Taking Governance to the Edge

Philip Boxer & Richard Veryard

November 2005

Summary

Discover the challenges faced by asymmetric forms of governance and an approach to working that focuses particularly on the way a business understands the risks that arise from how it relates to exogenous geometries, in addition to the more familiar risks associated with managing its endogenous geometry.

Introduction

There are two competing visions of service-oriented architecture (SOA) circulating in the software industry, which we can label as SOA 1.0 and SOA 2.0 (see Table 1). Our approach to governance is targeted at SOA 2.0. One of the central questions raised by Christopher Alexander in his latest four-volume work, *The Nature of Order*, is how to get order without imposing top-down (central) planning, or conversely how to permit and encourage bottom-up innovation without losing order (see Resources). Alexander promotes the concept of structure-preserving transformations and argues that under certain conditions large-scale order can emerge from an evolutionary process of small incremental steps, provided that each step is appropriately structure preserving. However, how (and in whose interests) do we define the structure that is being conserved, and at what level?

Note that with this concept of structure preservation, the leadership is not doing the design, but setting the parameters in which orderly design may take place. This function is very much a governance role. To answer the question, we need to distinguish governance from management.

Loose discussion of governance (both IT governance in general and SOA governance in particular) can easily spread until it appears to cover all aspects of management. So where does management stop and governance begin?

If we understand *management* as getting things done, then we can best understand *governance* as something like steering (as in *steering committee*). In the old days, it was common practice for large development projects/programs to have steering committees—often a separate one for each project—governing things like funding, scope, direction, and priorities. The steering committee was the forum to which the project manager was accountable. It was supposed to maintain a proper balance between the interests of the project and the interests of the whole organization, and to resolve conflicts.

However, there were several problems with this approach. First, the steering committee typically met infrequently and often only had a vague idea about what was going on. Second, the steering committees generally didn't talk to each other. Third, there were many important IT functions (such as architecture) and desired outcomes (such as productivity and reuse) that didn't have a steering committee. So the steering committee approach was incomplete, inconsistent, and often produced suboptimal results.

SOA 1.0	**SOA 2.0**
Supply-side oriented	Supply-demand collaboration
Straight-through processing	Complex systems of systems
Single directing mind	Collaborative composition
Controlled reuse	Uncontrolled reuse
Endo-interoperability (within single enterprise or closed collaborative system)	Exo-interoperability
Cost savings	Improved user experience

Table 1: Comparing SOA 1.0 and SOA 2.0

Real-Time Governance

The fashion shifted away from ad hoc steering committees, and many large IT organizations began to pay explicit attention to questions of IT governance. Where significant amounts of IT activity were outsourced, this governance included questions of procurement and relationships with key suppliers.

With SOA, we have a new approach to getting things done. We also have all the old problems of governance plus some new ones. There is a complex tapestry of service-oriented activity going on, all of which needs to be properly coordinated and aligned with business goals, and the timescale has changed. Instead of steering committees meeting every two months, we have real-time governance, covering both design governance and run-time governance.

We know that this cannot simply be reduced to a management problem. Many organizations experience difficulties in doing SOA properly, not because of technical problems but because of a lack of appropriate governance. How do you fund twin-track? How do you manage twin-track in such a way that the service goals are consistent with the business goals so that the SOA can be managed in an efficient way? And what happens when they are not consistent with each other?

We need to get away from the idea of a committee sitting around a table listening to progress reports and issue logs from project managers. We need to hold on to the idea that steering involves accountability to multiple stakeholders, and this accountability is included in a notion of governance that is critically about addressing questions of value, value for whom, and value to what end. These are essentially ethical questions (in the broadest sense of the word ethics—the science of value). Thus, while management is about getting things done, governance is about making sure the right things are done in the right way.

In a hierarchical IT organization in which “right” is defined at the top, this distinction might not seem to matter very much. However, if we are thinking about twin-track development where there are necessary tensions between business goals and service needs, let alone federated/distributed development where these tensions are replicated across multiple business entities, then it matters a great deal. As soon as two parallel activities (for example, service creation and service consumption) aren’t accountable vertically upwards to the same management point, governance becomes a question of negotiation between two separate organizations, rather than simple management resolution within a single right framework.

Put another way, hierarchical accountability depends on vertical transparency—what can be seen is what you can be held accountable for. Vertical transparency does not imply horizontal transparency across one or more vertically transparent hierarchies. Without horizontal transparency, how can service providers be held accountable to service needs and/or other business entities?

Governance has to determine how conflicts of interest between stakeholders are represented and contained in the interests of the whole. Put another way, it has to create a structure within which this representation is possible. If we understand stakeholders’ interests to be expressed as value from particular points of view (in a horizontal or vertical relationship to what is going on), then this question can be formulated in terms of a structure within which to distribute the benefits, costs, and risks between these interests. For example, SOA governance provides ground rules for interface agreements to be made and enforced. In the real world, we know that all agreements are subject to being reneged and renegotiated as requirements and conditions change. However, who incurs the risk of such changes, and who shall bear the cost of any change? If you decide you need to change the interface, am I forced to respond to this change, and if so, how quickly and who pays?

Structuring Transparency

There is also a conflict of interest between the present (short-term adaptation) and the future (longer-term adaptability). How shall adaptability be supported, how shall large complex solutions be permitted (nay encouraged) to evolve, and how shall this evolution be balanced against the need for order and short-term viability?

These questions can be asked at two levels: first at the management level, where there are a series of trade-offs and controls to be maintained, and second at the governance level, where we need to ask questions about the structure within which to distribute the benefits, costs, and risks on an ongoing basis.

Put another way, it’s a question of how we determine responsibilities and accountabilities, and the processes of negotiation between them. Ultimately, it’s a question of how to structure transparency: determining who is enabled to know the what, how, and when of things getting done in relation to whom. Governance vests authority by creating responsibilities with their associated accountabilities over the expertise and work mobilized by management (or rather managements), requiring that the appropriate forms of transparency be created.

One of the key principles for the management of complexity in IT and elsewhere is the separation of concerns. Separation of concerns implies selective attention: who pays attention to what. There is an important governance role here. Not only must governance make sure that attention is complete (at least in the sense that everything important is being attended to), efficient (without unnecessary duplication of attention), and connected (in the sense that the concerns can be brought together where necessary). It must also make sure that the conditions of transparency exist within which such attention is possible.

Creating these conditions of transparency implies a prior separation of unconcerns. Separating out what needs to be attended to from that which can (safely) be ignored brings us to a key governance concern: the governance of what can be ignored that is prior to the governance of conflicts of interest between concerns. What is anybody permitted to ignore? What forms of ignorance are mandated?

For example, SOA inherits notions of transparency from earlier work on open distributed processing. You can use a service without knowing its location; you can use data without knowing its source (provenance). This "not knowing" is very useful in some ways, but dangerous in others. It implies that there is no need for horizontal transparency.

SOA involves loose coupling (and horizontal coordination) not only between software artifacts, but also between the organization units that are responsible for these artifacts. The organization structure typically (although not always) reflects the software structure.

What happens to governance when we can no longer rely on a belief that somebody else is taking care of this risk?

Interoperability

It is sometimes supposed that the SOA agenda is all about decoupling. Requirements models are used to drive decomposition—the identification of separate services that can be used independently of each other when used. These separate services are then designed for maximum reuse, producing low-level economies of scale through satisfying an increased level of demand of any given type than was previously possible, and economies of scope through satisfying demand from a greater variety of contexts.

Clearly there are some systems that are excessively rigid, and will benefit from a bit of loosening up in such a way that their constituent services can be used independently of the system as a whole. This rigidity is only one side of the story, however. While some systems are excessively rigid, there are many others that are hopelessly fragmented: to get effective use from them, the independent services have to be made to work together. Thus, the full potential of SOA comes from decomposition and recomposition.

An important type of decoupling, as practiced by the military, is known as deconfliction. *Deconfliction* involves taking a whole mission and breaking it down into constituent missions that can be undertaken independently of each other, which produces a hierarchical chain of command in which the mission of any constituent component must depend on the way their mission fits into a larger whole defined by the way the command has imposed deconfliction. Deconfliction relates particularly to the effects that any mission creates, and the side effects of those effects on any other missions. Deconfliction therefore requires not only an understanding of how things work (that is, management), but also an understanding of how composite effects can be achieved from constituent effects with the minimum of conflict between the constituent components and the maximum efficiency in the utilization of resources. Deconfliction is therefore about uncoupling, but crucially about the way this uncoupling is done in relation to the overall mission. (see Table 2).

Deconfliction	**Interdependence**
Bush's gambit — filling the skies with bullets and bread — is also a gamble, Pentagon officials concede. The humanitarian mission will to some degree complicate war planning. What the brass calls "deconfliction" — making sure warplanes and relief planes don't confuse one another — is now a major focus of Pentagon strategy. "Trying to fight and trying to feed at the same time is something new for us," says an Air Force general. "We're not sure precisely how it's going to work out."	"We've gone from deconfliction of joint capability to interoperability to actually interdependence where we've become more dependent upon each other's capabilities to give us what we need."
"Facing the Fury", Time Magazine cover story, October 2001.	Interview with General Shoomaker. CSA, Oct 2004.

Table 2: From Deconfliction to Interdependence

The military take conflicts between the constituent components of a force very seriously—so-called friendly fire (where we kill our own side by mistake) is clearly a life and death issue. In our terms, friendly fire is an extreme example of interoperability failure. Deconfliction means organizing operations in a way that minimizes the potential risk of this kind of conflict, so that separate units or activities can be operated independently and asynchronously while still contributing to the overall mission.

But deconfliction often also involves a costly trade-off. Resources have to be duplicated, and potentially conflicting operations are deliberately inhibited. The pressures for more efficient use of resources forces specialization capable of delivering economies of scale and scope, combined with increasingly dynamic and political missions, to confront any chosen approach to deconfliction with increasing levels of interdependency.

The response to this issue, as the technologies of communications and control have become more sophisticated and reliable, is to increase the degree of coordination possible between the constituent components of a force, to allow units and activities to be composed in more powerful ways, which is the motivation for network-centric warfare. Rather than depending on a prior plan based on a particular deconfliction, the network enables commanders to coordinate dynamically the relationships between force components as the composite effects they need to produce themselves change. It is this use of the network that makes it possible to take power to the edge of the force where it meets the enemy.

Interop Risks

Commercial and administrative organizations typically attempt deconfliction through a management hierarchy and through an associated accounting structure of budgets and cost centers that create vertical transparency consistent with the form of deconfliction. This process is known to be inflexible and inefficient, producing silos of activity that are relatively impervious to demands for different organizations of their activities. Power to the edge (and, arguably, advanced forms of SOA) is not compatible with traditional budgeting and cost accounting structures.

Deconfliction leads us toward an arms-length notion of interoperability: X and Y are interoperable if they can operate side by side without mutual interference. This operation is the driver behind the uncoupling agendas of SOA, and it does yield improvements in economies of scale and scope.

There is also a positive notion of interoperability: X and Y are interoperable if there can be some active coordination between them. This notion requires us to go beyond the deconfliction per se, and not only to consider the assumptions about composition implicit in the form of deconfliction, but also to consider how they can be made explicit and dynamic through coordination. However, this is now network (horizontal) coordination rather than hierarchical top-down planning, which raises the same governance questions we discussed earlier in the relation between the twin tracks of pursuing business and service goals.

We are particularly focused on the risks associated with interoperability, not only because any given geometry is a particular coordination between its constituent services, but also because these are the ones that emerge as you coordinate across systems and organizations. (We have been following the recent developments around Hurricane Katrina with considerable interest because some of the public criticism of the Federal Emergency Management Agency [FEMA] clearly exposes difficulties in managing some of the interoperability risks.) How are we to think about the nature of these risks?

Governance here involves setting the terms of reference within which management is charged with making certain things interoperable. Interoperability risks may be ranked by severity: in this context this ranking means the extent to which a given risk jeopardizes the way we want to coordinate things.

Implicit in any form of deconfliction is an approach to coordinating the deconflicted activities through the way they will inter-operate. A simple view of interoperability is that it introduces a form of deliberate and selective ignorance: if you pay attention to X, you don’t have to pay attention to Y. For example, if you adopt this open standard, you don’t have to care which of these standard-compliant platforms may be used. This interoperation is a form of specialization (or separation of concerns) as described earlier, which makes this selective ignorance possible. It rests upon an assumption about what X and Y mean for those trying to use them to produce a combined effect.

Let’s think about interoperability of management spreadsheets in a large organization. Each manager produces his or her own spreadsheets in an idiosyncratic way to support a particular set of management decisions. Although they all import some data from the corporate database, they have mostly added data from elsewhere, and they have all formatted things differently. A senior manager, Joe, gives a presentation to a board meeting about a major strategic decision, supporting his recommendations with some charts drawn in Microsoft Excel that are derived from a complicated, handcrafted (and completely undocumented) spreadsheet. Joe’s colleagues find it impossible to understand his spreadsheet, or to import his analysis into their own spreadsheets for further analysis. Joe’s successor is more likely to build a new spreadsheet than try to use the existing one.

Interoperability fails at two levels here. Not only at the technical level of sharing the spreadsheet as a user-designed artifact, but it also fails at the level of *meaning*. The artifact is an expression of a framework of meaning created by Joe that is not shared by Joe’s colleagues and successors. Joe is trying to coordinate data in a way that requires him to make it interoperate in unfamiliar (nonstandard) ways. The very difficulties for managers collaborating on complex strategic decisions also reflect their potential value in creating new ways of acting.

Share the Commitment

What does this have to do with ignorance? It’s because of how much you need to know about Joe and his experience of management (that is, his framework of meaning) to make sense of his spreadsheet.The more complex the spreadsheet is, the more it becomes almost an embodiment of Joe himself and his way of paying attention to certain details of management. To use the spreadsheet, you almost need to get inside his skin, see things through his eyes. If Joe is powerful enough within the organization, then he can impose his experience of management on his colleagues and get them to use his spreadsheet, but that will typically result in interoperability problems elsewhere when data starts being used in new and unanticipated ways. A spreadsheet designed for reuse has to assume some level of shared understanding between the user and the originator.

For coordination between X and Y, they clearly have to be able to interoperate in a technical sense—Joe has to be able to send me his spreadsheet, and I have to have the right systems installed to be able to run it. But that is not sufficient. Let’s say we have P1 using X and P2 using Y. Coordination has to consider the effects not only of how X and Y interoperate, but also how P1 and P2 affect the meanings of X and Y. With coordination (or its lack) comes the risk that the way use is made of X by P1 and of Y by P2 will not produce the composite behavior expected. We can therefore understand the risks of coordination in the same way that we understand the challenges facing twin-track governance. They are created by a failure to establish a shared framework of meaning within which to act.

With directed composition (central planning, single design authority), the question of shared meaning and permitted ignorance is delegated vertically, but the resultant business geometry has to be endogenous to the interoperations of the activities under the hierarchy. Thus, A is decomposed into (or composed from) B and C; and if there are risks associated with the interoperability between B and C, then these risks are owned by the person in the design hierarchy that owns A. In an organization the requirement for vertical transparency is therefore to be able to enforce this shared commitment to vertical coordination.

In contrast, collaborative composition (planning at the edge between multiple design authorities) requires that shared meanings and permitted ignorances have to be negotiated, and that the resultant business geometry will have exogenous and endogenous elements that are assumed not all to be under the same hierarchy. Horizontal transparency therefore means being able to work out how all the pieces fit together as a whole in order that agreement can be negotiated about how to impose a single, albeit temporary, hierarchy for the particular purposes of collaboration, that is, horizontal coordination.

Ultimately there has to be a shared commitment to a single hierarchy, regardless of whether you are following a top-down (analytic, directed decomposition) process or a bottom-up (synthetic, collaborative composition) process, so long as there can be a shared commitment ultimately to a single hierarchy. The difference in the approaches lies in whether or not the resultant hierarchy is static or dynamic. From the point of view of a business providing services to its customers, static customization will involve agreeing to the hierarchy before entering into a business relationship. In contrast, dynamic customization implies that the very processes of agreeing appropriate hierarchies have to be part of the ongoing business relationship.

Risk Analysis

We are back to Alexander’s idea of the level at which we can be structure preserving: we have to be able to decide how much of the geometry has to be variable (underdetermining of the ways in which service users can dynamically customize its uses), and how much of it has to be fixed (overdetermining of the forms of business relationships that can be supported). Whereas symmetric forms of governance can impose vertical coordination, asymmetric forms of governance have to enable horizontal coordination, which raises new issues concerning the way trust is shared. Under vertical coordination it can be guaranteed by the contract with the superior context, whereas under horizontal coordination it has to be negotiated. This issue presents new challenges for distributed leadership under asymmetric forms of governance (see Resources by Huffington et al. and Boxer and Eigen).

In these terms we can see that vertical coordination is dominant when the HOW is dominant, whereas horizontal coordination is dominant when the WHY is dominant (see Figure 1). We can also see that as long as the HOW remains dominant, we are effectively externalizing the exogenous risks.

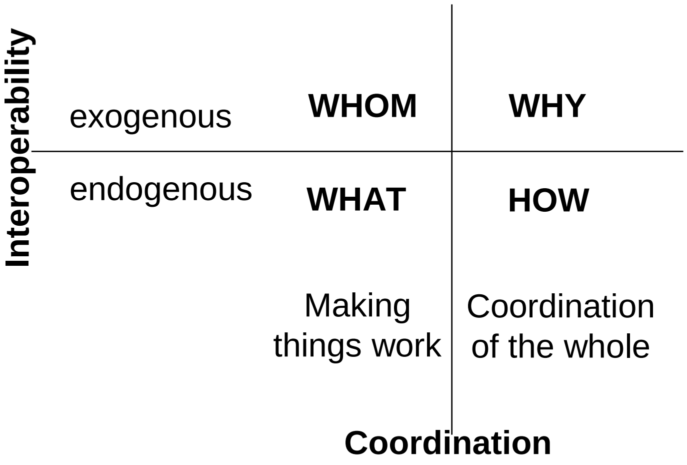

Figure 1: Relating the supply-side and the demand-side

We use methods of organizational analysis to distinguish the endo-interoperability risks (which come from failures within the organization) from three types of exo-interoperability risk (where the source is outside the organization). These exo-interoperability risks relate to what happens when a supplier's systems and services are combined with third-party systems and services as part of a user's solution. Thus, a supplier's system may not work as expected in the new context, it may not interoperate with other systems as expected, and the whole system of systems may not interact with the user's context of use as expected. These results can be thought of as errors of execution, of planning, and of intention within the user's domain.

We are aware of various analytical techniques for understanding and managing endo-interoperability risk. We are not aware of analytical techniques for understanding and managing exo-interoperability risk. This situation is not surprising in a world that defines its business as being one of pushing solutions, but in a service-driven world these risks begin to become the biggest as suppliers encounter the pull of emancipated users (see Hagel and Brown in Resources).

The shift from push to a pull is not only a matter of adopting new forms of governance and horizontal coordination. It also requires that the supplier adopt a *platform* mentality in which the platform is not theirs but the user's—at best they are providing a platform on which users can solve their problem. The general inability to manage exo-interoperability risks is not only a very major problem, but central to supporting a pull relationship with the user. Many of the supplier organizations we talk to are still in denial about this concern, but we are finding more user organizations that recognize the need for their suppliers to adopt a systematic approach to managing their exo-interoperability risks. What is at stake is the old arms-length procurement model that assumes that the user requirement can be separated from the context of use. Instead, users are looking for a collaborative approach to managing the risks attendant upon providing service platforms.

Comparison (WHOM-HOW)	The demand is defined in such a way that the context from which it arises is ignored, but the service is coordinated in a way that enables it to respond to that particular demand. This corresponds to the ‘comparison’ approach for the patient, who is looking for the best solution offered to their demand. For suppliers, this form of governance allows them to minimize their exposure to exo-risks by limiting the definition of demand that they will respond to (the user requirement), although they are still faced by integration risks inside the business.
Cost (WHOM-WHAT)	Not only is the demand defined in such a way that the context from which it arises is ignored, but the response of the service is proceduralized too, so that there is no need for an explicit coordinating process. This is the ‘cost’ approach, where both the nature of the demands and the responses to them have become standardized. Now the integration risk is minimized for the supplier, as well as the technology and engineering risks being proscribed.
Custom (WHY-WHAT)	An implicit form of coordination of how things work, often in the form of a particular budgetary regime, constrains the way the service is able to respond to the patient’s condition. This corresponds to the ‘custom’ approach, where the service is standardized, but the way it is provided into the patient’s context can be varied (viz ‘mass customization’). Here the supplier is exposed to integration risks again, as it builds more variability into the way its service works.
Destination (WHY-HOW)	Each of these three forms treats demand as symmetric to an implicit or explicit form of endogenous coordination. Only in the fourth case do we have asymmetric governance in which the endogenous and exogenous forms of coordination have to be aligned with each other. This corresponds to the ‘destination’ approach, where the patient goes to that place where they can get a treatment exactly aligned to the nature of their condition. It is also only in this case that the supplier takes on the exo-interoperability risks explicitly.

Table 3: Governance relationship patterns

Supply and Demand

The essential characteristic of this platform approach is that it manages both the supply side (that is, the endo-interoperability levels), and the demand side (that is, the exo-interoperability levels). In these terms we will see that the four positions in the cycle we described in "The Metropolis and SOA Governance" (*The Architecture Journal*, Vol. 1, No. 2 — see Resources) are four different patterns of governance relationships between the supply and demand sides, only one of which requires asymmetric governance. The other three use variations of vertical coordination (see Table 3).

If we adopt a platform approach, then we have to hold the four areas shown in Figure 2 in relation to each other in a way that is aligned to the WHY—in governance terms, the endogenous coordination imposed on the service by the trust (the HOW) has to be explicitly aligned to the particular form of exogenous coordination imposed on the patient's demand by his or her condition (the WHY).

The governance cycle shows how two different kinds of standardization (designed to reduce certain types of risk) have the effect of rotating a business or system away from asymmetric governance, while two different kinds of customization (which reintroduces certain types of risk) may give the business or system access to the potential rewards of engaging with asymmetry (see Figure 3).

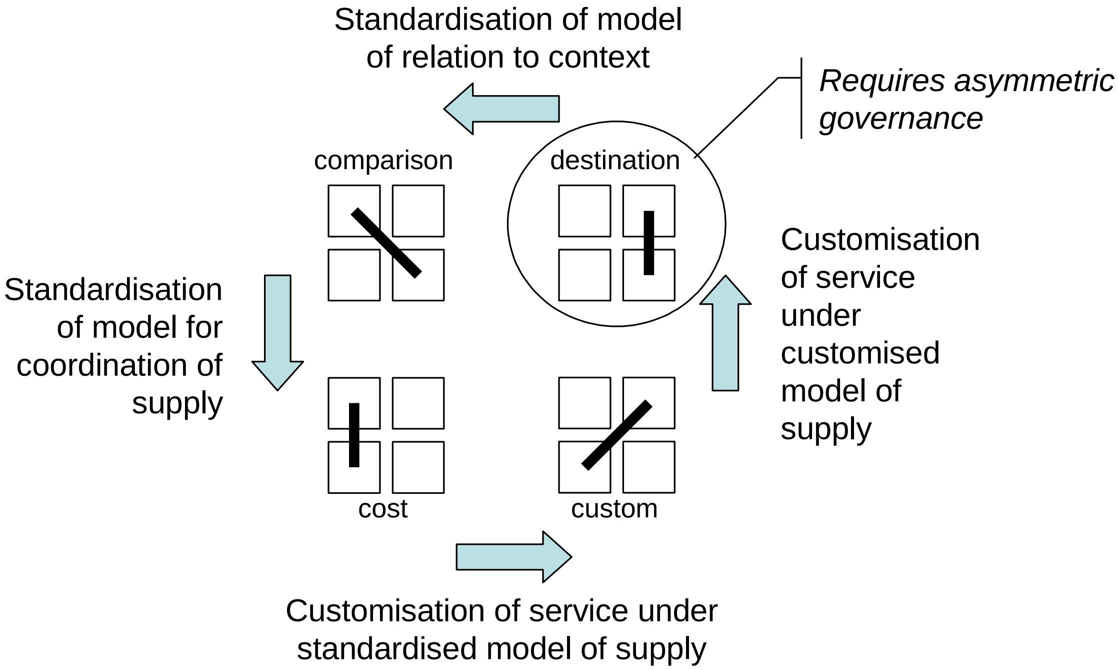
Standardisation of model of relation to context
Requires asymmetric governance
comparison
destination
Standardisation of model for coordination of supply
Customisation of service under customised model of supply
cost
custom
Customisation of service under standardised model of supply

Figure 2: The governance cycle

Where are the rewards commensurate with the risks in each case? We have to be able to understand how these different forms of governance are present or excluded within a supplying organization as it changes its relationship to demand, in response to changing competitive and demand circumstances. In Figure 3, the four forms are arranged to show the cycle discussed in our article "The Metropolis and SOA Governance" (see Resources). This cycle makes clear the transitions between each form, two of which require standardization, and two of which require customization:

- The move from destination to product involves reducing the exposure to integration risks by externalizing the exogenous risks.
- The move from product to cost involves reducing the exposure to the technology and engineering risks by standardizing the business model.
- The move from cost to custom involves increasing the exposure to integration risks again, but only the endogenous ones.
- Only the move from custom to destination faces the business with the exo-interoperability risks.

Where in the cycle the business needs to be will depend on how it chooses to balance risks and rewards. As Prahalad and Ramaswamy argue in "The Future of Competition" (see Resources), as demand has become increasingly asymmetric, so it has become increasingly critical for businesses to develop scalable models that can work in the destination quadrant as well. From an understanding of the whole cycle shown in Figure 3, it can be decided what forms of change are needed to capture the demands, and therefore what new forms of risk need to be mitigated in the pursuit of the rewards. While a lot is known about how to manage three of the forms of governance in this cycle, the difficulties of being able to transition through the fourth represents a major obstacle to the continuing growth of a business when faced with growing demand asymmetry.

In the Workshop

Turning to an orthotics case example, a clinic had been located within the Acute Trust to provide it with its own orthotics service (comparison), enabling the demand to be standardized to just those forms of demand that came from the consultants. Over time, the service itself and its budgets were standardized to align them to these forms of demand (cost). As a result, GPs had to refer patients through consultants, even when there was no real necessity to see the consultant, just to get access to the service. As a consequence, limited numbers of referrals were allowed directly to the service where the patient was known to the trust because of previous appointments (custom). The Primary Care Trust was responsible for all the patients in the catchment of the Acute Trust, however, and many of them were not receiving the service they needed. The challenge, therefore, was to enable the service to support these needs directly (destination). The Acute Trust resisted this support because of the need to fund such a service differently; and it was difficult for the Primary Care Trust to initiate because they didn't have the appropriate means for managing the balancing of the costs and risks of such a service. Put another way, new forms of asymmetric governance had to be developed.

To work with these issues within a supplying organization, we have developed a workshop process that distinguishes four teams: blue, white, red, and black, and is designed to unpack and articulate the different forms of interoperability risk associated with each of the quadrants. The workshop process is deliberately based on a military metaphor, but has been adapted for use by commercial/civilian organizations as well as military ones.

The typical objectives of the workshop are to learn collectively how these roles work in relation to one another in this specific organization at this time; to discover the extent to which this organization lacks capability in respect of these roles, and to start to develop this capability; and to obtain a snapshot of the current organization of demand facing this organization.

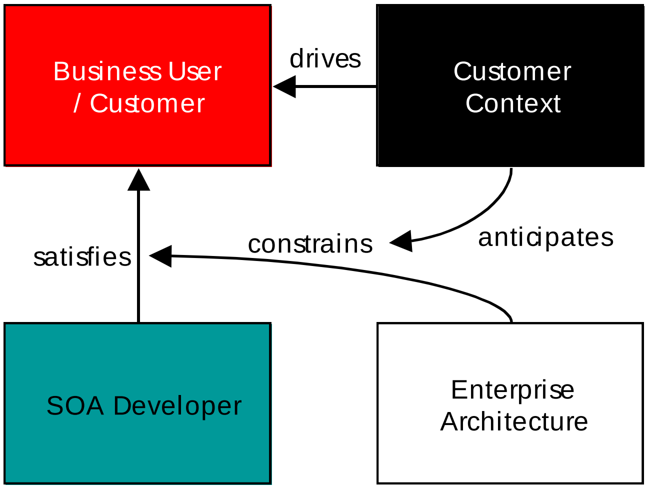

Figure 3: Role structure

Possible uses of this workshop process provide a way of approaching a number of issues relating to the impact of asymmetric forms of demand including: business strategy, organizational redesign, SOA design, SOA governance, security analysis, and governance (see Figure 3).

In our example, the blue team was the service, the white team was the Acute Trust, the red team was the patient and his or her GP, and the black team was the Primary Care Trust acting in the interests of the patients within its catchment. Table 4 shows how these teams relate to the different levels and asymmetries. The workshop works this way:

- Facilitating the recognition of the part played by each team, and the particular forms of risk each faced.
- Understanding the consequences of the presence/absence of certain colors for the governance process.
- Facilitating the conversations between all four colors in understanding the system as a whole and their interdependencies in managing risks.
- Using this understanding to develop strategies for managing the risks and agreeing on the basis on which white must determine what is in its interests.

Team	Team Style	Team Focus		Risk	Asymmetry
Blue (WHAT)	We do the business	The capabilities that our side want to use	Endogenous interoperation	I *behaviour* of technology	First
		What our side is capable of doing		II *engineering* of outputs	
White (HOW)	We are the ‘referees’ of …	… what it is in ‘our’ interests to do		III *integration* of business	Second
Red (WHOM)	We have the demand	What the demand side is capable of demanding of us	Exogenous interoperation	I. *execution* of solution	
		The way the demand-side makes use of what it demands		II *planning* of demands	Third
Black (WHY)	We anticipate …	… what are all the possible scenarios under which red team demands might arise		III *intentions* within context-of-use	

Table 4: Four Workshop Roles

Teamwork

The focus of the workshop is ultimately on the particular issues facing the white team in how it exercises governance:

- White constrains the way blue behaves in white's interests. Under conditions of asymmetric demand, white will have to choose to underdetermine blue and allow blue distributed leadership in the way blue responds to red within its particular black contexts.
- White therefore has to understand black to grant appropriate under-determination to blue to enable blue to satisfy red.
- In these terms, the symmetric governance of white becomes asymmetric as it addresses explicitly the consequences of black for how blue responds to red.

In this case the challenge was to make the asymmetric governance of the relationship between the service and the patients in the Primary Care Trust's catchment feasible, which in turn meant creating horizontal transparency: it had to be possible for the clinic to give an account of the way it aligned its treatment of a particular patient to the particular nature of that patient's condition and its outcomes. Thus, the Primary Care Trust was buying treatments of patients, rather than fixed numbers of treatment episodes, which involved creating a platform that could support this relationship and changing the procurement protocols to reflect the changed axis of accountability. In the article "The Metropolis and SOA Governance" (see Resources) we outlined the role played by this platform in support of a different form of governance. In an upcoming article we will address the analytical challenges involved in designing such a platform so that it is properly aligned to an asymmetric form of governance.

The asymmetry of governance explains a lot of the difficulties people have been experiencing with SOA. Many organizations will have enough short-term stuff to do anyway, without getting into this area, but we are starting to see organizations that are willing to take these challenges on board, and we are enjoying helping them.

Up to the latter part of the twentieth century, it could be assumed that the great majority of time spent by a business would be in the symmetric positions in the cycle. Only the generation of new business propositions needed to take place in the destination quadrant. The challenge presented by the twenty-first century is that these proportions are being reversed, so that although the symmetric positions remain important in harvesting the value of components within users' platforms, the greatest value will increasingly be created in the asymmetric part of the cycle. Developing an asymmetric capability is therefore of great importance to businesses increasingly competing in a flat world (see Friedman in Resources), in which component activities are outsourced under the pressures of globalization, digitization, and intensifying supply-side competition.

Resources

Boxer Research Limited www.asymmetricdesign.com

"From Push to Pull: Emerging Models for Mobilizing Resources," John Hagel and John Seely Brown (October 2005)

"Metropolis and SOA Governance," *The Architecture Journal* 5, Richard Veryard and Philip Boxer, Vol. 1, No. 2, (Microsoft Corporation, 2005)

"Special Issue on SOA Governance," CBDI Journal (November 2005)

"Taking power to the edge of the organisation: role as praxis," Philip Boxer and Carole Eigen, ISPSO 2005 Symposium, (2005)

The Future of Competition: Co-Creating Unique Value with Customers, C.K. Prahalad and Vebkat Ramaswamy (Harvard Business School Press, 2004)

The Nature of Order, Christopher Alexander (Center for Environmental Structure, 2003)

The World Is Flat: A Brief History of the Twenty-First Century, Thomas L. Friedman (Farrar, Straus and Giroux, 2005)

"What is the emotional cost of distributed leadership?" Working Below the Surface: The Emotional Life of Contemporary Organizations, Clare Huffington et al., Tavistock Clinic Series, (H. Karnac Ltd., 2004)

About the Authors

Philip Boxer is a strategy consultant based in the UK.

Richard Veryard is a writer, management consultant, and technology analyst based in London, UK.