

5.2 – Homework

- Foundations
- ADLs
- Arch. Development
- MiddleWare
- Tradeoff Analysis
- Product Lines
- Background
- Styles
- Towards Design
- Blackboards
- Quality Attributes
- Dynamics

✓ ●	Intro	4
✓ ●	Styles	28
✓ ●	Analysis	223
✓ ●	Selection	286
⇒ ●	References	371
	Formal Methods	
	Homework	
	Model Problem	
	Class References	
	Primary References	
	Other References	
	Bibliography	
	Backup & Deleted	
	Release History	

Originally developed from

- http://sunset.usc.edu/research/software_architecture/index.html and from
- <http://www-2.cs.cmu.edu/afs/cs.cmu.edu/academic/class/17655-s02/www/> and from and from other timely research

Foundations

- Required:
 - None
- Optional
 - None
- Week 1 Questions
 - Describe the current oversights in the software life-cycle with respect to architecture development.
 - Define the elements that constitute an architecture and describe the limitations of current programming languages and development techniques to capture architecture issues.
 - Describe the limitations of current design tools and techniques such as UML to capture architecture issues.
 - Why do we need an Architecture Description Language?

Background

- Required:

- ip Programming in the large vs Programming in the Small[20]

- ip B. W. Boehm and W. L. Scherlis. Megaprogramming. Software Technology Conference 1992, Los Angeles, April 1992.

- ip Perry, Dewayne E. and Alexander L. Wolf, "Foundations for the study of software architecture," Software Engineering Notes, Vol. 17, No. 4, October 1992, pages 40-52.

- ip David Parnas, Paul Clements and David Weiss, The Modular Structure of Complex Systems, IEEE Transactions on Software Engineering, SE-11 (3). March 1985.

- ip M. Shaw and P. Clements. A Field Boxology: Preliminary Classification of Architectural Styles for Software Systems. April 1996.[15]

- Optional:

- How Architecture Wins Technology Wars. Harvard Business Review, 71, 2, March-April 1993, pp. 86-96.

- Formulations and Formalisms in Software Architecture. Computer Science Today: Recent Trends and Developments

- How to Solve It: A New Aspect of Mathematical Methods. Princeton University Press 1973.

- Week 2 Questions

- How does your text book describe architectural styles?

- How have we described what constitutes an architecture in class?

- Describe what you consider to be an architectural style?

- Describe the differences between an architecture style and design patterns?

- How do you answer the question "What kind of architecture have you built?"

ADLs

● Required:

os N. Medvidovic. A Classification and Comparison Framework for Software Architecture Description Languages. Technical Report, UCI-ICS-97-02, University of California, Irvine, January 1997.

ip D. C. Luckham and J. Vera. An Event-Based Architecture Definition Language. IEEE Transactions on Software Engineering, pages 717-734, September 1995.

cs N. Medvidovic et al. A Language and Environment for Architecture-Based Software Development and Evolution. 21st International Conference on Software Engineering, Los Angeles, CA, May 1999.

os E. Di Nitto and D. Rosenblum, "Exploiting ADLs to Specify Architectural Styles Induced by Middleware Infrastructures" Proceedings of 21st. Int'l Conference on Software Engineering (ICSE 99), Los Angeles, CA, May 1999, pp. 13-22

● Optional:

– ACME: An Architecture Description of Component-Based Systems, Foundations of Component-Based Systems

* <http://www-2.cs.cmu.edu/afs/cs/academic/class/17655-s02/www/references/GMW00.pdf>

● Questions:

– What are the desired properties of an Architecture Description Language?

– What are their current limitations?

Styles

- Required:
 - IEEE 1471
 - ip The Modular Structure of Complex Systems
 - ip Using Style to Understand Descriptions of Software Architecture
 - ip Formal Specification and Analysis of Software Architecture Using the Chemical Abstract Machine Model [23]
 - ip P. Krutchen, The Software Architect, ICSE
 - Documenting Software Architectures: Recommendations for Industrial Practice. Carnegie Mellon University, School of Computer Science, CMU-CS-00-169
- Optional:
 - None
- Questions:
 - Describe Architecture Style
 - Why does IEEE-1471 suggest multiple views of an architecture?
 - What are the limitations of UML in capturing an architectural style?

Arch. Development

- Required:

ip D. Parnas, "On the Criteria To Be Used in Decomposing Systems into Modules", Communications of the ACM, 15(12), 1972, 1053-1058.

ip Formal Specification and Analysis of Software Architecture Using the Chemical Abstract Machine Model [23]

- Optional:

- Best Current Practices: Software Architecture Validation. AT&T Report

- Questions:

- What are the limitations of an Object Oriented approach in capturing the desired behavior of an architecture?

Towards Design

References

Selection

Analysis

Styles

Intro

- Required:

- os N. Medvidovic et al. Modeling Software Architectures in the Unified Modeling Language. ACM Transactions on Software Engineering and Methodology, 2002.

- os D. Garlan and A. J. Kompanek Carnegie Melon University. Reconciling the needs of Architectural Description with Object-Modeling Notations. 3rd International Conference on The Unified Modeling Language (UML 2000),

- ip G. Abowd, R. Allen, and D. Garlan, "Using Style to Understand Descriptions of Software Architecture", Proceedings of the ACM SIGSOFT'93 Symposium on Foundations on Software Engineering, Redondo Beach, CA, Software Engineering Notes, December 1993, pp. 9-20

- ip L. Davis, R.F. Gamble, and J. Payton, The impact of component architectures on interoperability, The Journal of Systems and Software Vol 61, pages 31-45. 2002

- Optional

- Specifying Systems [41]

- Questions

- Describe the ability of UML and Object-Oriented notations that you are familiar with to capture the issues addressed during architecture definition.

MiddleWare

- Required:

os N. R. Mehta et al. Towards a Taxonomy of Software Connectors. 22nd International Conference on Software Engineering, Limerick, Ireland, June 2000.

os E. M. Dashofy et al. Using Off-the-Shelf Middleware to Implement Connectors in Distributed Software Architectures. 21st International Conference on Software Engineering, Los Angeles, CA, May 1999.

os M. J. Maybee et al. Multilanguage Interoperability in Distributed Systems: Experience Report. 18th International Conference on Software Engineering, Berlin, Germany, March 1996.

os S. Vinoski. CORBA: Integrating Diverse Applications Within Distributed Heterogeneous Environments. IEEE Communications Magazine, February 1997.

os R. Natarajan and D. S. Rosenblum. Supporting Architectural Concerns in Component Interoperability Standards. IEEE Proceedings Software, December 2000.

- Optional

- Microsoft Corporation. A Blueprint for Building Web Sites Using the Microsoft Windows Platform, Draft, Version .9. January 2000.

- * <http://msdn.microsoft.com/library/en-us/dndna/html/dnablueprint.asp?frame=true>

- Monica Pawlan. Introduction to the J2EE Platform, Mar 23, 2001.

- * <http://developer.java.sun.com/developer/technicalArticles/J2EE/Intro/index.html>

- Component Object Model (COM), DCOM, and Related Capabilities. SEI, maint. Santiago Comella-Dorda, Mar 13, 2001.

- * <http://www.sei.cmu.edu/str/descriptions/com.html>

- Questions

- Compare and contrast the approaches for building large scale, complex distributed applications using the techniques in these papers.

- Identify key similarities and difference between opposing approaches.

BlackBoards

- Required:
 - ip H. Penny Nii, Black board systems at the Architecture Level, Expert Systems with Applications
- Optional:
 - None
- Questions:
 - Describe the components of the blackboard model.
 - Discuss problem solving approaches in the blackboard framework.
 - Why is this architecture a good starting point architecture to begin any requirements analysis.

Tradeoff Analysis

- Required:

- os Quality Attributes. Software Engineering Institute Technical Report, CMU/SEI-95-TR-021

- os The Architecture Tradeoff Analysis Method. Software Engineering Institute Technical Report, CMU/SEI-98-TR-008

- ip M. Klein, R. Kazman, R. Nord, A BASis (or ABASs) for Reasoning about Software Architectures,

- ip L. Davis, R.F. Gamble J. Payton, The impact of component architectures on interoperability, The Journal of Systems and Software, Vol 61, 2002, pgs 31-45

- ip J. Wang X. He, Y. Deng, Introducing software architecture specification and analysis in SAM through an example, Information and Software Technology, Vol 41, 1999, pgs 451-467

- Optional:

- None

- Questions:

- Using the SEI reports, discuss the elements of the tradeoff analysis method and how these elements organize our review of an architecture.

- Discuss how the ATAM process helps to identify architecture tradeoffs and sensitivity points.

- What are possible tradeoffs and sensitivity points?

Quality Attributes

- Required:

- os Achieving Usability Through Software Architecture. Software Engineering Institute Technical Report. CMU/SEI-2001-TR-005.

- os ATAM: Method for Architecture Evaluation. Software Engineering Institute Technical Report, CMU/SEI-2000-TR-004

- os The Architecture Tradeoff Analysis Method. Software Engineering Institute Technical Report, CMU/SEI-98-TR-008

- ip Y. Morisawa, K. Inuone, . Torii, Architectural styles for distributed processing systems and practical selection method

- Optional:

- None

- Questions:

- Discuss the decisions that you need to consider in evaluating competing architectures for compliance with your assigned attribute at different architecture levels.

Product Lines

- Required:

- ip D. Perry, Generic Architecture Descriptions for Product Lines, Bell Labs

- os "Nenad Medvidovic and Richard N. Taylor", Exploiting Architectural Style to Develop a Family of Applications

- cs D. Batory, "Product-Line Architectures"

- Optional:

- None

- Questions:

- Describe the features and considerations of a software architecture for a product line and how those features differ than than of a normal architecture.

Dynamics

- Required:

- ip J. Magee, J. Kramer, Dynamic Structure in Software Architecture:, ACM

- ip D. Luckman, J. Vera, An Event Based Architecture Definition Language, DARPA

- ip R. Taylor and N Medvidovic, A Component and Message Based Architectural Style for GUI Software,

- cs Peyman Oreizy. Issues in the Runtime Modification of Software Architectures

- cs Jeremy S. Bradburya, James R. Cordya, Juergen Dingela, Michel Wermelinger: A Classification of Formal Specifications for Dynamic Software Architectures

- Optional:

- None

- Questions:

- Using the last reference as a basis identify the considerations and tradeoffs for building a dynamic architecture.