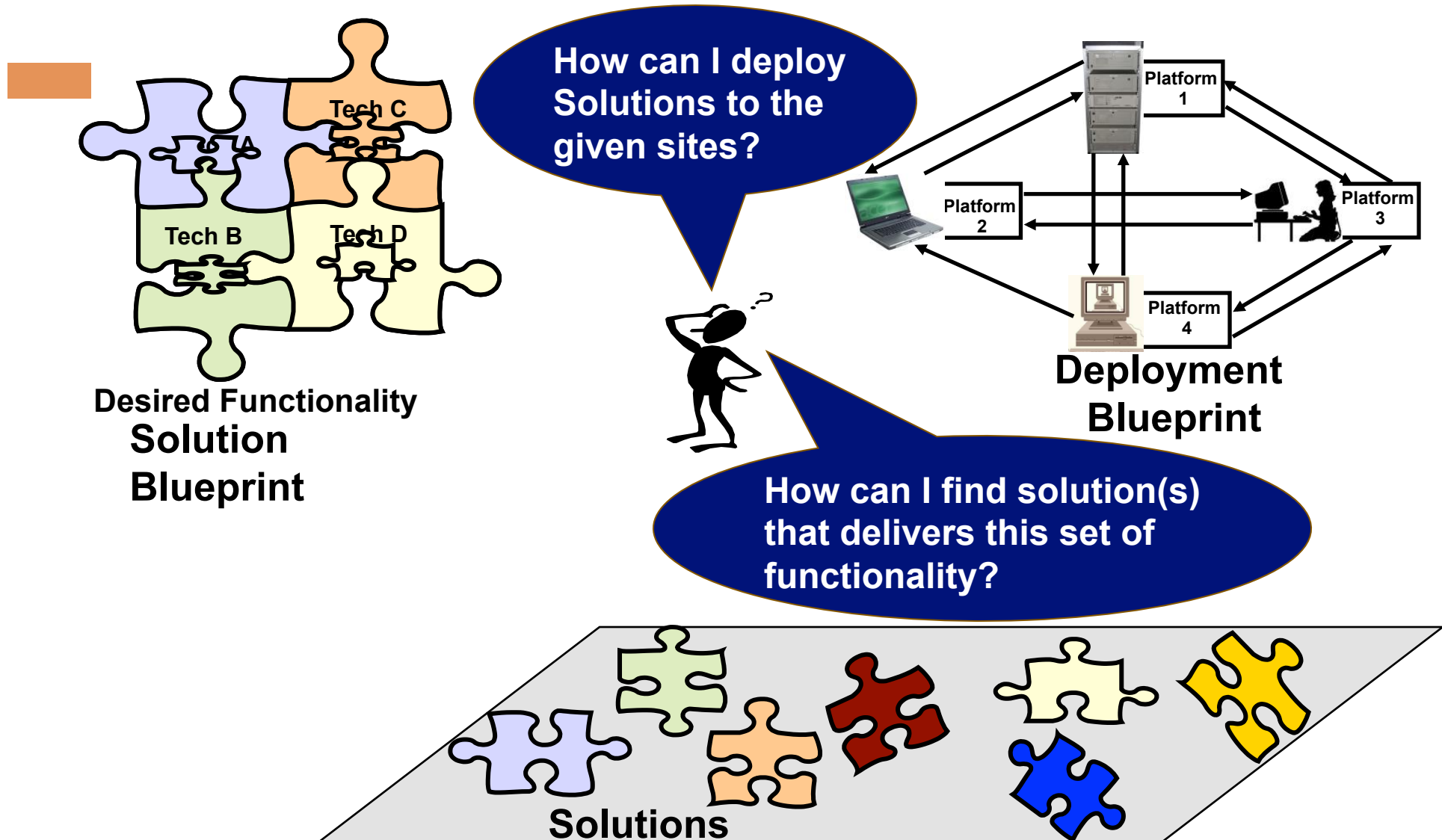
Building “families” of AWAREness blueprints permit powerful comparative analysis to explore the design space and a range of “what-if” scenarios.



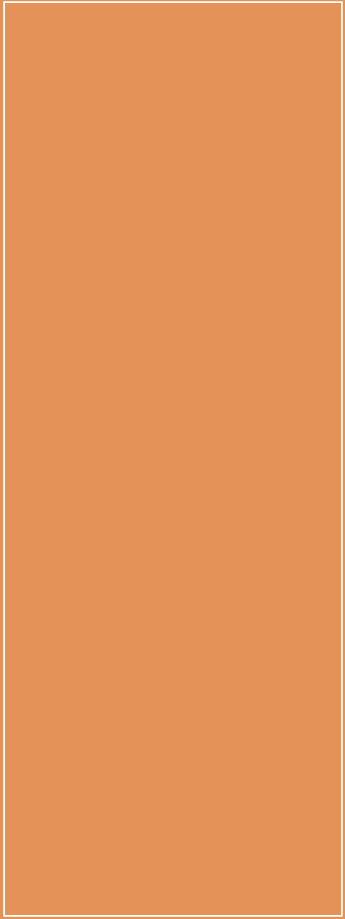
Requirements evolve at different rates, thus architectures evolve at different rates



Difference between Solution & Deployment Blueprints



Which views would be most useful to....

- 
- ▣ **You** to accomplish the work you do?
 - ▣ **Management?**
 - ▣ **Customers?**
 - ▣ **Users?**
 - ▣ **Developers? Builders?**
 - ▣ **Third party contractors?**
 - ▣ **Product Test team?**
 - ▣ **Marketing team?**

AWAREness Take-Away Messages

- Important that Architecture Views separate the “What” vs “How” to support ...
 - ▣ Evolution and Change
 - ▣ Reuse
 - ▣ Building families of designs
- Evaluation of “compliance” promotes quality, rationale, traceability, and customization (if needed)
 - ▣ Does BB comply with requirements?
 - ▣ Does SB comply with BB?
 - ▣ Does DB comply with SB?