Diverse Paths to a Common Governance Core:

Six OEMs and Four Platformers in the Mobility Sovereignty Transition

Akio TOKUDA*

Abstract

This paper applies the Platformer Defined Sovereignty (PDS) framework to empirical cases in the mobility industry. First, it analyses two Type 1 platformers — Tesla and BYD — showing that, despite shared classification, they operate fundamentally different sovereignty regimes; it then formalises the System-of-Systems Control Limit, a structural trade-off between temporal agility and spatial diversity. Second, it analyses six Type 3 OEMs (Toyota, Honda, Hyundai, Volkswagen, BMW, Mercedes-Benz) and shows that they converge on a strategic equivalence class through diverse paths determined by scale, diversity, and organisational capability. Third, it analyses two Type 2 platformers — Google and Huawei — developing the concept of time politics: the upper-order dimension governing when, in what order, and at what tempo the three branches of power operate. The paper closes with the Trust at Speed principle, which dissolves the speed-legitimacy trade-off through consortium-based architecture.

Keywords:

Platformer Defined Sovereignty; mobility industry; platform governance; SDV (Software Defined Vehicle); Trust at Speed

---

* Professor, College of Business Administration, Ritsumeikan University
Academic Visitor, Saïd Business School, University of Oxford

1. INTRODUCTION

This paper applies the analytical toolkit— the four governance mechanisms (G), four temporal disciplines (T), four governance handles (H), and three revenue functions (V) — to specific empirical cases in the mobility industry. G_1 (Apple-type vertical penetrating force), G_2 (Google-type horizontal expansive force), G_3 (Nokia-type collective binding force), G_4 (Samsung-type external negotiation force), and the structural-coupling index A_4 (governing the exponential integration-failure mechanism T_3 = p_eff ≈ p^A_4), as developed in the companion theoretical paper (Tokuda, 2026). As in that companion paper, the formulas presented below are structural propositions rather than econometric estimation models; they specify how variables relate under stated scope conditions, not calibrated numerical values to be estimated from data. The theoretical framework, which I call *Platformer Defined Sovereignty* (PDS), reorganises platform competition around the architecture of time: who sets the tempo of updates, who controls the cadence of integration, who governs the rhythm at which value flows back to its source.

The existing PDS framework identified four types of mobility platformer along two axes (OS governance holding × vertical-integration versus network). This paper takes the typology into empirical application across three sections.

Section 2 analyses the two Type 1 firms — Tesla and BYD — and demonstrates that, despite shared classification, the two firms operate fundamentally different sovereignty regimes. The section then extends the analysis into the System-of-Systems (SoS) domain, where mobility, energy, and urban infrastructure converge.

Section 3 analyses the six Type 3 OEMs — Toyota, Honda, Hyundai, Volkswagen, BMW, Mercedes-Benz. This section is the paper's central empirical contribution. The argument is that the six firms converge on a *strategic equivalence class* — a shared structural core within a *convergence band* that admits multiple stable configurations — through *diverse paths* determined by scale, diversity, and organisational capability.

Section 4 analyses the two Type 2 platformers — Google and Huawei — and develops the concept of *time politics* as the upper-order governance dimension above the three traditional branches of power.

Section 5 concludes with the *Trust at Speed* principle and its implications for the institutional architecture of mobility-industry governance.

Figure 1 maps the ten firms onto two axes. The upper half is further divided into infotainment and vehicle control domains — a boundary that distinguishes firms whose governance reaches into physical-safety architecture from those whose reach stops at the application layer. The six Type 3 OEMs in the lower right quadrant are the focus of Section 3: firms that do not currently hold OS governance but are actively developing proprietary OS capabilities in preparation for the full-scale SDV transition, each maintaining a vertically integrated strategy.

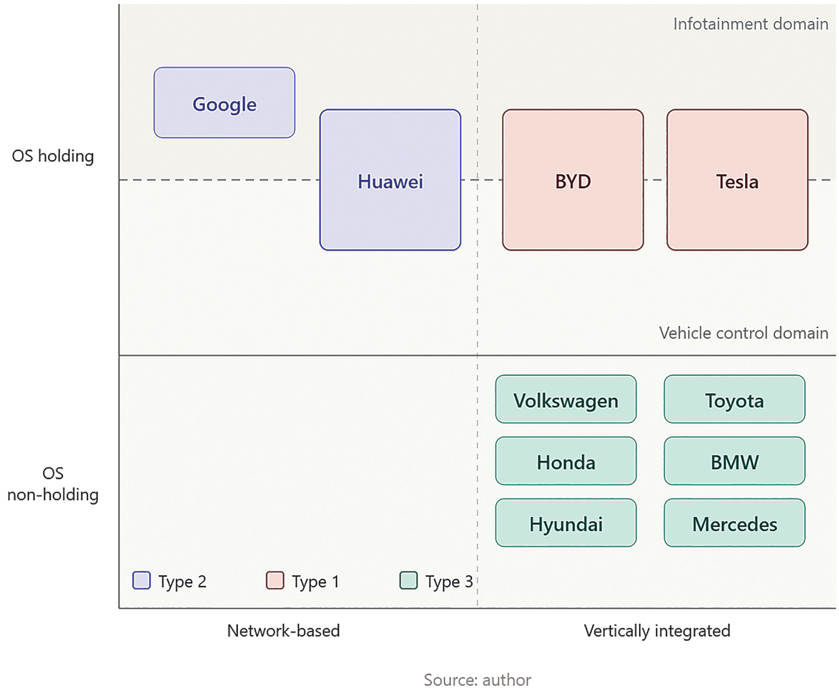

Figure 1. PDS Typology: OS Governance × Integration Model (Source: author)

2. TYPE 1 ANALYSED: TESLA, BYD, AND THE EXTENSION INTO SYSTEMS-OF-SYSTEMS

2.1 Setting the Frame

The foundational typology within the PDS framework placed both Tesla and BYD in Type 1 (OS holding × vertical integration) on the basis of two structural facts: both hold their own OS governance, and both operate vertically integrated stacks from cells (or batteries) through to user-facing services. The categorisation is correct at the level of the typology. Yet when the analytical toolkit of the PDS framework is applied to the internal architecture of each firm, the two diverge sharply.

Tesla's internal architecture is, in the framework's terms, a *closed-system continuous-flow model*. Tesla has transformed itself into a SaaS-type hardware operator, combining Apple's vertical-integration container with Google's continuous-update logic, operating both inside a single closed fleet. BYD's internal architecture is, by contrast, a *dual-mode platformer*. BYD maintains the Hard-Closed integration of Type 1 at the physical layer while simultaneously operating Soft-Open API release at the software layer.

The divergence demonstrates a property of the typology that the typology itself cannot show: *the same type can host fundamentally different sovereignty regimes*. The toolkit, not the typology, is what distinguishes them.

A methodological note before the case analysis proceeds. The framework's application to Tesla and BYD could be read as *after-the-fact rationalisation* — fitting observed differences into a sufficiently flexible vocabulary to produce the appearance of explanation. The paper takes this concern seriously and responds in two ways. *First*, the analysis is offered explicitly as *structural explanation of observed differences*, not as a derivation of those differences from first principles. The toolkit is applied to two firms whose differences were observable independently of the framework; the framework's contribution is to show that these differences are intelligible through specific combinations of G, T, H, and V — not arbitrary, but structured in ways the toolkit predicts. *Second*, each case analysis includes a small number of forward predictions whose subsequent observation (or non-observation) provides the framework's empirical accountability. For Tesla, the framework predicts that hardware refresh cycles will continue to lengthen (because S_Own × T_1(SW) revenue accumulates with longer hardware retention) and that retrofit investments will be increasingly rationalised as protecting the closed-fleet S_Own asset. For BYD, the framework predicts that structural lock-in (γ_BOM: formally defined in §2.3) on external OEM adopters will increase over time as e-Platform adoption deepens, and that BYD's negotiation position vis-à-vis those adopters will strengthen accordingly. These predictions are not claims the framework has confirmed; they are claims the framework commits to, with revisions to follow if the predictions fail.

2.2 Tesla: The Hybrid Architecture of SaaS-type Hardware

Tesla has been widely interpreted as the heir of Apple's vertical-integration model in the mobility industry. The framework, however, locates Tesla differently. Tesla is not simply a vertical Apple in cars; it is a hybrid in which the Apple structure houses Google-style continuous-update logic.

The pulse logic diverges from Apple. Model 3 launched in 2017 and remained essentially unchanged in body until the 2023 Highland refresh; Model Y similarly maintains a slow body-refresh cycle while the FSD computer migrated from HW3 to HW4 to HW4.5 over the same period. The pulse is in the brain, not in the body: T_1(HW) is deliberately long, while T_1(SW) is extremely fast and deep.

Closed-system synchronisation diverges from Google. Google's T_2 operates across a heterogeneous device base, with fragmentation (F) as the structural pressure. Tesla operates the same discipline in a *closed system*: the internal synchronisation rate s_in approaches 1.0 because Tesla controls every endpoint. Where Google accepts fragmentation as the cost of horizontal reach, Tesla eliminates it by limiting reach to its own fleet — and when retrofit is required, Tesla physically replaces the hardware (e.g., FSD computer replacements) to maintain software compatibility. Vertical integration permits a synchronisation discipline that horizontal-network platformers cannot achieve.

The Tesla revenue function is therefore a two-term formula:

$$V\_Tesla = [H \cdot T_1(HW)] + [k \cdot S\_Own \cdot T_1(SW) \cdot s\_in]$$

The first term is Apple-form: premium price × hardware update pulse. The second term is a specialized form of Google's V_2 applied to Tesla's closed fleet, where the internal synchronisation rate s_in approximates 1.0 because Tesla controls every endpoint and eliminates fragmentation internally.

The hybrid produces a counter-intuitive strategic implication: Tesla's profit is *not* maximised by frequent body refresh. It is maximised by long body retention combined with high-frequency brain updates. The longer a Tesla stays on the road, the larger S_Own grows; the more depth Tesla can push into T_1(SW), the larger the per-vehicle software revenue. The conventional Apple analogy implies annual model refresh; the Tesla data implies the opposite.

2.3 BYD: Dual-Mode Platformer

Where Tesla pulls Google-form synchronisation into a closed Apple-form container, BYD opens a Google-form network at one layer of its stack while maintaining Type-1 closure at the deeper layers.

The three-layer governance stack:

- IVI/HMI (DiLink): maximally open. Android-based with full app compatibility, SDK provision, broad OTA deployment.
- Middleware / by-wire APIs (D++): partially open. SDK and API access to vehicle sensors and selected control functions under graded permissions.
- e-Platform / BYD OS / battery integration (CTB): closed. Cell-to-Body battery integration, the 8-in-1 electric drive, and BYD OS are held internally and licensed selectively to other manufacturers.

This is a Type-1 firm exhibiting Type-2 behaviour at the IVI layer while preserving Type-1 closure at the platform layer. The framework calls this a *dual-mode* configuration.

The redefinition of T_2. To capture this configuration analytically, the framework introduces a refinement of T_2 that decomposes synchronisation rate into two factors:

$$T_2 = o\_ext \times \iota\_struct$$

where o_ext is external effective openness (breadth of external supply and adoption) and ι_struct is structural coupling (depth and irreversibility of integration). A pure vertical integrator has o_ext ≈ 0; a pure horizontal platformer has high o_ext but variable ι_struct. BYD's distinctive position is that *both* are high: e-Platform is offered to external OEMs (high o_ext), and once an OEM adopts e-Platform 3.0, more than 40% of vehicle BOM is committed to BYD's architecture (γ _BOM > 0.4, high ι _struct). The combination produces:

$$T_2(BYD) >> T_2(Type\ 1\ benchmark)$$

This is the formal expression of why BYD, despite Type-1 classification, exercises platformer-like governance over its adopters: structural-coupling generates irreversibility that purely horizontal platformers cannot achieve, while openness generates the reach that purely vertical integrators forgo.

The BYD revenue function:

$$V\_BYD = H\_BYD \cdot T_1(HW) + m\_BYD \cdot S\hat{}(ext) \cdot (T_2(HW) \cdot \iota\_struct + T_2(SW) \cdot o\_ext)$$

The first term is hardware sales revenue with rapid update tempo. The second term captures dual-flow value: hardware-layer revenue scaled by structural coupling, plus software-layer revenue scaled by external openness. BYD has approximately halved new-vehicle development lead time (to roughly 18 months) by exploiting this configuration.

2.4 The SoS Extension: Three-Layer Value Structure

Both Tesla and BYD are visibly extending their governance regimes into adjacent infrastructure — Tesla into electricity grids and virtual power plants, BYD into public transport, charging networks, and smart-city architectures. The framework formalises platformer value in the SoS extension as a three-term sum:

$$V\_i = P\_HW \cdot Q\_HW \cdot \sigma_1 + M\_SW \cdot S\_Veh \cdot \sigma_2 + R\_E \cdot N\_node \cdot \sigma_3$$

The first term is vehicle flow (hardware sales). The second term is vehicle stock (post-sale software services across the installed base). The third term — the new element — is SoS stock: value created at the boundary between the platformer's vehicle ecosystem and surrounding social infrastructure (charging networks, grid operators, urban OS, insurance, payment systems). N_node is the boundary count; R_E is the unit-event value at the boundary; σ_3 is the reach coefficient at which the platformer's value penetrates the boundary.

The bottleneck principle. The framework's most consequential SoS finding is the decomposition of σ_3 into three independent gates:

$$\sigma_3 = min\{\sigma\_c,\ \sigma\_m,\ \sigma\_l\}$$

where σ_c is the control gate (technical pass-through), σ_m is the market gate (commercial pass-through), and σ_l is the legitimacy gate (political pass-through). *SoS value is governed by its weakest link.* No matter how high a platformer's technical capacity, no matter how favourable its market position, if its legitimacy gate is closed by regulation or public concern, the value of its SoS extension collapses multiplicatively, not additively. This is the formal statement of why SoS expansion is not a technical question but a governance question.

2.5 The SoS Control Limit

The section closes with a structural constraint:

Time Agility × Spatial Diversity ≤ Control Limit

A platformer choosing high temporal agility (rapid pulse, frequent updates, real-time intervention) cannot simultaneously achieve high spatial diversity (broad coverage across heterogeneous adopters). Tesla operates at high temporal agility and restricts spatial coverage to its own fleet plus tightly integrated home-energy customers. BYD operates at low temporal agility and extends spatial coverage across heterogeneous adopters including public transport and policy-driven urban deployments.

The constraint is not a normative ceiling; it is a structural reality with two specific sources. *First*, the underlying PDS integration-failure mechanism defined as T_3 = p_eff ≈ p^(A_4): as spatial diversity rises, the structural-coupling index A_4 grows, and small drops in per-module reliability p compound exponentially through the system. High temporal agility — frequent updates — increases the rate at which p drops occur, multiplying the catastrophic-failure probability that diversity already amplifies. *Second*, the heterogeneity-driven test-cost explosion: every new adopter category (a new vehicle platform, a new region, a new use case) multiplies the validation surface that any single update must traverse before deployment, and validation cost scales super-linearly with configuration count. The product form of the constraint — agility *times* diversity, not the sum — reflects these two sources operating jointly: each amplifies the other, and neither can be relaxed without compromising operational safety. The constraint explains why no SoS platformer can be both Tesla and BYD at once, and why the choice between high-agility/low-diversity and low-agility/high-diversity is the strategic fork that defines SoS positioning.

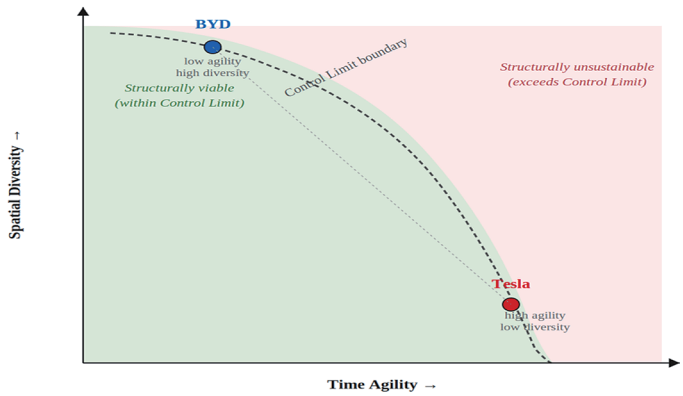

Figure 2. The SoS Control Limit: Time Agility×Spatial Diversity < Control Limit (Source: author)

3. Type 3 Analysed: Six OEMs and the Diverse Paths to a Common Governance Core

3.1 Vehicle OS as Governance Bundle

This section is the paper's central empirical contribution. The six firms — Toyota, Honda, Hyundai, Volkswagen, BMW, Mercedes-Benz — are all classified as Type 3 (OS non-holding × vertical integration). The argument is that *there is a shared structural core* common to all six firms — what § 3.5 develops as a *convergence band* or *strategic equivalence class* — and that the six firms approach this shared core through *diverse paths* (multiple stable implementations within the equivalence class) determined by physical and organisational constraints.

When industry discourse asks "which OEM will win the SDV race", it almost always answers in terms of software products: Will Toyota's Arene succeed? Will VW's CARIAD recover? Will BMW's OS 9 prove superior to Mercedes' MB.OS? These are not the right questions. They displace the deeper structural question onto the surface technical question.

The framework defines *Vehicle OS* not as a software product but as a *governance bundle* — the durable assignment of authority over five functions (update, integration, distribution, certification, capture) across the three layers of vehicle architecture.[1] Following Hart's (1995) incomplete-contracts framework, the governance bundle is in essence the location of *residual control rights*: in a world where every interaction between the vehicle and the platformer cannot be specified in advance by contract, who holds the final-call authority when something unanticipated happens?

The vehicle architecture is layered:

- *Layer 3*: control / real-time (ECU, RTOS). Safety-critical, deterministic, where software freedom is most constrained.
- *Layer 2*: adaptive / middleware (AUTOSAR Adaptive, DDS, middleware). The integration layer.
- *Layer 1*: UX / application (IVI, apps). The value-capture entry point.

The contest is decided in Layer 2. Layer 3 is too safety-bound to be the site of strategic differentiation; Layer 1 is too surface to determine sovereignty. T_3 (integration success rate) is the central temporal discipline; T_1, T_2, T_4 enter the system only by passing through Layer 2.

3.2 Why Vehicle OS Quality Is Observed Through Failures

A short observation that frames the entire analysis: Vehicle OS, when working, is invisible; when failing, it is everywhere visible. Three features distinguish vehicle OS from consumer-OS:

1. *Success is silence; failure is collapse.* The diagnostic of a well-governed vehicle OS is the absence of incidents at Layer 2.
2. *Failure scales exponentially through A_4.* A single boundary-condition failure at Layer 2 can propagate across a fleet of millions; the

1) The five functions describe what is done; the four handles (H_a: certification; H_b: distribution/update; H_c: runtime control; H_d: capture) describe where authority is held. Integration operates as an overarching design constraint rather than mapping to a single handle.

structural-coupling index determines whether propagation is contained or cascading.
3. *Success is unrewarded; failure is politicised.* OEMs are not incentivised to over-invest in invisible governance quality, yet the cost of under-investing is catastrophic.

The strategic question facing every Type 3 firm is therefore not “how can we win?” but “how can we govern Layer 2 such that failures we have not yet seen are contained when they occur?”

3.3 Six Firms, Four T-Variables

Volkswagen — external time-shortening through external injection. After publicly acknowledged delays in its proprietary SSP, VW announced a joint venture with Rivian to import next-generation SDV foundation. T_4 is being acquired through *discrete external injection* — not Samsung-style differential shortening of τ_self within an existing platform, but wholesale importation of a new “fast circuit” that creates $\Delta\tau$ by replacing τ_self itself.[2)]

Honda — three-clock parallelism. Honda runs three parallel tempos: ASIMO OS (proprietary stack, T_1 internally), Google built-in (AAOS, T_2 externally), and the Sony collaboration on Afeela (G_3 collective binding). The strategic question is whether Layer 2 can host three tempos without one eroding the others.[3)]

Hyundai — two-stage external injection. Hyundai proceeds through Google Maps Platform (Places API) first, then full AAOS / Pleos adoption. The two stages compound at Layer 2 — the second injection arrives before the first is fully integrated — and brand differentiation across Hyundai’s portfolio adds variance.[4)]

2) https://www.autoblog.com/features/software-issues-at-volkswagen-delaying-the-next-gen-ssp-platform (accessed 25 May 2026); https://rivian.com/newsroom/article/rivian-and-volkswagen-group-announce-plans-for-joint-venture (accessed 25 May 2026).
3) https://global.honda/en/newsroom/news/2021/c210924eng.html (accessed 25 May 2026); https://global.honda/en/cars-apps/ (accessed 25 May 2026).
4) https://www.hyundai.com/worldwide/en/newsroom/detail/hyundai-motor-group-and-google-collaborate-on-software-capability-for-future-mobility-innovation-0000000883 (accessed 25 May 2026); https://9to5google.com/2024/12/12/google-maps-is-coming-to-kia-and-hyundai-vehicles-ahead-of-android-automotive/ (accessed 25 May 2026).

Toyota — process internalisation through Arene. Arene is presented not as a competitive software product but as a *process discipline framework* — virtualisation, automated testing, reuse libraries, iterative improvement — through which T_3 is held high through engineering process rather than architectural simplification. Toyota holds tempo sovereignty internally and uses process repetition to suppress the integration risk that scale (~10 million vehicles per year) and diversity (full product portfolio) would otherwise generate.[5)]

BMW — isolation through bounded layer-management. BMW's OS 9 will be AOSP-based with the Shared Service Layer (SSL) maintained on Linux, with OTA operations controlled at the SSL level. The configuration imports external infrastructure while retaining update tempo inside an operational layer BMW controls.[6)]

Mercedes-Benz — whole-domain integration through MB.OS. MB.OS is presented as a chip-to-cloud purpose-specific architecture with whole-domain access, coupled with announced Google function and map integration. The configuration is the inverse of BMW's: rather than maintain boundaries and isolate, Mercedes attempts to integrate the whole domain under a single operational architecture.[7)]

3.4 The Two-Axis Mapping

The six firms are mapped onto a two-dimensional strategic space:

- *Horizontal axis*: Internal (T_1/T_2) ↔ External (T_3/T_4)
- *Vertical axis*: Process-embedded ↔ Isolation-ops

A note on the choice of axes. These two axes are not arbitrary; they correspond to two distinct dimensions that the strategic-management literature has long treated as fundamental but rarely combined. The hori-

5) https://woven.toyota/en/arene/ (accessed 25 May 2026); https://mobilitech.jp/?p=6134 (accessed 25 May 2026).

6) https://www.press.bmwgroup.com/global/article/detail/T0401875EN/bmw-group-to-build-on-android-open-source-project-aosp-for-%E2%80%9Cbmw-operating-system-9%E2%80%9D (accessed 25 May 2026); https://9to5google.com/2023/02/bmw-android-automotive-no-google-play/ (accessed 25 May 2026).

7) https://group.mercedes-benz.com/careers/about-us/mercedes-benz-operating-system/ (accessed 25 May 2026); https://mercedes-benz-media.co.uk/assets/documents/original/41930-EMBARGOEDMercedesBenzPressInformationStrategyUpdateMercedesBenzOperating-System.pdf (accessed 25 May 2026).

zontal axis (internal vs external tempo) is, in the framework's vocabulary, an operationalisation of the boundary-of-control question (Hart, 1995; Ulrich, 1995) — where the firm draws the line between activities it governs internally and activities it accepts from outside. The vertical axis (process-embedded vs isolation-ops) is an operationalisation of the mode-of-managing-complexity question (Henderson & Clark, 1990; Fujimoto, 1997) — whether the firm absorbs complexity through engineering process discipline or contains it through architectural isolation. The framework's contribution is not to introduce these two dimensions for the first time but to cross them in a way that the existing literature, treating product-architecture and governance-boundary questions separately, has not. The four quadrants that result are, in this sense, a re-organisation of existing theoretical resources around the empirical question of Layer 2 Vehicle OS governance, not a free-standing invention.

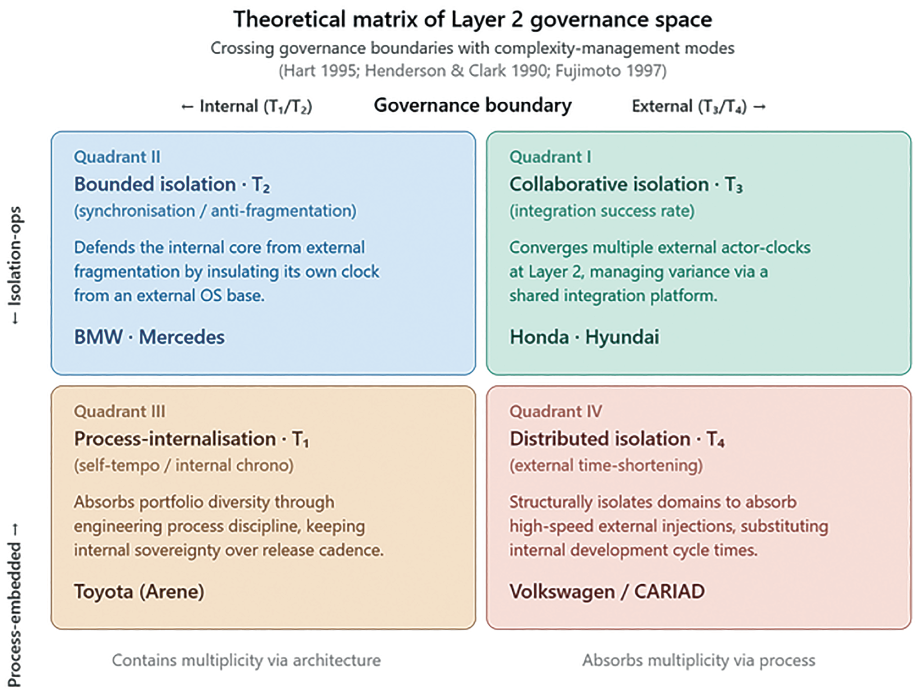

Figure 3. OEM Mapping: Internal vs External × Process-embedded vs Isolation-ops (Source: author)

Figure 3 locates the six firms within the resulting quadrants, each quadrant being governed by a distinct temporal discipline. Quadrant III (Process-internalisation, internal × process-embedded) is governed by T_1, the discipline of self-tempo: the firm absorbs portfolio diversity through engineering process discipline while retaining internal sovereignty over the hardware-software release cadence. This is the position of Toyota's Arene. Quadrant II (Bounded isolation, internal × isolation-ops) is governed by T_2, the discipline of synchronisation against fragmentation: the firm defends its internal core by insulating its own clock from an external OS base. Quadrant I (Collaborative isolation, external × isolation-ops) is governed by T_3, the integration success rate: the firm converges multiple external actor-clocks at Layer 2 and manages continuous variance through a shared integration platform — the position toward which both Honda and Hyundai gravitate. Quadrant IV (Distributed isolation, external × process-embedded) is governed by T_4, the discipline of external time-shortening: the firm structurally isolates domains to absorb high-speed external injections, substituting external cycle times for its own internal development. This is the position of Volkswagen's CARIAD.

Two features of this mapping deserve emphasis. First, the dominant temporal discipline (T_1 through T_4) is not a free choice but a structural consequence of where the firm sits on the two axes; the quadrant a firm occupies determines which clock it must master to survive. Second, firms sharing a quadrant are not therefore identical. BMW and Mercedes-Benz both occupy Quadrant II, yet their isolation is achieved by opposite means: BMW through procedural isolation — drawing an explicit boundary and absorbing external influence through the disciplined process of staged delivery and standardised rollback — and Mercedes through structural isolation, the integral MB.OS architecture swallowing external influence within a single whole-domain design. It is no accident that BMW, though located in the isolation quadrant, plots near the boundary with process-embedded territory: its isolation is sustained by a markedly Toyota-like engineering process. The matrix therefore locates the governance regime; the diversity of implementation within each regime — and the deeper paradox that Toyota and Mercedes, both masters of integral hardware capability, translate that capability into opposite SDV regimes — is the empirical regularity taken up in § 3.5 and § 3.7.

3.5 The Convergence Band: An Equivalence Class, Not a Single Configuration

A clarification on the analytical status of the claim that the six firms converge on a “common governance core” is essential before the empirical analysis proceeds. The earlier formulation, which I had used in working drafts, was too strong. A more accurate formulation, and one that the empirical evidence supports, is that the six firms converge on a *convergence band* — a region of the strategic space defined by a shared structural core, within which multiple stable configurations coexist.

The structural core, which is genuinely shared across all six firms:

Deep retention of residual control rights at the lower layers (Layer 1 and Layer 2), combined with surface boundary management at Layer 3 to accommodate external flexibility.

This is the *equivalence class* — the analytical category to which all six firms belong. The deeper layers are where residual control rights matter most — physical safety, product liability, operational continuity. No Type 3 firm can survive by surrendering control of these layers to an external platformer. The surface layer is where ecosystem dynamics, app development, and user-experience innovation are decided — and here, full closure suffocates the network effects on which competitive renewal depends. The strategic core is necessarily two-sided: closed below, open above, with a carefully designed boundary in between.

But within this equivalence class, the *implementation* admits multiple stable configurations. Layer 2 — the integration layer that is the contested zone for all Type 3 firms — can be governed through different combinations of internal and external time, and through different combinations of process discipline and isolation discipline. Toyota’s process-internalisation and Mercedes’ whole-domain integration are *not the same configuration*. They are different stable solutions within the same equivalence class — each viable under specific scale, diversity, and organisational-capability conditions. The framework’s claim is therefore narrower and stronger than a single-position claim:

The structural core (deep control + surface flexibility) is invariant across viable Type 3 configurations. The implementation within Layer 2 admits multiple equilibria, each conditioned by firm-specific physical and organisational constraints.

This formulation makes the empirical claim testable. A Type 3 firm that abandons either side of the structural core — either by surrendering Layer 1/Layer 2 control to a Type 2 platformer, or by closing Layer 3 to external ecosystems — falls outside the convergence band and, the framework predicts, becomes structurally unsustainable. A Type 3 firm that retains both sides of the structural core but selects a Layer 2 implementation poorly matched to its constraints — VW's CARIAD attempting Mercedes-style whole-domain integration at Toyota-scale diversity — produces the integration delays that empirical observation now confirms. The convergence band is the set of viable configurations; the implementation diversity within the band is the empirical regularity that the framework's multi-dimensional matrix (§ 3.4) makes visible.[8)]

3.6 Path Selection as Constraint, Not Choice

The deeper analytical claim is that the diverse paths are not, in any straightforward sense, *chosen* by the six firms. They are *constrained* — determined by the firms' physical and organisational conditions. Three constraints matter:

Scale (vehicle volume). Toyota produces approximately 10 million vehicles per year across the full segment portfolio. Mercedes produces a much smaller volume concentrated in premium segments. The same architectural choice has fundamentally different consequences. Mercedes can sustain whole-domain integration because the volume × diversity product is manageable; Toyota cannot.

Diversity (product portfolio breadth). The structural coupling A_4 grows with portfolio diversity, and the temporal cost of integration (Time $\propto$ 1 / p_eff, where p_eff ≈ p^A_4) compounds non-linearly. Toyota's process-internalisation path is the only feasible response to this combination; Mer-

8) The five functions are operational variables describing governance acts; T_3 and p_eff are outcome variables registering results under a given coupling structure (A_4). The framework maintains this separation to prevent conflating process and outcome in governance failure diagnosis.

cedes can sustain a different path because A_4 is structurally lower.

Organisational capability. Toyota's process-engineering tradition supports the process-internalisation approach. BMW, despite operating modular product architecture, has implemented CI/CD discipline that imports Toyota-style process governance into its software domain. Mercedes' integral product-architecture tradition supports the whole-domain integration approach. VW's attempt to combine Mercedes-style whole-domain integration (through CARIAD) with Toyota-scale diversity has, predictably from this framework, produced the integration delays publicly reported in 2024-2025.[9)]

3.7 The Toyota–Mercedes Inverse Translation

The most theoretically distinctive finding concerns Toyota and Mercedes. Both hold strong integral-type (*suriawase*) product-architecture capability in hardware — a tradition the production-management literature has analysed for decades (Fujimoto, 1997). The framework asks: how does this integral-type capability translate into SDV-era governance? The answer is *in opposite directions*.

- Toyota translates integral-type capability into *process internalisation*: the multiplicity of the portfolio is absorbed not by structural unification but by engineering process repetition, virtualisation, and reuse. The product architecture remains diverse; the process architecture is integral.
- Mercedes translates integral-type capability into *whole-domain integration*: a single MB.OS architecture covering chip to cloud, absorbing multiplicity through structural unification. The product architecture becomes integral at the system level.

These are *opposite directions of translation* from the same underlying organisational capability, conditioned by the scale and diversity each firm can sustain.

9) https://www.electrive.com/2025/10/06/vw-reassigns-cariad-as-coordinator-of-rivian-and-xpeng-software/ (accessed 25 May 2026); https://cariad.technology/de/en/news/stories/cariads-sdv-journey-and-ci-cd-pipeline.html (accessed 25 May 2026).

3.8 The Prescriptive Core: Combining Depth and Surface

The framework's prescriptive claim is direct and structurally distinctive. The future of Type 3 strategy lies not in any one firm's path but in the *combination* of approaches that the six firms together demonstrate:

> *Hold residual control rights tightly through process internalisation at Layer 1 and Layer 2 (the Toyota pattern). Manage surface boundaries through clean isolation at Layer 3 to allow external innovation (the BMW pattern). Together, these two disciplines constitute the strategic position toward which all six firms are converging.*

The Toyota pattern alone would suffocate the network effects that Layer 3 ecosystems require. The BMW pattern alone would not protect residual control rights at the depth required for physical-safety obligations. The combination is what the framework proposes — and it is what the six firms, in their diverse paths, are jointly demonstrating as the realistic outline of a Type 3 future.

All six firms are, in different ways and to different degrees, currently struggling with the integration discipline that Layer 2 governance demands — Volkswagen's CARIAD delays being the most publicly visible case. The framework's claim is that *the path to a defensible Type 3 position remains open for each of the six firms*, but only on two conditions: each must follow the path its physical conditions permit, and the industry as a whole must develop the consortium infrastructure that would allow their diverse paths to coexist. The opposite outcome — fragmentation, partial failure, the entry of a Type 2 platformer capturing the Layer 2 governance that no individual OEM has consolidated — is also possible.

4. Type 2 Analysed: Google, Huawei, and the Politics of Time

4.1 The CDA 230 Question

Section 230 of the U.S. Communications Decency Act (CDA 230), enacted in 1996, established the immunity regime that has underpinned the consumer-information platform economy for nearly three decades. Interactive computer service providers are not held liable as publishers for content posted by third parties. This immunity has been the structural foundation on which social media, search, and mobile advertising platforms have built their business models.

The mobility industry's encounter with platformer governance raises a structural question: *can the CDA 230 immunity regime extend to a vehicle operating system?* The vehicle is not a content-delivery medium. It is a physical artefact whose operation directly affects human life. The proposition that a platformer providing the OS that determines vehicle behaviour is not legally responsible for the consequences of that behaviour is, at minimum, in tension with the product-liability traditions of the automotive industry.

Three legal developments make this tension increasingly visible:

1. *Lemmon v. Snap Inc.* (2021), in which a feature awarding badges for speed achievements was held to be product design, not protected expression, and CDA 230 was held not to apply. The legal reasoning extends to vehicle OS design.
2. The EU New Product Liability Directive (2024), which explicitly defines software, operating systems, and AI components as products and includes cyber-vulnerabilities and inadequate updates as defects.
3. Narihara's (2019, 2023) *right to hack* arguments: API specifications and technical standards effectively constitute social norms, and OEM resistance to platformer-imposed specifications is not a breach of contract but an act of restoring the legitimacy of governance.

The CDA 230 question is the structural question that determines whether platformer governance over vehicle OS is *governance with responsibility* or *governance without responsibility*. The framework argues that the latter configuration is structurally incompatible with the mobility industry's physical-safety constraints.

4.2 Google: Governance Without Responsibility

Google's Vehicle OS strategy is structured around two technical artefacts that, in combination, produce governance without responsibility at scale.

Spatial separation: VHAL. The Vehicle Hardware Abstraction Layer is the architectural boundary at which Android Automotive OS (AAOS) meets the physical layer of the vehicle. Above the VHAL: Google-provided OS framework, applications, services. Below the VHAL: OEM-provided

Board Support Package, Linux kernel, hardware drivers. The technical purpose is integration; the governance purpose is the placement of physical responsibility outside Google's architectural domain.

Temporal separation: Project Treble. Treble decouples the Android framework from the underlying hardware abstraction through a stable vendor interface (HIDL / AIDL). The technical purpose is to remove the OEM-side bottleneck on OS updates. The governance purpose is the deliberate separation of two clocks. Cyber space — Google's domain — operates on a CI/CD cycle measured in days and weeks. Physical space — the OEM's domain — operates on safety-validated cycles measured in years. Treble allows each to run independently, with the strategic implication that *Google monetises the fast clock while the OEM carries the obligations of the slow clock.*

This is the operational architecture of governance without responsibility. The VHAL places responsibility outside Google's domain spatially. Treble places it outside Google's domain temporally.

The 2024 antitrust ruling. Google's automotive strategy is operating against a destabilising background. The U.S. Department of Justice antitrust ruling in 2024 against Google's default-search agreements has forced reconsideration of the business model. The default-purchase model that paid OEMs to maintain default-position privileges is being shut down. Google's response has been to shift capital from "rent on default position" to "ownership of physical infrastructure": Alphabet's projected 2026 capex of approximately $180 billion, nearly double the prior year, supports the transition from a *cyber platformer* to a *cyber-physical (CP) platformer*.

4.3 Huawei: Whole-Stack Internalisation

Huawei occupies the same typological cell as Google but pursues a structurally opposite architecture. Where Google places physical responsibility outside its domain through the VHAL, Huawei pulls the physical layer into its domain through the Device Abstraction Layer (DAL) — a service-oriented architecture in which sensors and actuators appear to the OS as plug-and-play peripherals managed by Huawei's stack.

The Huawei architecture extends across HarmonyOS (OS layer), Huawei ADS (advanced driving support), DriveONE (electric drivetrain), and Qiankun Cloud (vehicle cloud service). OEM partners participate at various depths through three engagement modes (Tier-1 supplier, Huawei Inside, HIMA brand alliance).

The commoditisation of the vehicle. Huawei's whole-stack internalisation has a structural consequence that the framework predicts: under sustained operation of this configuration, the vehicle is likely to be commoditised, particularly for OEMs adopting Huawei's stack at the HIMA brand-alliance depth. Where Google's architecture leaves the OEM with physical responsibility but no platform sovereignty, Huawei's architecture, if its current trajectory continues, would leave the OEM with neither — the OEM tending toward the role of a body-and-actuator supplier within Huawei's governance regime. The empirical evidence to date is consistent with this prediction but does not yet constitute confirmation; the prediction is offered as a structural expectation that the framework's analytical instruments generate, open to revision as the configuration matures.

The state-platformer dimension. Huawei's strategy is most coherently read within the context of its state-supported character. The combination of full-stack internalisation, OEM brand subordination, and the rapid pace at which HIMA-affiliated OEMs are accepting Huawei's stack is difficult to explain through market dynamics alone — though I would note that this is an interpretive claim, not an empirically demonstrated causal one. The framework offers the interpretation that the configuration is best understood as a state-supported industrial-policy initiative in which the boundary between commercial strategy and national-economic strategy is, by design, less sharply drawn than in market-economy configurations. This interpretation has predictive content: it expects that Huawei's strategic choices in the mobility industry will be more closely coordinated with Chinese industrial-policy timing than would be expected from a comparable commercial-only firm, and that the configuration's robustness to commercial reversals will be higher than market-only theory would predict. These expectations are offered as testable predictions of the framework, not as established empirical regularities.

The two Type 2 firms operate under fundamentally different governance regimes — one market-and-judicial, the other state-and-political — and exercise platformer governance through architectures reflecting those regimes.

4.4 OTA Reframed: "The Technology of Fast Return"

A central reframing deserves direct registration. In conventional industry discourse, Over-The-Air updates (OTA) are framed as the technology that enables continuous improvement — the "fast forward" of the SDV era. The framework argues this framing misses the structural significance.

In a system that controls safety-critical functions and is subject to product-liability law, the value of OTA is not the speed of forward improvement. It is *the speed of return*. OTA is the technology of fast rollback — the capacity to identify a defect, isolate the affected scope, and return affected vehicles to a previous safe state before the defect has propagated to catastrophic outcomes.

The reframing has consequences. Type 2 platformers, in their consumer-information histories, have not had to internalise this discipline at the depth that mobility now requires. Google's update regimes for Android (in consumer space) operate on the premise that occasional defects are absorbed by users. Mobility cannot operate on that premise.

4.5 Time Politics: The Upper-Order Governance Dimension

This is the framework's theoretical climax. The conventional three-powers framework (executive, judicial, legislative) treats power as a *content* question: who decides, who adjudicates, who legislates. The framework argues that, in the platformer-economy context, this is necessary but not sufficient. Power is exercised not only through the content of decisions but through the *temporal coordination* of decisions: when a decision is made, in what order, at what tempo, with what allocation of priority across competing demands.

Three operations constitute *time politics*:

- *Preemption*: the operation of relative-order manipulation. Interrupting an emerging consensus, advancing a decision before its consequences are visible, deferring a decision past the window in which it could be acted on. *Operational observation*: OTA rollout sequencing — which OEMs receive a security patch first, which regions receive a feature rollback last, which vehicle generations are sequenced ahead of others. The platformer who controls this sequencing exercises preemption in operational form.

- *Allocation*: the distribution of agenda priority. Determining which issues are advanced when, how many cycles are devoted to which topics, what gets refreshed and what is allowed to grow stale. *Operational observation*: update-window allocation — the proportion of release cycles devoted to security versus feature versus regulatory-compliance updates, the agenda priority given to safety-critical APIs versus user-facing applications. The platformer who controls this allocation determines which capabilities of the platform evolve and which atrophy.
- *Tempo*: the regulation of absolute periodicity. Setting common gating dates, distribution windows, rotation cycles for cryptographic keys, emergency slot allocation. *Operational observation*: release cadence — the periodicity of major OS versions, the rotation cycle of signing keys, the calendar slots reserved for emergency security patches. The platformer who controls this cadence sets the clock against which all other industry participants must align their own operations.

Each is a meta-operation that determines how the three branches operate. The branch that gets to preempt others advances its agenda; the branch that controls allocation determines which decisions ripen; the branch that sets tempo determines whose timescale the others must conform to. In conventional governance — where the three branches operate on roughly comparable time scales — time politics is implicit. In platformer-governed industries — where one branch's tempo (the platformer's CI/CD cycle) is orders of magnitude faster than another's (the regulator's review cycle) — time politics becomes explicit and decisive.

The framework's central claim: *the firm or institution that controls the time-political infrastructure of an industry controls the industry's governance, regardless of which branch holds nominal authority over content.*

4.6 Effective Governance and the Five Functional Clusters

The framework operationalises time politics through a function:

G_eff = w_1 · (Stop right) + w_2 · (Conformance adjudication) + w_3 · (Signing and distribution authority) + w_4 · (Evidence)

The four components correspond to: stop right (executive, immediate isolation — H_c); conformance adjudication (judicial, final compliance determination — H_d); signing and distribution authority (legislative, key holdership and release control — H_a × H_b); and evidence (the cross-cutting infrastructure that legitimates all three — H_d's record). The weights w_i represent the operational exercisability of each component.

G_eff is realised through five functional clusters that constitute the operational infrastructure of effective governance:

1. **Evidence cluster** — SBOM, configuration records, telemetry, test results, signatures gathered on a common, tamper-resistant timeline. The factual bedrock on which judicial determination rests.
2. **Proof cluster** — Machine-readable compliance rules operating continuously through CI/CD gates.
3. **Key cluster** — PKI and Root-of-Trust jointly operated; release authority for any code is the prerogative of joint custodians.
4. **Execution cluster** — Standardised isolation and rollback APIs; specified maximum times from detection to halt.
5. **Time cluster** — Industry-wide design of update rhythm, deployment windows, sequencing, rotation cycles. Coordinator of internal lead times (τ _self) and external release velocities (v_release).

The first four clusters correspond to the three traditional powers: evidence and proof together constitute the judicial foundation; the key cluster operates legislative authority over distribution; the execution cluster operates executive authority over physical interruption. The fifth cluster — *time* — is the time-political infrastructure: the determination of when, in what order, and at what tempo the four operational clusters are activated. Whichever institution controls the time cluster controls the effective governance of the industry, regardless of which actor holds nominal authority over content within any single power.

4.7 Standardisation Consortia as Operational Infrastructure

The framework distinguishes three stages of consortium maturity:

- *Stage 1*: Document-producing. Reference specifications, technical guidance, vocabulary harmonisation. AUTOSAR Classic in its founding decade operated at this stage.
- *Stage 2*: Process-embedded. Conformance testing, common toolchains, certification authority.
- *Stage 3*: Operational-infrastructure. A consortium that operates the three powers as runtime infrastructure: continuously running compliance pipelines, jointly held cryptographic keys, emergency-isolation APIs, time-political coordination.

The framework's prescriptive proposal is to develop consortia toward the operational-infrastructure form. The Uranos Ecosystem (the Japanese DFFT-aligned data governance initiative) and Catena-X (the EU AI Act / GDPR-aligned data space) are identified as initial — partial, exploratory — moves in this direction. Consortia of this form are the institutional form through which the Type 2 platformers' governance-without-responsibility configuration can be checked: not through individual-firm resistance, but through a consortium that embodies the three powers as common infrastructure.

5. Conclusion: Trust at Speed

The framework's empirical analysis closes with what it calls the *Trust at Speed* principle: the requirement that speed and legitimacy be made compatible rather than treated as competing demands. Conventional platformer logic optimises for speed at the cost of legitimacy (CDA 230 immunity is the regulatory expression of this trade-off). Conventional regulatory logic optimises for legitimacy at the cost of speed (the regulatory cycle is structurally slower than the platformer cycle). The framework's proposal is that neither side of this trade-off is acceptable in the mobility industry, and that a consortium operating the five functional clusters together is the institutional form through which the trade-off can be dissolved.

A precise definition is needed before this principle leaves the conclusion. Trust at Speed is not a slogan; it has a structural specification. Operationally, Trust at Speed = the implementation state of the Five Functional Clusters (Evidence, Proof, Key, Execution, Time) within the Three-Loop Model (Polity / Alliance / Machine) as established in the foundational structural framework. When the five clusters are operated as runtime infrastructure by a consortium that embeds Polity-in-the-Loop accountability through Alliance-in-the-Loop coordination, the speed-legitimacy trade-off is structurally dissolved because each cluster contributes both to operational tempo (the Time cluster sets industry-wide cadence; the Execution cluster enables fast rollback) and to legitimate accountability (the Evidence cluster establishes the factual record; the Proof cluster runs continuous machine-readable compliance; the Key cluster requires joint custodianship for any release). Trust at Speed is therefore not a regulative ideal toward which the industry should aspire; it is the *operational signature* of an institutional architecture that has been built. Where the architecture has been built, Trust at Speed obtains. Where it has not, the speed-legitimacy trade-off recurs in the forms documented throughout this paper.

Trust at Speed reframes the platformer-governance question. Instead of asking “how can platformers be made to accept responsibility?”, the framework asks “how can the temporal infrastructure of governance be reconstructed such that responsibility and speed are mutually reinforcing rather than opposing?” The answer is the consortium architecture.

The empirical analysis presented in this paper demonstrates that:

1. *Type 1 firms* (Tesla, BYD) operate fundamentally different sovereignty regimes despite shared typology, and their SoS extension is governed by the structural trade-off Time Agility × Spatial Diversity ≤ Control Limit.
2. *Type 3 firms* (Toyota, Honda, Hyundai, VW, BMW, Mercedes) converge on a strategic equivalence class — deep retention of residual control rights combined with surface boundary management — within which multiple stable configurations coexist (the *convergence band*). The diverse paths through which the six firms reach this band are determined by scale, diversity, and organisational capability. The path is not chosen; it is constrained.

3. *Type 2 firms* (Google, Huawei) exercise governance through architectures that locate the responsibility-and-governance asymmetry differently — Google through governance without responsibility (CDA 230-style immunity extended into physical industries), Huawei through whole-stack internalisation (commoditising the vehicle in the process). Neither configuration is sustainable in the long run under the mobility industry's physical-safety and product-liability constraints.

The framework's prescriptive contribution — outlined in adjacent research on the subject— proposes the receptive platform governed by juseki (responsibility-taking) as the institutional form toward which the industry's transition would point. The receptive platform operates the five functional clusters within a Three-Loop Model that coordinates Polity-in-the-Loop, Alliance-in-the-Loop, and Machine-in-the-Loop across heterogeneous time scales.

The book on which these papers are based ends with a question that re-orients the entire framework. The conventional industry question — "when will full self-driving be achieved?" — is the wrong question. The right question is structurally different: *Who will control the receptive-platform infrastructure that operates the consortium's five functional clusters within the Three-Loop Model, and who will design the social-impact circuit through which the systematic recombination of value returns to the infrastructure?*

The answer is not predetermined. It depends on whether the institutional architecture this framework proposes can be constructed in time, with the legitimacy, the operational capacity, and the cross-stakeholder coordination that the architecture requires. The empirical analysis in this paper does not predict which actor will provide the answer. It demonstrates that the question, posed at this level of structural precision, is itself the most important strategic instrument that the mobility industry currently has.

References

Cusumano, M. A., Gawer, A., & Yoffie, D. B. (2019). *The business of platforms: Strategy in the age of digital competition, innovation, and power*. Harper Business.

European Union. (2024). Directive (EU) 2024/2853 on liability for defective products [New Product Liability Directive].

Fujimoto, T. (1997). *Seisan shisutemu no shinkaron: Toyota jidōsha ni miru soshiki nōryoku to sōhatsu purosesu* [The evolution of production systems]. Yuhikaku.

Hart, O. (1995). *Firms, contracts, and financial structure*. Oxford University Press.

Henderson, R. M., & Clark, K. B. (1990). Architectural innovation: The reconfiguration of existing product technologies and the failure of established firms. *Administrative Science Quarterly*, 35(1), 9–30.

Lemmon v. Snap Inc., 995 F.3d 1085 (9th Cir. 2021).

Narihara, K. (2019). API to hō: Purattofōmu jidai no kihan keisei [APIs and the law: Norm formation in the platform era]. *Hōritsu Jihō*, 91(10), 12–18.

Narihara, K. (2023). Hakku suru kenri: Gijutsu hyōjun e no hōteki taikō kanōsei [The right to hack: Legal contestation of technical standards]. *Hōgaku Kyōshitsu*, 514, 44–51.

Ohno, T. (1978). *Toyota seisan hōshiki* [Toyota production system]. Diamond.

Parker, G. G., Van Alstyne, M. W., & Choudary, S. P. (2016). *Platform revolution: How networked markets are transforming the economy*. W. W. Norton.

Rochet, J.-C., & Tirole, J. (2003). Platform competition in two-sided markets. *Journal of the European Economic Association*, 1(4), 990–1029.

Teece, D. J. (1986). Profiting from technological innovation: Implications for integration, collaboration, licensing and public policy. *Research Policy*, 15(6), 285–305.

Teece, D. J. (2007). Explicating dynamic capabilities: The nature and microfoundations of (sustainable) enterprise performance. *Strategic Management Journal*, 28(13), 1319–1350.

Tokuda, A.(2026) Governing Through Time: A Theoretical Framework for Platformer-Defined Sovereignty in the Mobility Industry, *Journal of Design Science*, 6(1) (forthcoming)

Ulrich, K. T. (1995). The role of product architecture in the manufacturing firm. *Research Policy*, 24(3), 419–440.