

Towards a Practical Model to Facilitate Reasoning about REST Extensions and Reuse

Federico Fernández and Jaime Navón
Pontificia Universidad Católica de Chile

Context

- Interest in understanding and reusing the principles behind the early WWW architecture.
- Dr. Fielding, published a dissertation about designing network-based software architectures through the utilization of an abstraction called “architectural style”.
- In Chapter 5, REST is introduced as the architectural style that was designed to satisfy the requirements of the World Wide Web.

Context (cont.)

- Debates have arisen about the advantages and disadvantages of using REST over other approaches to design network-based software.
 - REST vs SOAP (Muehlen, Nickerson, & Swenson, 2004)
 - REST vs Web Services (Pautasso et al., 2008; Xu, Zhu, Liu, & Staples, 2008)
 - Resource Orientation vs. Service Orientation (Overdick, 2007).

The Problem

- In most papers the definitions introduced in the Dissertation are not used correctly, or not used at all.
- There is a need to describe and communicate new architectural styles based on REST, but not exactly REST.
- We need a practical model of architectural styles in general, and REST in particular.

Our Contribution

- We describe the requirements of a practical model of REST (and an architectural style in general).
- We suggest one model that satisfies most of these requirements. It is built on top of an existing visual planning tool.
- We populate the model with the information provided in Dr. Fielding Dissertation and show its potential use to facilitate REST understanding, extension and reuse.

Related Work

- “Composing architectural styles from architectural primitives”. Mehta, N. R., & Medvidovic, N. (2003).
- “Defining SOA as an architectural style”. Lublinsky, B. (2007).
- “The Resource-Oriented Architecture”. Overdick, H. (2007).
- “RPC and REST: Dilemma, Disruption, and Displacement”. Vinoski, S. (2008).
- “Restful web services vs. ‘big’ web services: making the right architectural decision”. Pautasso, C., Zimmermann, O., & Leymann, F. (2008)
- “Reference Architectural Styles for Service-Oriented Computing”. Dillon, T., Wu, C., & Chang, E. (2008)

Overview of Dr. Fielding's Dissertation

UNIVERSITY OF CALIFORNIA, IRVINE

Architectural Styles and the Design of Network-based Software Architectures

DISSERTATION

submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in Information and Computer Science

by

[Roy Thomas Fielding](#)

2000

Dissertation Committee:

Professor Richard N. Taylor, Chair

Professor Mark S. Ackerman

Professor David S. Rosenblum

Chapter 1

“Software Architecture”

Introduces the concepts about software architecture that are needed to understand the dissertation.

- Software Architecture
- Architectural Styles
- Architectural Elements
- Architectural Properties
- Architectural Design

Chapter 2

“Network-Based Application Architectures”

- Explains that the scope of the dissertation is the study of network-based software applications.
- Describes a hierarchy of architectural properties to compare different architectural styles.
 - User perceived performance, Network performance, Efficiency, Scalability, Simplicity, Evolvability, Extensibility, Personalization, Configurability, Reusability, Visibility, Portability, and Reliability.

Chapter 3

“Network-based Architectural Styles”

- Compares different network-based software architectural styles, considering their impact over the set of architectural properties defined in Chapter 2.
- The Author decided to use a qualitative method to describe these impacts.

Style	Derivation	Net Perform.	UP Perform.	Efficiency	Scalability	Simplicity	Evolvability	Extensibility	Customiz.	Configur.	Reusability	Visibility	Portability	Reliability
PF			±			+	+	+		+	+			
UPF	PF	-	±			++	+	+		±	++	+		
RR			++		+									+
\$	RR		+	+	+	+								
CS					+	+	+							
LS			-		+		+				+		+	
LCS	CS+LS		-		++	+	++				+		+	
CSS	CS	-			++	+	+					+		+
C\$SS	CSS+\$	-	+	+	++	+	+					+		+
LC\$SS	LCS+C\$SS	-	±	+	+++	++	++				+	+	+	+
RS	CS			+	-	+	+					-		
RDA	CS			+	-	-						+		-
VM						±		+				-	+	
REV	CS+VM			+	-	±		+	+			-	+	-
COD	CS+VM		+	+	+	±		+		+		-		
LCODC\$SS	LC\$SS+COD	-	++	++	+4+	+±±	++	+		+	+	±	+	+
MA	REV+COD		+	++		±		++	+	+		-	+	
EBI				+	--	±	+	+		+	+	-		-
C2	EBI+LCS		-	+		+	++	+		+	++	±	+	±
DO	CS+CS	-		+			+	+		+	+	-		-
BDO	DO+LCS	-	-				++	+		+	++	-	+	

Table 1: Matrix of architectural styles vs. software qualities

Chapter 4

“Designing the Web Architecture: Problems and Insights”

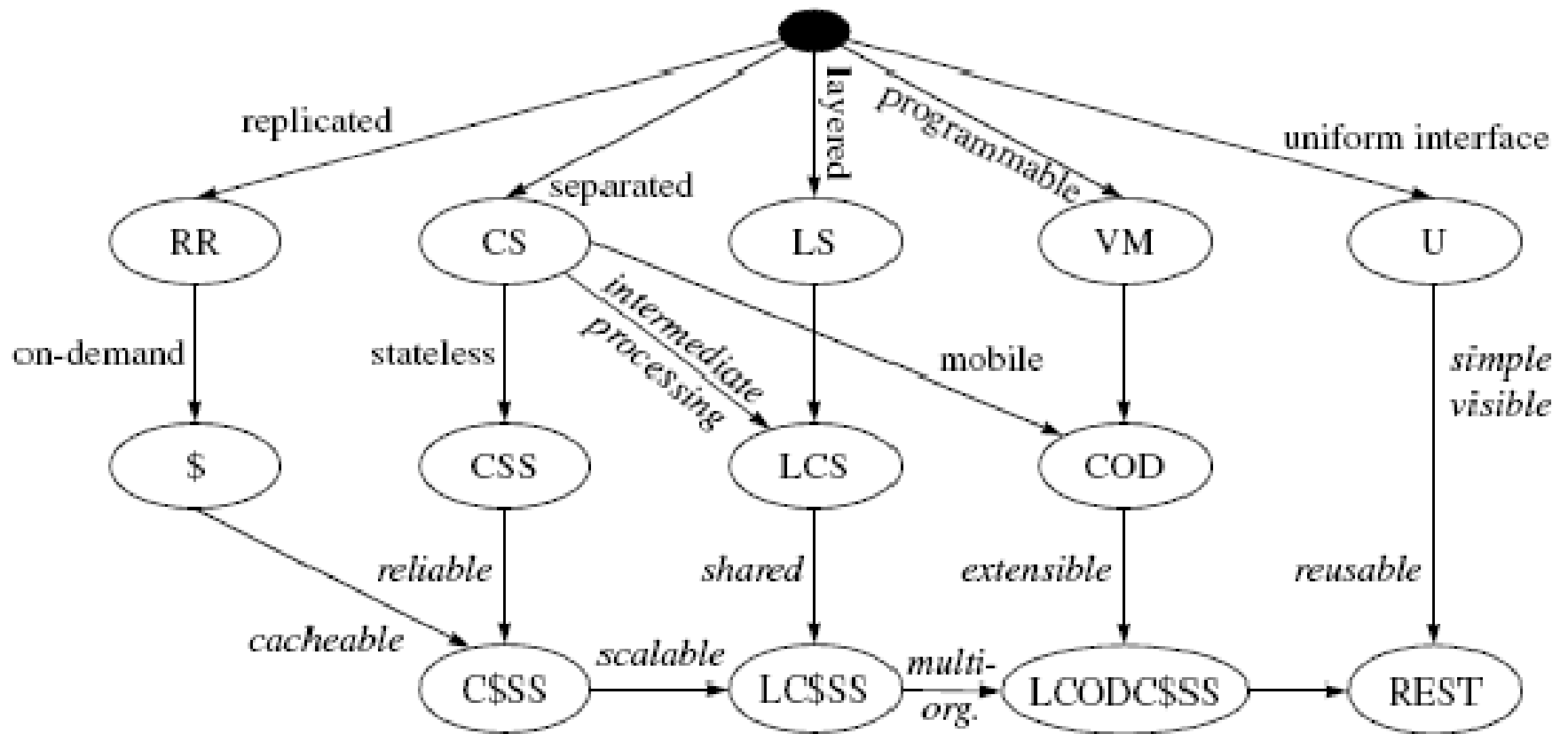
Explains the requirements of the WWW architecture:

- Low entry-barriers for the different types of users: authors, editors and readers
- Extensibility.
- Scalability to the level of the Internet
 - Anarchic scalability
 - Independent implementations

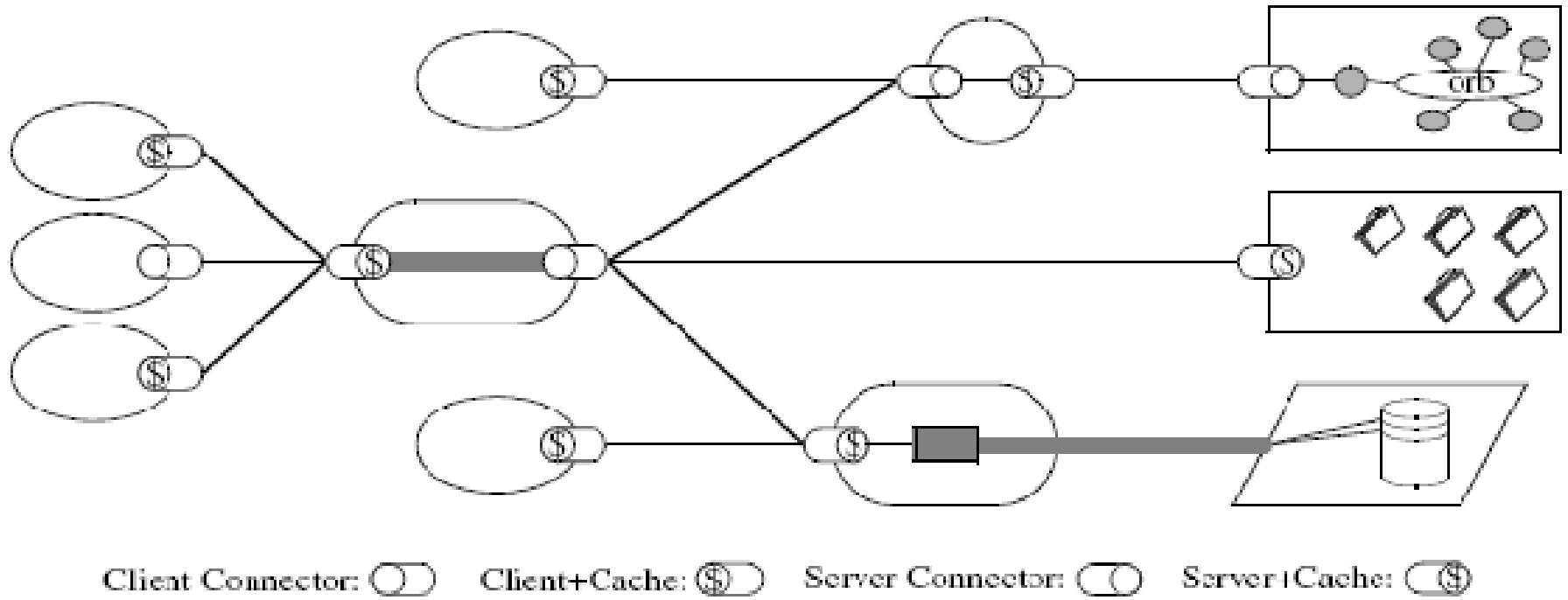
Chapter 5

“Representational State Transfer(REST)”

- Describes the architectural style used to guide the development of the standard protocols that constitute the WWW architecture.
- The rationale of REST is presented by the utilization of a styles derivation tree.
- The structure of REST is shown using Process View, Data View and Connector View.

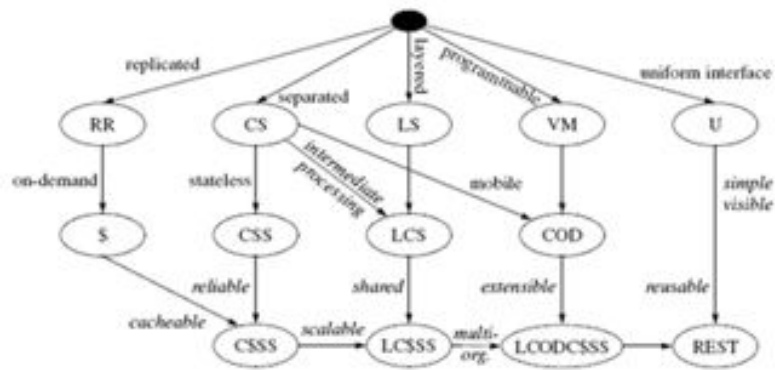


REST Derivation Tree



REST Process View

Towards a Practical Model of REST



REST derivation tree

Style	Definition	RR	CS	LS	VM	U	S	CSS	LCS	COD	CSSS	LCSSS	LCODCSSS	REST
RR		+												
CS			+											
LS				+										
VM					+									
U						+								
S							+							
CSS								+						
LCS									+					
COD										+				
CSSS											+			
LCSSS												+		
LCODCSSS													+	
REST														+

Matrix of styles vs architectural properties

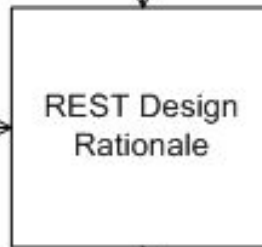


Text

model of

model of

model of

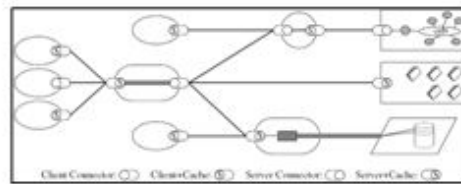
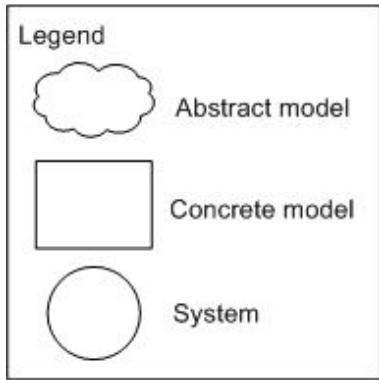


We would like a simple and practical model of this.

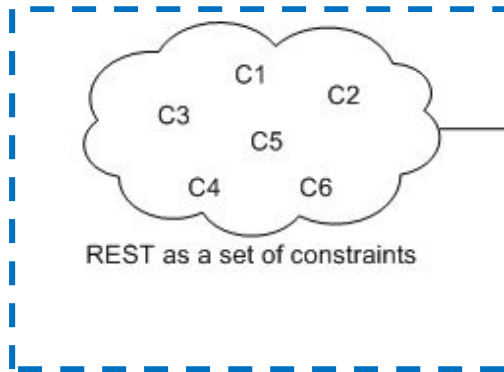
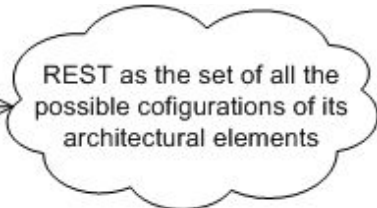
Models of the REST Design Rationale

Requirements of a Practical Model of the REST Design Rationale

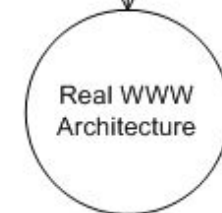
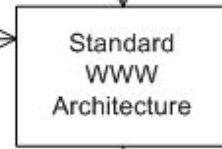
- The user should be able to...
 - Visualize and understand how each one of the architectural decisions of REST impacts the set of goals that guided its design (R1).
 - Visualize and understand the changes caused in the induced properties if new architectural decisions are added to REST, or if existing decisions are replaced for others (R2).
 - Visualize the set of alternative architectural decisions for each decision in REST (R3).
 - Easily modify the REST model to visualize and understand how each one of the architectural decisions of REST would impact a different set of goals (R4).
- The model should be a loyal representation of the Dissertation in order to be accepted by the stakeholders (R5).
- Prerequisite: we need a practical model of the REST structure.



REST process view



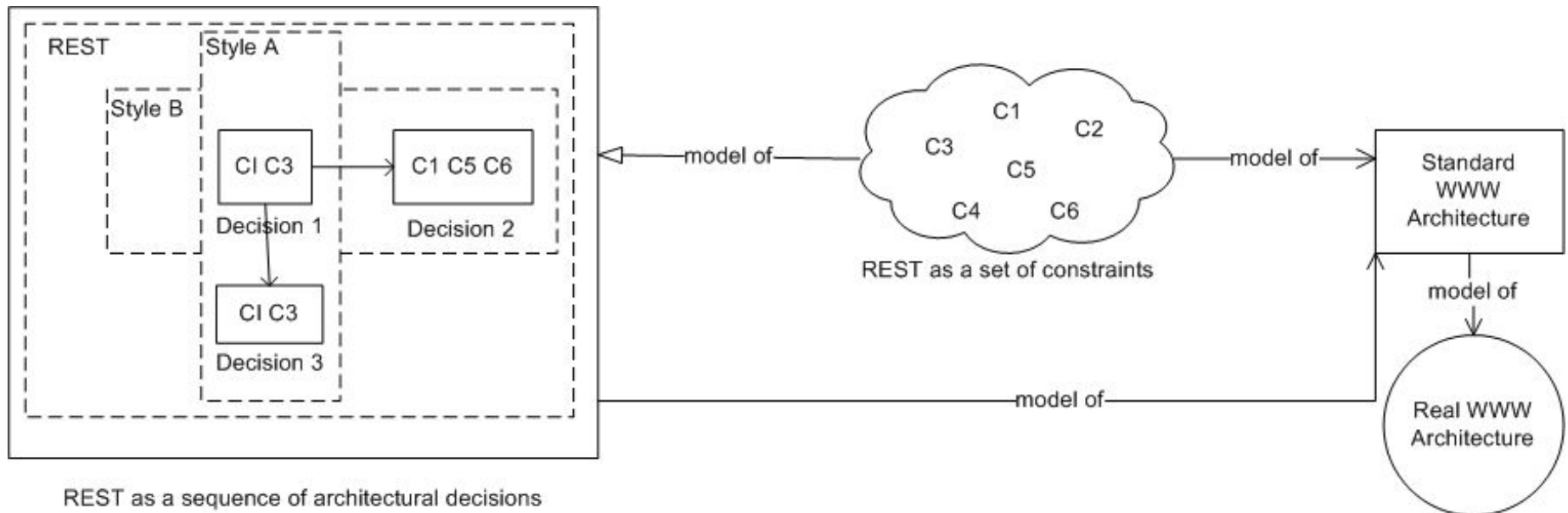
We would like a model of this.



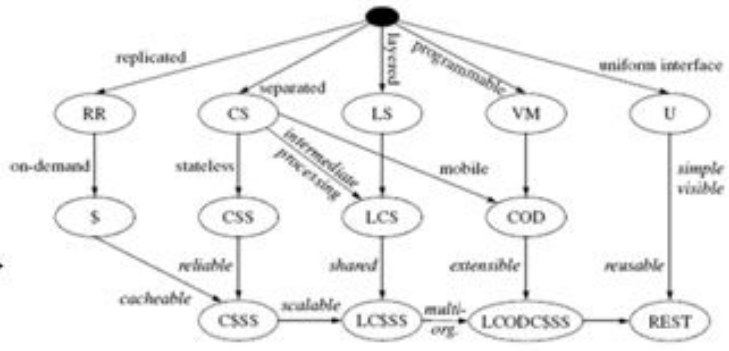
Models of REST Structure

Definition

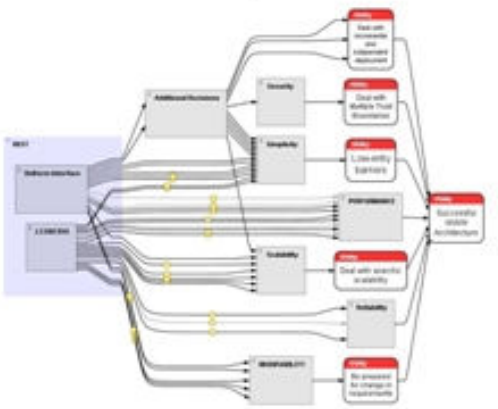
- “An architectural decision is a named set of constraints that can be added to an architectural style. The result of adding an architectural decision to an architectural style is another architectural style.”
 - **Corollaries:**
 - Any given architectural decision can be made only over certain architectural styles.
 - Every component of an architectural style is explicit.



Our proposed model of REST structure



REST derivation tree



REST Influence Diagram

Style	Replicated	Separated	Stateless	Mobile	Uniform Interface	Cacheable	Shared	Extensible	Reusable
RR	Y								
CS		Y							
LS			Y						
VM				Y					
U					Y				
S	Y					Y			
CSS		Y	Y			Y			
LCS				Y			Y		
COD				Y				Y	
CSSS		Y	Y			Y			
LCSSS				Y			Y		
LCODCSSS				Y				Y	
REST	Y	Y	Y	Y	Y	Y	Y	Y	Y

Matrix of styles vs architectural properties

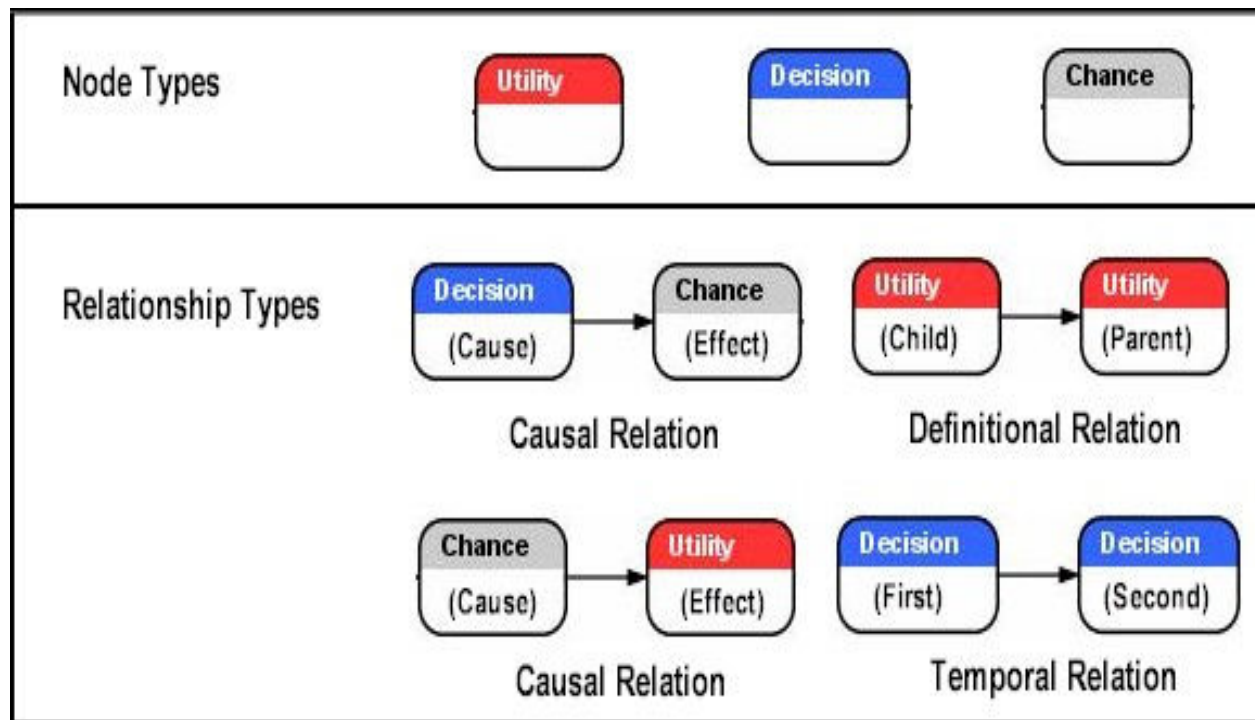


Text

Practical model of the REST design rationale

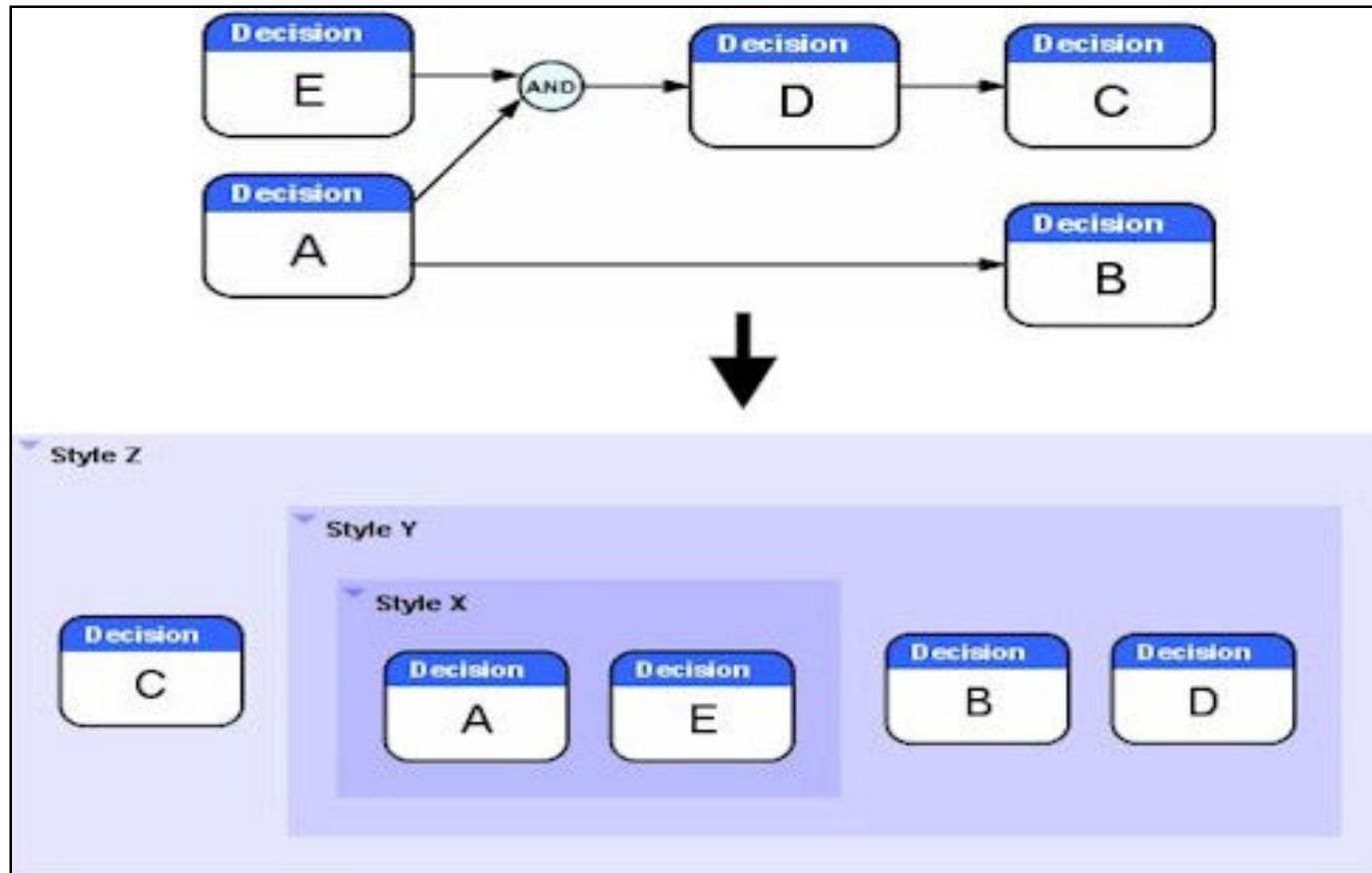
Extended Influence Diagrams

- Probabilistic graphical models.
- Similar to Bayes Networks, but in addition to causality they include: decisions, goals and goal hierarchies.

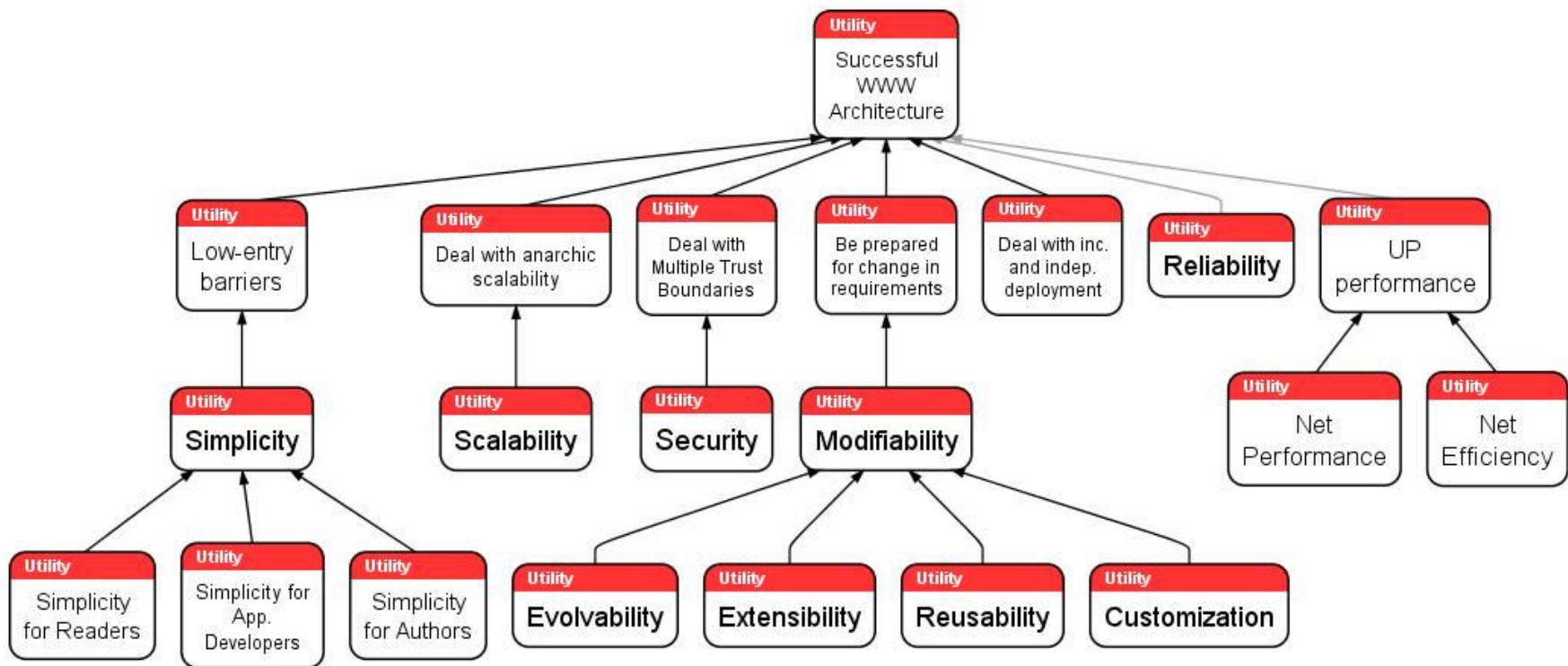


Syntax of the REST influence diagram

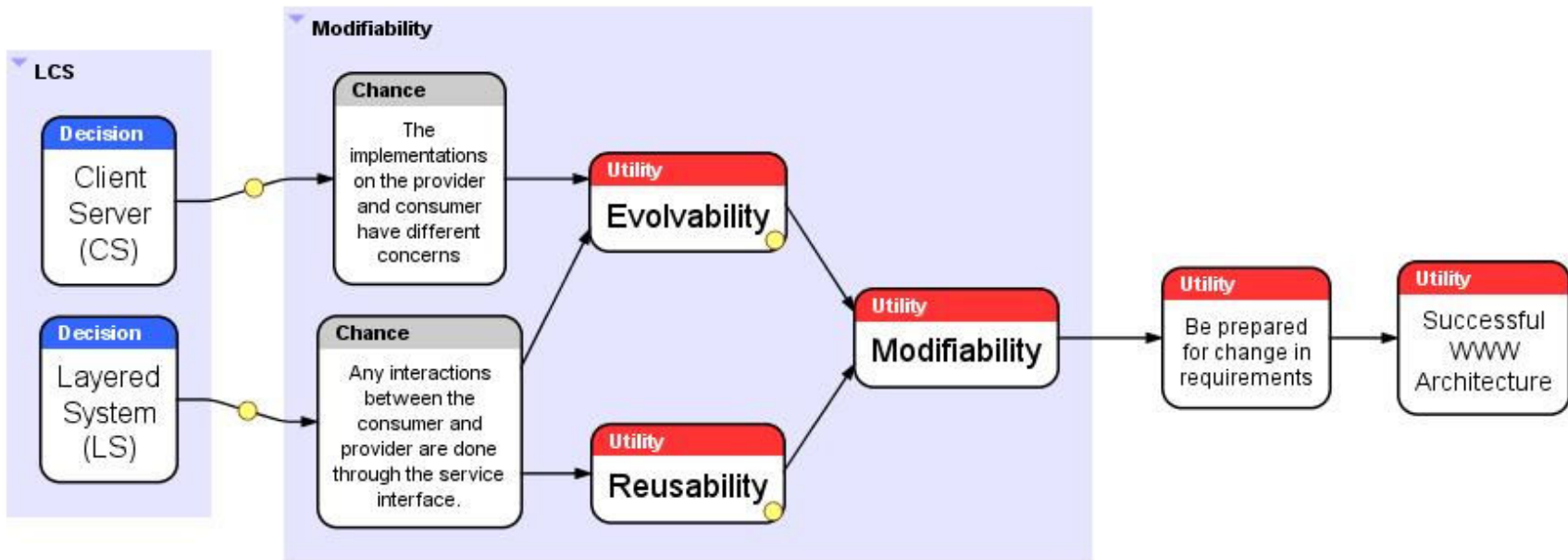
Adapting the notation to the tool



Example of how we draw sequences of architectural decisions



Goals of the standard WWW architecture



Part of the REST Influence Diagram

DEMO

Advantages

- The user can quickly evaluate the impact of using a subset of the architectural defined by REST.
- It was populated with the information available in the doctoral dissertation of Dr. Fielding.
- It facilitates reasoning about the construction of architectural styles similar to REST, adding or subtracting decisions and watching their impact on the set of desired properties.
- It facilitates reasoning about the applicability of REST to solve new problems. The user has to start modifying the goal hierarch and reinterpreting the path between each decision and each goal.

Limitations

- The texts describing the chances are not concise and is hard to define them in such a way that guarantees its reuse in future decision-goal paths.
- An inexperienced reader will not understand the diagram unless good comments are provided.

Future Work

- Create a naming system for the architectural styles that can be described with this model.
- Study the applicability of REST to other domains (e.g. SOA).

QUESTIONS?