The Federation Trilemma

Local autonomy, enterprise-wide consistency and speed of change: why an IT operating model can have any two, and why hybrid models sit in a middle that does not exist.

Abstract

Organisations that run information technology across several business units want three things at once: that each unit may decide its own systems (autonomy), that the enterprise has one landscape, one set of definitions and one answer (consistency), and that change is fast (speed). This paper argues that the three are pairwise compatible and jointly impossible. We at Nashua state the three goods precisely and show, with a minimal model of change and reconciliation, that their product is bounded by a constant set by the number of units, their interdependence and the latency of the reconciling forum, and derive the three coherent operating models: the centralised standard, the federated-but-slow and the autonomous-but-inconsistent. Each corner has its own governance instruments; the "hybrid" or "federated" model that most organisations declare mixes instruments from different corners and then loses one good covertly, through standards nobody follows, a board turned bottleneck, or shadow IT. The bound can be raised but not removed, by modularity, pre-agreed contracts and layering. We read the Dutch public sector through the model, where 342 constitutionally autonomous municipalities execute identical statutory tasks and the Common Ground movement attempts to move the frontier rather than escape it. This is theoretical research, and we close with how it could be measured and tested.

Keywords: IT governance; operating model; federation; enterprise architecture; trilemma; standardisation; autonomy; speed of change; modularity; Common Ground; Netherlands; municipalities

1. Introduction

Every organisation with more than one business unit has had the argument. The units want to choose their own systems, because their markets, their customers and their timing differ. The centre wants one landscape, one customer record, one chart of accounts and one security posture, because the enterprise reports, complies and defends itself as a whole. Everyone wants change to be fast, because the competition is. The argument is usually settled by a document that promises all three: a "federated" operating model, a "hybrid" governance, a "freedom within a framework." The document is followed, after some years, by a review that finds the framework unobserved, the architecture board a bottleneck, and a shadow landscape of systems bought without anyone asking. The review recommends a new operating model, which promises all three.

This paper argues that the argument cannot be settled by a document because the three goods cannot be held together. By *autonomy* we mean that a business unit decides what systems it runs, when it changes them and with whom; by *consistency* we mean that the enterprise has one landscape in the sense that matters, shared definitions, shared platforms where sharing is intended, and interoperability where it is not; by *speed* we mean the time from a need arising to its being met in production. Our claim is that an IT operating model can have any two of these at full strength, and that the third is then bounded by a constant that depends on the number of units, how interdependent their systems are, and how long it takes the enterprise to reconcile a change made by one unit with the others. The claim is the form of a trilemma, familiar from monetary economics (Mundell, 1963; Fleming, 1962), from political economy (Rodrik, 2011) and from distributed computing (Gilbert & Lynch, 2002; Brewer, 2012), and we think it is as useful here as it has been there: not because it tells an organisation what to choose, but because it tells it that it must.

The question the paper asks is therefore: **are local autonomy, enterprise-wide consistency and speed of change jointly achievable in an IT operating model, and if not, what are the coherent choices and how can an organisation tell which one it has actually made?** Section 2 places the question in the governance literature and among the trilemmas it resembles. Section 3 defines the three goods and the instruments that pursue them. Section 4 gives the model and the bound. Section 5 describes the three corners, their instruments and what each gives up, and Section 7 the pathology of the interior. Section 8 examines what can move the bound. Section 9 reads the Dutch public sector through the model and Section 10 traces a constructed case. Sections 12 to 15 draw implications, state limits and conclude.

The paper is written for the Dutch situation, which offers the trilemma in an unusually pure form. The Netherlands is a decentralised unitary state: its 342 municipalities execute the same statutory tasks under the same laws, with constitutionally protected autonomy over how they organise themselves, and they do so with information systems they choose and buy individually from a small market of suppliers. Consistency is demanded by law and by citizens who move between municipalities; autonomy is guaranteed by the constitution; and speed is demanded every time Parliament changes a rule that every municipality must then implement. The collective response, the Common Ground movement, is read in Section 9 as a deliberate attempt to move the frontier.

2. Background

2.1. Governance archetypes and operating models

The IT governance literature has described the choice between centre and units for decades without, to our knowledge, stating it as an impossibility. Weill and Ross (2004) catalogued the archetypes by which enterprises allocate IT decision rights, business monarchy, IT monarchy, federal, feudal, duopoly and anarchy, and showed that top performers matched the archetype to the decision. Ross, Weill and Robertson (2006) proposed four operating models on two dimensions, the degree to which business units standardise processes and the degree to which they integrate data: diversification (low on both), coordination (integration without standardisation), replication (standardisation without integration) and unification (both). Sambamurthy and Zmud (1999) argued that the right arrangement depends on multiple contingencies and that most enterprises end up federal. These frameworks are the vocabulary of every operating-model document, and they share a feature: they treat the choice as a position on a plane, where any point can be chosen, rather than as a constraint, where some points cannot.

The organisation literature knew the constraint earlier. Lawrence and Lorsch (1967) showed that differentiation among units and integration across them are in tension and that the integration required rises with differentiation. Mintzberg (1979) described the coordinating mechanisms available to an organisation and the conditions under which each works, and noted that the mechanisms are not freely combinable. Ashby's (1956) law of requisite variety implies that a centre which wants to control the variety of its units must itself have that variety, which is to say that consistency imposed from the centre has a cost that grows with the units' diversity. Conway (1968) observed that a system's structure copies the communication structure of the organisation that builds it, which in our terms means that an organisation that communicates through a central board will build a landscape with a central bottleneck.

2.2. Trilemmas as a form

A trilemma is a claim that three desirable properties can be held pairwise but not jointly. The best known in economics is the impossible trinity of Mundell and Fleming: a country cannot have a fixed exchange rate, free capital movement and an independent monetary policy at once. Rodrik (2011) proposed that democracy, national sovereignty and deep economic integration are likewise jointly unattainable. In computing, Brewer's conjecture, proved by Gilbert and Lynch (2002), holds that a distributed data store cannot simultaneously guarantee consistency, availability and tolerance of network partitions. Brewer's (2012) retrospective is instructive for the present paper: the theorem's value in practice was less the impossibility, which engineers worked around, than the discipline of saying explicitly which property a system gave up and when.

The CAP theorem is the nearest relative of our claim, and the resemblance is not accidental: an enterprise with autonomous units is a distributed system in which the units are nodes, the shared definitions are the replicated state, and the forum that reconciles changes is the consensus protocol. The present paper does not borrow the proof, which concerns asynchronous networks, but it borrows the form and the discipline.

2.3. Modularity

The third strand we draw on concerns what can move a constraint of this kind. Simon (1962) argued that complex systems that survive are nearly decomposable: their parts interact strongly within and weakly between, so that a change in one part rarely requires a change in another. Parnas (1972) made the same point for software: modules should hide the decisions most likely to change, so that changing one does not require changing the rest. Baldwin and Clark (2000) showed how design rules that fix interfaces let modules evolve independently, and how that independence has an option value. In the terms of this paper, these authors describe how to lower the interdependence between units, which is one of the two ways the trilemma's bound can be raised.

3. Three goods and their instruments

3.1. Definitions

We define each good so that it could be measured, because the argument turns on their being in tension and tension between undefined goods is rhetoric.

Autonomy (A). The share of decisions about a unit's systems, which to run, when to change them, with which supplier, that the unit takes without the consent of another party. Measured as the share of a unit's IT change decisions, weighted by spend, taken locally. $A = 1$ means the unit decides everything; $A = 0$ means the centre does.

Consistency (C). The degree to which the enterprise has one landscape where it intends to. Measured as the share of shared concepts (customer, product, account, employee, case) that have one definition and one system of record across the units, and the share of the estate that runs on the platforms designated as shared. $C = 1$ means one landscape; $C = 0$ means each unit's landscape is unrelated to the others'. Consistency is not uniformity: an enterprise may intend to share identity and finance and not front-ends, and C is measured against what it intends.

Speed (S). The rate at which admissible change is delivered relative to the rate at which it is demanded: the ratio of changes reaching production in a period to changes requested in it, or equivalently the inverse of the lead time relative to the lead time the units need. $S = 1$ means every request is met as fast as it is wanted; $S < 1$ means a growing backlog. "Admissible" matters and is discussed below: a change the model refuses by design is not a slow change but an excluded one.

3.2. Instruments

Each good is pursued with instruments, and instruments are where operating models become real. The instruments of autonomy are decentralised budgets, local supplier relationships, local product ownership and the absence of approval

steps. The instruments of consistency are standards and reference architectures, shared platforms and shared services, central data definitions and systems of record, and the forums, architecture boards, design authorities, data councils, that reconcile local proposals with them. The instruments of speed are short decision paths, product teams with end-to-end authority, automated delivery, and the elimination of forums from the path to production.

Listed this way, the tension is already visible. The instruments of consistency insert a forum into the path to production, which is what the instruments of speed remove. The instruments of autonomy remove the approval that the instruments of consistency require. An operating model that adopts all three sets adopts instruments that undo each other, and Section 7 describes what happens next. First we show that the tension is not merely between instruments but between the goods themselves.

4. A minimal model

4.1. Setup

Consider an enterprise with n business units. In each period, each unit has a demand for change in its systems at rate r: new features, new suppliers, new integrations, retirements. A fraction A of those changes is decided locally; the remainder is decided by the centre and does not concern us here, because the centre's decisions do not need reconciling with anyone. Of the locally decided changes, some touch things the enterprise intends to keep consistent, shared data, shared platforms, shared interfaces; the share that does so is C, because the more of the landscape is designated shared, the more local changes touch it. Each such change must be reconciled with the other units that depend on the shared thing; let k be the average number of other units involved, a measure of interdependence. Reconciliation is performed by a forum with finite attention: it can settle one pairwise reconciliation in time τ, serially.

4.2. The bound

The reconciliation load generated per period is the number of local changes that touch shared things, multiplied by the units each must be reconciled with:

$$\text{load} = nrACk. \tag{1}$$

Equation 1. Reconciliation load per period: local changes that touch shared things, times the units each must be reconciled with.

The forum's capacity is $1/\tau$ reconciliations per period. If load exceeds capacity the backlog grows and the delivered rate of admissible change is capacity-limited; the speed, delivered over demanded, is

$$S = \min\left(1, \frac{1/\tau}{nrACk}\right). \tag{2}$$

Equation 2. Speed as delivered over demanded change, capped by the forum's reconciliation capacity.

Whenever the forum is the binding constraint, which is the case the operating-model argument is about,

$$A \cdot C \cdot S \;\le\; \frac{1}{\tau rnk} \;\equiv\; \kappa. \tag{3}$$

Equation 3. The bound: the product of autonomy, consistency and speed cannot exceed the constant κ.

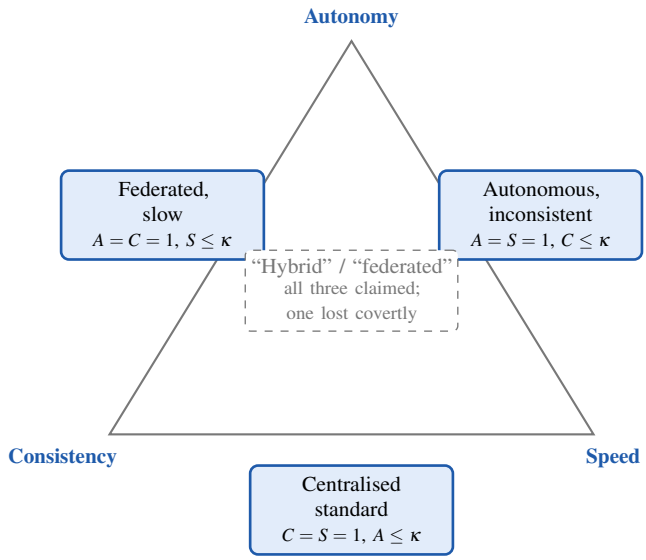

Figure 1. The federation trilemma. Each edge is a coherent operating model holding two goods at full strength; the third is bounded by κ. The interior is where most declared models sit and where none can stay.

The product of the three goods is bounded by κ, a constant set by the forum's latency τ, the demand for change r, the number of units n and their interdependence k. For any enterprise of realistic size, κ is small: with twelve units each wanting four reconcilable changes a month, each touching three other units, and a board that settles a reconciliation in a week, $\kappa \approx 1/36$. Two goods at full strength force the third to that order of magnitude.

4.3. What the bound says and does not say

Inequality (3) is a sketch, and three things should be said about it. First, it is an inequality about a regime, the one in which the reconciling forum binds; an enterprise with so little change, so few units or so little interdependence that the forum is idle is not in that regime and feels no trilemma. Most enterprises with several units are in it. Second, the goods are not symmetric in how they are lost. Autonomy and consistency are chosen, by design or by default; speed is what happens, and it is lost by queueing, which is why operating-model reviews so often find a backlog nobody designed. Third, the bound is not a law of nature but a consequence of reconciliation being serial, pairwise and slow. Each of those three can be attacked, and Section 8 examines what happens when they are.

The three corners follow directly. Set $A = 1$ and $C = 1$: speed is bounded by κ, the federated-but-slow model. Set $C = 1$ and $S = 1$: autonomy is bounded, $A \le \kappa$, the centralised standard. Set $A = 1$ and $S = 1$: consistency is bounded, $C \le \kappa$, the autonomous-but-inconsistent model. Figure 1 draws them.

4.4. Backlog dynamics: how speed is actually lost

The bound describes a steady state. The way an enterprise arrives at it is through a backlog, and the backlog has a dynamic worth stating because it is what operating-model reviews observe. Let $B(t)$ be the number of reconciliations

waiting. Then

$$\frac{dB}{dt} = nrACk - \frac{1}{\tau}, \tag{4}$$

Equation 4. Backlog dynamics: the queue of reconciliations grows whenever load exceeds capacity.

which is positive whenever load exceeds capacity, so the backlog grows linearly and without limit. The lead time a new proposal faces is the backlog ahead of it times τ, so lead time also grows linearly: an enterprise that is ten per cent over the forum's capacity adds ten per cent of a forum-cycle to every proposal's wait each period, indefinitely. This is why the complaint about governance is always that it is *getting* slower rather than that it is slow, and why staffing the forum, which raises $1/\tau$, buys only a proportional reprieve: the load is set by n, r, A, C and k, and a forum that is twice as fast is overwhelmed by an enterprise that is twice as large.

Three responses are observed, and the model places each. A fast-track exempts some proposals from reconciliation; it lowers C for those proposals, covertly. A tolerance for "technical debt" lets proposals through with reconciliation deferred; it lowers C now and raises load later. And a moratorium on change lowers r, which is the only response that attacks the load directly, and which no enterprise in a competitive market can sustain.

4.5. Parallel forums

The bound assumes one serial forum. An enterprise can run several in parallel, one per domain, and the bound for it is

$$A \cdot C \cdot S \;\le\; \frac{\phi}{\tau r n k}, \tag{5}$$

Equation 5. The bound with ϕ parallel forums, each reconciling its own domain.

where ϕ is the number of forums that can work independently. Parallelism helps exactly to the degree that the domains are independent: two forums that must reconcile with each other are one forum with an extra step. This is the same condition that Section 8 finds for layering, and it is not an accident; parallel forums and layered corners are the same move seen from governance and from architecture respectively. Where the domains are genuinely independent, ϕ can be large and the bound loose; where every domain touches every other, ϕ is one however many committees exist.

4.6. Reading the frontier

Figure 2 draws the bound as a frontier in the plane of consistency and speed, at full autonomy. An enterprise can be anywhere below its curve and nowhere above it, and operating-model documents that claim both goods near one are claiming a point above every curve. The figure also shows what the frontier-moving investments of Section 8 do: each pushes the curve outward, so that more consistency can be had at a given speed, but each curve still bends away from the corner. There is no investment that reaches it, because reaching it would require reconciliation to take no time, and reconciliation takes time because it is a decision.

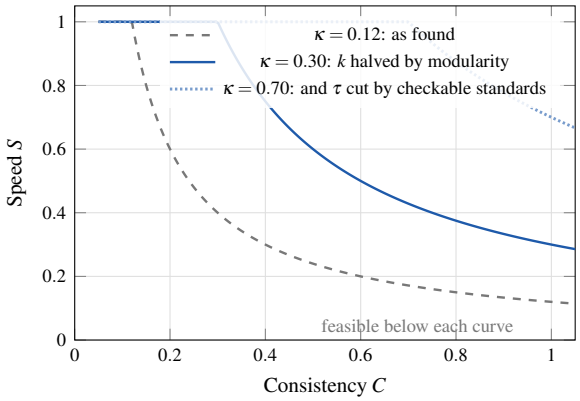

Figure 2. The frontier $S \le \kappa/C$ at full autonomy ($A = 1$). Each curve is the most speed an enterprise can have for a given consistency; points above it are infeasible. Lowering interdependence (k) and reconciliation latency (τ) moves the frontier outward but never to the corner $(1, 1)$: the trilemma is relaxed, not removed. Values illustrative.

5. The three corners

Each corner is a coherent operating model: a pair of goods held at full strength, a set of instruments that belong together, and a good that is given up explicitly. Table 1 summarises; the text describes each and names its nearest relative in the Ross, Weill and Robertson (2006) typology.

5.1. The centralised standard

One platform, or a small designated set, serves all units. A central product organisation owns it, develops it and changes it once for everyone. Units express their needs as configuration within the standard; needs that the standard cannot meet are refused, which is why the corner's speed is high: the forum is not on the path to production because there is nothing to reconcile, and the changes that would have required reconciliation are excluded rather than queued. Autonomy is given up openly. The corner suits enterprises whose units are similar enough that one standard fits, and it fails, visibly, when a unit's market requires what the standard refuses; the unit then either loses the market or builds outside the standard, at which point the enterprise has left the corner without deciding to.

5.2. The federated, slow model

Units implement locally against shared standards, definitions and reference architectures, and every local change that touches a shared thing passes through a forum that checks it against the others. Autonomy and consistency are both real: units choose, and the landscape coheres. Speed is given up, and the bound says by how much: lead time grows with the number of units, their interdependence and the forum's cycle. The corner is honest when the enterprise accepts the lead time, plans change on the forum's calendar and staffs the forum to run it. It is the natural model for enterprises whose units are genuinely different and whose shared concepts genuinely matter, regulated financial groups being the usual example, and it is the model most "federated" documents describe without accepting its price.

Table 1. The three coherent operating models, their instruments, and what each gives up.

Corner	Holds	Instruments that belong to it	Gives up, explicitly
Centralised standard (unification)	Consistency, speed	One platform or a small set; central product teams; local needs met by configuration; standards enforced by the platform, not by a board; central budget	Autonomy: units cannot choose systems or suppliers; needs outside the standard are refused, not queued
Federated, slow (coordination)	Autonomy, consistency	Shared standards and definitions; local implementation; architecture board and data council on the path to production; exception process; reference architecture	Speed: every local change touching shared things waits for reconciliation; lead time grows with units and interdependence
Autonomous, inconsistent (diversification)	Autonomy, speed	Decentralised budgets; local supplier choice; minimal interoperability standards; thin API contracts; after-the-fact mapping for reporting; tolerance of duplicate systems	Consistency: several definitions of shared concepts; duplicate platforms; enterprise questions answered by reconciliation after the fact or not at all

5.3. The autonomous, inconsistent model

Units choose and change freely, with decentralised budgets and local suppliers, and the enterprise connects them with the thinnest possible contracts: a few interoperability standards, published interfaces, and a mapping layer that reconciles data after the fact for reporting. Consistency is given up: there will be several customer records, duplicate platforms and enterprise questions that take a project to answer. The corner suits holdings whose units share little and conglomerates whose value is in the units rather than the whole, and it is honest when the enterprise accepts that it will report by reconciliation and will not have one landscape. Its characteristic failure is a regulator, an acquirer or a crisis that demands an enterprise-wide answer the model cannot give.

5.4. Admissibility and the meaning of speed

The centralised standard's speed deserves a second look, because it is where the model's definition of speed matters. The corner is fast for admissible change and refuses the rest. A unit that needed the refused change does not experience speed; it experiences exclusion, and if the exclusion is frequent the unit will act as though in the autonomous corner. The operating model is then two models at once, which is the interior. The lesson is that a corner is a choice about what counts as a legitimate need as much as about how fast legitimate needs are met, and that the centralised corner is coherent only where the standard's refusals are rare and accepted.

6. Instruments in detail

Operating-model documents are lists of instruments, and whether a document is coherent can be read from the list alone. Table 2 takes the instruments that recur in such documents and records, for each, the corner it belongs to and what it does to the three goods. An instrument raises the goods of its corner and lowers the third; a document whose instruments come from more than one corner has, by construction, adopted instruments that work against each other, and the daily contest between them is what produces the pathologies of Section 7.

Two patterns in the table deserve comment. The first is that the instruments most often found together in a "federated" document, an architecture board, decentralised budgets and product teams with end-to-end authority, come from three different corners. The board lowers speed, the budgets lower consistency, and the product teams lower consistency again, and the document has promised all three goods. The second is that the frontier-moving instruments in the last block are the only ones that do not appear with an arrow against any good. They are also the ones least often found in operating-model documents, because they are architecture rather than governance, and the documents are written by governance.

Skelton and Pais (2019) describe the platform team as a type that serves stream-aligned unit teams through a well-defined interface; in the table it is listed as frontier-moving with a condition, because a platform team whose interface changes with every request is a central forum in disguise, and a platform whose interface is stable is modularity. Galbraith's (1973) analysis of organisations as information-processing systems anticipates the whole table: coordination load can be met by increasing the capacity to process it, which is the forum, or by reducing the need for it, which is modularity, and the second is cheaper whenever it is available.

7. The interior: hybrid models and covert loss

7.1. What a hybrid claims

Most operating-model documents we have read claim all three goods. The claim is seldom stated as a claim; it is stated as a set of principles, "units own their business outcomes," "one enterprise, one landscape," "we move at the speed of the market," and a set of instruments drawn from all three corners: a reference architecture and a board (coordination), decentralised budgets and local supplier choice (diversification), and product teams with end-to-end authority (unification). Inequality (3) says the three goods cannot all be at full strength. Something gives, and because the document has not said what, it gives covertly.

7.2. The three pathologies

Which good is lost depends on which instrument wins the daily contest, and the loss has a recognisable form in each

Table 2. Common governance instruments, the corner each belongs to, and its effect on the three goods (↑ raises, ↓ lowers, · neutral). Frontier-moving instruments act on k or τ rather than on a good, and are the only ones that do not trade one good for another.

Instrument	Corner	A	C	S	Note
Single enterprise platform, central product team	Centralised	↓	↑	↑	Fast for admissible change; needs outside the standard are refused
Central IT budget, local charge-back	Centralised	↓	↑	·	Removes the local decision by removing the local money
Mandatory standard build and catalogue	Centralised	↓	↑	↑	Speed by limiting choice
Architecture board on the path to production	Federated	·	↑	↓	The forum of the model; its cycle is τ
Reference architecture with exception process	Federated	·	↑	↓	Exceptions are reconciliations by another name
Enterprise data definitions, data council	Federated	·	↑	↓	Raises C for shared concepts; adds a forum
Local implementation of shared standards	Federated	↑	↑	↓	The federated corner's defining move
Decentralised IT budgets	Autonomous	↑	↓	↑	Units buy what they need when they need it
Local supplier choice	Autonomous	↑	↓	↑	Divergence follows
Product teams with end-to-end authority	Autonomous	↑	↓	↑	Fast because nobody is asked
After-the-fact data mapping for reporting	Autonomous	·	↓	↑	Accepts inconsistency and reconciles later
Separation of shared data from local applications	Frontier ($k\downarrow$)	·	·	·	Fewer local changes touch shared things
Stable published interfaces, API contracts	Frontier ($k\downarrow$)	·	·	·	Modularity; Parnas's information hiding
Checkable standards, conformance in the pipeline	Frontier ($\tau\downarrow$)	·	·	·	Reconciliation becomes a test, not a meeting
Platform team serving unit teams	Frontier (both)	·	·	·	Coherent only if the platform's interface is stable

case.

Speed lost: the bottleneck. If the board is strong and the units compliant, consistency and autonomy hold and the backlog grows. Lead times lengthen; the board meets more often and falls further behind; a fast-track is created for "urgent" changes and soon carries most of them; the board is blamed for slowness and responds with more rigour. The enterprise is in the federated corner without having chosen it, and the pathology is that it still measures itself against the speed it promised.

Consistency lost: shadow IT and the paper standard. If the units are strong and the board weak, autonomy and speed hold and consistency is lost in two ways. Overtly, units buy and build around the board, and a shadow landscape grows that the enterprise discovers when it tries to answer an enterprise-wide question. Covertly, the board keeps approving, because refusing would slow things, and the standard becomes a document that describes a landscape that does not exist. The enterprise is in the autonomous corner and believes it is federated.

Autonomy lost: the central squeeze. If the centre is strong and uses the board as an instrument of control, consistency and speed hold for what the centre allows, and autonomy is lost: units find that every choice is already made. The enterprise is in the centralised corner, and the pathology is that its units still carry budgets and accountabilities that assume they can choose. Those units then act as in the second pathology, and the enterprise cycles.

7.3. Why the interior is unstable

An enterprise cannot stay in the interior because the instruments it has adopted contradict one another, and each day's decisions resolve the contradiction one way or the other. Over time the resolutions accumulate into a corner, usually not the one the document named. Reviews that find the enterprise in the wrong corner prescribe a new document that claims all three, and the cycle restarts. The way out is not a better document but an explicit choice, and the model's contribution is to make the choice unavoidable by showing that it is being made anyway.

8. Moving the bound

The bound $\kappa = 1/(\tau rnk)$ is a constant for a given enterprise, not a constant of nature. Three of its four terms can be changed by design, and the changes are what good architecture and good governance actually do. None removes the trilemma; each moves the frontier outward so that the chosen corner's sacrificed good is bounded less tightly.

8.1. Lower interdependence: modularity

The term k counts the units a local change must be reconciled with. It is set by the architecture: by how many shared things a local change touches, which is to say by how well the landscape is decomposed. Simon's near-decomposability, Parnas's information hiding and Baldwin and Clark's design rules are all prescriptions for lowering k: fix the interfaces, hide the decisions that change, separate what is shared from what is local so that a local change rarely crosses the line. A landscape in which data is held in shared registers behind stable interfaces and applications are local consumers of it has a much lower k than one in which each application holds its own copy of the shared data, because a change to a local application then touches nobody. This is, as Section 9 argues, the premise of Common Ground.

8.2. Lower latency: contracts instead of forums

The term τ is the time to reconcile one change with one other party. In the model it is the cycle of a forum, because that is how most enterprises reconcile. It need not be. If the standard is precise enough that conformance can be checked rather than decided, by a schema, a test, an automated gate, reconciliation drops from weeks to minutes and τ falls by orders of magnitude. The architecture board's work moves from deciding each change to deciding the standards against which changes are checked, which it does rarely. "Governance as code" is the practitioner name; in the model it is the substitution of a contract for a meeting.

8.3. Layering: a corner per layer

The bound applies to a set of shared things with a given interdependence. An enterprise's landscape is not one such set but several layers with different k between them: identity, security and master data, which nearly everything touches; core transactional processes, which some units share; and customer-facing front-ends and experiments, which almost nothing else depends on. Nothing in the model requires the same corner for every layer. An enterprise may be centralised for identity and master data, federated for the shared core, and autonomous at the edge, holding different pairs of goods at different layers, with each layer's sacrificed good bounded by that layer's own κ. This is Ross, Weill and Robertson's (2006) prescription to standardise the core and free the edge, restated as a consequence of the bound rather than a preference, and it comes with a condition the preference lacks: layering works only where the layers are nearly decomposable, so that a change at the edge does not touch the core. Where it does, the layers share a k and the trilemma applies to them jointly.

8.4. Time-splitting

A fourth move alternates corners over time: periods of consolidation in which the centre standardises, followed by periods of autonomy in which units diverge. The bimodal prescriptions of the last decade are a version of this. The model treats time-splitting as legitimate only where the two modes have low interdependence, so that changes made in the fast mode do not generate reconciliation load in the slow one; where they do, the backlog from the fast mode is simply paid in the slow mode, and the enterprise has moved the trilemma in time rather than escaped it.

8.5. Reducing n and r

The remaining terms are less often available. Reducing n, the number of units, is a decision about the enterprise's structure rather than its IT, and it is the centralised corner by another route. Reducing r, the demand for change, is rarely in the enterprise's gift. Both appear in the bound to remind the reader that the trilemma is felt most by large enterprises in fast-moving markets, which is where operating-model documents are most often written and least often honest.

9. The Dutch public sector

9.1. The trilemma in pure form

The Netherlands is a decentralised unitary state. Its 342 municipalities execute, under national law, the same tasks: population registration, social assistance and support, permits, local taxation, civic records. The Municipalities Act and the constitution protect their autonomy over how they organise themselves. Citizens move between municipalities and expect the same treatment; the national government changes the law and expects every municipality to implement the change by the same date; and national facilities, the citizen digital identity, the base registers, the personal government portal, must interoperate with every municipal system. Autonomy is constitutional, consistency is statutory, and speed is demanded by Parliament. The trilemma could not be stated more cleanly.

9.2. Where the municipalities have been

For most of the period since municipal administration was computerised, the municipalities have occupied the autonomous corner in law and something worse in practice. Each buys its own systems; the market that serves them is small, with a few suppliers holding most of it; and the systems are, with exceptions, integrated suites in which each application holds its own copy of shared data. In the model's terms, A is high by constitution, C is low by consequence, and k is high by architecture, because a change to any one application touches the data every other application copies. The result is an enterprise, if 342 autonomous bodies may be called one, that is inconsistent *and* slow: inconsistent because nobody reconciles, slow because when a national change arrives, each municipality must wait for its supplier to change the suite and then migrate. That is not a corner but the interior reached by default, and it is the position the model predicts for high autonomy, high interdependence and no forum.

9.3. Common Ground as a frontier-moving programme

The Common Ground movement, begun by the municipalities' association around 2017, proposes to separate data from applications: data held once, in registers behind standard interfaces, and applications as interchangeable consumers of it. Read through the model, this is an attack on k and τ together. Separating data from applications lowers k, because a change to an application no longer touches the data others depend on. Standard interfaces lower τ, because conformance to a published interface can be checked rather than negotiated. The movement's aim, in our terms, is to raise κ far enough that the municipalities can hold autonomy and

consistency, each choosing its applications against shared data, without the speed penalty that the federated corner otherwise charges.

The model also predicts where the programme strains, and we offer the prediction as a test rather than a verdict. The interfaces must be adopted by all 342 municipalities and by the suppliers who serve them; until they are, the data is not shared and k has not fallen. Adoption of a standard across hundreds of autonomous bodies is itself a reconciliation problem with $n = 342$, and the bound says it will be slow in proportion. The programme's progress should therefore be slower than its advocates hoped and faster than its critics feared, concentrated where the interfaces are few and stable, and limited where they are many and changing. The national layer, the digital identity, the base registers, has by contrast always been in the centralised corner, one facility for all, and has been fast and consistent at the cost of municipal autonomy over it, which no municipality has asked for. The two layers are, in effect, a layered design that nobody designed.

9.4. The same pattern one level up

The national government exhibits the trilemma at its own level. Ministries are autonomous in their administration, consistency across them is demanded by citizens and by a government that wishes to act as one, and the mechanisms for reconciling them, government-wide standards, a central CIO function, are forums with the latency the model expects. The parliamentary inquiry of 2014 into government ICT projects found governance weak and control insufficient, which is what the interior looks like from inside. We do not develop the national case here, but we note that the same model applies, with a smaller n and, in our reading, a higher k.

9.5. Shared service organisations as a corner move

Dutch municipalities have, for two decades, formed joint arrangements under public law to run services together: regional organisations that handle tax collection, social services or ICT for several municipalities at once. Read through the model, a shared ICT organisation is a move by its members from the autonomous corner toward the centralised one for the layers it covers: the members give up the choice of systems in those layers in exchange for consistency among themselves and the speed that a single roadmap brings. The move is partial, because membership is voluntary and the members retain their own councils, and the arrangements show the model's predicted strains: members who want a choice the shared organisation refuses, a governing board of member representatives that becomes a forum with its own τ, and a tendency, once the number of members grows, for the shared organisation to drift from the centralised corner into the federated one, where it is slow. The model does not say whether the arrangements are a good idea; it says which corner they occupy and what they will be blamed for.

9.6. Three predictions

The reading of the municipalities invites three predictions, stated so that they can be wrong.

First, the time from a national legal change to its implementation across all municipalities should fall where the affected data has been placed behind a standard interface adopted by most municipalities and their suppliers, and should not fall where it has not, regardless of other investment. The association's implementation monitoring, which records adoption of the interfaces by municipality, is the data.

Second, municipalities in shared ICT arrangements should show faster implementation of national changes than comparable municipalities outside them, for the layers the arrangement covers, and slower change in those layers for their own local needs. The first is the centralised corner's speed; the second is its price.

Third, the strain in the Common Ground programme should be concentrated in the standards that change most often. Where an interface has been stable for years, adoption should approach completeness; where it has been revised repeatedly, adoption should stall, because each revision is a new reconciliation across $n = 342$ parties. If adoption stalls instead on the stable interfaces, the model's reading of the programme is wrong.

10. A constructed case

The case is constructed, not observed. It shows a declared hybrid, the pathology that follows, and the re-cut into layers. It is not a report on any company and its figures are illustrative.

10.1. Group H as declared

Group H is a Dutch holding with six business units: a retail chain, a wholesale business, a logistics subsidiary, two acquired companies in adjacent markets, and a digital venture founded three years ago. Its operating model, adopted after a review, is "federated": a group architecture board sets standards and a reference architecture; units own their budgets and choose their suppliers within the standards; a group data office defines shared concepts; and the group has committed to "market speed," with product teams in each unit. The document claims all three goods and adopts instruments from all three corners.

10.2. What happens

Within two years the pathologies of Section 7 are all present, because the units differ in which instrument wins. The retail and wholesale units, compliant and interdependent, experience the bottleneck: the board's backlog reaches forty items, the average time to decide an exception reaches eleven months, and both units plan their roadmaps around the board's calendar while the group measures them against market speed. The two acquired companies, used to autonomy, experience the paper standard: they continue on their own platforms, obtain exceptions in bulk, and the reference architecture describes a landscape they do not run. The digital venture experiences shadow IT: it buys software on corporate cards, integrates it with nothing, and is held up as the group's fastest unit. The group data office finds four definitions of "customer" in production and cannot say how many customers the group has. A second review is commissioned and recommends a new operating model.

Table 3. Group H: the three goods by layer, before (declared hybrid, one κ for everything) and after (a corner per layer, with k and τ lowered where it matters). Values are illustrative.

	A	*C*	*S*
Before: declared hybrid, all layers			
Claimed	1.0	1.0	1.0
Measured	0.7	0.4	0.3
After: a corner per layer			
Identity, security, finance, master data	0.1	1.0	0.9
Shared order-to-cash core	0.8	0.9	0.5
Front-ends and experiments	1.0	0.3	0.95

10.3. Group H re-cut

Instead of a new document claiming all three goods, the group chooses a corner per layer, and names what each layer gives up. For identity, security, finance and the customer and product master data, the group chooses the centralised standard: one platform each, a central product team, local needs met by configuration, and the units' autonomy over these things given up explicitly. For the order-to-cash core that the retail, wholesale and logistics units share, the group chooses the federated model and accepts its price: shared definitions, local implementation, and a board that is staffed to a published cycle, with roadmaps planned on that cycle rather than against market speed. For customer-facing front-ends and the digital venture's experiments, the group chooses the autonomous corner, with thin API contracts to the core and an explicit statement that these systems will be inconsistent with one another and reconciled for reporting after the fact.

Two frontier-moving investments accompany the re-cut. The master-data layer is rebuilt as registers behind stable interfaces, so that a change to a front-end no longer touches the data the core depends on; k falls for the edge layer. And the board's standards for the core are made checkable, so that conformance is tested in the delivery pipeline and the board decides standards rather than changes; τ falls for the core layer. Table 3 shows the goods by layer before and after, as the group would measure them under Appendix A.

Nothing in the re-cut gives the group all three goods; the table shows every layer giving one up. What it gives the group is a landscape whose inconsistencies are where it chose to have them, a board whose slowness is on a published calendar, and units whose autonomy is real where it was granted and absent where it was not, which is what the first document promised and could not deliver.

11. A second constructed case: eight municipalities

Group H is private and chose its layers from a position of authority. The second case is public, where authority is shared and the corners must be negotiated. It is constructed, not observed, and is not a report on any region.

11.1. Region R as it stands

Eight municipalities in a region, with populations from fifteen to ninety thousand, run their own ICT: eight contracts with three suppliers, eight implementations of the same statutory processes, and eight versions of each shared concept. Each is autonomous by constitution; none is consistent with the others; and each is slow, because each waits for its own supplier to implement national changes and then migrates alone. They sit in the interior by default: the autonomous corner's inconsistency without its speed, because k within each municipality's integrated suite is high and no local change is cheap.

11.2. The proposal and the trilemma it hides

A proposal to form a shared ICT organisation promises the eight municipalities that each will keep its autonomy, that the region will have one landscape, and that national changes will be implemented faster. The model says the three cannot all be true, and the proposal's own instruments say which will give: a joint board of eight council representatives to approve changes is a forum with $n = 8$, and a single shared roadmap is the centralised corner's instrument. If the board is strong the arrangement is federated and slow; if the roadmap is strong it is centralised and the municipalities' autonomy is nominal.

11.3. Region R re-cut

The region chooses a corner per layer and writes the choice into the arrangement's founding document, which is where the trilemma is usually hidden and where it should instead be stated. For the base registers, identity, security and the hosting of the statutory suites, the region chooses the centralised corner: one implementation, one roadmap, local needs met by configuration, and an explicit clause that members give up the choice of systems in these layers and may not run alternatives. For the statutory processes themselves, where municipalities legitimately differ in policy, the region chooses the federated corner and prices it: shared data definitions, local configuration of process, and a board that meets on a published cycle whose length is stated in the founding document, so that a municipality joining knows the lead time it is buying. For the citizen-facing front-ends and local initiatives, the region chooses the autonomous corner: each municipality may build or buy what it likes against the shared data layer's published interfaces, and the region states that these will be inconsistent with one another and that it does not mind.

The frontier-moving investment that makes the layering possible is the one Common Ground prescribes: the shared data layer sits behind stable interfaces, so that a municipality's front-end change touches nothing in the core and the autonomous layer is genuinely decomposable from the centralised one. Without that investment the layers share a k, the front-end layer's changes generate reconciliation load in the core, and the region is federated and slow throughout, whatever the founding document says.

11.4. What the region gains

Table 4 shows the goods by layer before and after, as the region would measure them. The region does not gain all three goods anywhere. It gains a landscape whose inconsistency is confined to the layer where it was chosen, a core that implements national changes once rather than eight times, and a board whose cycle is a term of the arrangement rather than a surprise. The members gain a clear answer to the

Table 4. Region R: the three goods by layer, before and after. Values are illustrative.

	A	*C*	*S*
Before: eight autonomous estates	1.0	0.2	0.3
After: a corner per layer			
Registers, identity, hosting, statutory suites	0.1	1.0	0.8
Statutory processes (policy differs)	0.7	0.8	0.4
Front-ends and local initiatives	1.0	0.2	0.9

question that joint arrangements usually leave open, which is what they have given up and in exchange for what.

12. Implications

Name the corner. An operating-model document should state which good it gives up, where and by how much, and should be refused if it claims all three. The statement is uncomfortable, which is why documents avoid it, and the discomfort is the point: the choice is made daily whether or not it is written down.

Match instruments to the corner. The instruments of Table 1 belong to their corners. A board on the path to production is a federated instrument and should not appear in a model that promises market speed; decentralised supplier choice is an autonomous instrument and should not appear in one that promises one landscape. Mixed instruments are the mechanism of covert loss.

Layer, and check decomposability. Choose a corner per layer, and before doing so check that the layers are nearly decomposable: that a change at the edge does not touch the core. Where it does, the layers are one layer and must share a corner.

Spend on the frontier. The investments that help are the ones that lower k and τ: separating shared data from local applications, fixing interfaces, making standards checkable. These are architecture investments, and they are the ones that let an enterprise hold its chosen pair of goods with a less painful third.

Measure the goods. Appendix A gives measures for A, C and S and for k and τ. An enterprise that measures them learns which corner it is actually in, which is usually not the one it declared, and can then decide whether to move or to admit.

13. Objections and replies

The trilemma invites objections, and several are worth answering because each is also a claim found in operating-model documents.

"Agile at scale gives us all three." The scaled-agile frameworks promise autonomous teams, enterprise alignment and fast delivery, which is the trilemma's three goods by other names. In the model, the frameworks' alignment ceremonies, the quarterly planning events at which many teams reconcile their intentions, are the forum, with a τ of a quarter; their autonomy is autonomy within what the ceremony agreed; and their speed is speed within the quarter. The frameworks are the federated corner with a short, fixed cycle, and they are honest about it in their mechanics and less so in their marketing. The model does not say they are wrong; it says which corner they occupy.

"The cloud makes consistency free." A shared cloud platform lowers the cost of running shared infrastructure, and an enterprise that moves all its units to one provider has consistency in that layer at the centralised corner's price, the loss of the units' choice of provider. It does not make the units' applications or data consistent, and it does not lower k between them unless the applications are rebuilt to share data through stable interfaces. The cloud is a centralised corner for the hosting layer and nothing for the others.

"Automation will remove the forum." Checkable standards do lower τ and the paper counts them as frontier-moving. The objection overreaches when it claims that reconciliation can be fully automated, because reconciliation is a decision about whether a local change should be allowed to affect others, and a decision that can be fully automated is one that has already been made and written as a rule. The forum does not disappear; it moves from deciding changes to deciding rules, which it does less often. That is the gain, and it is real, but a rule is a standard, and standards are the consistency instrument.

"We are small enough not to feel it." True, and the model says so: κ is large when n, r and k are small. An enterprise of three similar units with little shared data can hold all three goods near one, because its forum is idle. The trilemma is a property of scale, change and interdependence, and enterprises that do not have them do not have it. The objection becomes false at the point where the enterprise grows, acquires or diversifies, which is usually the point at which it writes its first operating-model document.

"Simple rules are enough." Eisenhardt and Sull (2001) argued that firms in fast markets compete on a few simple rules rather than elaborate processes. In the model, simple rules are checkable standards: few, stable and testable, with a low τ. The argument supports the frontier-moving case, and it has the same limit: the rules must be few and stable, and an enterprise that writes many and revises them often has rebuilt the forum.

14. Limitations and research agenda

This is theoretical research. Four limitations follow, each with a test.

First, the bound assumes a single reconciling forum with serial attention. Real enterprises reconcile through several channels, some parallel, and the bound for such an enterprise is looser than (3) by the degree of parallelism. The qualitative claim, that the product of the goods is bounded, survives; the constant does not. The test is to measure forum

throughput directly and compare observed lead times with the bound's prediction.

Second, the goods are defined so that they can be measured but have not been measured. Appendix A proposes measures; applying them to a sample of enterprises that have declared operating models, and comparing the declared corner with the measured one, is the first empirical study the paper invites. Our prediction is that most declared hybrids measure as one of the three corners, and that which corner is predicted by which instrument the enterprise's budget process actually empowers.

Third, the municipal reading of Section 9 is interpretive, and its prediction about Common Ground is stated so that it can fail. A measurable version: the time from a change in national law to its implementation across all municipalities, before and after the adoption of standard interfaces for the affected data, should fall in proportion to the share of municipalities and suppliers that have adopted them, and should fall little where the interfaces are still changing. The data exists in the implementation records of the association and the ministries.

Fourth, the case is constructed. A field study of one enterprise's re-cut from a declared hybrid to a layered design, measuring the goods before and after, would be worth more than the case and is the natural next step.

Beyond these, the model invites extension to the question of how corners should be chosen, which it does not answer. Lawrence and Lorsch's finding that the required integration rises with differentiation suggests that the choice of corner should follow the differentiation of the units' markets, and Weill and Ross's finding that top performers match archetype to decision suggests that it should follow the decision. Both are compatible with the model and neither is derived from it.

15. Conclusion

An IT operating model can give its business units autonomy, give the enterprise one landscape, or move at the speed of the market, and it can do any two of these at once. It cannot do all three, because every local change that touches a shared thing must be reconciled with the others, reconciliation takes time, and time is what speed is made of. The product of the three goods is bounded by a constant set by the number of units, their interdependence and the latency of reconciliation. The three coherent operating models are the three corners of that constraint, each with its own instruments and its own explicit sacrifice. The hybrid models that most enterprises declare sit in the interior, which does not exist, and they leave it by losing one good covertly: in a backlog, in a shadow landscape, or in a central squeeze.

We at Nashua have shown that the bound can be moved, by lowering interdependence through modular architecture, lowering latency through checkable standards, and choosing a corner per layer where the layers are decomposable, and we have read the Dutch municipalities' Common Ground movement as a frontier-moving programme whose strains the model predicts. The trilemma is not a counsel of despair. It is a discipline, the one Brewer drew from his own theorem: say which good you give up, where, and when. Enterprises that say it govern the landscape they have. Enterprises that do not govern the one in the document.

Appendix A. Measuring the goods and the bound

Autonomy (A). From the IT change portfolio of a period: the share of change spend whose initiating decision was taken within a business unit without a consent step outside it. Decisions taken locally but subject to a board are counted as local if the board's role is advisory and as central if it can refuse. Measured per unit and averaged by spend.

Consistency (C). Two components, averaged. (i) For the set of concepts the enterprise designates as shared, the share that has exactly one authoritative definition and one system of record in production; a concept with two definitions scores zero. (ii) For the platforms the enterprise designates as shared, the share of the relevant estate, by application count or spend, that runs on them. The designation is the enterprise's intent and must be recorded before measuring; C is consistency against intent, not uniformity.

Speed (S). From the change portfolio: the ratio of admissible changes delivered to production in a period to admissible changes requested in it, where admissible means within the operating model's declared scope. Equivalently, the ratio of the lead time the units state they need to the lead time observed, capped at one. Changes refused by design are excluded from both numerator and denominator and counted separately as the refusal rate, which is the centralised corner's own price.

Interdependence (k). From the integration inventory: for each application, the number of other units' applications that consume data or services from it or supply them to it; averaged over applications touched by local changes in the period.

Reconciliation latency (τ). From the forum's records: the mean elapsed time from a proposal's submission to its decision, per party reconciled with. A board that decides a proposal involving three other units in six weeks has τ of two weeks per reconciliation.

The bound. With n, r (changes requested per unit per period that touch shared things), k and τ measured, compute κ and compare $A \cdot C \cdot S$ with it. An enterprise whose product exceeds κ is either not in the forum-bound regime or is measuring a good generously; both are worth knowing.

Appendix B. Diagnosing the corner from symptoms

Where measurement is not yet possible, the following symptoms indicate which corner an enterprise actually occupies, whatever its document says.

Federated, slow. A board backlog that grows; a fast-track that carries most changes; roadmaps planned around board dates; complaints about governance from units that nonetheless comply.

Autonomous, inconsistent. Several definitions of a shared concept in production; a reference architecture that describes systems not in use; exceptions granted in bulk; software bought outside procurement; enterprise questions answered by projects.

Centralised standard. A single platform roadmap that units queue for; needs refused rather than delayed; units with budgets they cannot spend on systems; local workarounds in spreadsheets and bought tools around the standard.

The interior. More than one of the above in different units at once, and a review in progress.

Appendix C. The corners against existing typologies

The three corners can be mapped onto the two typologies most used in operating-model documents. The mapping is approximate, because the typologies are positions on a plane and the corners are constraints, but it lets a reader translate.

Corner	Ross et al. (2006) operating model	Weill & Ross (2004) archetype
Centralised standard	Unification (and replication for the standard's rollout)	Business monarchy or IT monarchy
Federated, slow	Coordination	Federal; duopoly
Autonomous, inconsistent	Diversification	Feudal; anarchy
The interior	Documents that name coordination and behave as diversification	"Federal" as usually declared

Ross, Weill and Robertson's replication model, standardised processes without integrated data, has no corner of its own: it is the centralised corner applied to the process layer and the autonomous corner applied to the data layer, which is a layered design in the terms of Section 8, and it is coherent exactly when those two layers are decomposable.

References

[1] Ashby, W. R. (1956). *An introduction to cybernetics*. Chapman & Hall. https://doi.org/10.5962/bhl.title.5851

[2] Baldwin, C. Y., & Clark, K. B. (2000). *Design rules: The power of modularity*. MIT Press. https://doi.org/10.7551/mitpress/2366.001.0001

[3] Brewer, E. (2012). CAP twelve years later: How the "rules" have changed. *Computer*, 45(2), 23–29. https://doi.org/10.1109/MC.2012.37

[4] Conway, M. E. (1968). How do committees invent? *Datamation*, 14(4), 28–31.

[5] Eisenhardt, K. M., & Sull, D. N. (2001). Strategy as simple rules. *Harvard Business Review*, 79(1), 106–116. https://hbr.org/2001/01/strategy-as-simple-rules

[6] Fleming, J. M. (1962). Domestic financial policies under fixed and under floating exchange rates. *IMF Staff Papers*, 9(3), 369–380. https://doi.org/10.2307/3866091

[7] Galbraith, J. R. (1973). *Designing complex organizations*. Addison-Wesley.

[8] Gilbert, S., & Lynch, N. (2002). Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services. *ACM SIGACT News*, 33(2), 51–59. https://doi.org/10.1145/564585.564601

[9] Lawrence, P. R., & Lorsch, J. W. (1967). Differentiation and integration in complex organizations. *Administrative Science Quarterly*, 12(1), 1–47. https://doi.org/10.2307/2391211

[10] Mintzberg, H. (1979). *The structuring of organizations: A synthesis of the research*. Prentice-Hall.

[11] Mundell, R. A. (1963). Capital mobility and stabilization policy under fixed and flexible exchange rates. *Canadian Journal of Economics and Political Science*, 29(4), 475–485. https://doi.org/10.2307/139336

[12] Parnas, D. L. (1972). On the criteria to be used in decomposing systems into modules. *Communications of the ACM*, 15(12), 1053–1058. https://doi.org/10.1145/361598.361623

[13] Rodrik, D. (2011). *The globalization paradox: Democracy and the future of the world economy*. W. W. Norton.

[14] Ross, J. W., Weill, P., & Robertson, D. C. (2006). *Enterprise architecture as strategy: Creating a foundation for business execution*. Harvard Business School Press. ISBN 978-1-59139-839-4.

[15] Sambamurthy, V., & Zmud, R. W. (1999). Arrangements for information technology governance: A theory of multiple contingencies. *MIS Quarterly*, 23(2), 261–290. https://doi.org/10.2307/249754

[16] Simon, H. A. (1962). The architecture of complexity. *Proceedings of the American Philosophical Society*, 106(6), 467–482.

[17] Skelton, M., & Pais, M. (2019). *Team topologies: Organizing business and technology teams for fast flow*. IT Revolution Press.

[18] Tweede Kamer der Staten-Generaal (2014). *Naar grip op ICT: Parlementair onderzoek naar ICT-projecten bij de overheid*. Eindrapport van de Tijdelijke commissie ICT-projecten bij de overheid (Commissie Elias). Kamerstukken II 2014/15, 33 326, nr. 5. https://zoek.officielebekendmakingen.nl/kst-33326-5.html

[19] Vereniging van Nederlandse Gemeenten / VNG Realisatie. *Common Ground*. https://commonground.nl

[20] Weill, P., & Ross, J. W. (2004). *IT governance: How top performers manage IT decision rights for superior results*. Harvard Business School Press. ISBN 978-1-59139-253-8.