A Design Quality Learning Unit in Relational Data Modeling Based on Thrivings Systems Theory

Les Waguespack, Ph.D.
Professor & Chairperson of Computer Information Systems

Bentley University
Waltham Massachusetts

ISECON - New Orleans, LA - 2012-11-1
“How is it that one system is more effective, appealing, satisfying and/or more beautiful than another to its stakeholder community?”

“How is it that one system is more effective, appealing, satisfying and/or more beautiful than another to its stakeholder community?”


How is it that one Relational Data model is better than another?
“How is it that one system is more effective, appealing, satisfying and/or more beautiful than another to its stakeholder community?”

How is it that one Relational Data model is better than another?

What determines design quality in an Relational Data model?
We Are Here!

Thriving Systems Theory


Object-oriented ontology


Relational data ontology

Agile vocabulary of SCRUM
Pursuing a Universal Foundation of System Design Quality

from the beauty in nature to the quality in systems

properties of order in nature

Pursuing a Universal Foundation of System Design Quality

from the beauty in nature to the quality in systems

properties of order in nature
Pursuing a Universal Foundation of System Design Quality

from the beauty in nature

properties of order in nature

Stepwise Refinement
Modularization
Cohesion
Encapsulation
Composition of Function
Scale
Correctness
User Friendliness
Extensibility
Patterns
Reliability
Transparency
Programmability
Identity
Elegance

to the quality in systems
Pursuing a Universal Foundation of System Design Quality

from the beauty in nature

system design actions and choice properties

to the quality in systems

elaborate
modularize
factor
encapsulate
assemble
focus
align
accommodate
extend
pattern
normalize
expose
generalize
identify
coordinate

Stepwise Refinement
Modularization
Cohesion
Encapsulation
Composition of Function
Scale
Correctness
User Friendliness
Extensibility
Patterns
Reliability
Transparency
Programmability
Identity
Elegance
Pursuing a Universal Foundation of System Design Quality

elaborate
modularize
factor
encapsulate
assemble
focus
align
accommodate
extend
pattern
normalize
expose
generalize
identify
coordinate

Stepwise Refinement
Modularization
Cohesion
Encapsulation
Composition of Function
Scale
Correctness
User Friendliness
Extensibility
Patterns
Reliability
Transparency
Programmability
Identity
Elegance

from the beauty in nature
to the quality in systems

properties of order in nature

system design actions and choice properties
Pursuing a Universal Foundation of System Design Quality

from the beauty in nature

elaborate
modularize
factor
encapsulate
assemble
focus
align
accommodate
extend
pattern
normalize
expose
generalize
identify
coordinate

system design actions and choice properties

Object-oriented ontology

properties of order in nature

to the quality in systems

to the
object-oriented models that thrive

Cohesion
Modularization
Encapsulation
Composition of Function
Scale
Correctness
User Friendliness
Extensibility
Patterns
Reliability
Transparency
Programmability
Identity
Elegance
Pursuing a Universal Foundation of System Design Quality

from the beauty in nature

properties of order in nature

elaborate modularize factor

encapsulate assemble focus

align accommodate extend pattern

normalize expose generalize identify

coordinate

system design actions and choice properties
Pursuing a Universal Foundation of System Design Quality

elaborate, modularize, factor, encapsulate, assemble, focus, align, accommodate, extend, pattern, normalize, expose, generalize, identify, coordinate

Stepwise Refinement, Modularization, Cohesion, Encapsulation, Composition of Function, Scale, Correctness, User Friendliness, Extensibility, Patterns, Reliability, Transparency, Programmability, Identity, Elegance

from the beauty in nature to the quality in systems

properties of order in nature

object-oriented models that thrive

relational data models that thrive

agile project management models that thrive

system design actions and choice properties

relational data ontology

Object-oriented ontology

SCRUM ontology
Pursuing a Universal Foundation of System Design Quality

from the beauty in nature

elaborate
modularize
factor
encapsulate
assemble
focus
align
accommodate
extend
pattern
normalize
expose
generalize
identify
coordinate

system design actions and choice properties

to the quality in systems

object-oriented models that thrive

relational data models that thrive

agile project management models that thrive

business models that thrive ??

properties of order in nature

Cohesion
Modularization
Encapsulation
Composition of Function
Scale
Correctness
User Friendliness
Extensibility
Patterns
Reliability
Transparency
Programmability
Identity
Elegance

Stepwise Refinement

Object-oriented ontology

Relational data ontology

SCRUM ontology

Business process modeling vocabulary or ontology?
defining design quality
defining design quality

quality |ˈkwælətē| noun

1 the standard of something as measured against other things of a similar kind; the degree of excellence of something: an improvement in product quality | people today enjoy a better quality of life.

• general excellence of standard or level: a masterpiece for connoisseurs of quality | [as modifier]: a wide choice of quality beers.

2 a distinctive attribute or characteristic possessed by someone or something: he shows strong leadership qualities | the plant’s aphrodisiac qualities.
defining design quality

quality |ˈkwälətē| noun

1 the standard of something as measured against other things of a similar kind; the degree of excellence of something: an improvement in product quality | people today enjoy a better quality of life.

• general excellence of standard or level: a masterpiece for connoisseurs of quality | [ as modifier ] : a wide choice of quality beers.

2 a distinctive attribute or characteristic possessed by someone or something: he shows strong leadership qualities | the plant’s aphrodisiac qualities.

design |deˈzīn| verb [ with obj. ]

decide upon the look and functioning of (a building, garment, or other object), typically by making a detailed drawing of it: a number of architectural students were designing a factory | [ as adj. with submodifier ] (designed) : specially designed buildings.

• do or plan (something) with a specific purpose or intention in mind: [ with obj. and infinitive ] : the tax changes were designed to stimulate economic growth.
the individual’s experience of design quality

implementation

threshold

expectation

mindset
the individual’s experience of design quality

implementation

threshold

expectation

mindset

the “mental picture” the observer brings to the experience within which they will “understand” the experience
the individual’s experience of design quality

implementation | threshold | expectation | mindset

the subset of the observer’s mindset (conscious or unconscious) that is specifically relevant to the event

the “mental picture” the observer brings to the experience within which they will “understand” the experience
the individual’s experience of design quality

implementation

the point of encounter between the expectation and the system’s features

threshold

expectation

the subset of the observer’s mindset (conscious or unconscious) that is specifically relevant to the event

mindset

the “mental picture” the observer brings to the experience within which they will “understand” the experience
the individual’s experience of design quality

implementation
the assembled artifact’s realization that creates the opportunity for observation

threshold
the point of encounter between the expectation and the system’s features

expectation
the subset of the observer’s mindset (conscious or unconscious) that is specifically relevant to the event

mindset
the “mental picture” the observer brings to the experience within which they will “understand” the experience
the individual’s experience of design quality

implementation
the assembled artifact’s realization that creates the opportunity for observation

threshold
the point of encounter between the expectation and the system’s features

expectation
the subset of the observer’s mindset (conscious or unconscious) that is specifically relevant to the event

mindset
the “mental picture” the observer brings to the experience within which they will “understand” the experience
the community’s experience of design quality
the community’s experience of design quality

implementation
the assembled artifact’s realization that creates the opportunity for observation

threshold
the point of encounter between the expectation and the system’s features

expectation
the subset of the observer’s mindset (conscious or unconscious) that is specifically relevant to the event

mindset
the “mental picture” the observer brings to the experience within which they will “understand” the experience
Quality Design: The application of quality Principles in the process of creating artifacts
“How is it that one system is more effective, appealing, satisfying and/or more beautiful than another to its stakeholder community?”


How is it that one relational model is better than another?

What determines design quality in an relational model?
The Design Process
The Design Process

- Stakeholder intensions
- Requirement elements
- Model elements
- Design elements
The Design Process

Stakeholder Intensions
Requirement Elements
Model Elements
Design Elements
The Design Process

“rules of thumb”
Traditional patterns

Stakeholder intensions
requirement elements
model elements
design elements

Representational paradigm or Ontology
The Design Process

“Rules of Thumb”
Traditional patterns

Stakeholder intentions
Requirement elements
Model elements
Design elements

Representational paradigm or Ontology

Resulting Design Elements
The Design Process

“Rules of thumb”
Traditional patterns

Stakeholder intensions
requirement elements
model elements
design elements

Resulting Design Elements

Representational paradigm or Ontology

- functional dependency
- relation
- domain
- data
- value
- tuple
- membership IN
- property
- instance
- relationship
- entity integrity
- membership OF
- property
- atomicity
- property
- referential integrity
- property
- attribute
- association

©LJW, 2012 All Rights Reserved
The Design Process

Choice Property-Driven Design Principles

Stakeholder intensions
requirement elements
model elements
design elements

Representational paradigm or Ontology

Resulting Design Elements
These choice properties propose a coherent, descriptive language including:

- a vocabulary for describing and comparing aspects of system components and structures, and
- design actions to guide design choices leading to desirable system characteristics.
<table>
<thead>
<tr>
<th>Choice Property</th>
<th>Modeling Action</th>
<th>Action Rendition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Stepwise Refinement</td>
<td>elaborate</td>
<td>develop or present (a theory, policy, or system) in detail</td>
</tr>
<tr>
<td>2 Cohesion</td>
<td>factor</td>
<td>express as a product of factors</td>
</tr>
<tr>
<td>3 Encapsulation</td>
<td>encapsulate</td>
<td>enclose the essential features of something succinctly by a protective coating or membrane</td>
</tr>
<tr>
<td>4 Extensibility</td>
<td>extend</td>
<td>render something capable of expansion in scope, effect, or meaning</td>
</tr>
<tr>
<td>5 Modularization</td>
<td>modularize</td>
<td>employing or involving a module or modules as the basis of design or construction</td>
</tr>
<tr>
<td>6 Correctness</td>
<td>align</td>
<td>put (things) into correct or appropriate relative positions</td>
</tr>
<tr>
<td>7 Transparency</td>
<td>expose</td>
<td>reveal the presence of (a quality or feeling)</td>
</tr>
<tr>
<td>8 Composition of Function</td>
<td>assemble</td>
<td>fit together the separate component parts of (a machine or other object)</td>
</tr>
<tr>
<td>9 Identity</td>
<td>identify</td>
<td>establish or indicate who or what (someone or something) is</td>
</tr>
<tr>
<td>10 Scale</td>
<td>focus</td>
<td>(of a person or their eyes) adapt to the prevailing level of light [abstraction] and become able to see clearly</td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>accommodate</td>
<td>fit in with the wishes or needs of</td>
</tr>
<tr>
<td>12 Patterns</td>
<td>pattern</td>
<td>give a regular or intelligible form to</td>
</tr>
<tr>
<td>13 Programmability</td>
<td>generalize</td>
<td>make or become more widely or generally applicable</td>
</tr>
<tr>
<td>14 Reliability</td>
<td>normalize</td>
<td>make something more normal, which typically means conforming to some regularity or rule</td>
</tr>
<tr>
<td>15 Elegance</td>
<td>coordinate</td>
<td>bring the different elements of (a complex activity or organization) into a relationship that will ensure efficiency or harmony</td>
</tr>
<tr>
<td>Choice Property</td>
<td>Modeling Action</td>
<td>Action Rendition Through Relational Modeling</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>1 Stepwise Refinement</td>
<td>elaborate</td>
<td>defining and elaborating intention first through independent correctness then through collections of relations</td>
</tr>
<tr>
<td>2 Cohesion</td>
<td>factor</td>
<td>defining tuples: cogent, unambiguous: reality patterned thru attributes</td>
</tr>
<tr>
<td>3 Encapsulation</td>
<td>encapsulate</td>
<td>defining tuples: individually &quot;truthful:&quot; reality independently encapsulated thru attributes</td>
</tr>
<tr>
<td>4 Extensibility</td>
<td>extend</td>
<td>complex information realized through associations</td>
</tr>
<tr>
<td>5 Modularization</td>
<td>modularize</td>
<td>knowledge subdivided and compartmentalized; assuring extension based upon intention through normalization</td>
</tr>
<tr>
<td>6 Correctness</td>
<td>align</td>
<td>normalization, entity and referential integrity supporting intention fidelity</td>
</tr>
<tr>
<td>7 Transparency</td>
<td>expose</td>
<td>data attribute values revealing transparent integrity</td>
</tr>
<tr>
<td>8 Composition of Function</td>
<td>assemble</td>
<td>relational operators guarded by integrity constraints expressing the distilled content</td>
</tr>
<tr>
<td>9 Identity</td>
<td>identify</td>
<td>entity integrity disambiguating identity and capacitating association</td>
</tr>
<tr>
<td>10 Scale</td>
<td>focus</td>
<td>distilling the extension of the reality through the intention of the model</td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>accommodate</td>
<td>revealing correctness transparently thru the elegant simplicity of individual relations</td>
</tr>
<tr>
<td>12 Patterns</td>
<td>pattern</td>
<td>predictable form realized thru mathematically consistent relations</td>
</tr>
<tr>
<td>13 Programmability</td>
<td>generalize</td>
<td>elegant recursion thru the closure of relational operations</td>
</tr>
<tr>
<td>14 Reliability</td>
<td>normalize</td>
<td>easily verifiable synchronization between intention and extension</td>
</tr>
<tr>
<td>15 Elegance</td>
<td>coordinate</td>
<td>eliminating structural redundancy enabling almost endless composition by encapsulating relationships based upon identity</td>
</tr>
</tbody>
</table>
design quality across the system models

implementation  threshold  expectation  mindset
design quality across the system models

implementation  threshold  expectation  mindset

Choice Property-Driven Design Principles
design quality across the system models

implementation
  hardware & software architecture

threshold
  interfaces

expectation
  requirements engineering & specification

mindset
  object oriented “systems think”

Choice Property-Driven Design Principles
DILBERT® by Scott Adams

WALLY, HAVE YOU MADE ANY PROGRESS CODING YOUR MODULE?

PROGRESS IS DIFFICULT TO MEASURE IN THE SOFTWARE REALM.

YOU COULD MEASURE THE LINES OF CODE I PRODUCE, BUT THAT WOULD REWARD INEFFICIENCY.

THE ART OF THIS JOB IS BINDING THE RARE MOMENTS OF INSPIRATION TO KNOWLEDGE AND MACHINES.

IN FACT, JUST A MINUTE AGO I COULD FEEL THE INSPIRATION WELLING UP INSIDE ME.

BUT THEN YOU INTERRUPTED ME WITH YOUR NAIVE QUESTION AND THE MOMENT WAS LOST.

MAYBE YOU SHOULD GO BACK TO YOUR OFFICE AND REFLECT ON THE DAMAGE YOU'VE DONE HERE TODAY.

THERE GOES THE ONE PERSON WHO HAS LESS OF A REAL JOB THAN I DO.
Dogbert is a creativity consultant.

We don't need any of your "intuition" mumbo jumbo. We need quantitative data!

The only way to make decisions is to pull numbers out of the air, call them "assumptions," and calculate the net present value.

Of course, you have to use the right discount rate, otherwise it's meaningless.

Go away.
How is it that one system is more effective, appealing, satisfying and/or more beautiful than another to its stakeholder community? This question drove Christopher Alexander’s fifty-year quest to explain great physical architecture and give birth to pattern languages for building that underpin much of modern systems engineering.

How is it that so many individual stakeholders consistently recognize the same quality, the same beauty in a system? This question led George Lakoff to research the role of conceptual metaphor in human understanding.

What is essential to stakeholders’ satisfaction with systems? Fred Brooks addressed this question in No Silver Bullet: Essence and Accidents of Software Engineering.

This monograph fuses these diverse streams of thought in proposing Thriving Systems Theory by translating Alexander’s properties of physical design quality into the abstract domain of information systems and modeling. Metaphor-Driven Modeling incorporates the theory while examining its impact throughout the system life cycle: modeling, design and deployment. The result is holistic and innovative, a perspective on system quality invaluable to students, practitioners and researchers of software and systems engineering.

Les Waguespack is a computer science Ph.D., professor and chairperson of computer information systems at Bentley University, USA. Dr. Waguespack’s experience as programmer, software engineer, software architect, database architect, project manager and systems consultant underpins 35 years of teaching and research, the last 20+ years teaching object-oriented modeling and systems engineering to undergraduates, graduate students and practicing professionals.