Thriving System Design Principles Mapped to Relational

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

Bentley University
Waltham Massachusetts

Working Paper - 2012

Overview

- Review / Recap of Alexander's
  - Nature of Order
  - Wholeness
  - Centers and their properties
- Can we map centers, center properties, principles of thriving systems to Relational?

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!

Wholeness & Centers

- 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."

- "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."

Center Properties

- Alexander identifies fifteen properties of centers that contribute to the degree of life experienced by an observer.
- "Quantum mechanics asserts, via the mathematics, that particles are physically affected in their behavior by the wholeness of the space in which they move. ... [Wholeness] is not restricted to buildings or works of art, but is valid and essential even in those parts of the world that we have historically believed to be mechanical in nature." (Alexander)

Center Properties

Alexander's 15 Properties of Strong Centers
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

Choices as Centers

- To apply Alexander's concepts of physical structure to information systems they must first be translated from a language of physical space to a language of cognitive space where physical position and distance correspond to concepts of consonance in "fields" populated by abstractions rather than shapes. The term choice serves well for that translation of Alexander's term center into this cognitive space.
- Center ==> Choice

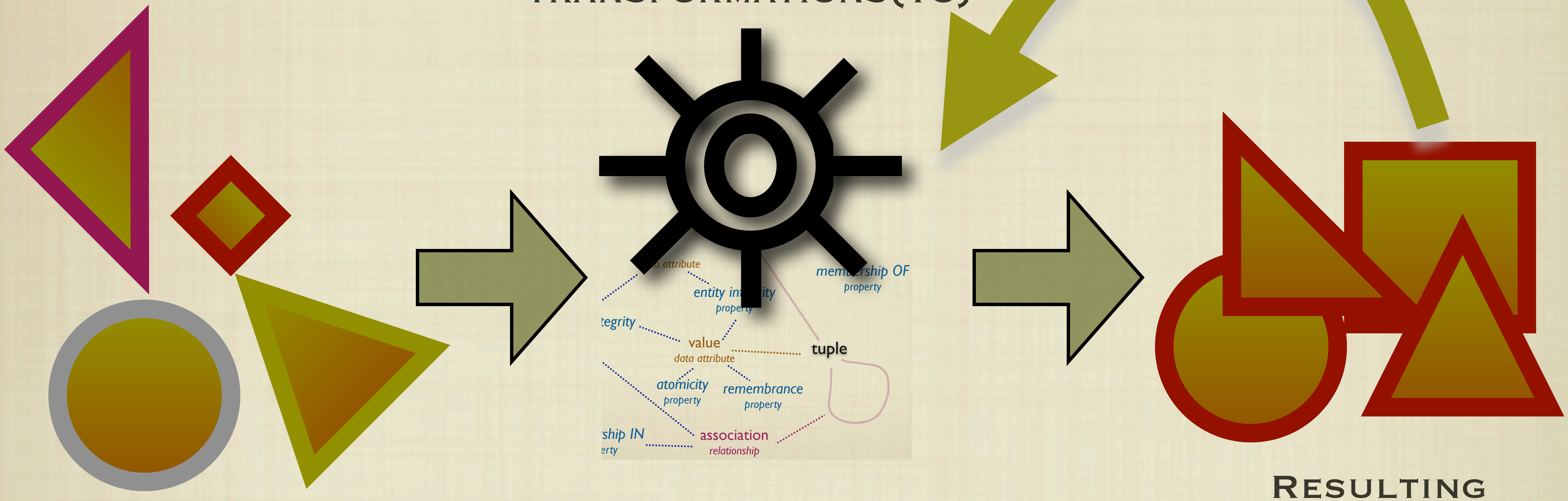
Properties Mapped

Alexander's 15 Properties of Strong Centers
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

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

The Design Process

Choice Property transformations(15)

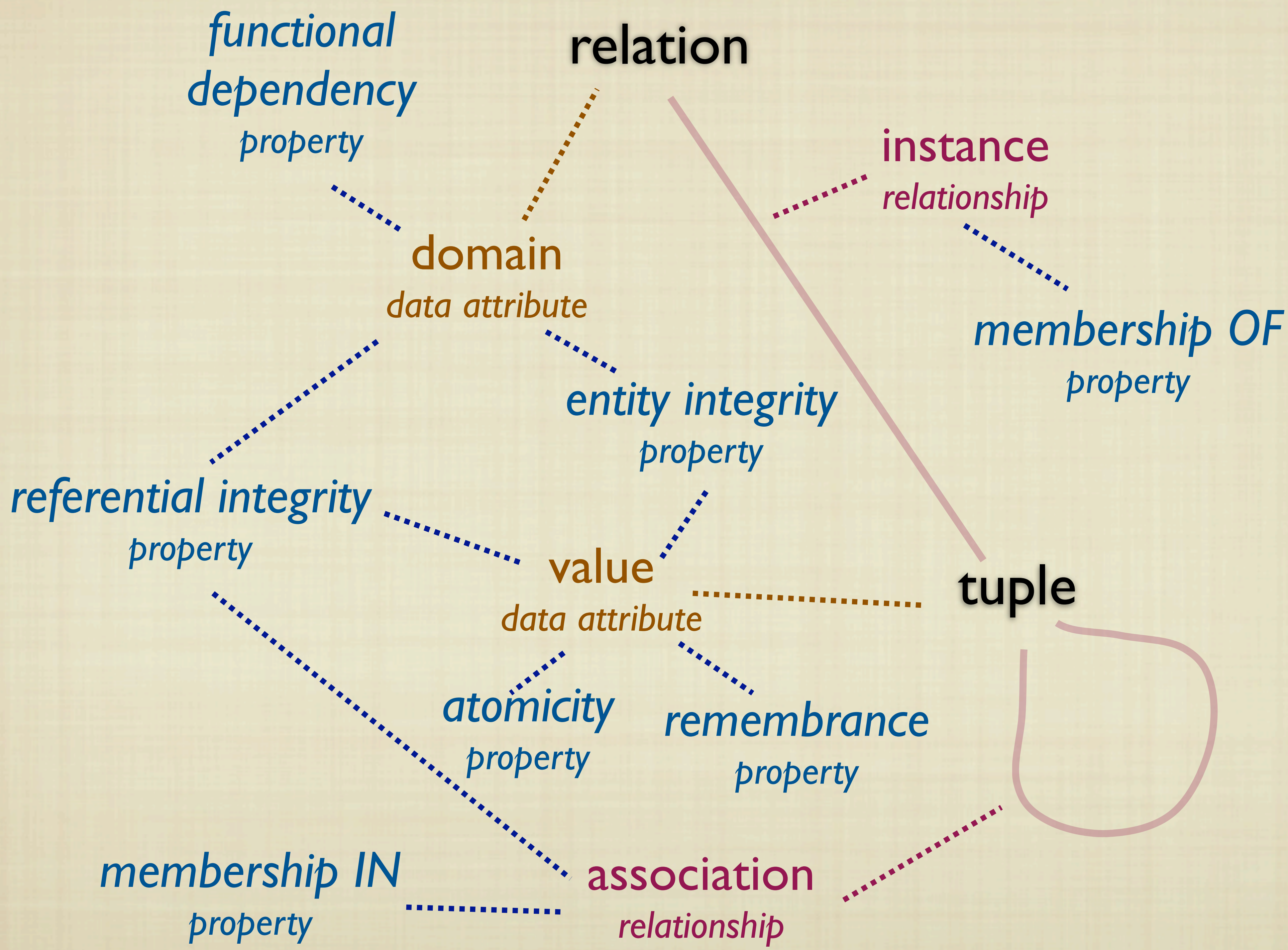


Stakeholder intensions
requirement elements
model elements
design elements

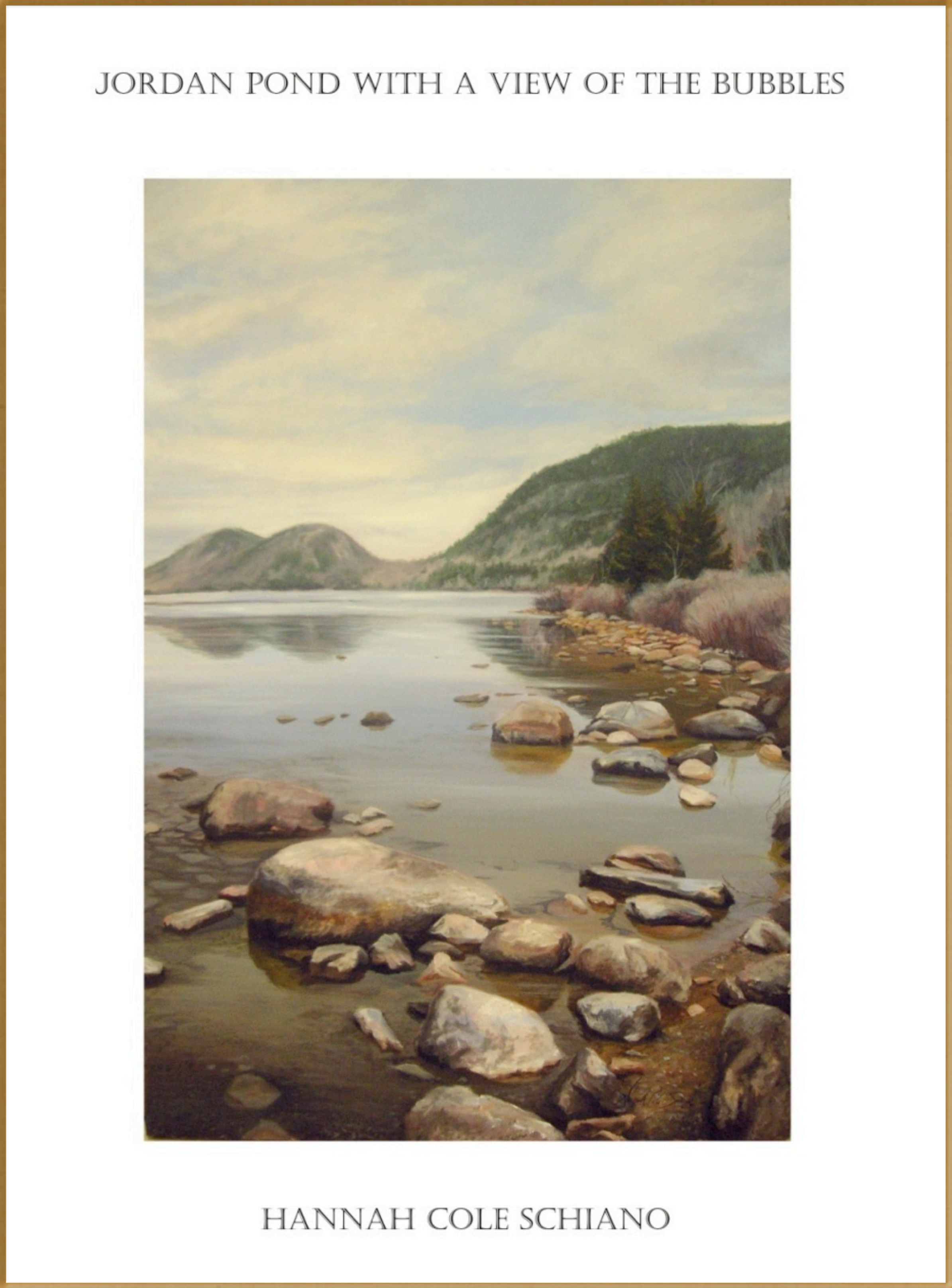
Representational paradigm or Ontology

Resulting Design Elements

Relational Ontology



Center Properties



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.

1. Stepwise Refinement: A system exhibits *stepwise refinement* if its components scale-up or scale down to reflect divide and conquer analysis and design at various levels. Thus, an observer of such a system can “zoom in” and “zoom out” and still retain a useful perspective. Furthermore, the composition of subsystems or system modules should effectively represent the primary concerns of the system as a whole.

Mapped to Relational

In the relational paradigm the strength of the information that a relation represents derives from the choice of attributes and the interdependence that form that entity of knowledge. Each relation depicts a cohesive, encapsulated and distinct segment of knowledge. The scope of knowledge included in any particular model is constructed by the aggregation of these distinct segments interwoven through their explicit relationships. A whole model is built up stepwise as the “subset of the universe” chosen for the model (its intension) is systematically surveyed, cataloged and defined in the collection of relations. Each relation’s integrity is achieved through its independent correctness separate and distinct from even those relations with which is maintains relationships. But the correctness of the whole proceeds from the assembly of the entire set of relations that together describe the reach of a model’s responsibilities.

Strong Centers

A strong center requires a field-like effect created by other centers.

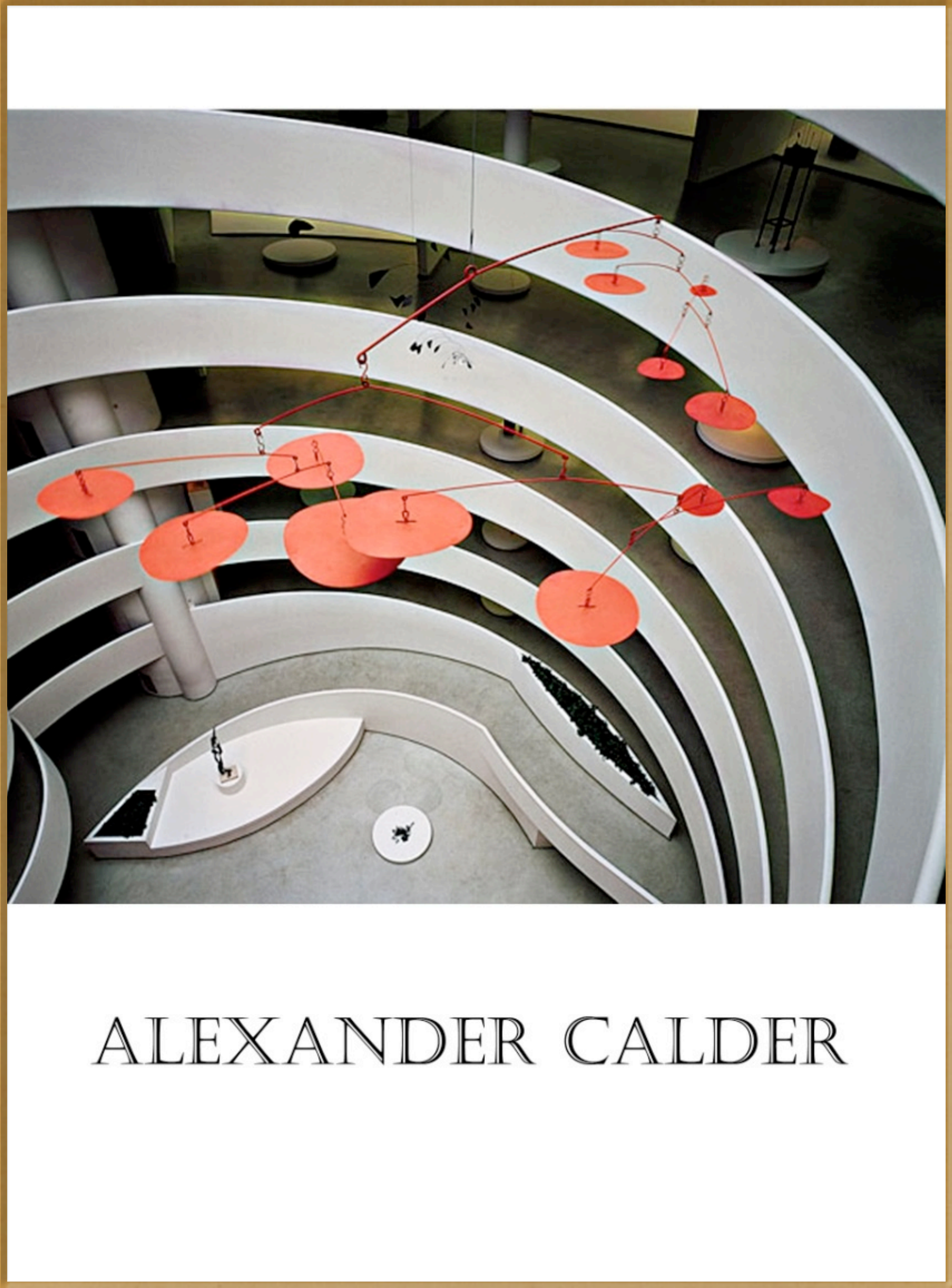
Good design offers areas of focus or weight.

2. Cohesion: System components are *cohesive* when the well-defined design choices they embody reinforce their contribution to the system as a whole. Thus, the concerns central to each component are clear, and distinct from the components that surround it.

Mapped to Relational

Each relation serves a separate role in the responsibility of representing domain knowledge. Relations enjoy identity as they distinctly capture and represent concepts in the form of facts collected to represent cogent, clearly defined information. The tuples within relations similarly represent cogent, unambiguously defined instances of reality patterned after the attribute structure of their containing relation while by virtue of their entity integrity they remain distinct from any other tuple therein. The population of tuples in a relation over time reflect the ebb and flow of experience that the relation captures in the dynamics of the represented reality (the extension). The attribute structure of the relation as a template for each of its tuples ensures that the experience remains comparable and thus understandable regardless of the number of instances that experience produces. The result is a collection of distinct knowledge experiences bound together by a structure that both explains the significance of each instance and enables the analysis of that experience in terms of the whole reality that the relation captures.

Center Properties



Boundaries

The field-like effect is strengthened by the creation of a ring-like center.

Outlines focus attention on the center.

3. Encapsulation: A system module is properly *encapsulated* when its separateness is balanced by a straightforward and intelligible description of "what" (defined by its interface) that module does to cooperate with the collective around it.

Mapped to Relational

In the relational paradigm the individual relation assumes the responsibility for capturing and defining the "reality," the "facts," the modeler chooses to instill in a model. The modeler's intension is represented in the structure of facts that each of its instances must be able remember. Each instance of the relation remembers by way of the data attribute value set in each tuple. The truthfulness of individual tuples can thus be independently established. An important part of the reality captured in each tuple is its individuality and the uniqueness of the information that it remembers in its data attribute values, its entity integrity. This individuality is determined solely by the values contained therein dependent on no other information or relationships; as characterized by Second Normal Form.

Alternating Repetition

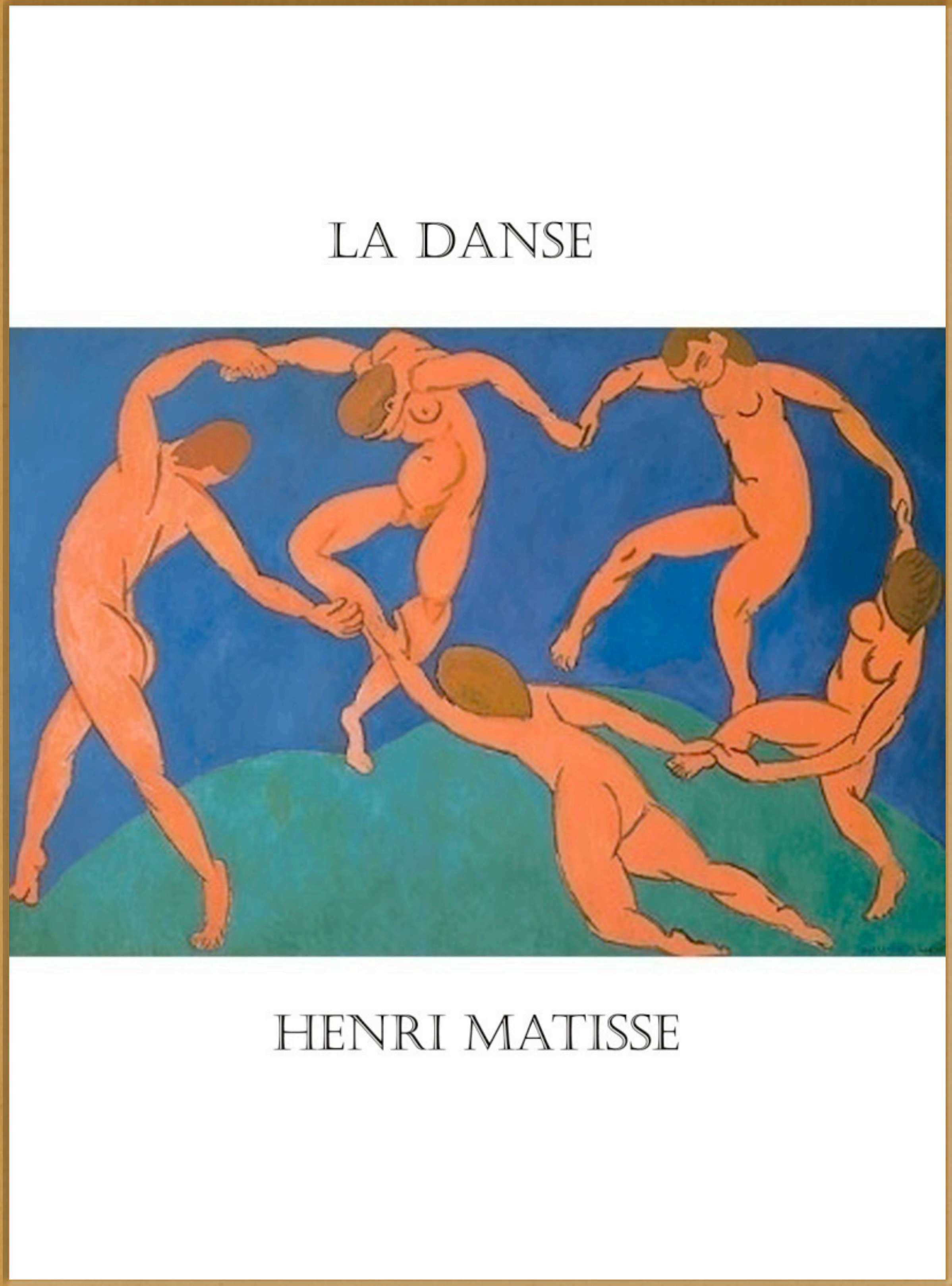
Centers are strengthened when they repeat, by the insertion of other centers between them.

Repeating various elements creates a sense of order and harmony.

4. Extensibility: Modules that are conceived to be reused and re-tasked after they have been implemented are said to be *extensible*. Such modules offer the potential for the system's function to be expanded even after the modules have been crafted. Furthermore, the participation of individual modules in different arrangements of cooperation reuses them to achieve more than a single purpose within the system as a whole.

Mapped to Relational

Although each relation (down to the individual tuple) represents an independent depiction of reality in a relational model, more complex information is possible through the relationships that associate relations. Associations permit the depiction of more elaborate descriptions of a model's responsibilities. Associations depict correspondence, interdependence or even ownership of concepts between and among relations. These associations are employed through the relational operators that combine or collect facts resident in multiple relations and render them correlated, organized and/or extracted as a consistent but new representation of knowledge contained in the model.



Positive Space

A center should draw strength from the centers immediately adjacent.

Background should reinforce rather than detract from the center.

5. Modularization: A system is appropriately *modularized* when its subsystems are crafted to always work in combination with other subsystems to achieve their collective purpose. Thus, each module depends at the outset of its conception on the subordinate cooperation of its neighbors. Furthermore, a module's primary function may be to organize or coordinate the contribution of the subordinates to a purpose for which individually they may be ignorant; reflecting a separation of concerns.

Mapped to Relational

By the nature of depicting model knowledge in a collection of individual relations that knowledge is subdivided and compartmentalized. Furthermore the process of normalization assures that the intension depicted by individual relations and combinations of relations through their associations is neither ambiguous, redundant nor inconsistent. The cohesion that distinguishes each relation's role in the intension of the model also segregates the concerns that accomplish the model's responsibilities and permit attention to be focused on relevant subsets of the overall model's complexity.

Good Shape

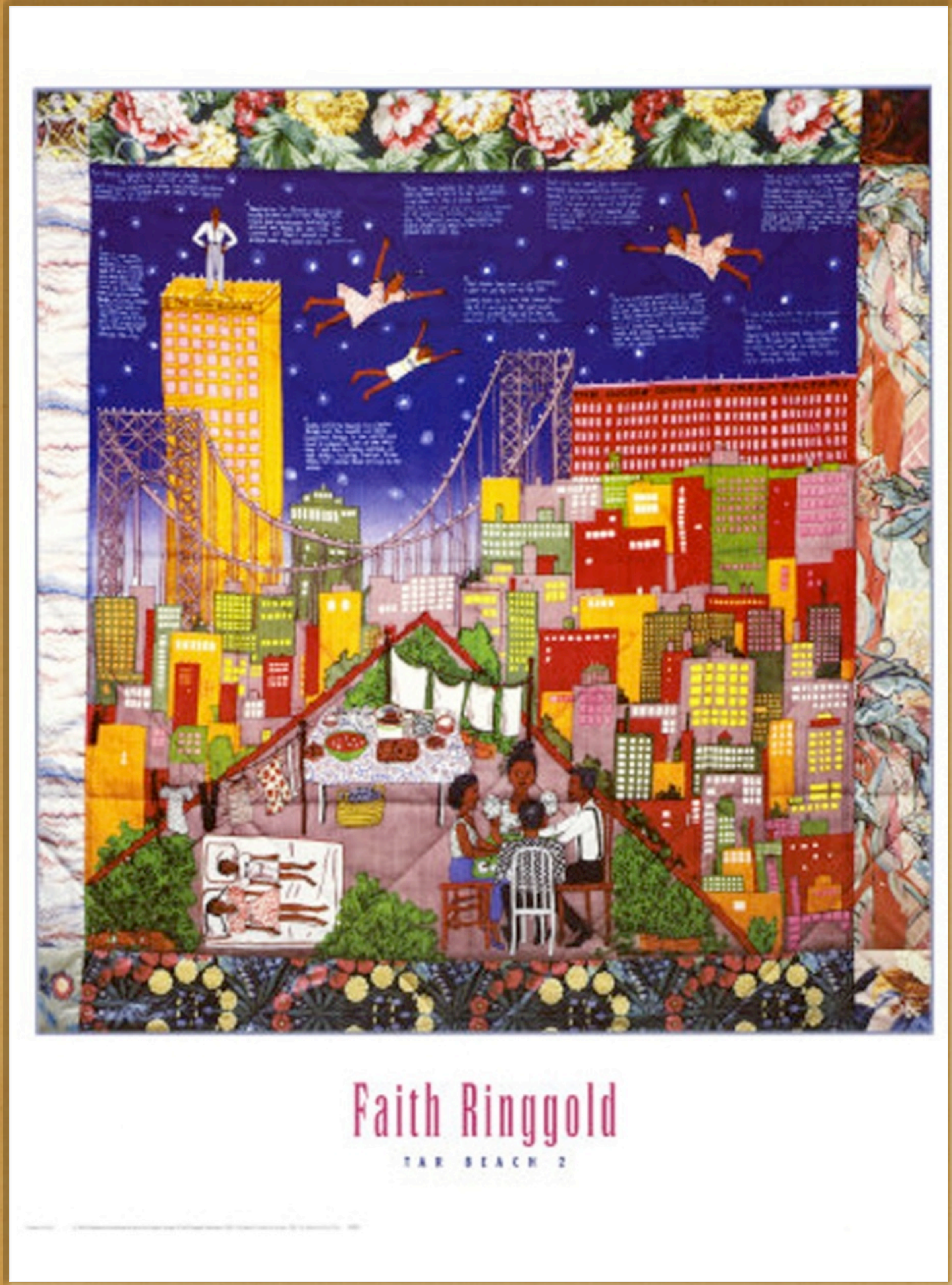
The strength of a given center depends on its actual shape. And the shape, its boundaries and the space around it must be made up of strong centers.

Simple forms create an intense, powerful center.

6. Correctness: *Correctness* is the presence of germane and essential system behaviors as specified by the requirements combined with the absence of extraneous behaviors. Although correctness does not presume to guarantee perfection, it denies design choices that compromise the quality of operation of the system.

Mapped to Relational

Entity integrity, referential integrity and normalization directly support a relational model's fidelity to the modeler's intension. Entity integrity assures that the uniqueness of each depiction of reality (extension) is enforced by the structure of the relation, intension, (the attribute set, their respective data attribute domains and the respective functional dependencies). The specification of that subset of attributes that will always contain a unique (combination of) value(s) defines the discriminating characteristics of that knowledge (the primary key) – the conformance to which is easily tested and thus protected. Referential integrity assures not only that data attribute values conform to the intension of their relation's data attribute domain but further, to the modeled intension of associations between tuples including the ownership relationship between relations. Normalization extends the assurance of fidelity (model to the modeler's intension) by assuring that the interrelationship among data attribute values not only supports entity integrity and referential integrity but, also inhibits the accidental loss of model knowledge (anomalies) through the action of relational operators.



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.

7. Transparency: *Transparency* is being able to observe discernible structure in a system, or how things fit and work together. During system design the purpose and composition of design choices must enjoy transparency, exposing the "patterns" and "weave" of their interconnectedness. Transparency reveals simple symmetry like record or file structures, which regularize the collection and organization of information. Transparency can also reveal more sophisticated mechanisms (e.g. inheritance and polymorphism), which express symmetries that span the definition and evolution of families of structure.

Mapped to Relational

The relational paradigm facilitates transparency in two obvious respects. Inspecting the relevant data attribute values is sufficient to assess every aspect of integrity whether entity integrity or referential integrity. These same continuously accessible values form the basis of all relationships among data attribute values or among relations. The consistency of each and every data attribute value can be certified. At any time before or after any and every relational database operation we can verify concurrence with the time independent definition of intension given by the data attribute set and their respective data attribute domains along with the designation of candidate and foreign keys. There are no implied or hidden definitions of association or dependence. Every aspect of tuple or relation fidelity is discerned through self-evident information. The result of any relational operator is determined solely by the data attribute values of the relations involved.

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.

8. Composition of Function: A non-trivial system (or subsystem) requires a *composition of functions* that are combined in a manner that effectively implements its required functionality. In such systems, components that interoperate with super-ordinate components to support a combined purpose tend to recede into the “shadows” as they perform their role largely anonymously. The combination of these components forms new choices of function or behavior that subsume their individual identities.

Mapped to Relational

Each relation in a relational model represents a fundamental aspect of intension in the modeler’s depiction of reality. Association and the use of relational operators effect that fundamental intension deriving an answer to any query we may invent based on that fundamental knowledge. The result of every relational operation is itself a relation. The modeler’s ingenuity and discipline in forming queries carefully that yield results, relations, that are themselves consistent with the integrity constraints of the model creates the potential of an endless cascade of query result as input to another query and so on. This is the direct result of the mathematical formalism upon which the relation model is based –the predominating strength of the relational paradigm. The form in which these queries may be posed to a relational system is constrained only by the choice of mathematical representations (e.g. tuple calculus or domain calculus) or transformations (e.g. relational algebra or relational calculus) to the underlying relational definition.

Contrast

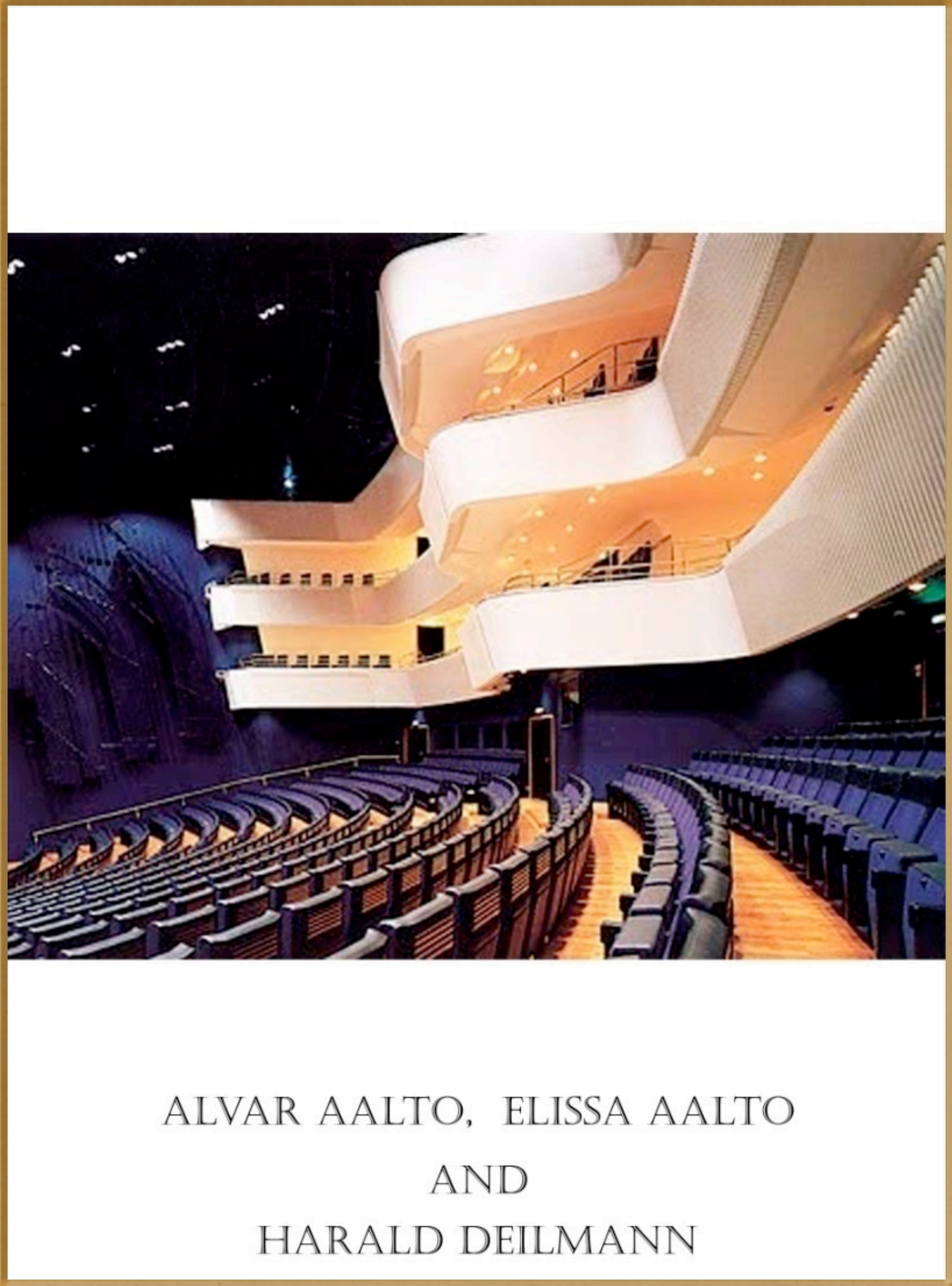
A center is strengthened by the sharpness of distinction between itself and the surrounding centers.

Unity is achieved with visible opposites.

9. Identity: *Identity* is the clarity of distinctiveness between modules in a system. Poor choices in the identity of modules in a system often gives rise to system components that address the same purpose, and thus cause confusion within the design of the system as a whole.

Mapped to Relational

Identity is at the root of recognition. In the physical world identity is literal based upon direct sensorimotor experience: by sight or touch and in some cases by sound or smell – a human experience of the “real” world. In the relational paradigm this human experience is applied directly by collecting those attributes that completely describe how any particular instance is unique – the combination of attributes that comprise the primary key. The primary key serves to anchor the knowledge that surrounds it – those additional attributes that further describe the tuple which it uniquely determines –those attribute values that are functionally dependent upon the primary key. No tuple is permitted to exist in the relational universe (extension) unless it has a primary key – entity integrity. Ownership as it is manifest through foreign key associations is anchored on the primary key of the owner tuple.



Gradients

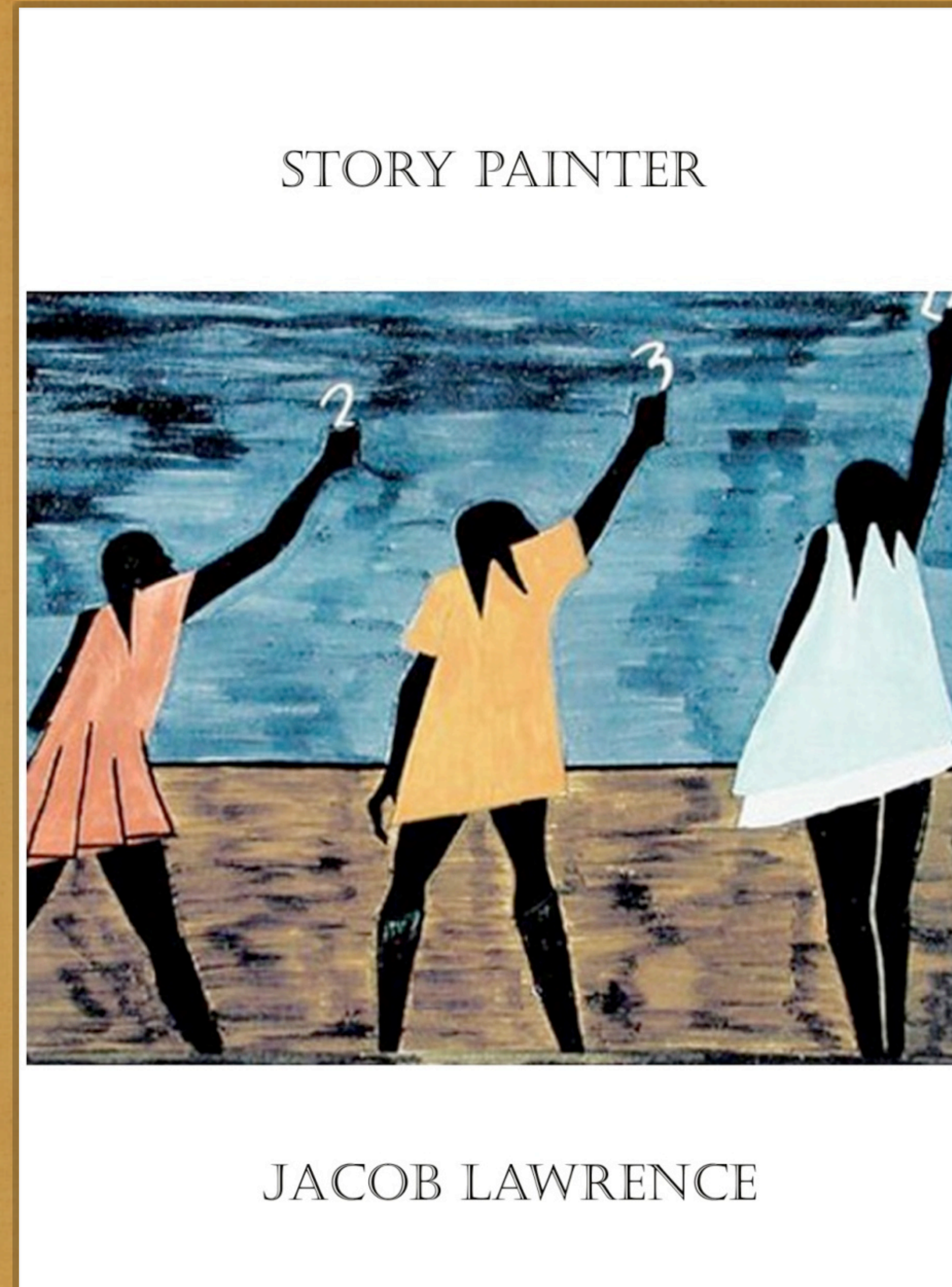
A center is strengthened by a graded series of different sized centers which then point to a new center.

The proportional use of space and pattern creates harmony.

10. Scale: *Scale* is the elaboration of system detail appropriate to the needs of particular observers. Using scale to describe different layers of a system is an important tool in complexity management in analysis, in design, in implementation and in documentation of complex systems.

Mapped to Relational

In many cases the only familiarity that is needed in a relational model is the intension – the collection of relations with their attribute sets and the respective associations. The knowledge structure and semantic relationships that may be mined through relational operators sufficiently defines any derivation of information representations that queries may be formulated to elicit. In terms of scale any relational model (intension or extension) may be expanded to incorporate additional knowledge. The modeler achieves this by grafting new knowledge onto existing relation structure through the alignment of data attribute domains and associations. A fluid unfolding of model scale is a challenge particularly if the structure underlying the knowledge segments to be coupled do not share the same modeler's intension.



Roughness

The way a center draws its strength from irregularities in sizes, shapes and arrangements.

Texture and imperfections convey uniqueness and life.

11. User Friendliness: Different users may have widely varying needs for system interaction and patterns thereof. *User friendliness* is achieved when the system is matched to the expectations of its users. Furthermore, the range and granularity of interface options should reflect the nature of the needs of the users to use the system in accomplishing their individual tasks.

Mapped to Relational

There is elegance in the succinctness and simplicity that arises from properly isolating domain knowledge in the respective relations. The use of user/client/customer familiar naming of relations and attributes and the choice of the commonly used, domain based attribute values lends a comfort level to the representation of problem domain experience. The relational model also enables the derivation of contained knowledge at levels of granularity much higher than the individual tuple or relation. This is because relational operations on relations produce relations as their result. Information derived from a relational database can be presented as if it were simply retrieved from a single physical relation. This illusion is easily achieved in relational programming languages that support the definition and storage of queries that may then be referenced themselves as relations without the users' notice (i.e. in ANSI SQL the "create view" syntax.). The facility of such extensions to apply relational operations so discretely creates virtually unlimited opportunities and permits what might otherwise be a complex and daunting algorithm of derivation to be completely ignored by the users.

Echoes

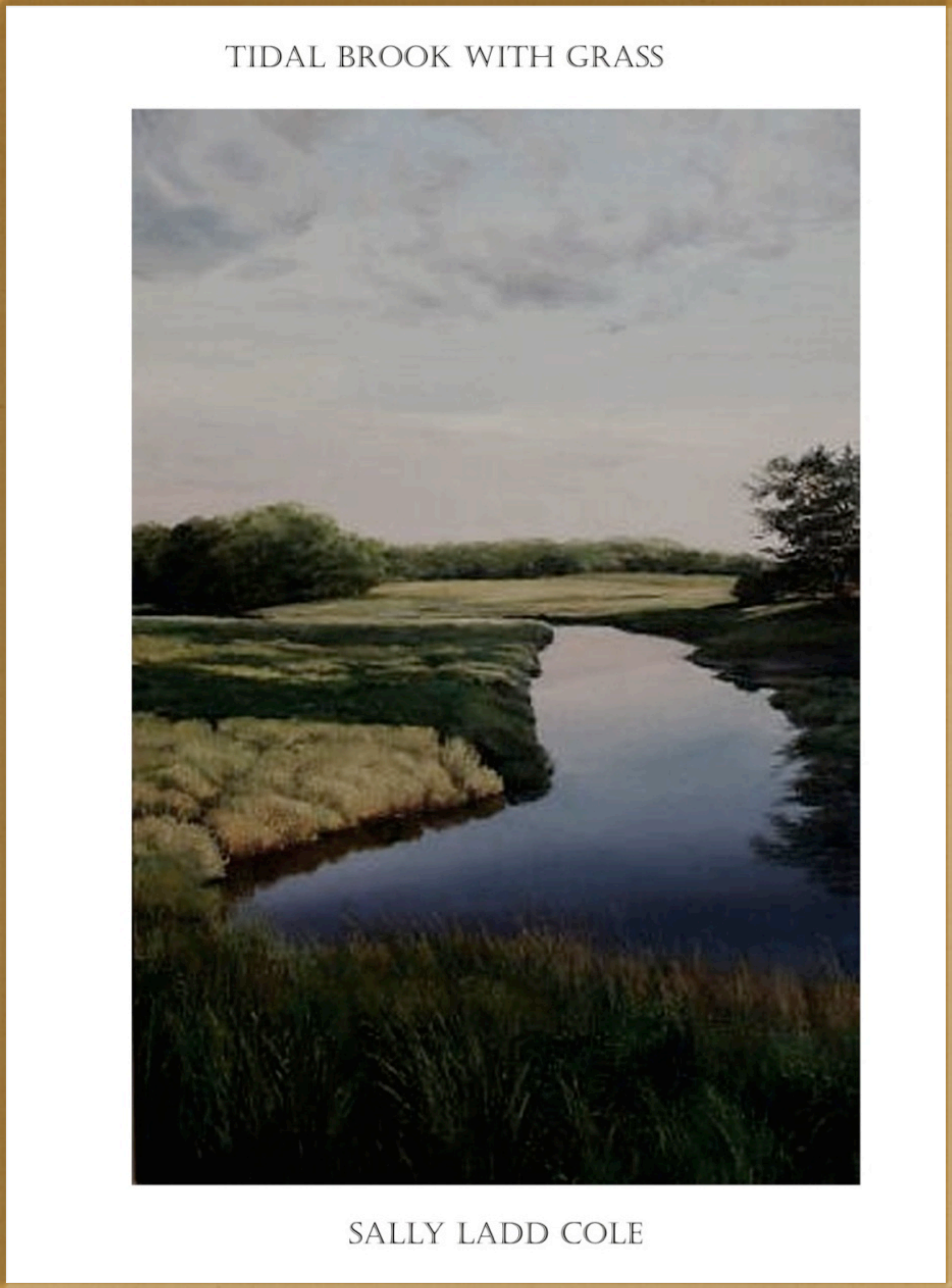
The strength of a given center depends on similarities of angle and orientation.

Similarities should repeat themselves throughout a design.

12. Patterns: *Patterns* in a system result from symmetry of purpose. Similarities and parallels that reside in purpose should be reflected explicitly in interface patterns. Patterns are often described in standards, guidelines and frameworks that help define the system. In addition, patterns of purpose can be found in collections of design choices.

Mapped to Relational

The most predominant pattern found in relational models is the regularity of structure that is embodied in the tuple that populates relations. This regularity assures that the same "questions" may be posed to each and every instance in a relation to elicit the same meaningful result. The tuples may be readily compared one to another and ordered that their factual content may be exhibited in a useful unfolding of multiplicity. At the next level of structure we find the foreign key relationship where an association between relations is constructed by choosing attributes in the two relations that proceed from the same attribute domain. The pattern is further emphasized by the property of referential integrity. This pattern of connecting facts between and among relations permits the stepwise assemblage of higher and higher levels of derived information. The association enables the traversal of a network of concepts and facts that are both defined by and operationally enabled by the foreign key construct. The use of these patterns by the relational model designer provides the opportunity to lay out domain knowledge in a predictable and usable mapping.



The Void

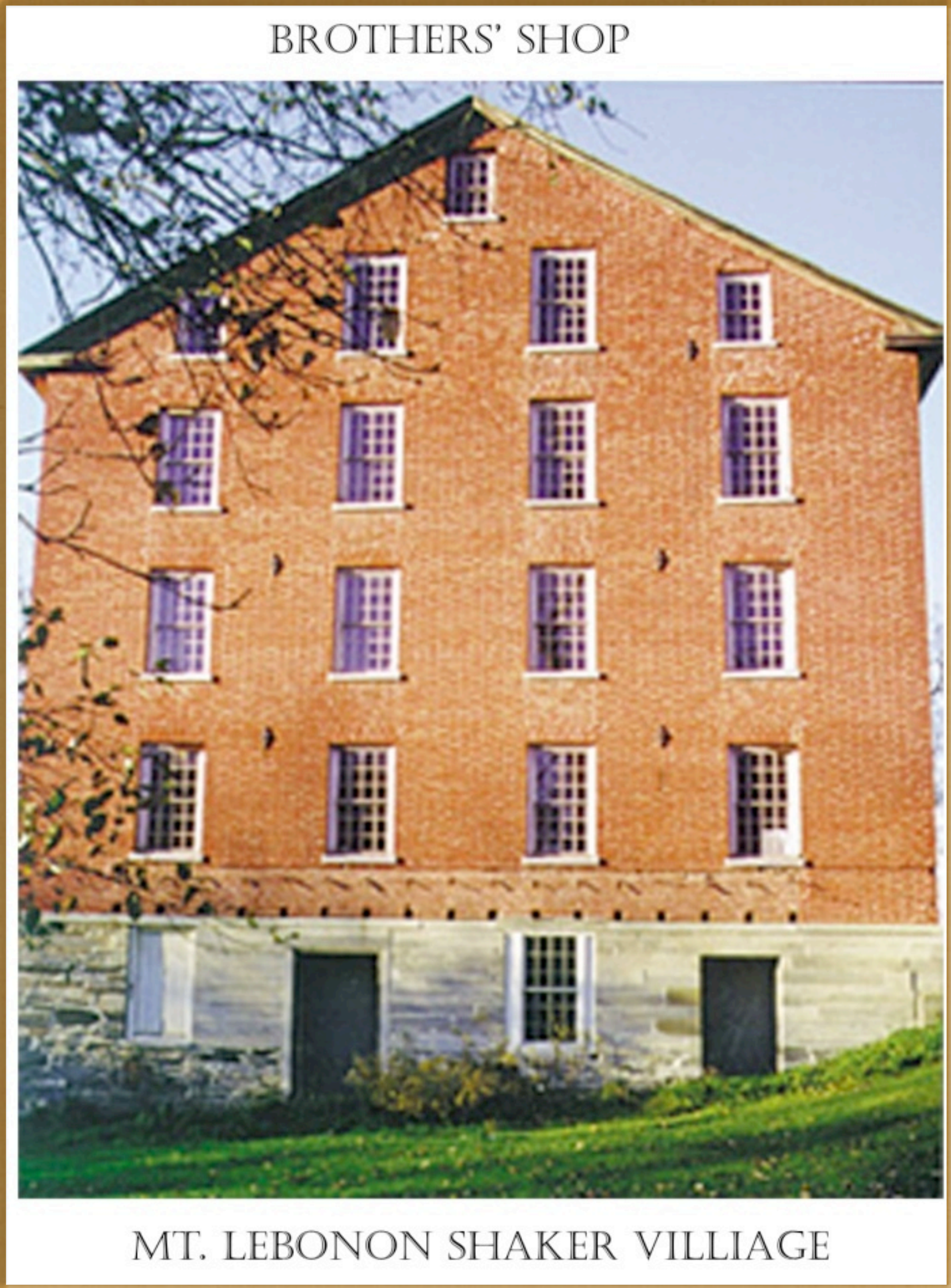
The intensity of every center depends on the existence of a still place – an empty center.

Empty spaces offer calm and contrast.

13. Programmability: Rather than being targeted to a single, narrow question or purpose, a *programmable* system provides its user with the means to dynamically re-target the system over time. Developers often make choices that are specifically intended to support a range of purpose achieved primarily by aggregating various collections rather than multiplying choices. This might be described as “component based design.”

Mapped to Relational

Returning again to the use of relational operations to compose higher and higher levels of information we see individual relations as building blocks that may be arranged (assembled through relational operations) to yield any reasonable arrangement or derivation of information that the underlying relations may possess. This is possible because of the individual identity that each relation fosters in its tuples and because of the predictable reliability that proceeds from the consistency and safety of relational operations that is guaranteed in a set of normalized relations. The extent of information mining that may be attempted is limited almost solely by the programmers’ imagination.



Simplicity and Inner Calm

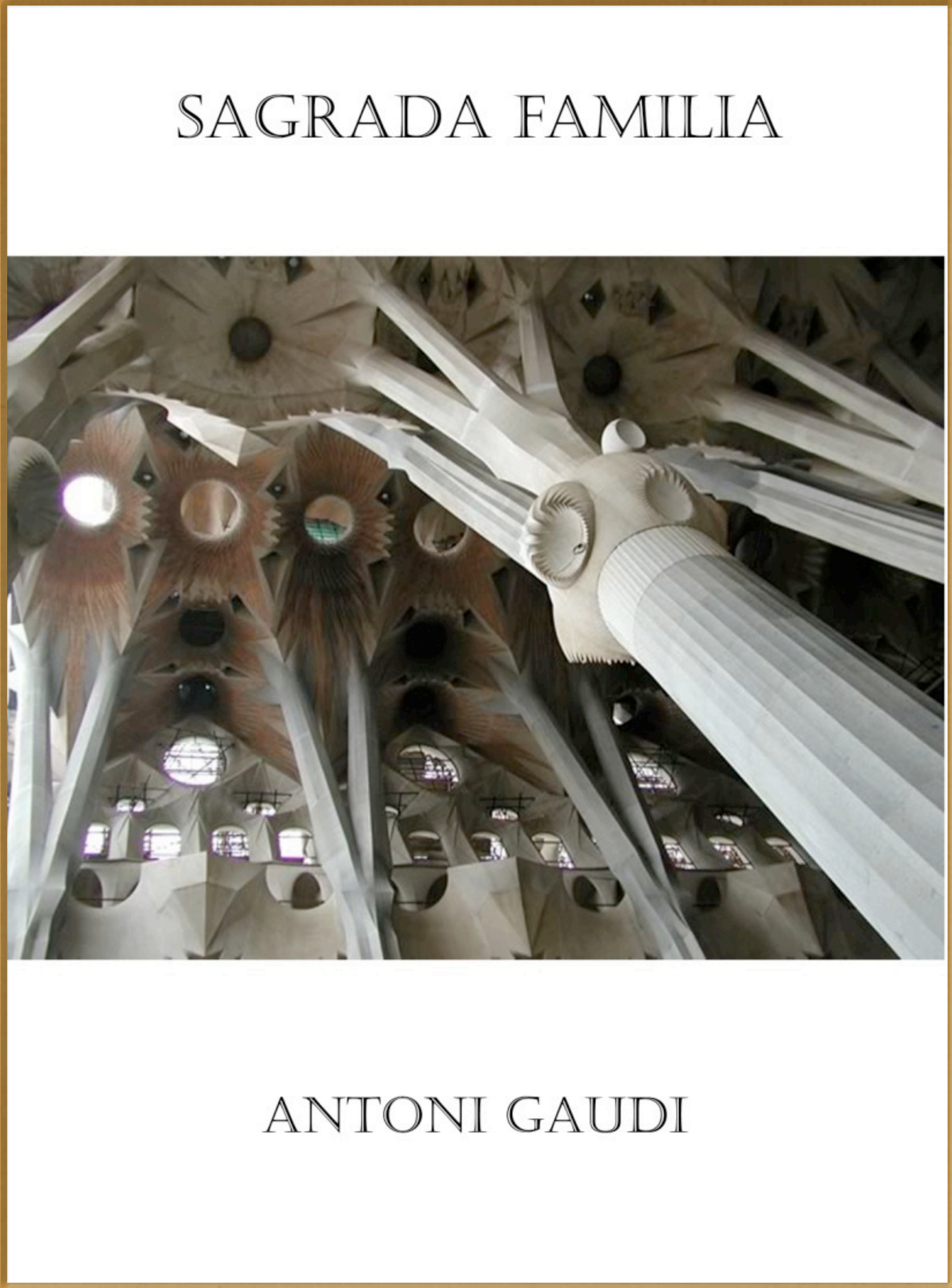
The strength of a center depends on its simplicity.

Use only essentials and avoid extraneous elements.

14. Reliability: *Reliable* systems function as designed without interruption. Another goal in achieving reliability is to eschew extraneous detail is to avoid unwanted or unexpected side effects that may reveal themselves and cause confusion. Such extraneous detail or embellishment can also lead to unwanted and unnecessary system maintenance.

Mapped to Relational

There is an overarching simplicity that results from the fact that all of the properties of integrity are based upon data attribute values that may be readily inspected before or after any relational operation. Intension is expressed in modeled expressions of integrity constraints that are domain specific. The synchronization between the intension and extension of the model is easily tested because of this simple transparency. Reliability is assured if valid relational operations are applied consistent with model integrity constraints and thus will always yield consistent (“truthful”) information.



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.

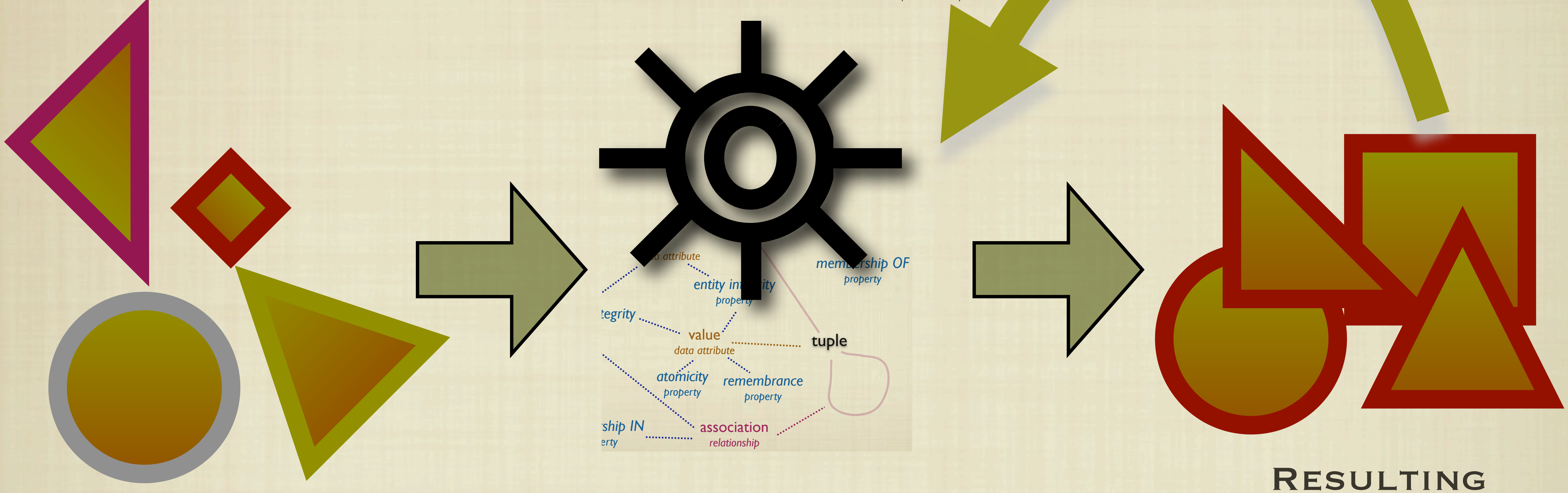
15. Elegance: System models that are consistent, clear, concise, coherent, cogent and transparently correct exude *elegance*. As each design choice is added into, deleted from, or modified in the collection its influence must be revisited, reconsidered, to assess the resulting affect on the system as a whole. Thus, elegance is an increase in wholeness that progresses toward effectiveness and efficiency.

Mapped to Relational

Elegance is achieved largely through the relational model when relations are modeled with a minimum of extraneous or redundant information. Indeed eliminating redundancy is common mantra of relational modeling. The laying out of basic facts divided into distinct encapsulated containers of knowledge and the subsequent composition of higher levels of derived information effects a sense of economy of form and abundant opportunity for exploring and extracting the knowledge a database so fashioned accommodates.

The Design Process

Choice Property transformations(15)

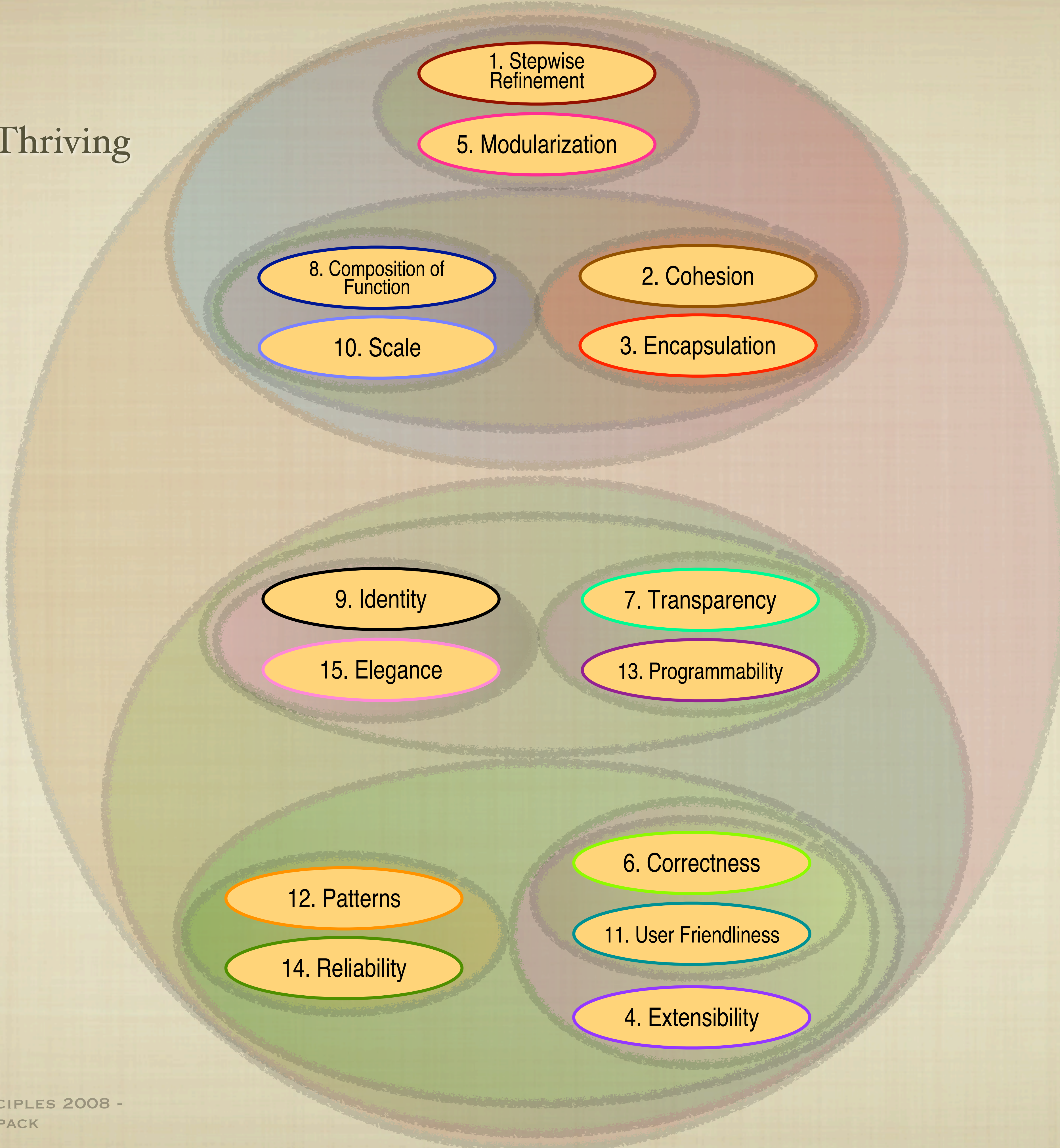


Stakeholder intensions
requirement elements
model elements
design elements

Representational
paradigm or Ontology

Resulting
Design
Elements

Toward an architectural philosophy of Thriving Systems



Teaching a Sense of Great Design

- "Perceive the wholeness and the impact of individual design decisions 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." (Waguespack)
- Realign the modeling focus. Focus on why to use the tools – not on the tools themselves. Redirect decision-making energy to the question, "How does each decision increase the life in the system by fulfilling the stakeholders' evolving concerns?" and "What does life mean to these stakeholders?"


WHEN I STARTED PROGRAMMING, WE DIDN'T HAVE ANY OF THESE SISSY "ICONS" AND "WINDOWS."
© 1992 United Feature Syndicate, Inc.
S.Adams
9-8
ALL WE HAD WERE ZEROS AND ONES -- AND SOMETIMES WE DIDN'T EVEN HAVE ONES.
I WROTE AN ENTIRE DATABASE PROGRAM USING ONLY ZEROS.
YOU HAD ZEROS? WE HAD TO USE THE LETTER "O."