Painting Systems: From Art to Systems Architecting

Alejandro Salado
Dept of Industrial & Systems Engineering, Virginia Tech
217 Durham Hall, 1145 Perry St, Blacksburg VA 24060 (USA)
+1-(540)231-0483
asalado@vt.edu

Luca Iandoli
Dept of Industrial Engineering, University of Naples Federico II (Italy)
School of Systems and enterprises, Stevens Institute of Technology (USA)
Piazzale Tecchio 80, 80125 Napoli (Italy)
+39-0817682935
luca.iandoli@unina.it

Giuseppe Zollo
Dept of Industrial Engineering, University of Naples Federico II (Italy)
Piazzale Tecchio 80, 80125 Napoli (Italy)
+39-0817682483
giuzollo@unina.it

Abstract. Systems engineering has successfully contributed to develop complex systems in industries such as defense, space, energy, or transportation. Systems in those industries have in common their engineered or socio-technical natures. However, systems engineering has been understood for a long time as both an art and a science. As a matter of fact, principles and practices of systems engineering have been exhibited in the creation of film original score, a major artistic endeavor. Therefore, it is sensible to investigate if and how the similarities and differences between the application of systems engineering in the development of engineered systems and the pursue of artistic endeavors can help in improving the understanding and practice in both domains. Specifically, this paper discusses how the same concepts are applied in artistic painting and in system architecture to reduce complexity and achieve elegance.

Introduction

Systems engineering is both an art and a science (Ryschkewitsch et al. 2009; Jansma, 2012). Its scientific side deals with producing actual designs; its artistic one drives the value of the system architecture (Muirhead and Thomas, 2010). Strong ability in both sides are exhibited in effective systems engineers, who possess a good balance of hard- and soft-skills (Ryschkewitsch et al. 2009; Jansma, 2012). However, most of the development and training of systems engineers has focused for a long time in the analytical and procedural side. As a result, there is a growing demand for promoting the artistic side and pursing elegance when architecting engineering systems (Griffin, 2010; Madni, 2012; Salado and Nilchiani, 2013a, 2013b).

So why *art* specifically? Research in neuroscience has made substantial progress in the last decade in improving our understanding about how our brain produces knowledge by extracting meaningful patterns from the flow of data that hit our senses. The complexity of the brain has evolved in time through the addition of newer neural layers over the older ones to provide additional and more sophisticated cognitive capabilities. In this way the more recent conscious and rational layers have been integrated with the older subconscious and emotional strata. Thus, human brain's ability to extract meaning from the sensory world is not only built on the analytical skills of the higher level layers, but also on heuristics that are affected by a number of "irrational" and emotional biases (Damasio, 1994; Kahneman and Twersky, 1982). By observing that such a mix of analytical and emotional thinking is particularly intense in the way we perceive art, several neuroscientists have turned their attention to the way human beings perceive art.

The idea behind this research is based on the observation that artists deliberately alter their perceptions to the light of their inner world and express the result of this synthesis in an artifact (e.g. painting, sculpture, etc.) that viewers can experience both emotionally and "rationally". In particular, according to some neuroscientists (Ramachandran, 2011; Zeki, 2009), the aesthetic enjoyment we feel in front of a masterpiece is an exceptional empirical situation to study a key human ability, i.e. how we spot meaningful patterns to reduce the complexity determined by ambiguous information. According to these studies understanding a painting is the result of the application of specific cognitive strategies and of a system of biological incentives that reward our ability to recognize meaningful patterns through a change in the activation of the brain's reward system (Salah and Salah, 2008).

Thus, both the aesthetic experience **and systems architecting seem to have in common the objective of reducing complexity.** Systems engineering is claimed to be valuable in developing complex system because of its inherent capability to reduce complexity (Haskins, 2010). System architecting enables structuring a problem in manageable pieces (Maier and Rechtin, 2009). One of the authors of this paper discussed in a previous paper how actual principles and practices of systems engineering are exhibited in the composition of a film original score, which helped in reducing the complexity of such artistic endeavor (Salado and Salado, 2015). As discussed in the previous paragraph, neuroscientists are interested in finding out how the irrational side of our brain reduces complexity to identify unconscious patterns in data in artistic pieces. Therefore, we claim that if evolution has hardwired in human brain cognitive abilities to recognize beauty, the aesthetic value of a representation must also matter for other practical purposes including the augmentation of our ability to understand and manage complexity and thus to design better course of actions to harness it. In order to explore this idea, we present how principles and practices that have been used for centuries in artistic painting are actually exhibited by *good* architectures and *elegant* systems.

Paper structure. The paper is organized as follows. First, a summary about heuristics, taxonomies, and paradigms to architect good or elegant systems is given. Then, we present a number of painting techniques that have been widely used to pursue beauty. Following that, we show some examples extracted from actual engineered systems that exhibit how applying those techniques, without being aware of them, resulted in improving their system architectures. Finally, we present some conclusions and ideas for future research.

The Art of Systems Architecting

Systems architecting has been described as an art by several authors in order to emphasize the differences in boundary conditions and solution approaches between architecting and engineering. Table 1 provides a comparison, adapted from (Maier and Rechtin, 2009).

Because problems are ill-structured at the time of architecting, it is hard to evaluate the *goodness* of an architecture. Some authors have recently employed the term *elegance*. Essentially, *goodness* or *elegance* refers to an attribute or combination of attributes that enables comparing two architectures in relative terms when their value in terms of performance is similar. However, even though most engineers would perceive the elegance of a system architecture once they see it, defining and measuring such an attribute is an elusive feat (Griffin, 2010). This matches the findings from neuroscientists presented in the introduction in this paper, with regards to the combination of emotional and rational brain activities when decoding information from an artistic piece.

Table 1: Architecting vs Engineering (adapted from (Maier and Rechtin, 2009))

Criterion	**Architecting**	**Intersection architecting & engineering**	**Engineering**
Problem	Ill-structured	Constrained	Understood
Goal	Satisfaction	Compliance	Optimization
Approach	Heuristics	Both	Equations
Techniques	Synthesis	Both	Analysis

Attempting to shine light into understanding what an elegant architecture is, some authors have identified attributes that elegant architectures have in common. In its most abstracted understanding, elegant architectures seem to exhibit effectiveness, efficiency, robustness, and minimum of unintended consequences (Griffin, 2010). Such conditions may be attained if an architecture (1) has a clear purpose, (2) is parsimonious, transparent, scalable, sustainable, usable, and predictable, (3) provides utility while generating bonding, (4) does all of that efficiently and affordably, and (5) can be evolved to satisfy emergent needs (Madni, 2012). Those attributes have been structured and organized in order to focus on the abstract concepts that yield elegance instead of limiting its understanding to specific attributes, which enables evolving the definition of elegance as new attributes or insights are identified (Salado and Nilchiani, 2013b). Essentially, as depicted in Figure 1, an elegant system is understood to (1) do what it is supposed to do and do it well, (2) do it when you need it to, (3) do it efficiently, and (4) adapt to what it may be needed to do (Salado and Nilchiani, 2013b).

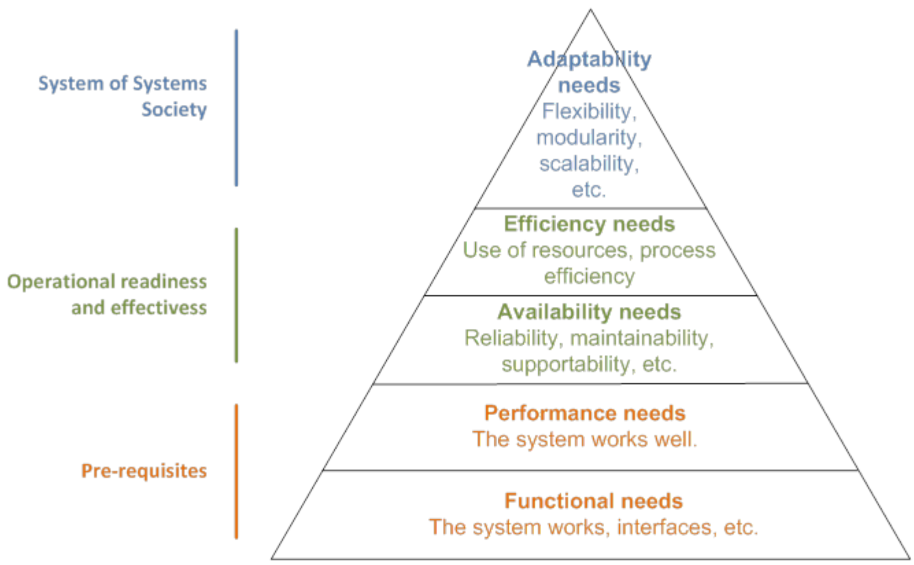


Figure 1. Elegant systems (Salado and Nilchiani, 2013b)

As introduced previously in Table 1, reflecting on experiences gathered during decades has led to identifying a number of heuristics that, when used adequately, usually improve the *goodness* of an architecture. A comprehensive set of them that addresses the architecture and its development is provided in (Maier and Rechtin, 2009); some of them are shown in Table 2.

Table 2: Heuristics to develop good architectures (adapted from (Maier and Rechtin, 2009))

Heuristics
Design the structure with good bones. The eye is a fine architect. Believe it. A good solution somehow looks nice.
Simplify. Simplify. Simplify. Simplify, combine, and eliminate. Don't make an architecture too smart for its own good. The most reliable part … is the one that isn't there – because it isn't needed.
Aggregate around testable units; partition around logical subassemblies. Never aggregate systems that have a conflict. System structures should resemble functional structure. Group elements that are strongly related; separate elements that are unrelated. Poor aggregation results in grey boundaries and red performance. Don't slice through regions where high rates of information exchange are required. Choose minimal communication between subsystems. Choose low external complexity and high internal complexity.
Complex systems will … evolve … if there are stable intermediate forms. Build in and maintain options in the design.

Complexity Reduction in Art

Aesthetic experience can be understood by the theoretical construct of Effective Complexity (Murray Gell-Mann, 1994), in which complexity is an intermediate property between perfect uniformity and total randomness as measured through a subjective interpretation of interest to the observer, that is, a model (Funes, 1996). In this perspective complexity is needed to find meaning beyond obvious predictability; besides, complexity resides in the relationship between the observer and the observed rather than being an objective property of systems or phenomena. This idea is echoed in in semiotics and art critique, where it is stated that "Art is without noise: art is a system which is pure, no unit ever goes wasted" (Roland Barthes, 1983, p. 261). In other words, in the best art there is no room for minimizing distraction because everything matters and each bit of information is as necessary as any other in the representation.

How do masters achieve effective complexity? Apparently artists resort, deliberately or unconsciously, to a set of universal rules that have been likely selected by evolution and which constitute the basic pattern recognition ability of the human brain (Ramachandran, 2011). As the Gestalt psychologists did at the beginning of the past century, some authors in the field of art also claim that human experience of reality cannot be reduced to the perception of elementary parts, but it's in fact active search of structures that support the understanding of reality as a whole. This is also in line with the principles of General Systems Theory (von

Bertalanffy, 1969), which informs the application of modern systems thinking (Boardman and Sauser, 2008) and systems engineering (Hitchins, 2007).

In this paper we argue that artists use two sets of strategies to achieve effective complexity: Noise-Killing strategies (NK) and Meaning-Adding strategies (MA). If one relies on NK strategies alone the experience is oversimplified and its interpretation becomes dull and trivial; conversely, too intense use of MA strategies makes a representation overwhelming and unnecessary complex.

We have identified four strategies for the NK type and four for the MA one using as our main references the laws of organized perception proposed by Gestalt psychologists (Wertheimer, 1923; Koffka, 2013) and by recent works in neuroaesthetics (Ramachandran, 2011). In table 3 we map our proposed strategies over the laws of Gestalt and Ramachandran's laws of artistic experience. We refer the reader to the above works to get detailed descriptions of Ramachandran and Gestalt laws while we describe below the proposed noise-killing and add-meaning strategies. It is important to specify that some of the Gestalt and neuroaesthetics laws have a significant but partial overlap with our definitions; besides, we have deliberately emphasized the architectural implications of the laws of perception in order to make them more applicable to systems engineering.

Table 3: NK &MA strategies

Gestalt laws	**Ramachandran law of artistic experience**	**NK & MA strategies**
Simplicity, Closure		Subtract details
Symmetry	Symmetry	Symmetry
Similarity, Continuity, Proximity	Grouping	List, group
	Generic Viewpoint	Split
Common fate	Peak shift	Emphasize
Past experience	Perceptual problem solving, visual metaphors	Remix & reconnect
	Isolation	Power of the center
Common fate, Proximity	Contrast	Contrast & balance

Strategy 1. Subtract details to get the whole. (NK) The fundamental idea is that artists suggest rather than describe reality (Zeki, 2009). They do so by reducing a complex system to a reduced set of much simpler entities that are remixed in different ways. Good examples are the seemingly coarse brushstrokes of Monet, Picasso's cubist landscapes (Fig. 2a), and the evanescent atmosphere in Turner's works.

Strategy 2. Use symmetry to structure experience. (NK) Symmetry, in art as well as in nature, allows our brain to generate complex shapes in a more efficient way. Symmetry helps to identify hierarchy, focal points, distances, directions and the visual weights of various elements as related to some axis or direction. In middle age and early Renaissance paintings with traditional religious subject symmetry is typically used to remind order, hierarchy, and centrality of the holy subject or event (e.g. as in crucifixion, ref. Fig. 2b).

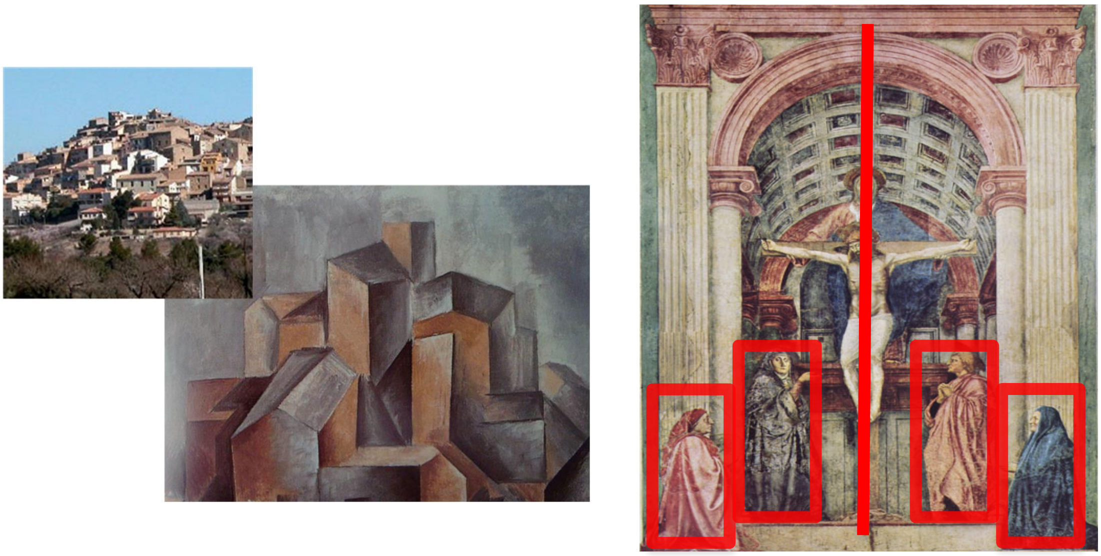


Figure 2. Example of Strategy 1 – Subtracting (a) & Strategy 2 – Symmetries (b)

Strategy 3. Use lists and groups. (NK) As discovered by Gestalt psychology, groups provide order by similarity, vicinity, closure, and segregation or by suggesting common destiny (ref. Fig. 3a).

Strategy 4. Split information at different levels. (NK) Complex information needs to be displayed at different levels of detail. For instance, contrasting foreground with background helps to focus on a particular object, but also puts the object into a context. Encapsulating details into a larger structure helps to guide the eye from the general to the particular and vice versa, as in the Little Street by Vermeer, in which the structure is defined by the regularity of the building façade and more detailed scenes of everyday life are encapsulated into one door opening in a room at the ground level and a little alley leading into what is presumably the backyard of the building (ref. Fig. 3b).

Figure 3. Example of Strategy 3 – Grouping (a) & Strategy 4 - Splitting

Strategy 5. Emphasize the differences over the averages. (MA) This strategy consists in the deliberate attempt to stress meaningful details or features, e.g. by rejecting realistic proportions as in the elongated necks of the ladies portrayed by Modigliani or as in the violation of the laws of statics in Botticelli's Venus to suggest that the goddess is in fact levitating instead of falling (ref. Fig. 4a).

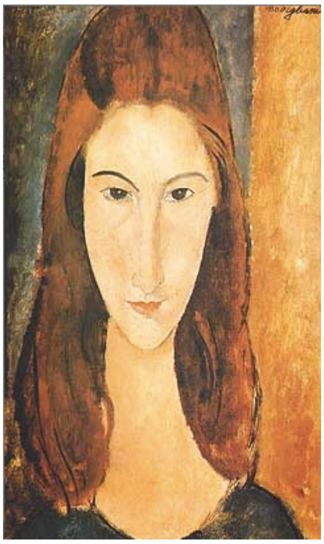


Figure 4. Example of Strategy 5 – Emphasizing differences (a) & Strategy 6 – Remix and reconnect

Strategy 6. Remix and reconnect. (MA) This strategy implies the disassembling and creative reconnection of visual elements into new combinations to deliver non obvious meaning as in Brueghel's Fall of the Angels where the infernal little figures are obtained through the grotesque, sadistic, and chaotic dismembering of bodies and their reassembling into monstrous creatures to suggest that evil is just loss of divine order (ref. Fig. 4b).

Strategy 7. Exploit the power of center. (MA) This strategy is about the creation of one dominant visual center or a few related ones by using colors and lines to guide the eye of the observer toward a particular object from which the attention is radiated back to the rest of the painting, as it happens in Van Gogh's Starry Night in which the sky vortexes help our gaze to meander across the beauty and the anguish of the night sky (ref. Fig. 5a).

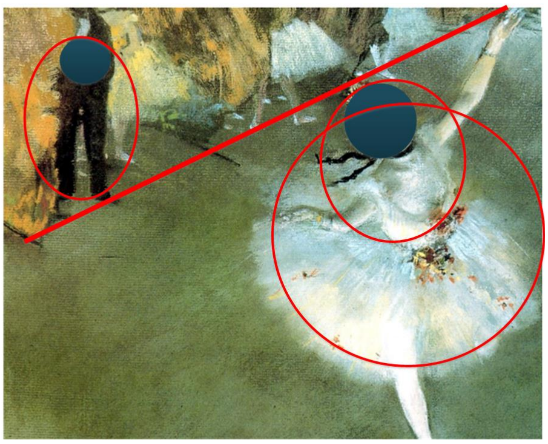


Figure 5. Example of Strategy 7 – Centring (a) & Strategy 8 – Contrast and balance (b)

Strategy 8. Contrast and balance. (MA) This strategy aims at creating tension between elements that are supposed to be in some sort of relationship using color or composition.

Examples are the use of dark and bright tones to separate lights form shadow, the juxtaposition of complementary colors, the spatial arrangement of shapes and elements around some direction to balance their visual weight. Degas' ballerinas are a beautiful example of how balance is achieved by trying to capture the harmonious tension in young dancers' moves (ref. Fig. 5b).

Artistic Principles Exhibited in Elegant Systems

In this section, we describe how the strategies presented in the previous section were intuitively employed by engineers (i.e. they were not aware of the specific techniques) in actual industrial projects to improve their system architectures. It should be noted that some aspects have been sanitized for confidentiality reasons. Furthermore, some examples actually exhibit several strategies at once. This is natural since actually *elegance* or *goodness* is achieved by combining some of the strategies as needed. For explanatory purposes though, we have tried to emphasize that one that is described in each strategy.

Strategy 1. Subtract details to get the whole. Figure 6 depicts three stages in the development of a system architecture using a N2 diagram where the dots represent interfaces, in a space optical instrument. In the initial architecture, depicted in (a), it was decided to break the system in 9 components in order to reduce contractual burden. This yielded an unexpected high external complexity, leading to multiple contractual changes resulting from the complexity in managing the interfaces, as well as an anticipated high complexity in the integration of the components. At a given point in the system development, the system was re-architected to reduce the external complexity of the system, thus clustering the components with high interaction, as shown in (b) and (c). Essentially, this subtraction of detail is usually employed in system architecting under the concepts of cohesion and complexity hiding.

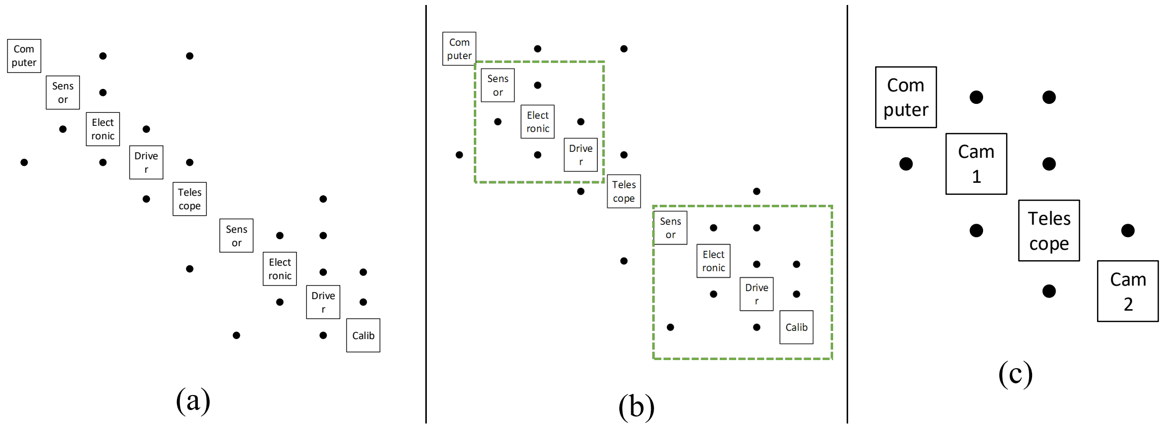


Figure 6. Example of Strategy 1 in Systems Engineering – Subtracting

Strategy 2. Use symmetry to structure experience. Figure 7 depicts three stages in the development of a system architecture for a satellite payload, where the boxes represent physical components and the lines communication between them. In this case, the connection marked with the red cross in (a) imposed severe difficulties for system integration and testing due to scalability. Furthermore, such a connection was critical for the performance of the component to the right of the cross (when looking at the picture). Scalability was pursued by improving the symmetry of the system architecture, as shown in (c). This was achieved by using Strategy 1, i.e. subtracting the detailed connection by combining both components into a single one and thus hiding it from the system, as shown in (b).

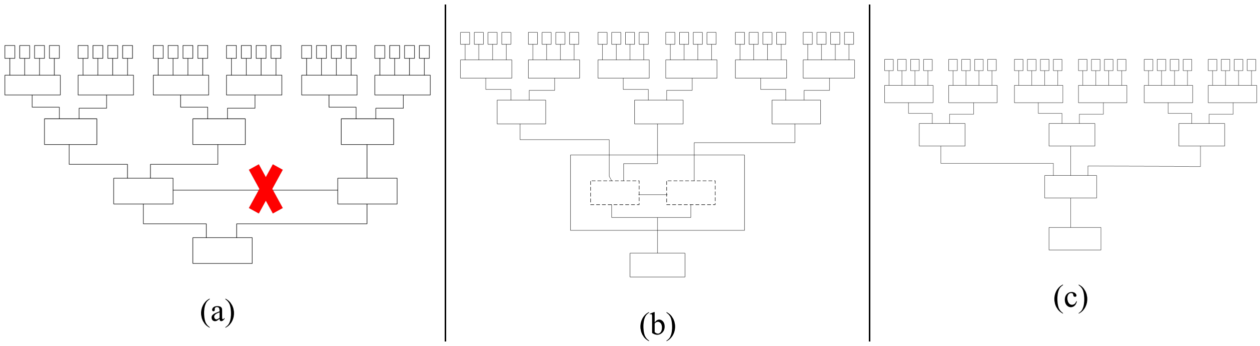


Figure 7. Example of Strategy 2 in Systems Engineering – Symmetries

Strategy 3. Use lists and groups. Figure 8 depicts the contractual architectures for two different satellite instruments. Each color represents one company. In (a) it was decided to allocate components to companies based on the nature of the component. As shown in the example, in this case all structural support parts were contracted to the same company directly by the payload contractor. This led to significant technical and contractual problems resulting from fuzzy responsibilities in the interfaces. In contrast, contracts in (b) were grouped according to the interfaces set forth by the physical architecture. In this case, ownerships of the contracts were transferred to each level in the architecture decomposition. Even if eventually the same company would provide different components, they would do so under different contracts. This resulted in better contractual and technical management due to better grouped interfaces. In systems engineering this is sometimes referred to as *clustering*.

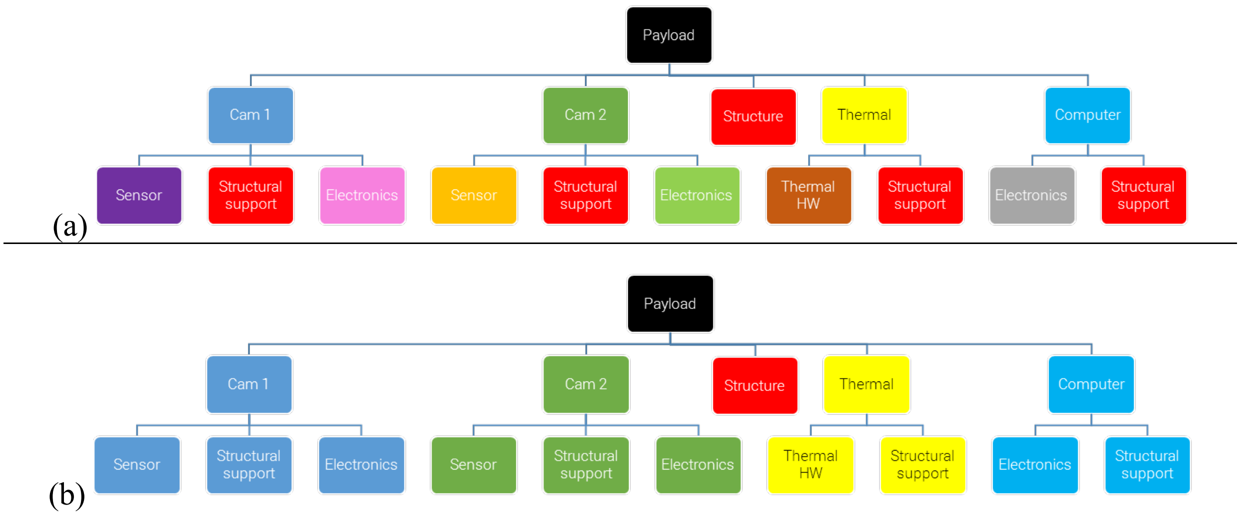


Figure 8. Example of Strategy 3 in Systems Engineering – Grouping

Strategy 4. Split information at different levels. Splitting lays at the core of systems engineering and architecting. It is understood as iteratively breaking down the problem in smaller parts that can be managed more easily. In systems engineering jargon, this is usually called *partitioning*. Using Figure 7 as an example, it would consist in doing the opposite effort. That is, starting with a first level architecture decomposition as depicted in (c), details are then elaborated on each one of those components, as shown in (b).

Strategy 5. Emphasize the differences over the averages. Figure 9 shows two architectures for a notional system. The green components, or parts of components, represent non-classified elements. The red components, or parts of components, represent elements that are classified.

In architecture (a) all components interface with a component that is classified. This imposes significant constraints in the development, integration, and test of the entire system. In contrast, in architecture (b) the uniqueness of the classified part has been emphasized and put aside from the system, limiting in this way its interface with just one component in the system. This enables developing the biggest part of the system without the stringent constraints of classified developments, limiting those efforts to a narrow, properly bounded part of the system. This example is also valid when considering the red part to be, for example, a technology with low readiness level or a technology with unique or demanding integration or testing constraints, among others.

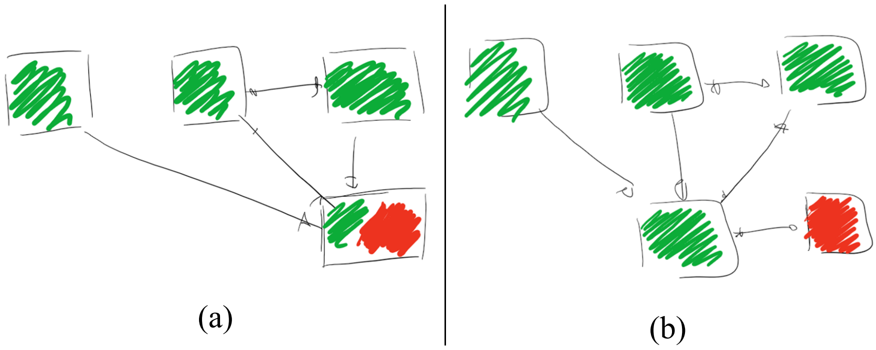


Figure 9. Example of Strategy 5 in Systems Engineering – Emphasize differences

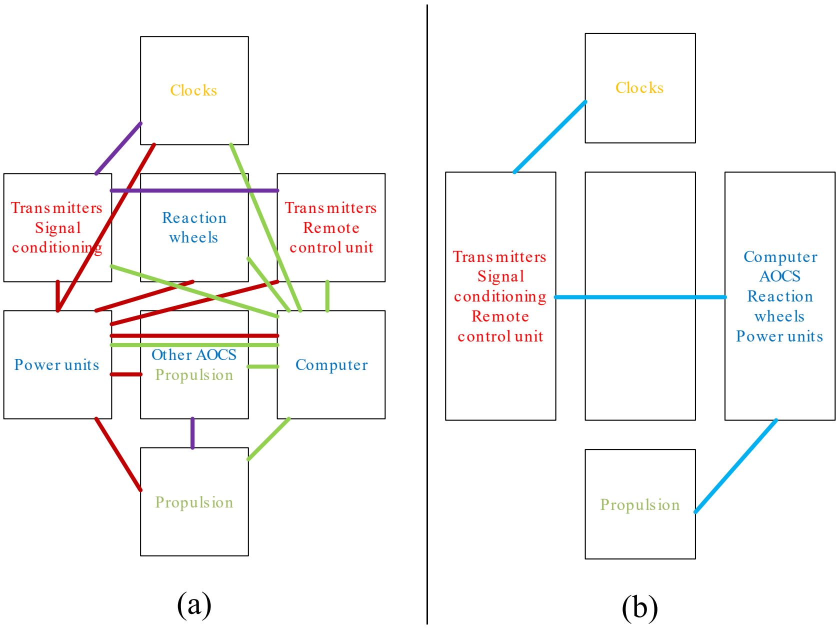


Figure 10. Example of Strategy 6 in Systems Engineering – Reconnect

Strategy 6. Remix and reconnect. Figure 10 depicts the physical architecture and design of two families of satellites. In (a), some technical constraints led to break up the system in multiple pieces. This resulted in severe integration constraints due to the high coupling between the various physical parts. In (b) changes needed to be done in order to satisfy new programmatic constraints, yet maintaining as much as possible the elements of the design in (a). Reconnecting the parts differently, without changing their design, resulted in a grouping that significantly reduced their coupling and enabled easier integration.

Strategy 7. Exploit the power of center. Figure 11 depicts two pairs of architectures. The first pair is represented by N2 diagrams. The architecture in (a) contains several interfaces between the various components. It is well known that this pattern increases the complexity of the system and the difficulty of its integration and testing, and usually reduces the adaptability and evolvability of the system. This results from the high coupling between all components. Using a command and control alternative, for example, as shown in (b), in which a centered component manages the communications, reduces such coupling. This pattern is known for its easier integration and testing, as well as its higher capability to be evolved. This does not contradict the benefits of networked systems, as shown when comparing (c) and (d). The diagram in (c) is another representation of a system that follows the pattern in (a). In this case, communication between all systems is achieved by point to point communication links. It has therefore the same problems as identified previously. In (d), however, a communication bus takes the center role. This provides at least similar benefits than the pattern in (b).

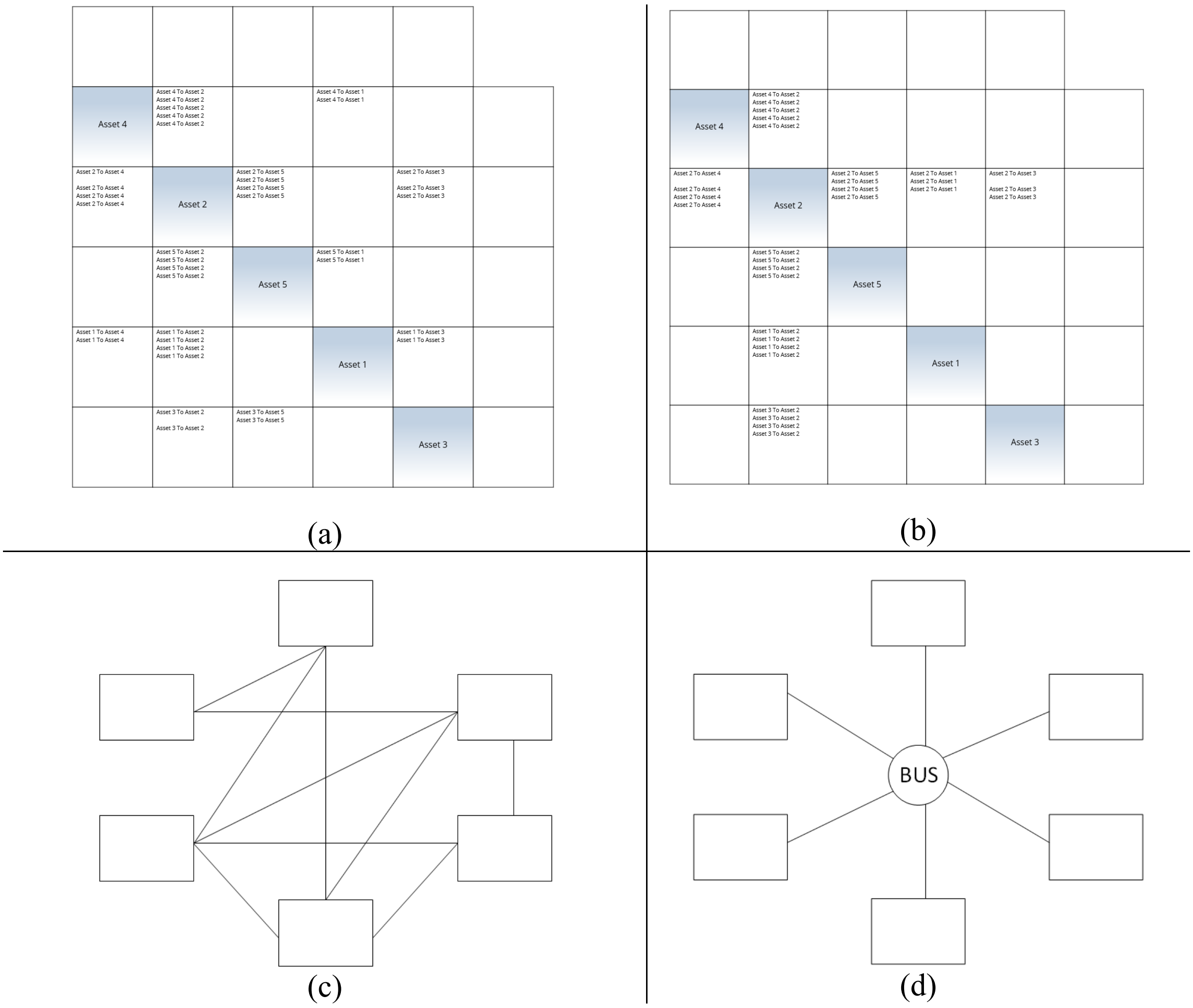


Figure 11. Example of Strategy 7 in Systems Engineering – Power of center

Strategy 8. Contrast and balance. Figure 10 exemplifies how balancing can help in reducing system complexity. The grouping in (b) balances the various functionalities to provide contrast between the various modules. As can be seen, payload units (written in red) are identified in one module, while platform ones (written in blue) are in a separate one.

Conclusions and Future Research

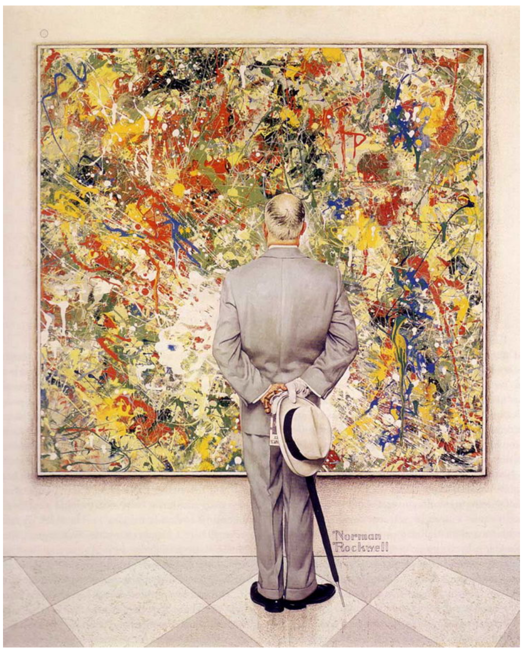
This paper has shown that systems engineers intuitively use similar complexity reduction techniques when architecting a system than those artists employ when creating artistic paintings. This further strengthens the notion that system architecting, or systems engineering more generally, is both an art and a science.

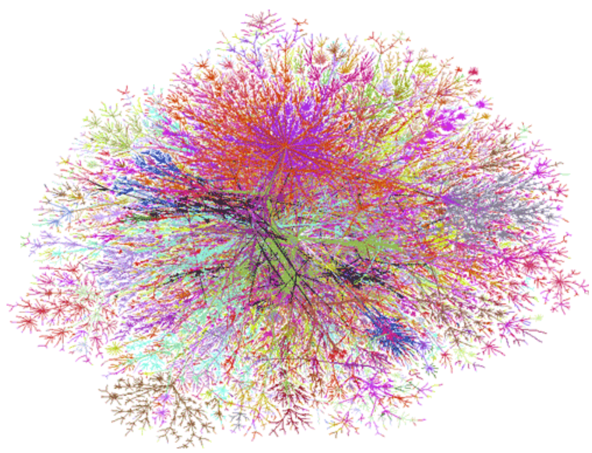
In particular, eight strategies have been presented, described with actual artistic pieces, and shown as exhibited in examples taken from actual industrial projects. We emphasize that the techniques are not necessarily used individually, but in most cases several techniques are used simultaneously. We also want to state that the ultimate objective of architecting is satisfying a given set of needs. The techniques presented in this paper should be vehicles to achieve that objective, but not be the objective themselves.

A topic for future research that remains open at the moment is how to foster the necessary creativity or skill to actually improve the way these techniques are used in an effective way. That is, even though several artists employ the presented techniques, some are better than the others. Therefore, is it possible to find a way to educate systems engineers to be better architects?

In addition, we also suggest to continue research in joining the domains of art and systems engineering. Although the integration with product designers is more and more common, we look deeper, to the underlying architecture that is not seeable by product users.

Finally, we anticipate interest in exploring if contemporary, abstract art can be employed to improve the understanding and practice of complex networked systems and data visualization. We leave Figure 12 below as inspiration for such thought.





(b)

(a)

Figure 12. Abstract art (a) and networked systems (b)

References

Barthes, R. 1983. *A Barthes reader*, Hill and Wang, edited by S. Sontag.

Boardman, J. and Sauser, B. 2008. *Systems Thinking: Coping with 21st Century Problems*. Boca Raton: CRC Press.

Damasio, A. 1994. *Descartes' Error: Emotion, Reason, and the Human Brain*, Avon Publishing.

Funes, P. 1996. *Complexity measures for complex systems and complex objects*, http://www.cs.brandeis.edu/~pablo/complex.maker.html#REF29981

Griffin, M. D. 2010. "How Do We Fix Systems Engineering?" *61st International Astronautical Congress*, Prague, Czeck Republic.

Haskins, C., ed. 2010. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities.* Version 3.2. Revised by M. Krueger, D. Walden, and R. D. Hamelin. San Diego, CA (US): INCOSE.

Hitchins, D. K. 2007. *Systems Engineering: A 21st Century Systems Methodology*. Chichester, UK: John Wiley & Sons, Ltd.

Jansma, P. A. 2012. "Exploring the Art and Science of Systems Engineering." *IEEE Aerospace Conference*, Big Sky, MT.

Kahneman, D. and Tversky, A. 1982. *Judgment Under Uncertainty: Heuristics and Biases*, Cambridge University Press.

Koffka, K. 2013. *Principles of Gestalt psychology*. Routledge.

Madni, A. M. 2012. "Elegant Systems Design: Creative Fusion of Simplicity and Power." Syst. Eng. 15(3): 347-354.

Maier, M. W. and Rechtin, E. 2009. *The art of systems architecting*. 3rd edition, CRC Press, Boca Raton, Fl, 2009.

Muirhead, B. K. and Thomas, D. 2010. "The Art and Science of Systems Engineering Tightly Coupled Programs." *SAE Int. J. Passeng. Cars. Electron. Electr. Syst.* 3(2):117-130.

Ramachandran, V.S. 2011. T*he Tell-Tale Brain: A Neuroscientist's Quest for What Makes Us Human*, W. W. Norton & Company.

Ryschkewitsch, M., Schaible, D., and Larson, W. 2009. "The Art and Science of Systems Engineering." *Syst. Res. Forum* 03, 81.

Salado, A. and Nilchiani, R. 2013a. "Elegant Space Systems: How Do We Get There?" *IEEE Aerospace Conference*, Big Sky, MT.

------. 2013b. "Using Maslow's Hierarchy of Needs to Define Elegance in System Architecture." *Procedia Computer Science* 16: 923-936.

Salado, A. and Salado, C. 2015. "Systems Engineering Practices Exhibited in the Creation of a Film Original Score." *INCOSE International Symposium*, 25: 1147-1158.

Salah, A.A.A.; Salah, A.A. 2008. "Technoscience art: A bridge between neuroesthetics and art history?". *Review of General Psychology* 12 (2): 147–158.

Wertheimer, M. 1923. *A source book in Gestalt Psychology*. London: Routledge & Kegan.

Zeki, S. 2008. *Splendors and Miseries of the Brain: Love, Creativity, and the Quest for Human Happiness*, Wiley-Balckwell.

Von Bertallanffy, L. 1969. *General Systems Theory – Foundations, development, applications*. New York, NY, USA: George Braziller, Inc.

Biography

Dr. Alejandro Salado is an assistant professor of systems science and systems engineering with the Grado Department of Industrial & Systems Engineering at Virginia Tech. He holds master degrees in electrical engineering, electronics engineering, space systems engineering, and project management. He received his PhD in systems engineering from the School of Systems and Enterprises at Stevens Institute of Technology. His research focuses on unveiling the scientific foundations of systems engineering and using them to improve systems engineering practice. He is also pioneering bridging the domains of art and systems engineering. Before joining academia, Alejandro spent more than 10 years developing space systems, holding the roles of chief architect and chief systems engineer in his last projects. Dr. Salado is a recipient of the Fulbright International Science and Technology Award and received the Best Academic Paper Award of the 2015 Conference on Systems Engineering Research (CSER).

Dr. Luca Iandoli is an Associate Professor of Engineering Management, School of Engineering, University of Naples Federico II, a Visiting Research Professor at the School of Systems and Enterprises, Stevens Institute of Technology (USA), and former Fulbright Visiting Scholar at the Center for Collective Intelligence of the Massachusetts Institute of Technology. He has published numerous papers on the analysis of collaborative dynamics in firms, networks and industrial districts through the use of computational methodologies (agent- based modeling, fuzzy logic, social network analysis). Dr. Iandoli's current research focuses on online collaboration, information diffusion and collective intelligence in online communities, and innovation in distributed systems. He is President elect of the International Council for Small Business and Entrepreneurship.

Dr. Giuseppe Zollo is a full professor of business economics and organization in the Department of Industrial Engineering, University of Naples Federico II. In 1985-1986, he was a visiting research associate in the Department of Economics of Northeastern University, Boston. He has published in several journals and has presented papers at international conferences on innovation management, organization, small innovative firms, and managerial application of fuzzy logic. In 1992, 1993, and 1995, he received the Entrepreneurship Award at the RENT Research in Entrepreneurship Workshops organized by the European Institute for Advanced Studies in Management; in 1994, he received the Best Paper Award from FGF, Universitat Dortmund at the IntEnt 94 Conference. Moreover, he is a member of several editorial boards of international journals. He is vice president of the International Association for Fuzzy-Set Management and Economy (SIGEF).