Bibliography
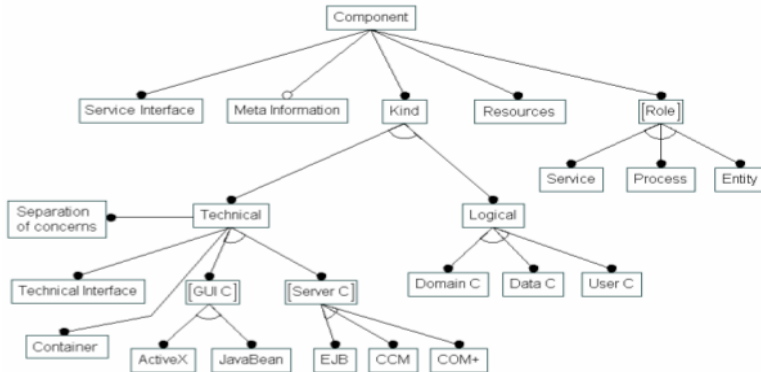
Abdel Hakim Hannousse

22th Meeting

Outline

1 Taxonomies for Component-Based Systems

2 Formalization of Component-Based Systems

3 Aspect-Oriented Programming

## M. Voelter. A taxonomy of components. Journal of Object Technology, 2(4):119-125, July-August 2003

N. Medvidovic and R. N. Taylor. A classification and comparison framework for software architecture description languages. IEEE Transactions on Software Engineering, 26(1):70-93, January 2000.

```
ADL
   Architecture Modeling Features
      Components
         Interface
         Types
         Semantics
         Constraints
         Evolution
         Non-functional properties
      Connectors
         Interface
         Types
         Semantics
         Constraints
         Evolution
         Non-functional properties
      Architectural Configurations
         Understandability
         Compositionality
         Refinement and traceability
         Heterogeneity
         Scalability
         Evolution
         Dynamism
         Constraints
         Non-functional properties
   Tool Support
      Active Specification
      Multiple Views
      Analysis
      Refinement
      Implementation Generation
      Dynamism
```

L. Mariani. A fault taxonomy for component-based software. In International Workshop on Test and Analysis of Component Based Systems (TACOS) satel lite workshop at the European Joint Conferences on Theory and Practice of Software (ETAPS), volume 82 of ENTCS, pages 55-65. Elsevier Science Publishers B. V., 2003.

Main Category	Sub-Categories
Syntactic	• Interface Violation
Semantic	• Misunderstood on the Behavior • Misunderstood on Parameters • Misunderstood on Events • Misunderstood on the Interaction Protocol
Non-Functional	• Performances • Quality of Service
Connectors	• Disagreement on the Protocol • Quality of Service
Infrastructure	• Underlying Services • Underlying System
Topology	• Callback • Re-entrance • Recursion
Other	• Multi-thread • Heterogeneous Languages • Persistence • Inconsistent Events

H. M. Kienle and H. A. Muller. A lightweight taxonomy to characterize component-based systems. In ICCBSS?07: Proceedings of the Sixth International IEEE Conference on Commercial-o?-the-Shelf (COTS)-Based Software Systems, pages 192-204, Washington, DC, USA, 2007. IEEE Computer Society.
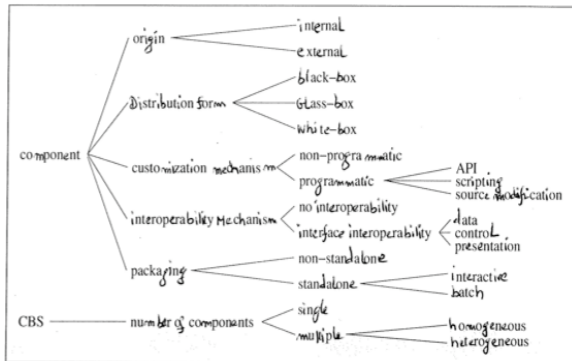
K. Bergner, A. Rausch, M. Sihling, A. Vilbig, and M. Broy. A formal model for componentware. In G. T. Leavens and M. Sitaraman, editors, Foundations of component-based systems, chapter 9, pages 189-210. Cambridge University Press, New York, NY, USA, 2000.

$$SimpleDeviceCSD \in descOf(typeOf(c)) \Rightarrow$$

$$\exists tim, con \in Component, ic, ia \in Interface, cn \in Connection:$$

$$typeOf(tim) = Timer \wedge parent(tim) = c \wedge$$

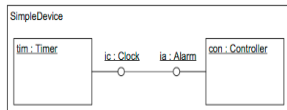$$typeOf(con) = Controller \wedge parent(con) = c \wedge$$

$$typeOf(ic) = Clock \wedge assigned(ic) = tim \wedge$$

$$typeOf(ia) = Alarm \wedge assigned(ia) = con \wedge$$

$$connIfs(cn) = \{ia, ic\} \wedge$$

$$(\forall t \in T : c \in component_t \Rightarrow$$

$$tim, con \in component_t \wedge ic, ia \in interface_t \wedge cn \in connection_t)$$

R. J. Allen. A formal approach to software architecture. PhD thesis, Carnegie Mellon University, Pitts- burgh, PA, USA, May 1997.

Style Simulation
   Interface Type SimInterface = *[Interaction of one simulation]*
   Connector UpdateValues($nsims$: 1..)
      Role Model$_{i\;nsims}$ = SimInterface
      Glue = *[Data travels from one Model to another]*
   Constraints
      $\exists C : Connectors \mid \{C\} = Connectors$
                    $\wedge Type(C) = $ UpdateValues
End Style.

Configuration SimpleSimulation2
   Style Simulation
      Component TerrainModel(map : **Function**)
         Port ProvideMap = SimInterface
         Computation = *[provide terrain data]*
      Component = VehicleModel
         Port Environment = SimInterface
         Computation = *[compute vehicle movement]*
   Instances
      Pittsburgh : TerrainModel(*[map of Pittsburgh]*)
      PAT_Bus : VehicleModel
      C : UpdateValues(2)
   Attachments
      Pittsburgh.ProvideMap, PAT_Bus.Environment as C.Model
End SimpleSimulation2.

P. Avgustinov, E. Bodden, E. Ha jiyev, L. Hendren, O. Lhot´ak, O. de Moor, N. Ongkingco, D. Sereni, G. Sittampalam, J. Tibble, and M. Verbaere. Aspects for trace monitoring. In K. Havelund, M. Nunez, G. Rosu, and B. Wolff, editors, Formal Approaches to Software Testing and Runtime Veri?cation, volume 4262 of Lecture Notes in Computer Science, pages 20-39. Springer Berlin / Heidelberg, November 2006.

C. Allan, P. Avgustinov, A. S. Christensen, L. Hendren, S. Kuzins, O. Lhot´ak, O. de Moor, D. Sereni, G. Sittampalam, and J. Tibble. Adding trace matching with free variables to aspectj. ACM SIGPLAN Notices, 40(10):345-364, 2005.

N. R. Mehta, N. Medvidovic, and S. Phadke. Towards a taxonomy of software connectors. In ICSE 00: Proceedings of the 22nd international conference on Software engineering, pages 178-187, New York, NY, USA, 2000. ACM.(1)

N. R. Mehta, N. Medvidovic, and S. Phadke. Towards a taxonomy of software connectors. In ICSE 00: Proceedings of the 22nd international conference on Software engineering, pages 178-187, New York, NY, USA, 2000. ACM.(2)