White Paper

The Art of Judgment: Integrated Judgment

WP0117 | November 2013

Ceri Williams

Ceri has thirty years in the IT industry, originally delivering complex control systems and subsequently broadening focus to Enterprise Architecture, Governance and transformation of the IT function. Working as a chief architect, consultant and coach, he enables FTSE 250 organizations to make medium and long term decisions on the shape of the Enterprise Architecture and positioning of the IT function.

He advocates putting people at the heart of technology and business change with focus on the human enablers and constraints. His work deals with the way in which rigorous engineering and architecture disciplines are integrated with the cognitive and behavioural capabilities of the people who practice them.

The three previous papers in this series placed the ideas of Sir Geoffrey Vickers into the context of Enterprise Architecture. Vickers proposed that as part of an overall Appreciative System, there are three distinct types of judgment exercised:

1. Reality judgment: concerning what is or is not the case;
2. Value judgment: concerning what ought or ought not be;
3. Instrumental judgment: concerning the best means available to reduce the mismatch between is and ought.

Cutting across each of these, he proposes two further dimensions:

- Allocative Judgment: concerning the optimal allocation of scarce resources (e.g. money, attention, real-estate) between competing initiatives, aiming to deliver the greatest net benefit in the planning time frame
- Integrative Judgment: choosing between mutually exclusive alternatives (e.g. change units) or modifying them so that they integrate in a meaningful way.

Summarized in this way, Vickers' proposal might look over analytical as it uses a deconstructive approach to the problem space, giving rise to categories of 'thing'. This was not his intention, but a response of the need to break large, complex things into smaller pieces to facilitate understanding and the creation of value. That said, I believe that continuity and integration across these domains is a critical issue and one that warrants a set of specific responses.

In this paper, I focus on some techniques to ensure that Enterprise Architects and close stakeholders maintain continuity and coherence across these types of judgment. Continuity enables coherence as it operates across:

- **Space** – the domain of interest (or 'System of Interest' in engineering terms). This includes the external environment – the 'space' outside the System of Interest and how it impacts the System itself.
- **Process** – interoperability and co-ordination of information and action between types of judgment
- **People** – the individuals involved in all the judgment processes
- **Time** – understanding a situation, analyzing options and putting choices into effect take time. The passage of time enables shifts to occur in Space, Process, People and Values
- **Values** – the objectives and criteria by which reality is selected (reality is a choice), a future envisioned and actions to achieve it are committed to. This includes focus and priorities as recognition of constraints (e.g. resources, budgets, time) demands continuity of focus and relative priorities.

Continuity across Space

Goalposts shift, boundaries are re-drawn. These constraints work to disrupt continuity of judgment across the systems landscape. 'Space' in this case can be drawn along a number of lines described in Paper 1 – Reality Judgment. In particular the Breadth of coverage has a critical influence and should be held constant across judgment processes. Breadth covers features such as the architecture layer (e.g. Business, Application, Infrastructure) and the business system area (e.g. Finance, Service Delivery, Supply Chain).

Management of continuity here can draw on more mature disciplines of Change Management applied to the Enterprise Architecture artifacts, along with core concepts of base lining and version control. Alongside these techniques for administering change, they should also operate leadership and management disciplines to promote stability and continuity of the system boundary – building in some resistance to change and smoothing volatility. If Instrumental Judgment is being exercised on analysis performed on the output of Reality Judgment that covered a different set of systems, it is unlikely to come up with the most appropriate Transformation Plan. Stability of Space promotes integrity (literally 'hanging together') and therefore reliability of decision-making.

Dis-integration is a problem due to shifts in Space is easily avoided where there are obvious differences – such as trying to make decisions about Supply Chain systems when the as-is Architecture available only covers Product Data Management. However, the risks are less easily avoided with more subtle taxonomical shifts – such as making decisions on an Integration Broker based on as-is capture for one set of systems and using them as the default decisions for data integration for Business Intelligence purposes with another set of systems. Managing Space Creep ensure that decisions designed for one situation are not re-purposed for another without conscious thought and action.

One critical reason for managing continuity of Space is to ensure that system boundaries are drawn and hold fast along lines that prevent 'externalization' of costs and dis-benefits. Many flawed strategic decisions can be traced to intentional or accidental drawing of system boundaries that lower cost and deliver benefits in one area (e.g. standard user device provisioning) and the expense of additional costs and dis-benefits in another (e.g. critical user applications not available on the platform). This is why it is essential to ensure that people (e.g. users) as part of the Business Architecture are inside rather than outside the system of interest.

Continuity Between Processes

Most humans struggle with the idea of iterative processes, although they don't struggle with the practice of them – we operate them all the time unconsciously. Each of the key judgment processes –Reality, Value and Instrumental feed and feed off each-other. To do this they must share similar concepts, semantics and information in a knowledge pool. Each process has its own need for information and the form in which it is consumed. Each process refines and augments this information, generating 'instructions' for the other phases – if a Value judgment is required on the Information Systems landscape that is based on business drivers or architecture that is yet to be defined, elaborate assumptions will not be adequate, however well-guessed – someone will need to be tasked with defining them. *Figure 1* illustrates this process.

Iteration was originally a mathematical term, but in the design arena now describes a process of development or refinement whose outputs feedback in to itself – where the results of on iteration influence or direct the next. This is the case where problems are 'wicked' or constraints are only discovered through implementation of the process. Examples of this include: discovering during Reality

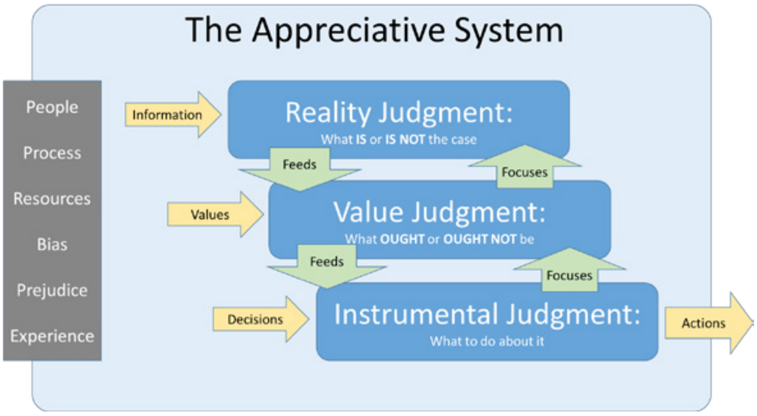

Figure 1: Iteration within the Appreciative System

Judgment that information is not available on current service levels of existing applications, and that creating the information is not feasible; or discovering that transition planning is not possible because the target architecture is defined at too high a level of abstraction. Feasibility is a regular driver of iteration: if the projected costs exceed outline budgets, then scope may need to be trimmed; if the transformation plan requires an excessively big bang with associated risks, then greater or different granularity may be needed in the Reality and Value judgment areas.

Knowing when to stop iterating is a decision point that most strategy and planning organizations struggle with. Continuity between processes also determines when enough work has been done in one area to feed another. For example, a key decision in the Reality process is the decision on when enough information has been captured; a key decision in the Value process is deciding when there is enough certainty to propose a future state.

The shared knowledge pool needs structuring in a way that provides internal continuity and supports judgment processes acting on a variety of timescales at a variety of levels of precision and abstraction. For readers familiar with IEEE 1471, the use of Projected views in preference to Constructed views promotes coherence, although needs significantly more disciplined governance to manage and are less adaptable and responsive.

Continuity of People

People have careers – they move on. People also change their minds, sometimes without realizing it. The average time in post for a recently appointed CIO is now expected to be a shade under 3 years. In addition to the CIO, Enterprise Architects will typically be working with multiple heads of business function or their deputies as well as key leaders in the IT function (e.g. Service Delivery, Development) and 3rd party suppliers. Churn across all these stakeholders significantly disrupts continuity across the judgment processes. Just to make this more challenging, the Enterprise Architect is typically engaged in facilitating and making medium to long term decisions whose initial delivery will outlast most individuals involved and whose costs and benefits will outlast them all (including themselves).

Many of the responses available to address this challenge lie in the areas that ensure continuity of process and values (see other sections in this Paper). In addition, countering the disruption of people churn depends on four things:

1. Repeatability of the process
2. Reliability of the process
3. Trust in the collective decision making of the people in the process
4. Enshrining of the commitment

Number 3 in particular demands a degree of transparency to ensure that decisions are made, and are seen to be made with good reason. Justice is done and is seen to be done – decisions are inclusive and seen to be inclusive. The mature Enterprise Architecture function and associated Stakeholders will declare their agendas so they are a visible and valuable part of the decision forming and making process. It will also ensure that the reasoning behind decisions are captured and associated with them in a transparent fashion. This requires a number of extensions to the usual EA meta-model to capture the rationale as an integral part of any future state definitions.

Collective decision-making is preferable here rather than Autocratic decision-making, as autocratic decision-making does not normally survive exit of the incumbent in the position (e.g. CIO). Long-term decisions that persist are usually made by groups of like-minded people, or groups whose members are prepared to compromise. Ensuring decisions are collectively made encourages objectivity through challenge and eases adoption. Decisions can be formally burned-in to a number of business commitments, including contracts with third parties, contracts of employment, ring-fenced budgets and incentive schemes, although 'making it stick' also depends on a different type of 'burn-in' – to the culture of the ogranization.

Continuity over Time

People forget, documents get lost. These constraints work to disrupt continuity of judgment over time. The instinctive response to this is to ensure that the output from each of the judgment processes is captured explicitly, often in some form of repository where tacit knowledge is turned into explicit knowledge. Some knowledge in this form remains stable over time (e.g. statements of principle, abstract design rules) while some degrades over time (e.g. technologically-specific statements of intent).

A complementary approach recognizes time constraints and applies the law of diminishing marginal returns – that the value of spending more time on a problem progressively diminishes. Timeboxing is as valuable a technique for long range strategic Architecture planning as it is for software delivery projects. At the same time, rushing to premature judgment risks forming judgments that are brittle and fail to stand the test of time. There is a direct tradeoff to be managed here – balancing the time it would need to make the 'perfect' decision, with the resources (including attention span) and windows of opportunity to act that emerge unpredictably.

Continuity of Values

People agree and disagree - with each other at any one time, and with themselves over time. Maintaining continuity of values across participants and individuals over time enables similar responses to be made to similar situations with tolerable variation and unpredictability. This promotes trust and the realization of ideas that take time to complete – this applies especially to any change that is of strategic or Enterprise significance.

As an example, Reims Cathedral is famous for its coherence - its unity of design and implementation. The degree of conceptual integrity it expresses is remarkable considering that the project took centuries and eight generations of builders and architects to complete. Its success is down to what Frederick Brooks in The Mythical Man Month [Ref 1] calls Conceptual Integrity. He poses the questions:

1. How is Conceptual Integrity to be achieved?
2. Does this means imply an elite, or aristocracy of architects and a horde of plebeian implementers whose creative talents and ideas are suppressed?
3. How does one keep the architects from drifting off into the blue with unimplementable or costly specifications?
4. How does one ensure that every trifling detail of an architectural specification gets communicated to the implementer, properly understood by him and accurately incorporated into the product?

All these questions are familiar to the experienced Enterprise Architect. Sir Geoffrey Vickers explores some of these challenges in his writing on Limits of the Regulable [Ref 2]. The tools in an Enterprise Architects toolbox are similar to those available to the architects of Reims Cathedral and the regulators of industry – Rules and Models. *Figure 2* below highlights, in common terms, the types of Artefact involved and graphically makes the point that the art of Architecture regulation lies in consciously choosing the level at which to regulate the emergence and implementation of system design and implementation. For example, the Enterprise Architect may choose to regulate inter-capability relationships (e.g. between Applications and Infrastructure) at a low level (Models), while regulating the content of the capability at a high level (Rules).

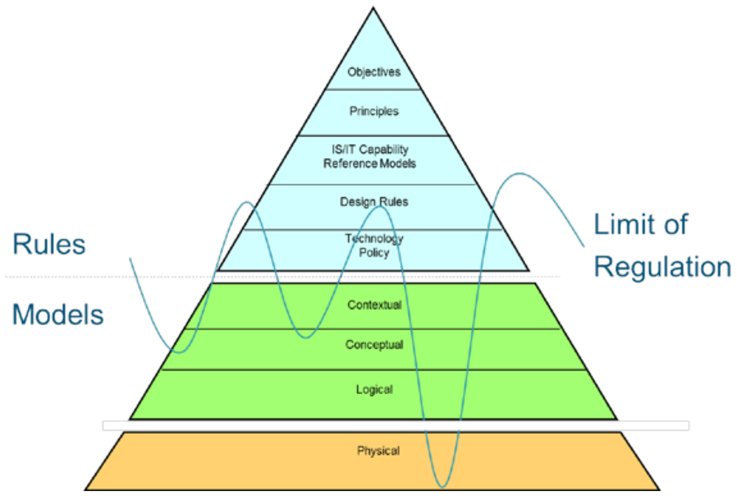

Figure 2: Flexing the limits of Regulation

Rules are flexible and ambiguous; Models are less flexible and more precise. Both are valid and appropriate under the right circumstances – experience and judgment is needed to understand when, which is appropriate. Rules and Service Models are relatively long-lived and resilient as they are tolerant of today's diversity and emergence of technology implementations over time. Resilience is a critical objective for Transition/ Transformation planning as circumstances change and problems are encountered. One useful concept here is that of a ratchet – that when circumstances do change and require adjustment of the target, the impact is contained and does not unravel the plan all the way back to the present time. The 'ratcheting' of decision and commit points is also a key quality to ensure that decisions can be trusted long into the future.

Continuity - of White Papers

I hope you have enjoyed this series of White Papers on *The Art of Judgment*. Please get in touch if you have views to offer on the topic and feedback on the series, either direct to Orbus or via my email at: ceri.williams@theintegrationpractice.co.uk

The next series of White Papers provides some continuity from The Art of Judgment, continuing along the theme of how best to equip an Enterprise Architect to deliver real and lasting effects by augmenting the traditional engineering disciplines with a Soft Systems approach. The next series starts in January with the first paper: A brief history of Systems Thinking and the Soft Systems method: A brief journey through the emergence of Soft Systems as a defined field, from before it was given a name to today's discipline. The connection with Enterprise Architecture and what it means for Enterprise Architects.

The first lines of the Wikipedia entry covering Soft Systems reads: "Soft systems methodology (SSM) is a systemic approach for tackling real-world problematic situations. Soft Systems provide a framework for users to deal with the kind of messy problem situations that lack a formal problem definition". Enterprise Architecture deals with real-world problematic situations and routinely encounters messy problem situations that lack a formal problem definition.

References (for this series):

Paper 1: Reality Judgment

[1] Vickers, G (1995) The Art of Judgment Centenary Edition. ISBN: 0-8039-7362-4

[2] Wicked Problems: http://en.wikipedia.org/wiki/Wicked_problem

[3] IEEE Std 1471 2000 ISBN: ISBN 0-7381- 2518-0 SH94869

Paper 2: Value Judgment

[2] http://www.newscientist.com/article/mg21729101.800-stupidity-what-makes-people-do-dumb-things.html

[4] http://papers.rumsfeld.com/about/page/authors-note

[5] http://www.kurtkleiner.com/stories/ut.why.smart.people.do.stupid.things.html

[6] http://en.wikipedia.org/wiki/MECE_principle

Paper 3: Instrumental Judgment

[2] http://en.wikipedia.org/wiki/Geoffrey_Vickers

[3] https://en.wikipedia.org/wiki/Unintended_consequences

Paper 4: Integrated Judgment

[1] The Mythical Man Month – Frederick P Brooks

[2] Vickers, G (1995) The Art of Judgment Centenary Edition. ISBN: 0-8039-7362-4: Chapter 8 – The Limits of the Regulable

Orbus Software
3rd Floor
111 Buckingham Palace Road
London
SW1W 0SR
United Kingdom

+44 (0) 870 991 1851
enquiries@orbussoftware.com
www.orbussoftware.com