Right – In building trades architecture they have one of the same problems we do in software architecture – getting everyone to visualize what it will be like before it exists. This example from http://www.alibaba.com/catalog/11454755/Architecture_Consulting.html.

What is Software Architecture?

CSSE 574: Session 8, Part 3

Steve Chenoweth
Phone: Office (812) 877-8974, Cell (937) 657-3885
Email: chenowet@rose-hulman.edu

This discussion

- Acknowledgements for RHIT's software architecture curriculum
- Nature of software architecture, from Bass, Ch 1-4

Acknowledgements

Some of the material in these slides, as usual, will be taken from *Software Architecture in Practice,* by Bass, Clements and Kazman.

Built on 7 years of RHIT arch classes taught by Steve, Mark Ardis, Lisa Kaczmarczyk, Shawn Bohner, Curt Clifton, and Chandan Rupakheti.

Mark (left, in chair), and Lisa.- also practicing architects.

Outline

- Definitions
- Reference Models and Architectures
- Consequences of Architectural Choice
- Promoting Reuse
- Architectural Structures

Definitions of Software Architecture (1/3)

- Many different definitions of software architecture
- See the SEI website “Community definitions” at: http://www.sei.cmu.edu/architecture/definitions.html for over 100 (at least one provided by my old groups!)

Definitions of Software Architecture (2/3)

From Bass et al.:

"The software architecture of a program or computing system is the structure or structures of the system, which comprise software elements, the externally visible properties of those elements, and the relationships between them."

Definitions of Software Architecture (3/3)

- So Bass's definition of architecture means *the design*, as described and in action:
  - As seen on paper or in models
  - The reality, used by the architect & others, the code that does what the design says
  - What the design "looks like" as a result
  - How it works to achieve that result

Bass's Definition of Software Architecture vs Arch Bus Model

- Not the same!
- To achieve the "architecture," the architect has to do a lot of other activities, like:
  - Interact with lots of people to figure out the problem and get solution ideas
  - Interact with lots of people to sell the solution (especially to the other developers who'll implement it)

Worth comparing vs an older field of architecture…

- From CMU School of Architecture's home page, 2003-4:

:. Career Paths

The School of Architecture at Carnegie Mellon University focuses on the fundamental knowledge bases you will need for an exciting multidisciplinary career dealing with the built environment. From products, to buildings, to cities, architects play a role in designing the future

- Private professional practice
- Professional practice within industry, business, government, non-profits
- Strategic, Regional & Master Planning
- Building Finance, Banking, Development
- Design Build & Construction Space
- Planning, Interior Design Industrial
- Design Facilities & Real Estate Management
- Historic Preservation, & Rebuilding Communities
- Entertainment & Communications
- Fine Arts

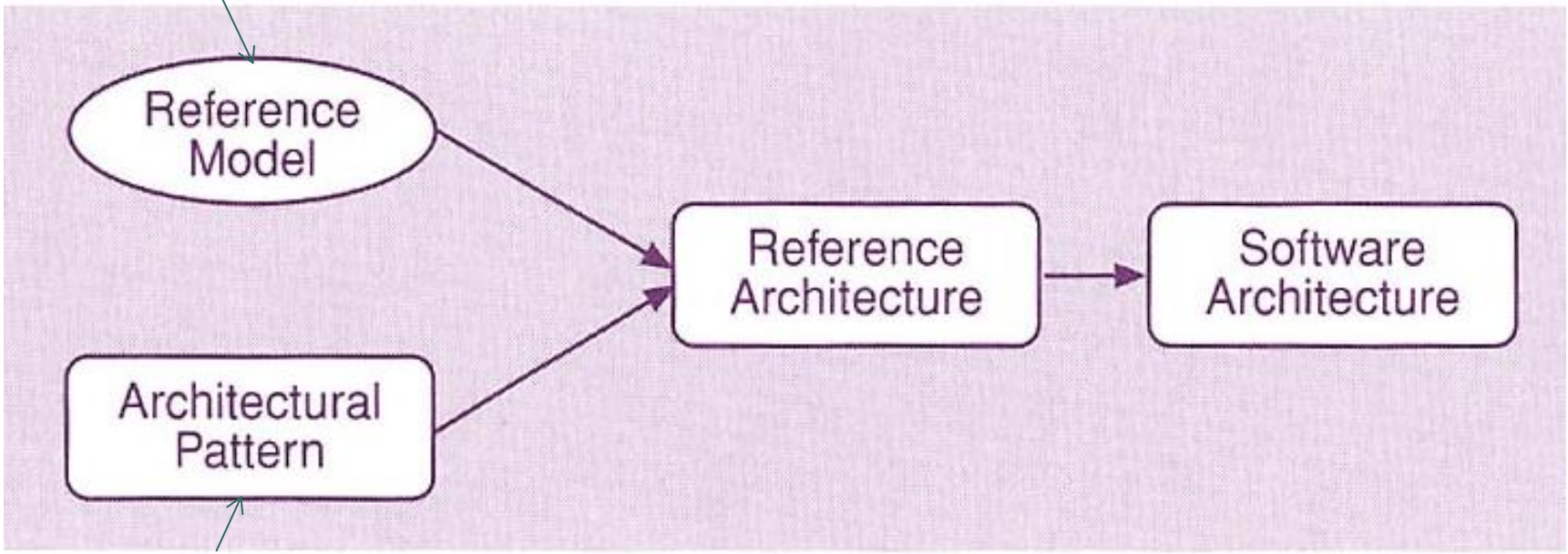
Which brings out a strategic issue about the software business:

- In other fields like building architecture and civil engineering, the architect also is responsible for project success.
  - The contractors, who do the work, work for them.
- In most software projects, the top people are mostly business majors.
- So, maybe we're still a young profession…

Jack Blake of Atlanta-based TVS & Associates architect of the RSA tower, stands by the tower construction site. From http://blog.al.com/pr/2007/05/architect_rsa_tower_a_spectacu.html .

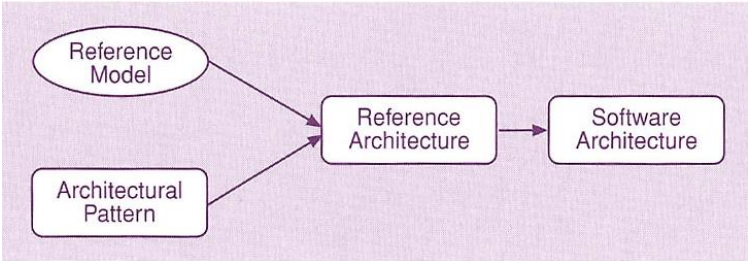
Some Related Terms (Bass's)

(Mostly about *Functionality*)



(Mostly about *Quality Attributes*)

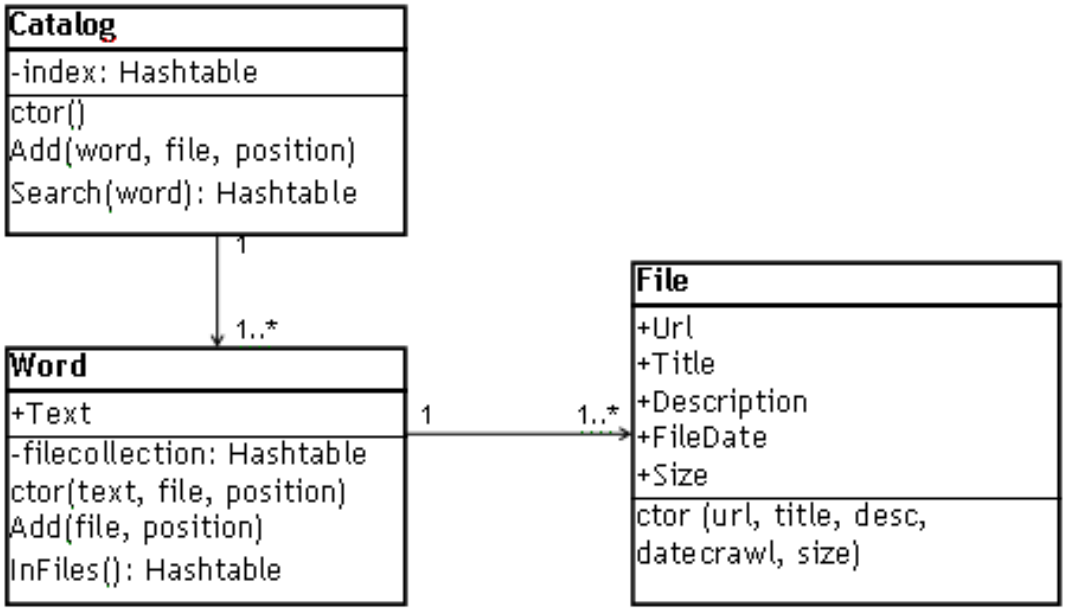
Reference Model (1/5)



- Division of *Functionality* with data flow between pieces, based on the problem you need to solve
- Example:
  - Compiler reference model includes a description of parts and data flow between them
  - Not the same as the “pattern,” which is a way to enable that particular flow.

Reference Model (2/5)

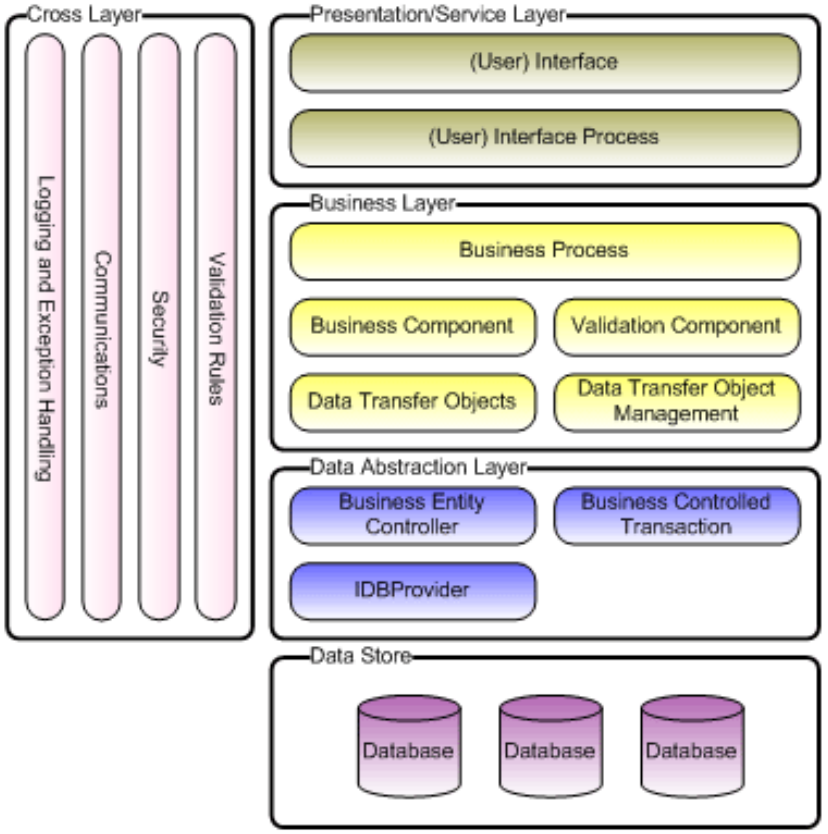
- Object Models in UML, for example:



Example from http://www.conceptdevelopment.net/Search/SearcharooV1/.

Reference Model (3/5)

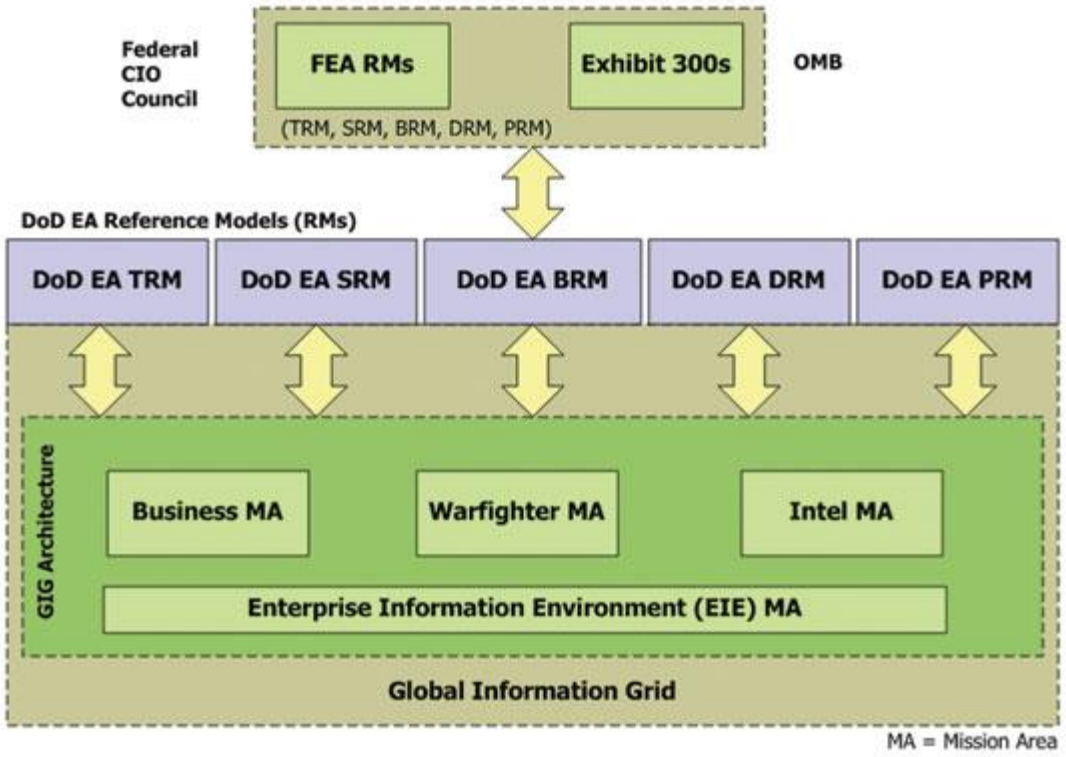
- Another example: Capgemini's "CDAF architectural reference model" for information systems they sell.
- Note the architectural style ID'ing of essential business components.



From msdn2.microsoft.com/en-us/library/bb286670.aspx.

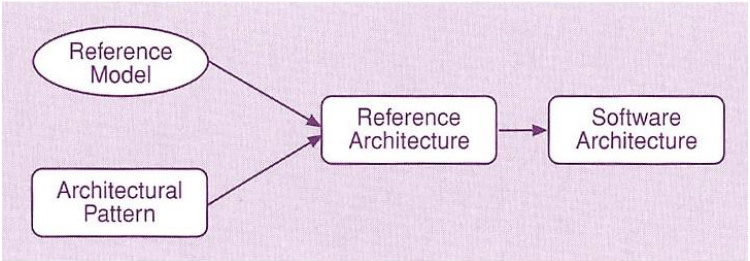
Reference Model (4/5)

- Another example: The DoD net-centric reference model.
- Note the functionalities & flows shown.
- This one's a ref model of ref models!



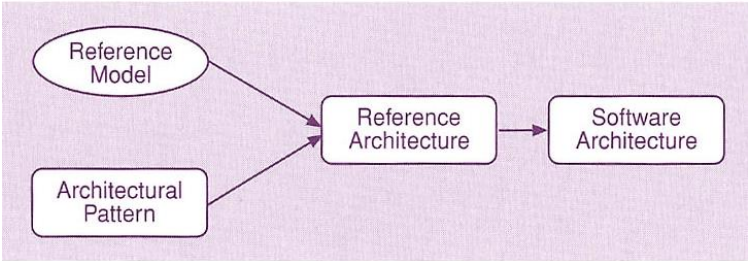
From https://www.softwaretechnews.com/stn_view.php?stn_id=7&article_id=12.

Reference Model (5/5)



- Where do reference models come from?
- For your project in this class – documents that were created by your clients! –
  - The use cases show interactions with users (and perhaps other systems)
  - The supplementary spec shows other functional and quality attribute interactions.
- In your arch doc, the reference model –
  - Is seen as a short version of the req with interactions and connections highlighted.
  - Matches the granularity of the prob stmt
  - Can be the *high-level OO model* of the system (like example 2/5)!

Architectural Pattern

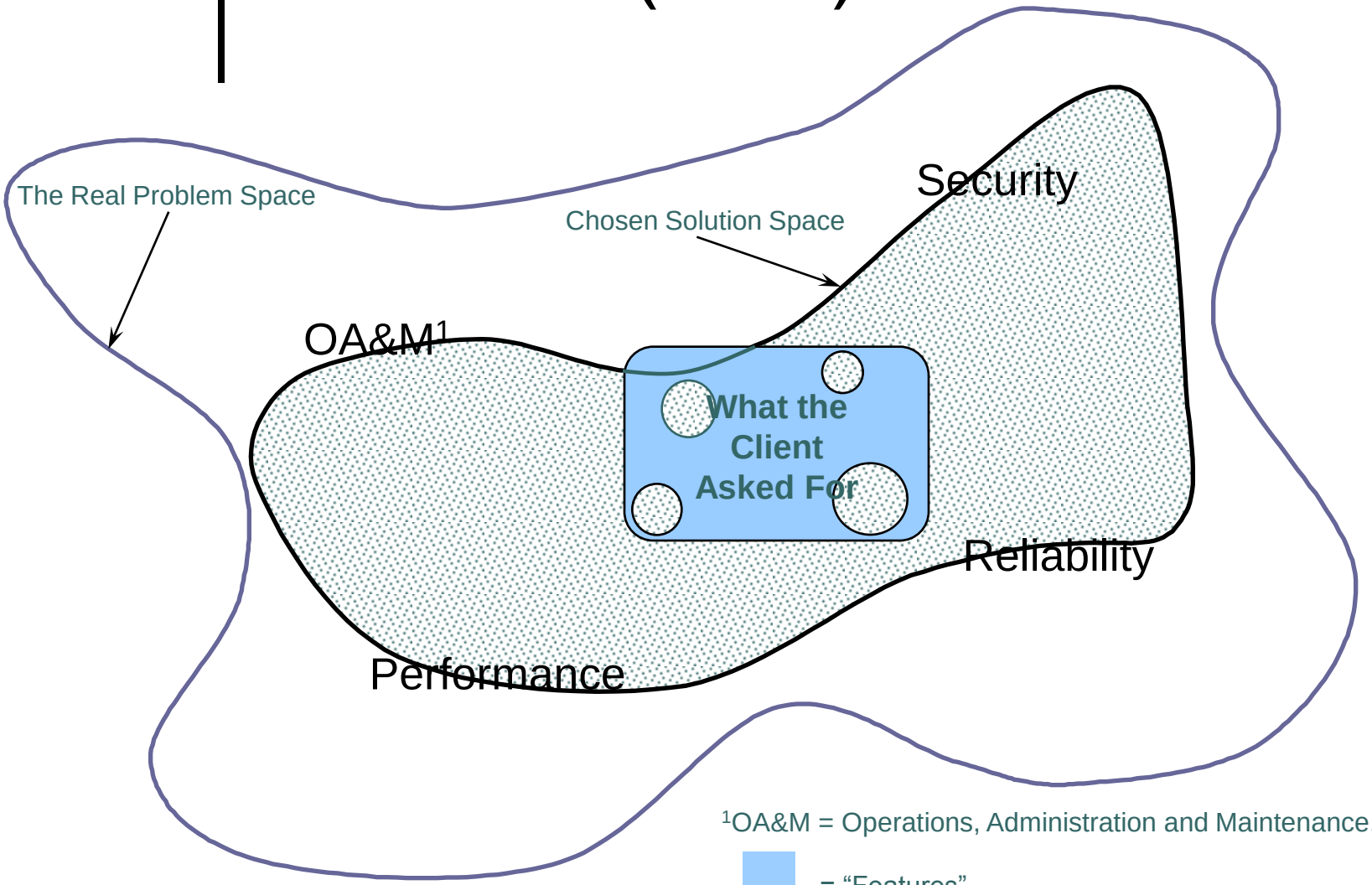


- Focused on achieving *Quality Attributes*
- These we'll see (article by Garlan & Shaw)
- Also known as "architectural style"
- Description of element and relation types with a set of constraints on how they may be used
- Examples:
  - Pipe and Filter
  - Client-Server

A caveat…

- Getting the Architectural Pattern right depends on how much you and your client know about the quality attributes.
- Clients often don't even want to talk realistically about QA's.
- The "requirements churn problem" is a killer for architecture -- adding global, architectural properties is hard!

A caveat (cntd)…



[1]OA&M = Operations, Administration and Maintenance.

= "Features"

= Overall, Including "Non-Functional" Requirements, etc.

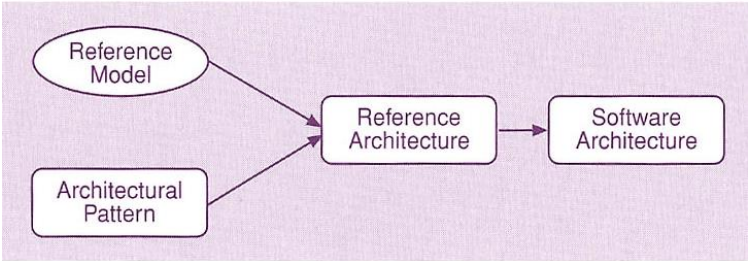
What gets built =

A caveat (cntd)…

- Here's how it can turn out:

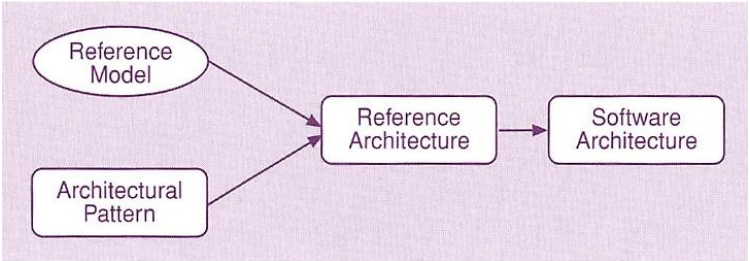
Escher's "Belvedere" in LEGO®. From www.andrewlipson.com/escher/belvedere.html .

Reference Architecture (1/2)



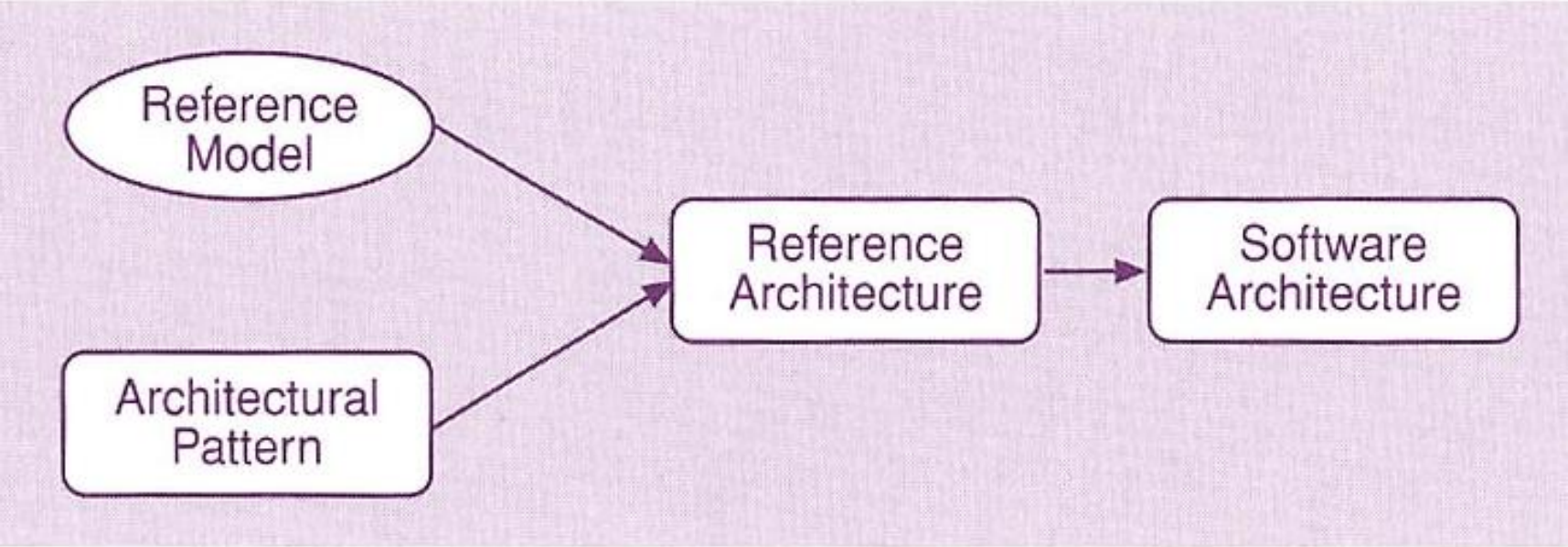
- Combine the first two!
- Reference model mapped onto software elements and data flows between them
  - Like. “Plug the names from the reference model into the boxes on the pattern” (as a start)
  - E.g., mapping “compiler” ref model onto “pipe and filter”
- Still not a software architecture -- too generic… but it’s a start

Reference Architecture (2/2)



- So, in your project, as architects, you:
  - Abstract a “main flow” of action from the req you get, as the “reference model”
  - Pick an arch pattern to map that onto
  - See how the interactions of the ref model look as your new ref arch!
- In your projects, this will be interesting to try, after the fact…

Putting Them All Together



What would you do next? At this point, you would start to loop back on the sources of all this, like the requirements, your client, other designers, and the people who will have to build it. (Notice the social ingredients!)

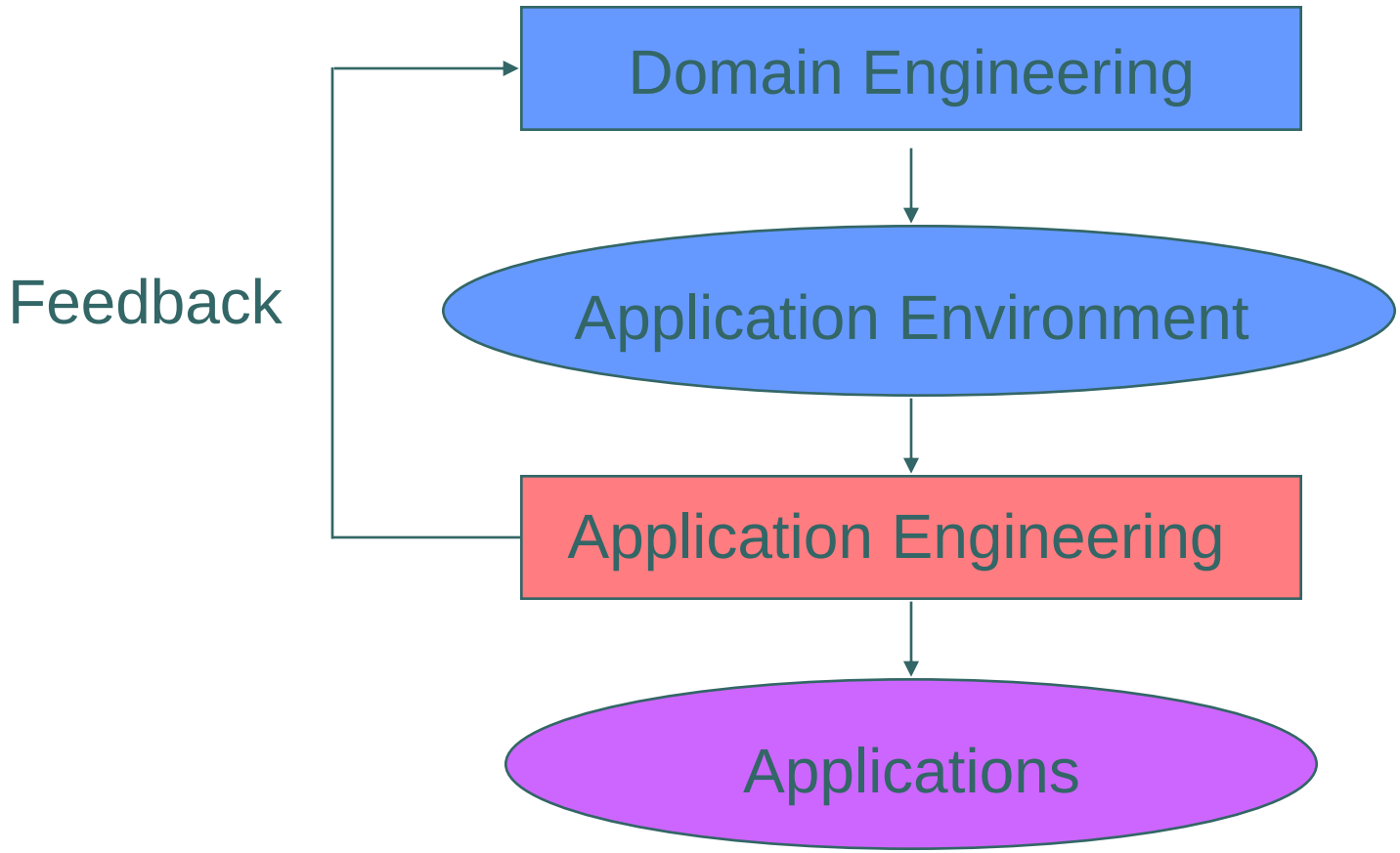
Consequences of Architectural Choice

- Defines constraints on implementation
- Dictates organizational structure
- Inhibits or enables system's quality attributes
- System qualities may be predicted
- Easier to manage change
- Helps in evolutionary prototyping
- Enables cost and schedule estimates

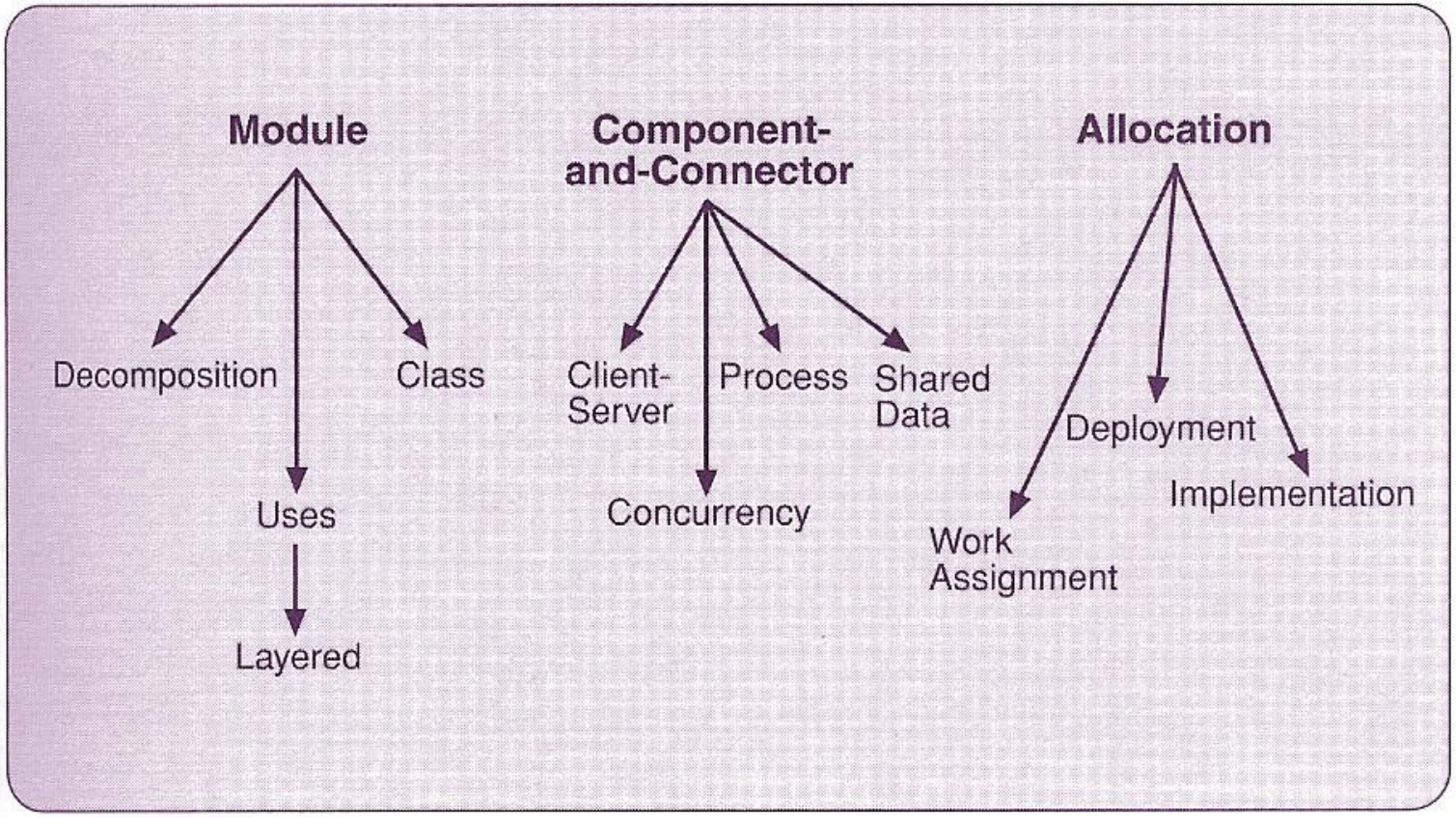
Architecture Promotes Reuse

- Product lines share a common architecture
- Externally-developed elements may be included
- Restrictions encourage reuse of design patterns
- Architecture can be basis for training

Product Line Engineering Model

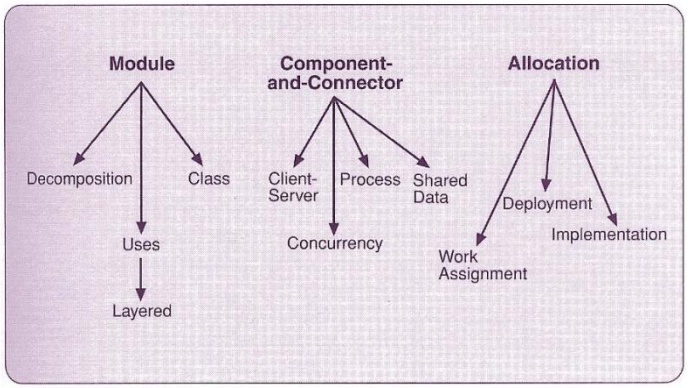


Architectural Structures



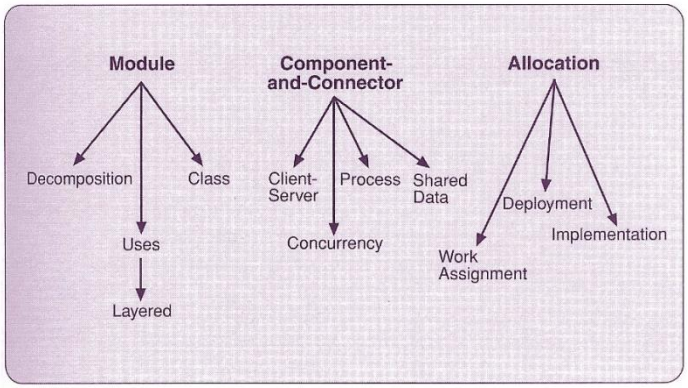
We'll be using all of these in your arch document!

Module Structures



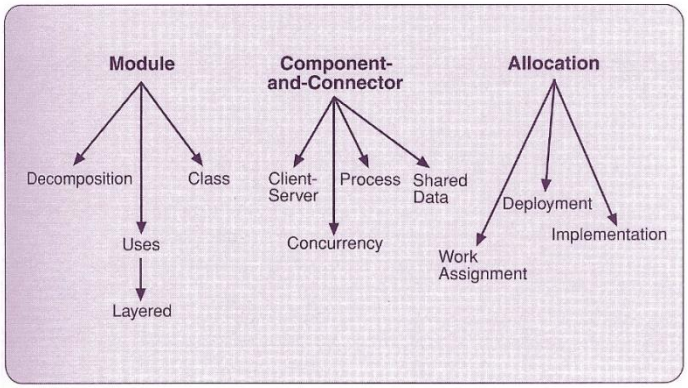
- Decomposition - contains
- Uses - calls
- Layered - controlled access
- Class - inheritance

Component-and-Connector Structures



- Client-Server
- Concurrency - parallel execution
- Process - synchronization
- Repository

Allocation Structures



- Work assignment - who does what
- Deployment - allocation to hardware (this is one the customer usually looks at a lot – why?)
- Implementation - mapping to files