Working Paper
Series C – Computational Governance
WP-C-L02-001

Computational Policy Design Framework

IDHUS Institute

This publication forms part of the constitutional editorial architecture of the IDHUS Institute and belongs to the Institutional Cognition Knowledge Series.

ISBN:

Made and printed in Spain

Index

Abstract

Public policy increasingly confronts problems whose informational complexity, systemic interdependence and uncertainty exceed the cognitive limits of conventional policy processes. Institutions respond to these conditions through distributed expertise, scientific analysis, administrative knowledge and computational tools, yet the use of computation within policy-making is often framed too narrowly as a means of prediction, optimisation or technical support. Such approaches risk overlooking a broader transformation: computational systems can participate not only in the analysis of predefined alternatives, but in the representation of public problems, the integration of heterogeneous evidence, the exploration of wider policy possibility spaces, the examination of trade-offs and uncertainty, and the recursive redesign of interventions in response to implementation evidence.

This Working Paper develops the **Computational Policy Design Framework (CPDF)** as a model for understanding these processes within the wider architecture of Institutional Cognition. Building upon the Institutional Cognition Conceptual Architecture and the Institutional Cognitive Architecture Model, CPDF treats computational capability as one component within a socio-technical policy-design system rather than as an autonomous source of policy authority. The framework distinguishes computational exploration from legitimate institutional judgement and examines how policy representations, evidence architectures, constraint architectures, policy possibility spaces, trade-off spaces, uncertainty, scenarios and feedback can be organised into a recursive process of policy learning.

Particular attention is given to the epistemic risks created by computational formalisation. The paper introduces the concept of **computational legibility bias** to describe the tendency of institutions to privilege forms of knowledge that are easier to encode and process, while potentially marginalising qualitative, contextual or tacit forms of evidence. It also develops the principles of **model pluralism**, **epistemic traceability** and **Policy Model Provenance** in order to preserve the visibility of assumptions, evidential sources and model evolution across policy cycles. These mechanisms become especially important as artificial intelligence, generative systems, computational agents and policy digital twins expand the scale and autonomy of computational participation within governance.

CPDF further reconceptualises implementation as an epistemic phase of policy design. Policy interventions generate evidence concerning the adequacy of prior representations, assumptions and models, allowing implementation feedback to contribute not only to parameter adjustment but to the possible re-representation of the problem itself. This recursive structure supports adaptive policy pathways, staged experimentation and learning before scale, while preserving constitutional and normative boundaries around legitimate decision-making.

The framework therefore positions Computational Policy Design between conventional policy analysis and automated governance. Its principal scientific contribution is to establish computation as a means of **expanding the space of institutional reasoning rather than replacing collective judgement**, providing a research architecture through which computational methods can be reconstructed,

compared, validated and experimentally investigated as components of accountable, adaptive and cognitively augmented policy-making.

David González
Director
IDHUS Institute

.

1. Introduction — From Policy Analysis to Computational Policy Design

Public policy has always depended upon the capacity of institutions to construct representations of problems that exceed the immediate experience of individual decision-makers. Statistical systems, scientific models, administrative records, expert analysis and institutional memory allow governments to reason about populations, economies, infrastructures and environmental processes that no single actor could observe directly. In this sense, policy-making has long relied upon technologies of cognition through which complex social realities become sufficiently visible and interpretable to support collective judgement. The contemporary expansion of computational capability extends this historical process, but it does so at a scale that begins to alter not only how institutions analyse proposed policies, but how the space of possible policies itself can be explored.

The distinction is significant. Conventional policy analysis frequently begins once a problem has already been framed and a limited number of alternatives have entered institutional consideration. Analytical methods are then used to estimate costs, assess consequences, compare options or evaluate implementation. Computational Policy Design begins earlier and potentially operates across a broader cognitive domain. Computational systems can contribute to the representation of the problem itself, integrate bodies of evidence that would otherwise remain fragmented, identify relationships across complex datasets, generate or discover alternative intervention configurations, explore possible consequences under multiple assumptions and maintain recursive connections between implementation evidence and subsequent redesign. Computation thereby becomes relevant not merely as an analytical instrument applied to policy choices, but as a means of expanding portions of the cognitive space within which policy alternatives become conceivable.

This expansion should not be confused with the automation of governance. Public policy cannot be reduced to the optimisation of an objective function because governance operates through competing values, contested interpretations, constitutional constraints and legitimate processes of collective choice. A computational system may identify an intervention capable of reducing a measurable social cost while remaining incapable of determining whether the distribution of resulting burdens is politically or ethically acceptable. It may compare alternative allocations of resources without possessing legitimate authority to decide which social priorities deserve precedence. It may reveal previously invisible trade-offs while remaining unable to resolve those trade-offs through computation alone. **The capacity to calculate consequences does not confer the authority to determine collective values.**

The **Computational Policy Design Framework (CPDF)** is developed in this Working Paper from that distinction. Its purpose is to examine how computational methods can augment the institutional processes through which public problems are represented, evidence is integrated, policy possibilities are explored, consequences are modelled and interventions are iteratively revised. The framework treats computational capability as one component within a wider architecture of Institutional Cognition rather than as an autonomous decision-making system. Computational models, artificial intelligence, simulations and data infrastructures therefore enter policy processes as cognitive nodes and interfaces whose institutional significance depends upon how they connect with expertise, deliberation, authority, implementation and feedback.

This positioning connects CPDF directly with the Institutional Cognition Conceptual Architecture and the Institutional Cognitive Architecture Model developed in the preceding L02 Working Papers. ICCA established the conceptual relationships through which Institutional Cognition can be studied as a coherent scientific field, while ICAM examined how cognitive functions become distributed across human, organisational and computational components within institutional systems. CPDF now focuses upon a particular domain within that architecture: the processes through which institutions construct and revise interventions in complex public systems. The central question is no longer simply whether computational methods can improve policy analysis, but **how computational capability can enlarge the representational and exploratory capacities of policy-making while preserving human judgement, institutional accountability and legitimate decision authority**.

2. The Cognitive Limits of Conventional Policy Design

The need for computational augmentation arises partly from a structural mismatch between the complexity of contemporary governance problems and the cognitive resources available to conventional policy processes. Public institutions routinely confront systems containing large numbers of interacting variables, multiple causal pathways, heterogeneous populations, uncertain behavioural responses and consequences distributed across different temporal and geographical scales. Climate transition, urban development, healthcare, technological governance, migration, energy security and economic resilience each involve relationships whose complexity exceeds what unaided human cognition can represent simultaneously.

Institutional organisation already provides one response to these limits. Expertise is distributed across specialised departments, scientific advisory bodies, statistical agencies and external knowledge networks, allowing collective systems to process far more information than individual decision-makers could manage. Yet distribution creates the architectural problems identified by ICAM: knowledge becomes fragmented, representations differ across professional communities and the integration of specialised evidence places increasing demands upon cognitive interfaces. Policy design therefore operates under constraints generated not only by limited individual cognition but by the difficulty of coordinating distributed institutional cognition.

These constraints become particularly visible in the exploration of policy alternatives. Human deliberation can compare a relatively small number of comprehensible options, and institutional processes frequently reduce complex policy spaces to a limited set of politically or administratively recognisable alternatives. This reduction is often necessary for collective choice, but it can occur before the wider space of possible interventions has been examined systematically. Policies inherited from previous periods, familiar administrative instruments and established ideological categories can consequently define the boundaries of institutional imagination even when other configurations might exist.

Temporal pressure further restricts exploration. Policy decisions frequently occur within political, administrative or emergency cycles that limit the amount of analysis possible before action becomes necessary. Evidence must be synthesised, alternatives compared and consequences anticipated within finite periods, creating incentives to rely upon established models and previously available options. Under conditions of crisis, these constraints become particularly severe, while under conditions of gradual systemic change they can produce a different problem: institutions may repeatedly optimise familiar interventions without reconsidering the representation of the problem from which those interventions emerged.

Computational methods can expand some of these constrained cognitive capacities. They can process large datasets, compare combinations of variables, explore scenarios, retrieve historical evidence and model relationships whose behaviour is difficult to infer intuitively. Artificial intelligence can increasingly contribute to evidence synthesis, pattern identification and the generation of possible alternatives, while simulation methods can explore the consequences of interactions across complex systems. Yet these capabilities do not remove bounded cognition; they redistribute it. Computational systems possess their own representational limitations, depend upon selected data and assumptions, and generate outputs that institutions must interpret through human and organisational processes.

The relevant scientific problem is therefore not how computation can overcome the cognitive limitations of policy-making altogether, but how **different forms of human, institutional and computational cognition can be combined so that the limits of one component are partially compensated by the capabilities of others**. Computational Policy Design begins from this more modest but potentially more productive proposition.

3. Policy Design as an Institutional Cognitive Process

Policy design can be understood as the institutional process through which a perceived public condition becomes transformed into an organised possibility for collective intervention. This transformation requires far more than selecting an instrument. Institutions must determine what kind of problem they believe they are confronting, identify which aspects of that problem are relevant to public action, integrate available knowledge, construct causal expectations, consider alternative interventions, anticipate consequences and determine which combinations of objectives and constraints should guide collective choice. Policy design is therefore intrinsically cognitive before it becomes administrative.

The process is also distributed. Problem definitions may originate within political institutions, scientific communities, civil society or administrative systems. Evidence may be produced by statistical agencies, universities, regulators, international organisations or local actors. Policy alternatives may emerge from professional expertise, political programmes, historical precedent or experimentation elsewhere. Legal institutions define boundaries of permissible action, while operational agencies contribute knowledge concerning implementation. The resulting policy is therefore rarely the product of a single cognitive centre; it emerges through an architecture connecting multiple forms of knowledge and authority.

Computational systems enter this architecture at several possible points. They may contribute to perception by identifying patterns within administrative or environmental data, to memory by retrieving relevant historical evidence, to reasoning by modelling causal relationships, to exploration by generating or comparing alternative policy configurations, and to anticipation by simulating possible consequences. These contributions correspond closely to the functional distinctions established within ICCA and the architectural relationships developed through ICAM. CPDF therefore does not create a separate theory of computational governance but specifies how computational capabilities can participate within existing processes of institutional cognition.

This perspective prevents a common conceptual error in discussions of algorithmic policy-making: the assumption that the relevant comparison is between human decision-making and machine decision-making. In most complex governance systems, neither exists independently. Human judgement already depends upon statistical models, databases, legal classifications and organisational procedures, while computational systems depend upon human choices concerning data, objectives, categories and institutional use. The meaningful unit of analysis is consequently the **socio-technical policy design architecture** through which different cognitive contributions become connected.

Understanding policy design as cognition also clarifies why computational improvement cannot be assessed solely through predictive accuracy or processing efficiency. A model may predict an outcome accurately while representing only a narrow dimension of the policy problem. An optimisation system may identify an efficient allocation while excluding relevant distributional concerns. A generative model may produce numerous plausible interventions without distinguishing which are institutionally feasible or normatively legitimate. The cognitive value of computation therefore depends upon its position within the wider design process.

Computational Policy Design should consequently be understood as the structured augmentation of institutional policy cognition rather than the automation of institutional choice. This distinction provides the conceptual boundary within which the remainder of CPDF is developed.

4. The Problem of Representation

Every policy process begins, explicitly or implicitly, with a representation of reality. Before an institution can intervene in unemployment, housing affordability, educational inequality, environmental degradation or technological risk, it must construct some account of what the relevant phenomenon is, how it can be observed, which populations or processes are involved and which relationships are believed to produce the outcomes of concern. Policy problems are therefore never encountered institutionally in their entirety; they become governable through representations that make selected dimensions visible while leaving others outside the immediate field of analysis.

Computation intensifies the importance of this representational process because computational systems require aspects of reality to become sufficiently formalised for processing. Variables must be selected, categories defined, relationships specified or inferred, data sources connected and assumptions incorporated into models. Even machine-learning systems that discover statistical patterns without explicit causal specification depend upon prior choices concerning what data are collected, how observations are labelled and which outcomes are treated as relevant. Computational power does not eliminate representation; it increases the scale and complexity at which representations can operate.

The distinction between the public problem and its computational representation must therefore remain foundational to CPDF. A computational model of urban mobility is not urban mobility itself, just as a model of poverty does not contain the complete social reality of poverty. Every representation compresses complexity in order to make particular relationships cognitively manageable. This compression is necessary for analysis, but it becomes dangerous when the boundaries of the representation disappear from institutional awareness and model outputs begin to be treated as direct descriptions of reality.

Representation also shapes intervention. The way a problem is modelled influences which policy alternatives become visible. If housing affordability is represented principally as an imbalance between aggregate supply and demand, computational exploration will tend to privilege interventions affecting those variables. If the representation incorporates spatial inequality, ownership concentration, credit structures, demographic change and transport accessibility, a different policy possibility space emerges. **The architecture of representation therefore precedes the architecture of solution.**

This relationship means that computational policy design must preserve the possibility of representational plurality. Complex public problems can often be represented through several legitimate models that emphasise different mechanisms or scales. Rather than treating disagreement between representations as an obstacle to be eliminated, institutions may use it as a source of cognitive information. Differences between models can expose uncertainty concerning causal structure, reveal assumptions that would otherwise remain hidden and demonstrate how policy conclusions depend upon alternative interpretations of the problem.

CPDF therefore treats representation not as a preliminary technical task but as a continuing institutional cognitive process. Models should remain revisable as new evidence emerges, implementation reveals unexpected behaviour or alternative interpretations become credible. Computational Policy Design begins not with the assumption that the problem has been correctly formalised, but with the recognition that **the representation through which an institution understands a problem is itself one of the principal objects of policy learning**.

5. Computational Representations and Model Pluralism

The diversity of public problems makes it unlikely that any single computational methodology can provide an adequate representation across governance domains. Statistical models can identify associations and estimate relationships within observed data; causal models attempt to represent mechanisms through which interventions produce effects; system-dynamics approaches can describe feedback relationships across aggregated variables; agent-based models can explore interactions among heterogeneous actors; network models can represent relational structures; machine-learning systems can identify complex patterns; knowledge graphs can connect distributed bodies of information; and digital twins can provide continuously updated representations of selected systems. Each form of modelling makes particular aspects of reality visible while simplifying others.

The choice of representation is therefore epistemically consequential. A model designed around aggregate variables may reveal system-level trends while concealing local heterogeneity. Agent-based representations can capture behavioural diversity while depending upon assumptions concerning individual rules. Machine-learning models may achieve strong predictive performance while providing limited insight into causal mechanisms. Highly interpretable models may sacrifice some predictive complexity in exchange for greater institutional transparency. Computational Policy Design must therefore evaluate models not only according to technical performance but according to the cognitive function they are expected to perform within policy architecture.

This leads CPDF towards a principle of **model pluralism**. Where public problems contain substantial uncertainty, institutions may benefit from maintaining several representations rather than relying upon a single dominant model. Competing models can be compared, their assumptions exposed and their divergent predictions examined. Agreement across substantially different modelling approaches may increase confidence in particular findings, while disagreement can reveal areas in which institutional knowledge remains fragile.

Model pluralism also protects against the gradual identification of what is computationally convenient with what is institutionally real. Different methods possess different data requirements, and institutions can become tempted to frame problems according to the information already available rather than according to the phenomena that matter. Maintaining several forms of representation creates opportunities for qualitative knowledge, expert judgement and historical evidence to challenge conclusions derived from highly formalised models.

Artificial intelligence expands this representational ecology further. Foundation models can interact with unstructured text, images and other forms of information that traditional computational systems found difficult to incorporate, potentially widening the range of evidence available to policy processes. Yet their flexibility introduces different uncertainties concerning provenance, reliability and interpretability. Their inclusion therefore strengthens rather than weakens the case for plural representations and cross-validation.

Model pluralism does not imply that every representation deserves equal epistemic status. Models remain subject to evidence, methodological scrutiny and domain-specific validation. The principle instead recognises that complex governance problems may require several partial representations whose combined use provides a more robust basis for institutional reasoning than premature dependence upon a single computational worldview.

6. Evidence Architecture and Computational Legibility

Computational representations depend upon evidence, but institutional evidence exists in forms that differ substantially in structure, accessibility and computational legibility. Administrative datasets, sensor networks and statistical records can often be incorporated relatively directly into computational systems, while qualitative research, historical knowledge, legal interpretation, professional expertise, citizen experience and tacit organisational knowledge are more difficult to formalise. The increasing use of computational methods therefore creates a risk that evidence will acquire institutional influence partly according to how easily it can be processed rather than according to its relevance to the problem.

CPDF describes the wider configuration through which knowledge enters computational policy processes as an **evidence architecture**. This architecture includes data sources, scientific literature, institutional memory, expert knowledge, legal frameworks, public consultation, implementation experience and other forms of evidence capable of informing problem representation or policy evaluation. Its purpose is not to imply that all forms of knowledge can or should be converted into the same computational format, but to ensure that computational processes remain situated within a broader epistemic environment.

The asymmetry between evidence that is easily formalised and evidence that resists formalisation can generate what may be described as **computational legibility bias**. Institutions may gradually privilege measurable phenomena because computational systems can represent them more efficiently, while dimensions of governance that depend upon contextual interpretation, lived experience or qualitative judgement receive less weight. Over time, the architecture of computation can therefore influence the architecture of institutional attention.

This problem is not entirely new. Statistical governance has always shaped public visibility through categories and indicators, and administrative systems necessarily simplify complex social conditions in order to act upon them. Computational capability extends this tendency by increasing the volume and granularity of formalised information that institutions can process. The danger is not quantification itself, but the possibility that increasing analytical sophistication creates an illusion of comprehensiveness precisely because what remains outside the model becomes less visible.

Evidence architecture should therefore preserve mechanisms through which non-computational knowledge can challenge computational representations. Domain experts may identify assumptions inconsistent with practice; affected communities may reveal consequences absent from administrative datasets; legal reasoning may establish constraints that cannot be inferred statistically; historical analysis may expose patterns invisible within short data series. These contributions do not compete with computation but provide cognitive interfaces through which its representations can remain connected to institutional reality.

The objective of Computational Policy Design is consequently not to maximise the proportion of governance that becomes computationally legible. It is to construct **an epistemically plural architecture in which computationally tractable evidence can be combined with forms of knowledge whose value does not depend upon their ease of formalisation**. Only within such an architecture can increased computational capacity expand institutional cognition without progressively narrowing the reality that institutions are capable of seeing.

7. The Policy Possibility Space

Once a public problem has been represented and relevant evidence has entered the design process, policy-making confronts another form of cognitive limitation: the number of possible interventions can become extraordinarily large. Conventional institutional processes necessarily reduce this complexity by concentrating attention upon a manageable set of alternatives, often derived from established policy instruments, political programmes, historical precedent or proposals developed by specialised actors. Such reduction makes deliberation possible, but it can also restrict institutional imagination before the wider range of potential interventions has been examined.

CPDF introduces the concept of the **policy possibility space** to describe the broader set of intervention configurations that could theoretically be considered within a given problem domain. A policy is rarely defined by a single choice. It may contain combinations of eligibility rules, financial parameters, implementation mechanisms, geographic targeting, regulatory thresholds, institutional responsibilities and temporal sequences, each of which can vary. Once these dimensions interact, the number of possible configurations expands rapidly beyond what human deliberation can explore systematically.

Computational methods can extend institutional access to this space. Search algorithms, optimisation techniques, simulation, generative systems and other computational approaches can identify combinations that might not emerge through conventional policy development. They can compare large numbers of configurations, identify regions of the possibility space associated with different outcomes and reveal relationships between design parameters that are difficult to anticipate intuitively. Computation therefore provides not only additional analytical power but potentially **additional policy imagination**.

The concept nevertheless requires careful boundaries. The computational possibility of an intervention does not establish its practical feasibility, legal permissibility, normative acceptability or political legitimacy. A model may identify configurations that produce attractive outcomes under its assumptions while requiring resources that do not exist, administrative capacities that cannot be created rapidly, legal changes that violate constitutional constraints or distributions of costs that institutions could not legitimately authorise. The possibility space must therefore be understood as progressively bounded by multiple forms of constraint.

This distinction is crucial because computational systems can create an appearance of neutrality around generated alternatives. A policy configuration may emerge from mathematical search rather than political advocacy, but the space within which that search occurs has already been shaped by choices concerning variables, objectives, constraints and evidence. Computational exploration does not escape institutional judgement; it relocates some of the choices through which judgement becomes embedded.

The scientific contribution of the policy possibility space is therefore not the promise that computation can discover an objectively optimal policy hidden within a vast combinatorial landscape. It is the more modest and potentially more valuable proposition that **institutions can use computation to explore a wider range of possible interventions while making explicit the assumptions and constraints that determine which possibilities become visible**. The next architectural problem is consequently to understand how those constraints should be represented.

8. Constraint Architecture and the Boundaries of Policy Possibility

The expansion of the policy possibility space immediately raises a question that computational exploration cannot answer through search alone: which possibilities belong within the domain of legitimate and practicable institutional action? A computational system may generate an enormous number of technically conceivable interventions, yet most public policy operates within boundaries established by material conditions, available resources, administrative capacity, legal frameworks, constitutional principles, political commitments and normative obligations. The policy possibility space is therefore not an unrestricted universe of mathematically imaginable interventions but a structured domain whose boundaries emerge from several different forms of constraint.

CPDF describes the configuration of these boundaries as a **constraint architecture**. Some constraints arise from the physical properties of the system under consideration: infrastructure cannot be constructed instantaneously, energy systems remain subject to material capacities and demographic processes cannot be altered through administrative instruction alone. Economic constraints concern available resources and the opportunity costs associated with their allocation, while institutional constraints reflect the implementation capacities, organisational structures and professional competencies through which policies must become operational. These limitations can sometimes be modified through policy itself, but they cannot simply be ignored within computational exploration.

Legal and constitutional constraints occupy a different position because they do not merely describe what institutions are capable of doing; they help define what institutions are legitimately permitted to do. A computational model may identify a highly effective intervention that conflicts with established rights, procedural requirements or distributions of authority. Treating such constraints as obstacles to optimisation would misunderstand their institutional function. **In legitimate governance, some constraints are constitutive rather than inefficient: they define the conditions under which collective power may be exercised.**

Normative and political constraints introduce further complexity. Public policies operate within societies containing multiple values and competing conceptions of desirable outcomes. Some commitments may be expressed through law, while others remain embedded in political mandates, institutional traditions or public expectations. Computational systems can represent aspects of these commitments, but their formalisation necessarily involves interpretation. The translation of a principle such as fairness into a measurable constraint, for example, requires decisions concerning what form of fairness is relevant and how competing claims should be treated.

Constraint architecture must therefore remain visible and contestable. If constraints are embedded deeply within computational systems without adequate traceability, policy alternatives may appear impossible for reasons that decision-makers cannot inspect. Conversely, treating every constraint as fixed would prevent policy design from exploring how institutional capacity, legal frameworks or resource structures themselves might evolve. Some constraints are hard, some are conditional and others represent assumptions that can legitimately become objects of policy deliberation.

The purpose of constraint architecture is consequently not to close the policy possibility space prematurely, but to make explicit **why particular regions of that space are considered physically possible, institutionally feasible, legally permissible or normatively admissible**. Computational exploration becomes institutionally meaningful only when the boundaries surrounding possibility remain connected to the forms of authority and judgement through which those boundaries acquire legitimacy.

9. Objectives, Values and the Limits of Policy Optimisation

Once a policy possibility space has been bounded by relevant constraints, computational design confronts another fundamental question: according to what criteria should alternatives be evaluated? Many computational methods are naturally formulated around optimisation, in which possible configurations are compared according to an objective function and the system searches for solutions that maximise or minimise a specified quantity. Within narrowly defined technical problems, this approach can be extraordinarily powerful. Public policy, however, rarely presents institutions with a single uncontested objective.

Governance problems usually involve multiple outcomes whose relationships are partly complementary and partly conflicting. A transport policy may seek to reduce congestion, improve accessibility, limit emissions, preserve affordability and distribute infrastructure equitably across regions. Healthcare policy may simultaneously concern health outcomes, waiting times, financial sustainability, geographic accessibility and equality of treatment. Environmental policy may involve emissions reduction, energy security, industrial competitiveness, employment and intergenerational consequences. These objectives cannot always be reduced to a common metric without introducing normative assumptions concerning how different forms of value should be compared.

The difficulty becomes greater when some policy values resist meaningful quantification. Rights, dignity, procedural fairness, democratic participation and cultural continuity may influence policy choices even when they cannot be represented adequately as quantities to be maximised. Their resistance to straightforward measurement does not make them irrelevant to rational policy design. On the contrary, it demonstrates why computational rationality must remain situated within broader institutional reasoning.

CPDF therefore distinguishes **computational optimisation from institutional judgement**. Optimisation can help institutions understand which configurations perform strongly according to explicitly specified criteria, but the specification of those criteria is itself a normative and political act. Choosing an objective function determines what the computational system treats as success, while assigning weights among objectives embeds assumptions concerning their relative importance. These choices should not disappear behind the technical sophistication of the resulting model.

This distinction also complicates the idea of an optimal policy. In multi-objective environments there may be no single configuration that dominates all others. Improving one outcome may require accepting deterioration in another, creating a landscape of alternatives characterised by different combinations of benefits and costs. The computational task therefore changes from discovering *the* optimal policy towards making the structure of competing possibilities more intelligible.

Computational Policy Design should not seek to remove value conflict from governance by translating it into optimisation. Its more defensible role is to reveal how different objectives interact, where trade-offs emerge and what consequences follow from alternative institutional priorities. Computation can thereby improve the informational conditions of judgement without claiming to resolve the normative questions that make judgement necessary.

10. Trade-off Spaces and the Architecture of Choice

The presence of multiple objectives transforms the policy possibility space into a landscape of trade-offs. Some interventions may perform strongly across several dimensions simultaneously, while others generate improvements in one domain at the cost of deterioration elsewhere. Computational methods can make these relationships more visible by identifying sets of policy configurations that represent different combinations of outcomes rather than compressing all objectives into a single ranking.

CPDF describes this domain as the **policy trade-off space**. The concept shifts attention from optimisation towards structured comparison. Instead of asking which policy is objectively best, institutions can ask which alternatives remain available when different objectives are given different degrees of priority, how much improvement in one dimension requires sacrifice in another and whether particular configurations provide comparatively balanced outcomes across several concerns. Computation becomes a means of mapping the consequences of alternative priorities rather than determining those priorities independently.

This approach has important implications for institutional deliberation. Political disagreement often concerns not the factual consequences of policies alone but the relative importance assigned to those consequences. Two actors may accept the same evidence while preferring different interventions because they evaluate distribution, risk, cost or temporal consequences differently. A computational system that hides these differences within an aggregated objective function may obscure the actual structure of disagreement. A system that exposes the trade-off space can instead make that structure more explicit.

Distributional analysis becomes particularly important. Aggregate improvements can conceal substantial differences among populations, territories or generations. A policy that produces a positive average outcome may impose concentrated costs upon particular communities, while another may distribute benefits more evenly at somewhat greater total expense. Computational exploration can help reveal these patterns when distribution is incorporated explicitly into the representation rather than treated as an external political consideration.

Trade-offs also occur across time. Policies generating immediate benefits may create longer-term vulnerabilities, while investments whose advantages emerge slowly may impose significant short-term costs. Discounting future outcomes provides one computational mechanism for comparison, but the choice of discount rate itself contains normative assumptions concerning the relative value assigned to present and future consequences. Temporal trade-offs therefore demonstrate again that computational techniques can formalise aspects of judgement without eliminating the values embedded within them.

The policy trade-off space thus becomes an interface between computation and deliberation. **Its purpose is not to transform political choice into a technical calculation, but to provide a richer representation of what different choices imply.** In this sense, computational capability can contribute to democratic and institutional reasoning precisely by clarifying where legitimate disagreement remains rather than attempting to make that disagreement disappear.

11. Uncertainty as a First-Class Component of Policy Design

Policy design necessarily concerns futures that have not yet occurred. Even when institutions possess extensive evidence concerning past relationships, interventions can alter behaviour, external conditions can change and complex systems can respond in ways that models fail to anticipate. Uncertainty is therefore not an incidental imperfection that disappears as computational capability improves; it is a structural property of governing open social systems.

Computational systems can nevertheless create a powerful appearance of precision. Numerical outputs, probability distributions and detailed simulations may appear epistemically stronger than qualitative judgements because their assumptions are less visible than their results. A projection expressed to several decimal places can conceal substantial uncertainty concerning data quality, causal structure or behavioural response. Increasing computational sophistication can therefore produce what might be called **precision without proportional certainty**.

CPDF treats uncertainty as a first-class component of computational policy design. Data may be incomplete, biased or measured imperfectly. Parameters estimated from historical observations may change under new conditions. Models may omit important causal relationships or represent them incorrectly. Human behaviour may respond strategically to policy interventions. External events may alter the environment in ways that were not included within the original analysis. At the deepest level, institutions may confront conditions in which neither reliable probabilities nor a complete set of possible futures can be specified.

These different forms of uncertainty matter because they require different forms of institutional response. Additional data may reduce uncertainty concerning some parameters while doing little to address uncertainty about model structure. Comparing several models may reveal whether conclusions depend heavily upon particular assumptions. Scenario exploration may become more useful than point prediction when future conditions cannot be estimated reliably. In situations of deep uncertainty, institutions may need to prioritise adaptability and robustness over attempts to identify the policy expected to perform best under one forecast.

The representation of uncertainty also affects institutional trust. Concealing uncertainty may make computational recommendations appear decisive in the short term, but repeated failure can undermine confidence in analytical systems. Explicit uncertainty can make recommendations appear less authoritative while providing decision-makers with a more accurate understanding of the knowledge available. **Epistemic honesty therefore becomes an architectural requirement rather than a weakness of computational policy design.**

The purpose of computation should consequently not be described as eliminating uncertainty from policy-making. Its more realistic contribution is to help institutions distinguish what is known relatively well, what depends upon assumptions, where models disagree and how policy alternatives behave when those uncertainties are varied. Computation becomes valuable not because it produces certainty, but because it can make parts of uncertainty more systematically explorable.

12. Scenario Exploration and Plural Futures

Once uncertainty is treated as intrinsic to policy design, prediction ceases to be the only meaningful relationship between computation and the future. Institutions frequently need to act in environments where several future conditions remain plausible and where assigning reliable probabilities to those conditions may be difficult. Under such circumstances, constructing a single expected future can narrow institutional attention precisely when adaptability requires a wider field of anticipation.

Scenario exploration provides an alternative cognitive strategy. Rather than asking computational systems to determine what will happen, institutions can use them to examine **what could happen under different combinations of assumptions, behaviours and external conditions**. Scenarios can vary economic trajectories, technological developments, demographic patterns, environmental conditions, behavioural responses or institutional capacities, allowing policy alternatives to be examined across multiple possible environments.

The value of scenarios lies partly in their capacity to expose hidden dependencies. A policy may appear highly effective when economic growth remains strong but perform poorly under prolonged stagnation. Another may depend upon technological adoption occurring more rapidly than historical evidence supports. A third may remain effective across several plausible conditions but provide less impressive results under the central forecast. These comparisons reveal information that disappears when policy evaluation concentrates upon a single expected outcome.

Scenario exploration also creates opportunities for combining quantitative and qualitative reasoning. Not every relevant future condition can be represented through a probability distribution or historical dataset. Geopolitical change, social transformation, regulatory shifts and technological breakthroughs may require structured narratives that can then inform computational experiments. Computational Policy Design should therefore treat scenarios as cognitive interfaces between formal models and broader institutional foresight rather than as purely numerical forecasts.

The use of plural futures carries its own risks. Institutions can generate so many scenarios that decision-making becomes paralysed, or select scenarios strategically to support preferred interventions. Scenario construction therefore requires transparent assumptions and disciplined relationships with available evidence. The objective is not to imagine every conceivable future but to examine a sufficiently diverse set of plausible conditions to test whether policy conclusions depend excessively upon one representation of what lies ahead.

Scenario exploration changes the role of computation from prediction towards structured anticipation. Instead of asking models to tell institutions which future will occur, CPDF uses computational capability to investigate how different policies might behave across futures whose uncertainty must remain visible.

13. Policy Robustness under Uncertain Conditions

The exploration of multiple futures creates a different criterion for evaluating policy alternatives. If no single forecast can be treated as sufficiently reliable, selecting the intervention that performs best under the most probable scenario may expose institutions to substantial failure when conditions differ from expectation. Policy design can therefore shift attention from optimality under one assumed future towards **robustness across a range of plausible conditions**.

Robustness does not necessarily mean selecting the policy with the highest average performance. A robust intervention may sacrifice some potential benefit under favourable conditions in exchange for reducing vulnerability under adverse ones. The appropriate balance depends upon the consequences of failure, the reversibility of the intervention and the capacity of institutions to adapt after implementation. Policies affecting critical infrastructure, public health or systemic risk may justify different robustness criteria from interventions that can be modified rapidly at relatively low cost.

This perspective also introduces the concept of **adaptive policy pathways**. Rather than committing permanently to a complete policy configuration at one moment, institutions can design sequences in which subsequent actions depend upon observed conditions. Initial interventions may preserve several future options, while monitoring systems identify thresholds at which alternative pathways should be activated. Policy design thereby becomes temporal and conditional rather than a single irreversible choice.

Computational systems can support such approaches by exploring how alternative pathways behave under different scenarios, identifying conditions under which strategies become vulnerable and examining the value of preserving flexibility. Yet the choice of acceptable vulnerability remains institutional. A model can estimate how often a policy fails under specified scenarios, but it cannot determine independently how much risk society should tolerate or which groups may legitimately bear that risk.

Robustness must also be distinguished from conservatism. Designing only for resilience against failure can discourage experimentation and lock institutions into familiar interventions. In some contexts, transformational policies may involve uncertainty precisely because historical evidence provides limited guidance. Computational Policy Design should therefore support comparison between robustness, adaptability and potential transformative benefit rather than treating any one criterion as universally dominant.

The deeper principle is that **policy quality under uncertainty depends not only upon expected performance but upon how interventions behave when expectations are wrong**. This moves CPDF beyond predictive policy analysis towards an architecture in which uncertainty, resilience and adaptive capacity become explicit components of institutional judgement.

14. Systemic Effects and the Limits of Linear Intervention

Public policy often begins from an intuitive causal structure in which an intervention is expected to produce a desired effect. Yet complex social, economic and environmental systems rarely respond as passive objects. Policies alter incentives, redistribute resources, change expectations and modify relationships among actors, whose responses can subsequently alter the conditions upon which the original policy was based. Intervention therefore becomes part of the system it attempts to govern.

This creates the possibility of **second-order and systemic effects**. A subsidy intended to encourage a desirable activity may alter market prices; a regulation may generate new organisational strategies; infrastructure investment may reshape settlement patterns; a welfare intervention may affect labour-market behaviour; environmental regulation may accelerate technological substitution. Some consequences reinforce the intended effect, others weaken it and still others emerge in domains that were not part of the original policy objective.

Computational methods can help institutions explore these interactions when representations include feedback, behavioural response and cross-domain relationships. System dynamics, agent-based modelling, network analysis and other approaches can reveal how apparently local interventions propagate through wider systems. Artificial intelligence may contribute by identifying patterns across heterogeneous evidence or detecting relationships that conventional analysis overlooks. Yet systemic modelling also increases representational uncertainty because each additional interaction introduces assumptions whose validity may be difficult to establish.

The possibility of emergent effects is particularly important. Some system-level outcomes arise from interactions among components rather than from any individual actor or policy mechanism. Such effects can be difficult to predict even when local behaviour is understood relatively well. Computational simulation can explore possible emergence, but it cannot guarantee anticipation of genuinely novel system behaviour. CPDF must therefore preserve humility concerning the extent to which complex governance systems can become predictable.

Cross-sector effects provide another reason for architectural integration. A policy designed within one administrative domain may generate consequences for others, while organisational boundaries can prevent those consequences from entering the original design process. Computational Policy Design can potentially connect representations across domains, but only if the institutional architecture permits the necessary evidence and expertise to interact. The systemic quality of the model therefore depends partly upon the systemic quality of the institution using it.

Policies should consequently be understood not as external solutions applied to passive problems, but as interventions introduced into systems that respond, adapt and sometimes reorganise around them. This conception strengthens the recursive character of CPDF and prepares the transition from anticipatory modelling towards implementation, observation and redesign.

15. From Prediction to Exploration

The preceding sections suggest a broader transformation in how computational capability should be understood within policy design. Much of the attraction of computational governance derives from prediction: if institutions could forecast economic behaviour, social outcomes, infrastructure demand or environmental change more accurately, they could presumably make better decisions. Predictive improvement can undoubtedly contribute to policy cognition, but treating prediction as the central purpose of computation risks narrowing the much wider forms of institutional reasoning that computational systems can support.

CPDF proposes **exploration** as a more general organising principle. Exploration includes prediction but extends beyond it. Institutions can explore alternative representations of a problem, compare different causal assumptions, search wider policy possibility spaces, map trade-offs among competing objectives, test interventions across multiple scenarios and examine how outcomes change when assumptions are varied. Computation becomes a means of interrogating possibilities rather than merely forecasting a single trajectory.

This distinction matters particularly under conditions of deep uncertainty. Prediction seeks to reduce uncertainty concerning what will happen, while exploration asks how institutional choices behave across what might happen. The former remains valuable when relationships are sufficiently stable and evidence is strong; the latter becomes increasingly important when systems are complex, adaptive or undergoing structural change. Intelligent policy design requires the capacity to move between these modes rather than treating one as universally superior.

Exploration also creates a different relationship between computational systems and human judgement. A predictive system can easily acquire the status of an oracle whose output appears to recommend a course of action. An exploratory system instead presents decision-makers with structured possibilities, dependencies, trade-offs and uncertainties that require interpretation. Its contribution lies in expanding the cognitive environment within which judgement occurs.

This does not automatically guarantee better governance. Decision-makers may still misuse models, ignore inconvenient scenarios or privilege computationally legible outcomes. Exploration must therefore remain connected to the evidence architecture, constraint architecture and legitimate authority structures established throughout CPDF. The framework does not remove institutional responsibility by increasing the number of possibilities that can be examined.

The transition from prediction to exploration nevertheless captures one of the central propositions of Computational Policy Design. **The most significant contribution of computation may not be its ability to tell institutions what will happen or what they should do, but its capacity to make a much larger and more structured field of possible actions and consequences available to collective reasoning.**

16. Computational Exploration and Institutional Judgement

The expansion of computational capability changes the informational environment within which policy decisions are made, but it does not remove the need for institutional judgement. On the contrary, as computational systems make larger possibility spaces, more complex trade-offs and wider ranges of uncertainty visible, the interpretative demands placed upon institutions may increase. Decision-makers must determine which consequences matter, which uncertainties are acceptable, which constraints should remain binding and how competing values should be reconciled within legitimate processes of collective choice.

This relationship requires a distinction between **computational exploration and institutional judgement**. Computational exploration concerns the capacity to represent problems, search policy possibility spaces, compare configurations, model consequences and examine behaviour across alternative scenarios. Institutional judgement concerns the interpretation of those findings within political, legal, ethical and administrative contexts through which collective decisions acquire meaning and authority. The two processes are interdependent but should not be collapsed into one another.

Institutional judgement cannot simply be added at the end of a computational pipeline. If human decision-makers encounter computational outputs only after problems, objectives, constraints and alternatives have already been formalised, many of the most consequential policy choices may already have occurred upstream. The definition of the problem, selection of evidence, construction of objectives and specification of constraints all contain forms of judgement. Meaningful human and institutional participation must therefore extend throughout the design process rather than functioning as a final approval mechanism.

This distributed relationship between computation and judgement also complicates familiar notions of human oversight. A decision formally authorised by a human actor may nevertheless be cognitively dominated by computational systems if the actor lacks practical capacity to interrogate the assumptions, evidence or alternatives underlying their outputs. Conversely, rejecting computational recommendations does not necessarily demonstrate meaningful human control if the reasons for rejection remain disconnected from evidence or institutional responsibility. The relevant question is not whether a human remains nominally present, but whether **the architecture preserves meaningful capacities for interpretation, challenge and legitimate choice**.

Different institutional contexts will distribute these functions differently. Technical agencies may rely heavily upon computational models while embedding expert review throughout their development; democratic bodies may use computational scenarios to structure deliberation without delegating value judgements; administrative organisations may employ optimisation for narrowly specified operational tasks while retaining broader policy objectives outside the model. CPDF does not prescribe a universal division of labour but requires that the relationship remain explicit.

Computational Policy Design therefore depends upon a form of cognitive complementarity. Computation can expand the range and complexity of relationships institutions are capable of examining, while institutional judgement provides contextual interpretation, normative evaluation and legitimate authority. **The objective is neither human control over an otherwise autonomous computational process nor computational replacement of bounded human reasoning, but an architecture in which different cognitive capacities remain mutually corrective and institutionally accountable.**

17. Epistemic Traceability in Computational Policy Design

As computational systems become more deeply embedded within policy processes, the ability to reconstruct how particular outputs were produced becomes increasingly important. A recommendation, scenario or projected consequence acquires institutional significance not merely because it has been generated by a sophisticated model, but because decision-makers can establish some relationship between the output and the evidence, assumptions, transformations and constraints from which it emerged.

CPDF describes this requirement as **epistemic traceability**. Traceability concerns the capacity to follow the cognitive lineage of a computational contribution through the policy design process. It asks which evidence entered the model, how variables were defined, what assumptions shaped relationships, which objectives and constraints were specified, what uncertainties were represented and how intermediate transformations contributed to the final output. The objective is not necessarily to make every computational operation intuitively understandable to every participant, but to preserve sufficient transparency that claims can be examined at the appropriate level of institutional responsibility.

This distinction matters because explainability and traceability are related but not identical. Some computational models may be technically complex while remaining traceable in terms of their data, assumptions, development process and validation history. Conversely, a simple model may be easy to explain while relying upon evidence whose provenance is uncertain or categories whose institutional meaning has never been examined. Traceability therefore extends beyond model interpretability towards the wider epistemic architecture in which computation operates.

Generative artificial intelligence makes this requirement particularly important. Systems capable of synthesising large bodies of information or generating policy alternatives may produce outputs whose apparent coherence exceeds the transparency of their underlying evidential relationships. If such outputs enter policy processes without adequate mechanisms for source verification and uncertainty representation, institutions may acquire analytical fluency at the cost of epistemic reliability. Computational capability must therefore be accompanied by infrastructures that preserve relationships between generated representations and the knowledge from which they derive.

Traceability also supports institutional learning. When a policy performs differently from what a model anticipated, researchers and decision-makers need to reconstruct which assumptions were incorrect, which evidence proved inadequate and whether the failure arose from the model, the implementation process or changes in the environment. Without traceability, computational errors become difficult to diagnose and institutional learning remains shallow.

Epistemic traceability transforms computational outputs from apparently self-contained answers into inspectable stages within a longer process of institutional reasoning. This property becomes essential if computation is to support accountable policy design rather than create new forms of cognitive opacity.

18. Policy Model Provenance

Epistemic traceability concerns the broader ability to reconstruct computational reasoning, while a related concept becomes necessary at the level of individual models and representations. Policy models evolve over time. Data sources change, assumptions are revised, parameters are recalibrated, objectives are modified and new modelling components are incorporated. Without systematic records of this evolution, institutions can lose the capacity to understand why a model produces particular outputs or how contemporary versions differ from those used in previous decisions.

CPDF introduces **Policy Model Provenance** as the structured preservation of the origins, assumptions, evidence, transformations, validation history and version development associated with computational models used in policy design. Provenance creates an institutional memory around modelling itself. It allows future analysts to determine not only which model was used but which version, under what assumptions, with what evidence and for which decision context.

The importance of provenance becomes visible when models migrate across institutional contexts. A model developed for one population, economic environment or administrative system may later be reused elsewhere because its technical infrastructure already exists. Yet relationships valid in the original setting may not transfer automatically. Provenance records can preserve information concerning the conditions under which the model was developed and validated, helping institutions recognise when adaptation or revalidation is necessary.

Model provenance also supports comparison. When several computational representations produce different conclusions, researchers need to understand whether disagreement arises from data, assumptions, causal structures, parameter choices or objectives. Preserving these differences explicitly makes model pluralism scientifically productive rather than merely confusing. Divergent outputs become opportunities to identify where institutional knowledge remains uncertain.

Artificial intelligence introduces additional provenance challenges because models may depend upon training processes and external datasets that policy institutions did not construct themselves. Foundation models may contain knowledge whose precise origin cannot always be reconstructed at the level required for conventional evidence assessment. Their use within high-stakes policy processes therefore requires careful differentiation between exploratory assistance and evidential claims whose provenance must meet stronger standards.

Policy Model Provenance should not be understood as an administrative documentation requirement added after modelling is complete. It forms part of the cognitive infrastructure through which models become institutionally usable across time. **A policy model without provenance may generate information in the present while weakening the institution's capacity to understand, challenge and learn from that information in the future.**

19. Deliberation within Computationally Expanded Policy Spaces

The ability to explore wider policy spaces does not eliminate deliberation; it changes the object of deliberation. Conventional policy processes often require participants to choose among a small number of preconstructed alternatives. Computational Policy Design can instead expose a larger landscape of configurations, trade-offs and uncertainties, potentially allowing institutional actors to deliberate not only about which option should be selected but about how the policy space itself has been constructed.

This creates opportunities for a more reflective form of policy reasoning. Participants may examine why particular objectives were included, how constraints were defined, which populations experience different consequences and how conclusions change when assumptions are varied. Computational systems can therefore become instruments for interrogating policy representations rather than merely producing recommendations from them.

For this potential to be realised, however, computational outputs must be translated into forms compatible with institutional deliberation. Highly dimensional possibility spaces or complex simulation results cannot simply be transferred into political or administrative processes without cognitive mediation. Interfaces must summarise patterns while preserving relevant uncertainty, identify significant trade-offs without hiding minority outcomes and allow decision-makers to investigate the assumptions behind particular results. The design of these interfaces becomes part of policy architecture itself.

There is also a risk that computational sophistication may narrow deliberation by creating asymmetries between actors capable of understanding models and those dependent upon expert interpretation. Technical expertise can become a form of cognitive gatekeeping if model assumptions or alternatives cannot be challenged by participants outside specialised communities. Computational Policy Design must therefore consider not only model performance but **deliberative accessibility**: the degree to which relevant institutional actors can meaningfully engage with the representations informing collective choice.

Accessibility does not require reducing complex models to simplistic explanations. Different participants may require different levels of representation, from technical documentation for specialists to structured trade-off visualisations for decision-makers and publicly understandable accounts for democratic scrutiny. What matters is that complexity does not become an implicit justification for removing computationally mediated policy choices from legitimate contestation.

Computationally expanded policy spaces can therefore deepen deliberation when they reveal relationships previously hidden by narrow option sets, but they can weaken deliberation when models become epistemic authorities whose assumptions are inaccessible. **The institutional value of computational exploration depends partly upon whether it expands not only what can be calculated, but what can be meaningfully questioned.**

20. Implementation as a Source of Policy Knowledge

Policy design does not end when an intervention is authorised. Implementation moves policy from representation into a social environment whose behaviour can differ substantially from the assumptions embedded within the design process. Administrative capacities vary, target populations respond strategically, local contexts modify effects and interactions with existing institutions generate consequences that models may not have anticipated. Implementation therefore produces knowledge that cannot be obtained through ex ante analysis alone.

CPDF treats implementation as an **epistemic phase of policy design** rather than merely the execution of a completed decision. The behaviour of a policy in practice provides evidence concerning the adequacy of problem representations, causal assumptions, institutional capacities and predicted consequences. Implementation can reveal whether constraints were represented realistically, whether behavioural responses differ from expectations and whether effects emerge in domains that were not included within the original model.

This perspective changes the relationship between policy design and evaluation. Evaluation is no longer confined to determining whether a policy succeeded according to predefined indicators; it becomes a mechanism for testing the representations that produced the policy. Unexpected outcomes may indicate implementation failure, but they may also reveal that the original model of the problem was incomplete. Distinguishing between these explanations becomes a central task of institutional learning.

Computational infrastructures can strengthen this connection by integrating implementation data into ongoing representations. Administrative records, monitoring systems, sensor data and other sources may allow institutions to observe policy behaviour more continuously than traditional periodic evaluation permits. Yet continuous data collection should not be confused with continuous understanding. New observations require interpretation, and indicators can capture only those dimensions of implementation that the evidence architecture makes visible.

Qualitative feedback remains particularly important because implementation frequently exposes contextual conditions that formal models did not represent. Front-line professionals, affected communities and local institutions may identify mechanisms invisible within aggregate datasets. Their knowledge can challenge computational representations and contribute to subsequent redesign.

Implementation therefore functions as an encounter between the policy model and the governed system. The differences that emerge between expected and observed behaviour are not simply deviations to be corrected; they are potential sources of knowledge through which both policy and model can evolve.

21. Feedback Architecture and Computational Policy Learning

If implementation generates new knowledge, institutions require mechanisms through which that knowledge can return to the design process. The feedback structures identified within ICAM become particularly important here because computational policy systems can accumulate enormous quantities of post-implementation information without necessarily translating that information into meaningful revision. Data availability and institutional learning remain distinct capacities.

CPDF therefore extends the concept of **feedback architecture** into the policy design process. Implementation evidence must become connected to the representations, assumptions, models and decisions that preceded it. If a policy produces unexpected outcomes, the relevant question is not merely whether performance indicators changed but which part of the original cognitive architecture requires reconsideration. The objective may have been inappropriate, the model incomplete, the assumed causal mechanism incorrect, the constraint architecture unrealistic or implementation capacity insufficient.

Computational systems can assist this diagnostic process by comparing predicted and observed behaviour, identifying persistent deviations and tracing patterns across implementation environments. Machine learning may detect relationships that were not included in original models, while simulation can test whether revised assumptions reproduce observed outcomes more successfully. These capabilities can accelerate learning, but they do not determine how evidence should be interpreted or whether policy objectives themselves require reconsideration.

Feedback also operates across several temporal scales. Some implementation effects appear rapidly and can support operational adjustment, while others become visible only after years. Policies may therefore require nested learning cycles in which short-term monitoring informs incremental modification while longer-term evaluation challenges deeper assumptions concerning problem representation or institutional strategy.

A computational policy system that continuously updates parameters without revisiting its conceptual structure may appear adaptive while remaining cognitively rigid. Genuine policy learning requires the possibility that evidence can alter not only values within a model but the model itself. **Feedback must therefore remain capable of reaching the representational foundations of policy design rather than merely recalibrating existing computational structures.**

This distinction connects CPDF directly with Institutional Cognition. Policy learning becomes recursive when consequences modify the cognitive architecture through which subsequent interventions are imagined. Computational systems can make this recursion faster and more evidence-rich, but only if the institutional architecture permits feedback to challenge established representations.

22. Recursive Computational Policy Design

The relationships developed throughout the preceding sections allow Computational Policy Design to be represented as a recursive process rather than a linear sequence. Public problems are represented through models that integrate selected evidence; those representations define policy possibility spaces whose boundaries depend upon constraints; alternative configurations are explored according to multiple objectives; trade-offs and uncertainties are examined; institutional deliberation interprets those findings; legitimate authorities select interventions; implementation generates new evidence; and feedback returns that evidence to the representations from which the process began.

The resulting structure can be expressed conceptually as:

Representation → Evidence Integration → Policy Space Exploration → Consequence Exploration → Institutional Judgement → Legitimate Choice → Implementation → Feedback → Re-representation

This sequence should not be interpreted as a rigid policy cycle. In practice, several processes occur simultaneously and repeatedly. New evidence can alter representations before policy selection; deliberation can challenge objectives and return the process to earlier stages; feasibility analysis can modify constraints; implementation experience can generate entirely new problem definitions. The value of the structure lies in identifying cognitive relationships rather than prescribing an administrative procedure.

Re-representation is particularly important. Many models of policy learning assume that evidence leads to adjustment within an existing policy framework. CPDF allows a deeper form of revision in which experience can alter the institution's understanding of the problem itself. A policy designed around one causal interpretation may reveal through implementation that another mechanism is more important. The possibility space can then change because the representation generating that space has changed.

This recursive structure also creates a natural role for computational memory. Models, assumptions, previous alternatives, implementation evidence and evaluation results can be preserved across policy cycles, allowing institutions to compare contemporary reasoning with earlier decisions. Policy Model Provenance becomes essential because recursive learning requires knowledge not only of what happened but of what institutions previously believed would happen and why.

The architecture therefore transforms policy from a finished intervention into a continuously revisable relationship between institutional knowledge and an evolving governed environment. This does not mean that every policy should change continuously; stability itself can be institutionally valuable, and excessive revision can create uncertainty or undermine implementation. Recursion means that policies remain cognitively open to revision when evidence justifies reconsideration.

Computational Policy Design is therefore best understood not as the computational production of policy, but as an iterative institutional learning architecture through which representations, interventions and evidence can continuously inform one another while legitimate authority remains responsible for collective choice.

23. Adaptive Policy Pathways and Conditional Design

The recursive structure of CPDF creates the possibility of designing policies whose future evolution is considered before implementation begins. Instead of selecting a single intervention and treating subsequent modification as evidence that the original design failed, institutions can construct **adaptive policy pathways** in which alternative future actions are anticipated and linked to observable conditions.

Such pathways are particularly useful when uncertainty is substantial but delaying all action would itself carry significant costs. Institutions may implement an initial intervention while preserving several future options, define indicators that signal when assumptions are no longer holding and establish decision points at which alternative strategies become appropriate. Policy design thereby incorporates adaptation as part of the original architecture rather than treating it solely as retrospective correction.

Computational methods can help explore these pathways by examining how sequences of decisions behave across multiple scenarios. The relevant policy possibility space becomes temporal: institutions are no longer comparing only alternative interventions at one moment but alternative sequences of intervention conditioned upon future observations. This greatly expands computational complexity while potentially providing a more realistic representation of governance under uncertainty.

Conditional design also places substantial demands upon monitoring and institutional memory. Adaptation rules are useful only if relevant changes can be detected and if future decision-makers understand why thresholds were established. Policy Model Provenance must therefore extend across time, preserving the reasoning connecting particular conditions with anticipated responses.

Predefined pathways should nevertheless remain revisable. Conditions may emerge that were not anticipated when the policy was designed, and rigid adherence to an adaptive algorithm could reproduce the same problems as rigid adherence to a fixed policy. **Adaptation must remain institutional rather than merely procedural**, combining pre-analysed options with the capacity for renewed judgement when circumstances exceed the original design.

Adaptive policy pathways therefore represent one practical expression of the broader CPDF principle that uncertainty should shape policy architecture rather than simply appear as an error term within prediction. Policies can be designed not only for expected conditions but for the possibility that institutional expectations themselves will require revision.

24. Policy Experimentation and Learning before Scale

Recursive policy design also changes the relationship between experimentation and governance. When uncertainty concerns not merely policy parameters but the validity of causal assumptions or behavioural responses, institutions may benefit from testing interventions at limited scales before committing to broader implementation. Policy experimentation can generate evidence about how governed systems respond while containing some of the risks associated with large-scale intervention.

Computational Policy Design can support experimentation before, during and after implementation. Models can identify which assumptions contribute most strongly to uncertainty and therefore where empirical testing may provide the greatest informational value. Simulation can compare possible experimental designs, while implementation data can subsequently update representations and refine future alternatives. Experimentation becomes part of a wider learning architecture rather than an isolated evaluation technique.

The institutional legitimacy of experimentation nevertheless requires careful attention. Public policy experiments affect real populations, and the distribution of risks and benefits cannot be justified solely by the informational value generated for future decision-making. Legal rights, ethical standards, informed participation where appropriate and mechanisms for accountability remain necessary. Computational identification of an informative experiment does not establish that the experiment should legitimately be conducted.

Nor should experimentation be understood as requiring conventional randomised trials in every context. Pilot programmes, phased implementation, regional variation, regulatory sandboxes and natural experiments can all generate useful evidence under different conditions. The appropriate method depends upon the policy domain, ethical constraints and the nature of the uncertainty being investigated.

A particularly important principle is **learning before scale**. Contemporary computational and AI-enabled systems can encourage rapid deployment because technical capability makes implementation appear easy. CPDF instead suggests that institutional uncertainty should sometimes produce deliberate staging, allowing models and policies to encounter reality before their assumptions become embedded across entire systems.

Policy experimentation thus provides another connection between computation and institutional learning. **The purpose of experimentation is not merely to determine whether an intervention works, but to improve the representations through which institutions understand why, where and under what conditions different interventions may work.**

25. The Emerging Architecture of Computational Policy Design

The components developed across the paper can now be assembled into an emerging architecture. Computational Policy Design begins with representations of public problems constructed through an evidence architecture that combines computationally tractable information with forms of knowledge that may resist formalisation. These representations define policy possibility spaces whose boundaries are shaped by physical, economic, institutional, legal, constitutional, normative and political constraints. Within those spaces, computational systems can explore alternative configurations, examine trade-offs, represent uncertainty and investigate behaviour across plural futures.

The outputs of this exploration enter institutional judgement through interfaces that must preserve epistemic traceability, uncertainty and deliberative accessibility. Legitimate institutions interpret competing objectives and consequences, authorise collective choices and determine which risks or distributions remain acceptable. Implementation then exposes the policy to the behaviour of the governed system, generating evidence that enters feedback structures capable of modifying parameters, models, assumptions or the original representation of the problem.

This architecture is recursive because no component should be treated as permanently settled. Evidence can challenge representation; deliberation can challenge objectives; implementation can reveal new constraints; model comparison can expose uncertainty; feedback can alter the possibility space. Computational systems contribute throughout this architecture, but their role differs according to the cognitive function being performed. Data systems may support sensing and memory, models may support representation and reasoning, simulations may support exploration, and artificial intelligence may assist evidence synthesis, pattern recognition or alternative generation.

The architecture is also socio-technical. Human actors, organisational structures and computational systems participate through different forms of capability and authority. No single component contains the policy design process as a whole. The quality of Computational Policy Design therefore depends upon the relationships among these components as much as upon the sophistication of any individual model.

This provides the direct connection back to ICAM. CPDF can be understood as a specialised cognitive architecture nested within the broader institutional cognitive architecture. Its computational models become cognitive nodes; its evidence pipelines and deliberative interfaces become cognitive interfaces; its implementation data create feedback flows; and its recursive learning processes contribute to institutional adaptation. **Computational Policy Design is therefore not an external technological layer added to governance, but a particular configuration through which computational capability can participate within institutional cognition.**

At this point, the framework begins to acquire its full form. Its defining proposition is not that computation should determine policy, nor even that every policy process should become computationally intensive. It is that **where complexity, uncertainty and combinatorial possibility exceed conventional cognitive capacity, institutions can construct accountable computational architectures that enlarge the space of reasoning while preserving the distinction between exploration and legitimate choice.**

26. Policy Digital Twins and Continuously Updated Representations

The recursive architecture developed through CPDF raises the possibility that some policy representations may evolve from periodically reconstructed models towards computational systems that are updated more continuously as new evidence becomes available. Developments in data infrastructure, sensing, simulation and artificial intelligence increasingly allow selected aspects of physical and social systems to be represented dynamically, creating what are often described as digital twins. Within public governance, however, the concept requires substantial qualification because societies and institutions cannot be reproduced computationally with the degree of correspondence sometimes implied by engineering applications.

CPDF therefore defines a **policy digital twin** more cautiously as a continuously or periodically updated computational representation of selected aspects of a governed system that can be used to explore possible interventions, monitor changing conditions and compare observed behaviour with modelled expectations. The emphasis upon *selected aspects* is essential. A policy digital twin is not a digital replica of society; it remains a constructed representation whose boundaries, variables and causal assumptions reflect particular institutional purposes.

The potential cognitive value of such systems arises from their capacity to connect representation, implementation and feedback more closely. Instead of constructing a model for a particular policy decision and allowing it to become progressively detached from changing conditions, institutions could update representations as new administrative, economic, environmental or behavioural evidence becomes available. Differences between modelled and observed behaviour could then become visible more rapidly, allowing assumptions to be reconsidered and alternative policy configurations explored.

Continuous updating nevertheless introduces its own epistemic risks. A model that absorbs new data frequently may appear increasingly accurate while preserving a conceptual structure that no longer represents the most important dimensions of the problem. Parameter adaptation is not equivalent to conceptual learning. Policy digital twins therefore require mechanisms through which institutions can revise not only the values within a model but the architecture of the representation itself.

The availability of real-time or near-real-time data can also encourage an excessive orientation towards phenomena that change rapidly and can be measured continuously. Many important governance processes develop slowly, remain difficult to quantify or become visible only through qualitative interpretation. A computational representation optimised for continuous sensing may consequently reproduce the computational legibility bias identified earlier in CPDF.

There are also significant institutional questions concerning data governance, privacy, access and authority. Continuously updated representations may require extensive information infrastructures whose creation alters relationships between citizens and public institutions. The cognitive benefits of richer representations cannot therefore be evaluated independently of the legal and normative conditions under which information is collected and used.

Policy digital twins should consequently be understood as one possible implementation of recursive computational policy design rather than as its inevitable technological destination. **Their value lies not in creating a perfect computational copy of the governed world, but in maintaining a revisable interface between institutional representations and changing evidence while preserving awareness of the distance that necessarily remains between model and reality.**

27. Artificial Intelligence within Computational Policy Design

Artificial intelligence substantially expands the range of computational capabilities potentially available to policy institutions, but CPDF deliberately treats AI as one family of technologies within Computational Policy Design rather than as its defining principle. Statistical modelling, optimisation, simulation, knowledge representation and other computational methods existed before contemporary artificial intelligence and will continue to perform important cognitive functions alongside it. The significance of AI lies in the additional forms of perception, synthesis, generation and interaction that it can introduce into this wider architecture.

Machine-learning systems can identify patterns across datasets whose scale or dimensionality exceeds conventional human analysis. Predictive models can estimate relationships relevant to policy planning, while generative systems can synthesise large bodies of textual information, assist with scenario construction and generate candidate policy alternatives. Multimodal systems may eventually integrate textual, visual, spatial and sensor information within shared analytical environments, increasing the range of evidence that computational policy processes can examine.

These capabilities can affect several stages of CPDF simultaneously. AI may contribute to evidence integration by retrieving and synthesising relevant knowledge; to problem representation by identifying patterns or constructing structured representations from heterogeneous information; to policy-space exploration by generating possible interventions; and to consequence exploration by assisting the construction or interpretation of scenarios. The same system may participate across several cognitive functions, reinforcing the ICAM principle that cognitive nodes need not correspond to a single specialised task.

Generative capability deserves particular caution because the ability to produce plausible policy alternatives differs fundamentally from the ability to establish their evidential quality, feasibility or legitimacy. A language model can generate hundreds of interventions that resemble plausible policy proposals, but the resulting expansion of the possibility space may increase rather than reduce institutional cognitive burden if mechanisms for validation, constraint checking and comparison are inadequate. Generativity therefore becomes useful only when embedded within architectures capable of disciplined evaluation.

Artificial intelligence also changes the interface between humans and computational systems. Natural-language interaction can make sophisticated analytical capabilities accessible to actors without specialised technical training, potentially reducing some forms of epistemic gatekeeping. At the same time, conversational fluency can conceal uncertainty and encourage users to attribute greater understanding or reliability to systems than their underlying models justify. Ease of interaction must therefore not be confused with epistemic transparency.

The appropriate question for CPDF is consequently not whether AI should design public policy. It is **which cognitive functions AI can augment, under what evidential conditions, through which interfaces, and within what structures of institutional**

responsibility. The detailed allocation of cognitive functions between humans and artificial systems belongs to the dedicated Human–AI Collaboration research of E2, but CPDF establishes the policy-design environment within which many of those interactions will occur.

28. Computational Agents and the Expansion of Policy Exploration

The development of increasingly autonomous computational agents introduces a further possibility within Computational Policy Design. Rather than responding only to individual analytical queries, computational systems may be capable of performing extended sequences of tasks: retrieving evidence, comparing models, generating alternatives, testing scenarios, identifying inconsistencies and revising intermediate analyses according to predefined objectives. Such systems could substantially increase the scale and speed at which portions of the policy possibility space are explored.

Within CPDF, computational agents should be understood as **delegated exploratory systems rather than autonomous policy authorities**. Their role may involve traversing complex analytical spaces that would be impractical for human teams to examine manually, but the significance of their outputs remains dependent upon the objectives, constraints and evidential architectures established by institutions. Greater computational autonomy in task execution does not imply greater legitimate autonomy in collective decision-making.

Agentic systems also make provenance more difficult and more important. When a computational process performs many intermediate operations, consults multiple sources and modifies its analytical strategy during execution, reconstructing the pathway from initial instruction to final output can become substantially more complex. Epistemic traceability must therefore extend beyond the final recommendation towards the sequence of actions through which the system produced it.

Multi-agent computational environments create additional possibilities. Different agents might be configured to explore alternative assumptions, represent different policy objectives, challenge proposed interventions or search for failure conditions. Such arrangements could provide computational analogues of adversarial review or model pluralism, increasing the diversity of perspectives examined before institutional deliberation. Yet artificial diversity should not be confused with genuine social or epistemic plurality. Agents ultimately operate within representations, objectives and training environments created through human and institutional choices.

The risk of automation bias also becomes more significant as agents perform longer sequences of apparently coherent reasoning. Decision-makers may find it increasingly difficult to identify where assumptions entered the process or which intermediate conclusions deserve scrutiny. Computational capability can therefore create a paradox: the more analytical labour institutions delegate, the greater the need for architectures capable of understanding and challenging what has been delegated.

Agentic computation may dramatically enlarge the exploratory capacity of policy institutions, but its institutional value will depend upon whether delegation remains accompanied by traceability, contestability and clearly bounded authority. CPDF therefore treats computational agents as extensions of policy cognition whose autonomy must remain functionally differentiated from legitimate institutional agency.

29. Failure Modes of Computational Policy Design

The increasing sophistication of computational policy systems creates new capacities but also new pathways through which institutional cognition can fail. Some failures originate within computational models themselves, while others arise from the relationships connecting models to evidence, deliberation, implementation and authority. CPDF therefore requires a vocabulary capable of distinguishing different failure mechanisms rather than treating all problematic outcomes as generic algorithmic error.

Representation failure occurs when the computational representation omits or mischaracterises important dimensions of the public problem. The model may be technically accurate within its defined boundaries while those boundaries exclude mechanisms essential to policy outcomes. Such failures are particularly difficult to detect when the representation has become institutionally dominant.

Evidence failure arises when the information entering computational processes is incomplete, biased, outdated or otherwise inadequate. The problem may concern data quality, but it can also reflect evidence architecture: relevant qualitative knowledge or institutional experience may exist without becoming computationally accessible.

Objective failure occurs when the criteria according to which policies are evaluated misrepresent the values or outcomes institutions actually seek to protect. Optimisation can then become highly effective at producing results that are institutionally undesirable because the computational objective has become detached from the broader purpose of governance.

Constraint failure appears when important physical, administrative, legal, constitutional or normative boundaries are absent, incorrectly represented or treated as more flexible than they legitimately are. A computationally attractive intervention can consequently emerge from a possibility space that contains options unavailable to legitimate institutions.

Model failure concerns incorrect assumptions, relationships or parameterisations within the computational system itself. Unlike representation failure, which concerns the broader framing of the problem, model failure occurs when the chosen representation does not behave adequately according to the phenomena it attempts to capture.

Interpretation failure occurs when technically valid outputs are misunderstood, uncertainty is ignored or model results are attributed a degree of authority they do not possess. Sophisticated computation can therefore produce poor policy cognition even when the underlying analysis is methodologically sound.

Integration failure arises when computational outputs remain disconnected from the institutional processes that could use them effectively. This mirrors the architectural failures identified within ICAM: analytical capacity can exist without becoming collectively consequential because interfaces between computation and institutional judgement remain weak.

Finally, **authority failure** occurs when computational contributions acquire forms of effective decision power that exceed their legitimate institutional role. This can happen explicitly through automated decision-making, but it can also occur implicitly when human decision-makers become unable or unwilling to challenge computational recommendations.

These failure modes can interact. Poor evidence can reinforce representation failure; opaque models can produce interpretation failure; institutional dependence upon computational systems can transform integration problems into authority problems. **The central lesson is that computational policy failure cannot be understood solely as technical malfunction. It can emerge anywhere within the architecture connecting representation, evidence, computation, judgement and legitimate action.**

30. Automation Bias and the Problem of Computational Authority

Among the failure modes identified above, authority failure deserves particular attention because computational systems possess forms of epistemic presentation that can alter institutional behaviour even when no formal decision rights have been delegated to them. Quantitative precision, analytical complexity and the perceived objectivity of computation can cause model outputs to acquire authority beyond what their evidential foundations justify.

This phenomenon is often described through automation bias, but within policy institutions the problem extends beyond individual psychological reliance upon automated recommendations. Organisational structures can themselves become dependent upon computational systems. Procedures may be redesigned around model outputs, staffing patterns may reduce independent analytical capacity and institutional routines may gradually assume that computational recommendations constitute the default position from which deviations require justification.

Under such conditions, formal human authority can coexist with substantial computational influence. A minister, regulator or administrator may retain the legal power to reject a recommendation while lacking the informational resources, time or expertise required to construct an independent judgement. Human approval then risks becoming ceremonial rather than cognitively meaningful.

The opposite response — requiring humans to make every decision independently of computational recommendations — would sacrifice many of the cognitive benefits CPDF seeks to develop. The objective is therefore not to minimise computational influence but to ensure that influence remains proportionate to evidential reliability and embedded within structures of challenge. Independent models, adversarial review, uncertainty representation, provenance records and alternative analytical pathways can all contribute to maintaining meaningful institutional judgement.

The distribution of responsibility becomes particularly important when computational recommendations contribute to harmful outcomes. If institutions cannot reconstruct how evidence, model assumptions and human decisions interacted, accountability can become diffused across technical teams, vendors, administrators and decision-makers. Epistemic traceability therefore supports not only learning but institutional responsibility.

Legitimate authority cannot be preserved merely by keeping a human signature at the end of a computational process. It requires an architecture in which authorised actors retain meaningful capacity to understand the grounds of decisions, interrogate relevant alternatives and assume responsibility for collective choices.

31. Constitutional and Normative Boundaries of Computational Policy Design

The distinction between computational capability and legitimate authority becomes most consequential where policy design encounters constitutional rights, democratic procedures and normative limits upon public power. Computational methods may reveal interventions that appear highly effective according to specified objectives while conflicting with principles that governance institutions are not entitled to optimise away.

Rights provide a clear example. Privacy, equality before the law, freedom of expression, due process and other constitutional protections constrain the range of interventions available to public institutions. Their function is not simply to impose costs upon otherwise efficient policy design but to establish boundaries within which public authority can legitimately operate. Constraint architecture must therefore treat such principles differently from ordinary resource limitations.

Democratic procedure introduces another form of boundary. Some policy choices require public deliberation, legislative authorisation or political accountability precisely because their consequences involve contested values. Computational evidence can inform these processes but cannot legitimately bypass them on the grounds that a model has identified a superior outcome. The existence of analytically compelling evidence does not dissolve institutional requirements concerning who may decide.

Normative boundaries can also be difficult to formalise. Concepts such as fairness, proportionality, dignity or acceptable risk contain interpretative dimensions that cannot always be reduced to stable computational criteria. Attempts to encode them may nevertheless be useful when institutions remain aware that formalisation represents one interpretation rather than the principle itself. Computational representations of normative commitments should therefore remain open to legal, political and ethical challenge.

These boundaries also protect epistemic pluralism. Democratic institutions allow different groups to contest not only policy preferences but representations of social problems themselves. A computational architecture that centralises problem definition within technical systems could narrow this contestability even while improving analytical consistency. Preserving legitimate disagreement therefore becomes part of the design of computational policy institutions.

CPDF consequently treats constitutional and normative constraints as **architectural conditions of legitimate computation**. They define where computational exploration can assist collective reasoning and where decisions must remain governed by forms of authority that computation cannot reproduce. The framework does not regard these boundaries as limitations upon institutional intelligence; they are part of what makes institutional intelligence compatible with legitimate governance.

32. The Limits of Computability in Governance

The development of increasingly powerful computational systems can encourage the assumption that more aspects of governance will eventually become representable, predictable and optimisable as data and models improve. CPDF rejects this assumption as both scientifically uncertain and institutionally unnecessary. Some dimensions of public policy may remain difficult to formalise not because computational capability is temporarily inadequate, but because governance concerns meanings, values and social relationships whose significance cannot be exhausted through computational representation.

Tacit knowledge provides one example. Experienced professionals often recognise contextual patterns that are difficult to articulate formally, while communities possess situated knowledge shaped by histories and relationships that administrative datasets capture only partially. Artificial intelligence may increasingly process unstructured information associated with these forms of knowledge, but processing linguistic representations of experience is not equivalent to possessing the social context from which that experience derives.

Novelty creates another limit. Computational models learn or operate through representations constructed from available evidence and assumptions, while governance sometimes confronts events whose relevant structure differs substantially from historical experience. Models can support reasoning under novelty, but their reliability may decline precisely when institutions most need adaptive judgement.

Normative conflict presents a deeper boundary. No increase in computational capability determines how societies should reconcile competing conceptions of justice, freedom, security, equality or responsibility. Computation can reveal consequences associated with alternative priorities, but the selection among those priorities remains a question of collective judgement.

Political action itself can also change the system being modelled. Citizens, firms and institutions respond to policies, predictions and classifications, sometimes strategically. A computational representation can therefore alter the behaviour it seeks to predict, creating reflexive relationships between governance knowledge and governed populations. Increasing model accuracy does not necessarily remove this reflexivity because actors can adapt to the model itself.

These limits do not establish a fixed frontier beyond which computation has no relevance. Technological capability will continue to expand, and processes currently resistant to formalisation may become increasingly tractable. The principle is methodological rather than technological: **institutions should not assume that what cannot currently be computed is cognitively irrelevant, nor that what can be computed captures the full meaning of the problem being governed.**

Computational Policy Design therefore remains intentionally incomplete. Its strength lies in augmenting selected forms of institutional cognition while preserving interfaces with forms of knowledge, judgement and authority that remain outside computational representation.

33. Data Governance as Cognitive Infrastructure

The computational capabilities described throughout CPDF depend upon infrastructures through which information becomes available, interoperable and institutionally usable. Data governance is therefore not merely a technical or administrative prerequisite for Computational Policy Design; it forms part of the cognitive architecture through which institutions perceive and represent the systems they govern.

The organisation of data influences which phenomena become visible, at what level of granularity and with what temporal delay. Fragmented administrative systems may contain extensive information while preventing it from contributing to integrated policy representations. Standardisation can increase interoperability while removing contextual distinctions important to local interpretation. Centralised data infrastructures can support cross-domain analysis while creating new concentrations of informational power.

Data quality also has architectural dimensions. Errors can originate during collection, classification, integration or transformation, and their consequences may propagate across multiple models once shared datasets become foundational components of policy analysis. Provenance must therefore apply not only to models but to data pipelines, allowing institutions to understand how observations were generated and transformed before entering computational representations.

Privacy and legitimate access impose essential constraints. The fact that combining datasets could improve predictive or analytical performance does not establish that institutions should be permitted to combine them. Data governance must mediate between cognitive capability and rights, ensuring that computational policy systems remain bounded by legitimate purposes and appropriate protections.

The growing use of external computational services creates further questions concerning institutional dependence. Models and data infrastructures supplied by private organisations may become cognitively central to public institutions while remaining partially opaque or governed by commercial incentives. Institutional cognitive sovereignty may therefore depend increasingly upon the capacity to understand, audit and, where necessary, replace critical computational infrastructure.

Data architecture shapes cognitive architecture because institutions can reason computationally only through the information structures they are capable of constructing and legitimately using. Computational Policy Design must consequently treat data governance as part of policy cognition itself rather than as an invisible technical layer beneath it.

34. Institutional Capacity for Computational Policy Design

The existence of advanced computational methods does not imply that institutions possess the capacity to use them effectively. Computational Policy Design requires organisational capabilities extending well beyond access to software or artificial intelligence. Institutions must be able to formulate meaningful modelling questions, evaluate evidence, interpret uncertainty, maintain data infrastructures, integrate technical and domain expertise and connect computational outputs to legitimate decision processes.

This requirement creates a distinction between **computational capability and institutional computational capacity**. A government may purchase sophisticated analytical systems while lacking the internal expertise required to evaluate their assumptions. External consultants or technology providers may produce advanced models whose outputs cannot be challenged independently by the institution commissioning them. Under such conditions, apparent increases in computational capability can coexist with reductions in institutional cognitive autonomy.

Capacity therefore includes the ability to refuse or modify computational recommendations, to compare alternative models and to recognise when a problem is poorly suited to computational formalisation. An institution dependent upon computation without possessing the expertise necessary to challenge it has acquired technology without acquiring corresponding cognitive capability.

Interdisciplinary integration becomes particularly important. Technical specialists understand modelling methods, but domain experts possess knowledge concerning the systems being represented; legal professionals understand institutional constraints; implementation actors understand operational realities; and political institutions determine legitimate priorities. Computational Policy Design requires interfaces capable of connecting these forms of expertise without subordinating all of them to technical modelling.

Capacity also has temporal dimensions. Models require maintenance, data infrastructures evolve and assumptions that were valid during initial deployment may become obsolete. Institutions therefore need durable organisational memory rather than temporary project teams whose knowledge disappears once a particular policy initiative concludes. Policy Model Provenance and documentation become part of this longer-term capacity.

The development of computational policy institutions should consequently be evaluated not by the sophistication of the technologies they acquire but by **their ability to integrate, interrogate, maintain and govern computational capability as part of institutional cognition**. This distinction will become particularly important for applied pilot programmes, where technological success may otherwise be mistaken for institutional learning.

35. Computational Policy Design as Institutional Augmentation

The arguments developed throughout the paper converge upon a conception of Computational Policy Design as **institutional augmentation**. Computation expands particular cognitive capacities: the amount of evidence that can be processed, the number of policy configurations that can be explored, the complexity of interactions that can be represented and the range of futures across which interventions can be examined. These expansions can substantially alter the conditions under which institutions reason about public problems.

Augmentation nevertheless differs fundamentally from substitution. The framework does not assume that computational systems should progressively replace human or institutional judgement as their technical performance improves. Some functions may indeed become highly automated where objectives are narrow, constraints stable and accountability structures clear, but public policy as a whole contains normative and political dimensions that cannot be reduced to computational performance.

Nor does augmentation imply that human cognition remains unchanged while computational capability is simply added around it. New tools alter institutional roles, professional practices and distributions of expertise. Decision-makers may reason differently when they can interactively explore scenarios; analysts may shift from constructing individual models towards supervising computational ecosystems; citizens may encounter policy debates increasingly mediated by simulations and data.

Computational augmentation therefore transforms the architecture in which human judgement occurs.

The quality of that transformation depends upon institutional design. Computational systems can expand cognitive diversity or concentrate epistemic authority; improve traceability or create opacity; strengthen institutional memory or generate dependence upon external infrastructures. Technology does not determine which of these outcomes occurs. The surrounding cognitive and governance architecture remains decisive.

This brings CPDF back to the theoretical foundation established through ICAM. **Computational capability becomes institutional intelligence only when it is integrated through architectures capable of connecting evidence, exploration, judgement, authority, implementation and learning.** The relevant object of research is consequently not the computational system in isolation but the institutional configuration through which its capabilities become collectively consequential.

Computational Policy Design can therefore be understood as one domain within the broader programme of Institutional Cognition: the study and development of architectures through which computational methods enlarge the policy reasoning available to institutions without displacing the legitimate processes through which societies determine what those institutions should ultimately do.

36. CPDF as a Research Framework

The development of Computational Policy Design as a scientific framework requires moving beyond the conceptual plausibility of computational augmentation towards systematic investigation of how different computational configurations affect real policy processes. The architecture developed throughout this Working Paper identifies a series of relationships that may become progressively researchable: the way problem representations influence the policy alternatives institutions are capable of considering, the consequences of alternative evidence architectures, the role of model pluralism under uncertainty, the effects of computational exploration upon deliberation, the relationship between model provenance and institutional learning, and the conditions under which computational capabilities become integrated into legitimate decision processes. CPDF therefore provides not a completed operational system but a structured field within which more specific hypotheses can be formulated.

The first methodological challenge concerns reconstruction. Researchers must be capable of observing how computational policy design actually occurs within institutions rather than relying exclusively upon formal descriptions of modelling processes. A government may possess advanced analytical tools whose outputs play only a marginal role in decision-making, while comparatively simple computational systems may become cognitively central because they structure how problems are framed or how alternatives are compared. Understanding these relationships requires attention to the wider socio-technical architecture in which models operate, including the actors who select evidence, define objectives, interpret outputs and determine whether computational findings acquire institutional significance.

Process tracing may therefore become particularly valuable. Rather than assessing computational systems only through technical performance, researchers can reconstruct how particular policy representations were developed, which forms of evidence entered them, how alternative scenarios were generated, how uncertainty was communicated and how outputs influenced deliberation. Such analysis could reveal where computational capacity genuinely expanded policy reasoning and where it merely reproduced assumptions already embedded within the institutional process.

Comparative research provides another important pathway. Institutions may use similar computational methods within very different governance architectures, producing substantially different cognitive consequences. A model embedded within a technically autonomous regulatory agency may perform a different institutional role from the same model used within a highly politicised ministerial process. Comparative investigation can therefore help distinguish properties arising from computational methods themselves from those produced by the architectures into which they are integrated.

The framework also lends itself to experimental and quasi-experimental research. Policy teams could compare conventional design processes with computationally augmented alternatives, examine whether wider possibility spaces generate meaningfully different policy options, or test whether explicit uncertainty representation changes institutional judgement. Such research must remain attentive to the difficulty of isolating cognitive processes within real organisations, but carefully designed studies could begin providing evidence concerning mechanisms that remain primarily theoretical at this stage.

CPDF therefore functions as a research framework by identifying **where computation enters policy cognition, what relationships may be altered by that entry and which institutional consequences should become objects of systematic observation**. Its scientific development will depend upon whether these distinctions generate findings that remain meaningful across different policy domains, institutional contexts and technological configurations.

37. Progressive Validation of Computational Policy Design

The validation of CPDF cannot depend upon demonstrating that computationally augmented policy processes consistently outperform conventional ones, because the framework does not make such a universal claim. Computational methods may improve some forms of institutional cognition under particular conditions while generating new weaknesses under others. Validation must therefore examine whether the relationships identified by CPDF help explain when computational augmentation expands policy reasoning, when it fails to do so and which architectural conditions influence the difference.

One pathway concerns representational validity. Researchers can examine whether computational representations capture the dimensions of public problems that prove relevant during implementation and whether model pluralism identifies uncertainties that would otherwise remain hidden. Where policies fail, retrospective analysis can ask whether the failure originated in the computational model, the evidence architecture, the institutional interpretation of outputs or the original framing of the problem. Such distinctions would provide a stronger test of CPDF than simply comparing predicted and observed outcomes.

Policy-space exploration can likewise be investigated empirically. If computation is expected to enlarge the set of alternatives institutions can examine, research should determine whether computationally assisted processes actually produce more diverse, feasible or substantively different interventions, or whether generated alternatives merely reproduce existing policy paradigms in more elaborate forms. The distinction between combinatorial expansion and genuine policy imagination will be especially important as generative AI becomes more widely integrated into public-sector analysis.

The relationship between computation and deliberation provides another validation domain. Computationally expanded possibility spaces may improve institutional reasoning when they expose trade-offs and uncertainty, but they may reduce deliberative quality when technical complexity concentrates epistemic authority within specialised groups. Comparative studies can therefore examine whether particular interface designs increase the capacity of non-technical decision-makers to interrogate assumptions and understand consequences without oversimplifying the underlying models.

Longitudinal analysis will be important for policy learning. CPDF proposes that implementation evidence can modify not merely policy parameters but the representations from which policy design begins. Researchers can therefore examine whether institutions using computational frameworks revise their models, evidence architectures and assumptions more systematically over time than comparable institutions relying upon more static analytical processes. Policy Model Provenance may provide especially useful empirical material for reconstructing such changes.

Simulation and controlled experimentation can complement these forms of evidence by isolating mechanisms that are difficult to observe directly. Yet computational experiments cannot validate CPDF independently, because a framework concerned with the integration of computation into institutions must eventually encounter institutional behaviour itself. **The strongest validation will therefore emerge through convergence among model analysis, comparative institutional research, implementation evidence and carefully designed policy experiments**, with disagreement among these sources treated as an opportunity for theoretical refinement rather than as a problem to be concealed.

38. Methodological Cautions in Computational Policy Research

Research into Computational Policy Design faces a methodological difficulty created by the very technologies it seeks to study: computational processes often generate far more visible and structured evidence than the human and organisational processes surrounding them. Models produce logs, outputs, parameters and version histories, while informal judgement, political interpretation and tacit expertise may leave much weaker empirical traces. Researchers could therefore overestimate the role of computation simply because computational activity is easier to observe.

A related problem arises from retrospective coherence. Once a policy decision has been made, institutional actors may reconstruct the modelling process as more systematic than it actually was, attributing influence to particular analytical outputs while underestimating the role of informal negotiations or prior political commitments. Process tracing must therefore distinguish between the formal architecture of computational policy design and the realised architecture through which the policy actually emerged.

Technological novelty introduces another methodological risk. Institutions may adopt new computational systems precisely in areas receiving high organisational attention, producing correlations between computational sophistication and policy performance that reflect broader institutional investment rather than the effects of computation itself. Comparative research will need to control, as far as possible, for differences in resources, leadership, administrative capability and policy complexity.

The rapid evolution of AI also complicates longitudinal research. Systems available at the beginning of a study may differ substantially from those used several years later, making it difficult to attribute institutional changes to a stable technological intervention. This reinforces the importance of studying cognitive functions and architectural relationships rather than particular products or model generations. CPDF should remain technologically open precisely because the underlying scientific questions concern how computation participates in policy cognition, not whether one contemporary tool performs better than another.

Measurement can create additional distortions. Indicators of computational adoption, number of models used, data volume or predictive accuracy may be relatively easy to construct while providing limited information about institutional cognitive quality. A policy institution using fewer computational systems may display stronger judgement and learning if those systems are better integrated into deliberation and feedback.

Research must therefore avoid substituting technological intensity for institutional capacity.

Finally, policy research occurs within politically consequential environments. Models may become associated with contested decisions, and access to internal assumptions or implementation data may be restricted. Researchers must remain sensitive to confidentiality, public accountability and the possibility that institutional participants have strategic incentives concerning how computational processes are represented. **Methodological credibility will depend upon preserving the distinction between studying computational capability and endorsing the decisions institutions make with it.**

39. CPDF in Relation to ICCA and ICAM

The development of CPDF becomes most intelligible when situated within the cumulative architecture established by the preceding L02 Working Papers. ICCA provided the conceptual structure through which Institutional Cognition could be understood as a field containing foundational, functional, architectural, operational and evolutionary dimensions. ICAM then translated selected aspects of that architecture into a model of how cognitive functions become distributed across nodes, interfaces, flows and feedback structures within institutions. CPDF represents the next level of specification by examining how computational capability can participate within one particularly important institutional cognitive process: the design and revision of public policy.

This progression allows Computational Policy Design to inherit several principles without needing to reconstruct them independently. The distributed nature of cognition explains why computational models cannot be analysed in isolation from human and organisational actors. Cognitive interfaces explain why technical capability matters only when outputs can become integrated into institutional judgement. Feedback architecture explains why implementation evidence must be capable of reaching modelling and representation processes if computational policy systems are to learn. The distinction between cognitive capability and legitimate authority provides the constitutional boundary within which computational exploration remains subordinate to accountable collective choice.

CPDF also contributes concepts that can flow back into the broader architecture. Computational legibility bias extends the analysis of institutional perception by identifying how technological representation may influence what institutions are capable of seeing. Policy possibility spaces and trade-off spaces provide specific operational forms of cognitive exploration. Epistemic traceability and Policy Model Provenance deepen the relationship between institutional memory and computational reasoning. Adaptive policy pathways provide a concrete expression of recursive institutional learning within policy processes.

The relationship among the three models is therefore cumulative without being hierarchical. ICCA does not dictate every distinction within ICAM, nor does ICAM determine the detailed architecture of CPDF. Each Working Paper introduces concepts appropriate to its analytical level while preserving traceable relationships with the preceding framework. Later empirical research may then require revision at one or several levels simultaneously.

This layered development illustrates the broader logic of the L02 generation. **The research programme is not attempting to build one comprehensive model of governance, but a family of interoperable frameworks whose increasing specificity allows different dimensions of Institutional Cognition to become progressively investigable without losing their conceptual connection to the wider field.**

40. CPDF and the Emerging Operational Model Suite

Within the emerging Operational Model Suite, CPDF occupies a position between architectural theory and the more specialised models concerned with human–AI collaboration, measurement, simulation and applied experimentation. Its contribution lies in defining a computationally augmented policy-design process that those subsequent models can examine from different perspectives.

The Human–AI Institutional Collaboration Architecture will extend questions that CPDF has deliberately left open concerning the distribution of cognitive tasks between human and artificial agents. Computational Policy Design establishes where AI may participate — evidence integration, exploration, scenario development or modelling — but E2 must investigate how responsibilities, verification, disagreement and oversight should be organised across human–AI configurations.

The Governance Cognitive Performance Indicator Framework will inherit several candidate dimensions from CPDF, including representational quality, epistemic traceability, uncertainty handling, policy-space diversity and feedback incorporation. Yet F2 will need to determine whether these properties can become meaningful observables and how they relate to broader institutional performance.

Governance Simulation Platforms will provide the computational environments through which many of the modelling relationships described by CPDF can be explored more systematically. CPDF establishes why simulation matters for policy design; G2 will investigate how simulation environments should be architected as research infrastructures capable of representing both governed systems and governing systems.

AI Governance Pilot Programmes will later provide an applied route for testing particular CPDF configurations within real institutions. Pilot projects could examine AI-assisted evidence synthesis, computational scenario exploration or alternative feedback architectures, but their scientific value will depend upon the experimental principles developed in I2.

The relationship with these future models reinforces a principle established throughout the L02 generation: **specialisation should increase methodological precision without producing conceptual isolation**. CPDF is valuable not because it contains every element required for computational governance research, but because it establishes the policy-design architecture from which more specialised investigations can proceed.

41. Future Research Directions

The research agenda emerging from CPDF is extensive because many of its central propositions remain theoretical until examined across real policy processes. One immediate priority concerns problem representation. Comparative research could investigate how different computational representations of the same public problem alter the range of policy alternatives considered, whether model pluralism improves institutional understanding and how competing representations are reconciled during deliberation.

The policy possibility space provides another substantial research domain. Future studies could examine whether computational exploration genuinely increases policy diversity, whether broader alternative spaces improve eventual decisions or instead create additional cognitive burden, and how institutions determine when sufficient exploration has occurred. Generative AI makes these questions particularly urgent because the cost of producing plausible alternatives may decline dramatically while the cost of evaluating them remains substantial.

Computational legibility bias also deserves empirical investigation. Researchers could compare which forms of evidence gain influence after computational systems are introduced and whether qualitative, local or tacit knowledge becomes less institutionally visible. Such studies would help determine whether epistemically plural evidence architectures can mitigate the narrowing effects of formalisation.

Uncertainty representation offers another avenue. Policy experiments could compare decision processes in which computational systems present point estimates with those presenting scenario ranges, model disagreement or structured uncertainty, examining whether different representations alter institutional judgement, risk perception or willingness to adapt.

Policy Model Provenance may support longitudinal research into institutional learning. By preserving the assumptions and evidence underlying successive model versions, institutions could provide unusually rich records of how policy understanding changes over time. Researchers could examine whether explicit provenance increases the capacity to diagnose failure or prevents recurrent mistakes.

Adaptive policy pathways create opportunities for studying policies designed explicitly around uncertainty and future revision. Their performance could be compared with more static interventions, particularly in domains characterised by rapid technological or environmental change. Such research would need to distinguish genuine adaptive learning from continual policy instability.

Finally, the institutional capacity required for CPDF deserves dedicated investigation. Computational capability may increasingly become available through external providers, but the development of genuine institutional computational capacity may depend upon internal expertise, governance structures, procurement models, data sovereignty and long-term organisational memory. **The future of Computational Policy Design will therefore depend as much upon institutional development as upon technological progress.**

42. Scientific Contribution of the Computational Policy Design Framework

The principal contribution of CPDF is to reconceptualise computation within public policy as an expansion of institutional cognitive space rather than as a mechanism for automating collective choice. Computational systems can allow institutions to represent complex problems more explicitly, integrate larger bodies of evidence, explore wider policy possibility spaces, examine trade-offs across multiple objectives, investigate uncertainty through plural scenarios and connect implementation evidence more closely with subsequent redesign. These capabilities can substantially enlarge the field within which policy reasoning occurs.

The framework simultaneously establishes why such enlargement cannot determine legitimate outcomes by itself. Public policy contains objectives that conflict, constraints that are constitutional rather than merely technical and forms of knowledge that cannot always be formalised adequately. Computational models depend upon representations whose boundaries shape the interventions they make visible, and their outputs remain subject to uncertainty, interpretation and political judgement. CPDF therefore embeds computational capability within an institutional architecture that preserves deliberation, epistemic plurality, traceability and legitimate authority.

Several concepts developed in the paper make this architecture more explicit. The policy possibility space distinguishes exploration from selection; constraint architecture clarifies why computational possibility does not imply institutional admissibility; trade-off spaces shift analysis away from single-objective optimisation; computational legibility bias identifies the epistemic risks created by formalisation; Policy Model Provenance provides a basis for institutional memory around computational reasoning; and recursive policy design connects representation, implementation and feedback into a continuous learning process.

The framework also establishes a more precise relationship between technology and institutional capacity. Access to sophisticated computational systems does not necessarily increase institutional intelligence if organisations cannot interrogate models, preserve provenance, integrate domain expertise or challenge computational recommendations. Computational capability becomes institutionally valuable only through architectures capable of transforming analytical expansion into accountable judgement.

The broader scientific contribution is therefore to establish **Computational Policy Design as a distinct object of Institutional Cognition research**, positioned between conventional policy analysis and more ambitious ideas of automated governance. It provides a framework through which computation can be studied as part of the cognitive organisation of policy-making while preserving the political, normative and constitutional dimensions that make public policy more than a technical optimisation problem.

43. Closing Perspective — Expanding the Space of Institutional Reasoning

The increasing computationalisation of governance creates an understandable temptation to frame the future of policy-making around prediction, automation and optimisation. If institutions could process enough data, model enough relationships and anticipate enough consequences, it might appear that many difficult policy questions could gradually be transformed into technical problems whose solutions become discoverable through computation. The analysis developed throughout this Working Paper suggests a different trajectory.

The greatest potential contribution of computation may lie not in reducing the need for institutional judgement but in expanding the environment within which that judgement can operate. Computational systems can reveal alternatives that would otherwise remain unexplored, expose relationships hidden by limited human attention, test policies across multiple possible futures and make trade-offs more visible. Yet each expansion creates additional questions concerning representation, evidence, uncertainty, values and authority. More computation can therefore produce more informed governance only when institutions also develop the architectures required to understand what computation makes visible and what it leaves outside the model.

This relationship explains why Computational Policy Design is fundamentally an Institutional Cognition problem. Models do not design policy independently; they participate within systems containing professional expertise, organisational memory, legal structures, political deliberation and legitimate authority. Their contribution depends upon cognitive interfaces connecting these elements, while their limitations become consequential when institutions allow computational representations to substitute silently for the realities they were designed to approximate.

CPDF therefore rejects both technological pessimism and computational determinism. Institutions need not avoid powerful computational systems because those systems are imperfect, nor should they assume that increasing technical sophistication will progressively eliminate the uncertainty and disagreement inherent in governance. The relevant scientific task is to understand how computational capacities can be organised so that they extend collective reasoning while remaining open to correction, contestation and revision.

The recursive architecture of Computational Policy Design provides one possible foundation for that task. Problems are represented, alternatives explored, consequences examined, choices authorised, interventions implemented and resulting evidence returned to the processes through which subsequent policies become imaginable. In this architecture, computation contributes throughout the cycle without becoming its sovereign centre.

The transition established by CPDF is therefore not from human policy-making towards computational policy-making, but from cognitively constrained policy processes towards richer forms of institutional reasoning in which

computational exploration and legitimate judgement remain deliberately connected without becoming indistinguishable.

About the IDHUS Working Papers

This publication forms part of the constitutional publication ecosystem of the IDHUS Institute. It should not be interpreted as an isolated academic paper but as one component within a progressively expanding scientific architecture dedicated to the development of the emerging discipline of **Institutional Cognition**.

Within the organisational architecture of the IDHUS Institute, the Working Paper occupies a distinctive position. It comprises the elementary cognitive module through which operational research is conducted. At the structural level, however, the Working Paper also functions as the smallest autonomous organisational unit capable of generating identifiable scientific contributions within the broader research ecosystem. For the way we conduct our research, it represents the point at which constitutional concepts, Research Series, Research Lines, methodologies, computational developments and empirical investigations converge into a coherent research product capable of contributing directly to the cumulative evolution of institutional knowledge.

Series A — Foundations of Institutional Cognition

Series A preserves the constitutional foundations of the entire research ecosystem. Rather than constructing new constitutional theories, it continuously refines, clarifies and strengthens the conceptual architecture established throughout *The IDHUS Method*, the *Research Validation Papers* and the *Research Atlas*. It provides the common scientific language that allows every other Research Series to remain conceptually coherent while supporting the long-term evolution of Institutional Cognition.

Research Lines

A1 — Constitutional Evolution of Institutional Cognition
A2 — Conceptual Architecture and Theory Integration
A3 — Epistemology of Institutional Intelligence
A4 — Philosophy of Adaptive Governance
A5 — Scientific Foundations of Planetary Cognitive Civilisation

Series B — Cognitive Architectures

Series B investigates how cognition emerges and operates across multiple organisational scales, from individuals and institutions to artificial intelligence systems and distributed governance environments. Research within this Series explores organisational learning, collective intelligence, institutional memory, cognitive networks and adaptive decision-making, providing many of the theoretical and computational models that support the broader research programme.

Research Lines

B1 — Individual and Collective Cognition
B2 — Institutional Cognitive Architectures
B3 — Distributed Intelligence Systems

B4 — Human-AI Cognitive Collaboration
B5 — Organisational Learning Architectures

Series C — Computational Governance

Series C examines how artificial intelligence, computational modelling and digital infrastructures are transforming governance and public administration. Its objective is to understand how computational systems become active components of institutional cognition through intelligent decision-support, AI-native public administration, digital governance architectures and adaptive regulatory systems capable of responding to increasing societal complexity.

Research Lines

C1 — AI-Augmented Public Administration
C2 — Computational Policy Design
C3 — Intelligent Decision-Support Systems
C4 — Algorithmic Governance Architectures
C5 — AI-Native Institutional Ecosystems

Series D — Planetary Systems

Series D expands the scope of research from individual institutions towards planetary-scale systems of governance and coordination. Drawing upon systems thinking, complexity science and computational simulation, it investigates global governance networks, civilisational dynamics, resilience, distributed coordination and the cognitive infrastructures required to understand increasingly interconnected planetary systems.

Research Lines

D1 — Global Cognitive Systems
D2 — Planetary Governance Networks
D3 — Civilisational Systems Dynamics
D4 — Global Coordination Mechanisms
D5 — Planetary Resilience Architectures

Series E — AI and Institutional Intelligence

Series E explores the relationship between human intelligence, institutional cognition and artificial intelligence. Rather than studying AI as an isolated technology, it investigates how hybrid cognitive systems emerge through the interaction of people, organisations and intelligent computational agents. This Series provides the principal research environment for understanding the future evolution of AI-enabled institutions and distributed governance.

Research Lines

E1 — Institutional Intelligence
E2 — Human-AI Collaboration
E3 — AI Governance Systems
E4 — Autonomous Institutional Agents
E5 — Constitutional AI for Public Governance

Series F — Cognitive Metrics and Measurement

Series F develops the methodological foundations required to observe and evaluate Institutional Cognition. It designs metrics, indicators, assessment frameworks and measurement methodologies capable of transforming constitutional concepts into operational research variables. These tools support empirical research, comparative analysis, validation programmes and evidence-based governance across the entire research ecosystem.

Research Lines

F1 — Institutional Intelligence Metrics
F2 — Governance Performance Indicators
F3 — Cognitive Capacity Assessment
F4 — Adaptive Systems Measurement
F5 — Planetary Governance Metrics

Series G — Simulation and Digital Twins

Series G establishes the computational laboratories of the Institute. Through simulation environments, digital twins, systems dynamics and agent-based modelling, it creates experimental platforms for exploring institutional behaviour under controlled conditions. These environments allow researchers to investigate alternative governance architectures, analyse complex adaptive systems and evaluate future scenarios before they emerge in practice.

Research Lines

G1 — Institutional Digital Twins
G2 — Governance Simulation Platforms
G3 — Multi-Agent Governance Models
G4 — Scenario Generation and Strategic Foresight
G5 — Computational Experimentation

Series H — Comparative Civilisational Studies

Series H investigates how different societies, institutions and historical periods have organised governance, knowledge and collective intelligence. By comparing diverse civilisational experiences, governance models and institutional traditions, it seeks to identify recurring cognitive principles while understanding the contextual diversity through which Institutional Cognition evolves across cultures and historical epochs.

Research Lines

H1 — Comparative Institutional Cognition
H2 — Civilisational Governance Evolution
H3 — Cross-Cultural Adaptive Systems
H4 — Historical Institutional Intelligence
H5 — Comparative Planetary Governance

Series I — Applied Governance Laboratories

Series I serves as the operational bridge between scientific research and institutional practice. It develops collaborative research environments in which conceptual models, computational systems and governance methodologies can be explored through pilot projects, organisational experimentation, innovation programmes and long-term partnerships with public institutions. Practical experience generated through these laboratories continuously feeds back into the broader research programme.

Research Lines

I1 — Public Sector Innovation Laboratories
I2 — AI Governance Pilot Programmes
I3 — Institutional Transformation Frameworks
I4 — Governance Prototyping and Testing
I5 — Living Governance Laboratories

Series J — Future Research Frontiers

Series J ensures that the research ecosystem remains permanently open to future scientific developments. It explores emerging technologies, new governance paradigms, evolving AI capabilities and other frontier topics whose long-term significance may not yet be fully understood. By institutionalising organised exploration, this Series enables the IDHUS Institute to adapt continuously to future scientific opportunities while preserving the constitutional coherence of its broader research architecture.

Research Lines

J1 — Emerging AI Paradigms
J2 — Future Institutional Architectures
J3 — Post-Governance Systems
J4 — Civilisational Foresight
J5 — Unknown Research Horizons

Internal Constitutional References

The present Working Paper forms part of the constitutional and operational architecture of the IDHUS Institute. It should be read in conjunction with the following foundational publications, which collectively establish the constitutional framework within which the Working Papers Programme develops.

Constitutional Framework

- **The IDHUS Canon**
- **The IDHUS Method**

Programme I Research Validation Papers

- **RVP-001 Operational Cognitive Architecture**
- **RVP-002 Operational Consciousness**
- **RVP-003 Collective Cognition**
- **RVP-004 Institutional Cognition**
- **RVP-005 Governance Intelligence**
- **RVP-006 Adaptive Public Administration**
- **RVP-007 AI-Augmented Public Administration**
- **RVP-008 Adaptive Public Policy**
- **RVP-009 Cognitive Public Services**
- **RVP-010 Self-Evolving Governmental Systems**

Research Atlas

- **Volume 0 Research Ecosystem Architecture**
- **Volume 1 Research Architecture**
- **Volume 2 Planetary Research Ecosystems**
- **Volume 3 Planetary Cognitive Systems**
- **Volume 4 Planetary Governance Architecture**
- **Volume 5 Civilisational Evolution**
- **Volume 6 Planetary Cognitive Civilisation**

Operational Research Programme

- **Operational Research Programme- Working Papers Framework**

These publications together establish the constitutional foundations upon which the operational research developed throughout the IDHUS Working Papers Programme is constructed.

Glossary

Adaptive Policy Pathway

A policy design approach in which future interventions or modifications are anticipated and linked to observable conditions, allowing institutional action to evolve as uncertainty is progressively reduced or circumstances change. Adaptive policy pathways treat policy as a temporally structured sequence of possible decisions rather than as a single irreversible intervention.

Artificial Intelligence within Computational Policy Design

The use of machine learning, generative systems, foundation models and related artificial intelligence capabilities to contribute to particular cognitive functions within computationally augmented policy processes. Within CPDF, AI constitutes one family of computational capabilities rather than the defining principle of Computational Policy Design.

Computational Agent

A computational system capable of performing extended sequences of delegated analytical or exploratory tasks, potentially including evidence retrieval, alternative generation, model comparison, scenario exploration or iterative analysis. Computational agents may expand institutional exploratory capacity but do not thereby acquire legitimate policy authority.

Computational Capability

The technical capacity of computational systems to perform functions relevant to policy cognition, including data processing, modelling, prediction, optimisation, simulation, evidence synthesis and alternative generation. Computational capability remains distinct from an institution's capacity to integrate, govern and critically evaluate those systems.

Computational Exploration

The systematic use of computational methods to investigate alternative problem representations, policy configurations, assumptions, scenarios, trade-offs and possible consequences. Computational exploration expands the field available to institutional reasoning without independently determining which outcomes should be collectively preferred.

Computational Legibility

The degree to which a phenomenon, form of evidence or institutional process can be represented in a form accessible to computational processing. Computational legibility depends upon categorisation, formalisation, data availability and the representational capacities of the computational systems involved.

Computational Legibility Bias

The tendency of computationally intensive institutions to privilege phenomena and forms of knowledge that are easier to formalise, quantify or process while potentially reducing the visibility or influence of contextual, qualitative, tacit or otherwise computationally resistant knowledge.

Computational Policy Design

The structured augmentation of institutional policy cognition through computational methods that support problem representation, evidence integration, policy-space exploration, consequence modelling, uncertainty exploration and recursive policy learning. Computational Policy Design expands institutional reasoning while preserving legitimate collective choice within accountable governance processes.

CPDF — Computational Policy Design Framework

The framework developed in this Working Paper for analysing how computational capabilities can participate within institutional policy design. CPDF represents the relationships among problem representation, evidence, policy possibility spaces, constraints, objectives, trade-offs, uncertainty, institutional judgement, implementation and feedback as components of a recursive socio-technical policy-design architecture.

Constraint Architecture

The configuration of physical, economic, institutional, legal, constitutional, normative and political constraints that structure the boundaries of a policy possibility space. Constraint architecture distinguishes what may be computationally conceivable from what is materially possible, institutionally feasible or legitimately admissible.

Deliberative Accessibility

The degree to which relevant institutional actors can meaningfully understand, interrogate and contest computational representations and outputs used within policy deliberation. Deliberative accessibility does not require equivalent technical expertise among all participants but requires interfaces through which computational complexity does not eliminate meaningful institutional scrutiny.

Evidence Architecture

The configuration through which different forms of evidence become available to and integrated within policy cognition. Evidence architecture may include quantitative data, scientific literature, administrative knowledge, institutional memory, qualitative research, expert judgement, citizen knowledge, legal interpretation and implementation experience.

Epistemic Traceability

The capacity to reconstruct how a computational policy output emerged from particular evidence, assumptions, variables, transformations, objectives, constraints and modelling processes. Epistemic traceability makes computational contributions inspectable within the wider process of institutional reasoning.

Implementation Feedback

Evidence generated through the operation of a policy in real institutional and social environments and returned to subsequent processes of evaluation, modelling and policy redesign. Implementation feedback may reveal weaknesses in policy parameters, implementation capacity, causal assumptions or the original representation of the problem.

Institutional Computational Capacity

The organisational capacity to integrate, interpret, interrogate, maintain and govern computational capabilities within institutional processes. Institutional computational capacity includes technical expertise, domain knowledge, data governance, organisational memory, model evaluation and the ability to challenge or reject computational recommendations.

Institutional Judgement

The interpretative and evaluative processes through which computational evidence, competing objectives, uncertainty, normative considerations and contextual knowledge are translated into institutionally meaningful choices. Institutional judgement remains connected to structures of responsibility and legitimate authority rather than being reducible to computational optimisation.

Learning before Scale

The principle that uncertain or novel policy interventions may benefit from staged implementation, pilots or other forms of experimentation before broad deployment. Learning before scale allows implementation evidence to test assumptions and improve policy representations before interventions become deeply embedded across institutional systems.

Model Failure

A failure arising when the assumptions, relationships, parameters or computational behaviour of a model inadequately represent the phenomena it is intended to analyse. Model failure is distinguished within CPDF from broader representation failure, which concerns the framing and boundaries of the policy problem itself.

Model Pluralism

The deliberate use and comparison of multiple computational representations of a public problem in order to expose assumptions, identify model-dependent conclusions and investigate epistemic uncertainty. Model pluralism does not imply equal validity among models but recognises the potential value of several partial representations under conditions of complexity.

Objective Failure

A Computational Policy Design failure arising when the criteria used to evaluate or optimise policy alternatives inadequately represent the outcomes, values or public purposes that institutions legitimately seek to advance.

Policy Digital Twin

A continuously or periodically updated computational representation of selected aspects of a governed system used to explore possible interventions, monitor changing conditions and compare observed behaviour with modelled expectations. A policy digital twin is a selective and revisable representation rather than a complete digital replica of society.

Policy Experimentation

The structured testing of policy interventions or assumptions through pilots, phased implementation, controlled trials, regulatory sandboxes, regional variation or other empirical approaches capable of generating evidence before or during broader implementation.

Policy Model Provenance

The structured preservation of the origins, evidence, assumptions, transformations, validation history, versions and decision contexts associated with a computational policy model. Policy Model Provenance provides institutional memory concerning the historical and evidential identity of the model across time.

Policy Possibility Space

The broader set of intervention configurations that can potentially be explored within a particular policy domain. Computational methods may expand institutional access to this space by allowing larger combinations of policy parameters and instruments to be examined systematically.

Policy Representation

A selective institutional representation of a public problem through which relevant phenomena, variables, relationships, populations and causal assumptions become sufficiently structured to support analysis and intervention. Within CPDF, the representation remains analytically distinct from the public reality it seeks to describe.

Policy Robustness

The capacity of a policy or policy pathway to maintain acceptable performance across a range of plausible conditions, scenarios or modelling assumptions. Robustness shifts attention away from maximum expected performance under a single forecast towards vulnerability when institutional expectations prove incorrect.

Policy Trade-off Space

The structured domain of policy alternatives characterised by different combinations of outcomes across multiple objectives. A policy trade-off space allows institutions to examine how improvements in one dimension relate to costs or deterioration in others without assuming that competing public values can always be reduced to a single optimisation function.

Recursive Computational Policy Design

A policy-design architecture in which representation, evidence integration, policy-space exploration, consequence exploration, institutional judgement, legitimate choice, implementation and feedback remain recursively connected. Evidence generated through implementation may therefore modify not only policy parameters but the representation from which subsequent design begins.

Representation Failure

A failure occurring when the conceptual or computational representation of a public problem omits, distorts or mischaracterises dimensions that prove consequential for policy understanding or intervention.

Scenario Exploration

The structured examination of policy behaviour across multiple plausible future conditions rather than reliance upon a single predicted future. Scenario exploration supports institutional reasoning under uncertainty by revealing dependencies, vulnerabilities and alternative consequences.

Socio-Technical Policy Design Architecture

The configuration of human actors, organisational structures, computational systems, evidence flows, interfaces and authority relationships through which computationally augmented policy design occurs. The concept emphasises that policy cognition cannot be attributed meaningfully to computational systems in isolation.

Systemic Effects

Consequences of policy interventions that emerge through interactions, feedback processes, behavioural adaptation or relationships across different components of a governed system. Systemic effects may include indirect, distributional, cross-sector, long-term or emergent consequences beyond the immediate target of an intervention.

Uncertainty

The condition in which relevant aspects of policy problems, causal relationships, model structures, parameters, behavioural responses or future environments cannot be known with complete confidence. CPDF treats uncertainty as a structural component of policy design to be represented and explored rather than as a temporary deficiency expected to disappear through computation.

IDHUS
Institute for the Development of Human Societies

