Thriving Systems Theory: Pursuing a Universal Foundation of System Design Quality

Les Waguespack, Ph.D.
Professor & Chairperson of Computer Information Systems
Bentley University
Waltham Massachusetts

Presented 2012-3-19 to: School of Computer and Information Sciences,
University of South Alabama
Overview of This Talk
Overview of This Talk

Can science explain design quality – great design?
Overview of This Talk

- Can science explain design quality – great design?
- Great designs seem always to proceed from great designers.
Overview of This Talk

- Can science explain design quality – great design?
  - Great designs seem always to proceed from great designers.
  - Is design quality “objective,” “subjective,” or BOTH!??
Overview of This Talk

- Can science explain design quality – great design?
  - Great designs seem always to proceed from great designers.
  - Is design quality “objective,” “subjective,” or BOTH!?
  - Might this inform/improve design of information systems?
Overview of This Talk

Can science explain design quality – great design?
- Great designs seem always to proceed from great designers.
- Is design quality “objective,” “subjective,” or BOTH!?
- Might this inform/improve design of information systems?

Enter Christopher Alexander -
Overview of This Talk

- Can science explain design quality – great design?
  - Great designs seem always to proceed from great designers.
  - Is design quality “objective,” “subjective,” or BOTH!?
  - Might this inform/improve design of information systems?

- Enter Christopher Alexander -
  - leading theorist in the architecture of visible structures
Overview of This Talk

- Can science explain design quality – great design?
  - Great designs seem always to proceed from great designers.
  - Is design quality “objective,” “subjective,” or BOTH!?
  - Might this inform/improve design of information systems?
- Enter Christopher Alexander -
  - leading theorist in the architecture of visible structures
  - patriarch of pattern languages and design patterns
Overview of This Talk

Can science explain design quality – great design?

- Great designs seem always to proceed from great designers.
- Is design quality “objective,” “subjective,” or BOTH!?
- Might this inform/improve design of information systems?

Enter Christopher Alexander -

- leading theorist in the architecture of visible structures
- patriarch of pattern languages and design patterns

Inspiring *Thriving Systems Theory*
Overview of This Talk

- Can science explain design quality – great design?
  - Great designs seem always to proceed from great designers.
  - Is design quality “objective,” “subjective,” or BOTH!?
  - Might this inform/improve design of information systems?

- Enter Christopher Alexander -
  - leading theorist in the architecture of visible structures
  - patriarch of pattern languages and design patterns

- Inspiring *Thriving Systems Theory*
  - speculating a fundamental and comprehensive taxonomy of system design quality
What do we “see” in great design?
What do we “see” in great design?
What do we “see” in great design?
What do we “see” in great design?
What do we “see” in great design?
What do we “see” in great design?
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 his 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 this theory while examining its impact throughout the systems 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 chairman 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.

Thriving Systems Theory and Metaphor-Driven Modeling

Leslie J. Waguespack

Monograph in Two Parts
Monograph in Two Parts

Thriving Systems Theory

- a vocabulary of elemental properties describing system elements appropriate to a particular domain
- a taxonomy of system quality resulting from the interplay of those elemental properties
Monograph in Two Parts

- Thriving Systems Theory
  - a vocabulary of elemental properties describing system elements appropriate to a particular domain
  - a taxonomy of system quality resulting from the interplay of those elemental properties

- Metaphor-Driven Modeling
  - metaphorology and its role in both the conception and communication of knowledge
  - metaphor as the implement of design in analysis, representation and realization of information systems
Monograph in Two Parts

- **Thriving Systems Theory**
  - a vocabulary of elemental properties describing system elements appropriate to a particular domain
  - a taxonomy of system quality resulting from the interplay of those elemental properties

- **Metaphor-Driven Modeling**
  - metaphorology and its role in both the conception and communication of knowledge
  - metaphor as the implement of design in analysis, representation and realization of information systems
What is Design Quality?
What is Design Quality?

- A degree of excellence (Oxford English Dictionary)
What is Design Quality?

- A degree of excellence (Oxford English Dictionary)
- Freedom from deficiencies or defects (Juran 2009)
What is Design Quality?

- A degree of excellence (Oxford English Dictionary)
- Freedom from deficiencies or defects (Juran 2009)
- Conformity to requirements (Crosby 1979)
What is Design Quality?

- A degree of excellence (Oxford English Dictionary)
- Freedom from deficiencies or defects (Juran 2009)
- Conformity to requirements (Crosby 1979)
- Fitness for use (Juran 2009)
What is Design Quality?

- A degree of excellence (Oxford English Dictionary)
- Freedom from deficiencies or defects (Juran 2009)
- Conformity to requirements (Crosby 1979)
- Fitness for use (Juran 2009)
- Fitness for purpose (Sales and supply goods act:UK 1994)
What is Design Quality?

- A degree of excellence (Oxford English Dictionary)
- Freedom from deficiencies or defects (Juran 2009)
- Conformity to requirements (Crosby 1979)
- Fitness for use (Juran 2009)
- Fitness for purpose (Sales and supply goods act: UK 1994)
- The degree to which the inherent characteristics fulfill requirements (ISO 9000:2005)
What is Design Quality?

- A degree of excellence (Oxford English Dictionary)
- Freedom from deficiencies or defects (Juran 2009)
- Conformity to requirements (Crosby 1979)
- Fitness for use (Juran 2009)
- Fitness for purpose (Sales and supply goods act:UK 1994)
- The degree to which the inherent characteristics fulfill requirements (ISO 9000:2005)
- Sustained satisfaction (Deming 1993)
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: the subset of the observer’s mindset (conscious or unconscious) that is specifically relevant to the event
- expectation: the “mental picture” the observer brings to the experience within which they will “understand” the experience
- mindset:
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

implementation  threshold  expectation  mindset
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
Thriving Systems Theory
Thriving Systems Theory

Properties of order in nature
Thriving Systems Theory

- Levels of Scale
- Positive Space
- Strong Centers
- Boundaries
- Deep Interlock and Ambiguity
- Gradients
- Good Shape
- Roughness
- Alternating Repetition
- Echoes
- Simplicity and Inner Calm
- Local Symmetries
- The Void
- Contrast
- Not Separateness

Properties of Order in Nature
Architectural Domains
Discipline Defined Design Characteristics
Vocabulary of Design Properties
Thriving Systems Theory

- Levels of Scale
- Positive Space
- Strong Centers
- Boundaries
- Deep Interlock and Ambiguity
- Gradients
- Good Shape
- Roughness
- Alternating Repetition
- Echoes
- Simplicity and Inner Calm
- Local Symmetries
- The Void
- Contrast
- Not Separateness

Properties of Order in Nature
Architectural Domains
Discipline Defined Design Characteristics
Vocabulary of Design Properties
A Search For Fundamental Principles Of System Design Quality
A Search For Fundamental Principles Of System Design Quality

Fred Brooks’ Essence and Accidents of Information Systems Development
A Search For Fundamental Principles Of System Design Quality

Fred Brooks’ Essence and Accidents of Information Systems Development

George Lakoff’s Theories of Conceptual Metaphor in Cognition
A Search For Fundamental Principles Of System Design Quality

- Fred Brooks’ Essence and Accidents of Information Systems Development
- Christopher Alexander’s Theory of Life in Architecture
- George Lakoff’s Theories of Conceptual Metaphor in Cognition
A Search For Fundamental Principles Of System Design Quality

Fred Brooks’ Essence and Accidents of Information Systems Development

Christopher Alexander’s Theory of Life in Architecture

George Lakoff’s Theories of conceptual Metaphor in Cognition
Threads of Theory in Design
Threads of Theory in Design

**Notes on the Synthesis of Form**
Christopher Alexander
1964

**The Oregon Experiment**
1975

**A Pattern Language**
1977

**A Timeless Way of Building**
1979

**The Phenomenon of Life**
2002

**The Process of Creating of Life**
2004

**A Vision of a Living World**
2005

**The Luminous Ground**
2004

...
Brooks, Frederick P.,
“Whereas the difference between poor conceptual designs and good ones may lie in the soundness of design-method, the difference between good designs and great ones surely does not. Great designs come from great designers. Software construction is a creative process. Sound methodology can empower and liberate the creative mind; it cannot inflame or inspire the drudge.”
“Whereas the difference between poor conceptual designs and good ones may lie in the soundness of design-method, the difference between good designs and great ones surely does not. Great designs come from great designers. Software construction is a creative process. Sound methodology can empower and liberate the creative mind; it cannot inflame or inspire the drudge.”

“Study after study shows that the very best designers produce structures that are faster, smaller, simpler, cleaner, and produced with less effort. [...] The differences between the great and the average approach an order of magnitude.”
“Whereas the difference between poor conceptual designs and good ones may lie in the soundness of design-method, the difference between good designs and great ones surely does not. Great designs come from great designers. Software construction is a creative process. Sound methodology can empower and liberate the creative mind; it cannot inflame or inspire the drudge.”

“Study after study shows that the very best designers produce structures that are faster, smaller, simpler, cleaner, and produced with less effort. [...] The differences between the great and the average approach an order of magnitude.”

“I believe the hard part of building software to be the specification, design, and testing of this conceptual construct, not the labor of representing it and testing the fidelity of the representation.”
George Lakoff’s Theories of Metaphor in Cognition
George Lakoff’s Theories of Metaphor in Cognition

- Human conception of the “subjective” is grounded in the experience of the “objective.”
Human conception of the “subjective” is grounded in the experience of the “objective.”

Human sensorimotor experience with environment shapes the conceptual structure of our cognition (“understanding”) and is evidenced through our language.

George Lakoff’s Theories of Metaphor in Cognition
George Lakoff’s Theories of Metaphor in Cognition

- Human conception of the “subjective” is grounded in the experience of the “objective.”

- Human sensorimotor experience with environment shapes the conceptual structure of our cognition (“understanding”) and is evidenced through our language.

- Metaphor is not only a linguistic implement of communication, but a fundamental mechanism of cognition.
Human conception of the “subjective” is grounded in the experience of the “objective.”

Human sensorimotor experience with environment shapes the conceptual structure of our cognition (“understanding”) and is evidenced through our language.

Metaphor is not only a linguistic implement of communication, but a fundamental mechanism of cognition.

The physiology of the human brain is “hard-wired” to store, retrieve, and correlate memory aided by categorization, assimilation, and extension through metaphors recorded, mapped, and processed via neural connections.
Human conception of the “subjective” is grounded in the experience of the “objective.”

Human sensorimotor experience with environment shapes the conceptual structure of our cognition (“understanding”) and is evidenced through our language.

Metaphor is not only a linguistic implement of communication, but a fundamental mechanism of cognition.

The physiology of the human brain is “hard-wired” to store, retrieve, and correlate memory aided by categorization, assimilation, and extension through metaphors recorded, mapped, and processed via neural connections.

Integrated in a spatial-motor sense of our surroundings (reaching for, moving toward or away from, being over, under, inside or outside of, surrounded by) the sensorimotor system of our experience is a continuous source of physical metaphors that frame our consciousness and our subjectivity.

Lakoff, G. and M. Johnson, Metaphors We Live By, University of Chicago Press, Chicago, IL, 1980.
Human conception of the “subjective” is grounded in the experience of the “objective.”

Human sensorimotor experience with environment shapes the conceptual structure of our cognition (“understanding”) and is evidenced through our language.

Metaphor is not only a linguistic implement of communication, but a fundamental mechanism of cognition.

The physiology of the human brain is “hard-wired” to store, retrieve, and correlate memory aided by categorization, assimilation, and extension through metaphors recorded, mapped, and processed via neural connections.

Integrated in a spatial-motor sense of our surroundings (reaching for, moving toward or away from, being over, under, inside or outside of, surrounded by) the sensorimotor system of our experience is a continuous source of physical metaphors that frame our consciousness and our subjectivity.

Lakoff, G. and M. Johnson, Metaphors We Live By, University of Chicago Press, Chicago, IL, 1980.

Is design quality “objective,” “subjective,” or BOTH!?
Christopher Alexander – Great Architect

OOPSLA 1996 Keynote Speech
Introduction by Jim Coplien
“Once in a great while, a great idea makes it across the boundary of one discipline to take root in another. The adoption of Christopher Alexander’s patterns by the software community is one such event.”
Christopher Alexander – *Great Architect*

OOPSLA 1996 Keynote Speech
Introduction by Jim Coplien

“Once in a great while, a great idea makes it across the boundary of one discipline to take root in another. The adoption of Christopher Alexander’s patterns by the software community is one such event.”

“Alexander both commands respect and inspires controversy in his own discipline; he is the author of several books with long-running publication records, the first recipient of the AIA Gold Medal for Research, a member of the Swedish Royal Academy since 1980, a member of the American Academy of Arts and Sciences, recipient of dozens of awards and honors including:
- the Best Building in Japan award in 1985,
- the American Association of Collegiate Schools of Architecture Distinguished Award.”
Christopher Alexander’s Theory of Life in Architecture
Christopher Alexander’s Theory of Life in Architecture

Humans perceive order “... as elements that systematically conform as constituent components of a whole achieving an arrangement of ‘WHOLENESS’.”
Humans perceive order “... as elements that systematically conform as constituent components of a whole achieving an arrangement of ‘WHOLENESS’.”

Presented any two systems as visual images, in excess of 80% of observers consistently agree upon which exhibits the greater degree of “Life.”
80+% Consistently Agree
on which system of visual images exhibits more “Life”
80+% Consistently Agree
on which system of visual images exhibits more "Life"
Christopher Alexander’s Theory of Life in Architecture

Humans perceive order “as elements that systematically conform as constituent components of a whole achieving an arrangement of ‘WHOLENESS’.”

Presented any two systems as visual images, in excess of 80% of observers consistently agree upon which exhibits the greater degree of “Life.”
Christopher Alexander’s Theory of Life in Architecture

Humans perceive order “as elements that systematically conform as constituent components of a whole achieving an arrangement of ‘WHOLENESS’.”

Presented any two systems as visual images, in excess of 80% of observers consistently agree upon which exhibits the greater degree of “Life.”

Order is dynamic rather than static. It derives not from multiplicity but from the transparency of unfolding structure.
Christopher Alexander’s Theory of Life in Architecture

Humans perceive order “as elements that systematically conform as constituent components of a whole achieving an arrangement of ‘WHOLENESS’.”

Presented any two systems as visual images, in excess of 80% of observers consistently agree upon which exhibits the greater degree of “Life.”

Order is dynamic rather than static. It derives not from multiplicity but from the transparency of unfolding structure.

“The concept extends to any space where objects & relationships are observed.”

The Nature of Order
The Nature of Order

“The arrangement of things is based upon their arrival at relative positions influenced by forces that guide their movement or evolution. Continuously guided by these forces order emerges and is preserved over time, space, or change as elements systematically conform as constituent components of a whole achieving an arrangement of ‘WHOLENESS’.”

(Alexander)
The Nature of Order

“The arrangement of things is based upon their arrival at relative positions influenced by forces that guide their movement or evolution. Continuously guided by these forces order emerges and is preserved over time, space, or change as elements systematically conform as constituent components of a whole achieving an arrangement of ‘WHOLENESS’.”

(Alexander)

Wholeness is stable, disorder is not!
Centers: the elements of architectural design that form the fabric of Wholeness
Centers: the elements of architectural design that form the fabric of Wholeness

- Center - “a distinct set of points in space, which, because of its organization, because of its internal coherence, and because of its relation to its context, exhibits centeredness, forms a local zone of relative centeredness with respect to other parts of space.” (Alexander)
Centers: the elements of architectural design that form the fabric of Wholeness

- Center - “a distinct set of points in space, which, because of its organization, because of its internal coherence, and because of its relation to its context, exhibits centeredness, forms a local zone of relative centeredness with respect to other parts of space.” (Alexander)

- A center expresses itself as a focus of attention. (Waguespack)
Centers: the elements of architectural design that form the fabric of Wholeness

- Center - “a distinct set of points in space, which, because of its organization, because of its internal coherence, and because of its relation to its context, exhibits centeredness, forms a local zone of relative centeredness with respect to other parts of space.” (Alexander)

- A center expresses itself as a focus of attention. (Waguespack)

- Centers are distinct but may combine to form nested centers.
Centers: the elements of architectural design that form the fabric of Wholeness

Center - “a distinct set of points in space, which, because of its organization, because of its internal coherence, and because of its relation to its context, exhibits centeredness, forms a local zone of relative centeredness with respect to other parts of space.” (Alexander)

A center expresses itself as a focus of attention. (Waguespack)

Centers are distinct but may combine to form nested centers.

“In any given region of space, some sub-regions have higher intensity as centers; others have less...or none. The overall configurations of their nested centers, together with their relative intensities, comprise a single structure – ‘the’ Wholeness of that region.” (Alexander)
Center Properties
Center Properties

- Centers and Wholeness are Everywhere
Center Properties

- Centers and Wholeness are Everywhere

- Center properties distinguish the influence a center exerts on an observer’s experience of an artifact that results in a degree of satisfaction.
Center Properties

- Centers and Wholeness are Everywhere

- Center properties distinguish the influence a center exerts on an observer’s experience of an artifact that results in a degree of satisfaction.

- A center’s influence is experienced both individually in the center itself and in the center’s orientation and interrelationship with the matrix of all centers in an artifact as a whole.
Center Properties

- Centers and Wholeness are Everywhere

- Center properties distinguish the influence a center exerts on an observer’s experience of an artifact that results in a degree of satisfaction.

- A center’s influence is experienced both individually in the center itself and in the center’s orientation and interrelationship with the matrix of all centers in an artifact as a whole.

- Satisfaction is the degree of “resonance” between a center’s cumulative center properties and the observer.
Center Properties

- Centers and Wholeness are Everywhere

- Center properties distinguish the influence a center exerts on an observer’s experience of an artifact that results in a degree of satisfaction.

- A center’s influence is experienced both individually in the center itself and in the center’s orientation and interrelationship with the matrix of all centers in an artifact as a whole.

- Satisfaction is the degree of “resonance” between a center’s cumulative center properties and the observer.

(Alexander)
Alexander’s 15 Center Properties Expressing Architectural Quality
Alexander’s 15 Center Properties Expressing Architectural Quality

- Strong Centers
- Not Separateness
- Boundaries
- Positive Space
- Deep Interlock and Ambiguity
- Good Shape
- Alternating Repetition
- Simplicity and Inner Calm
- The Void
- Levels of Scale
- Gradients
- Roughness
- Echoes
- Local Symmetries
- Contrast
- The Void
Levels of Scale: A strong center is made stronger partly by smaller strong centers contained in it, and partly by its larger strong centers which contain it. A balanced range of sizes is pleasing and beautiful.

Positive Space: A center should draw strength from the centers immediately adjacent. The background should reinforce rather than detract from the center.

Strong Centers: A strong center requires a field-like effect created by other centers. Good design offers areas of focus or weight.

Boundaries: The field-like effect is strengthened by the creation of a ring-like center. Outlines focus attention on the center.

Deep Interlock and Ambiguity: The intensity of a center can be increased when it is attached to nearby strong centers through a third set of strong centers that ambiguously belong to both. Looping, interconnected elements promote unity and grace.

Gradients: A center is strengthened by a graded series of different sized centers which then point to a new center. The proportional use of space and pattern creates harmony.

Local Symmetries: The intensity of a center is increased by the extent to which other smaller centers are themselves arranged in locally symmetrical groups. Organic, small-scale symmetry works better than precise, overall symmetry.

The Void: The intensity of every center depends on the existence of a still place – an empty center. Empty spaces offer calm and contrast.

Good Shape: The strength of a center depends on its actual shape. Its boundaries and the space around it must be made up of strong centers. Simple forms create an intense, powerful center.

Roughness: The way a center draws its strength from irregularities in sizes, shapes and arrangements. Texture and imperfections convey uniqueness and life.

Alternating Repetition: Centers are strengthened when they repeat, by the insertion of other centers between them. Repeating various elements creates a sense of order and harmony.

Echoes: The strength of a given center depends on similarities of angle and orientation. Similarities should repeat themselves throughout a design.

Simplicity and Inner Calm: The strength of a center depends on its simplicity. Use only essentials and avoid extraneous elements.

Contrast: A center is strengthened by the sharpness of distinction between itself and the surrounding centers. Unity is achieved with visible opposites.

Not Separateness: The strength of a center depends on the extent to which that center is merged smoothly with surrounding centers. Designs should be connected and complementary, not egocentric and isolated.

Alexander’s 15 Centers Properties Expressing Architectural Quality
Levels of Scale: A strong center is made stronger partly by smaller strong centers contained in it, and partly by its larger strong centers which contain it. A balanced range of sizes is pleasing and beautiful.

Local Symmetries: The intensity of a center depends on the extent to which other smaller centers are themselves arranged in locally symmetrical groups. Organic, small-scale symmetry works better than precise, overall symmetry.

The Void: The intensity of every center depends on the existence of a still place – an empty center. Empty spaces offer calm and contrast.

Positive Space: A center should draw strength from the centers immediately adjacent. The background should reinforce rather than detract from the center.

Strong Centers: A strong center requires a field-like effect created by other centers. Good design offers areas of focus or weight.

Boundaries: The field-like effect is strengthened by the creation of a ring-like center. Outlines focus attention on the center.

Deep Interlock and Ambiguity: The intensity of a center can be increased when it is attached to nearby strong centers through a third set of strong centers that ambiguously belong to both. Looping, interconnected elements promote unity and grace.

Gradients: A center is strengthened by a graded series of different sized centers which then point to a new center. The proportional use of space and pattern creates harmony.

Contrast: A center is strengthened by the sharpness of distinction between itself and the surrounding centers. Unity is achieved with visible opposites.

Not Separateness: The strength of a center depends on the extent to which that center is merged smoothly with surrounding centers. Designs should be connected and complementary, not egocentric and isolated.

Good Shape: The strength of a center depends on its actual shape. Its boundaries and the space around it must be made up of strong centers. Simple forms create an intense, powerful center.

Roughness: The way a center draws its strength from irregularities in sizes, shapes and arrangements. Texture and imperfections convey uniqueness and life.

Alternating Repetition: Centers are strengthened when they repeat, by the insertion of other centers between them. Repeating various elements creates a sense of order and harmony.

Echoes: The strength of a given center depends on similarities of angle and orientation. Similarities should repeat themselves throughout a design.

Simplicity and Inner Calm: The strength of a center depends on its simplicity. Use only essentials and avoid extraneous elements.
The Void: The intensity of every center depends on the existence of a still place – an empty center. Empty spaces offer calm and contrast.

Not Separateness: The strength of a center depends on the extent to which that center is merged smoothly with surrounding centers. Designs should be connected and complementary, not egocentric and isolated.

Contrast: A center is strengthened by the sharpness of distinction between itself and the surrounding centers. Unity is achieved with visible opposites.

Simplicity and Inner Calm: The strength of a center depends on its actual shape. Use only essentials and avoid extraneous elements.

Levels of Scale: A strong center is made stronger partly by smaller strong centers contained in it, and partly by its larger strong centers which contain it. A balanced range of sizes is pleasing and beautiful.

Positive Space: A center should draw strength from the centers immediately adjacent. The background should reinforce rather than detract from the center.

Strong Centers: A strong center requires a field-like effect created by other centers. Good design offers areas of focus or weight.

Local Symmetries: The intensity of a center is increased by the extent to which other smaller centers are themselves arranged in locally symmetrical groups. Organic, small-scale symmetry works better than precise, overall symmetry.

Boundaries: The field-like effect is strengthened by the creation of a ring-like center. Outlines focus attention on the center.

Deep Interlock and Ambiguity: The intensity of a center can be increased when it is attached to nearby strong centers through a third set of strong centers that ambiguously belong to both. Looping, interconnected elements promote unity and grace.

Gradients: A center is strengthened by a graded series of different sized centers which then point to a new center. The proportional use of space and pattern creates harmony.

Alternating Repetition: Centers are strengthened when they repeat, by the insertion of other centers between them. Repeating various elements creates a sense of order and harmony.

Echoes: The strength of a given center depends on similarities of angle and orientation. Similarities should repeat themselves throughout a design.

Roughness: The way a center draws its strength from irregularities in sizes, shapes and arrangements. Texture and imperfections convey uniqueness and life.

Good Shape: The strength of a center depends on its actual shape. Its boundaries and the space around it must be made up of strong centers. Simple forms create an intense, powerful center.

Alexander’s 15 Centers Properties Expressing Architectural Quality
Contrast: A center is strengthened by the sharpness of distinction between itself and the surrounding centers. Unity is achieved with visible opposites.
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levels of Scale</td>
<td>A strong center is made stronger partly by smaller strong centers contained in it, and partly by its larger strong centers which contain it. A balanced range of sizes is pleasing and beautiful.</td>
</tr>
<tr>
<td>Positive Space</td>
<td>A center should draw strength from the centers immediately adjacent. The background should reinforce rather than detract from the center.</td>
</tr>
<tr>
<td>Strong Centers</td>
<td>A strong center requires a field-like effect created by other centers. Good design offers areas of focus or weight.</td>
</tr>
<tr>
<td>Local Symmetries</td>
<td>The intensity of a center is increased by the extent to which other smaller centers are themselves arranged in locally symmetrical groups. Organic, small-scale symmetry works better than precise, overall symmetry.</td>
</tr>
<tr>
<td>The Void</td>
<td>The intensity of every center depends on the existence of a still place – an empty center. Empty spaces offer calm and contrast.</td>
</tr>
<tr>
<td>Good Shape</td>
<td>The strength of a center depends on its actual shape. Its boundaries and the space around it must be made up of strong centers. Simple forms create an intense, powerful center.</td>
</tr>
<tr>
<td>Roughness</td>
<td>The way a center draws its strength from irregularities in sizes, shapes and arrangements. Texture and imperfections convey uniqueness and life.</td>
</tr>
<tr>
<td>Alternating Repetition</td>
<td>Centers are strengthened when they repeat, by the insertion of other centers between them. Repeating various elements creates a sense of order and harmony.</td>
</tr>
<tr>
<td>Echoes</td>
<td>The strength of a given center depends on similarities of angle and orientation. Similarities should repeat themselves throughout a design.</td>
</tr>
<tr>
<td>Deep Interlock and Ambiguity</td>
<td>The intensity of a center can be increased when it is attached to nearby strong centers through a third set of strong centers that ambiguously belong to both. Looping, interconnected elements promote unity and grace.</td>
</tr>
<tr>
<td>Gradients</td>
<td>A center is strengthened by a graded series of different sized centers which then point to a new center. The proportional use of space and pattern creates harmony.</td>
</tr>
<tr>
<td>Contrast</td>
<td>A center is strengthened by the sharpness of distinction between itself and the surrounding centers. Unity is achieved with visible opposites.</td>
</tr>
<tr>
<td>Not Separateness</td>
<td>The strength of a center depends on the extent to which that center is merged smoothly with surrounding centers. Designs should be connected and complementary, not egocentric and isolated.</td>
</tr>
<tr>
<td>Simplicity and Inner Calm</td>
<td>The strength of a center depends on its simplicity. Use only essentials and avoid extraneous elements.</td>
</tr>
</tbody>
</table>

**Alexander’s 15 Centers Properties Expressing Architectural Quality**
Alexander’s 15 Centers Properties Expressing Architectural Quality

**Good Shape**: The strength of a center depends on its actual shape. Its boundaries and the space around it must be made up of strong centers. Simple forms create an intense, powerful center.

**The Void**: The intensity of every center depends on the existence of a still place – an empty center. Empty spaces offer calm and contrast.

**Levels of Scale**: A strong center is made stronger partly by smaller strong centers contained in it, and partly by its larger strong centers which contain it. A balanced range of sizes is pleasing and beautiful.

**Positive Space**: A center should draw strength from the centers immediately adjacent. The background should reinforce rather than detract from the center.

**Local Symmetries**: The intensity of a center is increased by the extent to which other smaller centers are themselves arranged in locally symmetrical groups. Organic, small-scale symmetry works better than precise, overall symmetry.

**Strong Centers**: A strong center requires a field-like effect created by other centers. Good design offers areas of focus or weight.

**Boundaries**: The field-like effect is strengthened by the creation of a ring-like center. Outlines focus attention on the center.

**Deep Interlock and Ambiguity**: The intensity of a center can be increased when it is attached to nearby strong centers through a third set of strong centers that ambiguously belong to both. Looping, interconnected elements promote unity and grace.

**Gradients**: A center is strengthened by a graded series of different sized centers which then point to a new center. The proportional use of space and pattern creates harmony.

**Contrast**: A center is strengthened by the sharpness of distinction between itself and the surrounding centers. Unity is achieved with visible opposites.

**Roughness**: The way a center draws its strength from irregularities in sizes, shapes and arrangements. Texture and imperfections convey uniqueness and life.

**Alternating Repetition**: Centers are strengthened when they repeat, by the insertion of other centers between them. Repeating various elements creates a sense of order and harmony.

**Good Shape**: The strength of a center depends on its actual shape. Its boundaries and the space around it must be made up of strong centers. Simple forms create an intense, powerful center.

**Echoes**: The strength of a given center depends on similarities of angle and orientation. Similarities should repeat themselves throughout a design.

**Not Separateness**: The strength of a center depends on the extent to which that center is merged smoothly with surrounding centers. Designs should be connected and complementary, not egocentric and isolated.

**Simplicity and Inner Calm**: The strength of a center depends on its simplicity. Use only essentials and avoid extraneous elements.
Levels of Scale: A strong center is made stronger partly by smaller strong centers contained in it, and partly by its larger strong centers which contain it. A balanced range of sizes is pleasing and beautiful.

Positive Space: A center should draw strength from the centers immediately adjacent. The background should reinforce rather than detract from the center.

Local Symmetries: The intensity of a center is increased by the extent to which other smaller centers are themselves arranged in locally symmetrical groups. Organic, small-scale symmetry works better than precise, overall symmetry.

The Void: The intensity of every center depends on the existence of a still place – an empty center. Empty spaces offer calm and contrast.

Not Separateness: The strength of a center depends on the extent to which that center is merged smoothly with surrounding centers. Designs should be connected and complementary, not egocentric and isolated.

Good Shape: The strength of a center depends on its actual shape. Its boundaries and the space around it must be made up of strong centers. Simple forms create an intense, powerful center.

Roughness: The way a center draws its strength from irregularities in sizes, shapes and arrangements. Texture and imperfections convey uniqueness and life.

Alternating Repetition: Centers are strengthened when they repeat, by the insertion of other centers between them. Repeating various elements creates a sense of order and harmony.

Echoes: The strength of a given center depends on similarities of angle and orientation. Similarities should repeat themselves throughout a design.

Deep Interlock and Ambiguity: The intensity of a center can be increased when it is attached to nearby strong centers through a third set of strong centers that ambiguously belong to both. Looping, interconnected elements promote unity and grace.

Gradients: A center is strengthened by a graded series of different sized centers which then point to a new center. The proportional use of space and pattern creates harmony.

Contrast: A center is strengthened by the sharpness of distinction between itself and the surrounding centers. Unity is achieved with visible opposites.

Not Separateness: The strength of a center depends on the extent to which that center is merged smoothly with surrounding centers. Designs should be connected and complementary, not egocentric and isolated.

Simplicity and Inner Calm: The strength of a center depends on its simplicity. Use only essentials and avoid extraneous elements.
Center Properties
Alexander’s 15 Center Properties

Levels of Scale
Strong Centers
Boundaries
Alternating Repetition
Positive Space
Good Shape
Local Symmetries
Deep Interlock and Ambiguity
Contrast
Gradients
Roughness
Echoes
The Void
Simplicity and Inner Calm
Not Separateness

ea taxonomy / vocabulary of design elements that evoke the experience of quality in architecture,

“The Nature of Order” discernible in physicality.
# Observable / Attainable

<table>
<thead>
<tr>
<th>Alexander’s 15 Center Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levels of Scale</td>
</tr>
<tr>
<td>Strong Centers</td>
</tr>
<tr>
<td>Boundaries</td>
</tr>
<tr>
<td>Alternating Repetition</td>
</tr>
<tr>
<td>Positive Space</td>
</tr>
<tr>
<td>Good Shape</td>
</tr>
<tr>
<td>Local Symmetries</td>
</tr>
<tr>
<td>Deep Interlock and Ambiguity</td>
</tr>
<tr>
<td>Contrast</td>
</tr>
<tr>
<td>Gradients</td>
</tr>
<tr>
<td>Roughness</td>
</tr>
<tr>
<td>Echoes</td>
</tr>
<tr>
<td>The Void</td>
</tr>
<tr>
<td>Simplicity and Inner Calm</td>
</tr>
<tr>
<td>Not Separateness</td>
</tr>
</tbody>
</table>
Observable / Attainable

observable: *as in assessing the characteristics of an artifact*

attainable: *as in manipulation through design choices*

“Order, Wholeness” can be pursued in design!

<table>
<thead>
<tr>
<th>Alexander’s 15 Center Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levels of Scale</td>
</tr>
<tr>
<td>Strong Centers</td>
</tr>
<tr>
<td>Boundaries</td>
</tr>
<tr>
<td>Alternating Repetition</td>
</tr>
<tr>
<td>Positive Space</td>
</tr>
<tr>
<td>Good Shape</td>
</tr>
<tr>
<td>Local Symmetries</td>
</tr>
<tr>
<td>Deep Interlock and Ambiguity</td>
</tr>
<tr>
<td>Contrast</td>
</tr>
<tr>
<td>Gradients</td>
</tr>
<tr>
<td>Roughness</td>
</tr>
<tr>
<td>Echoes</td>
</tr>
<tr>
<td>The Void</td>
</tr>
<tr>
<td>Simplicity and Inner Calm</td>
</tr>
<tr>
<td>Not Separateness</td>
</tr>
</tbody>
</table>

observable: *as in assessing the characteristics of an artifact*

attainable: *as in manipulation through design choices*
“The concept extends to any space where objects & relationships are observed.”

Alexander
“The concept extends to any space where objects & relationships are observed.”

Alexander
“The concept extends to any space where objects & relationships are observed.”

Alexander

Postulation:

“The elements of order, the WHOLENESS, that Alexander describes for physical architecture are perceptible in any architectural system – specifically information systems.”

Waguespack
Center transliterates as **Choice** in a conceptual world of abstractions
Center transliterates as *Choice* in a conceptual world of abstractions

Alexander’s concepts are expressed in a vocabulary of physical, visible structure.
Center transliterates as **Choice** in a conceptual world of abstractions

- Alexander’s concepts are expressed in a vocabulary of physical, visible structure.
- All systems (e.g. IS systems) can be expressed as abstractions analogous to physical structure and relationships.
Alexander’s concepts are expressed in a vocabulary of physical, visible structure.

All systems (e.g. IS systems) can be expressed as abstractions analogous to physical structure and relationships.

To explain the “wholeness” of abstractions Alexander’s properties must be transliterated from a vocabulary of physical space to a vocabulary of cognitive space.

Center transliterates as **Choice** in a conceptual world of abstractions.
Center transliterates as **Choice** in a conceptual world of abstractions

- Alexander’s concepts are expressed in a vocabulary of physical, visible structure.

- All systems (e.g. IS systems) can be expressed as abstractions analogous to physical structure and relationships.

- To explain the “wholeness” of abstractions Alexander’s properties must be transliterated from a vocabulary of physical space to a vocabulary of cognitive space.

- In this cognitive space we use the term **choice** as the dual of Alexander’s center.
Alexander’s concepts are expressed in a vocabulary of physical, visible structure.

All systems (e.g. IS systems) can be expressed as abstractions analogous to physical structure and relationships.

To explain the “wholeness” of abstractions Alexander’s properties must be transliterated from a vocabulary of physical space to a vocabulary of cognitive space.

In this cognitive space we use the term choice as the dual of Alexander’s center.

Center properties of the physical transliterate as structural, relational, and/or congruence characteristics of choices.
Translating the Properties from Centers to Choices
Translating the Properties from Centers to Choices

<table>
<thead>
<tr>
<th>Alexander’s 15 Center Properties</th>
<th>Vocabulary of Choice Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levels of Scale</td>
<td>Stepwise Refinement</td>
</tr>
<tr>
<td>Strong Centers</td>
<td>Cohesion</td>
</tr>
<tr>
<td>Boundaries</td>
<td>Encapsulation</td>
</tr>
<tr>
<td>Alternating Repetition</td>
<td>Extensibility</td>
</tr>
<tr>
<td>Positive Space</td>
<td>Modularization</td>
</tr>
<tr>
<td>Good Shape</td>
<td>Correctness</td>
</tr>
<tr>
<td>Local Symmetries</td>
<td>Transparency</td>
</tr>
<tr>
<td>Deep Interlock and Ambiguity</td>
<td>Composition of Function</td>
</tr>
<tr>
<td>Contrast</td>
<td>Identity</td>
</tr>
<tr>
<td>Gradients</td>
<td>Scale</td>
</tr>
<tr>
<td>Roughness</td>
<td>User Friendliness</td>
</tr>
<tr>
<td>Echoes</td>
<td>Patterns</td>
</tr>
<tr>
<td>The Void</td>
<td>Programmability</td>
</tr>
<tr>
<td>Simplicity and Inner Calm</td>
<td>Reliability</td>
</tr>
<tr>
<td>Not Separateness</td>
<td>Elegance</td>
</tr>
<tr>
<td>Stepwise Refinement</td>
<td>elaborate</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>Cohesion</td>
<td>factor</td>
</tr>
<tr>
<td>Encapsulation</td>
<td>encapsulate</td>
</tr>
<tr>
<td>Extensibility</td>
<td>extend</td>
</tr>
<tr>
<td>Modularization</td>
<td>modularize</td>
</tr>
<tr>
<td>Correctness</td>
<td>align</td>
</tr>
<tr>
<td>Transparency</td>
<td>expose</td>
</tr>
<tr>
<td>Composition of Function</td>
<td>assemble</td>
</tr>
<tr>
<td>Identity</td>
<td>identify</td>
</tr>
<tr>
<td>Scale</td>
<td>focus</td>
</tr>
<tr>
<td>User Friendliness</td>
<td>accommodate</td>
</tr>
<tr>
<td>Patterns</td>
<td>pattern</td>
</tr>
<tr>
<td>Programmability</td>
<td>generalize</td>
</tr>
<tr>
<td>Reliability</td>
<td>normalize</td>
</tr>
<tr>
<td>Elegance</td>
<td>coordinate</td>
</tr>
</tbody>
</table>
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.
Advancing the credibility of choice properties
Is there evidence that the choice properties can be identified in systems that are generally accepted as exhibiting great design?
Advancing the credibility of choice properties

- Is there evidence that the choice properties can be identified in systems that are generally accepted as exhibiting great design?

- Are there modeling actions in design that can intentionally affect the intensities of the various properties of a choice?
Advancing the credibility of choice properties

- Is there evidence that the choice properties can be identified in systems that are generally accepted as exhibiting great design?

- Are there modeling actions in design that can intentionally affect the intensities of the various properties of a choice?

- Are such modeling actions transferable across paradigms of artifacts? - Across paradigms of modeling?
Evidence of Choice Properties in Systems and Modeling
Evidence of Choice Properties in Systems and Modeling

Choice Properties in Extant Systems
Evidence of Choice Properties in Systems and Modeling

- Choice Properties in Extant Systems
  - APACHE web server *(Design Principles & Practices)*
<table>
<thead>
<tr>
<th>Choice Property</th>
<th>Modeling Action</th>
<th>Action Definition</th>
<th>Apache Exemplars of Choice Property Strength</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>
<td>Filter chains (2.0)</td>
</tr>
<tr>
<td>2 Cohesion</td>
<td>factor</td>
<td>express as a product of factors</td>
<td>Resources pools</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>
<td>Platform independence (2.0)</td>
</tr>
<tr>
<td>4 Extensibility</td>
<td>extend</td>
<td>render something capable of expansion in scope, effect, or meaning</td>
<td>Apache server API’s (public)</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>
<td>Apache server API’s (private)</td>
</tr>
<tr>
<td>6 Correctness</td>
<td>align</td>
<td>put (things) into correct or appropriate relative positions</td>
<td>HTTP implementation</td>
</tr>
<tr>
<td>7 Transparency</td>
<td>expose</td>
<td>reveal the presence of (a quality or feeling)</td>
<td>Apache portable runtime (2.0)</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>
<td>Filter chains (2.0)</td>
</tr>
<tr>
<td>9 Identity</td>
<td>identify</td>
<td>establish or indicate who or what (someone or something) is</td>
<td>Management interfaces (2.0)</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>
<td>Core elements of server</td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>accommodate</td>
<td>fit in with the wishes or needs of</td>
<td>Flexible configuration &amp; management (2.0)</td>
</tr>
<tr>
<td>12 Patterns</td>
<td>pattern</td>
<td>give a regular or intelligible form to</td>
<td>Module design patterns</td>
</tr>
<tr>
<td>13 Programmability</td>
<td>generalize</td>
<td>make or become more widely or generally applicable</td>
<td>Hierarchical &amp; layered configuration</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>
<td>Process lifecycle &amp; resource management</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>
<td>Configuration change management</td>
</tr>
<tr>
<td>Choice Property</td>
<td>Modeling Action</td>
<td>Action Definition</td>
<td>Apache Exemplars of Choice Property Strength</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------</td>
<td>------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>1 Stepwise Refinement</td>
<td>elaborate</td>
<td>develop or present (a theory, policy, or system) in detail</td>
<td>Filter chains (2.0)</td>
</tr>
<tr>
<td>2 Cohesion</td>
<td>factor</td>
<td>express as a product of factors</td>
<td>Resources pools</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>
<td>Platform independence (2.0)</td>
</tr>
<tr>
<td>4 Extensibility</td>
<td>extend</td>
<td>render something capable of expansion in scope, effect, or meaning</td>
<td>Apache server API’s (public)</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>
<td>Apache server API’s (private)</td>
</tr>
<tr>
<td>6 Correctness</td>
<td>align</td>
<td>put (things) into correct or appropriate relative positions</td>
<td>HTTP implementation</td>
</tr>
<tr>
<td>7 Transparency</td>
<td>expose</td>
<td>reveal the presence of (a quality or feeling)</td>
<td>Apache portable runtime (2.0)</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>
<td>Filter chains (2.0)</td>
</tr>
<tr>
<td>9 Identity</td>
<td>identify</td>
<td>establish or indicate who or what (someone or something) is</td>
<td>Management interfaces (2.0)</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>
<td>Core elements of server</td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>accommodate</td>
<td>fit in with the wishes or needs of</td>
<td>Flexible configuration &amp; management (2.0)</td>
</tr>
<tr>
<td>12 Patterns</td>
<td>pattern</td>
<td>give a regular or intelligible form to</td>
<td>Module design patterns</td>
</tr>
<tr>
<td>13 Programmability</td>
<td>generalize</td>
<td>make or become more widely or generally applicable</td>
<td>Hierarchical &amp; layered configuration</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>
<td>Process lifecycle &amp; resource management</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>
<td>Configuration change management</td>
</tr>
</tbody>
</table>
Evidence of Choice Properties in Systems and Modeling

- Choice Properties in Extant Systems
  - ✅ APACHE web server (Design Principles & Practices)
Evidence of Choice Properties in Systems and Modeling

Choice Properties in Extant Systems

- APACHE web server (Design Principles & Practices)
- network architecture
Evidence of Choice Properties in Systems and Modeling

- Choice Properties in Extant Systems
  - APACHE web server (Design Principles & Practices)
  - network architecture
- Choice Properties Mapped to Modeling
Evidence of Choice Properties in Systems and Modeling

- Choice Properties in Extant Systems
  - APACHE web server *(Design Principles & Practices)*
  - network architecture

- Choice Properties Mapped to Modeling
  - object-oriented modeling *(monograph Ch12)*
Evidence of Choice Properties in Systems and Modeling

- Choice Properties in Extant Systems
  - APACHE web server *(Design Principles & Practices)*
  - network architecture

- Choice Properties Mapped to Modeling
  - object-oriented modeling *(monograph Ch12)*
Evidence of Choice Properties in Systems and Modeling

- Choice Properties in Extant Systems
  - APACHE web server (Design Principles & Practices)
  - network architecture

- Choice Properties Mapped to Modeling
  - object-oriented modeling (monograph Ch12)
  - relational modeling (monograph Ch13)
Evidence of Choice Properties in Systems and Modeling

Choice Properties in Extant Systems
- ✓ APACHE web server (Design Principles & Practices)
- □ network architecture

Choice Properties Mapped to Modeling
- ✓ object-oriented modeling (monograph Ch12)
- ✓ relational modeling (monograph Ch13)
Evidence of Choice Properties in Systems and Modeling

- **Choice Properties in Extant Systems**
  - APACHE web server (Design Principles & Practices)
  - Network architecture

- **Choice Properties Mapped to Modeling**
  - Object-oriented modeling (monograph Ch12)
  - Relational modeling (monograph Ch13)
  - Agile methodology - SCRUM (HICSS-45)
Evidence of Choice Properties in Systems and Modeling

Choice Properties in Extant Systems
- APACHE web server *(Design Principles & Practices)*
- network architecture

Choice Properties Mapped to Modeling
- object-oriented modeling *(monograph Ch12)*
- relational modeling *(monograph Ch13)*
- agile methodology - SCRUM *(HICSS-45)*
Evidence of Choice Properties in Systems and Modeling

Choice Properties in Extant Systems
- APACHE web server (Design Principles & Practices)
- network architecture

Choice Properties Mapped to Modeling
- object-oriented modeling (monograph Ch12)
- relational modeling (monograph Ch13)
- agile methodology - SCRUM (HICSS-45)
- business process modeling
Evidence of Choice Properties in Systems and Modeling

Choice Properties in Extant Systems
- APACHE web server (Design Principles & Practices)
- network architecture

Choice Properties Mapped to Modeling
- object-oriented modeling (monograph Ch12)
- relational modeling (monograph Ch13)
- agile methodology - SCRUM (HICSS-45)
- business process modeling

Other System Domains
Evidence of Choice Properties in Systems and Modeling

Choice Properties in Extant Systems
- APACHE web server (Design Principles & Practices)
- network architecture

Choice Properties Mapped to Modeling
- object-oriented modeling (monograph Ch12)
- relational modeling (monograph Ch13)
- agile methodology - SCRUM (HICSS-45)

Other System Domains
- music
Evidence of Choice Properties in Systems and Modeling

- **Choice Properties in Extant Systems**
  - **✓** APACHE web server (Design Principles & Practices)
  - **☐** network architecture

- **Choice Properties Mapped to Modeling**
  - **✓** object-oriented modeling (monograph Ch12)
  - **✓** relational modeling (monograph Ch13)
  - **✓** agile methodology - SCRUM (HICSS-45)
  - **☐** business process modeling

- **Other System Domains**
  - **☐** music
  - **☐** english composition
Design (vt): The composition of deliberate formative decisions in the process of creating an artifact.
The Design Process
The Design Process

Stakeholder intentions
requirement elements
model elements
design elements
The Design Process

“rules of thumb”
Traditional patterns

Stakeholder intentions
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

Representational
paradigm or Ontology

Resulting
Design
Elements

©LJW, 2012 All Rights Reserved
The Design Process

Choice Property-Driven Design Principles

Resulting Design Elements

Stakeholder intentions
requirement elements
model elements
design elements

Representational paradigm or Ontology
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

Choice Property-Driven
Design Principles

- implementation
  - hardware & software architecture
- threshold
  - interfaces
- expectation
  - requirements engineering & specification
- mindset
  - object oriented “systems think”
Pursuing a Universal Foundation of System Design Quality

from the beauty in nature

properties of order in nature

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

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

from the beauty in nature
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

properties of order in nature

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

Properties of order in nature:
- Modularization
- Cohesion
- Encapsulation
- Composition of Functionality
- Scale
- Correctness
- User Friendliness
- Extensibility
- Patterns
- Reliability
- Transparency
- Programmability
- Identity
- Elegance

System design actions and choice properties:
- Elaborate
- Modularize
- Factor
- Encapsulate
- Assemble
- Focus
- Align
- Accommodate
- Extend
- Pattern
- Normalize
- Expose
- Generalize
- Identify
- Coordinate
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

object-oriented models that thrive

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
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

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

object-oriented models that thrive

object-oriented models that thrive

Relational data ontology

relational data models that thrive

Relational data ontology

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

correctness scale extensibility
patterns reliability transparency
programmability identity elegance

Stepwise Refinement Modularization Cohesion
Encapsulation Composition of Function Scale

Object-oriented ontology

Relational data ontology

SCRUM ontology

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

from the beauty in nature to the quality in systems

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

system design actions and choice properties

Object-oriented ontology
Relational data ontology
SCRUM ontology

Business process modeling vocabulary or ontology?
Toward a Vision of Thriving Systems
“The Nature of Order” is evident in choice properties observable in information systems!
“The Nature of Order” is evident in choice properties observable in information systems!

The translated properties denote design features that may appear to be discrete.

Toward a Vision of Thriving Systems
Toward a Vision of Thriving Systems

“The Nature of Order” is evident in choice properties observable in information systems!

The translated properties denote design features that may appear to be discrete.

Might property strength and interaction resonate as quality?
“The Nature of Order” is evident in choice properties observable in information systems!

The translated properties denote design features that may appear to be discrete.

Might property strength and interaction resonate as quality?

Might a taxonomy of quality emerge?
Choice Properties

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
4. Extensibility
5. Modularization
6. Correctness
7. Transparency
8. Composition of Function
9. Identity
10. Scale
11. User Friendliness
12. Patterns
13. Programmability
14. Reliability
15. Elegance

©LJW, 2012 All Rights Reserved
# Property Interaction

**Alexander’s Property Support Intersection**

<table>
<thead>
<tr>
<th>Alexander’s Property Support Intersection</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row item supported by column</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 Levels of Scale</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 Strong Centers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 Boundaries</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 Alternating Repetition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Positive Space</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 Good Shape</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7 Local Symmetries</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8 Deep Interlock and Ambiguity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9 Contrast</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 Gradients</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 Roughness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12 Echoes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 The Void</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14 Simplicity and Inner Calm</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15 Not Separateness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Property Interaction

**Alexander’s Property Support Intersection**

<table>
<thead>
<tr>
<th>Row item supported by column</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levels of Scale</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strong Centers</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boundaries</td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alternating Repetition</td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Positive Space</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Good Shape</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local Symmetries</td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deep Interlock and Ambiguity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contrast</td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gradients</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Roughness</td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Echoes</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The Void</td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simplicity and Inner Calm</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Not Separateness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Property Interaction

<table>
<thead>
<tr>
<th>Row item supported by column</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levels of Scale</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strong Centers</td>
<td></td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boundaries</td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alternating Repetition</td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Positive Space</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Good Shape</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local Symmetries</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deep Interlock and Ambiguity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contrast</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gradients</td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Roughness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>Echoes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The Void</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simplicity and Inner Calm</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>Not Separateness</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>Alexander’s Property Support Intersection Row item supported by column</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Levels of Scale</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strong Centers</td>
<td>•</td>
<td></td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boundaries</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alternating Repetition</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Positive Space</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Good Shape</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local Symmetries</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deep Interlock and Ambiguity</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Contrast</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Gradients</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Roughness</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Echoes</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>The Void</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Simplicity and Inner Calm</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>Not Separateness</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td></td>
</tr>
</tbody>
</table>
Choice Properties Substituted for Center Properties

- Extensibility
- Transparencies
- Cohesion
- Modularization
- Correctness
- Composition of Function
- Elegance
- Identity
- Programmability
- Transparency
- Reliability
- Patterns
- Extensibility
- User Friendliness

Stepwise Refinement

Encapsulation

Scale

Correctness
## Coherence Analysis

Coherence is the sum of the fraction of supporting properties that mutually overlap between two properties \( (0 \leq \varsigma \leq 2) \). The distance is \( (2 - \varsigma) \).

<table>
<thead>
<tr>
<th>Center Properties</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levels of Scale</td>
<td>1</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strong Centers</td>
<td>2</td>
<td>1.58</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boundaries</td>
<td>3</td>
<td>1.17</td>
<td>0.67</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alternating Repetition</td>
<td>4</td>
<td>0.75</td>
<td>1.33</td>
<td>1.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Positive Space</td>
<td>5</td>
<td>0.50</td>
<td>1.13</td>
<td>1.13</td>
<td>1.13</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Good Shape</td>
<td>6</td>
<td>1.25</td>
<td>1.71</td>
<td>1.13</td>
<td>0.83</td>
<td>1.25</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local Symmetries</td>
<td>7</td>
<td>1.50</td>
<td>1.17</td>
<td>1.58</td>
<td>1.17</td>
<td>0.88</td>
<td>1.25</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deep Interlock and Ambiguity</td>
<td>8</td>
<td>1.58</td>
<td>1.00</td>
<td>1.33</td>
<td>1.00</td>
<td>1.42</td>
<td>1.42</td>
<td>1.17</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contrast</td>
<td>9</td>
<td>1.58</td>
<td>1.00</td>
<td>1.33</td>
<td>1.00</td>
<td>1.42</td>
<td>1.31</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gradients</td>
<td>10</td>
<td>1.21</td>
<td>1.07</td>
<td>1.07</td>
<td>1.07</td>
<td>0.66</td>
<td>1.20</td>
<td>1.21</td>
<td>0.76</td>
<td>1.69</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Roughness</td>
<td>11</td>
<td>1.17</td>
<td>1.33</td>
<td>1.33</td>
<td>0.67</td>
<td>1.42</td>
<td>0.54</td>
<td>1.58</td>
<td>1.33</td>
<td>1.00</td>
<td>1.38</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Echoes</td>
<td>12</td>
<td>1.58</td>
<td>1.00</td>
<td>1.33</td>
<td>1.33</td>
<td>0.83</td>
<td>1.13</td>
<td>1.58</td>
<td>1.33</td>
<td>1.33</td>
<td>0.76</td>
<td>1.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>The Void</td>
<td>13</td>
<td>1.17</td>
<td>1.33</td>
<td>1.33</td>
<td>1.33</td>
<td>0.83</td>
<td>1.13</td>
<td>0.75</td>
<td>1.33</td>
<td>1.33</td>
<td>1.07</td>
<td>1.33</td>
<td>1.33</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Simplicity and Inner Calm</td>
<td>14</td>
<td>1.55</td>
<td>0.90</td>
<td>1.63</td>
<td>1.27</td>
<td>1.03</td>
<td>1.35</td>
<td>1.55</td>
<td>1.27</td>
<td>1.27</td>
<td>0.97</td>
<td>1.27</td>
<td>0.90</td>
<td>1.63</td>
<td>0.00</td>
</tr>
<tr>
<td>Not Separateness</td>
<td>15</td>
<td>1.61</td>
<td>1.38</td>
<td>1.38</td>
<td>1.38</td>
<td>1.20</td>
<td>0.93</td>
<td>1.21</td>
<td>1.38</td>
<td>0.45</td>
<td>1.71</td>
<td>1.07</td>
<td>1.38</td>
<td>1.07</td>
<td>1.66</td>
</tr>
</tbody>
</table>
## Coherence Analysis

<table>
<thead>
<tr>
<th>Choice Properties</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stepwise Refinement</td>
<td></td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cohesion</td>
<td>2</td>
<td>1.58</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Encapsulation</td>
<td>3</td>
<td>1.17</td>
<td>0.67</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extensibility</td>
<td>4</td>
<td>0.75</td>
<td>1.33</td>
<td>1.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modularization</td>
<td>5</td>
<td>0.50</td>
<td>1.13</td>
<td>1.13</td>
<td>1.13</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Correctness</td>
<td>6</td>
<td>1.25</td>
<td>1.71</td>
<td>1.13</td>
<td>0.83</td>
<td>1.25</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transparency</td>
<td>7</td>
<td>1.50</td>
<td>1.17</td>
<td>1.58</td>
<td>1.17</td>
<td>0.88</td>
<td>1.25</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Composition of Function</td>
<td>8</td>
<td>1.58</td>
<td>1.00</td>
<td>1.33</td>
<td>1.00</td>
<td>1.42</td>
<td>1.42</td>
<td>1.17</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identity</td>
<td>9</td>
<td>1.58</td>
<td>1.00</td>
<td>1.33</td>
<td>1.00</td>
<td>1.42</td>
<td>1.13</td>
<td>1.17</td>
<td>1.33</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scale</td>
<td>10</td>
<td>1.21</td>
<td>1.07</td>
<td>1.07</td>
<td>1.07</td>
<td>0.66</td>
<td>1.20</td>
<td>1.21</td>
<td>0.76</td>
<td>1.69</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User Friendliness</td>
<td>11</td>
<td>1.17</td>
<td>1.33</td>
<td>1.33</td>
<td>0.67</td>
<td>1.42</td>
<td>0.54</td>
<td>1.58</td>
<td>1.33</td>
<td>1.00</td>
<td>1.38</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Patterns</td>
<td>12</td>
<td>1.58</td>
<td>1.00</td>
<td>1.33</td>
<td>1.33</td>
<td>0.83</td>
<td>1.13</td>
<td>1.58</td>
<td>1.33</td>
<td>1.33</td>
<td>0.76</td>
<td>1.00</td>
<td>0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Programmability</td>
<td>13</td>
<td>1.17</td>
<td>1.33</td>
<td>1.33</td>
<td>1.33</td>
<td>0.83</td>
<td>1.13</td>
<td>0.75</td>
<td>1.33</td>
<td>1.33</td>
<td>1.07</td>
<td>1.33</td>
<td>1.33</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Reliability</td>
<td>14</td>
<td>1.55</td>
<td>0.90</td>
<td>1.63</td>
<td>1.27</td>
<td>1.03</td>
<td>1.35</td>
<td>1.55</td>
<td>1.27</td>
<td>1.27</td>
<td>0.97</td>
<td>1.27</td>
<td>0.90</td>
<td>1.63</td>
<td>0.00</td>
</tr>
<tr>
<td>Elegance</td>
<td>15</td>
<td>1.61</td>
<td>1.38</td>
<td>1.38</td>
<td>1.38</td>
<td>1.20</td>
<td>0.93</td>
<td>1.21</td>
<td>1.38</td>
<td>0.45</td>
<td>1.71</td>
<td>1.07</td>
<td>1.38</td>
<td>1.07</td>
<td>1.66</td>
</tr>
</tbody>
</table>

Coherence is the sum of the fraction of supporting properties that mutually overlap between two properties (0 ≤ ç ≤ 2). The distance is (2 - ç).
Choice Properties

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
4. Extensibility
5. Modularization
6. Correctness
7. Transparency
8. Composition of Function
10. Scale
11. User Friendliness
12. Patterns
13. Programmability
14. Reliability
15. Elegance
9. Identity
Choice Property Clusters

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
4. Extensibility
5. Modularization
6. Correctness
7. Transparency
8. Composition of Function
9. Identity
10. Scale
11. User Friendliness
12. Patterns
13. Programmability
14. Reliability
15. Elegance
Clusters with Supporting Properties

<table>
<thead>
<tr>
<th>Clusters</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Stepwise Refinement</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Modularization</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>7</td>
<td>9</td>
<td>11</td>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B Cohesion</td>
<td>4</td>
<td>7</td>
<td>9</td>
<td>10</td>
<td>13</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 Encapsulation</td>
<td>2</td>
<td>4</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C Composition of Function</td>
<td>4</td>
<td>5</td>
<td>8</td>
<td>9</td>
<td>11</td>
<td>12</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 Scale</td>
<td>1</td>
<td>2</td>
<td>7</td>
<td>9</td>
<td>11</td>
<td>12</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D Correctness</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>10</td>
<td>14</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E Patterns</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14 Reliability</td>
<td>6</td>
<td>7</td>
<td>12</td>
<td>13</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F Transparency</td>
<td>1</td>
<td>5</td>
<td>9</td>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 Programmability</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G Identity</td>
<td>3</td>
<td>5</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>13</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15 Elegance</td>
<td>3</td>
<td>5</td>
<td>8</td>
<td>10</td>
<td>11</td>
<td>13</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H Correctness</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>10</td>
<td>14</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I Patterns</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14 Reliability</td>
<td>6</td>
<td>7</td>
<td>12</td>
<td>13</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J Correctness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>9</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K Stepwise Refinement</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Modularization</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>7</td>
<td>9</td>
<td>11</td>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L Correctness</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>10</td>
<td>14</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M Patterns</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 Programmability</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N Stepwise Refinement</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Modularization</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>7</td>
<td>9</td>
<td>11</td>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O Correctness</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>10</td>
<td>14</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P Patterns</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 Programmability</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q Correctness</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R Patterns</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 Programmability</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S Correctness</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>10</td>
<td>14</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T Patterns</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 Programmability</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U Correctness</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>V Patterns</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 Programmability</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W Correctness</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X Patterns</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 Programmability</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Y Correctness</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 User Friendliness</td>
<td>2</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>12</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Z Patterns</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Property Cluster $\implies$ Design Quality
Property Cluster ==> Design Quality

- Each cluster reflects a unique blend of property resonance
Property Cluster ==> Design Quality

- Each cluster reflects a unique blend of property resonance
- Each cluster explains a particular quality of design
Property Cluster ==> Design Quality

- Each cluster reflects a unique blend of property resonance
- Each cluster explains a particular quality of design
- Each cluster describes a goal set in design that responds to stakeholder intentions
Design Quality Naming
Design Quality Naming

1. Stepwise Refinement

5. Modularization

goal-directed decomposition reflects stakeholder perception of relationships among concepts parts are essential to the distribution and tolerance of complexity
Design Quality Naming

Divisibility

1. Stepwise Refinement

5. Modularization
Design Quality Naming

Divisibility

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
5. Modularization

cohesion reflects choice self-sufficiency: well-formed with cogency

encapsulation bounds but also interfaces the choice to the surrounding collection
Design Quality Naming

Divisibility

1. Stepwise Refinement
5. Modularization

2. Cohesion
3. Encapsulation

Factorability
Design Quality Naming

Divisibility

- Composable choices enable growth in capacity or complexity; new choices composed from existing ones.
- Scale reflects a useful granularity of attention or focus.

Factorability

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
8. Composition of Function
5. Modularization
10. Scale
Design Quality Naming

Divisibility

1. Stepwise Refinement
5. Modularization

Constructibility

8. Composition of Function
10. Scale

Factorability

2. Cohesion
3. Encapsulation
Design Quality Naming

Divisibility

Constructibility

Factorability

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
5. Modularization
8. Composition of Function
10. Scale

soundness of individuals, internal stability and structural independence

opportunity of combining to build larger/more capable arrangements
Design Quality Naming

Divisibility

Constructibility

Scalability

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
4. Factorability
5. Modularization
6. Composition of Function
7. Scale
8. Division
9. Constructibility
10. Factorability
divisible, sound, static structure: building blocks; iteratively decomposed and assembled, separating and insulating concerns

scalability enabling composition, fusing independent self-sufficiency to span the breadth and width of stakeholder intentions
Design Quality Naming

- Divisibility
- Constructibility
- Scalability

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
4. Modularity
5. Modularization
6. Composition of Function
7. Scale
8. Robustness
9. Factorability
10. Scale
choice alignment with intentions (dynamically) a perception of “what is natural,” “seeing” what you expect reinforces reliance and trust
Design Quality Naming

- 1. Stepwise Refinement
- 2. Cohesion
- 3. Encapsulation
- 5. Modularization
- 8. Composition of Function
- 10. Scale
- 6. Correctness
- 11. User Friendliness

Divisibility
Constructibility
Scalability

Robustness
Factorability

Confidence
Design Quality Naming

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
5. Modularization
8. Composition of Function
10. Scale
12. Patterns
11. User Friendliness
14. Reliability
6. Correctness

Divisibility
Constructibility
Scalability

Robustness
Factorability
Confidence

proven formulae or techniques enable repeatable success consistently safe practice continues success with compatible challenges
Design Quality Naming

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
5. Modularization
8. Composition of Function
10. Scale

Divisibility

Constructibility

Scalability

Robustness

Factorability

Predictability

12. Patterns
14. Reliability

6. Correctness
11. User Friendliness

Confidence
Design Quality Naming

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
4. Transparency
5. Modularization
6. Correctness
7. Transparency
8. Composition of Function
9. Scale
10. Scale
11. User Friendliness
12. Patterns
13. Programmability
14. Reliability

keeping “technology” out of the way of intentions

enabling users to recognize then apply choices to their needs
Design Quality Naming

- Divisibility
  - 5. Modularization
- Constructibility
  - 1. Stepwise Refinement
  - 2. Cohesion
  - 3. Encapsulation
- Scalability
  - 8. Composition of Function
  - 10. Scale
- Robustness
  - 6. Correctness
- Factorability
  - 7. Transparency
  - 9. Identity
- Usability
  - 11. User Friendliness
  - 13. Programmability
- Predictability
  - 12. Patterns
  - 14. Reliability
- Confidence
  - 15. Elegance

fusing the conceptual with the linguistic (concise, precise, clear, distinct)
realizing resonance between choice and intentions like the “ring” in a chord
Design Quality Naming

- Divisibility
  - Modularization
  - Composition of Function
- Constructibility
  - Stepwise Refinement
  - Cohesion
  - Encapsulation
- Scalability
  - Refinement
  - Scale
- Intuitiveness
  - Identity
- Predictability
  - Patterns
  - Reliability
- Robustness
  - Correctness
  - Transparency
- Factorability
  - Reliability
  - Programmability
- Usability
  - Patterns
  - Reliability
- Elegance

- Achieving confidence in the “now”
- Enabling alignment into the future to embrace change
usability is the ease with which the user can grasp and apply a choice

intuitiveness expresses the “naturalness” of the choice’s expression of the intention it is supposed to represent (or implement)
Design Quality Naming

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
4. Extensibility
5. Modularization
6. Correctness
7. Transparency
8. Composition of Function
9. Identity
10. Scale
11. User Friendliness
12. Patterns
13. Programmability
14. Reliability
15. Elegance

Divisibility
Constructibility
Scalability
Intuitiveness
Effectiveness
Robustness
Factorability
Usability
Confidence
Fidelity

fidelity anchors choices in intentions
predictability fosters choice evolution with anticipated intentions, enabling long-term viability
Effectively representing intentions: choices that are understandable and applicable

Choices aligned in the current reality; poised to respond to change; to grow predictably and evolve.
living structure

robustness: soundness of structure, integrity of form, capacity for stability

vitality: alignment with stakeholder intentions, capacity for useful growth and unfolding over time
Design Quality Naming

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
5. Modularization
8. Composition of Function
10. Scale
6. Correctness
7. Transparency
9. Identity
15. Elegance
12. Patterns
14. Reliability
11. User Friendliness
13. Programmability
4. Extensibility

- Divisibility
- Constructibility
- Scalability
- Intuitiveness
- Effectiveness
- Predictability
- Sustainability

- Robustness
- Factorability
- Thriving
- Usability
- Vitality
- Confidence
- Fidelity
Design Quality Naming

1. Stepwise Refinement
2. Cohesion
3. Encapsulation
4. Extensibility
5. Modularization
6. Correctness
7. Transparency
8. Composition of Function
9. Identity
10. Scale
11. User Friendliness
12. Patterns
13. Programmability
14. Reliability
15. Elegance

Robustness
Thriving
Vitality
Vitruvius
De architectura
78 BC
Vitruvius
De architectura
78 BC

Vitruvius
De architectura
78 BC

1. Stepwise Refinement
5. Modularization

2. Cohesion
3. Encapsulation
10. Scale

8. Composition of Function

9. Identity

7. Transparency
13. Programmability

6. Correctness
11. User Friendliness
4. Extensibility

12. Patterns
14. Reliability
15. Elegance

firmitas
strength

venustas
beauty

utilitas
functionality
Thriving Systems Qualities

1. Stepwise Refinement
5. Modularization
8. Composition of Function
10. Scale

2. Cohesion
3. Encapsulation

9. Identity

7. Transparency
13. Programmability
15. Elegance

12. Patterns
14. Reliability

6. Correctness
11. User Friendliness
4. Extensibility
A Thriving System exhibits the confluence of design qualities described by *robustness* and *vitality* –
A Thriving System exhibits the confluence of design qualities described by *robustness* and *vitality* – beyond existing, beyond functional, beyond surviving.
A Thriving System exhibits the confluence of design qualities described by *robustness* and *vitality* –

*beyond existing, beyond functional, beyond surviving.*

It thrives not only because it supports and aligns with the stakeholders' intentions in the “now,” but –
A Thriving System exhibits the confluence of design qualities described by *robustness* and *vitality* –

\[ \text{beyond existing, beyond functional, beyond surviving.} \]

It thrives not only because it supports and aligns with the stakeholders’ intentions in the “now,” but –

It actually promotes the unfolding of those intentions through the conceptual clarity and efficiency with which it represents them – the symbiosis that great design has with an authentic requirement.
A Thriving System exhibits the confluence of design qualities described by *robustness* and *vitality* –

*beyond existing, beyond functional, beyond surviving.*

It thrives not only because it supports and aligns with the stakeholders‘ intentions in the “now,” but –

It actually promotes the unfolding of those intentions through the conceptual clarity and efficiency with which it represents them – the symbiosis that great design has with an authentic requirement.

Great design meets both “fields” of challenge: a model with strength in all the Thriving Systems qualities enumerated above, but inexorably grounded on an authentic representation of stakeholder intentions.
Shaping A Mindset For Great Design
Perceive the wholeness and the impact of individual design choices on the system as a whole – not only in the static present, but in the dynamic unfolding of the stakeholders’ perspectives of life; in the system they will live in.
Perceive the wholeness and the impact of individual design choices on the system as a whole – not only in the static present, but in the dynamic unfolding of the stakeholders’ perspectives of life; in the system they will live in.

Focus on why you use the modeling tools – not on the tools themselves. Redirect decision-making energy to the questions: “What does life mean to these stakeholders?” and “How does each choice increase the life in the system by fulfilling the stakeholders’ evolving concerns?”

Les Waguespack
Shaping A Mindset For Great Design

Perceive the wholeness and the impact of individual design choices on the system as a whole – not only in the static present, but in the dynamic unfolding of the stakeholders’ perspectives of life; in the system they will live in.

Focus on why you use the modeling tools – not on the tools themselves. Redirect decision-making energy to the questions: “What does life mean to these stakeholders?” and “How does each choice increase the life in the system by fulfilling the stakeholders’ evolving concerns?”

Les Waguespack
Monograph in Two Parts
Monograph in Two Parts

- Thriving Systems Theory
  - a vocabulary of elemental properties describing system elements appropriate to a particular domain
  - a taxonomy of system quality resulting from the interplay of those elemental properties
Monograph in Two Parts

Thriving Systems Theory
- a vocabulary of elemental properties describing system elements appropriate to a particular domain
- a taxonomy of system quality resulting from the interplay of those elemental properties

Metaphor-Driven Modeling
- metaphorology and its role in both the conception and communication of knowledge
- metaphor as the implement of design in analysis, representation and realization of information systems
Monograph in Two Parts

- **Thriving Systems Theory**
  - a vocabulary of elemental properties describing system elements appropriate to a particular domain
  - a taxonomy of system quality resulting from the interplay of those elemental properties

- **Metaphor-Driven Modeling**
  - metaphorology and its role in both the conception and communication of knowledge
  - metaphor as the implement of design in analysis, representation and realization of information systems
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.
YOU ENGINEERS HAVE DONE NOTHING ON MY PROJECT. YOU JUST KEEP SAYING I HAVEN'T GIVEN YOU SUFFICIENT REQUIREMENTS!

I DON'T KNOW WHAT ELSE YOU NEED AND YOU WON'T TELL ME WHAT YOU NEED!! IS THIS JUST YOUR WAY OF AVOIDING WORK?!!

I'LL BET YOU REGRET CHOOSING MARKETING AS A CAREER PATH.

IT LOOKS LIKE A LOT OF WORK.
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.