

# Introduction to the course of Software Architecture

March 2014

Ying SHEN

SSE, Tongji Univeristy



# Simple case study

Three vendors have been asked to submit bids for work that tracks student credentials...

All vendors supply demonstration systems that:

- Seem to meet the requirements
- Are all within similar cost estimates.
- All other things being equal ... Who do you pick ?

You ask them to come in and describe their architecture so as to get a better understanding of the long term costs to maintain and adapt the system to your planned expansions.

# The vendors

The We Know Better. Com → TheWKB.com

- An medium to large experienced company that has developed a wide range of similar products over several years for a variety of customers.

The Minority Owned Business. Com → TheMob.com

- A small company that has developed a similar product and is looking to expand their business

The Wet Behind the Ears. Com → TheWBTE.com

- A grad students run venture that is looking to capture the contract in order to acquire start up funding.

# The Proposed Architectures

## TheWKB.com

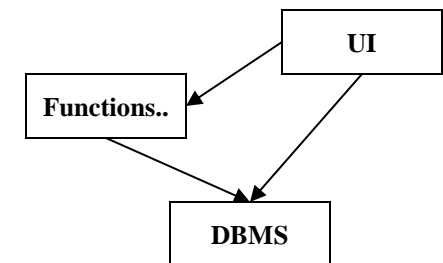
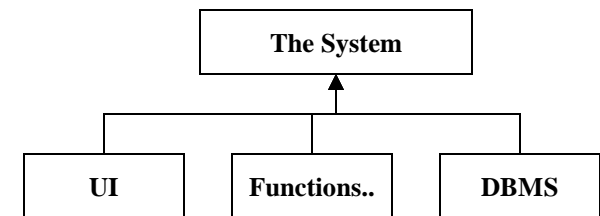
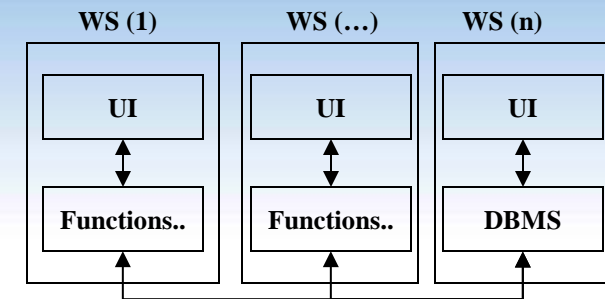
- depicts what they call an architecture by showing the allocation of functional pieces to hardware.

## TheMob.com

- depicts what they call an architecture by showing a functional block diagram.

## TheWBTE.com

- depict what they call an architecture by showing a logical view of the functional elements and their interconnection dependencies.

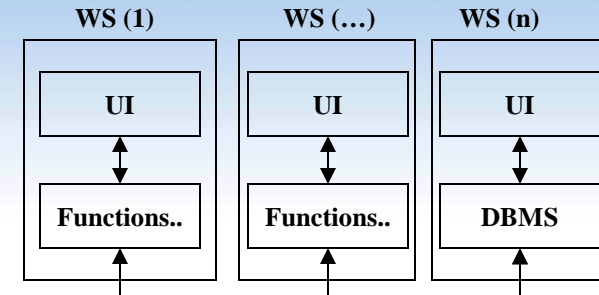


**These depictions and others have been used to describe architectures – So who is correct ? What's missing ?**

# So how do you even start to evaluate and compare the architectures against each other ?

How to determine (measure) which one is better at capturing:

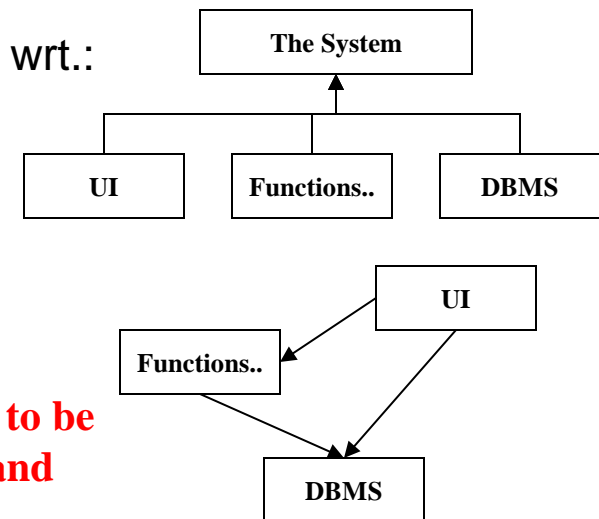
- Functional relationships ?
  - Data Flow ?
  - Control Flow ?
  - Event Handling ?
  - Fault-Tolerance ?
- Do the boxes and lines have the same meaning both internal to an architecture and across competing architectures ?**



How to measure which one is better at addressing the issues wrt.:

- Homogenous vs. Heterogeneous systems?
- Expandability ?
- Interoperability ?
- Maintainability ?
- Upgradeability ?
- Responsiveness ?
- ... etc..

**Who do you give the contract to?**  
**It should be obvious then that there needs to be a better (more formal) way of describing and evaluating an architecture !**



# Industry concerns

Industry - We need to be able to:

- Construct the Best Architecture to meet the requirements
- Evaluate Competing Product Architectures
- Share methods, techniques, patterns, idioms for structuring complex systems
- Exploit commonalities in specific domains to provide a reusable framework for product families.
- Educate ourselves on technique and formal methods

So what is meant by “Software Architecture”?

What are the issues that SW Architecture must address?

# Overview of this course

What is software architecture?

How do you use software architectures in practice?

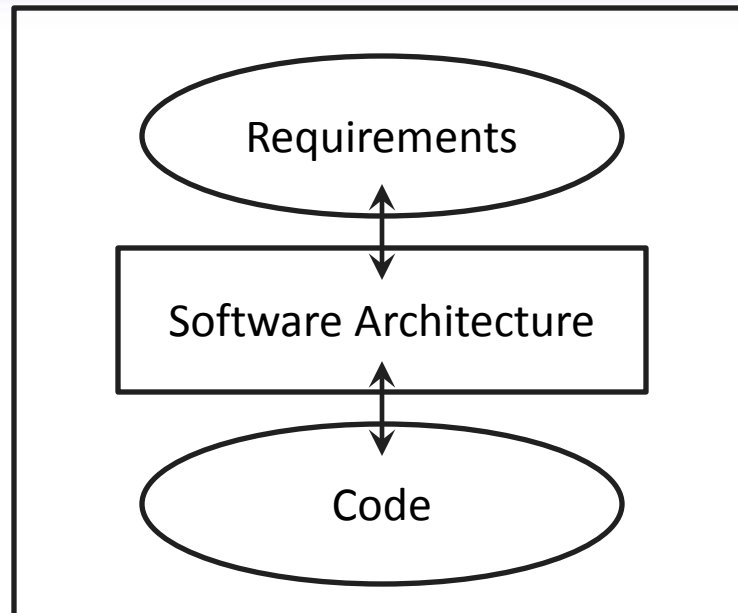
What value does software architecture provide?



# Overview of this course

## The role of SA in software development

- Bridging requirements and implementation





# Overview of this course

## The role of SA in software development

- *Understanding* of large SW system
- *Reuse* of SW components, framework...
- *Construction* of SW
- *Evolution* of SW
- *Analysis*
- *Management* of projects

# Overview of this course

## Software architectures can

- provide flexibility and adaptability in changing markets
- help reduce maintenance costs and amortize development costs
- assist in workforce organization and with project oversight and control
- establish a common corporate vocabulary
- shorten learning time
- ...

# Overview of this course

After attending this course, students will have a better understanding of

- ✓ *what is a software architecture*
- ✓ *why a software architecture is important*
- ✓ *the relationships between system qualities and software architectures*
- ✓ *software architectural patterns and tactics, and their relationship to system qualities*
- ✓ *attribute-driven design*
- ✓ *software architecture documentation*
- ✓ *software architecture evaluation*

# Roadmap of the course

What is Software Architecture?

Designing Software Architecture

- Requirements: quality attributes or qualities
- How to achieve requirements: tactics
- How do tactics lead to architectural styles
- Case studies on architectural styles, observed the achieved qualities
- The ADD method

Evaluating an architecture

Documenting software architecture

ADLs, PLs, COTS, SAr

# Course staff and web page

## Lecturer:

- Ying Shen, Rm. 408R, Jishi Building,
- [yingshen@tongji.edu.cn](mailto:yingshen@tongji.edu.cn)

## Office hour:

- Mon. ~ Fri.: 9:00am – 11:00am, 3:00pm – 5:00pm

## Course web page:

- <http://sse.tongji.edu.cn/yingshen/course/SA/SA14.htm>

# Expectations

Read assigned readings

Attend lectures and participate in discussions

Bring your ideas and concerns to class

Work effectively in a group setting

Learn how to use the tools and understand your project  
*very well*

Hand in your deliverables on time

**NO LATE  
DELIVERABLES!!**

# Course evaluation

Working in group:

## 1. Coursework (20 marks)

- Answer discussion questions at the end of the lecture (two times)
- Comment other groups' answers (two times)

## 2. Project (70 marks)

- Report (40 marks)
- Presentation (30 marks)

## 3. Participation (10 marks)

# Academic integrity and cheating

Cheating, plagiarism and other forms of academic fraud are taken very seriously by the University, the Faculty, and the teaching staff.

## Examples:

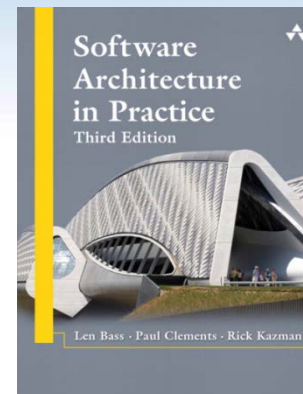
- Submitting the work of another person as your original work.
- Incorporating others work in your work and not attributing it.



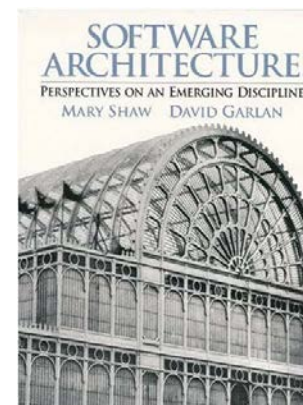
# References

## Main References:

Software Architecture in Practice  
3<sup>rd</sup> Edition



Software Architecture:  
Perspectives on an Emerging Discipline



# Working in groups and choosing a group

Group Size : 4-5

Understand the work habits and goals of your group members:

- Night person
- Start early
- Laid back
- Morning person
- Start at last minute
- Perfectionist

Identify members with good English and communication skills

# Asking questions

Ask me (email, office hours)

Ask in class

Discuss with your classmates or group members