Mapping Architectural Decay Instances to Dependency Models

Drexel University: Ran Mo, Yuanfang Cai
University of Southern California: Joshua Garcia, Nenad Medvidovic

Workshop on Managing Technical Debt, 2013

Architectural Decay Instances

- Software architecture drift and erode
- Architectural decay instances
- Architecture debt

Catalog of Architectural Decay Instances[Garcia et al. 2009]

- Scattered Parasitic Functionality
- Extraneous Adjacent Connector
- Brick Concern Overload
- Brick Use Overload
- Ambiguous Interface
- Brick Dependency Cycle
- Unused Interface
- Component Envy
- Connector Envy
- Duplicate Component Functionality
- Unused Brick
- Connector Dimension Overload
- Lego Syndrome
- Sloppy Delegation
- Brick Functionality Overload

Scattered Parasitic Functionality Decay Instance

- Definition
  - Multiple components are responsible for realizing the same high-level concern
  - Some of those components are responsible for orthogonal concerns.

- Formalization

$\exists z_1 \in T \mid P(z1 \mid c1) > th_1 \wedge \exists z_2 \in T \mid P(z_2 \in c_2) > th_1 \wedge z_1 = z_2 \wedge \exists z_3 \in T \mid P(z_3 \mid c_2) > th_2 \wedge z_1 = z_3$, where th_1, th_2 are proportions such that $0 \leq th1 \leq 1$ and $0 \leq th2 \leq 1$.

th_1 and th_2 specify the acceptable degree of scattering per topic

Challenge

- Heterogeneous elements
  - Component
  - Connecter
  - Concerns
  - Interfaces

- Difficult to automatically detect Decay Instances

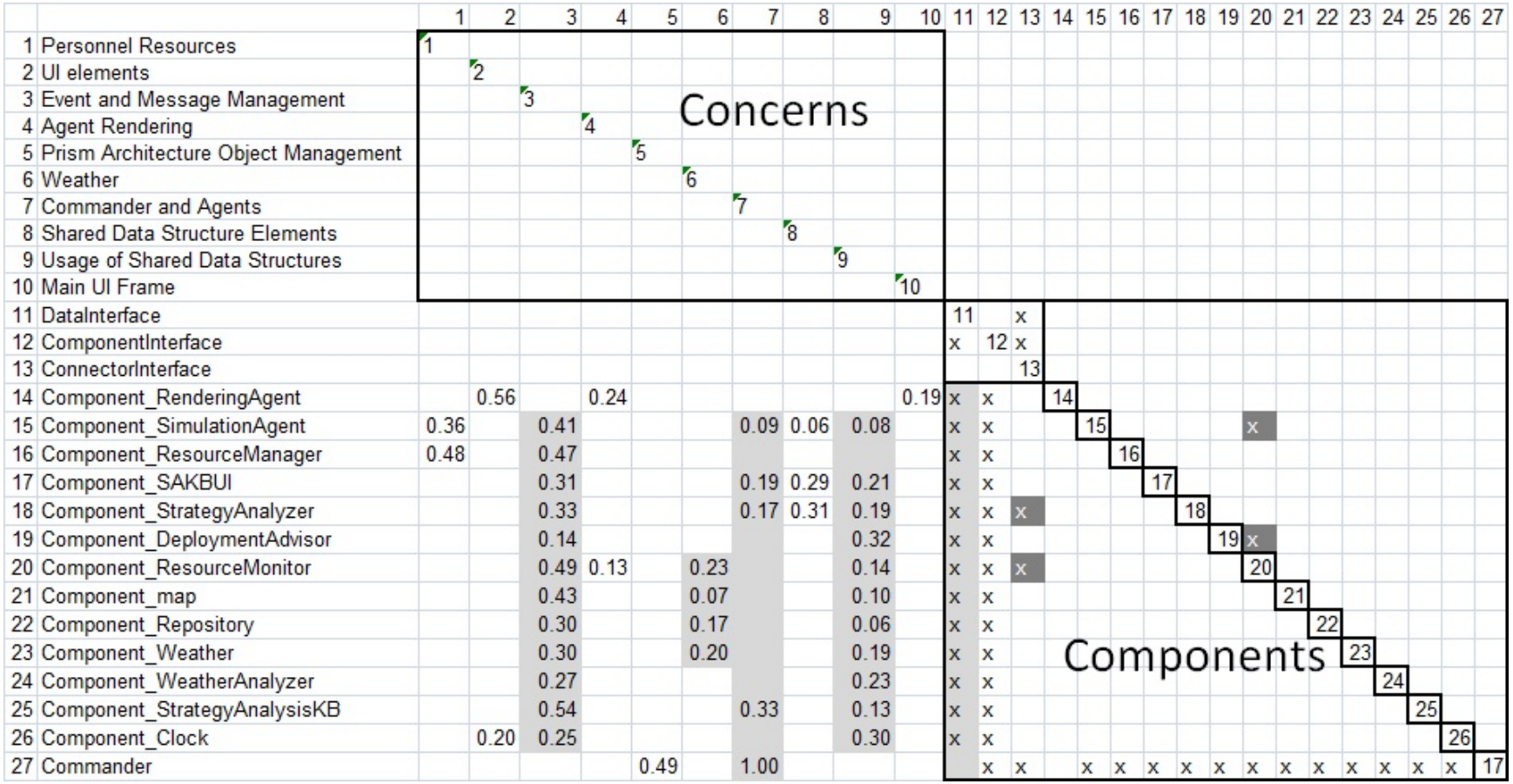
Our Approach

- Uniform the heterogeneous elements and their relations to dependency model
  - Extended Augmented Constraint Network(EACN)
    - Constraint Network
    - Dominance Relation
    - Clustering Set
    - Concern Elements relation
  - Design Structure Matrix(DSM)
    - Derived from EACN
    - Visualizing all the heterogeneous elements and their relations

EACN [Cai and Sullian 2012]

- Variables: Concerns, Components
- Constraints Network: StrategeAnalyzer => ConnectorInteface
- Dominance relation: (StrategeAnalyzer, ConnectorInteface)
- Clustering Set: Concern, Component
- Concern Element relations:<ResourceManager, Event and Message Management, 0.47>

Emergency Response System(ERS)- Design Structure Matrix(DSM)



		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
1	Personnel Resources	1												x														
2	UI elements		2																									
3	Event and Message Management			3																								
4	Agent Rendering				4																							
5	Prism Architecture Object Management					5																						
6	Weather						6																					
7	Commander and Agents							7																				
8	Shared Data Structure Elements								8																			
9	Usage of Shared Data Structures									9																		
10	Main UI Frame										10																	
11	DataInterface											11		x														
12	ComponentInterface											x	12	x														
13	ConnectorInterface													13														
14	Component_RenderingAgent		0.56		0.24						0.19	x	x		14													
15	Component_SimulationAgent	0.36		0.41				0.09	0.06	0.08		x	x			15					x							
16	Component_ResourceManager	0.48		0.47								x	x				16											
17	Component_SAKBUI			0.31				0.19	0.29	0.21		x	x					17										
18	Component_StrategyAnalyzer			0.33				0.17	0.31	0.19		x	x	x					18									
19	Component_DeploymentAdvisor			0.14						0.32		x	x							19	x							
20	Component_ResourceMonitor			0.49	0.13		0.23			0.14		x	x	x							20							
21	Component_map			0.43			0.07			0.10		x	x									21						
22	Component_Repository			0.30			0.17			0.06		x	x										22					
23	Component_Weather			0.30			0.20			0.19		x	x											23				
24	Component_WeatherAnalyzer			0.27						0.23		x	x												24			
25	Component_StrategyAnalysisKB			0.54				0.33		0.13		x	x													25		
26	Component_Clock		0.20	0.25						0.30		x	x														26	
27	Commander					0.49		1.00					x	x		x	x	x	x	x	x	x	x	x	x	x	x	17

Mapping and detecting Scattered Parasitic Functionality Decay Instance

- Definition and Formalization
  - Multiple components are responsible for realizing the same high-level concern; Some of those components are responsible for orthogonal concerns.

$\exists z_1 \in T \mid P(z1 \mid c1) > th_1 \wedge \exists z_2 \in T \mid P(z_2 \in c_2) > th_1 \wedge z_1 = z_2 \wedge \exists z_3 \in T \mid P(z_3 \mid c_2) > th_2 \wedge z_1 = z_3$, where th_1, th_2 are proportions such that $0 \leq th1 \leq 1$ and $0 \leq th2 \leq 1$.

th_1 and th_2 specify the acceptable degree of scattering per topic

- Mapping to the dependency model
  - <ResourceManager, Event and Message Management, 0.47>
  - <StrategyAnalysisKB, Event and Message Management, 0.54>
  - <StrategyAnalysisKB, Commander and Agents concern., 0.33>

Summary

- All the heterogeneous elements and their relations can be modeled in the dependency model
- Mapping all the decay instances to the dependency model
- Detect the decay instances

Question?