# Stage Gate Decisions Under Uncertainty

## Vulnerability, Robustness, and Dependencies in Sustainability-Oriented Megaprojects

MSc Thesis

Derk de Brauw

Delft University of Technology

# Stage Gate Decisions Under Uncertainty

## Vulnerability, Robustness, and Dependencies in Sustainability-Oriented Megaprojects

by

Derk de Brauw

Master of Science
*Construction Management and Engineering*
Faculty of Civil Engineering & Geosciences
at the Delft University of Technology

| | |
|---|---|
| **Student ID** | 4726731 |
| **Assessment Committee** | Prof. Ir. J.P.G. Hans Ramler |
| | Prof. Dr. Ir. Ruud Binnekamp |
| **Daily Supervisor Aramis** | Ir. Erik van der Vegt, Gasunie - CCS Business Unit |
| **Project Duration** | September, 2025 - March, 2026 |
| **Faculty** | Civil Engineering & Geosciences |
| **Company** | Gasunie |

Cover: Image source: Aramis Launch Stores website,
https://www.aramislaunchstores.com/ (accessed March 2026).
*Not a proper CCS platform; proper CCS platforms don't have helidecks.*

# Preface

This thesis is submitted in partial fulfilment of the requirements for the Master of Science in Civil Engineering, track Construction Management and Engineering, at the Faculty of Civil Engineering and Geosciences of Delft University of Technology.

For this thesis, I had the opportunity, through Gasunie, to contribute to the Aramis CCS project. For this, I am truly grateful. It was exactly the type of environment I was looking for, and it has been an exceptionally valuable experience. I could not have wished for a better context in which to conduct this research.

Aramis is a highly complex and ambitious project, characterised by many first-of-a-kind elements that aims to play a significant role in achieving the Netherlands' climate objectives. As a carbon capture and storage (CCS) initiative, it focuses on reducing industrial $CO_2$ emissions by transporting captured $CO_2$ via pipelines from onshore hubs to depleted offshore gas fields in the North Sea, where it can be permanently stored. The project is developed by a consortium including TotalEnergies, Shell, Energie Beheer Nederland (EBN), and Gasunie with an anticipated capacity of up to 22 million tonnes of $CO_2$ per year, Aramis represents a critical step towards decarbonisation in the Netherlands, particularly for hard-to-abate industries.

I would like to express my sincere gratitude to the entire Aramis project team. It has been a highly enriching environment, filled with knowledge, experience, and a strong sense of purpose. Working alongside such a diverse and dedicated group of professionals has been both inspiring and motivating. As a thesis student, one is aware of the time and attention required from others, while contributing only modestly in return during the process. This makes the responsibility to deliver a meaningful and well-founded result all the more important. I hope that this research contributes to a better understanding of the challenges faced in projects characterised by complexity and uncertainty.

I would like to express my sincere gratitude to my supervisor within the Aramis team, Erik van der Vegt, for his continuous guidance and support throughout this process. His ability to quickly engage with the research, challenge my thinking, and create the right conditions to keep progressing has made a significant difference.

I also extend my thanks to all professionals who took the time to participate in interviews. This includes members of both the Aramis project and the Porthos CCS project. Porthos, as a similarly complex and impactful initiative, provided valuable insights that have contributed significantly to this research.

Finally, I would like to thank my academic supervisors, Hans Ramler and Ruud Binnekamp, from Delft University of Technology. Their guidance, accessibility, and the level of independence they encouraged throughout the process have been greatly appreciated.

Working on projects such as Aramis challenges you to make the most of your potential. I am grateful to have been part of this project.

*Derk de Brauw*
*Delft, March 2026*

# Research Positioning

This section provides context on how the research approach was shaped and how it can be interpreted.

This research did not originate from a theoretical question, but from conversations with professionals working on projects that were struggling, not because the design or engineering were flawed, but because the world around them had changed.

In those conversations, one pattern kept returning. The problems were not located within the project itself, but outside it. Changing regulations, shifting markets, technological developments, supply chain disruptions, geopolitical developments, changes in project ownership - factors on which the project depended but over which no one had real control. In one case, this even led to a project being halted during construction, simply because the underlying business case had disappeared.

What became visible is that projects do not stand on their own. They rely on a set of external conditions that must hold for them to function. As long as those conditions remain intact, a project appears stable. But once they begin to shift, the entire foundation comes under pressure.

This observation says something fundamental about how the reality in which projects operate can be understood. There is clearly an external world that influences projects, but it does not behave in a stable or fully predictable way. Projects are designed based on assumptions about that world, while in practice those assumptions can change.

The first reflex in such situations is often to try to improve prediction. This also became evident in the conversations. How can we better anticipate? How can we know what will happen?

But this is precisely where the tension emerged. These situations did not reveal a lack of prediction, but rather its limits. Some developments proved difficult to anticipate, not because of insufficient data, but because they did not lend themselves easily to being captured in models or predefined categories.

As a result, the way of looking at the problem began to shift. Instead of asking how such developments could be predicted more accurately, the focus moved towards understanding how projects are positioned in relation to a changing environment, and what they implicitly rely on to remain stable over time.

This shift directly influenced the design of the research. An approach focused on measurement and prediction would aim to identify patterns in data and determine which factors lead to success or failure - a quantitative route. Here, a different path was taken. Rather than measuring effects, the emphasis is placed on making underlying structures explicit and open to interpretation.

As a result, the research takes on a qualitative and conceptual character. Not because it focuses on perceptions, but because it aims to make visible and organise relationships that usually remain implicit. Projects are approached as dependency structures: configurations that only function as long as certain external conditions remain in place.

The method that follows is therefore not a traditional analysis of datasets, but a systematic way of identifying, structuring, and interpreting these dependencies. By making them explicit, it becomes possible to understand where projects are vulnerable to changes in their environment.

This choice also determines the type of knowledge the research produces. The outcome is not a predictive model or general law, but a way of understanding projects more clearly - a framework that reveals what a project rests upon and where that foundation may come under pressure.

This also means that different choices could have led to different results. A quantitative approach might have identified statistical relationships, while a more interpretative approach could have focused on actor perspectives. In this research, a position is taken in between: it assumes a real external world, but approaches it by making its underlying structure visible. In doing so, the method makes explicit how internal project configurations are conditionally linked to developments in the external environment, thereby providing a basis for assessing whether projects remain aligned with the conditions required for their continued viability.

The value of this approach does not lie in delivering definitive predictions, but in providing guidance. It makes visible where uncertainty enters the system and how it affects the foundation of a project. This insight makes it possible to recognise earlier when a project begins to lose its alignment with the reality in which it operates.

# Abstract

Megaprojects continue to underperform despite decades of advances in planning, governance, and risk management. At stage-gates, project readiness is typically assessed through the identification and mitigation of discrete risks. While this event-oriented approach is effective for managing identifiable uncertainties, it assumes that relevant uncertainties can be sufficiently articulated at the moment decisions are taken. As a result, forms of uncertainty that are difficult to define - yet critical to project viability - may remain outside formal evaluation.

This research argues that such uncertainty does not primarily reside in isolated events, but in how external developments interact with the dependencies that underpin project viability. Dependencies function as the interface through which external uncertainty connects to the project. When these dependencies are structurally vulnerable and insufficiently stabilised, uncertainty can propagate through the project system before it becomes visible as articulated risk.

Viewing uncertainty through dependencies distinguishes between structural vulnerability and developmental robustness. Vulnerability arises from the position of dependencies within the project configuration. Robustness, in contrast, develops over time as enabling mechanisms stabilise the conditions under which dependencies can function as assumed at the moment of commitment. Project readiness at stage-gates is therefore not determined by the closure of risks alone, but by the extent to which critical dependencies are sufficiently stabilised to carry uncertainty into subsequent stage-gates.

To operationalise this perspective, the research develops the Megaproject Dependency Framework (MDF). This introduces a dependency-based perspective that enables projects to engage with uncertainty before it can be articulated as risk. Starting from the conditions required for project viability, this perspective links external uncertainty to the mechanisms through which the project is configured - such as design choices, sequencing, and contractual arrangements. This makes it possible to identify where uncertainty enters the project, how it may propagate through interdependent elements, and which dependencies are most likely to be affected by changes in the external environment. By combining structural vulnerability with uncertainty propagation, the approach provides a structured basis for identifying where stabilisation is required to support commitment, shifting attention from managing isolated risks to strengthening the conditions under which the project can remain viable.

The framework is applied to two carbon capture and storage megaprojects. The analysis shows that many articulated risks can be traced back to underlying dependencies and their development conditions. High-impact risks are consistently associated with dependencies exhibiting high levels of instability, defined as the combination of structural vulnerability and strong uncertainty propagation. External shocks act as triggers that activate this instability, making previously latent exposure operational.

The findings indicate that risks do not form an independent layer of analysis, but can be interpreted as partial and governance-visible expressions of deeper dependency dynamics - particularly in situations where uncertainty is difficult to articulate in advance. This positions dependency analysis as a precursor to risk articulation, offering a way to engage with uncertainty before it can be meaningfully expressed as discrete events.

This research shows that stage-gates do not eliminate uncertainty, but instead consolidate how exposure is carried forward. Project readiness is therefore best understood as the degree to which dependency instability is sufficiently bounded to support commitment. The MDF provides a complementary analytical lens to existing risk-based approaches, enabling projects to identify where uncertainty is likely to materialise and how it may propagate, without replacing established governance practices.

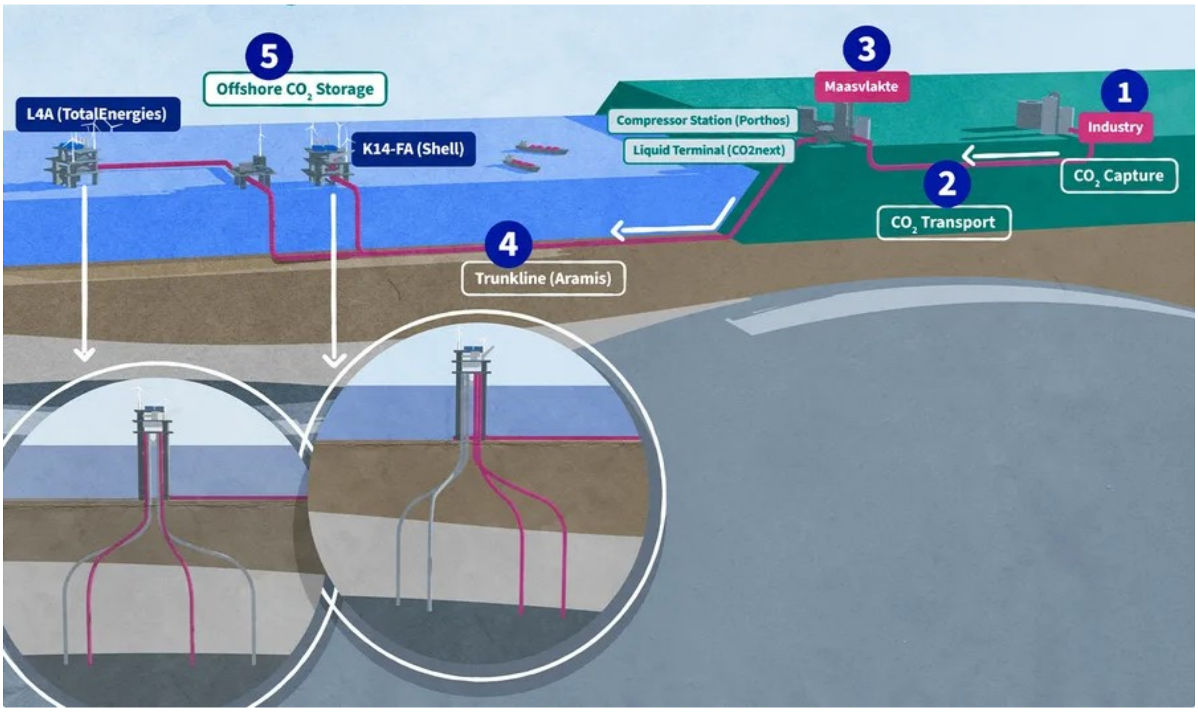


**Figure A:** Conceptual overview of the CCS value chain. This figure provides a simplified representation to orient the reader, illustrating the main stages of $CO_2$ capture, transport, and storage. It is included for contextual understanding only; a detailed system description is provided in Chapter 7 (Media - Gasunie; https://www.gasunie.nl).

# Contents

# 1

# Introduction: Persistent Underperformance and the Problem of Project Readiness

## 1.1. Persistent underperformance in megaprojects

Megaprojects fail systematically. Megaprojects are among the most complex forms of human enterprise. They are typically defined as large-scale ventures exceeding US$1 billion in cost, extending over many years, and involving multiple public and private stakeholders (Flyvbjerg, 2014, 2017). These projects are designed to deliver transformative social and economic outcomes, ranging from energy transitions and transport systems to digital and industrial infrastructure. However, they are also characterised by high failure rates. According to Flyvbjerg and Gardner (2023), only 8.5% of megaprojects hit the mark on both cost and time. And a miniscule 0.5% nail cost, time, and benefits. This introduces the “iron law of megaprojects: over budget, over time, over and over again.”

Importantly, these patterns persist despite decades of advances in project management methodologies, governance structures, and control mechanisms, suggesting that underperformance cannot be explained solely by inadequate execution or managerial shortcomings. Rather, it recurs across sectors, institutional settings, and project types, suggesting that the problem is structural rather than incidental.

Megaprojects are not self-contained endeavours. Their performance depends on conditions that extend beyond the boundaries of the project organisation and are embedded in broader institutional, technical, and socio-political systems. In project management literature, such conditions are often described as dependencies: external feasibility conditions on which project viability depends and that lie partly or fully outside direct project control following Rose et al., 2025. Dependencies therefore identify where project viability is structurally conditioned by its environment, rather than by internal tasks, risks, or assumptions. These dependencies shape the context in which projects operate and ultimately determine whether their intended objectives remain viable over time.

A recurring theme in the megaproject literature is the distinction between procedural success and strategic failure. Projects may comply with formal procedures, pass decision gates and demonstrate progress against predefined performance indicators while, at the same time, drifting away from the strategic conditions that originally justified their initiation (Flyvbjerg et al., 2003; Samset and Volden, 2016). In such cases, governance systems succeed in maintaining procedural order but fail to detect whether the foundational assumptions underpinning the project’s rationale continue to hold. As a result, projects can appear “under control” while becoming increasingly misaligned with their external context.

This dynamic is particularly pronounced in megaprojects characterised by high complexity and interdependence. Delivered through networks of multiple organisations, these projects are exposed to shifting external conditions that affect resources, coordination mechanisms, and institutional alignment

(Denicol et al., 2020). Conventional project management approaches tend to prioritise the management of discrete, quantifiable threats. While such approaches are effective for handling probabilistic risks within predefined boundaries, they offer limited support for understanding how broader external conditions evolve and interact over time (Perminova et al., 2008).

As projects progress, the external dependencies identified during the initial feasibility or decision-making stages often become implicit assumptions. Once implementation is underway, these assumptions are rarely revisited unless triggered by a major disruption or formal review (Samset and Volden, 2016). Governance systems therefore tend to react to visible symptoms - such as cost increases or schedule pressure - rather than to earlier shifts in the conditions that shape project viability. This reactive posture is reinforced by governance cultures that emphasise control, compliance, accountability and performance reporting over reflection and learning (DiMaggio and Powell, 1983; Ross and Staw, 1986).

Taken together, these patterns suggest that the consistent underperformance of megaprojects is not primarily the result of poor execution or inadequate control. Rather, they reflect deeper limitations in how projects account for and engage with the external conditions on which their success depends. Existing governance frameworks provide limited visibility into whether the assumptions underlying a project's strategic logic remains valid as external systems evolve. This absence of structured insight into the ongoing alignment between project objectives and their contextual dependencies forms the starting point for this research.

## 1.2. Sustainability-oriented megaprojects as open systems

Sustainability-oriented megaprojects differ fundamentally from classical megaprojects. Traditional megaprojects, such as bridges, airports, or highways, are typically conceived as large-scale engineering undertakings aimed at delivering a clearly defined physical asset within relatively bounded technical and organizational systems. Their objectives are often stable, their scope can be specified early, and performance is commonly assessed through indicators of time, cost, and quality under a dominant client or consortium structure (Miller and Lessard, 2000).

In contrast, sustainability-oriented megaprojects are transformative in nature. Instead of delivering stand-alone infrastructure assets, they are designed to contribute to wider socio-technical transitions, such as shifts in energy systems, industrial structures, or decarbonisation pathways (Li et al., 2024). Their purpose extends beyond technical performance or economic return. They are expected to generate value across multiple dimensions, including environmental outcomes and societal acceptance. As a result, their objectives are inherently plural and evolve over time in response to changing policy priorities, public expectations, and market dynamics.

Li et al. (2024) describe this shift through three conceptual stages: sustainability of megaprojects, sustainability for megaprojects, and sustainability by megaprojects. The first concerns minimizing environmental harm during delivery; the second integrates sustainability principles into governance and decision-making; and the third positions megaprojects as enablers of sustainability, creating the physical and institutional conditions for broader system transformation. The latter category – sustainability by megaprojects – is relevant for this research. Examples include carbon capture and storage networks, hydrogen corridors, and large-scale renewable energy hubs. Such projects do not merely aim to operate sustainably; they seek to create the physical and institutional conditions required for sustainability transitions within the systems they serve.

Sustainability-oriented megaprojects function as open socio-technical systems whose feasibility depends on alignment across a wide network of actors, infrastructures, and institutional arrangements that extend beyond the project organisation itself. Public authorities, private investors, operators, and regulators must coordinate across organisational and sectoral boundaries, often in the absence of a single actor with full decision authority (Miller and Lessard, 2000; Van Marrewijk, 2008). In such settings, project objectives, responsibilities, and even definitions of success are not fixed in advance but are continuously negotiated and reinterpreted over time.

This structural openness makes these projects highly sensitive to shifts in their broader policy and market environments. Their business cases are closely intertwined with external frameworks, including carbon pricing systems, subsidy regimes, permitting structures, and long-term industrial strategies. These frameworks are inherently political and therefore subject to revision as government priorities and societal debates evolve (Miller and Lessard, 2000; Samset and Volden, 2016). At the same time, many of the enabling technologies on which such projects rely, for example, hydrogen and carbon capture and storage, remain commercially immature, with uncertain cost trajectories and demand development.

Because policy, market, and technological systems co-evolve, sustainability-oriented megaprojects are exposed to forms of uncertainty that cannot be fully specified or stabilised at the time of decision-making. Assumptions about future demand, regulatory continuity, or infrastructure availability may appear reasonable when a project is initiated, but their validity can change as external conditions evolve (Samset and Volden, 2016; Benjaminsen and Sørnes, 2025). These dynamics unfold over long temporal horizons and at different speeds than project planning and delivery processes, creating the risk that projects remain operationally “on track” while their strategic rationale gradually erodes. Ashkanani and Franzoi (2023) observed the same in the context of the LNG (Liquefied Natural Gas) industry, large-scale industrial megaprojects frequently fail not due to technical errors but because the external context shifts faster than project governance can adapt. Their success thus hinges less on internal optimisation and more on the alignment of interdependent systems that operate under different institutional logics and time horizons.

As a result, sustainability-oriented megaprojects must be understood not as closed delivery systems, but as open systems whose viability is shaped by ongoing interactions with their enabling environment. Their success depends less on internal optimisation alone and more on the continued alignment between project objectives and the external systems they seek to transform. This open-system character fundamentally shapes how uncertainty emerges and why traditional approaches to project control and evaluation struggle to maintain visibility into long-term project viability.

## 1.3. Exogenous shocks as stressors revealing project vulnerability

Following Esposito and Terlizzi (2023) and Geraldi et al. (2011), megaprojects are frequently exposed to disruptive external events that can place sudden pressure on project delivery and decision-making. In project narratives, such developments are often described as external shocks that “hit” the project and push it off course. This interpretation places the focus on the shock itself, while overlooking the underlying conditions that made the project vulnerable to disruption in the first place (Samset and Volden, 2016).

From a governance perspective, exogenous shocks are better understood as stressors acting on an already complex and uncertain system. They do not introduce uncertainty into an otherwise stable project environment but interact with uncertainties that were already present - though often implicit - at the time key commitments were made. As Samset and Volden (2016) argue, many megaprojects continue to progress even as the strategic assumptions that originally justified them begin to erode, a phenomenon they describe as drifting logic. External shocks accelerate the visibility of this drift, but they do not initiate it.

This view aligns with broader work on uncertainty in large-scale projects. Benjaminsen and Sørnes (2025) show that in sustainability-oriented projects, actors often lack sufficient knowledge about how external systems - policy regimes, markets, or technologies - will evolve or interact over time. Under such conditions, uncertainty cannot be reduced to identifiable risks with assignable probabilities but reflects fundamental limits to what can be known in advance. Lempert et al. (2003)describe this as deep uncertainty, where decision-makers cannot agree on models, probabilities, or even the variables shaping future outcomes.

Within such settings, exogenous shocks do not function as abnormal events outside the project’s scope of concern. Instead, they act as moments of confrontation between project assumptions and external reality. When a shock occurs, it tests whether the conditions on which project viability depends were sufficiently understood, aligned, and stabilised at the time of commitment. While projects that appear robust may absorb such stress without major disruption, others experience cascading effects that expose weaknesses in their strategic foundations.

Importantly, this does not imply that shocks cause project failure. Rather, they expose whether project readiness was sufficient at the moments of commitment – stage-gates. When governance frameworks focus primarily on internal performance metrics and event-based risk registers, they may fail to detect growing misalignment between the project and its enabling environment. As a result, vulnerabilities embedded in external dependencies can remain latent until external pressure makes them visible, often at a point when corrective action becomes costly or politically difficult.

Understanding exogenous shocks in this way shifts analytical attention away from the unpredictability of the external environment itself and toward the project's capacity to carry uncertainty across stage-gates. It raises a critical question for project governance: not whether external shocks can be prevented, but whether the project entered its next stage with a sufficiently robust foundation to withstand them. This question, however, is rarely addressed explicitly within existing evaluation and decision-making frameworks, creating a persistent visibility gap in how project readiness is assessed.

## 1.4. The visibility gap in current project governance

Although dependencies play a foundational role in shaping project viability, they tend to fade from the view once a project moves beyond the front-end phase. Figure 1 illustrates a typical sequence of stage-gates through which the project progresses, including the terms as Opportunity Framing Workshop (OFW), Basis of Design (BoD), Project Technical Review (PTR), Front-End Engineering Design (FEED), Project Execution Review (PER), and the Final Investment Decision (FID).

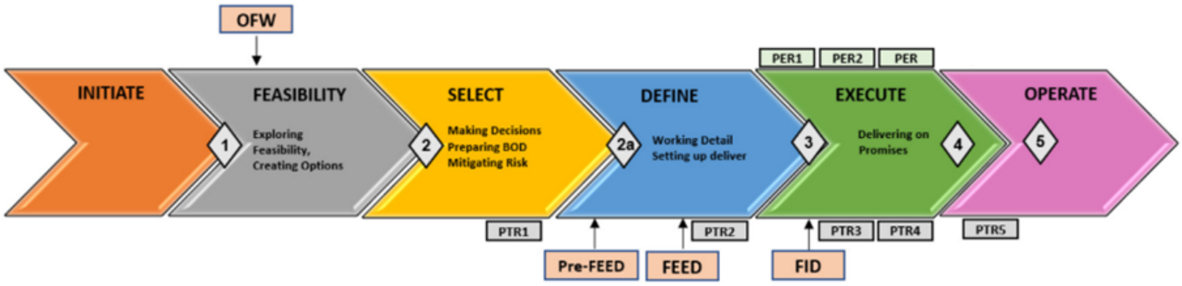


**Figure 1:** Stage Gates (Ashkanani and Kerbache, 2023)

During early decision-making, dependencies are often acknowledged as part of feasibility studies or strategic appraisals. However, once key gates have been passed and the project goes to the next stage, these dependencies are usually treated as fixed assumptions rather than conditions that may still evolve. (Samset and Volden, 2016). As a result, projects are then governed on the basis of assumptions that no longer reflect the external context in which they operate.

This creates a critical blind spot in project governance. In this research, project governance is understood as the set of decision-making structures and evaluation practices through which project objectives, assumptions, and commitments are defined, legitimised, and maintained across project gates (Samset and Volden, 2016). Unlike quantifiable risks, which can be recorded in risk registers or addressed through scenario analysis, dependencies do not lend themselves easily to probabilistic assessment or predefined mitigation strategies. Their development is shaped by factors largely outside direct project control, introducing uncertainty. Because uncertainty cannot be adequately represented within conventional control instruments, dependencies tend to remain outside formal governance routines and are rarely revisited at later decision points (Samset and Volden, 2016; Perminova et al., 2008).

In sustainability-oriented megaprojects, this governance blind spot is particularly consequential. These projects operate in environments characterised by complex interdependencies. Within such environments, projects are susceptible for drifting logic (Samset and Volden, 2016). Because most governance frameworks emphasise performance compliance rather than contextual validity, decision-makers may remain unaware that the project is losing alignment with its enabling environment

(Benjaminsen & Sørnes, 2025). This tendency is reinforced by institutional pressures that favour consistency, accountability, and procedural correctness. Once stage gates are passed and commitments formalised, revisiting earlier assumptions may be perceived as disruptive or illegitimate, even when external conditions have clearly shifted (DiMaggio and Powell, 1983; Ross and Staw, 1986). Therefore, governance routines prioritise control and reporting over reflection. This preserves procedural success while discouraging critical reassessment of the project's foundational logic.

As a result, uncertainty in sustainability-oriented megaprojects is often addressed indirectly, through adjustments to scope, schedule, or cost, rather than by reconsidering whether the underlying conditions for project viability still hold. External dependencies - despite being central to feasibility - remain largely invisible until misalignment manifests as performance problems or strategic disputes. At that point, corrective action tends to be reactive and constrained by sunk commitments (Samset and Volden, 2016; Too and Weaver, 2014).

The fact that megaprojects failures keeps on happening shows that there is a fundamental problem with the way projects are currently being managed. Governance systems are well equipped to monitor delivery performance, but poorly suited to observe whether a project continues to address the right problem under the right conditions. For projects that depend on evolving external systems, this imbalance between control and reflection represents a fundamental weakness. Understanding and addressing this gap is therefore essential for assessing project readiness at stage-gates and sustaining alignment between project rationale and external reality as projects progress.

## 1.5. Research objective and research questions

This research starts from the observation that in sustainability-oriented megaprojects, uncertainty does not primarily arise from isolated risk events, but from evolving dependencies on external systems. These dependencies condition project viability, yet they develop under limited knowledge and fragmented responsibility. As a result, projects may enter subsequent phases based on assumptions whose validity can no longer be taken for granted.

The central objective of this research is to develop a conceptual framework that supports the assessment of project readiness at moments of important stage-gates, by making visible how fundamental dependencies condition project viability. Rather than attempting to predict or control uncertainty, the framework distinguishes between structural vulnerability, which is inherent to critical dependencies, and robustness, which depends on whether the conditions required to stabilise these dependencies have been sufficiently developed by the time a stage-gate is passed.

Although the framework does not prescribe actions or predict outcomes, it shifts managerial focus by highlighting where uncertainty accumulates and where governance interventions are likely to be effective. It provides a structured way to reflect on whether a project is sufficiently prepared to carry uncertainty into the next stage-gate, given the state of its dependencies. By doing so, the research helps to develop a better understanding of how governance can support the making of informed commitment decisions in the face of uncertainty in sustainability-oriented megaprojects. While the framework is anchored in governance - supporting readiness assessment at stage-gates - it has second-order implications for project management by indicating which dependencies require stabilisation to sustain a robust project environment. The framework does not tell project teams what to do; it clarifies what they are already exposed to.

Based on this objective, the main research question is formulated as:

> *How can project readiness at stage-gates be assessed by analysing vulnerability and robustness in fundamental dependencies of sustainability-oriented megaprojects?*

This question is addressed through the following sub-questions:

1. *What qualifies as a fundamental dependency in sustainability-oriented megaprojects, and how do such dependencies differ from risks, tasks, or internal assumptions?*

2. *How does vulnerability arise as an inherent property of dependencies that are structurally critical to project viability?*

3. *How can robustness be understood as a conditionally developed property that shapes how dependencies function across project stage-gates?*

4. *How do stage-gates and commitment moments expose projects to the operationalisation of unresolved vulnerability?*

5. *How can recurring patterns of risk emergence and materialisation be explained as manifestations of unresolved vulnerability rather than as isolated events?*

## 1.6. Research approach and case selection

This research adopts a design-oriented case study approach aimed at developing and illustrating a conceptual framework for assessing project readiness under uncertainty. The approach recognises that sustainability-oriented megaprojects operate under conditions of uncertainty, where system behaviour cannot be reliably predicted and where strategic assumptions evolve over time. The study develops and refines a conceptual lens that helps explain how vulnerability and robustness shape project outcomes once commitment decisions are made.

The research follows a proof-of-concept logic. Existing theory is synthesised and reorganised to construct a dependency-based framework, which is then applied to an empirical case to explore its explanatory value. The case is not used to test hypotheses in a statistical sense, but to examine whether the framework provides meaningful insight into how uncertainty propagates and why certain risks materialise after passing stage-gates.

The primary case study is the Aramis CCS (Carbon Capture Storage) project, selected because of its scale, capital intensity, sustainability-oriented objectives, and reliance on coordinated performance across a complex value chain. Aramis provides a rich context in which multiple external dependencies intersect and evolve over time. The case allows for detailed identification and structuring of fundamental dependencies and for analysing how vulnerability and robustness develop across project phases.

To support validation and comparative reflection, the framework is additionally applied to the Porthos CCS project. Porthos serves as a reference case with a comparable technological and institutional context but has had different project trajectory and system configuration. Unlike Aramis, Porthos has already made its final investment decision and is currently in the execution phase. This enables us to retrospectively examine how dependency-related vulnerabilities manifested themselves after this financial decision was made. In terms of system openness, Porthos is largely configured as a closed system, with limited access through compressors, whereas Aramis is explicitly designed as an open-access $CO_2$ transport and storage system. This structural difference implies distinct dependency profiles, particularly with respect to market commitment, interface synchronisation, and scope stability. The comparison therefore allows assessment of whether vulnerability patterns observed in Aramis are specific to an open-access configuration or reflect broader dynamics in sustainability-oriented megaprojects.

Together, the two cases allow the framework to be examined across different project configurations and commitment gates, highlighting how vulnerability and robustness manifest under varying dependency structures.

## 1.7. Thesis outline

The remainder of this thesis is structured as follows.

**Chapter 2.** Reviews how risk and uncertainty are conceptualised in megaproject decision-making. It distinguishes between probabilistic risk and epistemic and ontological uncertainty, and shows how their interaction creates conditions of deep uncertainty that cannot be fully resolved in advance, particularly at moments of long-term commitment.

**Chapter 3.** Examines governance and evaluation practices in megaprojects, focusing on stage-gate processes, value assurance reviews, and final investment decisions. It shows how prevailing governance logics prioritise control, compliance, and the procedural closure of uncertainty, while offering limited visibility into how commitment remains exposed to evolving external conditions that are fundamental to project viability.

**Chapter 4.** Introduces a dependency-based perspective on uncertainty. It conceptualises fundamental dependencies as the means through which external uncertainty becomes connected to project viability, and develops the distinction between structural vulnerability and developmental robustness.

**Chapter 5.** Operationalises this perspective in the Megaproject Dependency Framework (MDF). It explains how dependencies, structural vulnerability, stabilising development conditions, and uncertainty propagation can be analysed together in order to assess structural exposure at stage-gates.

**Chapter 6.** Presents the analytical logic of the MDF in an integrated form. It explains how governance contexts, propagation, vulnerability, and development indicators combine into a dependency-based approach for interpreting uncertainty in practice.

**Chapter 7.** Applies the MDF to the Aramis CCS project. It analyses how dependencies are structured across governance contexts, how uncertainty propagates through the project, and how dependency instability becomes visible as a precursor to risk articulation.

**Chapter 8.** Provides comparative validation through the Porthos CCS project. It reconstructs how dependency instability was activated under external disturbances and how this later manifested as observed risks, thereby illustrating the framework’s explanatory value and its implications for dependency-aware project design.

**Chapter 9.** Reinterprets the role of risk in project governance through dependency logic. It argues that some risks should be understood as late manifestations of dependency instability, develops the governance shifts implied by this perspective, rethinks stage-gates and FID readiness, and outlines how the MDF can be applied in practice.

**Chapter 10.** Concludes the research by answering the research questions, synthesising the key findings, positioning the theoretical contribution, reflecting on implications for project governance practice, and discussing limitations and directions for future research.

# 2

# Uncertainty Beyond Risk in Megaprojects

## 2.1. Risk-based reasoning and its scope

Risk management has long been a central part of project governance, offering structured methods to identify, assess, and mitigate potential adverse events. When the boundaries of a problem are clear, causes are reasonably understood, and mitigation measures fall within the control of the project organisation, this approach allows for focused action and clear accountability.

Megaprojects, however, increasingly operate beyond these boundaries. Their scale, interdependence, and exposure to evolving external systems mean that many critical assumptions cannot be translated into discrete events with assignable probabilities (Denicol et al., 2020; Bates, 2018; Damayanti et al., 2021; Esposito and Terlizzi, 2023). Rather than managing bounded variability, project actors are often confronted with uncertainty about the validity of the assumptions on which plans, contracts, and business cases are built.

As noted in The Future of Megaproject Management (McDermott et al., 2024), megaprojects frequently rely on assumptions whose future stability cannot be guaranteed. Probability distributions are unknown, and forecasts rest on premises that may shift over time. In such situations, risk management can become an exercise in misplaced precision: numerical assessments give the impression of control but hide the fragility of their underlying assumptions. (Flyvbjerg and Gardner, 2023). This creates an illusion of predictability, which is reinforced by governance systems that encourage treating assumptions as fixed in order to facilitate commitment (Lenfle and Loch, 2010; McDermott et al., 2024).

The limitations of risk-based reasoning become particularly evident when identified risks cannot be meaningfully mitigated. When exposure originates in external systems, projects may document risks without being able to influence the conditions that cause them (Perminova et al., 2008; Samset and Volden, 2016). Risk registers may expand while underlying exposure remains unchanged. According to Nachbagauer and Schirl-Boeck (2019), control-oriented approaches lose effectiveness when uncertainty exceeds the domain of prediction.

This represents a key aspect of risk-based reasoning. It performs well where uncertainty can be reduced to events within stable boundaries but offers limited guidance when those boundaries themselves are unstable. In such cases, what matters is not the probability of specific events, but whether the assumptions on which commitment rests remain valid as the project progresses.

## 2.2. Uncertainty as a pre-event condition

Uncertainty in megaprojects is best understood not as a residual category beyond risk, but as a pre-event condition under which decisions are made. Rather than emerging at the moment an adverse event occurs, uncertainty exists beforehand and shapes choices regarding the scope, design, governance and commitment (Samset and Volden, 2016).

In complex governance settings, decision-making necessarily takes place under conditions of limited and contested knowledge (Dewulf and Biesbroek, 2018). This is not exceptional but structural: uncertainty is an inherent feature of decision-making contexts involving interacting technical systems, multiple actors and evolving institutional environments.

This pre-event uncertainty is particularly pronounced in megaprojects because key decisions involve long-term commitments made in conditions where external systems develop independently of project control. At the moment of commitment, decision-makers cannot know whether the assumptions embedded in business cases, contracts, or system designs will remain valid over time. Nevertheless, these assumptions must be considered stable enough to justify moving forward. Therefore, uncertainty does not delay decision-making; rather, it defines the conditions under which decisions are inevitably taken.

This perspective challenges the logic underlying many project evaluation practices. Stage-gate models and approval processes typically assume that uncertainty can be adequately resolved before commitment, and that any remaining exposure is manageable. However, from a pre-event perspective, the central issue is whether the assumptions on which commitment rests will remain viable as the project progresses. As Samset and Volden (2016) demonstrate, projects can continue to advance even when the strategic justifications for them are gradually undermined.

Understanding uncertainty in this way shifts the evaluative focus of project governance. Rather than asking whether all major risks have been identified and mitigated, the more fundamental question becomes whether the project is prepared to carry unresolved uncertainty into its next stage-gate. This reframing is essential for analysing long-term commitment decisions, where the consequences of unresolved uncertainty often become visible only after points of no return have been passed.

## 2.3. Epistemic and ontological uncertainty in projects

Building on established work in project governance and uncertainty studies, this research distinguishes between epistemic and ontological uncertainty as two analytically distinct, yet interacting, sources of exposure (Perminova et al., 2008; Samset and Volden, 2016; Dewulf and Biesbroek, 2018). This distinction provides conceptual clarity without reducing uncertainty to probabilistic risk. Epistemic uncertainty arises from limited, incomplete, or fragmented knowledge about system behaviour, external conditions, or causal relationships. In megaprojects, it commonly concerns how regulatory frameworks will be interpreted in practice, how external actors will respond to new system arrangements, or how technologies will perform at scale (Benjaminsen and Sørnes, 2025). While epistemic uncertainty may be reduced over time through learning or experience, it cannot be fully resolved at stage-gates.

Ontological uncertainty, by contrast, is inherent to the open-ended and emergent nature of complex socio-technical systems. It reflects the fact that future system states are not yet determined and cannot be fully anticipated, as they depend on the evolving interaction of multiple actors, institutions, and processes (Dewulf and Biesbroek, 2018). Megaprojects with a focus on sustainability are particularly vulnerable to ontological uncertainty because they are embedded in, and often seek to transform, the very systems on which their feasibility depends.

Both forms of uncertainty are critical for decision-making, but they challenge governance in different ways. Epistemic uncertainty concerns the reliability of available knowledge at the time decisions are taken, while ontological uncertainty challenges the assumption that future conditions can be stabilised at all. Neither can be eliminated through conventional risk management, which presupposes stable boundaries and identifiable events. Instead, these forms of uncertainty condition the validity of the assumptions on which commitments are based.

This study therefore focuses on epistemic and ontological uncertainty because they directly affect the stability of the external dependencies on which project viability rests. Other forms of uncertainty, such as ambiguity arising from divergent interpretations among actors, are acknowledged but lie outside the analytical scope of this research (Dewulf and Biesbroek, 2018).

## 2.4. Why uncertainty cannot be fully addressed through risk mitigation

Risk management reaches its limits when decision-making takes place under conditions of uncertainty rather than calculable risk (Klinke and Renn, 2002). When causal relationships are unclear, system responses are non-linear, or future developments cannot be meaningfully anticipated, mitigation measures no longer reduce exposure in a reliable way. Under such conditions, risk management tends to shift from stabilising the system to documenting uncertainty.

As a result, risks may be formally identified and assessed without altering the conditions that generate vulnerability. Klinke and Renn (2002) therefore argue that under uncertainty the core governance challenge is not optimising outcomes but avoiding irreversible and highly vulnerable states. This limitation is particularly of importance in sustainability-oriented megaprojects, where uncertainty concerns the stability of policy frameworks, institutional arrangements, market formation, or system-wide coordination. In such cases, mitigation can create an appearance of control while leaving structural vulnerability intact.

What ultimately matters is not whether individual risks have been mitigated, but whether commitments preserve the project's capacity to absorb change without cascading consequences. When uncertainty affects the validity of system conditions rather than the likelihood of discrete events, risk mitigation alone cannot secure long-term project viability.

## 2.5. Implications for commitment decisions at stage-Gates

The limitations of risk mitigation have direct implications for commitment decisions at project stage-gates. At these moments, uncertainty is not eliminated but translated into formal commitments that shape the project's future room for manoeuvre. Once decisions are approved, projects enter trajectories that become increasingly difficult to reverse as technical designs, contractual structures, and institutional expectations are locked in.

Under conditions of uncertainty, governance should therefore prioritise avoiding irreversible exposure rather than assuming that future adjustments will remain possible (Klinke and Renn, 2002). However, stage-gate decisions are often treated as confirmations of readiness, even when uncertainty about external conditions persists or increases. As Samset and Volden (2016) show, projects may continue to advance while the strategic premises that justified them gradually erode.

From this perspective, the core governance problem is not uncertainty itself, but the lack of visibility into how uncertainty affects the robustness of commitments. Decisions are taken as if unresolved uncertainty can be managed later, while each stage-gate reduces the project's capacity to adapt without significant cost or loss of legitimacy. When misalignment eventually becomes visible, the project may already be structurally exposed.

These observations point to a fundamental gap in current project governance: the absence of mechanisms to assess whether commitments made at stage-gates preserve robustness under uncertainty. Chapter 3 therefore turns to the governance and evaluation practices through which uncertainty is translated into commitment and exposure.

# 3

# Governance and Evaluation Under Uncertainty

## 3.1. Stage-gates, value assurances, and final investment decisions in megaproject governance

In many megaprojects, formal decisions regarding continuation and commitment are organised through stage gate models, Value Assurance Reviews, and ultimately a Final Investment Decision. These instruments structure when information must be produced, how alternatives are assessed, and under which conditions financial and political commitments become irreversible (Flyvbjerg, 2014; Esposito and Terlizzi, 2023).

Within this structure, stage gates operate as decision thresholds at which uncertainty is assumed to have been reduced to a level that justifies increased commitment. The underlying logic presumes that relevant uncertainty can either be addressed prior to passing the gate through further analysis and design refinement, or managed after the gate through established risk management practices (Lenfle and Loch, 2010; Nachbagauer and Schirl-Boeck, 2019). Value Assurance Reviews function within the same logic, assessing whether assumptions, analyses, and documented risks are sufficiently robust to warrant continuation.

Implicit in this arrangement is the expectation that uncertainty can be translated into manageable risk by the time of Final Investment Decision, allowing commitment to appear rational and defensible. Stage gate systems therefore primarily serve to stabilise scope, design, and financing by linking commitment to the formal completion of prescribed evaluation and assurance steps (Flyvbjerg, 2014; Damayanti et al., 2021).

This approach to orientation reflects a broader planning and control paradigm that prioritises predictability, compliance and accountability. While effective in structuring formal decision processes, such governance arrangements provide limited support for reflecting on evolving assumptions and dependencies when uncertainty cannot be fully resolved prior to commitment (Lenfle and Loch, 2010).

## 3.2. Event-oriented evaluation and the illusion of control

Within stage-gate governance architectures, progress and viability are assessed through the identification and management of discrete risks, issues, and deviations from predefined plans. Uncertainty can be made more manageable by translating it into identifiable events with probabilities, impacts and mitigation measures. This enables documentation, review and formal approval within decision-making processes (Perminova et al., 2008; Klinke and Renn, 2002).

This evaluative logic performs important governance functions. By structuring uncertainty as a set of manageable risk events, it supports accountability, auditability, and decision justification. Risks can be assigned, reviewed at predefined moments, and formally closed, signalling procedural robustness (Power, 2007; Flyvbjerg, 2014). On its own terms, event-oriented evaluation does not fail; it aligns with institutional demands for transparency and accountability.

However, this approach only works if uncertainty can be expressed as identifiable events before commitment is taken. It assumes that uncertainty can be described in terms of specific risks with defined causes and consequences.

This assumption becomes problematic when uncertainty concerns slowly evolving conditions such as changing policy frameworks, emerging markets, institutional alignment, or system integration. These are not discrete events. They are structural conditions that shape whether events may occur in the first place (Dewulf and Biesbroek, 2018; Benjaminsen and Sørnes, 2025). Because such conditions develop gradually and do not present themselves as single incidents, they often remain outside formal evaluation. They only become visible once their effects begin to materialise within the project.

As a result, stage-gate governance can produce an illusion of control. Green dashboards completed assurance reviews, and formally mitigated risks indicate procedural soundness. Yet they provide limited insight into whether the assumptions underpinning earlier decisions remain valid (Nachbagauer and Schirl-Boeck, 2019; McDermott et al., 2024). Evaluation becomes focused on what can be measured and reported, while shifts in foundational conditions remain largely invisible.

This limitation is not due to insufficient rigour or managerial failure, but to the structural orientation of event-based evaluation itself. By privileging identifiable risks over evolving conditions, stage-gate governance is well equipped to manage deviations from plan, but poorly equipped to detect erosion in the assumptions that justified commitment in the first place. It is within this gap - between procedural control and substantive exposure - that projects may progress through successive gates while becoming increasingly misaligned with their evolving environment.

The persistence of this evaluative framing cannot be explained by governance structures alone, but is reinforced by organisational, behavioural, and institutional dynamics.

## 3.3. Internal project perspectives and decision lock-in

This following section examines how organisational, behavioural, and institutional dynamics further reinforce this pattern of procedural control, helping explain why such misalignment often remains unchallenged at the moment of commitment. It examines why this framing persists in practice, even when uncertainty increases and underlying assumptions remain fragile. It analyses how internal project perspectives, behavioural biases, and institutional pressures interact to stabilise commitment and limit critical reconsideration at stage-gates. Together, it explains why projects can continue through successive decision points while confidence grows and exposure remains largely unexamined.

### 3.3.1. Inside-view dominance in stage-gate evaluation

Evaluation at stage-gates is predominantly shaped by an inside view. Information about project status, risks, and readiness is generated and interpreted within the project organisation and its immediate governance environment. Progress assessments rely mainly on internally produced analyses and documentation, anchoring the framing of uncertainty, risk identification, and interpretation of deviations in the project's own assumptions and problem definitions. External or system-level perspectives tend to receive limited attention as long as the project appears "on track" in terms of schedule, budget, and formally reported risks.

This pattern reflects Flyvbjerg's distinction between the inside view and the outside view in project decision-making. The inside view prioritises project-specific characteristics and internal forecasts, while systematically underweighting external reference classes and broader contextual developments (Flyvbjerg, 2006, 2014). Although outside-view approaches are widely recommended to improve decision quality, empirical research shows that stage-gate evaluations remain dominated by inside-view reasoning once projects move beyond early conceptual phases.

Studies of front-end decision-making reinforce this observation. Samset and Volden (2016) show that early concept selection often relies on internal problem framings rather than systematic exploration of the opportunity space. Once a preferred concept is established, later evaluations focus on justifying and refining that choice, rather than reopening more fundamental questions about its continued relevance or viability. Governance arrangements even further reinforce this tendency: front-end evaluation typically builds on iterative refinement of internally generated analyses, with limited scope for sustained external challenge once a project trajectory has been set (Williams, 2019).

Taken together, these dynamics indicate that stage-gate evaluation is not merely influenced by internal perspectives but structurally predisposed toward them. This does not imply negligence or bad faith on the part of those involved in the project, but rather reflects how evaluation practices are organised to maintain coherence, accountability and momentum within the project system. The consequence is that uncertainties originating outside the immediate project boundary are less likely to be surfaced or questioned at stage-gates.

### 3.3.2. Behavioural biases and the reinforcement of commitment

Within a control oriented governance logic, behavioural and institutional biases tend to reinforce existing evaluation frames rather than fundamentally reshape them. Optimism bias and strategic misrepresentation encourage favourable interpretations of cost, schedule, and benefit assumptions, while downplaying complexity and residual uncertainty (Flyvbjerg, 2014). Early commitment to a preferred concept and gradual lock in on initial assumptions further limit the exploration of alternatives, stabilising the prevailing frame of evaluation and reducing the visibility of uncertainty, particularly under political pressure and high strategic ambition (Samset and Volden, 2016).

As commitment deepens and resources are invested, confirmation bias and escalation dynamics become more pronounced. Organisations increasingly interpret unfavourable information in ways that sustain the existing project trajectory, while institutional arrangements and reputational considerations make deviation more costly (Ross and Staw, 1986; Flyvbjerg, 2014). Under such conditions, uncertainty encountered during project development is more likely to be reframed as manageable within the existing framework than treated as a signal that underlying structural assumptions require reconsideration.

Importantly, these behavioural and institutional mechanisms do not explain why uncertainty is framed as controllable in megaproject governance. Rather, they help explain why such framing persists over time. Formal evaluation frameworks define uncertainty as manageable by design; biases increase the likelihood that uncertainty will be interpreted accordingly at the moment of commitment, and that information challenging this interpretation will be marginalised. Together, these dynamics reinforce confidence and continuity even as exposure to external uncertainty grows.

### 3.3.3. Institutional pressure and the rationalisation of continuation

Also, as projects advance through successive stage-gates, decision-making shifts from questioning whether the project remains justified to assessing whether stopping can still be defended. Lock-in dynamics, reputational exposure, and accumulated commitments push organisations structurally toward continuation, not because uncertainty has been resolved, but because reversal becomes increasingly costly and difficult to legitimise (Samset and Volden, 2016).

This dynamic is reflected in the persistent failure rates of megaprojects. If stage-gates, Value Assurance Reviews, and Final Investment Decisions functioned as a robust selection and termination mechanisms, a substantial share of underperforming projects would never have progressed beyond early commitment stages. The fact that many projects continue despite escalating costs, delays, and strategic misalignment suggests that these instruments stabilise commitment more effectively than they filter out vulnerable projects. Formal optionality may exist on paper, but in practice it erodes as financial, legal, and political commitments accumulate.

Political and organisational expectations further reinforce this pattern. In policy-driven infrastructure projects, continuation often carries symbolic value, signalling commitment to broader strategic objectives such as climate policy, industrial transition, or national competitiveness. Under such conditions, projects acquire meaning beyond their technical or economic rationale, making deviation from the agreed path politically costly (Flyvbjerg and Gardner, 2023). Political sponsors and public authorities may therefore exert pressure to maintain progress.

As a result, continuation is frequently framed as the most defensible option within existing governance arrangements, while stopping becomes exceptional and difficult to justify. Commitment thus reflects not the resolution of uncertainty, but the institutionalisation of exposure under conditions of political visibility and organisational pressure - setting the stage for the governance blind spot addressed in the following section.

## 3.4. Commitment as exposure: the governance blind spot

Building on work on lock-in and early commitment in megaprojects (Flyvbjerg, 2014; Samset and Volden, 2016), and on risk concepts such as irreversibility and vulnerability (Klinke and Renn, 2002), this section reframes commitment as a mechanism of exposure. Rather than viewing commitment as the end of the evaluation process, this approach conceptualises commitment as the point at which uncertainty becomes part of an evolving system of dependencies. Although previous studies have documented the consequences of early lock-in, path dependency and escalating vulnerability, they have not explicitly theorised commitment itself as the process by which such exposure is consolidated and carried forward into subsequent stages of the project.

When uncertainty is procedurally closed at the moment of commitment, the central governance problem shifts. The issue is no longer how to make the “right” decision ex ante, but what this decision exposes the project to over time. Commitment does not resolve uncertainty; it fixes the project into a set of technical, contractual, institutional, and organisational dependencies whose continued viability depends on external conditions beyond the project’s direct control. These dependencies may remain stable, but they may also erode, shift in meaning, or interact in unforeseen ways as the surrounding context evolves. From this perspective, commitment is not a neutral administrative step at the end of evaluation, but a multiplier of uncertainty. By stabilising assumptions, interfaces, and responsibilities, commitment makes the project increasingly dependent on conditions that cannot be fully specified or governed at the moment of decision. This argument brings together insights from research on early lock-in and institutional vulnerability (Flyvbjerg, 2014; Samset and Volden, 2016; Klinke and Renn, 2002), building on them by focusing directly on commitment as the mechanism through which exposure to evolving dependencies arises.

Existing governance arrangements, however, continue to treat commitment as the point at which uncertainty is brought under control. Evaluation criteria are applied, risks are documented, and formal accountability is discharged. What remains largely invisible is what the project becomes dependent on after commitment, and how the robustness of those dependencies may change over time. No core governance instrument systematically assesses the structure and sensitivity of these dependencies at the moment of commitment; they persist instead as background assumptions embedded in contracts, interface agreements, and risk registers. The result is a governance blind spot. Commitment consolidates vulnerability without making its underlying dependencies visible or traceable. Addressing this visibility gap requires shifting attention away from decisions as isolated events and toward the dependencies through which exposure unfolds over the project lifecycle. The following chapter develops this perspective by examining how such dependencies can be identified, characterised, and monitored as evolving conditions of project viability.

# 4

# From Uncertainty to Vulnerability: a Dependency Perspective

Building on the preceding analysis, Chapter 4 introduces a dependency-centred framework that conceptualises uncertainty not as a set of risks, but as a structural property of project viability.

## 4.1. Dependencies as structural carriers of uncertainty

Building on the understanding of sustainability-oriented megaprojects as open systems, uncertainty cannot be adequately understood as a collection of isolated risks or discrete events. Instead, it emerges from the web of relationships through which projects are connected to their enabling environment. Dependencies between systems, actors, technologies, institutions, and objectives shape where uncertainty enters the project, how it persists, and how its effects propagate over time.

From this perspective, uncertainty is inherently relational. It arises not only from incomplete knowledge about the future, but also from the fact that project outcomes depend on interactions among interdependent elements whose behaviour cannot be fully predicted in advance. Dependencies therefore carry epistemic uncertainty related to limits of knowledge and foresight, as well as ontological uncertainty related to the emergent and evolving behaviour of complex socio-technical systems (Floricel et al., 2016; Dewulf and Biesbroek, 2018). This distinction between epistemic and ontological uncertainty is analytically important, as it later provides the basis for categorising and interpreting different types of dependencies within the framework.

If uncertainty is embedded in interdependent structures, then its effects cannot be understood solely in terms of isolated events. Instead, uncertainty takes on a structural form that reflects how dependencies are configured and connected. In this sense, uncertainty leads to vulnerability when the structure of interdependencies allows disturbances to spread and strengthen each other.

Research on interdependent risks and uncertainties supports this interpretation by showing that vulnerability depends less on individual factors than on how they are connected through dependency structures. Network-based studies demonstrate that uncertainties become critical not because they are likely, but because their position within a system allows their effects to propagate, interact, and reinforce one another over time (Qazi et al., 2020; Guan et al., 2021). Vulnerability, from this perspective, is not associated with isolated risk events, but with dependency structures that enable cascading effects and feedback dynamics. Importantly, these dependencies are not limited to technical connections within the project system. Uncertainty also arises from relationships with external actors and institutions, such as regulators, strategic partners, financiers, and public authorities. Project outcomes depend on how these actors make decisions, coordinate actions, and interpret their responsibilities over time.

Governance research shows that strategic and institutional uncertainty emerges from such interdependencies between actors, as well as from changing rules, expectations, and alignments across policy arenas (Jensen et al., 2006; Dewulf and Biesbroek, 2018). In this sense, vulnerability is not only a function of technical system design, but also of how organisational and institutional relationships evolve. When dependencies remain implicit or taken for granted, uncertainty often appears as an abstract or residual problem. Making dependencies explicit does not necessarily introduce entirely new risks. Rather, it draws attention to where project viability depends on forms of coordination, timing, or commitment that are often taken for granted and rarely examined explicitly (Danilovic and Sandkull, 2005). In this way, dependency analysis does not replace risk identification but reframes how existing exposure is understood and where its structural foundations lie. Vulnerability arises not because a specific failure has occurred, but because shifts in these assumed conditions can disrupt multiple interdependent elements at once.

Taken together, this implies that uncertainty cannot be understood or governed by focusing on individual risks in isolation. Vulnerability is a structural condition shaped by how dependencies are configured and by how changes in one part of the system propagate through others over time (Qazi et al., 2016; Qazi et al., 2020). This chapter therefore adopts a governance-oriented interpretation in which dependencies are treated as structural carriers of uncertainty. This provides the conceptual basis for analysing vulnerability and robustness as properties of project viability, rather than as outcomes of individual risk events.

## 4.2. Defining fundamental dependencies through project scope and viability

What are then the dependencies in a project and what makes a dependency dominant, because not all dependencies that shape a project are equally important for its long-term viability. Now, while complex projects involve numerous technical, organisational, and contractual interdependencies, only a subset of these dependencies is fundamental in the sense that they determine whether the project continues to make sense under changing conditions. Identifying such fundamental dependencies therefore requires a shift in perspective: rather than starting from project activities or risk registers, it requires starting from what it means for the project to remain viable, given its scope and intended contribution of succes.

To clarify this logic, it is useful to begin with a simple analogy before returning to sustainability-oriented megaprojects.

### 4.2.1. Project scope defines viability

Whether a project is considered viable depends partly on how its purpose and scope are defined within governance and strategic frameworks.

Consider a simple analogy. A holiday, in its most basic sense, consists of travel, accommodation, and activities. Defined in this way, almost any arrangement could qualify. However, when the scope is specified more precisely - for example, as a *successful* holiday - additional criteria immediately come into play, such as a *safe* travel, a *comfortable* stay, and *enjoyable* activities. These criteria do not prescribe specific actions but define the conditions under which the holiday can be considered viable.

Importantly, such viability criteria are not fully controllable. *Safety*, *comfort*, and *enjoyment* depend on conditions that extend beyond individual choices, including the reliability of transport, the quality of infrastructure, and the behaviour of others – a world of ontological uncertainty. Even when all immediate preparations are in place, vulnerability remains, because the outcome depends on continued alignment between these conditions and the chosen course of action. The decision to go on holiday does not create risk in itself; it commits the traveller to a context in which existing vulnerabilities may become operational.

The same logic applies to sustainability-oriented megaprojects. Their scope is typically defined not only in terms of deliverables, but in terms of broader contributions such as system transformation, environmental performance, or public value. Viability is therefore determined by criteria that cannot be fully engineered or controlled. These criteria define what must hold for the project to remain meaningful over time, and they form the basis for identifying which dependencies are fundamental and subjected to ontological uncertainty.

### 4.2.2. Viability criteria precedes dependencies

Viability criteria define what must hold for a project to remain meaningful over time. They articulate the qualities that justify continued commitment rather than prescribing specific activities or solutions. As illustrated by the holiday analogy introduced earlier, such criteria define what it means for an endeavour to be viable (e.g. safe or enjoyable), before any decisions are made about routes, means, or actions. In this sense, viability criteria precede both strategy and control: they establish what counts before decisions are made about how to act.

Once viability criteria are articulated, dependencies emerge as the conditions on which these criteria rely. Dependencies are not selected arbitrarily, nor do they follow directly from technical design choices. Instead, they are derived from the question of what the project must continue to depend on in order for its viability criteria to remain satisfied. In this sense, dependencies are relational and conditional: they link the project's intended contribution to external systems, actors, and processes whose continued alignment cannot be assumed.

Identifying these viability dependencies is critical precisely because they are exposed to ontological uncertainty. Changes in external systems, institutional arrangements, or societal priorities can act as exogenous shocks that directly affect whether viability criteria continue to hold, even when the project itself remains internally coherent. Viability dependencies thus constitutes the primary interface through which external uncertainty enters the project system.

This derivation is particularly significant for sustainability-oriented megaprojects. Viability criteria often include alignment with policy objectives, sustained participation across a value chain, and long-term public legitimacy. Each of these criteria implies dependencies that extend beyond the project boundary. Policy alignment depends on regulatory stability and political support; value-chain participation depends on market uptake and counterpart commitment; legitimacy depends on societal acceptance and institutional endorsement. These dependencies are not outcomes to be delivered by the project, but conditions that must hold for the project to continue to make sense.

Crucially, dependencies derived from viability criteria differ from operational requirements. While operational elements can often be adjusted, substituted, or optimised, dependencies tied to viability criteria are harder to revise without reinterpreting the project's purpose. If such dependencies shift, the issue is not merely one of performance shortfall, but of whether the project's original rationale remains intact. Dependencies thus function as carriers of exposure: they connect viability criteria to evolving external conditions that the project cannot fully control.

By tracing dependencies back to viability criteria, it becomes possible to distinguish those dependencies that are merely instrumental from those that are fundamental. Fundamental dependencies are those whose continued alignment is necessary for the project's viability - viability dependencies - as defined by its scope and intended contribution. This derivation provides a principled basis for identifying where vulnerability is likely to concentrate, which is taken up in the next section.

## 4.3. Structural vulnerability as a property of project viability

If project viability constitutes the interface through which external uncertainty enters the project system, vulnerability must be understood as a structural property of that interface. Vulnerability does not arise from isolated failures or unexpected events, but from the degree to which a project's viability remains exposed to changes in the external conditions on which it depends. From this perspective, vulnerability precedes performance outcomes and exists even when a project is internally coherent and progressing according to plan.

In broader literature, vulnerability is commonly defined in terms of exposure, sensitivity, and the capacity to cope with external stressors (Turner, 2003; Adger, 2006; Urruty et al., 2016). These formulations emphasise that vulnerability is not synonymous with damage or failure but reflects the extent to which a system is positioned to be affected by external change. Risk scholarship similarly highlights that vulnerability is closely linked to irreversibility and the consequences of commitments that constrain future options (Klinke and Renn, 2002). Taken together, these perspectives detach vulnerability from discrete events and locate it in the structural relationship between a system and its environment.

Applied to sustainability-oriented megaprojects, this implies that vulnerability is an inherent feature of the project whose viability depends on continued alignment with evolving policy regimes, markets, technologies, and societal expectations. Even in the absence of identifiable risks or performance deviations, such projects remain vulnerable because shifts in these external systems can undermine the conditions under which the project continues to make sense.

Structural vulnerability is therefore unevenly distributed across the dependency structure. It is most consequential where external change affects dependencies that sustain the project's strategic rationale, rather than only those that influence its execution. When policy priorities shift, market participation weakens, or institutional support erodes, the issue is not merely that operational risks materialise. Instead, the alignment between the project's underlying assumptions and its external context begins to weaken. Vulnerability manifests as this growing misalignment between the conditions on which the project depends and the environment in which it operates.

Importantly, understanding vulnerability as a property of project viability reframes its governance implications. Vulnerability cannot be eliminated through improved forecasting, additional controls, or more detailed risk registers, because it does not originate in uncertainty about specific events. It originates in exposure created by dependence on external systems that evolve beyond the project's control. Vulnerability can increase without triggering formal governance responses.

This interpretation positions vulnerability as a diagnostic concept. It directs attention to where project viability is exposed to external uncertainty, how that exposure is structured through those dependencies, and why projects may remain fragile even when they appear internally coherent and well managed.

In line with broader systems literature, vulnerability can be distinguished from related properties such as robustness and resilience, which describe how systems respond to disturbance rather than how they are positioned in relation to it (Urruty et al., 2016). Building on this distinction, the next section examines robustness as a conditionally developed property of project viability, reflecting whether - and for how long - these dependencies remain aligned within tolerable bounds.

## 4.4. Robustness as a conditionally developed property

If vulnerability describes how project viability is exposed to external uncertainty through fundamental dependencies, robustness concerns whether - and for how long - these dependencies remain aligned within tolerable bounds. Robustness is therefore a developed property, reflecting the capacity of a project's dependency configuration to absorb variation without undermining its viability. In line with systems literature, vulnerability can be understood as exposure to external stress, while robustness refers to the range of conditions under which a system can maintain its core purpose without requiring recovery or adaptation (Urruty et al., 2016). This capacity depends on the project's context, on prior commitments that constrain flexibility, and on how dependencies are configured and interconnected.

Applied to sustainability-oriented megaprojects, robustness is deliberately shaped by how projects position themselves with respect to the dependencies that carry uncertainty. Decisions about scope, interfaces, contractual arrangements, reliance on external actors, and tolerance for variation determine how exposed the project will be once it enters its operating context.

This perspective distinguishes robustness from control. Robustness is not produced through tighter governance or post hoc risk mitigation, but through targeted ex ante and early-phase interventions that shape dependencies and their tolerances. While robustness can be strengthened before and around moments of commitment, after commitment it is primarily tested as external conditions evolve. Understanding robustness in this way shifts attention away from how projects respond once risks have materialised, toward how they position themselves under conditions of uncertainty at the moment of stage-gates. Robustness is built precisely because future risks cannot be fully anticipated: by shaping dependencies to tolerate variation. Projects seek to limit how uncertainty translates into destabilising risks. The next section therefore examines commitment and stage-gate moments at which the robustness of a project's dependencies is tested and, in some cases, irreversibly exposed.

## 4.5. Commitment as a test of dependency robustness

The total logic of this chapter can be illustrated by returning briefly to the holiday analogy introduced earlier. If a *successful holiday* defines the scope, then *safe* travel, a *comfortable* stay, and *enjoyable* activities constitute the viability criteria that must hold for that scope to remain meaningful. These criteria do not only define what counts as success but also shape the bounded space of dependencies on which that success relies, such as modes of transport, accommodation options, or available infrastructure. These criteria are inherently subject to uncertainty: some aspects can be clarified through information and checks, while others remain fundamentally unpredictable.

Preparations such as checking oil levels or inspecting the car reduce epistemic uncertainty about whether the vehicle is fit for the journey. Other choices - such as selecting a larger or more reliable car, avoiding particularly risky routes, or allowing additional travel time - do not reduce uncertainty itself, but address ontological uncertainty by shaping how exposed the journey will be once it begins. Together, these choices shape robustness in relation to existing vulnerability.

Importantly, robustness and scope are interrelated. If *safe* travel cannot be reasonably ensured given the condition of the car, the traveller may change transportation modes or adjust the scope of the holiday - for example by travelling shorter distances or lowering expectations about comfort or activities. Such adjustments do not respond to realised risks but reposition the journey in relation to uncertainty by redefining dependencies and viability criteria.

The decision to leave - analogous to a project stage gate - does not create new risks but places the traveller in a context where existing vulnerability becomes operational. Only once the journey starts does it become visible whether the prior configuration of dependencies was robust enough to accommodate unforeseen conditions without undermining the viability of a *"successful* holiday."

Importantly, this analogy should not be read as an example of risk management. The choices described do not aim to identify, quantify, or mitigate specific risk events. Instead, they address uncertainty at a more fundamental level by shaping how the journey is positioned relative to conditions that cannot be fully anticipated. Robustness is built not by reducing the likelihood of particular failures, but by configuring dependencies such that unforeseen developments are less likely to compromise the viability of the overall endeavour. In this sense, uncertainty precedes risk: risks materialise only once the journey has begun, whereas robustness is established in advance by how exposure is structured. The decision to depart does not follow from having "managed all risks", but mostly from accepting residual uncertainty under a chosen configuration of dependencies.

This distinction becomes important when a single dependency is critical for achieving a viability criterion. If a *safe* travel depends heavily on the car, then vulnerability at that point affects the viability of the entire holiday. In such a situation, checking the oil level becomes a priority - not because the oil level represents the highest quantified risk to the overall scope, but because uncertainty about the car's functioning directly conditions whether the viability criterion can be met at all. The intervention therefore addresses a point of concentrated uncertainty rather than a probabilistic risk, aiming to prevent uncertainty from becoming operational across the full scope of the journey.

Applied to sustainability-oriented megaprojects, this perspective clarifies why fundamental dependencies deserve explicit attention at moments of commitment such as stage-gates and Final Investment Decisions. Just as a holiday depends on a small number of conditions that determine whether *safe* travel is possible at all, the viability of sustainability-oriented megaprojects rests on a limited set of dependencies whose failure would affect the meaning of the project as a whole. These dependencies do not appear as discrete risks, but as conditions whose continued alignment cannot be guaranteed and whose uncertainty becomes operational only after commitment.

Chapter 4 has argued that uncertainty in such projects is not primarily located in individual events, but in the structure of dependencies through which project viability is exposed to external systems. Vulnerability describes where and how this exposure exists; robustness reflects how projects position themselves relative to that exposure; and commitment marks the moment at which these conditions are tested under heightened constraints. Together, these concepts explain why traditional evaluative practices struggle to maintain visibility into the foundations of project viability, because they focus on identifiable risk events and compliance criteria rather than on the evolving alignment of structural dependencies on which project viability depends.

This insight leads directly to the central implication of this research: if uncertainty cannot be eliminated before commitment, and if robustness is determined by how dependencies are configured rather than by risk mitigation alone, then governance requires a way to keep fundamental dependencies visible over time. Chapter 5 addresses this need by introducing an evaluative perspective designed to surface dependency vulnerability and robustness at stage-gates and commitment moments, enabling more informed decisions without assuming that uncertainty can be resolved in advance.

# 5

# Operationalising Dependency-Based Analysis

Figure 1 and 2 together present the analytical architecture of the Megaproject Dependency Framework (MDF), which is repeated in Appendix B.1. Figure 5.1 decomposes the framework into its core analytical components, while Figure 5.2 shows how these components are brought together into an integrated governance-oriented representation of project exposure.

Figure 1 hould be read from left to right. It starts with the project scope, which defines what constitutes strategic success. This scope is translated into a set of viability dependencies: the external conditions that must hold for this success to remain plausible. Each viability dependency is realised through a set of enabling dependencies, which specify the mechanisms through which these conditions are operationalised in practice. In turn, enabling dependencies are informed by sub-dependencies (development indicators), which capture how the conditions underlying these mechanisms are developing over time. Together, these nested layers form the structural interface through which external uncertainty connects to the project.

Two distinct forms of uncertainty are represented. Ontological uncertainty relates to the inherent variability of the external systems on which viability and enabling dependencies rely. This form of uncertainty cannot be resolved in advance and gives rise to structural vulnerability, which therefore applies to these dependencies. Epistemic uncertainty, by contrast, concerns the evolving state of knowledge about how enabling conditions are developing. This is captured through sub-dependencies, which function as development indicators and serve to progressively clarify how these conditions are unfolding, without reducing the underlying vulnerability.

The middle layer of the figure translates these two forms of uncertainty into analytical properties. Ontological uncertainty is interpreted through structural vulnerability, characterised by autonomy, bandwidth, and feedback strength. Epistemic uncertainty is interpreted through development status, expressed through the green-orange-red categorisation. In parallel, the MUF-derived governance contexts provide an interpretive structure to locate where uncertainty becomes visible and consequential within the project. Dependencies and their associated development indicators are positioned within these contexts based on where their effects are primarily experienced, while secondary placements capture how uncertainty may extend into other domains through propagation. This allows uncertainty to be understood not as a linear cause-effect chain, but as a condition that is interpreted and reinterpreted across governance domains.

The lower part of the figure shows how these elements are analytically reduced into three core dimensions: a structural vulnerability position, a development status, and a contextual placement. These dimensions do not yet form the framework itself, but represent the building blocks through which exposure can be interpreted.

Figure 1 therefore functions as an analytical decomposition of the MDF.

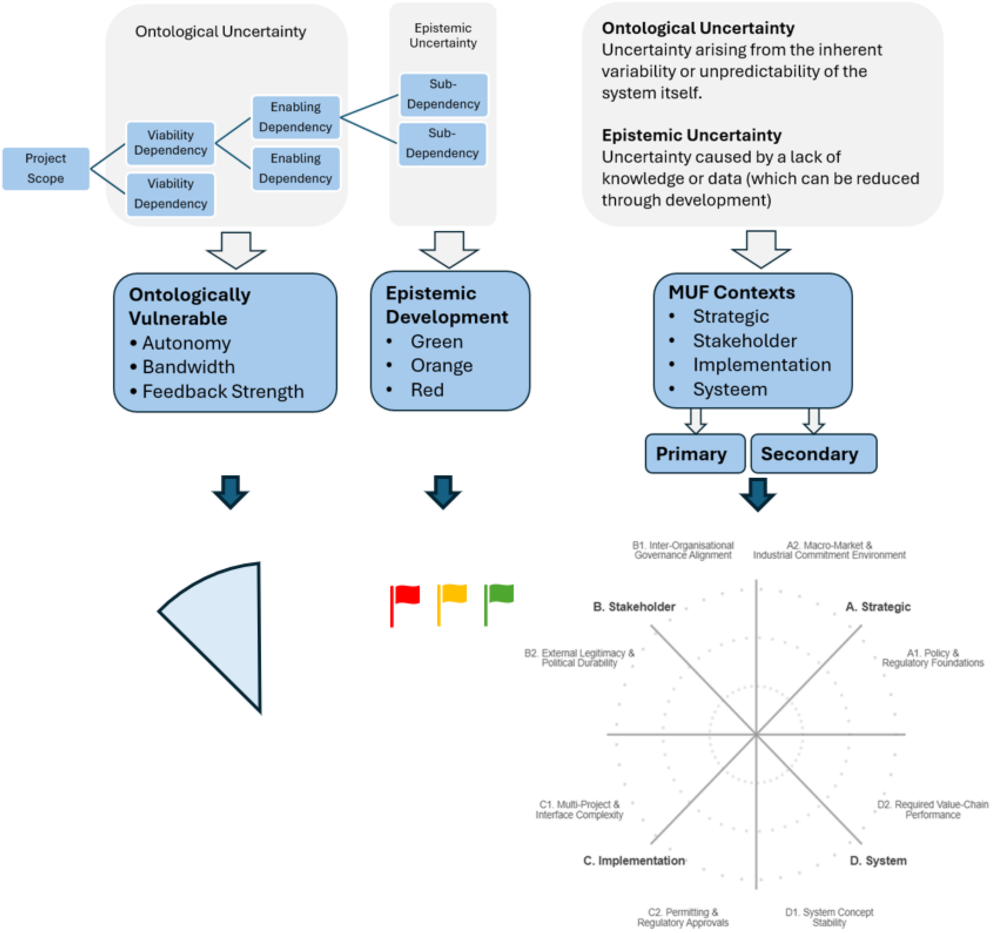


**Figure 1:** Graphical abstract of MDF model components

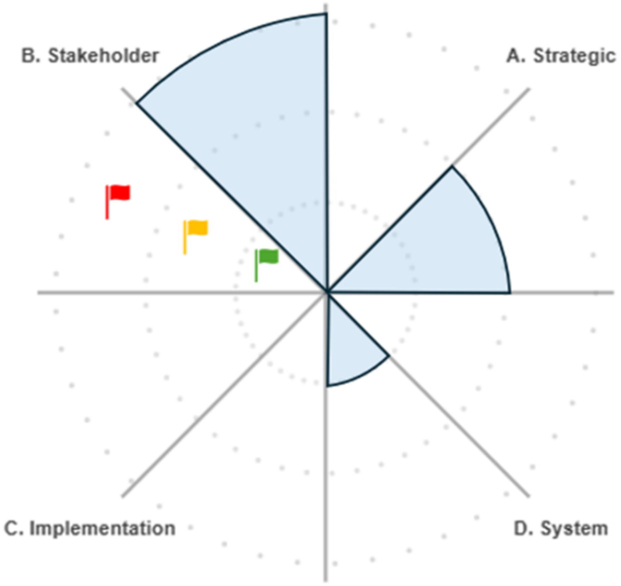


**Figure 2:** Megaproject Dependency Framework application

The following sections unpack each of these components in detail. Figure 2 subsequently shows how these components are recombined into a single integrated representation of dependency exposure.

## 5.1. Purpose and scope of the Megaproject Dependency Framework

Chapters 1 to 4 have shown that uncertainty in sustainability-oriented megaprojects does not primarily arise from isolated risk events. Instead, it is embedded in evolving dependencies on external systems that shape whether the project remains viable. These dependencies introduce both epistemic uncertainty and ontological uncertainty. Because such dependencies develop over time and extend beyond the project's direct control, they are difficult to fully address in advance and often remain only partially visible within prevailing governance and evaluation practices. The central implication is that project readiness at stage gates cannot be assessed solely by asking whether risks have been identified and mitigated. What also matters is where project viability is structurally exposed, how sensitive that exposure is, and how external developments begin to interact with the assumptions underlying commitment. This chapter introduces the Megaproject Dependency Framework (MDF) as a structured way to analyse these issues. The framework does not seek to eliminate uncertainty or predict specific outcomes. Instead, it makes fundamental dependencies explicit, characterises their structural vulnerability, and provides a way to interpret how external developments relate to project viability at stage gates, including the Final Investment Decision. To do so, this chapter defines a set of analytical properties through which fundamental dependencies can be described and examined in a consistent manner. The following sections introduce complementary characteristics that together clarify how project viability is structurally exposed to uncertainty:

1. Dependencies as viability, enabling and sub-conditions;
2. The structural vulnerability inherent to viability and enabling dependencies;
3. The contextual domains in which uncertainty originates and propagates;
4. Development indicators as observation points;
5. An interpretive status that reflects the direction and consolidation of those developments.

Together, these properties form the internal analytical logic of the MDF. They allow dependencies to be analysed as structured forms of exposure rather than as isolated risks to be mitigated. The MDF is not a case specific tool or a decision algorithm. It is a framework designed to make explicit how uncertainty is structurally embedded in dependency configurations and potentially carried through stage-gates

## 5.2. Dependencies as viability conditions

In order to operationalise a dependency-based analysis, it is first necessary to clarify what the project depends on in order to remain viable. Only then can the question of how exposed those dependencies are be asked.

Building on the distinction between ontological and epistemic uncertainty developed in Chapters 2 and 4, the MDF differentiates between three analytically distinct roles through which dependencies condition project viability under uncertainty. These roles reflect different functions in dependency exposure analysis, rather than representing a hierarchical decomposition of the project system.

### Viability Dependencies

At the highest level, viability conditions describe the external conditions that must hold for the project to remain strategically viable and ultimately succeed. These conditions are not fully under the control of the project organisation. Their future existence, stability, or direction cannot be predicted with certainty. For this reason, viability conditions are subject to ontological uncertainty. They represent forms of structural exposure that the project must accept at stage gates. They define the outer boundaries of what makes the project meaningful in its wider policy, market, and institutional environment.

*Viability conditions applied to the 'holiday' example:*

Whether safe travel is possible at all defines a viability condition for the scope: ‘successful holiday’. It cannot be guaranteed in advance and must be accepted when deciding to depart, regardless of how well the journey is prepared, the uncertainty is accepted.

#### Enabling Dependencies

Viability conditions are realised through enabling dependencies. These describe the mechanisms through which viability conditions can be materialised in practice. Enabling dependencies translate high level structural exposure into more concrete coordination mechanisms across the value chain. They combine ontological uncertainty, because their future alignment is not guaranteed, with epistemic uncertainty, because knowledge about their development evolves over time. Like viability conditions, enabling dependencies cannot be fully controlled. They remain objects of acceptance at stage gates, even when efforts are made to stabilise them.

*Enabling dependencies applied to the ‘holiday’ example:*

Choices such as the type of vehicle, the selected route, or the decision to avoid certain region’s structure how safe travel is pursued and comply with the criteria for safety, while remaining subject to the same ontological uncertainty during travel.

#### Sub-Dependencies - Development Indicators

Sub dependencies, also referred to as development indicators, capture how enabling dependencies are developing over time. They reflect the degree to which relevant conditions have been established, verified, or stabilised. Unlike viability conditions and enabling dependencies, sub dependencies are subject only to epistemic uncertainty. Their direction of development can be observed, interpreted, and updated as new information becomes available. They therefore provide insight into how external developments are unfolding in relation to the project’s structural exposure. While they do not remove vulnerability, they inform whether enabling mechanisms are becoming more robust or more fragile as the project moves toward the next stage gate.

*Sub-dependencies applied to the ‘holiday’ example:*

Checks such as oil level, tyre pressure, or engine warnings provide observable indications of whether the conditions required for the chosen vehicle and route to function as intended are currently satisfied.

Taken together, this distinction clarifies the different forms of exposure embedded in the MDF. Viability conditions and enabling dependencies are exposed to structural vulnerability as a consequence of their ontological uncertainty: their future existence, stability, and effectiveness cannot be predicted or secured at the moment of commitment and must therefore be accepted as part of the project’s strategic exposure. Development indicators (or sub-dependencies), are not exposed to vulnerability in this sense. Instead, they are exposed to degrees of development, reflecting the extent to which epistemic conditions relevant to enabling dependencies have been established or verified over time. This distinction provides the conceptual basis for the following sections, which examine vulnerability and development as analytically separate but interrelated dimensions of dependency exposure.

## 5.3. Structural vulnerability of dependencies

Having defined viability and enabling dependencies as carriers of ontological uncertainty, this section characterises the extent to which these dependencies are exposed to external change. Here, structural vulnerability refers to the inherent sensitivity of a dependency to disruption, misalignment or erosion due to its reliance on external systems that are beyond the control of the project.

Importantly, vulnerability is treated here as a structural property of dependencies. It applies exclusively to viability conditions and enabling dependencies, as these concern external conditions whose future existence, stability, or effectiveness cannot be guaranteed. Their uncertainty relates to how the broader system itself may evolve, which is subjected to ontological uncertainty. Development indicators, by contrast, do not carry vulnerability, because they do not condition viability directly, but provide epistemic information about how external developments relate to existing enabling (and thus indirectly viability) dependencies. Their uncertainty concerns interpretation and knowledge, not the existence of the underlying condition itself. For this reason, development indicators are not exposed to ontological uncertainty and therefore do not possess structural vulnerability, since vulnerability is defined here as sensitivity to changes in the external environment on which viability depends. Development indicators function as observational points that make visible how enabling dependencies are evolving in relation to external conditions. This will be further elaborated in chapter 5.4, but it is important to understand that only viability and enabling dependencies and not sub-dependencies, exhibit vulnerability because of ontological uncertainty. This means that the properties of vulnerability described in following sub paragraphs are concerning viability and enabling dependencies.

### 5.3.1. Vulnerability as structural exposure

In the sustainability, risk, and systems literature, vulnerability is commonly understood as a structural and relational property of a system and its surrounding environment. It is described in terms of exposure, sensitivity, and the ability to remain functional when external conditions change, emphasising that vulnerability depends on how a system is positioned in relation to its environment rather than on isolated failures or incidents (Turner, 2003; Urruty et al., 2016).

In this research, vulnerability is explicitly applied to viability and enabling dependencies. Vulnerability is therefore structural: it describes how sensitively a dependency is positioned with respect to external uncertainty, independent of current progress, performance, or management quality.

To make this structural exposure analytically usable, vulnerability is characterised in this research through three properties that describe how a dependency relates to its environment: autonomy, bandwidth, and feedback strength. These properties are introduced as an analytical constructs that structure interpretation:

- Autonomy: captures the extent to which a dependency can remain viable without ongoing alignment from external systems.
- Bandwidth: reflects how much change in external conditions a dependency can absorb while still being considered viable
- Feedback strength: describes how strongly changes in a dependency spread and intensify effects across the wider project system.

These properties are further specified in the following sub-sections. Each is introduced conceptually before being operationalised in the case application in Chapter 6, where their combined assessment informs the overall vulnerability positioning of dependencies. Together, these properties show how vulnerability takes shape through exposure, sensitivity, and propagation within dependency structures.

Consider, for example, a game of chess and the way individual pieces are positioned on the board. A queen typically exhibits high autonomy, as she depends less on immediate support; high bandwidth, due to a wide range of possible moves; and strong feedback strength, since her loss significantly alters the configuration of the game. A pawn, by contrast, has low autonomy and limited bandwidth, and its loss usually propagates locally rather than system wide. Framing vulnerability through autonomy, bandwidth, and feedback strength makes these structural differences explicit, avoiding the ambiguity that arises when exposure, sensitivity, and propagation are treated as loosely defined or overlapping notions.

Ultimately, the characterisation of autonomy, bandwidth, and feedback strength relies on professional judgement, as experts are required to assess how the combination of these characteristics translates into a dependency's level of vulnerability. The rationale behind these combinations is further elaborated in chapter 6.3, where the model application is explained in detail. For now, details about the definitions of autonomy, bandwidth and feedback strength are repeated in appendix A.1, which also indicates the extent to which they may occur in the context of this study - their scoring possibilities.

### Autonomy

Dependencies with a low autonomy rely heavily on the stability, decisions, or cooperation of surrounding political, institutional, market, or infrastructural systems. Dependencies with higher autonomy retain their viability even when such external alignment weakens or fluctuates.

Autonomy does not refer to managerial control, contractual arrangements, or governance capacity. Rather, it reflects structural reliance: a dependency may be carefully managed and strategically prioritised yet still exhibit low autonomy if its continued viability depends on sustained external support. As such, autonomy captures how directly a dependency is exposed to changes beyond the project's control. This understanding of autonomy aligns with vulnerability frameworks that define exposure in terms of a system's dependence on external conditions rather than its internal capacity for control or management (Turner, 2003; Urruty et al., 2016).

### Bandwidth

A dependency with narrow bandwidth is highly sensitive: relatively small external changes can render it ineffective. A dependency with broader bandwidth can tolerate greater variation. This notion of bandwidth reflects the sensitivity dimension of vulnerability, which captures how much external variation can be absorbed before system functioning degrades (Turner, 2003; Urruty et al., 2016).

### Feedback Strength

Dependencies with strong feedback strength transmit local disturbances into wider system consequences. Affecting multiple interfaces, dependencies, or strategic conditions. Dependencies with weak feedback strength tend to confine effects more locally.

Feedback strength captures interdependence. A dependency may be limited in scope yet exert strong feedback if changes cascade through tightly coupled interfaces or decision sequences. This emphasis on propagation and cascading effects is consistent with interdependency-based analyses of project systems, which show how local changes can amplify across tightly coupled structures (Qazi et al., 2016; Guan et al., 2021).

Thus, viability conditions and enabling dependencies are subject to vulnerability as a consequence of ontological uncertainty. Vulnerability in this sense refers to the extent to which changes in the external environment affect a dependency's capacity to function in support of the project's scope by shaping its autonomy, bandwidth, and feedback strength.

These properties that together characterise vulnerability require interpretive assessment in order to be applied. In this research, their evaluation is conducted qualitatively, based on professional judgement and expert interpretation rather than quantitative measurement.

## 5.4. Sub-dependencies as epistemic development conditions

While vulnerability describes structural exposure, it does not explain how that exposure is currently informed by ongoing developments. Sub-dependencies - also referred to as development indicators - represent epistemic development conditions that describe how external developments relate to the assumed functioning of enabling dependencies. They capture forms of uncertainty that are knowledge-dependent and can be explored, tested, verified, or provisionally stabilised through observation, interaction, and learning. In this sense, sub-dependencies do not remove ontological uncertainty or change the underlying vulnerability of dependencies. Rather, they make visible how that exposure is currently informed by ongoing epistemic development. Their role is purely: to provide a structured basis for understanding how assumptions underlying enabling dependencies are currently holding as external conditions evolve. This is why they are referred to as sub-dependencies because they do not constitute conditions for project viability themselves.

This allows developmental information to be interpreted at moments of commitment, without treating epistemic development as a proxy for risk levels, project performance, or mitigation effectiveness.

In the holiday example, sub-dependencies refer to the conditions that allow the car to travel the required distance. Uncertainty about these conditions can be reduced by checking the state of key components or by obtaining advice from the car dealer, which helps reduce epistemic uncertainty about whether the car can perform as intended.

### 5.4.1. Epistemic role of sub-dependencies

Epistemic development should not be understood as a linear progression or a maturity sequence. Sub-dependencies may be explored, tested, verified, or provisionally consolidated, but these states reflect qualitative forms of knowledge rather than fixed stages or levels of maturity. Epistemic development in megaprojects is inherently non-linear, as knowledge is generated in open systems who's political, market, institutional, and technological conditions continue to evolve. New information may therefore undermine previously held assumptions, causing development to stall, reverse, or require reassessment.

In this sense, a sub-dependency that has been tested may revert to being explored, verified assumptions may lose validity, and previously consolidated understandings may deconsolidate as external conditions change. Such regression does not indicate project failure, declining performance, or managerial shortcomings. Rather, it is a logical and unavoidable consequence of epistemic uncertainty unfolding within ontologically uncertain environments.

Importantly, consolidation - understood here as provisional stabilisation of available knowledge sufficient to inform strategic interpretation at a given moment - should not be interpreted as certainty, completion, or reduced vulnerability. A provisionally consolidated sub-dependency merely indicates that available knowledge is sufficiently stabilised to support strategic interpretation at a given moment. Consolidation is therefore temporary, context-dependent, and inherently revisable. Even when epistemic development appears favourable across multiple sub-dependencies, the underlying vulnerability of viability and enabling dependencies remains unchanged.

This distinction is central to the MDF, as it prevents developmental information from being misread as proof of control, readiness, or robustness, and distinguishes the framework from maturity models and stage-gates.

### 5.4.2. Interpreting developmental consolidation under uncertainty

Development status is introduced to support interpretation, not evaluation: it provides insight into how epistemic conditions are unfolding without implying control, progress, or mitigation. This section introduces development status as a way to make sense of how such conditions are currently unfolding over time. It helps relate epistemic development to the assumed functioning of enabling dependencies.

Development status is in this research expressed through a green, orange, red classification. These categories reflect two epistemic dimensions: direction and degree of provisional stabilisation. Direction indicates whether observed developments align with, remain uncertain in relation to, or work against the assumed functioning of an enabling dependency. Provisional stabilisation indicates the extent to which available knowledge has become sufficiently settled to support interpretation at a given moment, while remaining revisable.

Within this logic:

- A Green-status indicates that external developments are moving in a direction supportive of the required feasibility condition and that relevant knowledge has provisionally stabilised.
- An Orange-status reflects directional alignment without such stabilisation, indicating that development remains fluid or contested.
- A Red-status signals misalignment, indeterminacy, or the erosion of previously held assumptions.

Importantly, a Green-status does not mean that a dependency is resolved, secured, or less vulnerable. It only shows that, based on current knowledge, external developments are moving in a direction that supports how the dependency is expected to function.

By keeping development status strictly epistemic, the MDF remains suitable for decision-making in situations of ontological uncertainty. At moments of commitment, such as Final Investment Decisions, this uncertainty is not removed but accepted. Development status helps decision-makers understand how current knowledge relates to that uncertainty, without treating it as a measure of risk, impact, or the effectiveness of mitigation.

A typical sub-dependency associated with an enabling dependency is the legal robustness of a required permit. A green status indicates that, based on current knowledge, the permit can be treated as a functioning enabling condition, with no substantive challenges undermining this assumption. An orange status applies when the permit remains subject to limited or procedural uncertainty, while its continued validity can still reasonably be assumed. A red status reflects a situation in which legal challenges or regulatory developments introduce substantive indeterminacy, such that the permit can no longer be relied upon as a stable assumption, even if no formal decision has yet been taken.

Together, vulnerability explains how a project is structurally exposed, while sub-dependencies clarify why this exposure becomes relevant at a given moment, based on how development conditions are unfolding. These two elements describe the nature and current interpretation of exposure.

However, understanding exposure also requires identifying where it becomes visible and how it travels across the project configuration. Structural vulnerability may exist in one part of the system, while its consequences emerge in another. To interpret this movement, uncertainty must be positioned within the broader governance domains through which dependencies interact. This step is analytically important because exposure only becomes actionable when it intersects with governance domains in which decisions are made. This will be explained in the following paragraph.

## 5.5. Contextual placement of uncertainty

Having established which dependencies exist, how vulnerable they are, and how their epistemic conditions are developing, the remaining question is where this exposure becomes visible in project governance. This section addresses the remaining question: where this exposure becomes visible and meaningful within the governance and decision-making landscape of megaprojects. To answer this question and elaborate on why this is so important, the Megaproject Dependency Framework (MDF) builds primarily on the Megaproject Uncertainty Framework (MUF), from which the MDF also derives its name (thus be aware of the difference MUF vs. MDF). The MUF - megaproject uncertainty framework - provides a clear way to locate uncertainty across recurring governance contexts.

Answering the where-question of vulnerability becoming consequential requires a view of uncertainty that goes beyond simple cause-effect relationships. Such relationships are typical of risk-based methods, which aim to predict chains of events and their impacts. In megaprojects, as outlined in Chapter 2, uncertainty is not primarily about identifiable or foreseeable events. Instead, it concerns how assumptions are continuously reinterpreted in open systems with many interacting actors, changing external conditions, and limited predictability. As a result, uncertainty rarely moves through projects in a linear way where one event directly leads to a single outcome. Rather, it becomes relevant through shifting interpretations, often with delay, feedback, and movement across different parts of the project (McDermott et al., 2024).

The MUF, from the literature, provides a conceptual basis for understanding this non-linear and interpretative propagation of uncertainty. It identifies recurring governance contexts in which uncertainty is experienced and reinterpreted, without treating these contexts as causal domains. Building on this logic, the next subsection introduces the MUF and its contextual structure. This is followed by a translation of that structure into the dependency-centred logic of the MDF.

### 5.5.1. Megaproject Uncertainty Framework - MUF

The Megaproject Uncertainty Framework (MUF), from the literature, conceptualises uncertainty in megaprojects by identifying where assumptions become contested and reinterpreted across governance contexts, rather than treating uncertainty as a sequence of risk events (McDermott et al., 2024).

The MUF, see in figure 4, builds on earlier research from Stevens' mega-systems profiler, in figure 3. This does so by translating Stevens' system-level description of large-scale socio-technical systems, into a framework tailored to a megaproject assessment tool. Where Stevens' profiler characterises the nature of mega-systems as a whole, the MUF reframes this logic to show where, within a megaproject, uncertainty is experienced and discussed (Stevens, 2011; McDermott et al., 2024). In this way, the MUF shifts attention from describing system characteristics to identifying the governance contexts in which uncertainty shapes interpretation and decision-making.

Central to the MUF, and also Stevens, is the distinction of four recurring governance contexts: Strategic, System, Implementation, and Stakeholder. These contexts indicate where uncertainty becomes relevant in practice: where assumptions are questioned, where interpretations shift, and where decisions are reconsidered. Uncertainty in megaprojects is therefore not confined to a single context, but may move across contexts through processes of reinterpretation, feedback, and delayed effects, rather than through linear cause-effect relationships. Together, these four governance contexts form a map of the uncertainty landscape experienced in megaproject, showing where and how uncertainty is experienced, discussed and reinterpreted in practice (McDermott et al., 2024).

The Stevens and the MUF are visualised as a radial diagram, illustrating how projects may range from relatively stable and predictable conditions, represented by the centre of the circles, to increasingly uncertain and interdependent situations across the four governance contexts, represented by the outside of the circles.

Importantly, this representation is intended as a sense-making and diagnostic aid, not as a causal or predictive model. It helps to locate where uncertainty is most dominant at a given moment, without implying that developments in one context directly cause outcomes in another.

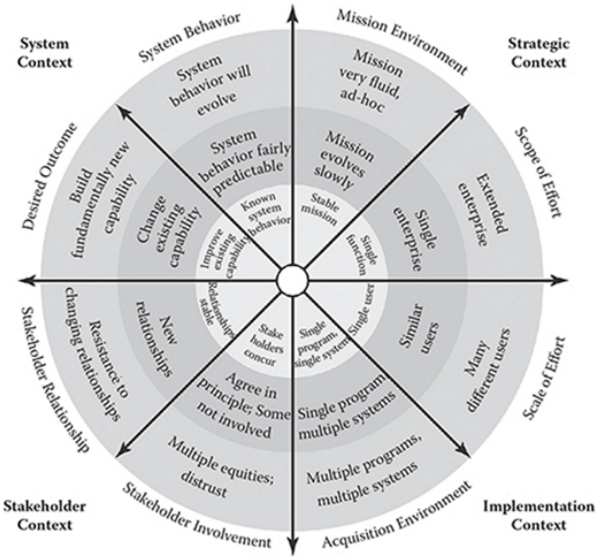


**Figure 3:** Stevens mega-systems, (Stevens, 2011)

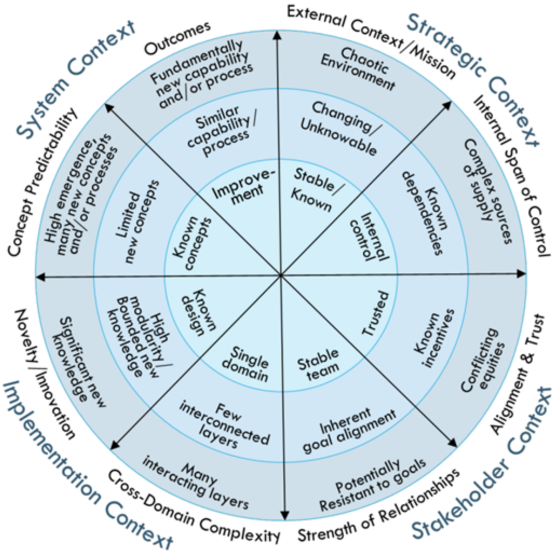


**Figure 4:** Megaproject Uncertainty Framework - MUF (McDermott et al., 2024)

## 5.5.2. Non-linear propagation across MUF contexts

Within the MUF, uncertainty does not spread through simple cause-effect chains, where a development in one context automatically leads to a specific outcome in another. Instead, uncertainty moves through changes in how assumptions are understood and discussed. When assumptions shift in one governance context, this can change how problems are framed, priorities are set, and decisions are approached in other parts of the project.

Empirical cases underlying the MUF show that strategic, system, implementation, and stakeholder contexts rarely evolve independently. Developments in one context are repeatedly reinterpreted in relation to others, often with delay and feedback. As a result, uncertainty does not remain local but travels across the project through evolving interpretations and governance choices, rather than through predictable event chains (McDermott et al., 2024).

This understanding of non-linear, cross-context propagation explains why uncertainty in megaprojects cannot be captured through causal sequencing alone. Instead, context is required to identify where uncertainty becomes relevant and consequential within the governance landscape. The contextual domains therefore provide practical anchors for locating where uncertainty attaches to the project, offering a clearer point of reference than would be possible if exposure were treated as diffuse or system-wide.

### 5.5.3. Translating MUF's uncertainty contexts into a dependency-centred placement

The translation from uncertainty to dependency framing does not extend the MUF but repurposes its contextual sensitivity for a fundamentally different analytical task. It has to be able to interpret commitment exposure at the level of concrete dependencies.

Building on the contextual logic of the MUF, the MDF keeps the distinction between the four governance contexts - Strategic, System, Implementation and Stakeholder - as interpretive lenses through which to understand where uncertainty becomes significant in the governance of megaprojects. In the MUF, these contexts together form an uncertainty landscape that supports leadership sense-making and situational awareness, without tracing those signals back to specific structural elements of the project, which is of importance in this research for a dependency view (McDermott et al., 2024). The MDF adopts this contextual sensitivity but translates it into a dependency-centred framework that anchors uncertainty in project dependencies rather than in the contexts themselves.

Whereas the MUF primarily profiles where uncertainty shows up, the MDF specifies what carries that uncertainty by linking signals in a context to the dependencies. In the MDF, uncertainty does not reside in the context categories, as is done in the MUF, but in dependencies. Governance contexts function as views on these dependencies, indicating where shifts in interpretation, discussion, or action become decision-relevant, rather than as containers of uncertainty.

This translation is particularly important for the interpretation of commitment decisions. The MUF is designed to support sense-making and situational awareness by revealing where uncertainty is accumulating or intensifying across governance contexts. It deliberately remains agnostic about which specific project elements are implicated, because its primary function is to surface and visualize patterns of uncertainty rather than, the MDF, to analyse commitment positions at the level of concrete dependencies.

The MDF does not seek to extend or refine this function, but to build on it for a different analytical purpose. It starts from the recognition that moments of strategic commitment require uncertainty to be interpreted in relation to specific project assumptions. The MDF therefore specifies which dependencies are exposed, how they are exposed and in which governance context this exposure becomes relevant. In doing so, it connects contextual signals of uncertainty to the concrete assumptions on which commitment decisions rest.

Through this approach, the MDF retains the MUF's non-linear and interpretive view of how uncertainty develops and spreads in megaprojects, while adding more concrete structure to support interpretation. It therefore allows decision-makers to connect signals of increasing uncertainty to a clearly defined set of dependencies, instead of only uncertainty. The importance of this is to able understand how these dependencies are vulnerable, and to identify in which underlying governance context they need to be reconsidered before a commitment is confirmed.

After all, the MUF and the MDF are not closely related frameworks and should not be read as variations of the same model. They serve different purposes and operate at different levels of abstraction. The MUF is designed as a high-level learning and assessment framework to support situational awareness and future AI-supported uncertainty monitoring. The MDF, by contrast, is a dependency-based framework intended to interpret concrete project exposure in decision-making. The connection between the two lies only in a limited set of shared principles - most notably the use of governance contexts as interpretive lenses and the recognition that uncertainty propagates in a non-linear way. Beyond these elements, the MDF departs from the MUF in structure, focus, and analytical intent.

## 5.6. The MDF as an integrated analytical framework

By doing this, the MDF does not eliminate uncertainty; rather, it makes explicit what is accepted when a commitment is made under uncertain conditions. This chapter has defined all the main characteristics of the Megaproject Dependency Framework (MDF) that is needed for the analytical framework to interpreting uncertainty at stage gates. The MDF structures uncertainty by distinguishing four complementary dimensions: dependencies, vulnerability, epistemic development, and contextual placement (derived from the MUF).

The MDF integrates four analytical questions - what, why, how, and where - to support the interpretation of uncertainty at moments of strategic commitments.

Dependencies specify what external conditions must hold for project viability. Both viability and enabling dependencies are subject to ontological uncertainty and are taken into account when making decisions about acceptance and commitment. Structural vulnerability characterises how exposed these dependencies are, through autonomy, bandwidth, and feedback strength. Sub-dependencies explain why exposure becomes important by capturing epistemic development conditions that shape how assumptions about enabling dependencies are currently holding.

Contextual placement indicates where this exposure becomes governance-relevant, using the four megaproject uncertainty contexts as interpretive lenses. It becomes governance-relevant when it enters the domains in which strategic choices, commitments, or adjustments can be made, influencing how situations are interpreted and how confidently such decisions can be taken.

Together, these elements provide a structured way to interpret dependency-related exposure without reducing uncertainty to risk, causality, or mitigation.

# 6

# Overview of the Megaproject Dependency Framework

This chapter demonstrates how the Megaproject Dependency Framework (MDF) can be applied in practice. The chapter deliberately adopts a structured and tool-like format to make the individual components of the framework explicit and traceable. The purpose is to show how dependency-related exposure can be interpreted in a systematic and replicable way.

Specifically, the application demonstrates how exposure can be categorised according to three complementary dimensions. Firstly, it demonstrates where uncertainty becomes relevant to governance by positioning dependencies within the four governance contexts derived from the MUF.

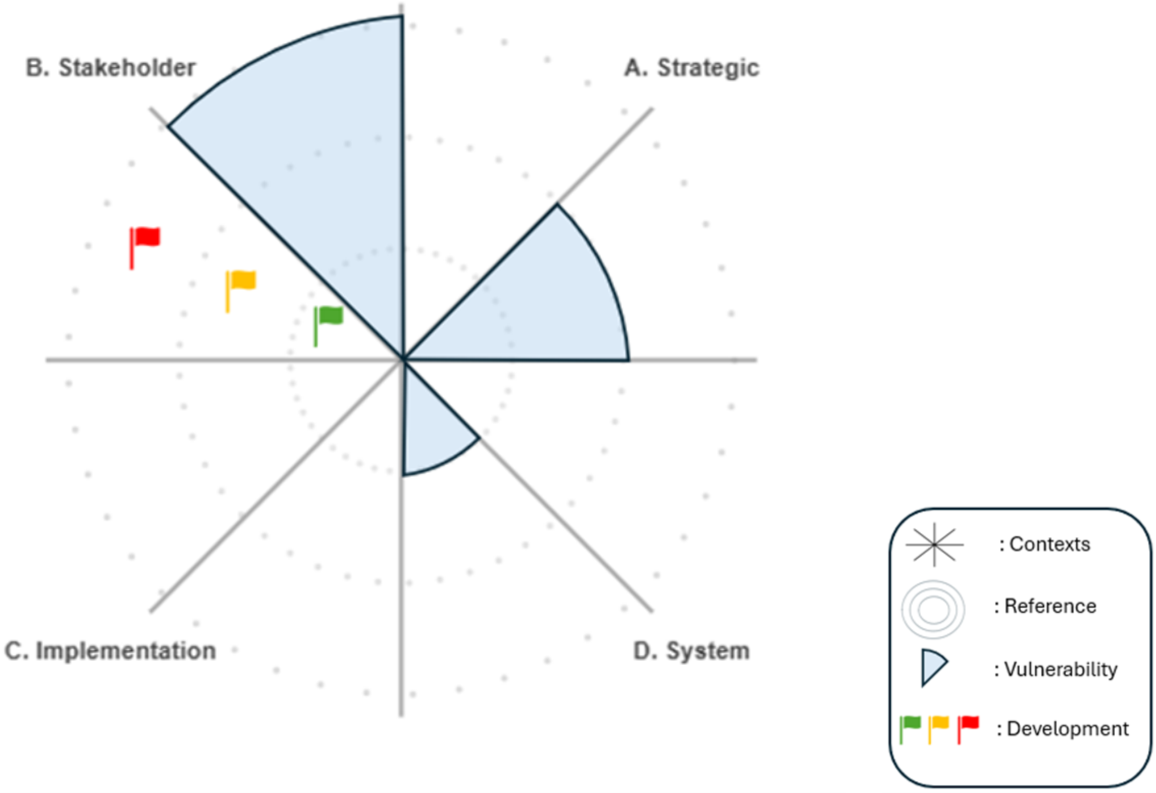


**Figure 1:** Application MDF. The right panel provides the legend.

Secondly, it explains how exposure arises by characterising the structural vulnerability of these dependencies. Thirdly, it clarifies why such exposure may become more significant over time by examining the epistemic development conditions associated with sub-dependencies. Together, these elements provide a coherent structure for interpreting uncertainty, rather than reducing it to discrete risks or outcomes.

Figure 1 presents the integrated application of the MDF, with a legend. The example illustrates three different degrees of vulnerability, as indicated by the three dotted circles, across contexts: high vulnerability in the Stakeholder sub-context, moderate vulnerability in the Strategic sub-context, and low vulnerability in the System sub-context. Figure 1 also displays three development conditions within the Stakeholder sub-context, indicated by coloured flags corresponding to the inner, middle, and outer rings.

## 6.1. Contexts

The MDF uses the same four governance contexts as Stevens and the MUF - Strategic, Stakeholder, Implementation, and System – as it organises the interpretation of uncertainty, see Figure 2. Together, these contexts provide a structured reference frame for locating where uncertainty becomes governance relevant.

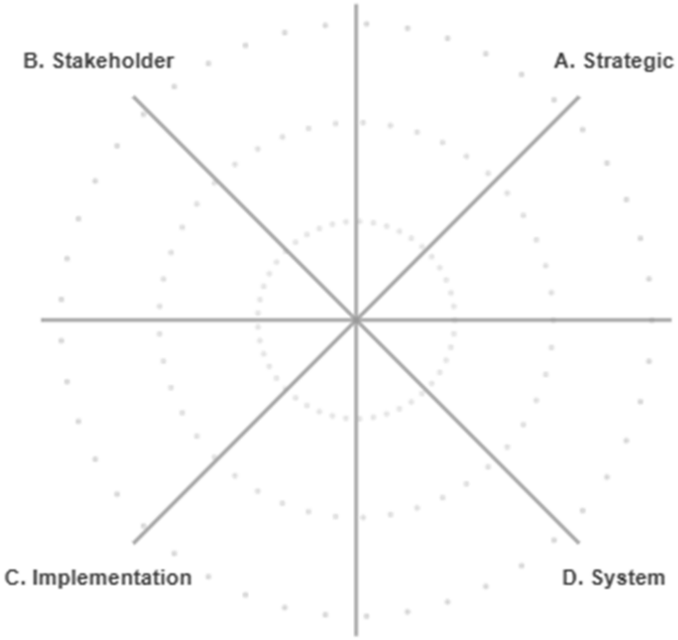


**Figure 2:** Governance contexts

The contexts represent distinct but interconnected domains in which project assumptions are interpreted, discussed, and acted upon:

**(A) Strategic:** concerns the external policy, regulatory, and market conditions that determine whether the project is permissible, attractive, and viable in principle.

**(B) Stakeholder:** concerns the governance relationships and sources of legitimacy that shape decision authority, coordination between organisations, and political durability.

**(C) Implementation:** concerns the conditions under which the project can be realised in practice, including multi-project coordination, interfaces, and project-specific approvals.

**(D) System:** concerns the coherence of the system concept and the ability of the integrated value chain to deliver the required performance.

At this stage, the contexts are intentionally kept generic and unfilled. They indicate where uncertainty may become relevant, without specifying which concrete project elements are involved.

Following the logic of the MUF, each context can be refined into project-specific sub-contexts. This refinement results in eight sub-contexts as the subdivision of the four quadrants, represented in figure 2 (also represented in figure 3 and figure 4 but to other situations / applications). These sub-contexts do not introduce new dimensions of uncertainty but specify how uncertainty manifests within a particular project setting.

Sub-contexts are used to anchor dependencies to concrete elements of the project environment, such as policy regimes, market commitments, governance arrangements, interface complexity, or system-level performance expectations. They are not introduced here. In this chapter, the analysis remains at the level of the four primary contexts; the specification of sub-contexts follows in the next chapter, where the MDF is applied to a concrete project case.

## 6.2. Propagation

In the MUF, the four megaproject governance contexts - Strategic, System, Implementation, and Stakeholder - are introduced precisely because uncertainty in megaprojects cannot be meaningfully assessed through specific cause–effect relationships (McDermott et al., 2024). Rather than tracing causal chains, the MUF treats uncertainty as propagating through changes in how assumptions are interpreted, discussed, and acted upon across these contexts. Together, the four contexts provide a complete interpretive landscape for positioning uncertainty within governance domains, without first having to establish direct causal sequences.

The MDF adopts this contextual logic but applies it to a different analytical object and in a more constrained form. Instead of analysing uncertainty in abstract terms, the MDF focuses on specific dependencies. Each dependency bundles assumptions about external conditions that are critical to project viability. When uncertainty affects these assumptions, it is immediately clear what element of the project is implicated, because the uncertainty is already embedded in the dependency itself.

However, the governance context in which a dependency becomes uncertain is often not the context in which its implications are first experienced. A shift in underlying assumptions may originate in one governance domain, while its effects become visible and actionable in another. Propagation is therefore introduced to distinguish between the context in which uncertainty is positioned within a dependency and the context in which its effects first surface in governance and decision-making.

To keep this interpretation tractable, the MDF assigns each dependency a single primary context - where the uncertainty carried by the dependency is most directly relevant - and a single secondary context - where its effects are most likely to be felt through feedback, reinterpretation, or interaction with adjacent logics. This dual placement is a deliberate analytical simplification. It makes dependency-related exposure interpretable without reverting to linear cause-effect reasoning.

By constraining propagation to a primary and secondary context, the MDF departs from the MUF's holistic re-evaluation of all four contexts, while retaining the same fundamental rationale: contexts are used to interpret uncertainty precisely because causal modelling is insufficient for understanding how uncertainty becomes governance-relevant in megaprojects (McDermott et al., 2024).

### 6.2.1. Why propagation is essential

Propagation is analytically necessary because it transforms uncertainty from an abstract condition into governance-relevant exposure. Without propagation, a dependency may be identified as uncertain, but it remains unclear where within the governance system this uncertainty becomes consequential. Building on the insight that megaproject failures emerge from uncertainty spreading across multiple domains rather than remaining confined to a single silo, the contextual logic adopted from the

Megaproject Uncertainty Framework provides a way to locate where uncertainty matters without relying on causal modelling.

By assigning each dependency a primary and a secondary context, the MDF makes visible which governance arenas are likely to confront the uncertainty first, and where secondary effects are likely to surface. In this way, propagation localises uncertainty within the governance structure instead of treating it as a diffuse, system-wide condition, which is done in the MUF.

This propagation logic still aligns with insights from the mega-system research of Stevens (2010) and McDermott et al. (2024), who emphasise the interdependence between strategic assumptions, system design, implementation dynamics, and stakeholder relations. Their analyses suggest that disruptions in megaprojects rarely remain confined to a single domain but reflect interactions across domains that were assumed to be loosely coupled. In this sense, propagation analysis does not attempt to establish direct causal chains, but to make visible how shifts in one domain may structurally relate to consequences in others.

At the same time, limiting propagation to a primary and secondary context preserves analytical tractability. This constraint avoids an “everything affects everything” representation, while retaining the essential insight that uncertainty is relational, non-linear, and cross-contextual. Propagation therefore acts as a deliberate abstraction, enabling a more focused interpretation of uncertainty in governance and decision-making without disregarding the underlying complexity of megaproject systems.

## 6.3. Vulnerability score – viability and enabling Dependencies

In the MDF application, vulnerability is represented by the position and surface area within the context to which the viability and enabling dependencies are assigned, see figure 3. It illustrates how vulnerabilities vary across dependencies within three sub-contexts. No propagation is considered.

The surface area assigned to a dependency reflects its degree of vulnerability and is determined by the combined characterisation of autonomy, bandwidth, and feedback strength. These properties are not interpreted separately or aggregated into a numerical score; instead, their combination is used to assign each dependency to one of three qualitative rings - inner, middle, or outer. The combination, seen in table 6.1, is read per row, resulting in a ring value – the fourth column.

The mapping of these property combinations and ring placement is defined based on expert judgement. Dependencies positioned in the inner ring are structurally more resilient, while dependencies positioned toward the outer ring are more vulnerable, for example when low autonomy and narrow bandwidth coincide with strong feedback effects it has an outer ring vulnerability. The table shows that autonomy and feedback strength are assessed on a low-medium-high scale, while bandwidth is assessed on a low–high scale. These ratings are chosen to support consistent expert judgement.

In the illustrative figure 3, three vulnerabilities of dependencies are positioned across different contexts. One dependency is located in the outer ring within the Stakeholder context, indicating a high degree of structural vulnerability. A second dependency occupies the middle ring in the Strategic context, reflecting moderate vulnerability, while a third dependency is positioned in the inner ring within the System context, indicating comparatively low vulnerability.

Definitions and scoring of its autonomy, bandwidth and feedback strength can also be found in Appendix A Model Application. Appendix A Model Applications contains appendixes: A.1, A.2, A.3, A.4 and are all appendixes showing information applicable to the model characteristics.

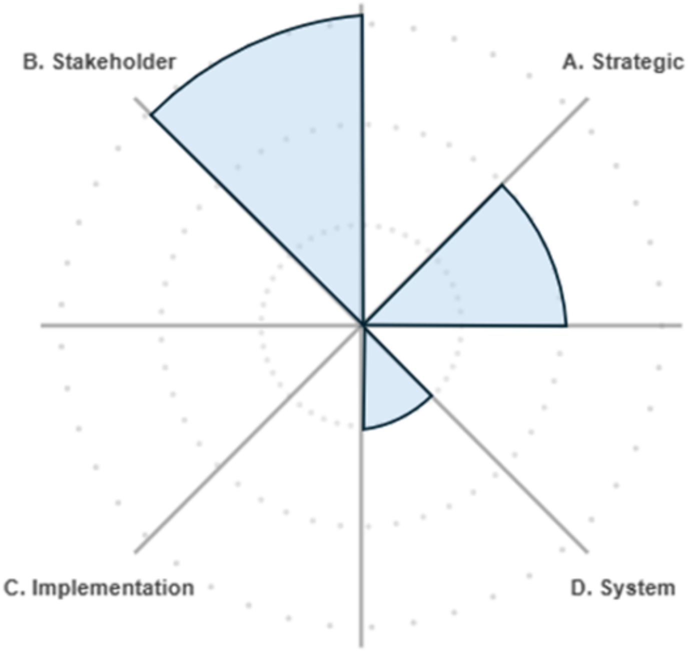


**Figure 3:** Vulnerabilities

| Autonomy | Bandwidth | Feedback Strength | Ring |
|---|---|---|---|
| Low | Low | Low | **OUTER** |
| Low | Low | Medium | **OUTER** |
| Low | Low | High | **OUTER** |
| Low | High | Low | **MIDDLE** |
| Low | High | Medium | **MIDDLE** |
| Low | High | High | **OUTER** |
| Medium | Low | Low | **MIDDLE** |
| Medium | Low | Medium | **MIDDLE** |
| Medium | Low | High | **OUTER** |
| Medium | High | Low | **MIDDLE** |
| Medium | High | Medium | **MIDDLE** |
| Medium | High | High | **MIDDLE** |
| High | Low | Low | **MIDDLE** |
| High | Low | Medium | **MIDDLE** |
| High | Low | High | **OUTER** |
| High | High | Low | **INNER** |
| High | High | Medium | **INNER** |
| High | High | High | **MIDDLE** |

**Table 6.1:** Ring classification as a function of autonomy, bandwidth, and feedback strength.

## 6.4. Development indicators

In the MDF application, sub-dependencies are used to indicate how the epistemic conditions underlying a dependency are currently developing.

Development indicators are aligned with the radial structure of the map: flags are shown in green when positioned in the inner ring, orange in the middle ring, and red in the outer ring. See figure 4.

For example, within the Stakeholder context, one sub-dependency may carry a green flag reflecting stable governance arrangements, while another shows an orange flag, or a red flag indicating that evolving political or legitimacy conditions are undermining key assumptions. The definitions of a green, orange or red development indicator are repeated in Appendix A.3.

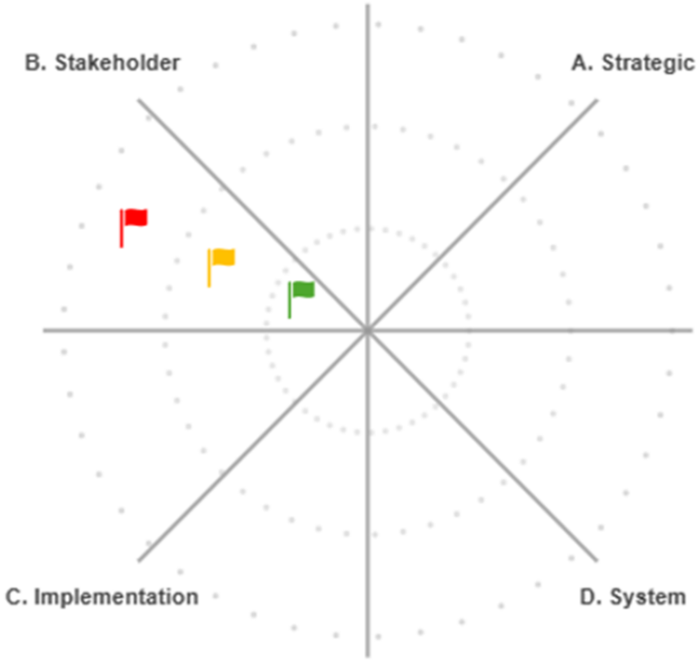


**Figure 4:** Sub-Dependencies (Development Indicators)

## 6.5. MDF application example

To illustrate the application of the MDF consider the simplified holiday ‘project’. The overall project scope: a *successful* holiday. Which is defined by three viability conditions: *safe* travel, *comfortable* accommodation, and *enjoyable* activities. All three conditions must hold for the holiday to be considered *successful*.

Focusing on the viability condition safe travel, the holiday project depends on a set of enabling dependencies which form the basis of this safe travel. One of which is the availability and functioning of a car. In this example, the car serves as the enabling dependency through which safe travel is realised.

The car, as an enabling dependency, can thus be characterised in terms of vulnerability. It exhibits high autonomy, as it can remain operational without continuous alignment from external systems - for example, unlike air travel, which depends on airlines, airports, and scheduling systems beyond the traveller’s control. At the same time, the car has low bandwidth, since relatively small deviations from assumed conditions, such as mechanical issues or delays, can already compromise safe travel. Its feedback strength is assessed as medium, as problems with the car propagate to other aspects of the holiday - such as arrival timing, accommodation check-in, and planned activities - but do not automatically destabilise the holiday as a whole. As shown in Table 6.1 (also repeated in Appendix A.2), the combination of high autonomy, low bandwidth, and medium feedback strength places the car in the middle ring, indicating a moderate level of structural vulnerability.

The car is primarily positioned in the Strategic sub-context, see figure 5, as its availability fundamentally shapes the feasible scope of the holiday; without a functioning car, entirely different destinations or travel arrangements would be required. The implications of uncertainty in this dependency are also felt in the Stakeholder context, as the family depends on this specific vehicle for safe and comfortable travel. Propagation therefore reflects that uncertainty embedded in the car primarily affects strategic choices, while also having a secondary impact on stakeholders. It could be argued that it would also cause implementation uncertainty, but in this case, the stakeholder share appears to be of greater importance and is therefore the secondary chosen context.

In addition, the example illustrates how development indicators further qualify this vulnerability. A warning signal from the engine results in a red flag in the System sub-context, indicating that current technical conditions undermine key assumptions about the car's reliability. An upcoming annual inspection generates an orange flag in the Implementation context, reflecting uncertainty related to maintenance and readiness. These sub-dependencies may themselves propagate: technical issues signalled at the system level can heighten safety concerns for stakeholders, while unresolved maintenance issues may also affect strategic confidence in the viability of the travel plan. The secondary placement of sub-dependencies is indicated by small star symbols shown alongside the flags, highlighting where development conditions may also become relevant beyond their primary context.

This illustrative example represents only a limited subset of the viability and enabling dependencies that may determine a successful holiday. Even within this narrow scope, the analysis demonstrates that, in specific governance contexts, uncertainty associated with a single enabling dependency can become apparent, though not dominant. In this case, uncertainty related to the car is most visible in the Strategic context, while also affecting the Stakeholder context in a secondary manner. The example demonstrates how the MDF supports nuanced interpretation of uncertainty - showing neither stability nor failure, but a condition of moderated exposure - by making visible where uncertainty is located, how

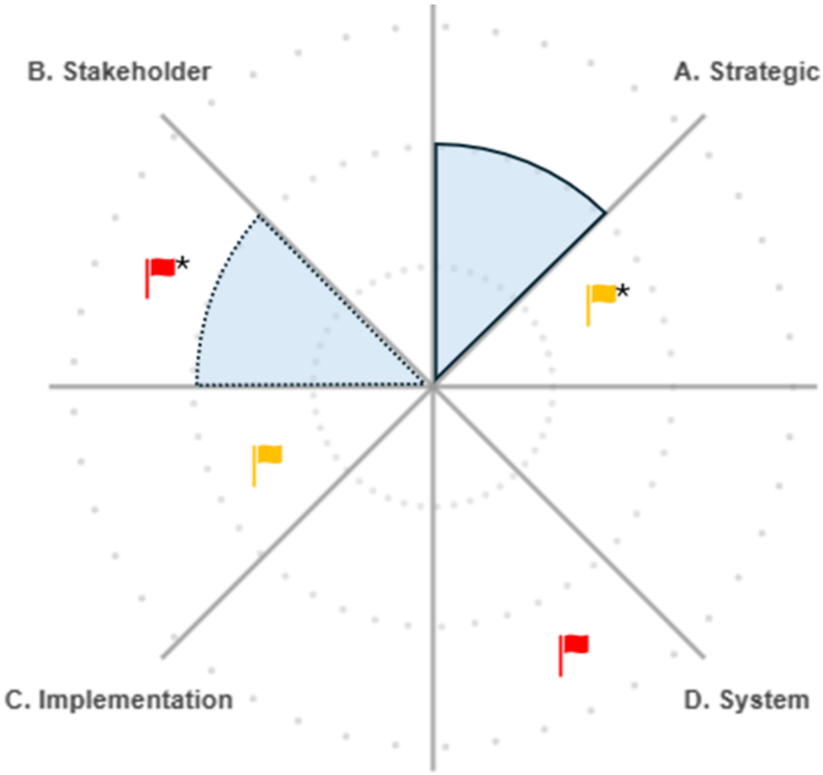


Figure 5: Holiday Example with Enabling Dependency: Car

it propagates, and which contexts are most affected under conditions of incomplete information. This simple example illustrates that strategic choices - the car - regarding a single enabling dependency can generate concentrated vulnerability within stakeholder - the family - domain.

In a conventional setting, such a situation could be analysed through a risk-based approach, where the likelihood of failure and its consequences are estimated and mitigated. However, this assumes that the relevant failure modes are sufficiently known. If the car were instead a first-of-a-kind technology, its behaviour under real-world conditions could not be reliably anticipated, and the associated risks would be difficult to define in probabilistic terms. In such cases, the analytical focus shifts from estimating risks to understanding how strongly the project depends on this element, how sensitive that dependency is to external variation, and how disturbances would propagate through the system. This is where the MDF provides additional insight.

With the analytical components of the MDF now specified and illustrated, the framework can be applied to a real-world megaproject to examine how vulnerability and robustness manifest under concrete governance conditions.

# 7

# Case-Study: Aramis CCS

The Aramis initiative concerns the development and operation of a 22 mega-ton $CO_2$ per annual open-access transport and storage infrastructure. The project is developed by a consortium consisting of Shell, TotalEnergies, Gasunie, and Energie Beheer Nederland. Aramis enables $CO_2$ captured at industrial facilities to be transported to depleted gas fields under the North Sea, where it can be stored permanently. In doing so, the project supports Dutch and European climate objectives, particularly for industrial sectors where emissions are difficult to eliminate in the short term.

Aramis is designed as an open-access, multi-user CCS infrastructure. It connects multiple industrial $CO_2$ sources to multiple offshore storage sites through a shared transport system. Rather than being developed for a single emitter or a single storage field, the infrastructure is intentionally structured to expand over time, allowing additional sources, transport routes, and storage locations to be connected in phases. In this study, the term Aramis initiative therefore refers specifically to the $CO_2$ transport infrastructure that links onshore industrial capture with offshore geological storage. This design reflects the expectation that CCS will be deployed as a multi-actor system, in which several emitters and storage operators rely on a common transport backbone.

Together with $CO_2$ capture at industrial sites and permanent offshore storage, Aramis forms part of an integrated CCS value chain. This value chain consists of a sequence of connected segments, each of which contributes to the overall functioning of the system. See figure 1.

## 7.1. The Aramis CCS value chain

### 1. $CO_2$ capture - industrial emitters

$CO_2$ is captured at industrial installations such as refineries, chemical plants, and hydrogen or power facilities. These capture assets lie outside the Aramis project boundary, but their timely development, volume delivery, and contractual commitment are essential for the viability of the overall system. In this segment, individual industrial emitters are the primary actors.

### 2. Onshore transport to the Maasvlakte - pipelines and shipping

Captured $CO_2$ is transported to the Rotterdam/Maasvlakte area either via onshore pipelines or by ship, depending on the location and connectivity of the emitter. This dual access route is a defining feature of the Aramis system, as it allows both pipeline-connected industrial clusters and more remote or international sources to feed into the CCS chain. In this onshore segment, Aramis interfaces with two closely related CCS initiatives that provide complementary access routes to the Maasvlakte hub:

$CO_2$Next, a joint project by Gasunie, Vopak, Shell, and TotalEnergies, which aims to develop an open-access terminal for liquid $CO_2$ in the Port of Rotterdam. $CO_2$ next enables shipping-based supply of $CO_2$ and provides flexibility for emitters that are not connected to an onshore pipeline network.

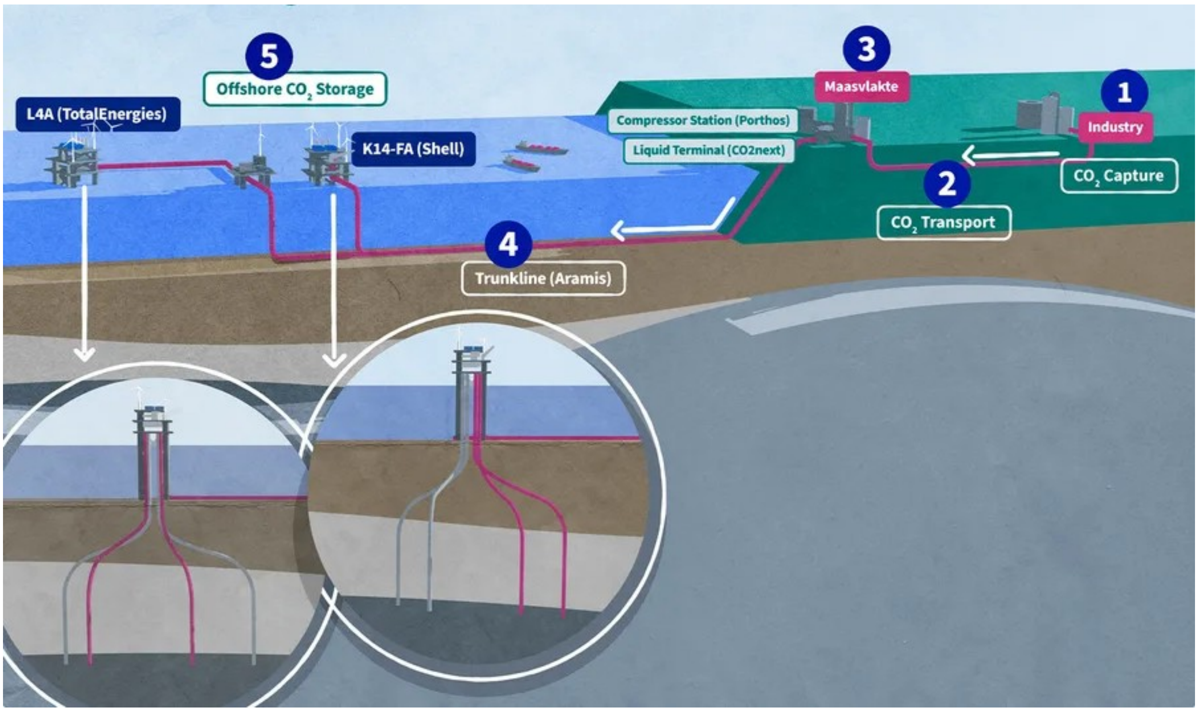


**Figure 1:** Value Chain Aramis Initiative, (Media - Gasunie; https://www.gasunie.nl)

Porthos, a project for $CO_2$ transport and storage in the Port of Rotterdam, developed as a partnership between Energie Beheer Nederland (EBN), Gasunie, and the Port of Rotterdam Authority. Porthos provides pipeline-based $CO_2$ transport from industrial emitters in the Rotterdam area to offshore storage and is physically and operationally connected to the Aramis system at the Maasvlakte. In addition, Porthos is responsible for the development of the $CO_2$ compression facilities at the Maasvlakte, which will also be used by the Aramis system.

### 3. Collection hub at the Maasvlakte - terminal and compression

At the Maasvlakte, $CO_2$ streams are brought together at a collection manifold (also referred to as the mixing point), located downstream of the Porthos compressor stations and the $CO_2$Next heat exchangers. At this point, the conditioned $CO_2$ streams are combined into a single flow before entering the transport system. Gas-phase $CO_2$ delivered by pipeline is compressed to the pressure required for offshore transport, while liquid $CO_2$ delivered by ship is received, temporarily stored, and pumped onward. The terminal function is developed in cooperation with $CO_2$next, while compression is coordinated with infrastructure developed under Porthos. At this point, different inflows are combined into a single offshore transport stream.

### 4. Offshore transport - the Aramis trunkline

From the Maasvlakte, $CO_2$ is transported via the Aramis 200 km offshore trunkline in North-West direction to where most of the Dutch North Sea gas reservoirs are located. This trunkline forms the backbone of the Aramis system and is designed to carry large volumes of $CO_2$ over long distances, enabling connection to multiple offshore storage locations. This 200 km offshore trunkline and a small part that is onshore, is the scope of the Aramis Joint Team to deliver. The scope of the Aramis Joint Team therefore covers the physical CCS infrastructure, including approximately 10 km of onshore pipeline, 200 km of offshore trunkline, and the distribution hub facility.

### 5. Offshore distribution-hub - platform and distribution

At the offshore distribution-hub (D-HUB), $CO_2$ is routed through different injection platforms. This d-hub configuration allows storage sites to be connected modularly and supports phased system expansion. Multiple storage operators' interface with the transport system at this point.

**6. Offshore injection and permanent storage - depleted gas fields**

From the injection platforms, $CO_2$ is injected and stored in depleted gas fields 3 to 4 km below the North Sea seabed via wells by storage companies TotalEnergies, Shell, Eni Energy Netherlands and other storage facilities.

CCS initiatives connected to the Aramis system

Within this evolving system, several CCS initiatives are relevant because they either already provide access to $CO_2$ transport and storage or are expected to do so in the coming years, most of which are linked to Aramis, seen in Figure 2. Two of the most important thus: Porthos and $CO_2$Next.

Porthos represents a closed-system CCS configuration focused on a defined group of emitters, with a 30 km onshore pipeline, 20 km offshore and an offshore storage site. Its earlier operational timeline establishes an initial $CO_2$ transport and storage route in the Rotterdam area, creating a functional reference point for large-scale CCS deployment prior to the commissioning of Aramis.

$CO_2$next, by contrast, is designed as an open-access entry point to the CCS system, specifically addressing liquid $CO_2$ delivered by ship. Its role is not to transport $CO_2$ offshore itself, but rather to facilitate the collection, temporary storage and subsequent delivery of $CO_2$ into shared transport infrastructure. This lowers the entry barriers for emitters without pipeline access and enables cross-border $CO_2$ flows.

Beyond the Rotterdam area, several corridor projects aim to connect additional industrial regions to the Dutch CCS backbone. The Delta Rhine Corridor is a planned bundle of underground pipelines for $CO_2$ and hydrogen between the Port of Rotterdam, the Moerdijk industrial area, and the German border. The Delta Schelde $CO_2$-connection aims to connect the Rotterdam region with Zeeland and the Antwerp area. In the long term additional connections could be developed to for example the Eemshaven or Chemelot.

Together, these initiatives illustrate that the Aramis Initiative is embedded in a multi-project, cross-border CCS landscape, in which transport infrastructure, terminals, and storage developments are gradually interconnected. This broader system context is important for understanding Aramis not only as a single project, but as a central component within a growing European $CO_2$ transport network.

**Figure 2:** Overview of CCS Projects linking Aramis

## 7.2. Delineation of sub-contexts

Within the MDF, uncertainty contexts are used to locate and interpret where dependency-related uncertainty becomes governance-relevant for the project. The four higher-level context sets - strategic, stakeholder, implementation, and system - serve as organising categories, while the eight sub-contexts (A1-D2) specify the distinct domains in which such uncertainty is interpreted.

These sub-contexts do not represent project phases, risk categories, or managerial responsibilities. Instead, they delineate where uncertainty originates and how it enters the project's governance space, that is, where it becomes relevant for interpretation, discussion, and decision-making. Where uncertainty becomes governance-relevant is therefore not given a priori but must be specified in relation to the project under study.

In the MDF, this specification is explicitly case-based. For the Aramis project, the contextual structure is derived from detailed project knowledge and professionals and is intended to organise where uncertainty related to fundamental dependencies becomes visible and meaningful within the project's governance landscape.

The resulting set of sub-contexts is designed to be mutually exclusive and collectively exhaustive (MECE). Each sub-context delineates a distinct domain in which uncertainty associated with project dependencies may surface and be interpreted, without implying that the contexts themselves constitute conditions of project viability or performance.

Taken together, the contextual structure spans the full range of domains through which uncertainty affecting project viability may surface: from policy and market foundations (A1-A2), through inter-organisational governance and external legitimacy (B1-B2), to multi-project coordination and project-specific regulatory approvals (C1-C2), and finally to system-level configuration and required value-chain performance (D1-D2). In this way, the contexts structure how uncertainty related to dependencies is understood and addressed. In figure 3, all contexts are shown (context and definitons are also repeated in Appendix A.4). What they represent in the case of Aramis will now be explained.

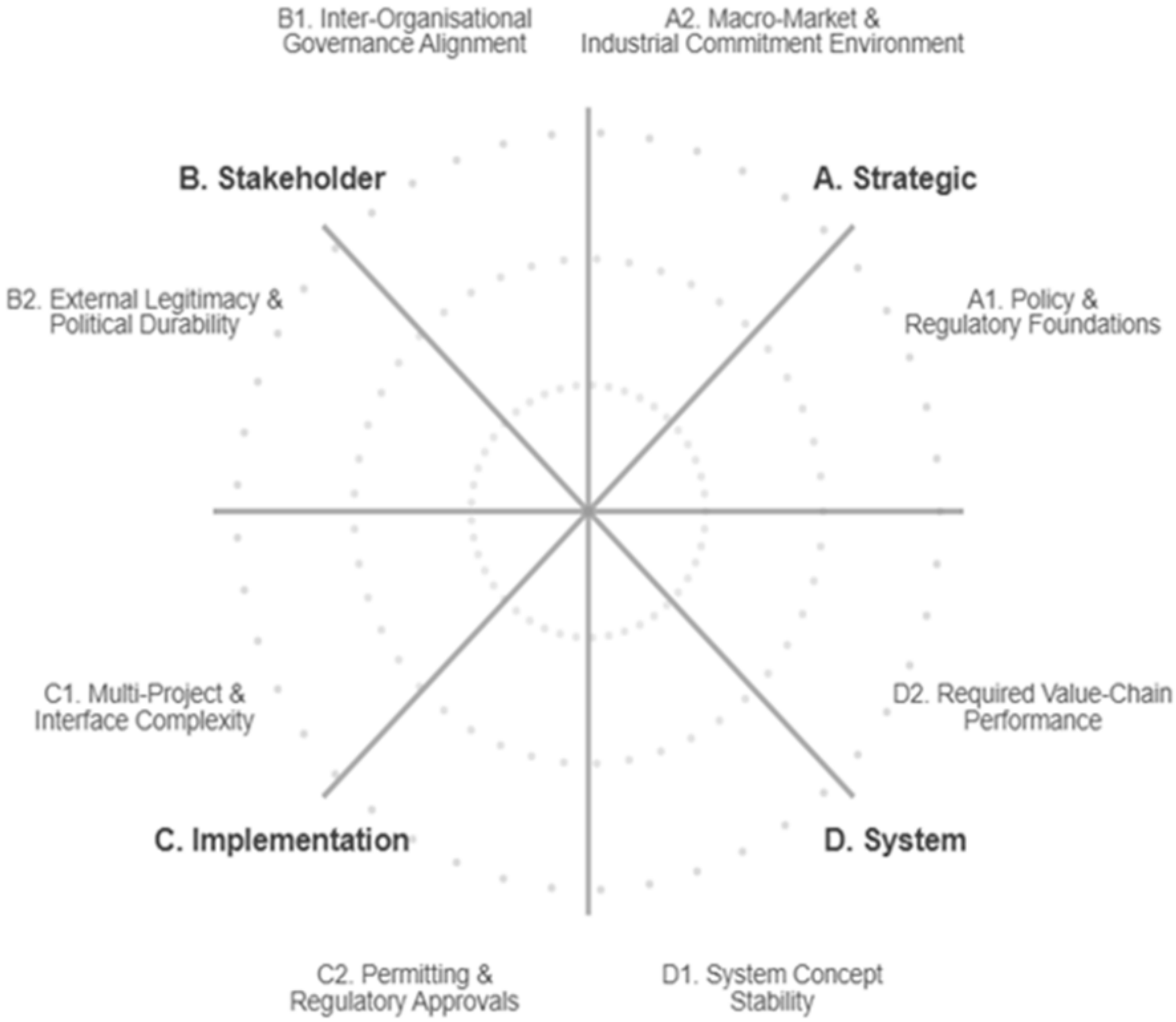


**Figure 3:** Contexts & Sub-Contexts

**Context A - Strategic context**

The strategic context captures exogenous conditions that make the project permissible, attractive, or viable in the first place. These conditions lie outside direct project control and determine whether the Aramis system can legitimately exist within its broader institutional and economic environment.

- A1 - Policy & Regulatory Foundations

This context concerns uncertainty related to the institutional and policy regime within which the project operates. It addresses how stable and predictable the rules of the game are, rather than whether a specific approval is granted. Relevant uncertainties include the design and durability of subsidy schemes (such as SDE++), ETS rules and price formation, state-aid frameworks, statutory obligations, and the functioning of the permitting regime as a system. A1 therefore addresses the question: *“Can this project exist within the policy and regulatory system?”*

- A2 - Market & Industrial Commitment

The market context captures uncertainty about whether and how market actors commit to the project. This includes investment decisions by emitters, suppliers, and off-takers; volume commitments; business case framing; first-mover dilemmas; and relocation or exit options. The focus is on participation and commitment behaviour, not on technical design or governance arrangements. A2 therefore addresses the question: *“Do parties participate, when, and under what conditions?”*

### Context B - Stakeholder Context

The stakeholder context concerns the relationships between organisations and their environment, shaping decision-making capacity, coordination, and legitimacy.

- B1 - Inter-Organisational Governance

This context captures uncertainty about how parties cooperate and make decisions across organisational boundaries. Relevant issues include unclear roles and responsibilities, contractual inconsistencies, interface governance, the mandate of the system integrator, and the presence (or absence) of back-to-back logic between contracts. Market choices and technical interfaces are explicitly excluded from this context. B1 therefore addresses the question: *“Who decides what, when, and with which mandate?”*

- B2 - External Legitimacy  Political Durability

B2 concerns uncertainty about external acceptance and political robustness of the project as a whole. This includes public acceptance, political support, fairness and distributional concerns, safety perceptions, and the accountability position of public authorities. The focus is not limited to NGOs or opposition groups, but on whether the project remains politically defensible over time. B2 therefore addresses the question: *“Can this project remain viable in the public and political domain?”*

### C - Implementation Context

The implementation context captures uncertainty related to how the project is realised in practice, particularly where multiple projects and decision trajectories need to be aligned.

- C1 - Multi-Project  Interface Complexity

This context concerns uncertainty arising from coordination between parallel projects and systems. It includes synchronisation challenges, parallel development paths, interface dependencies, sequencing issues, and inter-project planning. It explicitly excludes governance mandates and fundamental system design. C1 therefore addresses the question: *“Do all moving parts come together at the right time?”*

- C2 - Permitting  Regulatory Approvals

C2 captures uncertainty related to concrete, project-specific approvals required prior to final investment decisions. This includes individual permits, legal finality, safety case acceptance, procedural requirements, and approval timelines. Unlike A1, the focus here is not on the regime, but on definitive project-level clearance. C2 therefore addresses the question: *“Do we obtain final legal approval for this project?”*

### Context D - System Context

The system context captures uncertainty related to the properties of the system as a whole, which ultimately determine what the project can deliver.

- D1 - System Concept Stability

D1 concerns uncertainty about the fundamental system concept. It includes architectural choices, role allocation across the value chain, and the definition and boundaries of the system itself. It explicitly excludes questions about operational performance. D1 therefore addresses the question: *“Do we know what kind of system we are building?”*

- D2 - Required Value-Chain Performance

D2 concerns uncertainty about whether the system can deliver the level of performance that has been agreed across the value chain. This includes risks of insufficient volumes, availability problems, injectivity limitations, and ramp-up failure. D2 therefore addresses the question: *“Can the system actually perform as required?”*

Use of Contexts in the Case Analysis

In the subsequent sections, each fundamental dependency identified for the Aramis project is assigned a primary context, indicating where uncertainty primarily originates, and a secondary context, indicating where its effects are likely to propagate first. This structured contextual placement enables interpretation of exposure.

## 7.3. Dependency identification

To identify the viability, enabling, and sub-dependencies of the Aramis project, the scope of the project must first be made explicit. For Aramis, this scope is defined as:

*“Establish an open access, efficient and non-discriminatory $CO_2$ transportation system to facilitate the abatement of emissions of Dutch and European industrial emitters, up to 22 Mega-ton per annual (MTPA) through a public/private partnership.”*

This scope functions as the analytical starting point for dependency identification, as it specifies both the intended function of the system and the external conditions on which its strategic relevance depends.

From this scope, it follows logically that the value chain must be established, as it makes explicit how Aramis can achieve strategic success if certain external requirements hold. In functional terms, the Aramis value chain comprises industrial emitters that capture and deliver $CO_2$; transport via onshore feeder pipelines and/or shipping; aggregation and conditioning at a $CO_2$ collection hub (terminal and compression); offshore transport through a trunkline and distribution hub; and final injection of offshore storage capacity. Each of these components must function in alignment for Aramis to operate as an open-access CCS system.

Parallel to these value-chain components, viability dependencies are defined. These represent the external, ontologically uncertain requirements that must continue to hold for Aramis to remain viable, given its open-access design and scale intent. In other words, they express the fundamental prerequisites for the system to function as intended. The following six viability dependencies are defined for the Aramis project:

1. ***Industrial decarbonisation business-case stability enabling FID for capture investments***

   A sufficiently strong and durable economic and policy incentive environment exists and persists, within which industrial emitters take timely final investment decisions for $CO_2$ capture and contractually commit $CO_2$ delivery, allowing Aramis to rely on meaningful volumes as an open-access system.

2. ***Timely development and capacity of feeder and terminal infrastructure;***

   $CO_2$ supply channels - including onshore feeder pipelines and liquid $CO_2$ terminals and shipping - are available in time and are sufficiently dimensioned to support the intended throughput towards the Aramis collection hub.

3. ***Compression and interface readiness for trunkline injection;***

The $CO_2$ collection hub, consisting of the compressor station and terminal, is operationally ready and interface-compatible (in terms of pressure, temperature, $CO_2$ specification, and control systems) to enable stable $CO_2$ flow into the offshore trunkline.

4. ***Pipeline and distribution hub readiness;***

   The offshore trunkline and distribution hub are technically and operationally ready to continuously distribute $CO_2$ to multiple storage locations within defined design boundaries.

5. ***Storage capacity and operability readiness;***

   Sufficient offshore storage capacity with proven injectivity and operability is available in time, under valid permits and monitoring regimes, such that $CO_2$ throughput from Aramis is not structurally constrained.

6. ***System-wide strategic enablers and constraints;***

   The broader strategic environment - including policy and regulatory conditions, public and political legitimacy, macro-infrastructure development, and market organisation - remains sufficiently supportive to justify and sustain an open-access $CO_2$ transport system at a scale of up to 22 Mtpa.

These viability dependencies form the fundamental requirements of the Aramis project. The following subsections further elaborate on them through the enabling dependencies that support their realisation across the value chain.

### 7.3.1. Viability & enabling dependencies

Enabling dependencies describe the mechanisms at value-chain segment level through which the project's viability dependencies are operationalised. They express each viability condition in terms of segment-specific interfaces, arrangements, and decision structures, and provide the analytical link between the viability-level dependencies and the sub-dependencies identified in this study. Each enabling dependency is assigned a unique code of the form Cx.y, where C denotes an enabling dependency, x refers to the associated viability dependency, and y indicates the sequential number of the enabling dependency within that viability category. These enabling dependencies with their associated viability dependencies will now be elaborated.

It is important to note that all this following information regarding the dependencies was gathered at a specific point during the FEED phase based on interviews with Aramis colleagues. It therefore represents a snapshot and does not apply to the entire FEED phase of a project. During this phase of the project, this information is constantly changing and is therefore only a momentary reflection of the situation. Below, all six viability and all 27 enabling dependencies will be elaborated on.

**1. *Industrial decarbonisation business-case stability enabling FID for capture investments***

Value-chain anchor: Emitters

This viability dependency concerns the industrial parties that are potential users of the Aramis system. These are companies that need to capture $CO_2$, condition it and connect it to the transport network. For Aramis as an open-access infrastructure, it is essential that a sufficient number of these parties make timely investment decisions for capture installations and commit contractually to $CO_2$ supply. The associated enabling dependencies therefore describe the factors that determine whether, when and under what conditions emitters will reach a capture FID and are able and willing to commit to volumes.

Enabling Dependencies:

- ***C1.1 Cost-effectiveness of CCS versus alternatives;***
  Drivers which determe whether CCS remains more attractive than paying the EU-ETS price (of emitting $CO_2$) or pursuing alternative routing or abatement options.
- ***C1.2 $CO_2$ specification as access and cost criterion;***
  $CO_2$ quality requirements gate access to the system and directly affect capture costs; overly

restrictive or overly permissive specifications affect overall system viability.

- ***C1.3 Emitter FID decision complexity;***
  Capture investment decisions depend on timely access to data, permits, utilities, and cross-chain alignment; complexity increases decision friction and delays.
- ***C1.4 Total volume commitment reliability;***
  Stability of committed volumes and delivery timing; changes propagate into rework, tariff adjustments, and credibility effects across the chain.

2. ***Timely development and capacity of feeder and terminal infrastructure***

Value-chain anchor: Transport (feeders and shipping to Maasvlakte)

This viability dependency relates to the supply of $CO_2$ to the Maasvlakte. Even if emitters are willing to supply $CO_2$, Aramis can only function if the physical supply channels - pipelines, terminals and shipping - are available in good time and offer sufficient capacity. The enabling dependencies under this viability dependency illustrate how synchronisation, routing choices and flexibility in this supply structure determine the actual utilisation of the Aramis system.

Enabling Dependencies:

- ***C2.1 Infrastructure synchronisation across feeder assets;***
  Alignment of feeder (emitter to infrastructure) pipelines, terminals, and shipping windows with Aramis start-up and growth trajectories.
- ***C2.2 Routing flexibility and competition exposure;***
  Availability of alternative routes - particularly shipping to other destinations - can divert volumes and undermine tariff formation and long-term commitments.
- ***C2.3 Balancing and contingency arrangements;***
  Mechanisms to absorb outages through spare capacity, re-routing, or temporary takeover, preventing system-wide disruption.

3. ***Compression and interface readiness for $CO_2$ flow to the trunkline***

Value-chain anchor: $CO_2$ collection hub (terminal and compression)

This viability dependency focuses on the junction where various $CO_2$ streams converge: the collection hub on the Maasvlakte. Here, gaseous and liquid $CO_2$ streams must be compressed to dense-phase $CO_2$ configuration for transport via the offshore trunkline. The enabling dependencies describe the technical, operational and organisational conditions under which compression and interfaces function in such a way that a stable and safe flow is possible.

- ***C3.1 $CO_2$next system readiness and commitment;***
  The terminal must be delivered on time, be technically compatible, and secure availability and cost conditions that underpin chain operability.
- ***C3.2 Compression capacity, availability, and redundancy;***
  Compression availability is a precondition for FID; rules on redundancy, remedies, and attribution must be explicit.
- ***C3.3 Porthos system integration and readiness;***
  Porthos start-up, governance arrangements, and alignment on specification and availability determine integrated hub operability.
- ***C3.4 Design constraints from external system drivers;***
  Exogenous requirements (e.g. power availability, utility access, safety norms) shape design choices, deliverability, and approval evidence.

4. ***Pipeline and Distribution Hub readiness***

Value-chain anchor: Onshore & Offshore trunkline and offshore distribution hub

This viability dependency concerns the offshore transport and distribution system itself. The trunkline and distribution hub form the backbone of Aramis and determine whether $CO_2$ can actually be transported to multiple storage locations. The enabling dependencies under this category provide insight into which aspects of integrity, safety, execution and resilience are decisive for the continuous functioning of this part of the chain.

Enabling Dependencies:

- ***C4.1 Offshore transport integrity and operability;***
  Dense-phase transport integrity - including pressure/temperature envelopes, corrosion behaviour, and upset response - governs steady-state operability.
- ***C4.2 Safety-by-interface dependency;***
  Safety performance depends on partner interfaces, safeguards, trip logic, and safety-case evidence across connected assets.
- ***C4.3 Single-point-of-failure exposure;***
  Failure of the trunkline or distribution hub halts the chain; recovery time and repair capability determine commercial and operational impact.
- ***C4.4 Off-specification and upset handling procedures;***
  Clear decision logic for stop, vent, recovery, and control responses is required to manage off-spec events.
- ***C4.5 Execution capacity and schedule realism;***
  Contractor and vessel availability, together with realistic planning assumptions, determine pre-FID deliverability and schedule credibility.

**5. *Storage capacity and operability readiness***

Value-chain anchor: Storage (fields, platforms, and wells)

This viability dependency focuses on the availability and operational readiness of offshore storage locations. Aramis can only function if sufficient storage capacity with proven injectivity is available in a timely manner and can be deployed operationally. The enabling dependencies describe how the characteristics of storage fields, permits, operational flexibility and the willingness of storage operators to invest together determine whether storage can actually function as the final link in the chain.

Enabling Dependencies:

- ***C5.1 Storage capacity portfolio adequacy;***
  A sufficiently diversified and timely developed portfolio of fields is required to support throughput and ramp-up.
- ***C5.2 Injectivity and operational flexibility;***
  Injectivity and operational flexibility determine whether volumes can be injected safely without excessive buffering.
- ***C5.3 Storage availability and revenue exposure;***
  Injection downtime affects revenues and subsidies and creates pressure under send-or-pay arrangements across the chain.
- ***C5.4 $CO_2$ specification bounded by storage constraints;***
  Storage licence conditions bound acceptable $CO_2$ composition, propagating upstream into hub and capture design.
- ***C5.5 Storage investment commitment logic;***
  Storage operators commit capital only when risk-reward profiles are acceptable and FID alignment exists across the chain.

**6. *System-wide strategic enablers and constraints***

Value-chain anchor: Chain-wide (open-access system up to 22 Mtpa)

This viability dependency relates to preconditions that cannot be attributed to a single specific link in the chain but affect the Aramis system as a whole. These are chain-wide aspects such as tariff setting, integration mandate, synchronisation of investment decisions and legal finality. The enabling dependencies in this category describe how these overarching conditions enable or limit the coherence and scalability of an open-access $CO_2$ system up to 22 Mtpa.

Enabling Dependencies:

- ***C6.1 Tariff formation and cost discovery;***
  Tariffs emerge from market prices and chain costs and remain uncertain until cost estimates mature and volumes stabilise.
- ***C6.2 System integration governance;***
  The mandate of the system integrator and change-control arrangements determine safe and workable integration across parties.
- ***C6.3 FID synchronisation across the value chain;***
  Investment decisions are interdependent; synchronisation is critical to avoid stranded exposure prior to FID.
- ***C6.4 Chain-wide contractual coherence;***
  End-to-end contractual alignment (e.g. back-to-back terms, IOAs) limits residual exposure at interfaces.
- ***C6.5 Strategic optionality and system rigidity;***
  The degree of lock-in increases after commitment, reducing flexibility and making decoupling politically and economically sensitive.
- ***C6.6 Permitting finality and legal operability;***
  Legal finality and acceptance of approvals determine whether FID is both legally and practically feasible.

Together, the identified viability and enabling dependencies constitute the analytical basis of the Aramis dependency model. Each dependency has subsequently been subjected to a structured qualitative assessment based on expert judgement provided by professionals within the Aramis Joint Team. As part of this assessment, dependencies were assigned a primary and secondary placement within one of the eight MDF sub-contexts (A1-D2), indicating where uncertainty primarily originates and where it first becomes governance-relevant.

In addition, each dependency was evaluated in terms of autonomy, bandwidth, and feedback strength, resulting in a vulnerability characterisation consistent with the framework introduced in Chapter 5. The resulting context placements, vulnerability scores, and indicative propagation paths are documented in detail in Appendices B. All these are regarding Aramis dependencies information, devided in different sub-appendixes. Together with concise explanatory notes they justify the applied assessments. Here an overview of what can be found in Appendix B:

- Appendix B.1 shows a graphical abstract of the model characteristics.
- Appendis B.2 gives a total overview of all Aramis dependencies data.
  - Appendix B.2.1 shows the data of the viability dependencies.
  - Appendix B.2.2 shows the data of the enabling dependencies.
  - Appendix B.2.3 shows the data of all sub-dependencies.
- Appendix B.3 gives a list of all viability and associated enabling dependencies.
- Appendix B.4 elaborates on all viability dependencies characteristics.
- Appendix B.5 elaborates on all enabling dependencies characteristics.

### 7.3.2. Sub-dependencies

In total, 100 sub-dependencies (development indicators) are identified through the systematic interpretation of project documentation and expert input. Each sub-dependency is uniquely associated with a single enabling dependency, with no overlap between enabling categories. Table 7.1 presents an overview of these relationships (P-S, in the second colum stands for: Primary-Secondary placement of the sub-dependency in the MDF contexts). For each enabling dependency, the table indicates its associated propagation pattern, its vulnerability classification (with the outer ring representing higher vulnerability), the total number of linked sub-dependencies, and the distribution of their development status across red, orange, and green indicators. The scoring of all sub-dependencies was conducted by professionals with domain-specific expertise relevant to each sub-dependency and reflects their assessment of the current development status.

| Enabling | P - S | Ring | #Subs | Red | Orange | Green |
|---|---|---|---|---|---|---|
| C1.1 Cost-effectiveness of CCS versus alternatives | A2 - A1 | MIDDLE | 6 | 1 | 0 | 5 |
| C1.2 $CO_2$ specification as access and cost criterion | D1 - A2 | OUTER | 5 | 2 | 1 | 2 |
| C1.3 Emitter FID decision complexity | A2 - B1 | MIDDLE | 5 | 0 | 5 | 0 |
| C1.4 Total volume commitment reliability | D2 - C1 | OUTER | 4 | 0 | 1 | 3 |
| C2.1 Infrastructure synchronisation across feeder assets | C1 - D2 | INNER | 5 | 2 | 1 | 2 |
| C2.2 Routing flexibility and competition exposure | A2 - D2 | OUTER | 3 | 1 | 2 | 0 |
| C2.3 Balancing and contingency arrangements | B1 - D2 | MIDDLE | 3 | 0 | 2 | 1 |
| C3.1 $CO_2$Next system readiness & commitment | C1 - D2 | OUTER | 6 | 2 | 0 | 4 |
| C3.2 Compression capacity, availability & redundancy | D2 - B1 | OUTER | 4 | 0 | 3 | 1 |
| C3.3 Porthos system integration & readiness | C1 - D2 | OUTER | 5 | 0 | 4 | 1 |
| C3.4 Design constraints from external system drivers | A1 - C2 | OUTER | 4 | 1 | 2 | 1 |
| C4.1 Offshore transport integrity and operability | D1 - C1 | OUTER | 3 | 0 | 2 | 1 |
| C4.2 Safety-by-interface dependency | C1 - D2 | MIDDLE | 4 | 0 | 2 | 2 |
| C4.3 Single-point-of-failure exposure | D2 - B2 | OUTER | 2 | 0 | 0 | 2 |
| C4.4 Off-spec and upset handling procedures | C1 - D2 | OUTER | 3 | 0 | 3 | 0 |
| C4.5 Execution capacity & schedule realism | C1 - A2 | OUTER | 3 | 1 | 2 | 0 |
| C5.1 Storage capacity portfolio adequacy | D2 - C1 | OUTER | 4 | 1 | 0 | 3 |
| C5.2 Injectivity and operational flexibility | D2 - D1 | OUTER | 3 | 0 | 2 | 1 |
| C5.3 Store availability and revenue exposure | D2 - A1 | INNER | 2 | 0 | 2 | 0 |
| C5.4 $CO_2$ specification bounded by storage constraints | C2 - D1 | OUTER | 2 | 0 | 1 | 1 |
| C5.5 Stores investment commitment logic | A2 - D2 | MIDDLE | 4 | 0 | 4 | 0 |
| C6.1 Tariff formation and cost discovery | D1 - A2 | OUTER | 4 | 0 | 2 | 2 |
| C6.2 System integration governance | B1 - D1 | MIDDLE | 3 | 0 | 1 | 2 |
| C6.3 FID synchronisation across the value chain | C1 - B2 | OUTER | 2 | 1 | 1 | 0 |
| C6.4 Chain-wide contractual coherence | B1 - D2 | OUTER | 4 | 0 | 3 | 1 |
| C6.5 Strategic optionality and system rigidity | D1 - A1 | OUTER | 3 | 0 | 2 | 1 |
| C6.6 Permitting finality and legal operability | C2 - B2 | OUTER | 4 | 1 | 3 | 0 |

**Table 7.1:** Enabling Dependencies Details (P-S: Primary-Secondary)

## 7.4. MDF implementation

This paragraph examines a single, illustrative application of the MDF by focusing on one enabling dependency and its associated viability dependency and sub-dependencies. The example is selected because it captures the interaction between technical system design, market participation, and governance considerations that characterises the Aramis value chain.

The enabling dependency under consideration is C1.2 - $CO_2$ specification as an access and cost criterion, which forms part of the first viability dependency: Industrial decarbonisation business-case stability enabling FID for capture investments.

This viability dependency C1 has four enabling dependencies, which specifies the mechanism through which this requirement is realised, one of those is C1.2.

Enabling Dependency: *(C1.2) - $CO_2$ specification as an access and cost criterion.*

The sub-dependencies capture the development conditions that influence how this enabling mechanism evolves and how stable its underlying assumptions are. For C1.2 these are defined as:

- Purity variation by industrial source.
- Additional processing requirements driven by specification stringency.
- First-mover disadvantage under tight specification.

- Location of quality measurement and enforcement (upstream versus post-mixing).
- Specification-driven CAPEX/OPEX uncertainty at emitter level.

Elaboration: *C1.2 - $CO_2$ specification as an access and cost criterion*

In the Aramis offshore pipeline, $CO_2$ is transported in a dense phase (near-supercritical conditions), enabling high-capacity and energy-efficient transport. Under these pressure-temperature conditions, even small deviations in $CO_2$ composition - such as residual water, acid-forming impurities, oxygen, or trace components - can lead to severe corrosion mechanisms in carbon-steel pipelines. The combination of dense-phase $CO_2$, moisture, and impurities significantly increases the risk of corrosion and integrity degradation, not only during steady-state operation but also under transient and abnormal operating conditions such as start-up, shutdown, or upset events.

For this reason, $CO_2$ purity requirements - referred to as the $CO_2$ specification, or $CO_2$ spec. - in Aramis are designed to prevent the transported $CO_2$ from becoming corrosive across normal, transitional, and abnormal operating regimes. But higher purity, means higher cost for capturing and measuring. The specification envelope therefore also functions as a critical safeguard for pipeline integrity, compressor operation, and downstream storage acceptance. Importantly, these specifications are not defined in isolation of Aramis: they are developed with the aim of maximum alignment with existing and planned $CO_2$ transport infrastructures, in order to support interoperability, reduce system-wide rigidity, and avoid unnecessary specification divergence across the broader CCS network. As a result, $CO_2$ specification is not a purely technical parameter, but a chain-wide design and governance choice with direct financial, operational, and legitimacy implications.

Below there will be elaborated how the combination of this viability, enabling and sub-dependency will be processed into the MDF. Therefore, first of all, the context placement and vulnerability of the viability and enabling dependency.

Viability dependency:

***Industrial decarbonisation business-case stability enabling FID for capture investments***

Context placement Viability: A2 (primary) / D2 (secondary). The uncertainty associated with this viability dependency primarily concerns investment and commitment behaviour by industrial emitters (A2). When this uncertainty materialises, it becomes governance-relevant first through impacts on throughput, utilisation, and system performance (D2).

Vulnerability profile of this viability dependency: Autonomy = Low; Bandwidth = Low; Feedback strength = Low → Outer ring. C1 depends heavily on external economic and policy incentives (e.g. $CO_2$ pricing and subsidies) and on emitter willingness to pay. Small changes in these conditions can quickly affect FID decisions, while feedback to other dependencies remains limited. C1 is structurally vulnerable and therefore positioned in the outer vulnerability ring has four enabling dependencies and twenty sub-dependencies (Red: 3; Orange: 12; Green: 5).

Enabling dependency:

***(C1.2) - $CO_2$ specification as an access and cost criterion***

Context placement: D1 (primary) / A2 (secondary). This enabling dependency is primarily a system-concept question (D1) it determines whether the system can function and is a fundamental concept for the safe transport of $CO_2$. When the specification is tightened or remains unsettled, the uncertainty becomes governance-relevant in the market and industrial commitment context (A2), because stricter requirements may cause emitters to reconsider whether participation in the system is economically viable or attractive.

Vulnerability profile: Autonomy = Low; Bandwidth = Low; Feedback strength = High → Outer ring. C1.2 is weakly autonomous because it requires chain-wide agreement; it has a narrow tolerance because small specification changes can significantly affect feasibility; and it exhibits strong feedback, as specification choices propagate across capture design, compression, hub operation, offshore integrity, and storage acceptance. Five sub-dependencies are associated with C1.2 (Red: 2; Orange: 1; Green: 2).

Sub-dependencies:

The following sub-dependencies of C1.2 represent development conditions that influence how the vulnerability of the $CO_2$ specification dependency evolves over time.

***Purity variation by industrial source*** → A2, D1, Green

- A2 - Market & Industrial Commitment

  Because the variation originates from which actors participate in Aramis and which industrial sectors connect to the system, each with distinct capture processes and $CO_2$ compositions.
- D1 - System Concept Stability

  Because sector-specific differences in $CO_2$ composition constitute an inherent boundary condition for defining the acceptable system specification and system scope.

***Additional processing requirements driven by specification stringency*** → D1, A2, Orange

- D1 - System Concept Stability

  Because the strictness of the $CO_2$ specification is a fundamental system design choice that determines the level of conditioning and purification assumed across the chain.
- A2 - Market & Industrial Commitment

  Because stricter specifications directly affect capture costs and therefore influence emitters' willingness to invest and commit volumes.

***First-mover disadvantage under tight specification*** → A2, B2, Red

- A2 - Market& Industrial Commitment

  Because a tight specification increases costs and risks for early participants, reducing their willingness to commit as first movers.
- B2 - External Legitimacy & Political Durability

  Because uneven cost and risk allocation between early and later participants can raise fairness concerns and challenge the political defensibility of the open-access system.

***Location of quality measurement and enforcement (upstream vs post-mixing)*** → D1, C1, Green

- D1 - System Concept Stability

  Because the choice of where $CO_2$ quality is measured and enforced defines responsibility boundaries within the system concept.
- C1 - Multi-Project & Interface Complexity

  Because this choice becomes visible at interfaces between projects and assets during commissioning and operational coordination.

***Specification-driven CAPEX/OPEX uncertainty at emitter level*** → D1, A2, Red

- D1 - System Concept Stability

  Because uncertainty about the $CO_2$ specification affects assumptions regarding capture configuration and required conditioning.
- A2 - Market & Industrial Commitment

  Because this uncertainty is decisive for emitter investment decisions and their willingness to contractually commit $CO_2$ volumes.

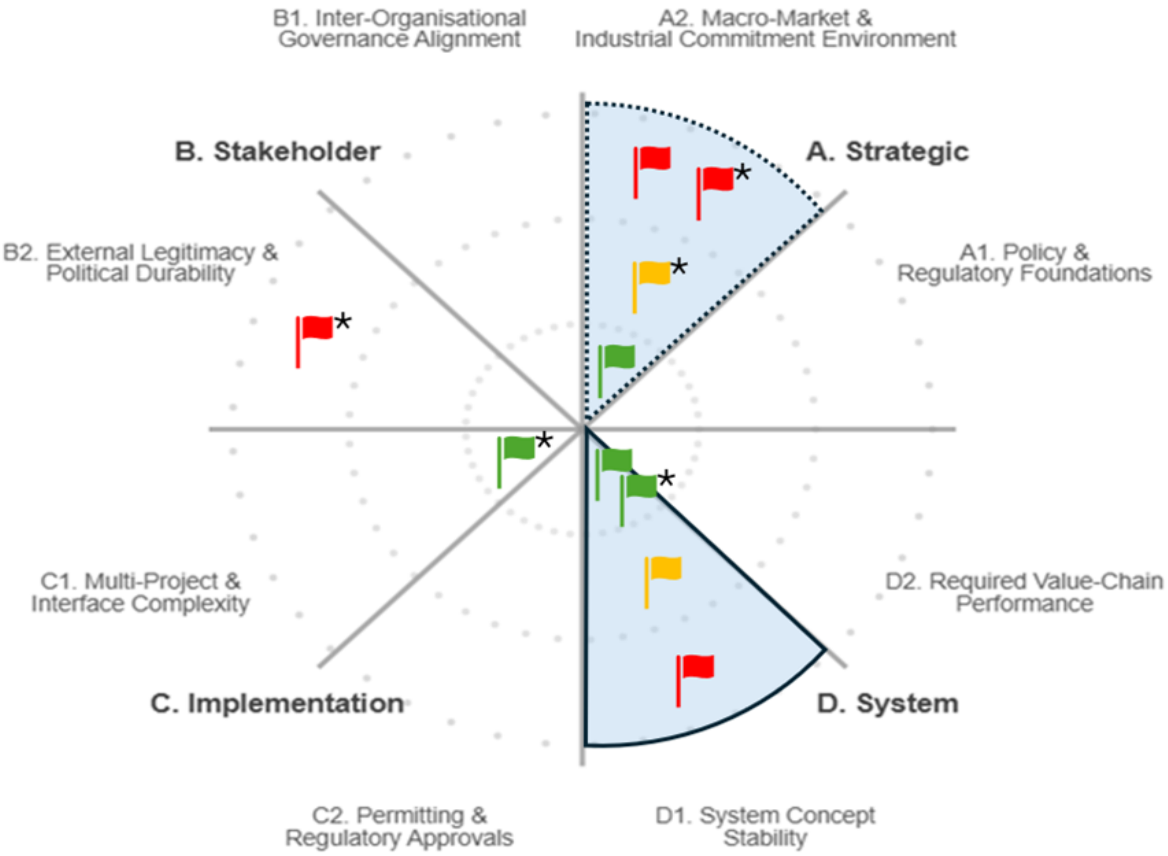


**Figure 4:** C1.2 $CO_2$ specification as an access and cost criterion

| C1.2 | P. | S. * | Status |
|---|---|---|---|
| **$CO_2$ specification as access and cost criterion** | **D1** | **A2 *** | **OUTER** |
| Purity variation by industrial source | A2 | D1 * | Green |
| Additional processing requirement driven by specification stringency | D1 | A2 * | Orange |
| First-mover disadvantage under tight specification | A2 | B2 * | Red |
| Location of quality measurement and enforcement (upstream vs post-mixing) | D1 | C1 * | Green |
| Spec-driven CAPEX/OPEX uncertainty at emitter level | D1 | A2 * | Red |

**Table 7.2:** Sub-dependencies (development indicators) associated with C1.2.

Figure 4 shows the profile of enabling dependency C1.2, including its vulnerability placement, its propagation pathways, and the associated sub-dependencies. Table 7.2 provides an overview of these sub-dependencies and their propagation patterns, indicated by P; Primary, and S*; Secondary placements, indicated with a star.

System Concept Stability (D1) and Market & Industrial Commitment (A2) are the dominant contexts shaping the vulnerability profile of this dependency. The sub-dependencies further show that uncertainty propagates primarily between System Concept Stability (D1) and Market & Industrial Commitment (A2), with additional spill-over into External Legitimacy (B2) and Interface Complexity (C1).

When analysed from the viability level C1.2 falls under: Industrial decarbonisation business-case stability enabling FID for capture investments - C1.2 stands out as particularly critical. This follows from the combination of its placement in the outer vulnerability ring and the presence of two red sub-dependencies, as shown in Table 7.3.

The enabling dependencies associated with this first viability dependency represent the emitter side of the value chain and are listed in Table 7.3. The table shows the contextual placement of each enabling dependency, its vulnerability status, and the distribution of underlying development conditions across red, orange, and green scores.

Taken together, this example illustrates how the MDF can be applied at both the enabling and viability levels to identify where uncertainty concentrates within the project system. It demonstrates how a focused dependency-based analysis can reveal which aspects of a viability requirement warrant closer monitoring, without reducing uncertainty to risk or prescriptive intervention.

| **Industrial decarbonisation business-case stability enabling FID for capture investments** | **P.** | **S.** | **Ring** | **#Subs** | **Red** | **Or.** | **Gr.** |
|---|---|---|---|---|---|---|---|
| C1.1 Cost-effectiveness of CCS versus alternatives | A2 | A1 | MIDDLE | 6 | 1 | 0 | 5 |
| C1.2 $CO_2$ specification as access and cost criterion | D1 | A2 | OUTER | 5 | 2 | 1 | 2 |
| C1.3 Emitter FID decision complexity | A2 | B1 | MIDDLE | 5 | 0 | 5 | 0 |
| C1.4 Total volume commitment reliability | D2 | C1 | OUTER | 4 | 0 | 1 | 3 |

**Table 7.3:** Viability Dependency with Associated Enabling Dependencies information

The following section extends this analysis to the full dependency landscape of the Aramis project.

## 7.5. Integrated analysis: critical and dominant pathways

The previous section examined enabling dependency C1.2 in detail. This example illustrates how instability in a single dependency can emerge when structural vulnerability coincides with unfavourable development conditions. The broader question, however, is what this means for the dependency structure of the Aramis project as a whole. In the total picture every viability dependency is assessed on its vulnerability, as well as the enabling dependencies, and the status of the sub-dependencies. This enables an integrated view of vulnerabilities and uncertainty propagations of all dependencies.

As seen in the previous example, both the dependency's Viability (C1) and Enabler (C1.2) are positioned in the outer ring. The enabler (C1.2.) carries two sub-dependencies that are assessed red. Together, this configuration indicates that relatively small disturbances in the external environment may affect the stability of this viability requirement. When such disturbances occur, they are most likely to affect the enabling mechanisms that are themselves structurally exposed. Within those mechanisms, the impact of such disturbances typically materialises in the development conditions that remain insufficiently consolidated, represented here by the red sub-dependencies.

This configuration forms what can be described as a critical dependency pathway. This is because of the combination of its viability, enabling and sub-dependency which is; outer times outer plus red:

- Viability = outer x;
- Enabler = outer +;
- Dependency = red;

This is called a Critical Dependency Pathway. This pathway defines the highest structural vulnerability, when the individually assessed dependencies are added together.

In such a - system configuration - pathway, exposure originates at the level of the viability requirement, passes through an enabling mechanism that carries structural vulnerability, and ultimately manifests in development conditions that remain unstable. Structural vulnerability therefore indicates where external developments are most likely to interact with the dependency structure of the project.

While structural vulnerability indicates where uncertainty can enter the system, a second system property determines how consequential such disturbances may become: the dominance of propagation paths. Each sub-dependency carries a propagation pathway between governance contexts. When many sub-dependencies follow the same pathway, these connections form dominant propagation pathways. Propagation pathsways are the primary and secondary contexts placements in the MDF.

The more frequently such pathways occur, the more strongly uncertainty is able to propagate through the dependency structure and affect multiple parts of the project system.

Thus, structural vulnerability is the overall view of all dependencies their vulnerability and development status added together; defined by critical dependency pathways. Uncertainty propagation is the dominance of occurrence of the MDF governance contexts.

Structural vulnerability and uncertainty propagation therefore capture two different aspects of system exposure. Structural vulnerability indicates where disturbances are most likely to affect the project, while dominant propagation patterns indicate the extent to which such disturbances may amplify and affect the wider system by propagating uncertainty. When structurally vulnerable dependency pathways coincide with dominant propagation routes, dependency instability are created.

To interpret this interaction, the analysis in this section positions both dimensions in a two-dimensional matrix, see figure 5. The vertical axis represents structural vulnerability, while the horizontal axis reflects the dominance of uncertainty propagation pathways. Together, these dimensions provide a system-level interpretation of where dependency instability is most likely to arise within the Aramis project.

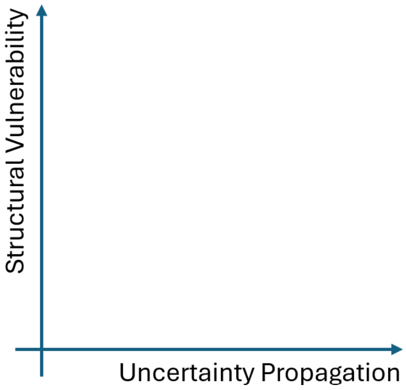


**Figure 5:** Dependency Instability

Sub-sections 7.5.1 and 7.5.2 will provide more detail on these two axes. Section 7.5.3 will revisit the overall implications of the dependency instability matrix.

### 7.5.1. Structural vulnerability and critical dependency pathways

A pathway is considered critical when a viability dependency and its enabling mechanism are both positioned in the outer vulnerability ring and at least one underlying development condition remains red. In the analysis this configuration is referred to as outer × outer + red.

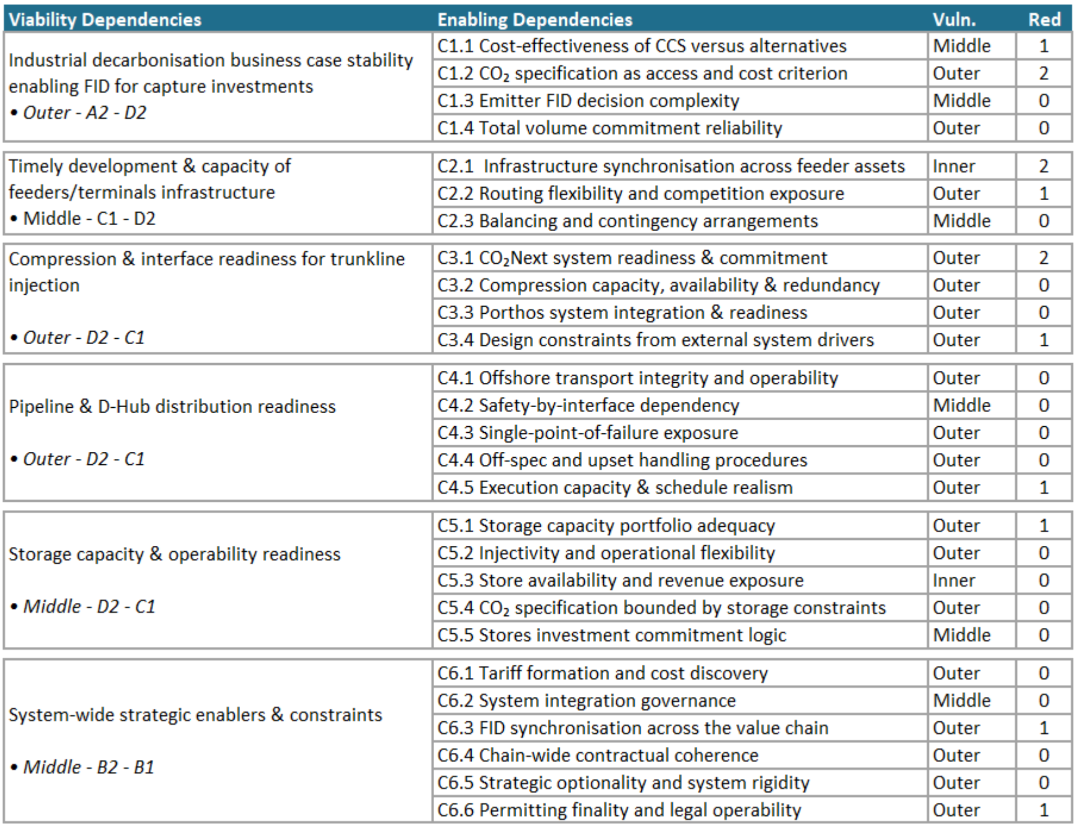
Configurations of this type represent locations where external disturbances are most likely to affect the stability of the project system. When both the viability requirement and the enabling mechanism are structurally exposed, external developments can interact directly with development conditions that remain insufficiently stabilised. By contrast, configurations such as inner × inner combined with green development conditions are far less likely to translate external shocks into system instability.

From a project governance perspective, this distinction has direct implications for Final Investment Decision (FID) readiness. Before an FID can be considered structurally robust, dependencies should be sufficiently stabilised to absorb disturbances in the external environment. Configurations of outer × outer + red therefore signal locations where the current dependency structure remains vulnerable.

Applying these criteria to the Aramis dependency landscape reveals four critical dependency pathways: C1.2, C3.1, C3.4 and C4.5 (see table 7.4). In each of these cases, an enabling dependency positioned in the outer vulnerability ring coincides with red development conditions in its sub-dependencies, and is beneath a viability dependency that is structurally exposed itself, as are C1, C3 and C4. C2, C5 and C6 have a ’middle’ vulnerability score, as can be seen in the first column of Table 7.4. The second column shows the associated enabling dependencies with the viability dependencies, the third column the vulnerability of the enabling dependency, and the forth column: ’Red’, the associated red development conditons underneath this enabling dependency.

A second group of dependencies also combines outer enabling vulnerability with red development conditions but is associated with viability dependencies positioned in the middle ring. These include C2.2, C5.1, C6.3 and C6.6. While these pathways remain structurally exposed, their overall vulnerability is lower because the viability requirement itself is less exposed.

Taken together, these pathways identify where external disturbances are most likely to interact with the dependency structure of the project. The locations where this structural exposure currently manifests itself become visible through the sub-dependencies, which capture the development conditions associated with each enabling dependency.



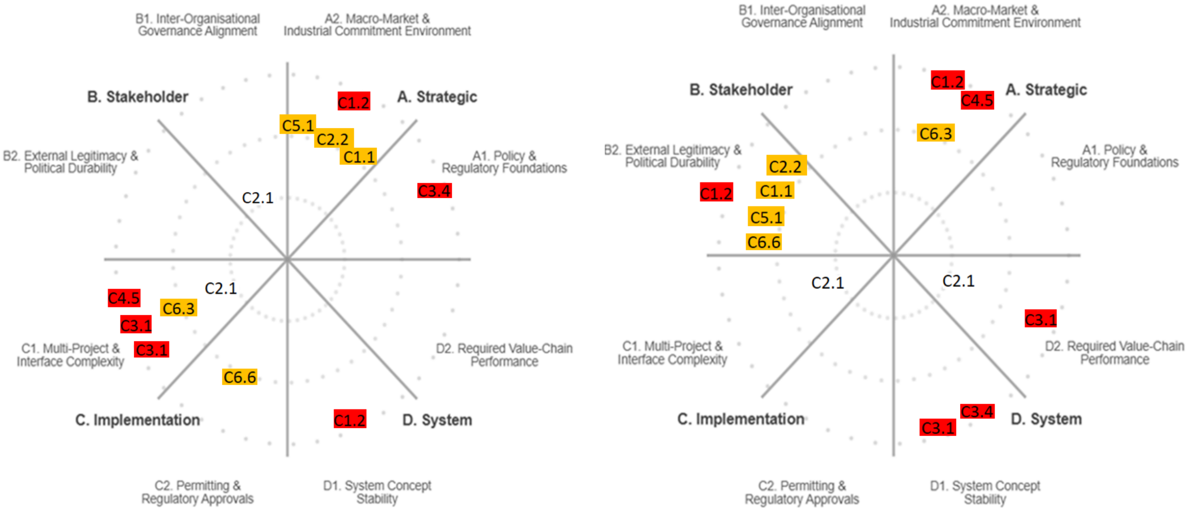
| Viability Dependencies | Enabling Dependencies | Vuln. | Red |
|---|---|---|---|
| Industrial decarbonisation business case stability enabling FID for capture investments<br>• *Outer - A2 - D2* | C1.1 Cost-effectiveness of CCS versus alternatives | Middle | 1 |
| | C1.2 $CO_2$ specification as access and cost criterion | Outer | 2 |
| | C1.3 Emitter FID decision complexity | Middle | 0 |
| | C1.4 Total volume commitment reliability | Outer | 0 |
| Timely development & capacity of feeders/terminals infrastructure<br>• Middle - C1 - D2 | C2.1 Infrastructure synchronisation across feeder assets | Inner | 2 |
| | C2.2 Routing flexibility and competition exposure | Outer | 1 |
| | C2.3 Balancing and contingency arrangements | Middle | 0 |
| Compression & interface readiness for trunkline injection<br>• *Outer - D2 - C1* | C3.1 $CO_2$Next system readiness & commitment | Outer | 2 |
| | C3.2 Compression capacity, availability & redundancy | Outer | 0 |
| | C3.3 Porthos system integration & readiness | Outer | 0 |
| | C3.4 Design constraints from external system drivers | Outer | 1 |
| Pipeline & D-Hub distribution readiness<br>• *Outer - D2 - C1* | C4.1 Offshore transport integrity and operability | Outer | 0 |
| | C4.2 Safety-by-interface dependency | Middle | 0 |
| | C4.3 Single-point-of-failure exposure | Outer | 0 |
| | C4.4 Off-spec and upset handling procedures | Outer | 0 |
| | C4.5 Execution capacity & schedule realism | Outer | 1 |
| Storage capacity & operability readiness<br>• *Middle - D2 - C1* | C5.1 Storage capacity portfolio adequacy | Outer | 1 |
| | C5.2 Injectivity and operational flexibility | Outer | 0 |
| | C5.3 Store availability and revenue exposure | Inner | 0 |
| | C5.4 $CO_2$ specification bounded by storage constraints | Outer | 0 |
| | C5.5 Stores investment commitment logic | Middle | 0 |
| System-wide strategic enablers & constraints<br>• *Middle - B2 - B1* | C6.1 Tariff formation and cost discovery | Outer | 0 |
| | C6.2 System integration governance | Middle | 0 |
| | C6.3 FID synchronisation across the value chain | Outer | 1 |
| | C6.4 Chain-wide contractual coherence | Outer | 0 |
| | C6.5 Strategic optionality and system rigidity | Outer | 0 |
| | C6.6 Permitting finality and legal operability | Outer | 1 |

**Table 7.4:** Total of Viability and Enabling dependencies and red sub-dependencies, to see critical dependency pathways

Across the Aramis dependency landscape, thirteen sub-dependencies were assessed as red by members of the Aramis project team. Each of these development conditions occurs along a propagation pathway between governance contexts within the MDF. Figures 6a and 6b project these red sub-dependencies onto the MDF context structure using their primary and secondary context placement. In this way, the figures visualise where destabilising development conditions occur within the governance landscape of the Aramis project.

The radial position of each label reflects the structural vulnerability of the dependency pathway in which the sub-dependency occurs. This position represents a weighted combination of the vulnerability of the viability dependency and that of the enabling dependency. Sub-dependencies associated with outer × outer + red configurations therefore appear on the outer boundary of the diagram, while combinations such as outer × inner + red appear closer to the centre.

For example, the two red sub-dependencies beneath C1.2 appear at the outer edge of the diagram.



**(a)** Primary context placement **(b)** Secondary context placement

**Figure 6:** Projection of red sub-dependencies onto the MDF governance contexts showing their primary and secondary placements.

This reflects the combination of an outer viability dependency and an outer enabling dependency. By contrast, the two red sub-dependencies associated with C2.1 appear closer to the inner circle. In this case the enabling dependency carries outer vulnerability, but the associated viability dependency is positioned in the middle ring, resulting in a lower overall pathway vulnerability.

The thirteen red sub-dependencies and their associated propagation pathways can be summarised as follows:

- A2-B2, 4 occurrences
- C1-D2, 2 occurrences
- C1-A2, 2 occurrences
- D1-A2, B1-C1, A1-D1, C1-D1, C2-B2 → 1 occurrences each

The propagation patterns show that instability in Aramis does not spread randomly, but repeatedly follows a limited number of cross-context dependency paths.

(seen Table 7.5)

The A2 → B2 pathway illustrates how fluctuations in market and industrial commitment directly translate into pressures on external legitimacy. These are not isolated issues. The underlying development conditions show that instability in industrial participation - such as uncertain volumes or delayed commitments - does not remain confined to the market domain. It directly affects how the project is perceived and supported by external stakeholders, thereby influencing its political and institutional legitimacy.

**A2 Macro-Market & Industrial Commitment Environment – B2 External Legitimacy & Political Durability**

| Enabling | Sub-dependency with red development status |
|---|---|
| C1.1 | Industrial relocation or cross-border storage |
| C1.2 | First-mover disadvantage under tight specification |
| C2.2 | Volume leakage through arbitrage to non-Aramis routes |
| C5.1 | Opportunity funnel potential prospects |

**Table 7.5:** Propagation path A2 → B2 (Market commitment affecting external legitimacy).

A second recurring mechanism is visible in the C1 → D2 pathway (see Table 7.6), where interface complexity translates into required value-chain performance. The associated development conditions indicate that performance risk is not primarily driven by technical failure, but by misalignment between independently governed system components. This shifts the interpretation of performance exposure from execution uncertainty to coordination dependency.

**C1 Multi-project & Interface Complexity – D2 Required Value-Chain Performance**

| Enabling | Sub-dependency with red development status |
|---|---|
| C2.1 | Feeder shipping readiness alignment |
| C3.1 | $CO_2$Next commissioning alignment with Aramis start-up |

**Table 7.6:** Propagation path C1 → D2 (Interface complexity affecting value-chain performance).

Similarly, the C1 → A2 pathway shows that interface complexity feeds back into market commitment. Here, structural constraints—such as contractor scarcity and interdependent investment timing—affect the willingness and ability of actors to commit. This demonstrates that market commitment is not only an economic decision, but also conditioned by the feasibility of coordinated execution across the value chain.(see Table 7.7).

**C1 Multi-project & Interface Complexity – A2 Macro-Market & Industrial Commitment Environment**

| Enabling | Sub-dependency with red development status |
|---|---|
| C4.5 | Contractor scarcity & schedule gate |
| C6.3 | Mutual dependence of investment commitments |

**Table 7.7:** Propagation path C1 → A2 (Interface complexity affecting market commitment).

The single-occurrence propagation paths listed in Table 7.8 further reinforce this pattern. Although individually less frequent, they consistently connect system design, policy conditions, and implementation sequencing across domains. Their distribution indicates that design decisions, regulatory structures, and operational alignment are tightly coupled, even when not part of dominant propagation pathways.

**Selected single-occurrence propagation paths across contexts**

| Enabling | Sub-dependency with red development status | Propagation |
|---|---|---|
| C1.2 | Spec-driven CAPEX/OPEX uncertainty at emitter level | D1 → A2 |
| C3.1 | $CO_2$Next interface compatibility (pressure, quality, flow envelope) | C1 → D1 |
| C3.4 | Micro-tunnel removal obligations | A1 → D1 |
| C2.1 | Sequencing uncertainty between parallel feeder developments | B1 → C1 |
| C6.6 | Legal finality versus reversibility | C2 → B2 |

**Table 7.8:** Single-occurrence propagation paths across contexts.

Taken together, these propagation directions demonstrate that destabilising development conditions are not confined to a single governance domain. It is precisely through these interdependencies that uncertainty can propagate across the project architecture, linking market behaviour, institutional conditions, system design, and execution performance into a single, structurally coupled system.

### 7.5.2. Dominant propagation pathways

While the previous subsection identified where structural vulnerability exists in the dependency structure, the propagation analysis examines how uncertainty travels through the system once such vulnerability is activated.

Table 7.9 summarises all propagation pathways observed across the sub-dependency dataset. For each primary-secondary context combination the table shows the total number of occurrences as well as the distribution of red, orange and green development conditions. The final column identifies the enabling dependencies associated with red development conditions.

Across the dataset, thirty-six possible propagation paths occur. However, the distribution of occurrences shows that uncertainty propagation is not evenly distributed across the dependency network. Instead, a limited number of pathways account for a disproportionate share of the observed interactions.

The most dominant pathway is C1 → D2, which appears twelve times. Two red development conditions occur along this path: $CO_2$Next commissioning alignment with Aramis start-up (C3.1) and feeder shipping readiness alignment (C2.1). Both relate to the synchronisation of independently governed infrastructure assets.

Substantively, this pathway connects multi-project interface complexity (C1) with required value-chain performance (D2). Governance relevance therefore emerges where interface alignment determines whether performance assumptions can hold. The presence of red development conditions beneath C3.1 and C2.1 indicates that this interface alignment remains insufficiently consolidated. The example also illustrates how uncertainty can reinforce itself across the system. If $CO_2$Next infrastructure is not yet operationally mature, potential emitters may delay investments in their own feeder infrastructure. This mutual dependency strengthens uncertainty across the value chain.

A second highly visible propagation pathway is A2 → B2, which appears five times and contains the highest concentration of red development conditions. Four destabilising development conditions occur along this path: industrial relocation or cross-border storage (C1.1), first-mover disadvantage under tight specification (C1.2), volume leakage through arbitrage to alternative transport routes (C2.2), and uncertainty regarding opportunity funnel prospects for storage capacity (C5.1).

Together these development conditions reveal how uncertainty in market and industrial commitment propagates into the domain of external legitimacy and political durability. If emitters delay participation, explore alternative storage routes, or face unclear commercial conditions, the political and societal defensibility of the CCS system may also weaken. In this way, market hesitation and legitimacy pressure reinforce each other within the governance landscape.

An important observation from Table 7.9 is that the enabling dependencies associated with red development conditions occur primarily within these dominant propagation paths. Of the thirty-six possible context combinations observed in the dataset, the critical development conditions cluster within the most frequently occurring pathways.

This clustering effect provides an important analytical insight. Dominant propagation pathways indicate where uncertainty is most structurally concentrated within the dependency network. When development conditions remain unresolved along these dominant paths, uncertainty can reinforce itself across interconnected governance domains.

From a systems perspective, the frequency of propagation therefore reflects the systemic magnitude of uncertainty within the project structure. While structural vulnerability indicates where instability may enter the system, dominant propagation pathways indicate how extensively that instability can travel once activated.

Within the analytical framework developed in this research, these propagation pathways therefore represent the uncertainty dimension of the dependency landscape. Combined with structural vulnerability, they form the basis for analysing dependency instability across the project system.

| Path | Primary | Secondary | Total | Red | Orange | Green | Critical enabling dependencies |
|---|---|---|---|---|---|---|---|
| 1 | C1 | D2 | 12 | 2 | 4 | 6 | C2.1, C3.1 |
| 2 | B1 | D2 | 10 | 0 | 5 | 5 | |
| 3 | A2 | B2 | 5 | 4 | 1 | 0 | C1.1, C1.2, C2.2, C5.1 |
| 4 | D1 | A2 | 5 | 1 | 3 | 1 | C1.2 |
| 5 | A1 | A2 | 4 | 0 | 4 | 0 | |
| 6 | A2 | D2 | 4 | 0 | 2 | 2 | |
| 7 | B1 | C1 | 4 | 1 | 2 | 1 | C2.1 |
| 8 | C1 | A2 | 4 | 2 | 2 | 0 | C4.5, C6.3 |
| 9 | C1 | B1 | 4 | 0 | 3 | 1 | |
| 10 | D1 | C1 | 4 | 0 | 2 | 2 | |
| 11 | D2 | A2 | 4 | 0 | 2 | 2 | |
| 12 | D2 | C1 | 4 | 0 | 3 | 1 | |
| 13 | A1 | D1 | 3 | 1 | 1 | 1 | C3.4 |
| 14 | B1 | B2 | 3 | 0 | 2 | 1 | |
| 15 | C1 | D1 | 3 | 1 | 1 | 1 | C3.1 |
| 16 | C2 | B2 | 3 | 1 | 2 | 0 | C6.6 |
| 17 | A1 | C1 | 2 | 0 | 2 | 0 | |
| 18 | B1 | D1 | 2 | 0 | 0 | 2 | |
| 19 | B2 | C1 | 2 | 0 | 2 | 0 | |
| 20 | D2 | D1 | 2 | 0 | 1 | 1 | |
| 21 | A1 | C2 | 1 | 0 | 1 | 0 | |
| 22 | A2 | A1 | 1 | 0 | 1 | 0 | |
| 23 | A2 | B1 | 1 | 0 | 1 | 0 | |
| 24 | A2 | C1 | 1 | 0 | 0 | 1 | |
| 25 | A2 | D1 | 1 | 0 | 0 | 1 | |
| 26 | B2 | A1 | 1 | 0 | 0 | 1 | |
| 27 | B2 | A2 | 1 | 0 | 1 | 0 | |
| 28 | B2 | B1 | 1 | 0 | 0 | 1 | |
| 29 | B2 | C2 | 1 | 0 | 1 | 0 | |
| 30 | B2 | D1 | 1 | 0 | 1 | 0 | |
| 31 | C1 | B2 | 1 | 0 | 1 | 0 | |
| 32 | C2 | A2 | 1 | 0 | 1 | 0 | |
| 33 | C2 | B1 | 1 | 0 | 1 | 0 | |
| 34 | D1 | B1 | 1 | 0 | 1 | 0 | |
| 35 | D1 | D2 | 1 | 0 | 1 | 0 | |
| 36 | D2 | A1 | 1 | 0 | 1 | 0 | |

**Table 7.9:** Observed propagation paths sub-dependencies and associated status distributions.

### 7.5.3. Dependency instability

Figure 7 combines the two analytical dimensions developed in the previous subsections into a single representation. The figure plots the thirteen red sub-dependencies identified in the Aramis dependency landscape. Each point therefore represents a development condition that remains insufficiently stabilised within the project structure.

The vertical axis represents the structural vulnerability of the dependency pathway in which the development condition occurs. This vulnerability is derived from the combined positioning of the viability dependency and the enabling dependency within the dependency matrix. Pathways in which both dependencies are positioned in the outer vulnerability ring and carry a red development condition are placed in the upper part of the figure. Conversely, pathways with lower structural exposure appear closer to the lower boundary. The vertical positioning should therefore be interpreted as an ordinal indication of relative vulnerability rather than a precise quantitative measure.

The horizontal axis represents uncertainty propagation within the dependency network. This dimension is approximated using the frequency with which specific propagation paths occur in the sub-dependency analysis. When a propagation pathway appears multiple times, this indicates that uncertainty is not isolated but repeatedly manifests along the same structural connection. Dependencies associated with frequently recurring propagation paths are therefore positioned further to the right. As with the vertical axis, this positioning reflects relative differences rather than exact numerical distances.

The placement of each point within one of the four quadrants is thus determined by a categorical classification of these two dimensions: high versus low structural vulnerability, and high versus low uncertainty propagation. The exact position of a point within a quadrant does not carry additional analytical meaning beyond this classification. The figure should therefore be interpreted as a structuring device that highlights relative patterns of concentration, rather than as a precise spatial measurement.

The position of each point in the matrix therefore reflects two characteristics simultaneously: the structural vulnerability of the dependency within the project system, and the systemic uncertainty associated with the propagation path in which the development condition occurs. Together, these dimensions define what is referred to in this study as dependency instability: the interaction between vulnerability and uncertainty within the dependency architecture.

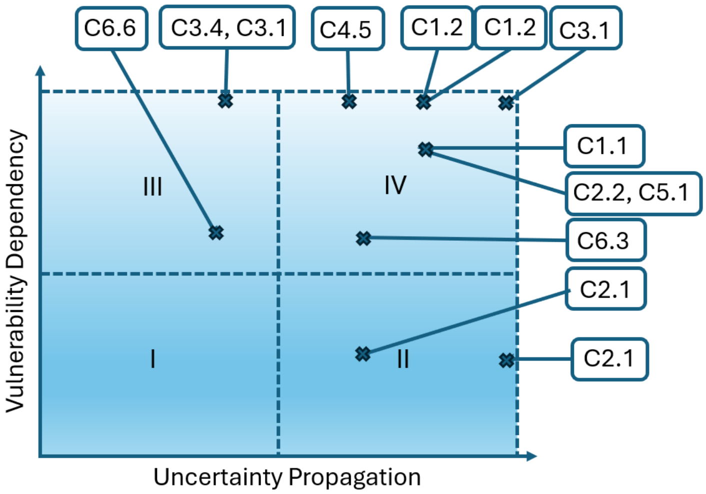


**Figure 7:** Dependency Instability

This becomes visible, for example, in the case of C3.1 - $CO_2$Next system readiness and commitment. The associated development conditions appear in the upper-right corner of the diagram. This reflects two characteristics simultaneously. First, the dependency pathway itself is highly vulnerable, as it combines an outer viability dependency with an outer enabling dependency and red development conditions. Second, the propagation path C1 → D2 is the most dominant pathway observed in the dataset. As a result, uncertainty affecting this dependency is both structurally exposed and embedded in one of the most active propagation channels within the project system.

In practical terms this means that unresolved development conditions surrounding $CO_2$Next readiness sit at a structural coordination point of the CCS system. $CO_2$Next forms a central interface between multiple independently governed infrastructure elements within the value chain and therefore plays a significant role in determining whether the Aramis system can operate as intended. At the same time, the $CO_2$Next project itself remains in a relatively uncertain phase and is vulnerable to changes in its external environment.

If readiness at this interface remains uncertain, the alignment between upstream feeder infrastructure and downstream transport and storage capacity becomes difficult to stabilise. The dependency therefore acts as a systemic pressure point: uncertainty surrounding $CO_2$Next readiness does not remain local but can influence multiple coordination relationships across the value chain.

The figure therefore visualises how structural vulnerability and uncertainty propagation jointly shape the instability of the dependency landscape.

### 7.5.4. Interpreting dependency instability for governance

The combination of structural vulnerability and uncertainty propagation divides the matrix into four analytical zones, each representing a different configuration of dependency instability within the project system (Figure 7).

Zone I represents low vulnerability and limited uncertainty propagation. Development conditions in this area occur within relatively stable dependencies and along propagation paths that appear infrequently. Disturbances affecting these conditions are therefore unlikely to destabilise the broader system.

Zone II represents low vulnerability but high uncertainty propagation. Development conditions occur along dominant propagation pathways, meaning that uncertainty may travel through multiple governance contexts. However, because the underlying dependency itself is not structurally exposed, the likelihood that external disturbances activate this propagation remains limited.

Zone III represents high vulnerability but limited propagation. In this case the dependency itself is exposed to disturbances, but the propagation potential of uncertainty remains relatively contained within the system.

Zone IV combines both high vulnerability and strong uncertainty propagation. Development conditions located in this zone are embedded in exposed dependencies and dominant propagation pathways. Disturbances affecting these conditions are therefore both more likely to occur and more likely to spread across the dependency network.

Within the matrix, several development conditions appear in Zone IV, where high structural vulnerability coincides with dominant uncertainty propagation. These include the most significant development conditions associated with C3.1 - $CO_2$Next system readiness and commitment, C1.2 - $CO_2$ specification as an access and cost criterion, C2.2 - routing flexibility and competition exposure, and C5.1 - storage capacity portfolio adequacy.

A common characteristic of these conditions is that they concern parts of the system that are still evolving and where key design, market, or coordination arrangements are not yet fully established. In practice, such conditions resemble first-of-a-kind (FOAK) situations in which actors operate in environments characterised by significant ontological and epistemic uncertainty. Under these circumstances the governance challenge is not only to manage identifiable risks, but to deal with uncertainty that is still unfolding and only partially understood.

The matrix makes visible where such conditions occur within the dependency structure and how they relate to dominant propagation pathways. From a governance perspective, this indicates that such conditions require stabilisation before conventional risk management becomes effective. Stabilisation can occur in two ways. Structural vulnerability may decrease as enabling mechanisms mature and become less exposed to external disturbances. Alternatively, uncertainty propagation may decrease as development conditions consolidate and become less embedded within dominant propagation pathways. In both cases the position of the development condition shifts away from Zone IV toward more stable regions of the matrix.

The dependency instability matrix 7, together with the context projection shown in Figure 6, provides a structured way to understand both where uncertainty may enter the project system and how it may propagate across governance contexts. The two figures therefore serve complementary analytical purposes.

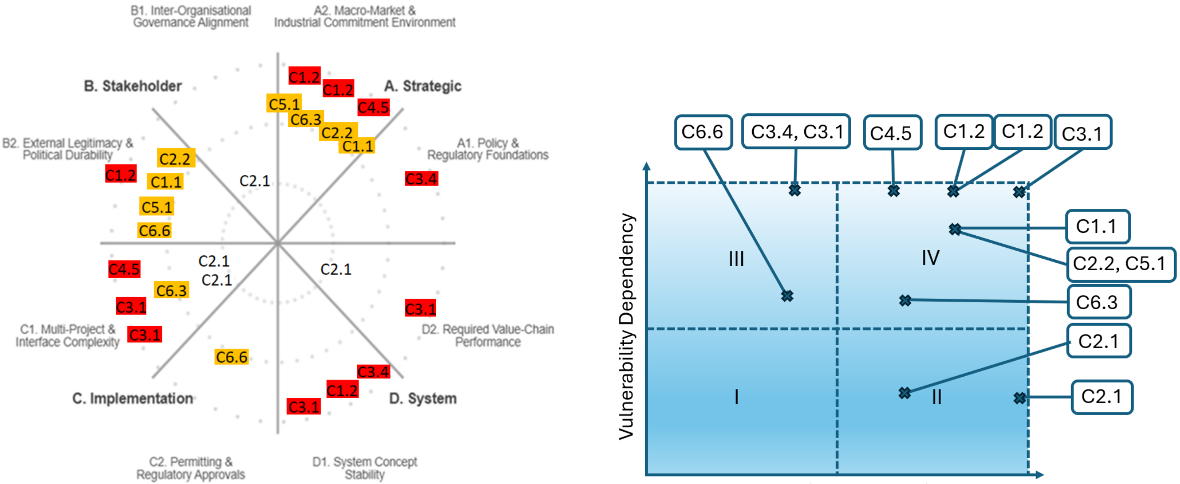
The matrix identifies the structural exposure of the system by combining vulnerability and uncertainty propagation. Vulnerability indicates how exposed a dependency is to disturbances in the external environment, while uncertainty propagation reflects how strongly uncertainty may spread through the dependency network once such disturbances occur.

Figure 7.6 complements this perspective by projecting the development conditions onto the governance contexts of the MDF. This projection reveals where uncertainty manifests itself within the governance structure of the project and which contexts are most directly affected when destabilising development conditions occur.

Taken together, these two representations provide a structured lens for interpreting project uncertainty. The matrix highlights which dependencies require stabilisation, while the context projection clarifies where the consequences of that instability are most likely to materialise within the governance landscape.

In this sense, the dependency instability matrix can be interpreted as a precursor to conventional risk analysis. Structural vulnerability resembles the notion of probability, while uncertainty propagation approximates the systemic impact that disturbances may generate. The MDF context projection then reveals where those impacts are likely to unfold within the governance system of the project. The framework therefore does not replace risk analysis, but precedes it by clarifying the structural conditions under which risk mitigation becomes meaningful.

By combining these perspectives, the framework provides a structured basis for decision-making at critical governance moments such as Final Investment Decision (FID) or stage-gate transitions. Rather than attempting to reduce uncertainty directly to risk registers, the framework first clarifies which dependencies must stabilise and where instability would manifest if they do not. In doing so, it offers decision-makers an analytical lens that reduces the complexity of the dependency landscape while making visible where uncertainty concentrates and how it may influence the viability of the project.



**(a)** Reproduction of Figure 6: Projection primary and secondary placement of red sub-dependencies onto the MDF

**(b)** Reproduction of Table 7: Dependency Instability

**Figure 8:** Combined view of dependency instability and sub-dependency projection.

Taken together, Figure 8 clarify how and where dependency instability becomes relevant within the project system. Figure 8a, shows in which governance contexts uncertainty is most likely to take hold, by projecting the development conditions onto the MDF and thereby revealing where structurally exposed dependencies are located. Figure 8b provides a complementary perspective by positioning these same conditions along the two underlying dimensions of dependency instability: structural vulnerability and uncertainty propagation.

Combined, the figures make it possible to distinguish not only where uncertainty may manifest, but also how it behaves once it enters the system. They show which dependencies are most exposed to external disturbance and to what extent uncertainty can propagate through them. It shows the location of exposure and its potential impact on the project system.

From a governance perspective, this implies that the central task is not to interpret uncertainty only in terms of discrete risks, but to understand which dependencies combine high vulnerability with strong propagation potential. These dependencies determine where disturbances are most likely to become consequential and therefore require stabilisation before they manifest as risks.

## 7.6. A risk oriented governance view

Chapters 4 and 5 established that viability and enabling dependencies carry structural vulnerability. These dependencies are exposed to developments in external systems beyond the direct control of the project and therefore represent the structural conditions on which project viability depends.

Sub-dependencies, by contrast, function as epistemic development indicators. They do not alter the underlying vulnerability of the dependency itself but indicate how the assumptions surrounding enabling mechanisms are evolving under changing external conditions.

Section 7.5 combined these two perspectives by analysing both structural vulnerability and uncertainty propagation within the dependency network. The resulting dependency instability matrix revealed where exposed dependencies coincide with dominant propagation pathways and where development conditions remain insufficiently stabilised. These configurations represent locations where uncertainty may both enter the project system and spread across governance contexts, which was demonstrated in the MDF.

Within this framework, risks are not treated as independent objects of management but as downstream manifestations of dependency instability. Disturbances affect the project system where dependencies are structurally exposed, while dominant propagation pathways determine how uncertainty may spread once such disturbances occur.

In this sense, risks do not primarily originate from isolated events but from the interaction between critical dependency pathways and dominant propagation pathways. Critical dependencies, defined by the configuration outer × outer + red, indicate where uncertainty is most likely to affect the project system. Dominant propagation pathways indicate how extensively that uncertainty may travel through the dependency network. Together, they provide a structural indication of where both the likelihood and the systemic impact of instability are concentrated.

The dependency instability matrix can therefore be interpreted as a precursor to conventional risk analysis, particularly in phases where the project has not yet reached FID and where key parts of the system still contain first-of-a-kind characteristics. In such situations, many relevant uncertainties have not yet consolidated into clearly articulable risks. This is especially the case where design choices, coordination arrangements, and market commitments are still evolving under conditions of limited precedent. Under these circumstances, risk articulation can be expected to remain difficult, precisely because uncertainty is still embedded in the dependency structure rather than already visible as discrete risk events.

The analysis in this section therefore examines whether the top risks identified in Aramis can indeed be understood as later expressions of the critical dependencies and dominant propagation paths identified earlier. If so, this would confirm that the dependency instability matrix offers a more structurally accurate lens for the pre-FID phase than a conventional risk-based view alone.

### 7.6.1. Reclassification of project risks under conditions of uncertainty

To connect the dependency-based analysis with the existing governance practice of the Aramis project, the next step examines the set of risks that are actively monitored within the project's risk register. At the time of analysis, sixteen items were identified by the project organisation as “top risks”.

Rather than treating these items as risks in the analytical sense, the first step examines the underlying issues they describe. The dependency-based framework developed in the previous sections distinguishes between discrete events and situations where the registered “risk” reflects uncertainty surrounding the stability of external conditions or enabling mechanisms on which the project depends.

From the perspective of this framework, entries in a risk register may therefore describe different types of exposure. Some correspond to relatively well-defined events whose occurrence and consequences can be articulated within a traditional risk formulation. Others instead reflect uncertainty regarding the evolution or consolidation of dependencies that remain outside the project's direct span of control.

This distinction is analytically relevant because the dependency-based approach developed in this thesis interprets risks as observable expressions of deeper structural conditions within the dependency landscape. What appears in the risk register may therefore not always represent a discrete event, but may instead reflect uncertainty regarding the viability or functioning of key dependencies.

For this reason, the first analytical step in this section involves a reclassification of the sixteen registered top risks. Each item is examined to determine whether it primarily reflects a discrete event-based risk or whether it expresses uncertainty related to the development of external dependencies. This step allows the subsequent analysis to relate the project's risk landscape to the critical dependencies and dominant propagation pathways identified in Section 7.5. This step therefore does not redefine the project's risks, but reinterprets them through the dependency framework developed in this study.

### 7.6.2. Identification of non risks: event based logic versus structural uncertainty

The reclassification was conducted using the sixteen "top risks" registered within the Aramis project. A detailed review showed that twelve of these items do not represent discrete, triggerable events. Instead, they describe ongoing conditions, structural arrangements, or external dynamics that evolve over time and cannot be reduced to a clearly identifiable moment of occurrence.

These items exhibit the following characteristics:

- They do not possess a clearly identifiable occurrence moment. Rather than describing a triggerable incident, they concern ongoing conditions whose evolution cannot be reduced to a single event.
- Their drivers are located in external institutional, market, regulatory, or governance systems beyond the project's span of control. The project depends on their evolution but cannot determine their trajectory.
- The associated "mitigation" actions are not preventive controls aimed at avoiding an event. They consist primarily of monitoring, coordination, negotiation, or alignment efforts, which confirms that the project is managing exposure rather than controlling occurrence.
- Their persistence is rooted in epistemic or ontological uncertainty concerning the stability of external conditions and enabling mechanisms. They are therefore not meaningfully expressible in terms of discrete probability of occurrence.

Among these twelve items, three represent internally rooted uncertainties, relating to internal shareholder approval processes, internal governance arrangements, and the organisational mandate of the joint project team. While these issues remain relevant for project governance, they do not reflect structural exposure to external systems and therefore fall outside the scope of the dependency-based analysis.

The remaining nine out of the sixteen top items, however, are external in nature. They arise from conditions located beyond the project boundary and therefore fall squarely within the domain of the dependency framework. The Appendix C.1 shows the risk from the risk register and the analysis/elaboration on how the risks are reframed to dependencies. First the original descriptions of these risks are provided in Appendix C.1, which gives an overview of the risks from the risk register, including their probabiliy times impact score. The tables below the register in Appendix C.1 analyses these items, showing the hidden external dependency, their current mitigation and why that mitigation strategy remains limited: because the underlying issue concerns uncertainty surrounding external conditions rather than a clearly defined event. In these tables the risks are reformulated as dependencies on external systems, expressing them in terms of the external conditions on which the project relies. This step does not yet analyse how these dependencies interact with the broader dependency structure, but clarifies which underlying external conditions are reflected in the project's risk register.

However, such dependencies rarely operate in isolation. Each of them is typically associated with a wider set of development conditions that influence how the dependency evolves over time. The next section therefore examines these external risks through the lens of the sub-dependencies identified in the MDF analysis, allowing the analysis to identify which development conditions must stabilise before the associated risks can meaningfully diminish.

## 7.7. Selection of the nine external uncertainties for dependency analysis

This analysis yields the following nine risks: R-X, R-Y, R-Z, R-A, R-B, R-C, R-D, R-E, R-F. The risks reframed to dependencies, as derived in Appendix C.2.

### Case Example: Unfolding Risk R-X into Its Dependency Structure

In Appendix C.3.2, an total overview of all nine risks are unfolded into sub-dependencies. Appendix C.3.1 gives an elaborated example of only risks R-X to see how these development indicators are actually underlying the risk and what has to be stabilised first. This section will further elaborate on how that is done.

This section provides a complete decomposition of Risk R-X, one of the most consequential externally rooted uncertainties identified in the Aramis risk register. This risk is formally framed as a volume shortfall event, but when interpreted through the dependency framework, it becomes clear that the underlying problem is not a discrete risk event but a structural uncertainty concerning the external conditions required for emitters to reach timely and binding investment commitments.

1. R-X is reframed as a dependency (see Appendix C.2)

This reframing shows that it is not a risk that can be mitigated through isolated measures. It is a condition on which the project depends. The project cannot “control” its occurrence; it can only align itself with how external commitment conditions evolve.

2. Enabling Dependencies Associated with R-X

This R-X dependency condition is operationalised through multiple enabling dependencies, including:

- C1.1 - Cost effectiveness of CCS versus alternatives
- C1.2 - $CO_2$ specification as an access and cost criterion
- C1.3 - Emitter FID decision complexity
- C1.4 - Total volume commitment stability
- C2.2 - Routing flexibility and competition exposure
- C4.3 - Single point of failure exposure
- C5.5 - Store investment commitment logic
- C6.1 - Tariff formation and cost discovery
- C6.3 - FID synchronisation across the value chain
- C6.6 - Permitting finality and legal operability

These enabling dependencies describe the structural mechanisms through which emitter commitments may stabilise over time. Their associated sub-dependencies capture the underlying development conditions that shape how these mechanisms currently evolve.

When R-X is examined through this lens, it no longer appears as a single risk item but as a configuration of 24 interacting epistemic uncertainties, shown in Table 7.10. Rather than representing a discrete event, the risk unfolds into a set of development conditions that jointly determine whether emitter commitments can stabilise.

R-X Dependency Unfolding

| Sub dependency (development indicator) | Sub (G/O/R) | Primary | Secondary | Value chain segment |
|---|---|---|---|---|
| Industrial relocation or cross border storage | Red | A2 | B2 | Emitters / Capture |
| Tariff versus ETS/penalty comparison | Orange | A1 | A2 | Emitters / Capture |
| Volume-dependent tariff escalation (€/t sensitivity) | Orange | A2 | D2 | Emitters / Capture |
| Subsidy sufficiency under cost overrun scenarios | Orange | A1 | A2 | Emitters / Capture |
| Re-application risk under SDE++ ranking rules | Orange | A1 | A2 | Emitters / Capture |
| Purity variation by industrial source | Green | A2 | D1 | Emitters / Capture |
| Additional processing requirement driven by spec stringency | Orange | D1 | A2 | Emitters / Capture |
| Spec-driven capex/opex uncertainty at emitter level | Red | D1 | A2 | Emitters / Capture |
| Dependency on timely publication of tariffs and specifications | Orange | A1 | A2 | Emitters / Capture |
| Technology maturity and deployment readiness | Orange | C1 | A2 | Emitters / Capture |
| Data certainty at investment committee stage | Orange | A2 | B1 | Emitters / Capture |
| Emitter permit and utilities complexity | Orange | C2 | A2 | Emitters / Capture |
| Volume change triggering chain-wide re-studies (flow assurance, hydraulics) | Green | A2 | C1 | Emitters / Capture |
| Anchor-emitter dependency effects | Green | A2 | D2 | Emitters / Capture |
| Shipping as alternative routing option | Orange | A2 | B2 | Feeder Transport |
| Volume leakage through arbitrage to non-Aramis routes | Red | A2 | B2 | Feeder Transport |
| Trunkline downtime halting entire value chain | Green | D2 | A2 | Offshore |
| Launch stores 5 megaton start-up momentum | Orange | C1 | B1 | Stores |
| Stores competition by design to enhance the market for CCS commitment | Orange | B1 | C1 | Stores |
| BAFO / CFT timing and late tariff certainty | Green | A2 | D2 | Cross-Cutting |
| Aramis government volume guarantee (Volloprisico) | Green | B2 | A1 | Cross-Cutting |
| Chicken-and-egg dynamics across >20 parties | Orange | C1 | B1 | Cross-Cutting |
| Mutual dependence of investment commitments | Red | C1 | A2 | Cross-Cutting |
| Interdependency of permits across chain segments | Orange | C2 | B1 | Cross-Cutting |

**Table 7.10:** R-X unfolded in its sub-dependencies underlying the risk

No single mitigation can resolve this configuration. Even if one enabling condition improves, for example through increased tariff clarity, others may deteriorate, such as cross-border leakage risks or specification-related cost exposure. The structural exposure therefore lies in simultaneity: emitter commitments only consolidate when several enabling mechanisms stabilise together.

From a dependency perspective, what appears in the risk register as one item therefore represents a cluster of interrelated development conditions that collectively determine whether the underlying viability condition can hold.

The same unfolding procedure was subsequently applied to all nine externally rooted risks. Each registered risk was decomposed into its associated enabling dependencies and further into sub-dependencies capturing the epistemic development conditions beneath them (Appendix C.3.2: All risks unfolded in sub-dependencies).

The result is a total unfolding into 174 sub-dependencies (development indicators) out of the nine top-risks:

- R-X in 24 sub-dependencies
- R-Y in 45 sub-dependencies
- R-Z in 34 sub-dependencies
- R-A in 9 sub-dependencies
- R-B in 8 sub-dependencies
- R-C in 6 sub-dependencies
- R-D in 19 sub-dependencies
- R-E in 21 sub-dependencies
- R-F in 8 sub-dependencies

This demonstrates that the nine “risks” in the register do not represent nine isolated exposure points. Instead, they reflect configurations of epistemic uncertainties embedded in enabling mechanisms across the value chain.

The dependency-based unfolding therefore transforms the interpretation of the risk landscape. What initially appears as a limited set of discrete threats reveals itself as a structured network of development conditions. These conditions must stabilise collectively before the underlying dependencies can consolidate and project commitment can be considered robust.

### 7.7.1. Cross-risk ranking of enabling and sub-dependencies

Having unfolded the nine externally rooted risks into their underlying 174 sub-dependencies, the first analytical step examines how often these development conditions recur across the nine risks. This cross-risk recurrence analysis identifies which sub-dependencies appear most frequently in the risk landscape, independent of their current development status (green, orange, red).

Recurrence indicates how often particular development conditions are embedded within different risks. Table 7.11 presents a condensed overview of this ranking, extracted from the full aggregation analysis in Appendix C.3.3. This Table shows the ID, vulnerability of the associated enabling dependency, followed by which sub-dependency, the recurrence in the 9 risks, and the status of the sub-dependency.

A high recurrence frequency suggests that multiple risks are informed by the same underlying development conditions. In other words, the risk landscape reflects overlapping sub-mechanisms rather than isolated risk events. This does not imply a direct causal relationship between risks; rather, it shows that different risks are associated with the same recurring patterns of epistemic development conditions. Recurrence therefore highlights which development conditions exert the widest structural influence across the risk landscape.

| ID | Vuln. | Associated Enabling | Sub-dependency (development indicator) | # / 9 Risk | Status |
|---|---|---|---|---|---|
| C4.2 | MIDDLE | Safety-by-interface dependency | Failure-mode discovery & safety case evidence (no rulebook) | 5 | Orange |
| C1.2 | OUTER | $CO_2$ specification as access and cost criterion | Purity variation by industrial source | 4 | Green |
| C4.1 | OUTER | Offshore transport integrity and operability | Corrosion and impurity sensitivity | 4 | Orange |
| C6.1 | OUTER | Tariff formation and cost discovery | BAFO / CFT timing and late tariff certainty | 4 | Green |
| C6.1 | OUTER | Tariff formation and cost discovery | Aramis government volume guarantee (Vollooprisico) | 4 | Green |
| C6.6 | OUTER | Permitting finality and legal operability | Legal finality versus reversibility | 4 | Red |
| C1.2 | OUTER | $CO_2$ specification as access and cost criterion | Additional processing requirement driven by specification stringency | 3 | Orange |
| C1.2 | OUTER | $CO_2$ specification as access and cost criterion | Spec-driven CAPEX/OPEX uncertainty at emitter level | 3 | Red |
| C1.4 | OUTER | Total volume commitment reliability | Volume change triggering chain-wide re-studies (flow assurance, hydraulics) | 3 | Green |
| C1.4 | OUTER | Total volume commitment reliability | Porthos start-up credibility as market confidence trigger | 3 | Orange |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | $CO_2$Next government volume guarantee (Vollooprisico) | 3 | Green |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | Cost allocation under interface design changes | 3 | Green |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | $CO_2$Next interface compatibility (pressure, quality, flow envelope) | 3 | Red |

**Table 7.11:** Sub-dependencies most frequently occurring across the analysed risks.

**Sub View with Red Development Signals**

From the cross risk recurrence list, the analysis then isolates all sub dependencies with a red development status, shown in table 7.12.

For each red sub dependency, the table specifies:

- The vulnerability ring of its enabling dependency,
- The associated enabling dependency,
- How often the sub occurs across the nine risks, and
- In which risks it appears.

This view is more than a list of deteriorating signals. It identifies the sub dependencies that simultaneously:

- Recur across several risks,
- Currently exhibit red development, and
- Sit under enabling dependencies with outer ring structural vulnerability

This combination highlights where adverse developmental movement aligns with structurally exposed parts of the dependency configuration.

Whereas Table 7.11 showed which sub mechanisms are structurally recurrent, table 7.12 highlights where those same mechanisms are now developing unfavourably, due to its combination of all outer enabling dependencies together with red sub-dependencies.

| ID | Vuln. | Enabling label | Sub (mechanism) | # / 9 Risks | Status | In Risks |
|---|---|---|---|---|---|---|
| C6.6 | OUTER | Permitting finality and legal operability | Legal finality versus reversibility | 4 | Red | 72, 106, 119, 62 |
| C1.2 | OUTER | $CO_2$ specification as access and cost criterion | Spec-driven CAPEX/OPEX uncertainty at emitter level | 3 | Red | 44, 72, 68 |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | $CO_2$Next interface compatibility (pressure, quality, flow envelope) | 3 | Red | 65, 414, 62 |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | $CO_2$Next commissioning alignment with Aramis start-up | 2 | Red | 72, 65 |
| C6.3 | OUTER | FID synchronisation across the value chain | Mutual dependence of investment commitments | 2 | Red | 44, 72 |
| C2.2 | OUTER | Routing flexibility and leakage exposure | Volume leakage through arbitrage to non-Aramis routes | 1 | Red | 44 |
| C4.5 | OUTER | Execution capacity & schedule realism | Contractor scarcity & schedule gate | 1 | Red | 62 |

**Table 7.12:** Red sub-dependencies associated with enabling dependencies positioned in the outer vulnerability ring.

The result is a map of cross risk pressure points: red sub dependencies that cluster beneath enabling dependencies positioned in the outer structural vulnerability ring, meaning that the underlying mechanisms are both structurally exposed and epistemically trending in an adverse direction. A particularly notable outcome is the prominence of C6.6 - Permitting finality and legal operability, with the sub-dependency Legal finality versus reversibility, which combines:

- Outer ring structural vulnerability,
- Red development, and
- The highest cross risk recurrence among red subs.

This makes it the most clearly concentrated red sub dependency under an outer-ring enabling mechanism across the nine analysed risks.

### Enabling-Level Aggregation

To understand how the 174 identified sub-dependencies distribute across the dependency structure, all subs are aggregated to their associated enabling dependencies. The full aggregation is presented in Table 7.13, and can also be found in Appendix C.3.4.

This aggregation provides a structural overview of the risk landscape by showing:

- how often the dependency appears across the nine risks,
- how many epistemic development signals (sub-dependencies) are associated with each mechanism, and
- the distribution of these sub-dependencies across red, orange, and green development statuses.

| ID | Enabling Label | Vuln. | Appears in #/9 | Subs in #/9 | Red | Orange | Green |
|---|---|---|---|---|---|---|---|
| C4.2 | Safety-by-interface dependency | MIDDLE | 6 | 12 | 0 | 7 | 5 |
| C1.4 | Total volume commitment reliability | OUTER | 6 | 8 | 0 | 3 | 5 |
| C3.1 | $CO_2$Next system readiness & commitment | OUTER | 5 | 15 | 5 | 0 | 10 |
| C1.2 | $CO_2$ specification as access and cost criterion | OUTER | 5 | 12 | 3 | 3 | 6 |
| C6.1 | Tariff formation and cost discovery | OUTER | 5 | 12 | 0 | 4 | 8 |
| C6.6 | Permitting finality and legal operability | OUTER | 5 | 12 | 4 | 8 | 0 |
| C3.4 | Design constraints from external system drivers | OUTER | 5 | 9 | 1 | 5 | 3 |
| C3.3 | Porthos system integration & readiness | OUTER | 4 | 11 | 0 | 9 | 2 |
| C4.1 | Offshore transport integrity and operability | OUTER | 4 | 9 | 0 | 7 | 2 |
| C6.5 | Strategic optionality and system rigidity | OUTER | 4 | 4 | 0 | 3 | 1 |
| C5.5 | Stores investment commitment logic | MIDDLE | 3 | 8 | 0 | 8 | 0 |
| C4.4 | Off-spec and upset handling procedures | OUTER | 3 | 5 | 0 | 5 | 0 |
| C6.2 | System integration governance | MIDDLE | 3 | 5 | 0 | 2 | 3 |
| C1.1 | Cost-effectiveness of CCS versus alternatives | MIDDLE | 2 | 10 | 1 | 9 | 0 |
| C6.4 | Chain-wide contractual coherence | OUTER | 2 | 8 | 0 | 6 | 2 |
| C3.2 | Compression capacity, availability & redundancy | OUTER | 2 | 6 | 0 | 4 | 2 |
| C1.3 | Emitter FID decision complexity | MIDDLE | 2 | 5 | 0 | 5 | 0 |
| C2.2 | Routing flexibility and leakage exposure | OUTER | 2 | 5 | 2 | 3 | 0 |
| C6.3 | FID synchronisation across the value chain | OUTER | 2 | 4 | 2 | 2 | 0 |
| C2.1 | Infrastructure synchronisation across feeder assets | INNER | 2 | 3 | 2 | 1 | 0 |
| C2.3 | Balancing and contingency arrangements | MIDDLE | 2 | 3 | 0 | 3 | 0 |
| C4.3 | Single-point-of-failure exposure | OUTER | 2 | 2 | 0 | 0 | 2 |
| C5.4 | $CO_2$ specification bounded by storage constraints | OUTER | 2 | 2 | 0 | 2 | 0 |
| C4.5 | Execution capacity & schedule realism | OUTER | 1 | 2 | 1 | 1 | 0 |
| C5.3 | Store availability and revenue exposure | INNER | 1 | 2 | 0 | 2 | 0 |

**Table 7.13:** Distribution of development indicator statuses across enabling dependencies for the nine analysed risks.

Table 7.13 shows that several enabling dependencies recur frequently across the nine risks. For example, C4.2 - Safety-by-interface dependency and C1.4 - Total volume commitment reliability each appear in six risks. However, recurrence alone does not indicate developmental instability. In both cases the associated sub-dependencies are entirely orange or green, meaning that no unfavourable development signals are currently observed within those mechanisms.

A different pattern emerges when the number of associated sub-dependencies is considered. The column "Subs" shows how many development indicators are associated with each enabling dependency across the nine risks. In this distribution, C3.1 - $CO_2$Next system readiness & commitment stands out most prominently, with 15 sub-dependencies across five risks, of which five are assessed as red.

Several other enabling dependencies also contain relatively large numbers of sub-dependencies. C4.2, C1.2, C6.1, and C6.6 each contain twelve sub-dependencies distributed across the nine risks. Their development signals, however, differ markedly. While C4.2 and C6.1 contain no red development indicators, C1.2 contains three red sub-dependencies and C6.6 contains four.

Taken together, this distribution shows that a substantial share of the red development indicators occurs within a small subset of enabling dependencies that also contain relatively large numbers of associated sub-dependencies. In other words, the mechanisms that account for the largest share of development signals are also those in which most of the unfavourable development indicators appear.

To isolate these mechanisms more clearly, Table 7.14 extracts the enabling dependencies that combine red development indicators with outer structural vulnerability, and shows in which risks these red sub-dependencies occur.

Within this subset, C3.1 - $CO_2$Next system readiness & commitment, C1.2 - $CO_2$ specification as access and cost criterion, and C6.6 - permitting finality and legal operability stand out most clearly. These enabling dependencies combine two characteristics simultaneously: they contain a relatively large share of the aggregated sub-dependencies identified across the nine risks, and they account for a substantial portion of the red development indicators. Appendix C.3.5 shows in more details de dominant propagation paths and in which paths these red sub-dependencies are associated and clustered.

| ID | Enabling label | Vuln. | In #/9 Risks | Enabling Subs Count | Red Subs | In Risks |
|---|---|---|---|---|---|---|
| C3.1 | $CO_2$Next system readiness & commitment | OUTER | 5 | 15 | 5 | 72, 65, 414, 62, 40 |
| C1.2 | $CO_2$ specification as access and cost criterion | OUTER | 5 | 12 | 3 | 44, 72, 68, 106, 62 |
| C6.6 | Permitting finality and legal operability | OUTER | 5 | 12 | 4 | 44, 72, 106, 119, 62 |
| C3.4 | Design constraints from external system drivers | OUTER | 5 | 9 | 1 | 72, 68, 106, 414, 62 |
| C2.2 | Routing flexibility and leakage exposure | OUTER | 2 | 5 | 2 | 44, 72 |
| C6.3 | FID synchronisation across the value chain | OUTER | 2 | 4 | 2 | 44, 72 |
| C4.5 | Execution capacity & schedule realism | OUTER | 2 | 4 | 2 | 40, 62 |

**Table 7.14:** Enabling Dependencies with Aggregated Subs in Risks

### Risk-Level Perspective

This structural concentration becomes even clearer when the individual risks are examined in more detail.

Table 7.15 lists the nine externally rooted risks together with their PxI score, the number of associated sub-dependencies identified in the unfolding analysis, and the enabling dependencies that contain red development indicators underlying each risk.

The table shows that the highest-ranked risks in the register are consistently associated with the same enabling mechanisms. The two risks with the highest PxI score - R-X and R-X (both PxI = 80) - each contain multiple enabling dependencies with red development indicators. These mechanisms therefore do not belong to a single risk item but recur across several risks.

Seen together, the nine risks therefore do not represent nine independent exposure points. Instead, they repeatedly draw upon a limited set of enabling mechanisms that structure the underlying dependency landscape.

| Row | Risk | P x I | Subs | Enabling with Red in Subs |
|---|---|---|---|---|
| 1 | R-X | 80 | 24 | **C1.2**, **C2.2**, **C6.3** |
| 2 | R-Y | 80 | 45 | **C1.2**, **C3.1**, **C6.3**, **C6.6** |
| 3 | R-Z | 80 | 34 | **C3.1** |
| 4 | R-A | 80 | 9 | **C1.2** |
| 5 | R-B | 64 | 8 | **C6.6** |
| 6 | R-C | 50 | 6 | **C6.6** |
| 7 | R-D | 50 | 19 | **C3.1** |
| 8 | R-E | 40 | 8 | **C3.1**, **C4.5**, **C6.6** |
| 9 | R-E | 40 | 21 | **C3.1**, **C4.5**, **C6.6** |

**Table 7.15:** Risk items and associated enabling dependencies carrying red sub-dependencies.

### Integrating the Enabling and Risk Perspectives

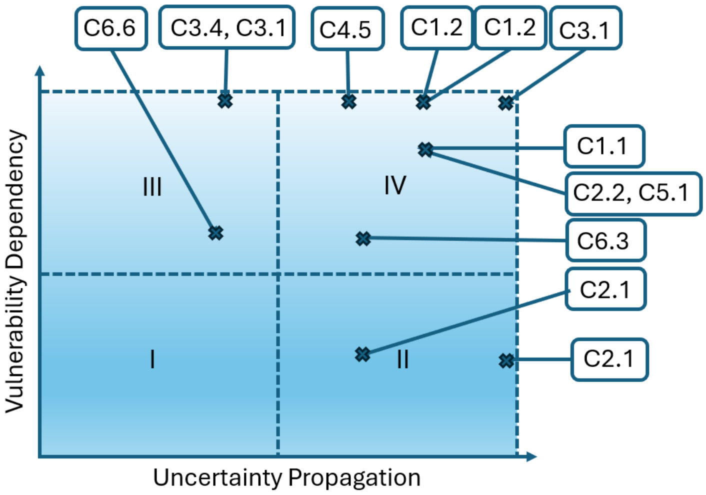


**Figure 9:** Reproduction of Table 7: Dependency Instability

Taken together, Table 7.13, 7.14, 7.15, reveal that the risk landscape of the Aramis project is structured by a limited number of enabling mechanisms rather than by a wide range of independent risk drivers.

Table Table 7.13 first shows how the 174 sub-dependencies distribute across the enabling dependency structure. Two patterns stand out. First, the enabling dependencies that host the largest number of sub-dependencies - most notably C3.1, C1.2, C6.6 and C4.2 - account for a substantial share of the development indicators observed across the nine risks. Second, many of the red development indicators appear within enabling dependencies positioned in the outer vulnerability ring, indicating that the most critical development signals tend to occur in structurally exposed dependencies.

Table 7.14 then isolates this subset by focusing specifically on enabling dependencies that combine outer structural vulnerability with red development indicators. The table therefore highlights the enabling mechanisms where unfavourable development signals coincide with structural exposure. Within this subset, C3.1 - $CO_2$Next system readiness & commitment, C1.2 - $CO_2$ specification as access and cost criterion, and C6.6 - permitting finality and legal operability stand out because they combine a relatively large number of development indicators with multiple red signals.

Table 7.15 subsequently reverses the analytical perspective by examining the risks themselves. Instead of asking which enabling dependencies host critical development signals, the analysis now considers which enabling mechanisms underlie the individual risks in the register. This perspective shows that the same enabling dependencies repeatedly appear beneath the most highly ranked risks. The two risks with the highest PxI scores, R-X and R-X, both draw upon several of the enabling mechanisms identified in Table 7.14, including C1.2, C3.1, C6.3, and C6.6. Other high-ranked risks are similarly associated with these same dependencies.

Seen together, the three tables demonstrate that the risks recorded in the register are not independent exposure points. Instead, they emerge from a limited number of enabling mechanisms where many development conditions converge and where several critical development signals occur.

This pattern becomes particularly clear when these findings are related back to the dependency instability matrix shown in Figure 9. The enabling dependencies that dominate the risk register - especially C3.1, C1.2, C2.2 and C6.3 - are positioned in the upper-right quadrant of the matrix, where high structural vulnerability coincides with strong uncertainty propagation. Dependencies located in this region combine exposure to disturbances originating in external systems with a high potential for uncertainty to propagate across multiple governance contexts.

The alignment between the dependency instability analysis and the empirical risk register is therefore striking. The enabling dependencies identified as structurally critical within the MDF are precisely those that recur most prominently within the project's articulated risks.

In this sense, the risk register does not introduce a new structure of exposure. Rather, it expresses the same structural instability that the dependency analysis reveals within the project system. The most highly ranked risks correspond to the dependencies where both vulnerability and uncertainty are most pronounced.

This correspondence provides a strong validation of the analytical framework developed in this research. The dependency instability matrix anticipates where risks are most likely to arise by identifying the dependencies where structural exposure and uncertainty propagation coincide. The risk register subsequently makes these structural exposures visible within the language of project risk

## 7.8. Conclusion: dependency instability and risk articulation

This chapter analysed the Aramis project from two complementary analytical entry points.

First, a system-oriented perspective mapped the project's dependency architecture in order to identify where the project is structurally exposed to external conditions beyond its span of control. By analysing viability and enabling dependencies in terms of structural vulnerability, uncertainty propagation, and contextual placement, the analysis revealed which dependencies have the potential to transmit disturbances across the project system and in which governance contexts these effects would become consequential.

This system view resulted in the formulation of the Megaproject Dependency Framework (MDF) and the dependency instability matrix, which together identify where structurally exposed dependencies coincide with dominant propagation pathways. In this sense, the dependency instability matrix can be interpreted as an analytical precursor to conventional risk articulation: it identifies the structural conditions under which disturbances are most likely to propagate through the project system before they appear as formally articulated risks in governance practice.

Second, a risk-oriented perspective reconstructed the project's risk register by unfolding the nine externally rooted risks into their underlying enabling dependencies and development conditions. This reconstruction showed that risks do not represent isolated exposure points but repeatedly draw upon the same underlying enabling mechanisms.

When both perspectives are integrated, a clear pattern emerges. The enabling dependencies that appear most prominently in the structural dependency mapping - particularly C3.1 ($CO_2$Next system readiness and commitment), C1.2 ($CO_2$ specification as access and cost criterion), and C6.6 (permitting finality and legal operability) - are the same mechanisms that recur most frequently across the project's risk register.

This convergence indicates that the risk register does not introduce a separate layer of uncertainty. Rather, it articulates instability already present in the dependency architecture of the project. Risks therefore emerge where structurally exposed enabling mechanisms coincide with development conditions that have not yet stabilised.

From a governance perspective, this implies that assessing project readiness cannot rely solely on monitoring individual risks. Instead, attention must also be directed toward the development of the enabling mechanisms that repeatedly underpin those risks. As these mechanisms stabilise, the structural exposure of the project configuration decreases. If they remain developmentally strained, the underlying vulnerability persists regardless of how risks are articulated.

This distinction clarifies an important governance implication. Risk mitigation becomes meaningful only once the enabling mechanisms that generate those risks begin to stabilise. Attempting to mitigate risks while the underlying dependencies remain structurally exposed risks treating symptoms rather than addressing the conditions that generate them. In this sense, the dependency-based analysis reframes the governance task from mitigating articulated risks toward stabilising the enabling mechanisms that underpin them.

The analysis presented in this chapter therefore establishes the internal coherence of the dependency-based analytical framework. Structural dependency mapping and risk reconstruction, although starting from different analytical entry points, converge on the same enabling mechanisms as the primary carriers of uncertainty within the Aramis configuration.

In this sense, the governance challenge in early-stage megaproject development is not primarily to mitigate risks, but to stabilise the dependency conditions from which those risks arise.

The next chapter extends this evaluation through external comparison with the Porthos project, examining whether similar dependency patterns emerge in a comparable CCS development context.

# 8

# External Validation Through Risk Reconstruction: Porthos CCS

Chapter 7 demonstrated that risks articulated in the Aramis project do not represent independent exposure points. The dependency instability matrix showed where structurally exposed dependencies coincide with uncertainty propagation across governance contexts. These locations therefore indicate where structural instability is most likely to surface in governance practice in the form of articulated risks.

Chapter 8 examines whether this structural logic also holds beyond the Aramis case. To do so, it applies the same dependency-based analytical lens to the Porthos CCS project, focusing on the risks that remained open at Final Investment Decision (FID). More specifically, the chapter investigates whether these risks can be reconstructed as manifestations of enabling dependencies whose underlying instability had not yet been sufficiently stabilised at the moment of commitment.

Before analysing the Porthos risk register, a dependency structure was formulated using the same analytical logic applied in the Aramis case. This structure was developed independently from the risks themselves and serves as a structural hypothesis about where exposure could reside within the Porthos configuration. The subsequent analysis then examines whether the risks that remained open at FID correspond to these enabling dependencies and their insufficiently stabilised development conditions.

### Hypothesis

*The risks that remained open at FID in Porthos can be reconstructed as manifestations of dependency instability. Where uncertainty propagation coincides with structural vulnerability, these dependencies remain susceptible to activation under external disturbance, after which instability becomes visible as project risk.*

The hypothesis is supported if the majority of selected risks correspond to enabling dependencies whose instability had not yet been sufficiently stabilised at the moment of commitment.

### Analytical Logic

This hypothesis implies a different interpretation of risk than is commonly applied in project risk management. Rather than treating risks as discrete events requiring mitigation, the analysis interprets them as signals of unresolved structural exposure within the dependency architecture of the project.

Within the logic of the Megaproject Dependency Framework:

- Viability and enabling dependencies represent structural conditions for project viability. They carry ontological uncertainty and cannot be eliminated; they can only be stabilised.
- Development conditions determine whether structural vulnerability remains latent or becomes operationally relevant.

- Exogenous shocks (e.g., market volatility, regulatory change) do not create new dependencies but increase pressure on existing ones.
- Risks emerge when structurally exposed dependencies remain insufficiently stabilised under changing conditions.

Under this perspective, risks are not independent events but manifestations of dependency-level vulnerability under stress.

**Validation Procedure**
To test the hypothesis, the analysis follows a structured reconstruction approach:

1. Select the risks that remained open at Final Investment Decision (FID) in the Porthos project.
2. Identify the underlying uncertainties associated with each risk, distinguishing observable symptoms from their structural origins.
3. Map these uncertainties to the enabling dependencies they represent within the MDF structure.
4. Analyse the dependency paths through which uncertainty propagates, identifying how these dependencies connect to broader project domains.
5. Assess the structural vulnerability of the identified dependencies, examining whether their configuration exhibits limited tolerance to variation in project conditions.
6. Examine the disturbances that interacted with these dependencies, evaluating whether external triggers activated the latent instability embedded in the dependency configuration.
7. Evaluate whether the realised risk can therefore be reconstructed as the manifestation of activated dependency instability.

If the majority of open FID risks can be systematically reconstructed in this way, the hypothesis is empirically supported.

## 8.1. Porthos CCS megaproject

Porthos is a large-scale carbon capture and storage (CCS) megaproject located in the Port of Rotterdam, see figure 1. The project is being developed and built by a consortium consisting of EBN (Energie Beheer Nederland), Gasunie, and the Port of Rotterdam Authority. The project captures $CO_2$ from a predefined group of industrial emitters - primarily Shell, ExxonMobil, Air Liquide, and Air Products - and transports it through an onshore pipeline of approximately 30 km to the compressor station. From there, a subsea pipeline of roughly 20 km transports the $CO_2$ to the injection platform located 20 km offshore. The $CO_2$ is subsequently injected into depleted natural gas reservoirs at depths exceeding 3 kilometres beneath the North Sea. The system is expected to become operational in 2026.

With a planned transport capacity of approximately 2.5 million tonnes of $CO_2$ per year, Porthos represents the first CCS transport-and-storage system at this scale within the European Union. The allocated storage volume amounts to 37.5 megatons, corresponding to roughly five percent of current Dutch industrial emissions. The project has been designated a Project of Common Interest by the European Union and received funding under the Connecting Europe Facility, reflecting its strategic importance for European decarbonisation infrastructure.

A central technical component of the Porthos system is the compressor station, which enables the transport of captured $CO_2$ from multiple emitters toward the offshore storage site. Because industrial capture volumes fluctuate over time, the system must accommodate variable inflow conditions while maintaining stable transport pressure. Three compressors increase the pressure of the $CO_2$ stream from approximately 30 bar to between 90 and 130 bar, enabling transport through the offshore pipeline and injection into the depleted gas field where storage pressures approach 300 bar. The compressor installation has been designed with future system integration in mind and may also serve subsequent CCS developments such as Aramis and $CO_2$next.

Structurally, Porthos differs from Aramis in two important respects. First, the system includes the compressor station as an integrated component of the project architecture. Second, Porthos operates largely as a closed-access value chain with predefined counterparties and early contractual lock-in. Whereas Aramis is designed as an expandable open-access backbone intended to accommodate future emitters and storage sites, Porthos is configured around a limited set of initial participants and fixed storage assets.

Unlike Aramis, Porthos has already passed Final Investment Decision (FID) and entered execution. During implementation, the project encountered significant delays and several manifested uncertainties, including nitrogen-related permitting constraints, cost escalations linked to market conditions, integration challenges across chain segments, and coordination pressures among stakeholders. These developments provide an observable post-FID uncertainty trajectory.

The purpose of this chapter is therefore not to compare project outcomes, but to examine whether the dependency logic identified in Chapter 7 also explains the uncertainties encountered in the Porthos project. By analysing the risks that remained open at Final Investment Decision (FID), the chapter investigates whether these risks can be reconstructed as manifestations of enabling dependencies whose instability remained unresolved at the moment of commitment. If this reconstruction holds for the majority of cases, the analysis provides an external validation of the Megaproject Dependency Framework.

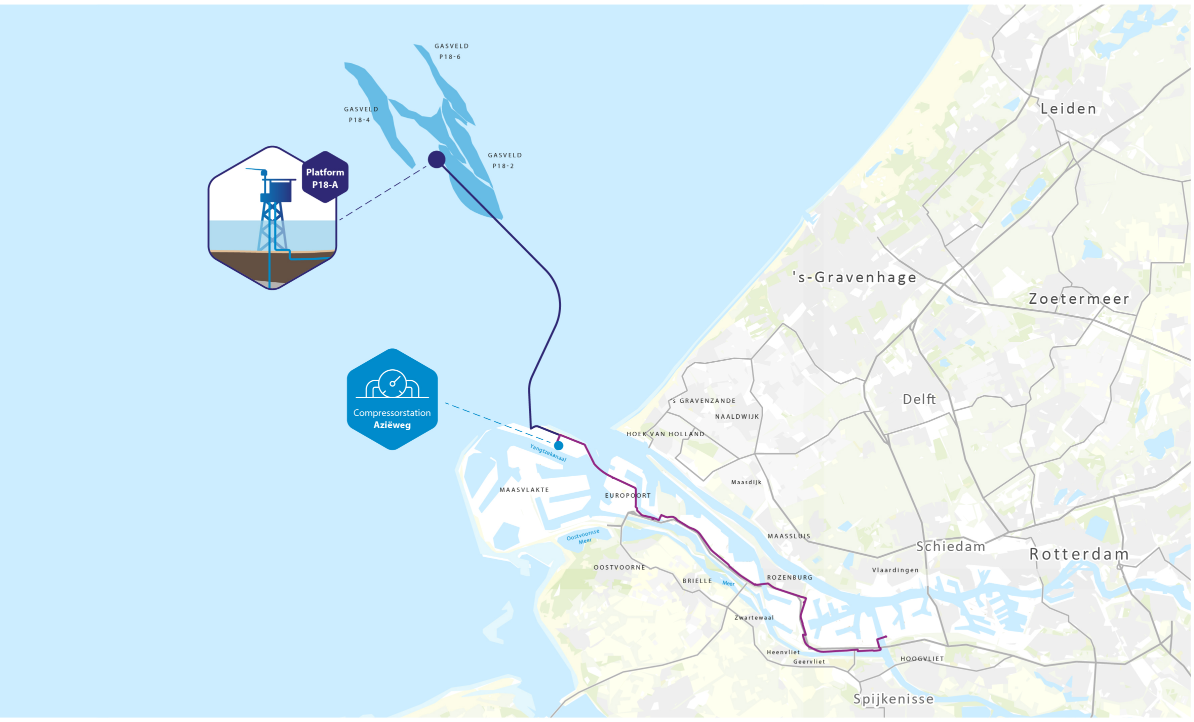


**Figure 1:** Porthos CCS, (Media - Porthos. (2026, February 25). Porthos. https://www.porthos$CO_2$.nl/media/)

## 8.2. Selection of Porthos risks and ex-ante dependency structure

Before analysing the risks recorded in Porthos at the moment of FID, an enabling-dependency configuration was first established. This configuration was formulated prior to consulting the Porthos risk register and therefore represents an ex-ante structural hypothesis about where vulnerability could reside within the project's dependency architecture.

The preliminary configuration was based on two sources:

1. The structural similarities between the Aramis and Porthos value chains;
2. Insights from Aramis professionals with experience in both projects.

Using the same value-chain segmentation applied in Chapter 7, a Porthos-specific set of 23 enabling dependencies was formulated. Although Porthos differs from Aramis in several respects - most notably its closed-access configuration and compressor-centric design - the underlying CCS value-chain architecture remains structurally comparable. This allows the same analytical logic of enabling dependencies to be applied while adapting the specific dependency set to the Porthos configuration.

This step is methodologically critical because it separates structural reasoning from empirical observation. The enabling dependencies were formulated ex-ante as a structural hypothesis about where vulnerability could arise in a full-chain CCS system. They were therefore not reverse-engineered from the risks observed at FID.

The enabling dependencies initially identified as structurally relevant for Porthos were organised along the same value chain segmentation as applied in Aramis, resulting in a slightly different set of enabling dependencies. Resulting in Table 8.1:

| Value Chain Segment | Enabling Dependencies |
|---|---|
| Emitter | C1.1 Cost-effectiveness of CCS versus alternatives |
| | C1.2 $CO_2$ specification as access and cost criterion |
| | C1.3 Emitter decision complexity |
| | C1.4 Total volume commitment reliability |
| Transport Feeder Backbone | C2.1 Persistence of industrial activity within the Port of Rotterdam footprint |
| | C2.2 Routing flexibility and competition exposure |
| | C2.3 Compatibility of long-term industrial location with a fixed CCS backbone |
| | C2.4 Exposure of the backbone concept to industrial transition, relocation or decline |
| Conditioning System Interface | C3.1 Commitment and realization of future third-party CCS systems |
| | C3.2 Interface overcapacity driven by uncertain future integration needs |
| | C3.3 Design constraints from external system drivers |
| | C3.4 Execution capacity schedule realism |
| Distribution – $CO_2$ flow into main pipeline | C4.1 Offshore transport integrity and operability |
| | C4.2 Safety-by-interface dependency |
| | C4.3 Single-point-of-failure exposure |
| | C4.4 Off-spec and upset handling procedures |
| Capacity Operational Readiness | C5.1 Storage capacity portfolio adequacy |
| | C5.2 Injectivity and operational flexibility |
| Cross-Cutting | C6.1 Tariff formation and cost discovery |
| | C6.2 FID synchronisation across the value chain |
| | C6.3 Chain-wide contractual coherence |
| | C6.4 Strategic optionality and system rigidity |
| | C6.5 Permitting finality and legal operability |
| | C6.6 Funding eligibility auditability (CEF) / cost traceability |

**Table 8.1:** Porthos enabling dependencies structured by value-chain segment.

### Selection of Risks for Validation

The next step was to confront the ex-ante dependency configuration with the risks recorded in the Porthos risk register at the moment of Final Investment Decision (FID). At that time (Y) risks were formally registered, of which X were selected for analysis, because of their external dependency characteristics. The real risks with description at the moment of FID are listed in Appendix D.1, while Appendix D bundles all appendices related to the Porthos case.

The selected X risks for validation were risks:

- #X

These risks were selected because they represent externally rooted uncertainties rather than purely internal execution issues. In other words, they concern forms of exposure that lie largely outside the direct control of the project organisation and therefore provide suitable cases for testing a dependency-based interpretation. The specifics of this analysis where the Porthos risks are reframed to external dependencies can be found in Appendix C.2.

They reflect, for example:

- Market and tariff related rigidity,
- Specification and commissioning uncertainty,
- Permitting and regulatory fragility,
- Subsidy eligibility and inter organisational governance alignment.

**Mapping Risks to Enabling Dependencies - Porthos**

| Risk ID | Code | Dependency Name |
|---|---|---|
| #X | C1.1 | Cost-effectiveness of CCS versus alternatives |
| #X | C3.4 | Execution capacity & schedule realism |
| #X | C6.1 | Tariff formation & cost discovery |
| #X | C6.3 | Chain-wide contractual coherence |
| #X | C1.2 | $CO_2$ specification as access & cost |
| #X | C4.3 | Single-point-of-failure exposure |
| #X | C4.4 | Off-spec & upset handling procedures |
| #X | C5.2 | Injectivity & operational flexibility |
| #X | C4.1 | Offshore transport integrity & operability |
| #X | C6.5 | Permitting finality & legal operability |
| #X | C6.3 | Chain-wide contractual coherence |
| #X | C1.2 | $CO_2$ specification as access & cost |
| #X | C1.4 | Total volume commitment reliability |
| #X | C5.2 | Injectivity & operational flexibility |
| #X | C6.5 | Permitting finality & legal operability |
| #X | C6.3 | Chain-wide contractual coherence |
| #X | C6.6 | Funding eligibility & auditability (CEF) |

**Table 8.2:** Enabling dependencies identified within selected Porthos risks.

Each selected risk was analysed and mapped to the enabling dependencies that logically carried the underlying uncertainty described in the risk register. The result is presented in Table 8.2.

Table 8.2 does not yet analyse uncertainty propagation. Instead, it identifies which enabling dependencies are implicated in each risk. The table therefore establishes a structural linkage between governance-articulated risks and the enabling mechanisms that carry the underlying uncertainty.

Importantly, this mapping did not require introducing new categories of enabling mechanisms beyond the preliminary set. The risks populate and specify the existing dependency configuration rather than expanding it. This indicates that the ex-ante structural hypothesis was sufficiently comprehensive to accommodate the risks observed at FID without conceptual adjustment.

The validation logic therefore starts with Table 8.2. If the selected risks can be systematically reconstructed in terms of these enabling dependencies, and if the underlying uncertainties correspond to conditions that had not yet stabilised at FID, the hypothesis is supported.

The next section therefore turns to the empirical reconstruction of uncertainty in the Porthos case.

## 8.3. Empirical reconstruction of uncertainty propagation in Porthos

Section 8.3 reconstructs how the uncertainties underlying the selected Porthos risks at FID became visible in practice after FID, and which enabling dependencies structurally carried that exposure. Rather than modelling a full sub-dependency structure as in Aramis, the analysis starts from empirically observed project difficulties, unexpected developments, and their realised consequences. These are then interpreted through the enabling-dependency layer of the MDF.

In the Porthos case, propagation is not analysed as an independent phenomenon. Instead, primary and secondary context placements are derived to indicate where uncertainty originated and where it became consequential once carried by a specific enabling dependency. In this way, the reconstruction does not begin from propagation paths as an analytical end in themselves. Instead, it begins from the risks and the underlying uncertainties they express, and then examines which enabling dependencies made those uncertainties governance-relevant within the project configuration.

### Interview Design

The professionals involved in Porthos were from complementary governance positions:

- Interviewee 1: A pre-FID Business Opportunity Manager/Joint Venture Representative who functioned as the linking pin between project team and shareholder;
- Interviewee 2: A post-FID Shareholder Representative responsible for governance alignment and mandate clarity;
- Interviewee 3: A Value Assurance Review Specialist (VAR) team member who assessed front-end maturity and stage-gate readiness prior to FID.

Together, these perspectives span business-case formation, shareholder decision-making, and independent assurance. This triangulation makes it possible to analyse uncertainty not merely as a technical issue, but as a governance condition that became consequential across multiple parts of the project.

Specific quotations and their contextual interpretation are presented in Appendix D.5 and more specifically structured in Appendix D.6.

### Analytical Reconstruction Logic

The analytical starting point was not a full sub-dependency model, but the concrete project difficulties described in interviews and supporting material. Each selected risk was examined by identifying the underlying uncertainties and unexpected developments associated with it. These were then mapped to the enabling dependencies they represent within the Megaproject Dependency Framework.

Because the detailed sub-dependency layer developed for Aramis could not be replicated for Porthos within the available time and data constraints, the analysis remains at enabling-dependency level. This does not weaken the reconstruction. On the contrary, the analysis focuses specifically on those uncertainties that did not remain hypothetical, but generated tangible tensions in governance and execution practice. As a result, the reconstructed dependencies are not inferred from abstract possibility, but from observed uncertainty that became consequential within the project.

This approach allows structurally vulnerable enabling dependencies to be identified without modelling all potential development conditions. Rather than reconstructing every possible epistemic condition, the analysis concentrates on those uncertainties that demonstrably generated pressure within the project system. The resulting mapping therefore shows which enabling dependencies were exposed, which uncertainties triggered that exposure, and how these dependencies correspond to the risks articulated in the Porthos risk register.

A first example concerns tariff fixation under the SDE++ framework. Interview evidence showed that the tariff had been fixed early in the development process, after which macro-economic volatility, inflation, and uncertainty surrounding ETS trajectories significantly affected the economics of the project. These developments exposed dependencies such as C1.1 (Cost-effectiveness of CCS versus alternatives) and C6.1 (Tariff formation and cost discovery). The key issue was not simply that a financial risk had been recorded, but that the stabilising conditions required to absorb such volatility - such as tariff indexation, re-opener clauses, or better synchronisation between subsidy logic and tariff confirmation - had not been structurally secured before commitment.

A second example concerns FOAK technical behaviour surrounding dense-phase $CO_2$ handling. Interview material referred to uncertainties around seal behaviour, flow-coating integrity, filter fouling, and commissioning under real operating conditions. These issues exposed dependencies such as C1.2 ($CO_2$ specification as access and cost criterion), C4.1 (Offshore transport integrity and operability), and C4.4 (Off-spec and upset handling procedures). Here too, the significance of the risk did not lie only in the later technical incidents themselves, but in the fact that concept-level assumptions had not yet been sufficiently stabilised through qualification, testing, and operational definition at the moment of FID.

### From Empirical Difficulties to Structured Dependencies

These examples illustrate the reconstruction logic applied across the Porthos case. The same analytical procedure was used for each enabling dependency: empirically observed project uncertainties were traced back to the dependency that structurally carried the exposure, after which their contextual placement and stabilising development conditions were derived.

Appendix D.7 presents the full structured reconstruction of the enabling dependencies identified for Porthos, including their empirical grounding, vulnerability characterisation, and contextual placement within the MDF framework.

An illustrative example is C1.1 – Cost-effectiveness of CCS versus alternatives, whose propagation path was reconstructed as A2 → D2.

### C1.1 - Cost Effectiveness of CCS vs Alternatives (A2 → D2)

Empirical basis (interview reconstruction):

Interview material consistently indicated that emitter participation and commitment were highly sensitive to the expected $CO_2$ price trajectory and the fixed tariff structure agreed under the subsidy framework. At the same time, macro-economic shocks - most notably inflation and energy-market volatility following the war in Ukraine - significantly increased project costs while the transport tariff had already been fixed due to SDE++ subsidy requirements.

This combination created pressure on the economic viability of CCS relative to alternative decarbonisation options. Financial projections indicated that internal rates of return remained limited, meaning that the economic buffer against external volatility was small.

One interviewee summarised the situation as follows:

*“CCS is financially marginal but has large climate impact; the lower IRR was accepted because of the societal objective.”*

**Primary context - A2:**

Uncertainty originated in macro-market conditions and emitter willingness-to-pay. Fixed tariffs combined with ETS price uncertainty and evolving climate policy expectations directly affected emitter commitment and investment confidence.

**Secondary context - D2:**

The economic pressure translated into system-level performance exposure: tighter margins increased sensitivity to cost escalation and reduced the economic robustness of the configured value chain.

**Visible structural vulnerability:**

A rigid tariff structure combined with volatile ETS and macro-economic conditions reduced the system’s autonomy to absorb external shocks.

**Derived stabilising development conditions:**

- Tariff mechanisms incorporating indexation or re-opener clauses instead of rigid fixation.
- Transparent ETS and price assumptions shared across the value chain to anchor emitter decision-making.
- Stronger synchronisation between SDE++ requirements and tariff confirmation processes.

This example illustrates the reconstruction logic used throughout the section. First, interview material identified uncertainties that had become consequential in practice. Second, these uncertainties were mapped to the enabling dependency that structurally carried the exposure. Third, primary and secondary context placements were derived to show where the uncertainty originated and where it became governance-relevant.

Applying this procedure across all identified enabling dependencies resulted in the dependency configuration presented in Table 8.3. Table 8.4 then aggregates these primary and secondary placements into recurring context transitions across the Porthos configuration. Table 8.4 is generalised in its description, a detailed explanation of case specific Porthos issues can be found in Appendix D.4.

Table 8.3 shows that several dependencies repeatedly originate in D1 and C1, while their effects frequently surface in D2. This indicates that uncertainty in Porthos often arose either from concept-level assumptions or from chain-interface conditions and subsequently became consequential at the level of required value-chain performance. Table 8.4 confirms this pattern by showing that the most recurrent transitions are D1 → D2, C1 → D2, and B1 → C1, with C2 → B2 and A2 → D2 also appearing as meaningful routes.

| Porthos Dependencies | Primary | Secondary |
|---|---|---|
| C1.1 Cost-effectiveness of CCS versus alternatives | A2 | D2 |
| C1.2 $CO_2$ specification as access and cost criterion | D1 | D2 |
| C1.3 Emitter decision complexity | B1 | A2 |
| C1.4 Total volume commitment reliability | A2 | D1 |
| C2.1 Persistence of industrial activity within the Port of Rotterdam footprint | A2 | D2 |
| C2.2 Compatibility of long-term industrial location dynamics with a fixed CCS backbone | D1 | A2 |
| C2.3 Exposure of the backbone concept to industrial transition, relocation or decline | A2 | C1 |
| C3.1 Commitment and realization of future third-party CCS systems | B1 | C1 |
| C3.2 Interface overcapacity driven by uncertain future integration needs | C1 | D1 |
| C3.3 Design constraints from external system drivers | D1 | C1 |
| C3.4 Execution capacity & schedule realism | C1 | D2 |
| C4.1 Offshore transport integrity and operability | C1 | D2 |
| C4.2 Safety-by-interface dependency | C1 | D2 |
| C4.3 Single-point-of-failure exposure | D1 | D2 |
| C4.4 Off-spec and upset handling procedures | C1 | D2 |
| C5.1 Storage capacity portfolio adequacy | D1 | D2 |
| C5.2 Injectivity and operational flexibility | D1 | D2 |
| C6.1 Tariff formation and cost discovery | D1 | A2 |
| C6.2 FID synchronisation across the value chain | B1 | C1 |
| C6.3 Chain-wide contractual coherence | B1 | C1 |
| C6.4 Strategic optionality and system rigidity | D1 | A2 |
| C6.5 Permitting finality and legal operability | C2 | B2 |
| C6.6 Funding eligibility & auditability (CEF) / cost traceability | A1 | D1 |

**Table 8.3:** Derived Dependencies with Associated Primary and Secondary

| Primary Context | Secondary Context | Count | Generalised Interpretation of Underlying Uncertainty |
|---|---|---|---|
| D1 | D2 | 4 | Uncertainty in the system concept (e.g., design assumptions, configuration options, FOAK elements) translated directly into uncertainty about system-wide performance and operational availability. |
| C1 | D2 | 4 | Uncertainty in chain interfaces (e.g., synchronisation, capacity constraints, technical compatibility, off-spec behaviour) manifested as deliverability constraints and cost impacts across the value chain. |
| B1 | C1 | 3 | Uncertainty in inter-organisational governance (e.g., roles, mandates, contractual coherence) materialised as interface coordination challenges and planning alignment issues. |
| D1 | A2 | 3 | Uncertainty in system scope and future configuration propagated into market/commitment uncertainty due to its implications for cost allocation, pricing logic and willingness-to-pay. |
| A2 | D2 | 2 | Uncertainty in market or price conditions translated into performance pressure, reduced financial buffer, and increased sensitivity of value-chain stability. |
| A2 | D1 | 1 | Uncertainty in future demand or volume profiles required design and scope adjustments to maintain system robustness under varying inflow conditions. |
| A2 | C1 | 1 | Uncertainty in user or market behaviour necessitated interface adjustments to ensure operational continuity and maintainability. |
| D1 | C1 | 1 | Uncertainty in conceptual design choices shaped the coordination load and technical interfacing requirements across system components. |
| B1 | A2 | 1 | Uncertainty in multi-party governance dynamics influenced broader market alignment and commitment behaviour. |
| C1 | D1 | 1 | Uncertainty in future system interconnections required design flexibility and enabled overcapacity or adaptation measures. |
| C2 | B2 | 1 | Uncertainty in project-specific permitting processes influenced political and societal assurance, as well as administrative readiness. |
| A1 | D1 | 1 | Uncertainty in policy, subsidy eligibility, or regulatory conditions required adjustments in system design and compliance documentation. |

**Table 8.4:** Aggregated Paths Enabling Dependencies Porthos

### C1.1: Porthos vs. Aramis

The C1.1 dependency, in the earlier example also functions as how the Aramis case differs form Porhtos. Methodologically, in Aramis, the dependency structure was constructed as a forward-looking and structurally complete mapping exercise. In Porthos, the analysis is retrospective and reconstructive: it examines which enabling dependencies became consequential in practice and whether their stabilising conditions had sufficiently consolidated by the time of FID.

This distinction matters. The Porthos dependency set does not represent a theoretical inventory of potential exposure, but an empirically grounded configuration of enabling mechanisms that proved consequential in practice.

The C1.1 dependency appears as critical in both projects, yet its contextual positioning differs.

C1.1 (Cost Effectiveness of CCS vs Alternatives) also appears as a critical enabling dependency in the Aramis case, but its contextual placement differs between the two projects. In Aramis, C1.1 (Cost Effectiveness of CCS versus Alternatives) primarily originates in A2 (Market & Industrial Commitment), but its secondary landing in A1 (Policy & Regulatory Foundations) follows logically from the project's open access design and earlier maturity. In such a configuration, the attractiveness of CCS is closely tied to emitters' willingness to commit, meaning that shifts in cost effectiveness first manifest as market level hesitation (A2). However, when these shifts affect a broad set of prospective users, the issue no longer concerns individual emitters alone. It starts to challenge the policy assumptions, subsidy logic and collective decarbonisation rationale that underpin CCS as a system-wide mitigation pathway.

In Aramis, cost effectiveness shocks therefore propagate from A2 into A1, because an open access backbone depends on policy endorsement and regulatory support that remain justified only when CCS is competitively positioned against alternative abatement options.

In Porthos, by contrast, C1.1 lands in D2 (Required Value-Chain Performance). Because the project operates as a closed-access chain with fixed tariffs and pre-committed customers at FID, macro-economic shifts do not primarily question policy legitimacy or participation logic. Instead, they compress margins and affect economic deliverability within the system itself. The same dependency is therefore present, but due to system configuration and maturity, the uncertainty becomes governance-relevant at the level of performance (D2) rather than policy foundations (A1).

In short, the dependency is structurally the same, but the system configuration (open vs. closed access) and the project's position in the development lifecycle determine how external uncertainty propagates and where it becomes governance relevant within the MDF context structure.

### Linking Dependencies to FID-Open Risks

Finally, the risk-dependency linkage introduced in Table 8.2 was extended with the contextual pathways derived in Table 8.3 resulting in Table 8.5 and Table 8.6.

Details for this can be found in Appendix D.2 confirming the enabling dependency link to the risk and in Appendix D.3 shows the total of enabling depedencies, with the risk linkage and elaboration of its association. With resulting Tables 8.5 and Table 8.6 - followed from the interviews.

This extension is analytically important because it demonstrates that the selected Porthos risks can be traced back to a limited number of enabling dependencies and associated uncertainty paths. It also establishes the first axis of the later instability interpretation: the risks do not emerge randomly across the project, but repeatedly along a limited number of higher-uncertainty paths.

Section 8.3 therefore provides the empirical basis for the next step in the analysis. It has shown which enabling dependencies underpinned the selected FID-open risks, through which contexts their uncertainty propagated, and which real project difficulties and external developments activated them in practice. The next section interprets these findings in structural terms by asking what they reveal about dependency instability and why these risks materialised at the moment of commitment.

| Risk ID | Code | Enabling Dependency | Primary | Secondary |
|---|---|---|---|---|
| #X | C1.1 | Cost-effectiveness of CCS versus alternatives | A2 | D2 |
| #X | C3.4 | Execution capacity & schedule realism | C1 | D2 |
| #X | C6.1 | Tariff formation & cost discovery | D1 | A2 |
| #X | C6.3 | Chain-wide contractual coherence | B1 | C1 |
| #X | C1.2 | $CO_2$ specification as access & cost | D1 | D2 |
| #X | C4.4 | Off-spec & upset handling procedures | C1 | D2 |
| #X | C4.3 | Single-point-of-failure exposure | D1 | D2 |
| #X | C5.2 | Injectivity & operational flexibility | D1 | D2 |
| #X | C4.1 | Offshore transport integrity & operability | C1 | D2 |
| #X | C6.3 | Chain-wide contractual coherence | B1 | C1 |
| #X | C6.5 | Permitting finality & legal operability | C2 | B2 |
| #X | C1.2 | $CO_2$ specification as access & cost | D1 | D2 |
| #X | C1.4 | Total volume commitment reliability | A2 | D1 |
| #X | C5.2 | Injectivity & operational flexibility | D1 | D2 |
| #X | C6.5 | Permitting finality & legal operability | C2 | B2 |
| #X | C6.3 | Chain-wide contractual coherence | B1 | C1 |
| #X | C6.6 | Funding eligibility & auditability (CEF) | A1 | D1 |

**Table 8.5:** Risks with Underlying Enabling Dependencies

| Primary | Secondary | Path Count Total | Amount of risks in Paths |
|---|---|---|---|
| D1 | D2 | 4 | 2 |
| C1 | D2 | 4 | 3 |
| B1 | C1 | 3 | 3 |
| D1 | A2 | 3 | 1 |
| A2 | D2 | 2 | 1 |
| A2 | D1 | 1 | 1 |
| A2 | C1 | 1 | |
| D1 | C1 | 1 | |
| B1 | A2 | 1 | |
| C1 | D1 | 1 | |
| C2 | B2 | 1 | 2 |
| A1 | D1 | 1 | 1 |

**Table 8.6:** Aggregated Propagation Paths and Risks Associated

## 8.4. Dependency instability and risk manifestation in Porthos

Section 8.3 established that the selected FID-open risks in Porthos can be reconstructed in terms of enabling dependencies and dominant propagation paths. That reconstruction already showed that the risks were not independent events, but were repeatedly underpinned by a limited number of dependencies that carried uncertainty through the project system. What remains is to interpret what this means for project readiness at the moment of commitment.

The central argument of this section is that the realised risks in Porthos did not originate as isolated events, but emerged where enabling dependencies remained insufficiently stabilised at the moment of commitment. Within the MDF, stabilisation refers not to the elimination of risks, but to the reduction of dependency instability. Dependency instability arises where high uncertainty propagation coincides with structural vulnerability. When such dependencies are insufficiently stabilised, both real project events and external disturbances can activate this instability and allow risks to manifest in governance practice.

In this sense, the analytical distinction developed throughout the research becomes explicit. Conventional risk management starts from articulated risks and seeks to mitigate them. A dependency-stabilisation perspective starts one step earlier: it asks whether the enabling mechanisms carrying project viability are sufficiently robust to absorb uncertainty and external disturbance without translating that pressure into realised risks. Porthos provides an empirical case in which this distinction becomes visible.

### 8.4.1. Uncertainty propagation as the first dimension of instability

The first dimension of dependency instability is uncertainty propagation. Table 8.6 shows that the reconstructed Porthos risks are not distributed randomly across governance contexts, but arise along a limited number of dominant propagation paths. In particular, D1 → D2, C1 → D2, and B1 → C1 dominate the reconstructed dependency and risk landscape, while C2 → B2 appears less frequently but proves analytically important.

At this point, an important methodological distinction with the Aramis analysis should be made. In Chapter 7, uncertainty propagation was assessed in greater detail through the full set of sub-dependencies, which allowed for a more granular representation of how uncertainty moved through the system. In Porthos, by contrast, the reconstruction remains at enabling-dependency level. The propagation paths in Table 8.6 therefore represent a coarser but still meaningful approximation of uncertainty propagation within the dependency structure. They show which governance transitions repeatedly underlie the realised risks, even if they do not capture the same level of internal differentiation as in Aramis.

This distinction also explains why the dominant propagation structure is not identical across the two cases. Aramis and Porthos are configured differently as systems, and their enabling dependencies therefore propagate uncertainty differently. Aramis, as a more open and expandable system, produces stronger propagation through implementation and interface-heavy configurations. Porthos, as a more fixed and closed-chain configuration, shows a stronger concentration of uncertainty in transitions such as D1 → D2 and C1 → D2, where concept assumptions and interface conditions translate more directly into performance exposure. The difference is therefore not an inconsistency, but a consequence of project configuration.

Even at enabling level, however, Table 8.6 still shows that the selected risks consistently emerge from a limited number of higher-propagation routes. Risks X and X, for example, are strongly embedded in the D1 → D2 transition, where concept-level assumptions become visible as performance and operability exposure. Risks X, X, and X repeatedly involve C1 → D2, showing how interface and execution uncertainty translate into deliverability and cost pressure. Risks X, X, and X repeatedly involve B1 → C1, indicating that governance and contractual alignment problems materialise through chain interfaces rather than remaining confined to the governance domain itself.

At the same time, the Porthos reconstruction also shows that lower-frequency paths should not be interpreted as unimportant. The clearest example is C2 → B2, which appears in Table 8.6 only in relation to Risks X and X. The enabling dependency behind this path is C6.5 – Permitting finality and legal operability. These risks concern legal and regulatory finality, including permit appeal procedures and the legislative uncertainty surrounding the waste-tax issue, namely that permanently stored $CO_2$ could be classified as taxable waste under Dutch law. In terms of uncertainty propagation alone, this path would not extend far along the x-axis of the dependency-instability matrix, because it is not among the most recurrent or systemically diffused propagation routes.

However, this is precisely what makes the example analytically valuable. It shows that uncertainty propagation and vulnerability must remain distinct dimensions. A dependency may propagate uncertainty less widely through the system, yet still be highly vulnerable. This was also visible in the Aramis analysis where the permit-related dependencies were also not among the highest-uncertainty paths, yet still occupied a structurally exposed position due to their vulnerability. The Porthos case confirms this same principle empirically. The permit appeal led to at least a year of standstill, showing that C6.5 possessed high practical vulnerability even though its propagation profile was less dominant than paths such as D1 → D2 or C1 → D2.

This provides an important clarification of what uncertainty propagation actually captures. High-propagation paths indicate greater systemic interdependence, complexity, and potential for uncertainty to spread across multiple parts of the project configuration. Lower-propagation paths do not imply low importance; rather, they may indicate dependencies whose impact is severe but more concentrated. In the permit case, the consequence was extreme - project standstill and large contractual exposure - yet the uncertainty did not necessarily cascade through every other dependency in the same way as a strongly interdependent concept or interface dependency might. Once legal finality was restored, other project activities could in principle resume. By contrast, dependencies located on dominant propagation paths imply a more deeply interwoven uncertainty structure, in which disturbances are more likely to reverberate across multiple interfaces and performance domains.

The permit case therefore already illustrates the core logic of dependency instability. Proceeding without full stabilisation of such a dependency did not create a “new” risk to be mitigated later. It meant accepting a known combination of vulnerability and uncertainty, which in this case indeed manifested. The broader implication is that the uncertainty axis of dependency instability is empirically visible in Porthos, but it must always be interpreted together with vulnerability. Propagation shows how far and through which routes uncertainty may travel; it does not by itself determine how damaging that uncertainty will be once activated.

### 8.4.2. Structural vulnerability as the second dimension of dependency instability

While Section 8.4.1 established the first dimension of dependency instability through uncertainty propagation, demonstrating the second dimension - structural vulnerability - is analytically more challenging in the Porthos case. In the Aramis analysis, vulnerability could be inferred more directly from the structural configuration of dependencies, particularly where high exposure coincided with insufficiently stabilised development conditions. The most critical cases emerged where enabling dependencies positioned in the outer vulnerability ring were connected to viability dependencies that were themselves also structurally exposed, while simultaneously carrying red development conditions. These 'outer × outer + red' configurations indicated situations in which both the structural dependency and the conditions required to stabilise it remained fragile.

For Porthos, however, such a full analytical reconstruction cannot be replicated. The available data do not allow a comparable modelling of development conditions at the level of sub-dependencies, and therefore do not permit a formal identification of compound high-vulnerability configurations within the dependency matrix. As a result, vulnerability cannot be demonstrated analytically in the same way as in the Aramis case. Instead, it must be established empirically: through observing how specific dependencies behaved once uncertainty materialised in practice.

This also raises a more fundamental question: what would a configuration comparable to an outer × outer + red vulnerability actually imply in practical project terms? Conceptually, such a configuration indicates that the viability and enabling conditions through which the project scope and its strategic success must unfold are themselves structurally fragile. In other words, the elements through which the project scope and its strategic success are realised would each be individually exposed, allowing disturbances to interact with them in ways that could easily destabilise the system. When such dependencies are activated, their impact is therefore likely to extend beyond local disturbances and affect the project's ability to realise its objectives.

A first indication of how vulnerability becomes visible in practice can be observed by contrasting situations in which uncertainty could be absorbed with those in which it generated structural consequences. An illustrative example is the withdrawal of Shell as one of the emitters during the development process. At first sight, the loss of a major customer might suggest a potentially destabilising event for the project. However, this was not the case in commercial terms. The transport capacity of 2.5 MTPA had already been effectively allocated across multiple emitters, and the volume that dropped out could be redistributed relatively quickly among other parties. The withdrawal therefore represented a setback, but it did not undermine the fundamental viability of the chain. In this respect, the system demonstrated resilience: the dependency structure surrounding market commitment was able to absorb the disturbance without producing broader instability.

Yet the same event also revealed that vulnerability may reside elsewhere in the system. Once Shell withdrew, the consequences did not remain limited to lost volume. The technical configuration of the pipeline network had assumed a specific intake route, and the absence of that route created an operational constraint: the pipeline could no longer be cleaned using the originally planned pigging procedure. An additional pipeline segment therefore had to be engineered in order to restore the operability of the system. This illustrates an important feature of vulnerability in sequential infrastructure systems such as Porthos. While the commercial dependency surrounding emitter commitment proved capable of accommodating the disturbance, the technical configuration was more sensitive to the loss of one of its inflow components. The resulting adjustment therefore had a larger structural impact than the loss of the $CO_2$ volume itself.

A comparable pattern can be observed in C3.4 – Execution capacity and schedule realism. The offshore contractor market on which Porthos depended was structurally constrained, with only a small number of contractors capable of performing the required retrofit work. This created a narrow execution margin in which relatively small shifts in project timing could produce disproportionate consequences for price and availability. When the permit appeal delayed the project schedule, Porthos missed its procurement window and was forced to re-enter the contractor market under less favourable conditions, resulting in significantly higher tender prices. Interviewees noted that contractors were well aware of their market position and of the limited number of alternative suppliers available for this type of work. In this case, the dependency showed limited tolerance for changes in project timing, revealing a structural vulnerability in the execution configuration.

Interview evidence further suggested that this situation also reflected a sequencing issue in project governance. A lesson frequently mentioned by interviewees was that a Best and Final Offer (BAFO) should ideally only be requested once key permits have obtained legal finality. In the Porthos case, contractors were effectively placed on standby while permit procedures were still ongoing, which created additional cost exposure once the schedule shifted.

A related form of vulnerability appears in C1.2 – $CO_2$ specification as an access and cost criterion. As a first-of-a-kind system, several assumptions regarding the behaviour of materials and components under dense-phase $CO_2$ conditions had not yet been sufficiently validated during the design phase. During FEED, engineers already recognised uncertainties regarding seal performance, coating behaviour, and impurity management, yet the empirical basis for these assumptions remained limited. Interviews indicate that the practical implications of these uncertainties only became fully visible during integrated testing and commissioning, when issues such as seal interaction with $CO_2$, coating degradation, or filter fouling emerged under real operating conditions. In this case, the dependency's technological characteristics meant that design assumptions remained partly unverified at the moment of commitment, allowing deviations to become visible only once the system was exposed to operational conditions.

Interviewees also suggested that more extensive testing and earlier experimental validation could potentially have reduced part of this uncertainty.

A third example concerns C6.1 – Tariff formation and cost discovery. The tariff structure for $CO_2$ transport was fixed relatively early in the development process in order to align with subsidy frameworks and provide certainty to emitters. While this initially appeared to stabilise the commercial structure of the project, it also reduced the system's flexibility in responding to changing cost conditions. When project costs increased significantly during development, the tariff structure proved largely inflexible, causing cost escalation to translate directly into pressure on the project's financial returns. Structural vulnerability therefore became visible through the rigidity of the commercial configuration once external conditions changed.

Taken together, these examples illustrate a common underlying pattern. Structural vulnerability in the Porthos case becomes visible where enabling dependencies exhibit limited tolerance for disturbances arising from project conditions. Some elements of the project configuration were able to accommodate change without destabilising the system, while others translated relatively small shifts in timing, technological behaviour, or commercial conditions into broader structural consequences.

This suggests that structural vulnerability in the Porthos configuration can be understood as the limited tolerance of certain enabling dependencies to variation in project conditions. Where such tolerance is low, relatively small disturbances in timing, technological behaviour, market conditions, or governance arrangements can translate into disproportionately large project consequences.

This clarifies the second dimension of dependency instability. Whereas uncertainty propagation indicates the routes through which disturbances may travel, structural vulnerability determines how sensitive the dependencies on those routes are to changing conditions. The next section therefore turns to the role of external disturbances as activation triggers, examining how this latent vulnerability became visible in practice.

### 8.4.3. External disturbances as activation triggers

While Section 8.4.2 demonstrated that several enabling dependencies in the Porthos configuration exhibited structural vulnerability, this vulnerability alone does not automatically produce observable risks. Vulnerability may remain latent as long as project conditions remain within the tolerance range of the dependency configuration. Dependency instability becomes especially visible when disturbances interact with these vulnerable dependencies and exceed that tolerance. In such situations, the disturbance does not create a new risk category but activates instability that is already structurally present in the dependency structure.

This mechanism corresponds with the dependency exposure logic discussed in Chapter 7. There it was shown that dependencies positioned in the outer vulnerability ring function as points where external uncertainty can interact with the project system. These dependencies represent locations where the project is structurally exposed to developments in its external environment. Figure 2 illustrates this principle: external disturbances originate outside the project boundary but interact with the system precisely at those locations where dependencies create structural exposure.

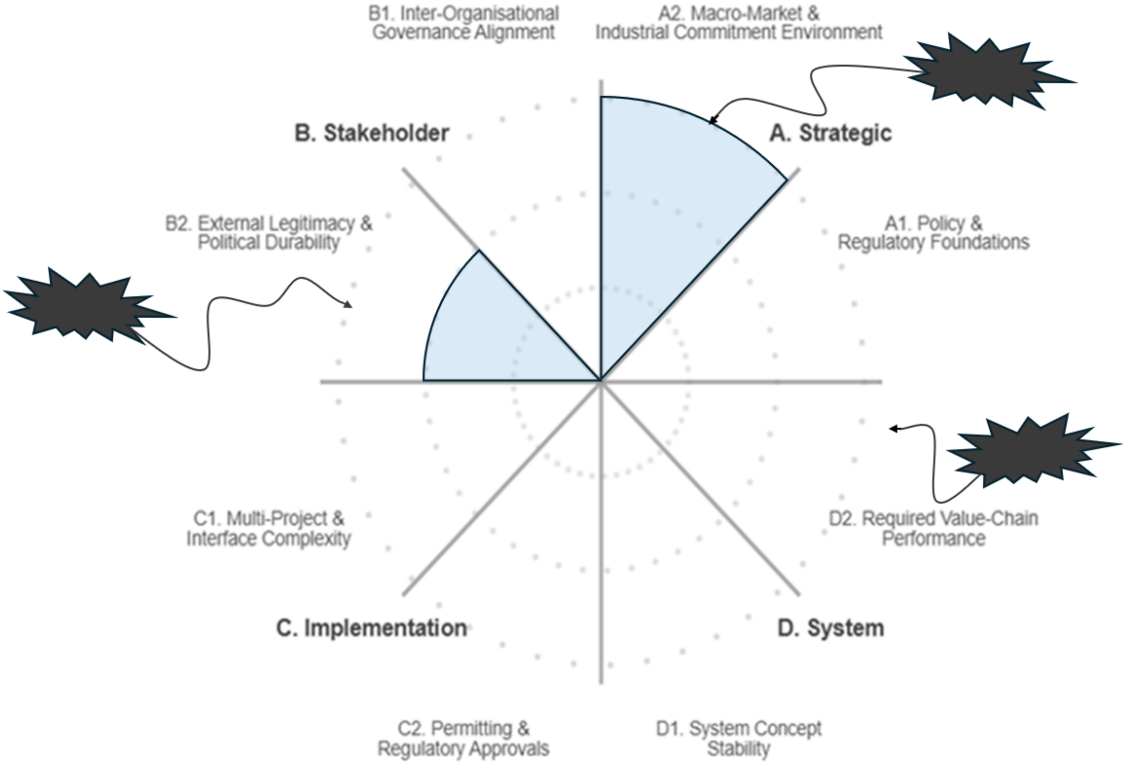


**Figure 2:** External Shocks

The Porthos case provides several empirical examples of such disturbances. These did not originate within the internal governance of the project itself, but arose from changes in the broader institutional, economic, technological, and political environment in which the project developed. Interview material revealed that Porthos encountered a wide range of such disturbances during its development trajectory. These included macro-economic shocks such as the Ukraine war and subsequent inflation, regulatory and institutional uncertainties such as the incorrect Dutch transposition of EU regulation and the RvS permit appeal procedures, market disturbances such as offshore contractor scarcity and volatile gas prices, as well as technological and operational uncertainties related to first-of-a-kind system behaviour. Additional disturbances emerged from strategic shifts among emitters, political legitimacy pressures, uncertainty surrounding subsidy frameworks such as SDE++ and CEF, and interdependencies with parallel projects such as the Aramis development.

Although these disturbances differed substantially in origin and character, they interacted with a relatively limited set of enabling dependencies that had already been identified as structurally sensitive within the project configuration. Regulatory disturbances interacted primarily with the permitting dependency (C6.5), while contractor scarcity and inflation placed pressure on execution capacity (C3.4). Technological exposure during system integration interacted with the $CO_2$ specification dependency (C1.2) and related operational constraints. Macroeconomic developments, in turn, exposed the rigidity of the tariff formation mechanism (C6.1). In each of these cases, the disturbance did not introduce a fundamentally new problem but revealed the limited tolerance of the dependency once project conditions shifted.

From the perspective of the MDF, these disturbances therefore function as activation triggers. They interact with existing dependency structures and make latent instability visible once uncertainty propagation encounters structural vulnerability. In this sense, the disturbances themselves should not be interpreted as the origin of project risks. Rather, they reveal where the dependency configuration lacked sufficient robustness to accommodate variation in project conditions.

Taken together, this suggests that realised risks in Porthos should be understood as the observable outcomes of activated dependency instability: situations in which uncertainty propagation encounters structural vulnerability and becomes visible once triggered by external disturbances.

The final step of the analysis is therefore to examine how these activated instabilities manifested themselves within the project's risk landscape. Section 8.4.4 turns to the risk register of Porthos and shows how the realised risks correspond to the enabling dependencies and propagation paths identified in the preceding sections.

### 8.4.4. Risk manifestation as the observable outcome of dependency instability

The preceding sections established the three elements that together define dependency instability in the MDF framework. Section 8.4.1 showed that uncertainty in the Porthos project propagated through a limited number of recurring dependency paths. Section 8.4.2 demonstrated that several enabling dependencies exhibited structural vulnerability, meaning that they possessed limited tolerance for variation in project conditions. Section 8.4.3 subsequently showed that a wide range of external disturbances interacted with these vulnerable dependencies and activated this latent instability.

The final step is therefore to examine how this activated instability became visible in the project's risk landscape. The key point is that the realised risks in Porthos should not be interpreted as isolated project events. Rather, they were the observable outcomes of dependencies that remained insufficiently stabilised at the moment of commitment. In each case, a structural vulnerability already present in the dependency configuration was activated by a concrete trigger, after which a recognisable risk outcome appeared in governance practice and, in several cases, materialised in execution. Several examples will be given, which are then summarised in Table 8.7.

**C6.5 - Permitting finality & legal operability**

The legal end-test of Porthos lay outside project control: an appeal at the Council of State (RvS) or delays in the national transposition of European regulation could bring the programme to a standstill. This is precisely what occurred. The RvS procedure held Porthos in near standstill for almost a year, while contractors had to remain on standby and a last-minute ministerial guarantee - expensive but effective for only a single day - was required to bridge the delay.

This should be understood as the activation of a structural vulnerability rather than the emergence of a new risk. The appeal functioned as an external shock to which the critical chain of the project was not robustly prepared. Low autonomy and high feedback in relation to permitting meant that legal uncertainty immediately translated into programme-wide consequences. In this context, risk #X and elements of risk #X materialised. Documentary mitigations such as comfort letters or ministerial guarantees could not eliminate the ontological problem of legal finality.

**C3.4 - Execution capacity & schedule realism**

Porthos depended on a highly constrained supplier market for critical offshore retrofit work packages. A shift in the procurement window - such as that caused by the RvS standstill - required retendering in a tighter and more expensive market environment. In practice, prices doubled during re-tendering and risk #X materialised through increased pressure on project returns.

Process mitigations such as tender boards, contingencies, or early procurement proved insufficient because the market itself constituted the bottleneck. The dependency was already structurally vulnerable: the available supplier base was very limited, the timing window was narrow, and contractors were fully aware of their leverage once the project lost its original sequencing advantage. What materialised was therefore not simply a procurement problem, but the exposure of an execution dependency that had not been stabilised against scheduling shocks.

### C1.2 - $CO_2$ specification as access & cost criterion

The critical point in this case is not merely that the system was first-of-a-kind, but that key assumptions about how $CO_2$ quality would interact with materials and components had not yet been sufficiently demonstrated before commitment. During commissioning, latent assumptions became visible: seals absorbed $CO_2$ and could fail, flow-coating might detach, and filter fouling emerged under dense-phase conditions. These were not unexpected in the sense of being unimaginable; they had been recognised as concerns during FEED. But their actual behaviour had not yet been empirically proven in the integrated system.

This matters because risk #X and later risk #X did not arise from a new technical issue emerging out of nowhere. They were the operational manifestation of a dependency whose ontological uncertainty persisted beyond the design phase. Mitigations such as integrated commissioning and open-item strategies can reduce epistemic uncertainty, but they cannot remove the underlying ontological behaviour of $CO_2$-material interaction. The test did not create the vulnerability; it exposed it.

### C5.2 - Injectivity & operational flexibility

Injectivity represents a physical boundary condition: what enters the pipeline must ultimately enter the reservoir. During ramp-up, the actual reservoir response determines whether the chain can operate in dense phase and at the required throughput. Small deviations directly affect availability and turnaround duration, leading to risk #X and, under reduced injectivity scenarios.

This dependency is important because it also provides one of the few examples in which instability was actively reduced before FID. A whipstock is a downhole device used in a wellbore to deflect drilling or access side tracks. In the Porthos case, the presence of a whipstock created uncertainty about whether the intended injection configuration would function as required. Porthos addressed this through whipstock removal prior to FID - an expensive but decisive intervention that eliminated a specific ontological uncertainty and reduced instability in advance. This illustrates the difference between mitigation and stabilisation very clearly: contracts or procedures cannot change the underlying physics, whereas physical intervention can.

### C6.3 — Chain-wide contractual coherence

The contractual architecture across the value chain proved sensitive to shifts in cost and timing. When inflation and the RvS delay propagated through the project, pressure leaked into the interfaces: renegotiations with customers, stricter back-to-back regimes, and tighter consistency requirements for CEF audits.

As a result, risk #X, risk, ..., and risk #X appeared not randomly but precisely where governance misalignment offered little tolerance for disturbances. Legal checks and document coherence may improve procedural alignment but cannot eliminate structural misalignment when external shocks occur.

### C6.1 - Tariff formation  cost discovery

The tariff for $CO_2$ transport was fixed early in the development process through the SDE++ framework. While this increased certainty for emitters, it reduced the flexibility of Porthos to respond to changing cost conditions. When macro-inflation and supply-chain pressures significantly increased CAPEX, the project's IRR profile deteriorated and risk #X became visible.

Reserves and cost control represent cost-side mitigations, but the tariff mechanism itself remained rigid. The outcome was therefore not a new risk but the exposure of a tariff regime that lacked adjustment mechanisms.

### C4.1 - Offshore transport integrity & operability

The offshore component of Porthos was technically and organisationally vulnerable because the $CO_2$ transport system depended on an existing gas platform owned by another operator. During conversion the platform still had to deliver gas to another field, meaning that disturbances in the shared infrastructure could directly affect Porthos.

When additional technical measures proved necessary to maintain safe gas transport, the exact exposure identified in risk #X became visible. Although workaround solutions could partially reduce this risk, the fundamental vulnerability remained: Porthos lacked full control over a system component essential to its own operability.

### C3.1 - Late scope uptake for Aramis (compressor & interface)

Shortly before FID, Porthos incorporated additional scope related to the future Aramis project, including compressor services and a new interface. Because Aramis itself had not yet taken FID, a future dependency was integrated prematurely. This created a configuration with low autonomy, narrow bandwidth late in the lifecycle, and strong feedback through HAZOP and operability interactions.

During execution, this resulted in extensive HAZOP findings, while the later delay of the Aramis FID extended the uncertainty tail. The key point is that the risks that appeared were not separate technical anomalies. They were the surface effects of introducing an immature future interface into a system that was already approaching lock-in. Documentary mitigations such as checklists or coordination meetings cannot stabilise a fluid interface whose upstream project has not yet stabilised itself.

### C1.1 - Cost-effectiveness of CCS versus alternatives

The cost-effectiveness of CCS for emitters depended on the ETS/$CO_2$ price and the transport tariff. When inflation and uncertainty surrounding European climate policy coincided with a fixed tariff regime, the economic balance shifted unfavourably for Porthos. Willingness-to-pay declined and project returns became more marginal, reinforcing the economic context behind risk #X.

Here too, the issue was not that a new market risk suddenly emerged. The commercial structure had been configured around assumptions regarding ETS trajectories, customer willingness-to-pay, and a fixed tariff. When those assumptions shifted, the chosen structure offered little tolerance for rebalancing. The absence of adjustment mechanisms meant that the economic context fed directly into the pressure captured in the risk register.

### Synthesis

Across all cases the same pattern emerges. A pre-existing vulnerability within the dependency configuration was activated by a concrete trigger, after which a recognisable risk outcome followed. In many instances the mitigations that had been foreseen were primarily epistemic - documents, studies, coordination processes - while the underlying problems remained ontological in nature: legal finality, market scarcity, tariff rigidity, FOAK technological behaviour, physical reservoir response, or immature future interface design.

This is the crucial point. The realised risks in Porthos did not originate because the project failed to list enough risks or because individual mitigations were poorly designed. They originated where dependencies had not yet been sufficiently stabilised at the moment of commitment. In MDF terms, risk manifestation occurred where uncertainty propagation coincided with structural vulnerability and was subsequently activated by external disturbance.

That is the crux of the Porthos case. The project did not primarily encounter a series of unrelated external events. It encountered the consequences of entering commitment with dependencies whose instability remained unresolved. Risk management, in this sense, began too late: it responded to symptoms once they surfaced, whereas the relevant question should have been whether the underlying dependencies had reached sufficient stabilisation before FID.

From this perspective, the Porthos case confirms the central argument of this thesis: risks do not originate as isolated events but emerge where uncertainty propagation, structural vulnerability, and external disturbances intersect within the dependency structure of the project. Commitment readiness should therefore not be judged primarily by the apparent closure of risk items, but by the extent to which critical dependencies have been stabilised such that their instability is no longer easily activated under plausible stress.

| Dependency | Vulnerability + Trigger | Manifested Risk | Why mitigations proved insufficient |
|---|---|---|---|
| **C6.5 — Permitting finality & legal operability** | Dependent on judicial rulings and legislation; RvS appeal and incorrect NL implementation of EU rules activated uncertainty. Porthos had no control over timing or outcome. | CONFIDENTIAL. | CONFIDENTIAL |
| **C3.4 — Execution capacity & schedule realism** | Scarce offshore capacity and narrow installation window. RvS delay combined with inflation shifted planning; contractors gained leverage during retendering. | CONFIDENTIAL | CONFIDENTIAL |
| **C1.2 — $CO_2$ specification (FOAK material behaviour)** | Behaviour of seals, coatings and filters under dense-phase $CO_2$ remained unproven; commissioning exposed hidden D1 assumptions. | CONFIDENTIAL | CONFIDENTIAL |
| **C5.2 — Injectivity & operational flexibility** | Injectivity is a hard physical constraint; small deviations in reservoir response propagate directly. Ramp-up and first injection activate sensitivity. | CONFIDENTIAL | CONFIDENTIAL. |
| **C6.3 — Chain-wide contractual coherence** | Chain contracts sensitive to cost and schedule shocks; governance misalignment between JV partners. Inflation and RvS delays strained coherence. | CONFIDENTIAL. | CONFIDENTIAL |
| **C6.1 — Tariff formation & cost discovery** | Tariff fixed early under SDE++; no recalibration mechanism. Inflation raised CAPEX while tariff remained static. | CONFIDENTIAL | CONFIDENTIAL |
| **C4.1 — Offshore transport integrity & operability** | Porthos depended on the TAQA platform for throughput; gas delivery from Q16 had to continue during platform modification. | CONFIDENTIAL | CONFIDENTIAL |
| **C3.1 — Late Aramis scope (compressor + interface)** | Additional subsystem introduced shortly before FID; the Aramis project itself had not yet reached FID, extending uncertainty. | CONFIDENTIAL | CONFIDENTIAL |
| **C1.1 — Cost-effectiveness (tariff vs ETS/$CO_2$ price)** | Economic viability of CCS for emitters depends on ETS price relative to fixed tariff; ETS uncertainty and inflation deteriorated the business case. | CONFIDENTIAL. | CONFIDENTIAL |

**Table 8.7:** Dependency vulnerabilities, manifested risks, and limits of mitigation strategies in the Porthos project.

### 8.4.5. Empirical implications for dependency-aware project design

The preceding reconstruction not only confirms how dependency instability became visible in Porthos; it also clarifies why the MDF is particularly valuable before such instability is translated into governance-visible risks.

The examples discussed in this chapter do more than illustrate how vulnerabilities propagated through Porthos; taken together, they clarify the central insight of this research: many of the situations that later appear as risks originate in the underlying dependency structure of the project. The Megaproject Dependency Framework (MDF) does not attempt to predict specific risk events. Instead, it highlights where combinations of vulnerability and uncertainty may later propagate through the system, regardless of which particular events eventually occur. This perspective is particularly informative during feasibility and conceptual design, where projects must make strategic configuration choices while the concrete risk landscape is still largely indeterminate. At that stage, traditional risk registers often provide limited guidance because the relevant events have not yet materialised. The Porthos case therefore provides several examples that illustrate how examining the dependency structure can offer a valuable counterbalance in early decision-making-helping projects recognise where strategic exposure may already be embedded before passing critical stage-gates.

A first example is the rapid replacement of $CO_2$ volumes when one emitter withdrew. What appears as a fortunate operational outcome is, from a dependency perspective, a deliberate design strength: Porthos had structured its market side with sufficient diversity and over - subscription so that losing a single emitter did not propagate into the transport design, commissioning sequence, or financial viability. If the project had relied heavily on one customer, the same event would almost certainly have generated system-wide disruption. The MDF thus reframes the episode: the success did not lie in managing a risk once it occurred, but in creating a configuration with robust bandwidth so that the disturbance never became a risk in the first place.

A second example concerns the sequencing of the BAFO before permit finality. Interviews reveal that the extremely limited pool of offshore contractors placed Porthos in a structurally vulnerable position; delays outside the project's control (e.g., the RvS appeal) immediately shifted bargaining power to those contractors, inflating prices. The commonly cited lesson - "BAFO only after permits are final" - therefore addresses only the surface of the issue. The MDF points to a more fundamental design question: how should projects organise themselves when strategic success depends on structurally scarce capabilities? If contractor availability constitutes a critical enabling dependency, it may be insufficient to treat procurement as a sequential step that simply follows the project schedule. Instead, dependency awareness raises a more radical possibility: that the project's timeline itself should adapt to contractor availability rather than assuming contractors will adapt to the project. Particularly in FOAK contexts, where both technical and institutional uncertainty remain high, this would imply earlier alignment, deeper collaboration, or alternative contracting structures with key contractors. In that sense, the procurement sequence in Porthos illustrates not merely an operational lesson but a deeper architectural question: whether the project should be designed around the realities of its most critical dependencies rather than assuming those dependencies will conform to the project's intended schedule.

Perhaps the clearest illustration of dependency - aware decision - making is the whipstock intervention. Long before FID, Porthos invested in an expensive and technically challenging operation to verify that a critical injection well could be made suitable for $CO_2$ storage. From a traditional risk-management standpoint, spending millions before commercial commitments are secured appears hard to justify. But from a dependency perspective, the logic is straightforward: the feasibility of the entire storage system rested on this single mechanical assumption. The whipstock removal was thus an act of early ontological stabilisation - resolving a vulnerability that, if left unresolved, would have cascaded across practically every downstream dependency, from injectivity performance to commissioning strategy and contract liability. Rather than mitigating a risk, the project removed the source of instability altogether.

Taken together, these examples demonstrate what it means to approach feasibility and design through the lens of dependency stability. They are not idealised cases of project control; they are moments in which the structure of dependencies becomes visible in practice. The MDF brings these interactions to light at a stage when strategic choices are still adjustable - when alternatives for configuration, sequencing, and stabilisation are still available, and before structural vulnerabilities become baked into the project architecture.

This becomes especially critical at stage-gates. At each gate, a project effectively "locks in" a set of dependencies - technical, contractual, organisational - that will later define its exposure. Once these commitments are made, unresolved vulnerabilities have no choice but to reappear as risks, because the project no longer retains the freedom to reconfigure its underlying dependencies. Examining dependency exposure before passing such gates therefore offers a more reliable basis for commitment readiness than traditional risk registers: it identifies where the project's strategic foundations are not yet stabilised, regardless of which specific events might later occur.

## 8.5. Dependency instability as precursor to risk

The empirical analysis of Porthos suggests that many situations later framed as risks originate much earlier in the project's dependency structure. Rather than emerging as isolated events, risks become visible when unresolved combinations of vulnerability and uncertainty embedded in the project's dependency architecture turn operational under contextual pressure.

This implies that risk manifestation is often less a matter of unforeseeable events than of strategic exposure created during earlier feasibility and design stages. It is in those stages that projects configure, sequence, and attempt to stabilise their critical dependencies. Once such configurations are committed through stage-gate decisions, the project becomes structurally exposed to their behaviour. When contextual conditions subsequently shift, this exposure surfaces in the form of identifiable risks.

The critical governance question is therefore not only how projects respond once risks appear, but how dependency structures are configured before commitment. Examining these structures makes it possible to identify where strategic exposure is already embedded in the project architecture, even when concrete risk events cannot yet be meaningfully articulated.

In this sense, the Megaproject Dependency Framework (MDF) does not replace conventional risk management, but complements it by directing analytical attention to an earlier layer of project uncertainty. By making visible how vulnerability and uncertainty interact within the dependency structure, the framework helps reveal where the strategic foundations of the project remain insufficiently stabilised before these exposures materialise as risks.

Risk, in other words, is not the beginning of project exposure, but its delayed governance-visible expression.

# 9

# Reframing Risk Through Dependency Logic

## 9.1. Risk as a late manifestation of dependency instability

The analyses of Aramis and Porthos support a different interpretation of what “risk” represents in megaproject governance. In conventional project management logic, risks are treated as discrete future events characterised by probability and impact. They are identified, assigned an owner, and addressed through mitigation measures. Within that logic, readiness at decision gates such as FID is commonly framed in terms of whether the most material open risks have been reduced, transferred, or accepted.

The dependency-based analysis developed in this research suggests a different analytical starting point. Rather than treating risks as primary objects of concern, it interprets many risk items as late manifestations of dependency instability. In this view, risk is not the origin of project exposure, but the moment at which underlying structural exposure becomes visible within governance practice.

Dependency instability, as defined in this research, arises where structural vulnerability coincides with uncertainty propagation. Structural vulnerability indicates how exposed a dependency is to disturbances in the external environment. Uncertainty propagation indicates how strongly uncertainty may spread through the dependency network once such disturbances occur. Where these two dimensions coincide, the project carries unstable structural exposure. Risks emerge when that instability becomes operational under contextual pressure.

This interpretation is consistent with the distinction between dependencies and sub-dependencies developed earlier in the thesis. Dependencies carry structural vulnerability because project viability depends on external developments that lie partly outside direct project control. Sub-dependencies do not alter that vulnerability; they provide epistemic signals about whether the external conditions supporting those dependencies are moving in a favourable, uncertain, or unfavourable direction. Dependency instability therefore does not arise simply because uncertainty exists, but because structurally exposed dependencies remain embedded in pathways through which uncertainty can propagate.

The empirical analyses in Chapters 7 and 8 illustrate this mechanism in two different project settings. In Aramis, several enabling dependencies already combined high structural vulnerability with strong uncertainty propagation while the project remained in its pre-FID phase. In Porthos, the risks visible at FID could be traced back to dependencies whose stabilising conditions had not yet sufficiently consolidated. In both cases, the issues later framed as risks were not newly created problems, but governance-visible expressions of instability already present in the project’s dependency structure.

This also explains why conventional risk management often struggles to grasp the real nature of exposure in sustainability-oriented megaprojects. Risk management becomes governance-relevant only once uncertainty can be translated into identifiable events, scenarios, or deviations. But many of the most consequential exposures originate earlier, when projects are still defining their configuration, sequencing, interfaces, and commitment logic. At that stage, uncertainty is not yet well represented as discrete risk items; it is embedded in the dependency architecture of the project itself.

The governance implication is therefore not merely that risks should be mitigated more effectively, but that structurally unstable dependencies should be stabilised before their exposure turns operational. Mitigation responds to articulated events. Stabilisation addresses the dependency conditions that make such events consequential in the first place. The permitting experience in Porthos illustrates this distinction clearly: guarantees and comfort arrangements could mitigate some anticipated consequences, but only legal finality could stabilise the dependency itself.

Risk, in this sense, is not the root object of governance but a signal that dependency instability has become active. The real object of governance is therefore the stability of the dependency configuration on which the project's strategy depends.

## 9.2. Why dependency instability matters before risk articulation

Disturbances do not affect project systems uniformly. External shocks - such as macroeconomic shifts, regulatory reinterpretations, geopolitical events, or strategic repositioning by actors in the value chain - enter the project primarily through viability-level conditions. Because these conditions lie partly outside direct project control, their disturbance does not immediately produce identifiable risk events. Instead, the resulting pressure travels through the project's dependency structure.

Within this structure, it is not the project activities themselves that determine how disturbances propagate, but the dependencies on which those activities rely. Dependencies constitute the structural points where the project architecture connects to its external environment. When external assumptions change, uncertainty propagates through these dependencies. The degree to which this propagation affects the project depends on the structural vulnerability of those dependencies. Where vulnerability is high, disturbances are more likely to translate into operational consequences.

This relationship can be interpreted as a structural analogue to conventional risk logic. In risk management, the significance of a risk event is typically described through two dimensions: likelihood and impact. Within the dependency perspective developed in this research, these dimensions correspond to structural properties of the dependency architecture itself. Vulnerability functions as the precursor to likelihood, indicating how exposed a dependency is to external disturbance. Uncertainty propagation functions as the precursor to impact, determining how strongly a disturbance may spread through the project system once it enters. Dependency instability therefore represents a structural condition that precedes the emergence of identifiable risks.

Figure 1 illustrates this relationship. The dependency instability matrix positions dependencies according to their structural vulnerability and the degree to which uncertainty can propagate through them. Dependencies located in the lower-left quadrant I combine low vulnerability with limited uncertainty propagation and therefore represent relatively stable structural elements. Quadrant II represents situations in which uncertainty may propagate but where vulnerability remains limited, allowing disturbances to be absorbed without systemic consequences. Quadrant III contains dependencies that are structurally vulnerable but exposed to relatively limited uncertainty propagation, requiring careful monitoring but not necessarily systemic intervention.

The most critical region lies in Quadrant IV, where high vulnerability coincides with strong uncertainty propagation. Dependencies positioned in this region represent structurally unstable conditions. When external shocks interact with such dependencies, disturbances are more likely to propagate through the project architecture and become operationally visible as risks.

This dynamic becomes particularly pronounced in projects that contain first-of-a-kind (FOAK) elements or form part of emerging value-chain infrastructures, such as the CCS systems analysed in this thesis. In such contexts, both vulnerability and uncertainty propagation are difficult to eliminate. FOAK technologies introduce technical and operational uncertainties that cannot yet be fully stabilised, while value-chain configurations create dependencies on external actors whose commitments evolve sequentially. As a result, projects must operate within a structural environment in which both vulnerability and uncertainty propagation remain inherently present.

Under such conditions, external shocks cannot be treated simply as isolated risk events. Macroeconomic volatility, regulatory reinterpretation, technological learning effects, and strategic repositioning by actors in the value chain are inherent features of long-duration capital-intensive systems. The relevant governance question therefore shifts from asking whether shocks will occur to understanding where they will interact with structurally vulnerable dependencies and how uncertainty will propagate through the project architecture.

From this perspective, dependency instability provides an earlier analytical lens than conventional risk registers. Rather than describing disturbances once they have already materialised as risks, it reveals where the project's dependency structure creates conditions under which disturbances are likely to propagate. In that sense, dependency instability does not replace risk management; it identifies the structural exposure from which risk behaviour later emerges.

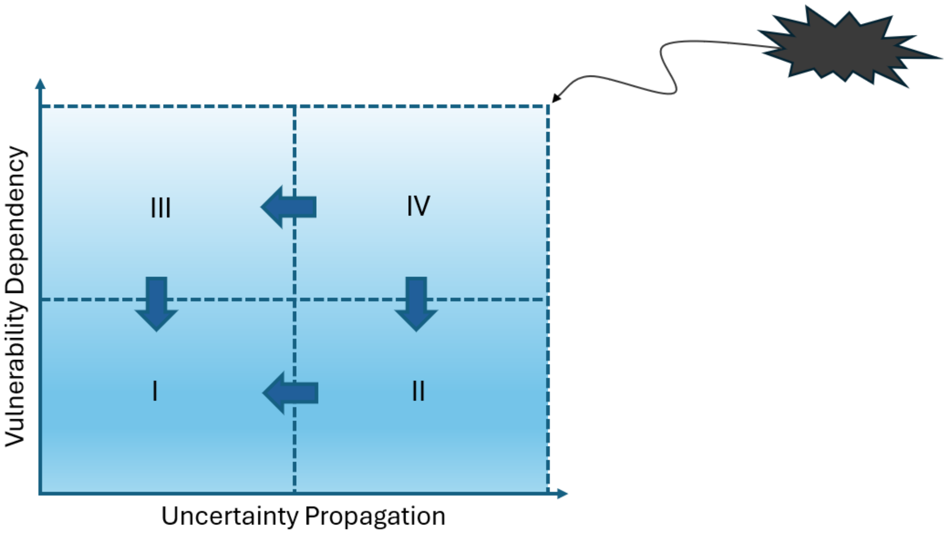


**Figure 1:** Dependency Instability with an extenal shock impacting the point of greatest instability

## 9.3. Implications for project governance

The preceding analysis suggests that megaproject governance may need to shift its focus from managing risks to stabilising the dependency structures that generate those risks. If risks represent the governance-visible manifestation of deeper dependency instability, then managing uncertainty cannot rely solely on the mitigation of articulated risks. Instead, projects are encouraged to pay greater attention to the stability of the dependency architecture on which their strategy relies.

This perspective implies several shifts in how projects approach uncertainty in practice.

### 9.3.1. Governance shifts in dependency-oriented projects

- **Shift 1: From mitigation to stabilisation**

Traditional project governance focuses on mitigating identified risks once they have been articulated in risk registers. Mitigation strategies typically aim to reduce the likelihood or impact of specific events through contingency planning, contractual safeguards, or additional buffers.

The dependency perspective shifts attention one level earlier. Rather than primarily responding to articulated risks, projects should aim to stabilise the dependencies through which uncertainty may propagate. Stabilisation involves resolving or reducing structural vulnerability before disturbances translate into operational consequences. This implies intervening in the dependency architecture itself rather than merely preparing responses to possible disturbance events.

The whipstock intervention in Porthos illustrates how certain forms of uncertainty cannot initially be articulated as risk. At the time, the key question was not how to mitigate a known issue, but whether whipstock removal would work at all. As one interviewee described: "…already two years before FID, a first whipstock was removed… to see, does it work at all?"

Importantly, this intervention involved the removal of a whipstock in a single well as a test case, rather than addressing all wells at once. The objective was not to resolve the issue across the system, but to determine whether the underlying mechanism was feasible in principle.

Prior to this intervention, the uncertainty could not be meaningfully expressed in terms of likelihood or impact. Once the feasibility was demonstrated, however, the remaining whipstocks could be assessed more concretely and articulated as risks.

In this sense, the intervention did not reduce a predefined risk, but enabled its formulation. It transformed an initially undefined uncertainty into a set of identifiable and manageable risks. Within the dependency perspective, this can be understood as the stabilisation of a critical dependency: not because uncertainty disappeared, but because it became sufficiently bounded to be governed within the project.

Within the dependency instability matrix (Figure 1), such interventions correspond to a movement away from Quadrant IV - where high vulnerability and strong uncertainty propagation coincide - toward Quadrant II, where structural vulnerability is reduced even though uncertainty may still be present.

- **Shift 2: From planning to alignment**

Much of project planning is still presented as if external actors and boundary conditions will align with the project's internal schedule. In practice, however, projects are structurally dependent on decision-making, capacity, and regulatory processes beyond their control.

Dependency awareness therefore goes beyond merely identifying such dependencies. It involves recognising that they actively shape the rhythm and sequencing of the project. This has implications for how planning is approached: while risks may be incorporated through probabilistic scheduling, deeper forms of uncertainty require a more adaptive and scenario-oriented approach.

In practice, this means that sequencing must be aligned with the maturation of critical dependencies, that external conditions may need to be actively influenced (for example through governance and contractual arrangements), and that the project configuration should be designed to reduce vulnerability to external disturbances.

The procurement experience in Porthos illustrates this dynamic. Contractor engagement was initially sequenced in parallel with the permitting process. When permit finality was delayed, the contractual arrangements with contractors lost their validity as well. The issue was therefore not simply a delay in regulatory approval, but the propagation of uncertainty from one dependency into another.

The commonly stated lesson learned - "BAFO after permits" - captures the practical response. The dependency instability matrix clarifies the structural logic behind this lesson: where commitments are sequentially linked, uncertainty propagation must be reduced before downstream dependencies are activated.

In matrix terms (Figure 1), such adjustments correspond to a horizontal movement from Quadrant IV toward Quadrant III, reducing the degree to which uncertainty can cascade through the project architecture.

- **Shift 3: From control to robustness**

Traditional risk management often focuses on anticipating and controlling discrete disturbance events. The dependency perspective does not prescribe how project structures should be designed, but provides a clearer lens to understand how structural configurations influence the system's ability to absorb disturbances. Particularly, in situations where the likelihood and nature of disturbances cannot yet be meaningfully expressed in probabilistic terms.

The Porthos market configuration illustrates this principle. When one emitter withdrew from the project, the disturbance did not destabilise the broader system because the project had not been structurally dependent on a single participant. The design of the participation structure provided sufficient diversity to absorb the disturbance without systemic consequences.

The dependency perspective does not prescribe how project structures should be designed, but provides a clearer lens to understand how structural configurations influence the system's ability to absorb disturbances. By making explicit how dependencies shape both vulnerability and uncertainty propagation, it offers a more grounded basis for designing project structures that can remain robust under conditions that cannot yet be fully articulated as risks.

In matrix terms, such strategies reduce structural vulnerability by decreasing the project's reliance on any single dependency. This corresponds to a vertical movement from Quadrant III toward Quadrant I, where both vulnerability and uncertainty propagation remain limited.

### 9.3.2. Underlying shift: from execution control to structural control

Underlying these governance shifts is a more fundamental reorientation in how projects are conceived and controlled. Conventional project governance is structured around three primary dimensions: time, budget, and benefits. These dimensions implicitly define how projects are organised and controlled. Schedules are treated as targets to be met, budgets as constraints to be managed, and benefits as outcomes to be delivered.

The dependency perspective developed in this research suggests a different interpretation. Rather than treating time as a fixed target, it should be understood as a function of the conditions required for the project to succeed. In other words, the question is not how quickly a project can be delivered, but whether the dependencies on which it relies have reached a sufficient level of stability to support that delivery.

This implies a shift from target-driven control to condition-driven design. Time is no longer the primary driver of sequencing; instead, sequencing follows the maturation of critical dependencies. The central governance question becomes whether the dependency structure has been configured such that strategic success remains achievable under changing external conditions.

This reframing provides a different basis for stage-gate decision-making. Instead of evaluating readiness primarily in terms of schedule, budget, or risk, stage-gates become moments at which the stability of critical dependencies is assessed. A project is ready to pass a gate not because it is "on track", but because the dependencies on which it relies have reached a level of maturity that makes further commitment structurally defensible.

In this view, sequencing is no longer driven solely by internal planning logic, but by the readiness of critical dependencies. Benefits are not simply targets to be realised, but outcomes that depend on the alignment between external conditions and the project's internal configuration.

Applied to megaprojects, this perspective leads to a different central question: not whether the project is on schedule, but whether it is being configured in such a way that its strategic objectives remain achievable under the conditions on which it depends. Time, budget, and benefits remain important reference points, but their interpretation becomes conditional on the extent to which the underlying dependencies have been sufficiently stabilised to support those objectives.

### 9.3.3. Implications for governance practice

Together, these governance shifts illustrate that dependency instability is not merely an analytical concept, but a design variable that can be actively shaped through project governance. Reducing uncertainty propagation stabilises how disturbances travel through the project system, while reducing structural vulnerability decreases the likelihood that such disturbances become consequential in the first place.

At the same time, this perspective does not imply that existing governance approaches based on mitigation, planning, and control should be replaced. These approaches remain essential, particularly where uncertainty can be articulated in the form of identifiable risks. However, the findings of this study indicate that not all relevant uncertainty can be captured in this way. In projects such as Aramis and Porthos, a significant part of the exposure is rooted in conditions that are not yet fully knowable in terms of what will happen, when, or how.

The contribution of the dependency perspective therefore lies in complementing, rather than replacing, existing governance logic. It provides a way to engage with forms of uncertainty that precede risk articulation and cannot yet be meaningfully expressed in probabilistic terms. In such cases, the nature of uncertainty is ontological rather than event-based: what may happen, how it may unfold, and when it may occur are not yet sufficiently defined to be captured as discrete risks. Risk articulation assumes that uncertainty can be translated into identifiable events with estimable likelihood and impact. However, the cases studied here show that a substantial part of project uncertainty does not meet this condition. Instead, uncertainty exists as a broader, less defined field of potential developments that cannot yet be reduced to risk statements without oversimplification. In these situations, the problem is not that risks are insufficiently analysed, but that the underlying uncertainty cannot yet be articulated as risk at all.

The dependency perspective addresses this gap by shifting attention away from the articulation of events toward the structure through which such uncertainty may materialise. By focusing on dependencies, projects gain a way to engage with uncertainty that remains undefined in terms of specific outcomes, but is already present in the conditions on which the project relies. This provides a basis for targeted stabilisation of vulnerable elements, alignment of interdependent commitments, and the development of more robust project configurations. In doing so, projects gain a more structured understanding of where uncertainty resides and how it may affect the conditions for strategic success.

In this sense, the shift is subtle but consequential. It is not primarily about managing uncertainty more precisely, but about structuring the project in a way that makes it more robust to uncertainty that cannot yet be fully understood. Rather than relying solely on risk articulation, projects are encouraged to organise their design and decision-making around the dependencies that underpin strategic success.

The dependency instability matrix (Figure 1) therefore functions not only as an analytical representation of exposure, but also as a practical governance instrument. It helps identify where intervention is required and how projects can deliberately strengthen those dependencies that are most critical to maintaining the viability of the project under uncertainty.

## 9.4. Rethinking stage-gates and FID readiness

### Stage-gate framing

Stage-gates in megaproject development mark moments at which projects commit to strategic configurations, contractual structures, and irreversible financial investments. Among these decisions, the Final Investment Decision (FID) represents the most prominent example. Conventionally, readiness for such commitments is evaluated through risk registers, quantified contingencies, contractual closure, and formal gate reviews. Residual risks may remain but are considered acceptable if their expected impact lies within defined tolerance levels.

From a dependency perspective, stage-gates represent more than administrative decision points. They are moments at which the dependency architecture of the project becomes effectively locked in. Technical configurations, contractual relationships, sequencing logic, and governance structures become fixed in ways that are difficult to reconfigure afterwards. As a result, the stability of the dependency structure at the moment of commitment becomes critical for how the project will respond to future disturbances.

The findings from Aramis and Porthos suggest that this framing may be incomplete when considered in isolation, as it does not fully capture how projects operate under conditions of pervasive and only partially articulable uncertainty, nor provide sufficient insight into how the underlying dependency structure shapes their ability to remain viable when such uncertainty materialises.

### From risk closure to vulnerability stabilisation

Both cases show that risks present at FID are rarely random. They cluster around structurally central enabling dependencies and propagate through the dependency architecture of the project. When such risks later materialise, this is often interpreted as misfortune, escalation, or inadequate mitigation. Within the MDF logic, however, a different explanation becomes plausible: the project crossed the gate while the stabilising development conditions of critical dependencies could have been further strengthened.

A risk register evaluates scenario-based disturbances, while the MDF provides a complementary perspective by examining the structural buffering capacity of dependencies that carry ontological uncertainty.

Where the risk register asks:

- “Are the listed threats manageable?”

The MDF asks:

- “Are the structurally vulnerable dependencies stabilised enough to prevent cascading propagation under plausible stress?”

Importantly, this perspective does not imply that all dependencies can, or should, be fully stabilised prior to commitment. In practice, certain dependencies remain outside the direct control, capability, or mandate of the project organisation. For example, market formation mechanisms or subsidy regimes may depend on external actors such as governments or regulators.

The value of a dependency perspective therefore lies not only in enabling stabilisation, but in making explicit where such stabilisation is limited or contingent. It supports a more informed governance discussion by clarifying which dependencies remain structurally exposed, to what extent this exposure is understood, and how it is reflected in the overall risk profile of the project.

In this sense, the objective is not to eliminate all high-uncertainty dependencies, but to make deliberate choices about them. While individual projects may justifiably carry elevated levels of uncertainty, governance decisions - particularly at portfolio level - benefit from understanding how such exposures accumulate and interact across multiple projects.

### Reframing what FID makes visible

Reframed through dependency logic, the relevant FID question becomes: Are the critical enabling mechanisms sufficiently stabilised to withstand foreseeable shifts in their external conditions without destabilising the project?

Dependencies carry ontological vulnerability. That vulnerability does not disappear before FID and cannot be eliminated through mitigation plans. What can be developed, however, are the sub-dependencies - the stabilising development conditions that determine whether that vulnerability remains latent or becomes active under contextual pressure.

From this perspective, assessing FID readiness involves examining whether:

- The development conditions attached to dominant dependencies are sufficiently matured;
- Epistemic uncertainty has narrowed to a defensible bandwidth rather than remaining unbounded; and,
- Exposure under plausible context variation has been structurally examined and not merely documented.

The purpose is not to eliminate vulnerability - which is impossible - but to ensure that it remains latent under foreseeable stress.

Viewed through a dependency lens, a stage-gate can be understood not only as a test of risk identification, but as an assessment of whether the underlying dependency structure, that tests whether the dependency structure on which the project relies has reached a level of structural consolidation that allows it to absorb plausible external disturbances without cascading failure.

### The Porthos illustration

Porthos provides a clear demonstration of how insufficiently stabilised dependencies can later manifest as operational issues despite formal risk acceptance. At FID, several exposures remained open in the risk register. These exposures were part of the risk-reward profile on which the investment decision was based, supported by mitigation strategies and an understanding of their potential impacts at the time.

Yet execution showed that many of these exposures were not “risks” in the event-based sense; they can instead be interpreted as expressions of underlying enabling dependencies whose stabilising conditions could have been further developed.

For example: $CO_2$ specification and material-behaviour uncertainty had not been fully stabilised for the full operating window. What appeared later as corrosivity and seal failure was the activation of an already present dependency level vulnerability.

Although the permits had been granted, their legal finality remained uncertain due to the pending Council of State appeal. This uncertainty placed the project in a prolonged state of legal limbo: construction could not begin, contractors had to remain on standby, and Porthos was forced to obtain a one - day ministerial guarantee to cover potential downside. In this sense, the unresolved vulnerability was already shaping project behaviour long before the ruling itself.

Tariff rigidity was fixed before macroeconomic shocks materialised. Without adaptive clauses, the system lacked bandwidth to absorb price shifts.

These were not failures of foresight. They were cases in which dependency vulnerability remained active at FID.

Interviews confirmed this governance gap. Practitioners explicitly recognised that structural uncertainty was difficult to capture. Quotes about the difficulties of dealing and processing uncertainties in dealy management are depicted in the Appendix D.5.1.

Even in the Porthos’s top risk, uncertainty itself became the risk. This explicitly highlights the linguistic and conceptual limits of classical risk frameworks.

### Structural certainty vs probabilistic certainty

Risk registers provide a structured way to evaluate scenario-based disturbances and can be effectively integrated into planning through probabilistic scheduling. However, as discussed in Section 9.3, not all forms of uncertainty can be articulated in this way. In particular, deeper forms of uncertainty — where the nature, timing, or mechanism of potential disturbances remains undefined — require a more adaptive and scenario-oriented approach.

A dependency perspective complements this by focusing not only on potential events, but on the structures through which uncertainty may propagate. It helps to reveal how dependencies shape the rhythm and sequencing of the project, and how disturbances may travel across interdependent elements once they occur.

From this perspective, the challenges observed in Porthos can be more clearly understood. Rather than attributing these outcomes to shortcomings in anticipation or preparation, they can be interpreted as situations in which dependency vulnerability remained present at the moment of FID. What later appeared as isolated issues can instead be seen as the manifestation of structural conditions whose stabilising development had not yet fully matured.

**Implications for commitment governance**

Consistent with the broader argument in Section 9.3, this reframing does not imply that existing risk-based gate criteria should be replaced, nor that FID should await perfect information. Rather, it suggests that megaprojects should shift more from: evaluating risks, to evaluating the stability of dependencies.

In other words:

- Risk acceptance is insufficient if vulnerability remains undampened.
- Open risks are not inherently problematic; open vulnerabilities are.
- The decisive question is not whether exposure is known, but whether it is buffered.

Across both Aramis and Porthos, uncertainty followed the architecture of dependencies. Nothing new emerged; what is already structurally present becomes operational.

FID readiness is therefore best understood as a test of structural consolidation rather than as the formal approval of a risk inventory.

Stage-gates thus become moments at which this perspective can be applied explicitly, allowing decision-makers to assess not only the visibility of risk, but also the underlying stability of the dependency structure on which the project relies.

## 9.5. Scope and applicability of the dependency framework

The relevance of this dependency-based perspective becomes particularly visible in projects where the dependency structure itself is dense, externally conditioned, and difficult to stabilise before commitment decisions are taken.

Both Aramis and Porthos operate in an unusually complex institutional and industrial environment. They are multi-actor CCS megaprojects embedded in evolving regulatory regimes, exposed to volatile market conditions, and dependent on coordinated commitment from multiple independent organisations across a value chain. In such contexts, uncertainty is not merely technological or financial; it is systemic, relational, and structurally interdependent.

Uncertainty exists in all megaprojects, which is precisely why the Megaproject Uncertainty Framework (MUF) was developed. However, the two CCS cases investigated here exhibit an exceptionally high density of external viability conditions and cross organisational enabling dependencies. Their success depends not only on internal execution performance but on the alignment of emitters, transport operations, storage operators, regulatory authorities, subsidy regimes, and long term policy stability. These projects are not isolated investments; they are coordinated value chain configurations.

This dependency density makes structural vulnerability more consequential. The interaction between viability assumptions and enabling mechanisms is amplified because multiple semi-autonomous actors must commit under shared but only partly controllable conditions. As a result, dependencies - such as permit irrevocability, $CO_2$ specification setting and control, system readiness, tariff formation, and value chain performance - carry disproportionate systemic weight.

The interviews conducted for Aramis and Porthos show that, in such environments, uncertainty repeatedly exceeds the descriptive capacity of classical risk management. Practitioners acknowledged that, here two examples (All quotes about the difficulties of capturing uncertainties are depicted in Appendix D.5.1).

- “Not every uncertainty can be clearly defined as a risk. If you only talk about risks, you may miss the major uncertainties.”
- “We did have FOAK listed in the risk register... but we didn’t know how those risks would manifest themselves.”

In addition to these explicit statements, the interviews also revealed numerous situations in which risk items failed to accurately reflect the true nature of the exposure. Mitigation strategies reinforce this interpretation. Many of the so-called ‘risk mitigations’ in both projects were, in practice, attempts to compensate for insufficiently stabilised development conditions. These interventions provided temporary assurance but did not fundamentally stabilise the underlying dependencies.

In both cases, teams struggled to articulate these structural uncertainties within the vocabulary of probability-impact risk management. Interviewees described uncertainty as:

- “difficult to define,”
- “greater than any individual risk,”
- “47-dimensional rather than 2D,”
- and in Porthos’s top risk: “uncertainty itself became the risk.”

These observations reveal a recurring governance gap: practitioners experience structural uncertainty, but risk management frameworks tend to translate this into discrete, event-based items. As a result, foundational dependency exposures remain implicit until they activate under pressure.

The Megaproject Dependency Framework (MDF) is therefore most applicable to sustainability oriented, capital intensive megaprojects that:

- Contain first-of-a-kind (FOAK) technologies or organisational arrangements whose operational behaviour cannot yet be fully stabilised;
- Operate across a value chains;
- Require synchronised commitment from multiple independent organisations; and
- Depend on the continued alignment of externally conditioned viability assumptions with internal project architecture.

Its relevance increases as projects move away from single-owner, internally controllable systems toward multi-actor configurations in which strategic success depends on cross-boundary coordination. In this sense, Aramis and Porthos function as high intensity cases that make dependency behaviour observable in ways that more self contained projects would not.

The framework’s transferability to other megaprojects is therefore likely to be strongest in contexts characterised by high levels of epistemic uncertainty, particularly where FOAK elements are present. However, this does not mean the framework is only applicable to such projects. Rather, its added value becomes more pronounced as uncertainty increases - particularly when key elements of the project cannot yet be fully defined or predicted.

From the MDF perspective, megaproject risk behaviour is not primarily event-driven, but structurally conditioned. The governance challenge therefore shifts from cataloguing risks to understanding and stabilising the dependencies through which uncertainty may propagate. The Megaproject Dependency Framework provides both an analytical lens to interpret such exposure and a governance language to engage with it before it manifests as operational risk.

## 9.6. Applying the MDF in practice: a generalised approach

The practical value of the MDF lies in its ability to provide structured insight at stages of a project where uncertainty is still difficult to articulate and cannot yet be meaningfully expressed as risk. In projects characterised by first-of-a-kind elements, sequential value-chain formation, or strong dependence on external conditions, many of the most consequential exposures cannot yet be articulated as discrete risks. The MDF adds value precisely in this phase, by making visible where strategic exposure is already being created within the project architecture before it becomes formally expressed in risk registers.

Applying the framework starts with clarifying the project’s viability logic: what constitutes strategic success, and which external conditions must hold for that success to remain plausible. Without this reference point, uncertainty remains abstract and cannot be meaningfully assessed.

A critical next step is to translate this abstract uncertainty into concrete points of interaction within the project. Projects are exposed to uncertainty through external conditions, but these conditions only become actionable where they intersect with the mechanisms through which the project is configured - such as design choices, contractual arrangements, sequencing logic, and system interfaces. The essential requirement is therefore not the categorisation itself, but the establishment of a clear link between the conditions on which project success depends and the mechanisms through which these conditions are operationalised.

This translation makes uncertainty tangible. It reveals where uncertainty enters the project, how it connects to the project's configuration, and where it may begin to affect the ability to achieve strategic objectives.

In this study, this was operationalised by mapping uncertainty propagation through the governance contexts - strategic, stakeholder, implementation, and system - which describe project uncertainty at a system level, derived from Figure 3, and specified by the specific sub-contexts (see Figure 3). This made it possible to position where uncertainty enters the project and how it propagates through governance contexts across the dependency structure. Structural vulnerability was assessed using autonomy, bandwidth, and feedback strength, while the development of underlying conditions was evaluated through the development status of their stabilising conditions (green, orange, red). Together, these assessments provided a structured representation of both the sources and behaviour of uncertainty, forming the basis for the dependency instability matrix.

This assessment of structural significance can be generalised into two fundamental questions: how exposed the project is to disturbances affecting these elements, and how strongly such disturbances would propagate through the system. The specific method used to answer these questions may vary, but the underlying logic remains the same: projects must form a judgement on both the vulnerability of their configuration and the extent to which uncertainty can spread across it.

By combining vulnerability and uncertainty, projects can identify where instability is most likely to concentrate and which elements are therefore structurally most critical to stabilise before further commitment. This shifts attention from managing isolated risk events to strengthening the structure through which such events would unfold.

In this sense, the MDF functions as an early interpretive governance lens. It helps projects understand where uncertainty enters, how it may propagate, and how strongly it threatens the conditions on which strategic success depends. Its contribution lies not in replacing risk management, but in preceding it: providing a structured way to identify which parts of the project must stabilise if later risk behaviour is to remain bounded.

Building on this understanding, the framework supports targeted evaluation. It enables projects to assess:

- Which dependencies are most likely to be affected by changes in the external environment, and therefore warrant prioritised stabilisation;
- Where sequencing assumptions may need to be reconsidered;
- Which design or governance choices create disproportionate exposure, and;
- Whether the overall project configuration is sufficiently robust to support commitment.

The objective is not to eliminate uncertainty, but to configure the project such that uncertainty is less likely to propagate into system-wide consequences. In doing so, the framework supports a more robust project design by focusing attention on the structural conditions that determine whether strategic success remains attainable under changing external circumstances.

# 10

# Conclusions and Reflection

This research shows that stage-gates in sustainability-oriented megaprojects cannot be adequately understood through risk-based evaluation alone. While risk registers capture identifiable events, a substantial part of project uncertainty exists in a more fundamental form: arising from external conditions that cannot yet be fully articulated, and becoming relevant to the project through the dependencies that link these conditions to its viability

By developing and applying the Megaproject Dependency Framework (MDF), this research demonstrates that project exposure is structurally conditioned by the interaction between vulnerability and uncertainty within the project's dependency architecture. In this sense, the MDF functions as a precursor to risk articulation: it reveals how external uncertainty connects to the project through dependencies, how it may propagate across the dependency structure, and which elements are therefore most likely to become points of exposure before they can be expressed as discrete risks.

From this perspective, readiness at stage-gates is not determined by the closure of risks alone, but also by the extent to which critical dependencies have been sufficiently stabilised to carry uncertainty into subsequent project phases. This does not replace risk-based evaluation, but complements it by providing insight into forms of uncertainty that precede and shape risk behaviour.

After developing the conceptual framework and applying it to the Aramis CCS project - and subsequently testing its consistency through comparative reflection on the Porthos CCS project - this final chapter brings the findings together and clarifies the overall contribution of the research.

The remainder of this chapter synthesises these insights. Section 10.1 answers the research questions by integrating the conceptual and empirical findings. Section 10.2 summarises the main empirical patterns identified in the case studies. Section 10.3 positions the contribution of the MDF within existing literature. Section 10.4 reflects on implications for governance practice, and Section 10.5 discusses limitations and directions for future research.

Together, this chapter synthesises the main arguments of the research and situates its contribution within broader discussions on uncertainty, governance, and commitment in sustainability-oriented megaprojects.

## 10.1. Answering the research questions

This section addresses the main research question and its sub-questions by integrating the conceptual development of the MDF with the empirical findings from the Aramis and Porthos cases.

**Research questions**

The central question guiding this research is:

*How can project readiness at stage-gates be assessed by analysing vulnerability and robustness in fundamental dependencies of sustainability-oriented megaprojects?*

This question is supported by five sub-questions addressing:

- The nature of fundamental dependencies;
- The origin of structural vulnerability;
- The development of robustness;
- The role of stage-gates in relation to exposure, and;
- The relationship between dependency structures and observed risk patterns.

**Integrated answer**

The findings show that conventional risk-based approaches do not fully capture the type of uncertainty that characterises sustainability-oriented megaprojects. In particular, they struggle to engage with external uncertainties that are difficult to define in advance, yet have a decisive influence on project viability.

These uncertainties do not initially appear as discrete risks. They become relevant to the project through the dependencies that link external conditions - such as regulatory decisions, market formation, and system performance - to the project's internal configuration.

From this perspective, assessing readiness at a stage-gate - most notably at FID - requires more than evaluating whether identified risks are acceptable. It requires understanding how uncertainty is structurally connected to the project and how it may propagate through its dependency architecture.

The MDF provides a way to make this explicit.

It shows that:

- Dependencies define where and how external uncertainty connects to the project;
- Vulnerability expresses how sensitive these dependencies are to external variation and how strongly disturbances may propagate;
- Robustness reflects the extent to which the underlying conditions of these dependencies have been stabilised; and
- Propagation patterns reveal how uncertainty is likely to move through the system once activated.

Within this logic, stage-gates do not represent moments at which uncertainty is resolved. They represent moments at which a particular configuration of vulnerability and uncertainty is formalised and carried forward.

FID, in particular, fixes how the project will absorb or transmit uncertainty under changing external conditions.

A key implication of this perspective is that the MDF provides a way to engage with forms of uncertainty that initially exist as ontological uncertainty - uncertainty that cannot yet be clearly defined or articulated. By analysing how such uncertainty connects to the project through dependencies, the framework enables a progressive structuring of this uncertainty into more interpretable forms. In this sense, dependencies function as a translation mechanism through which initially undefined uncertainty becomes increasingly bounded and, eventually, expressible in epistemic terms.

External shocks play a critical role in this process. They do not introduce fundamentally new uncertainty, but activate vulnerability that is already present within the dependency structure. What appears as sudden risk emergence can therefore be understood as the operationalisation of pre-existing dependency instability under changing external conditions.

Project readiness should therefore be understood as a structural condition: the extent to which critical dependencies are sufficiently stabilised such that their inherent vulnerability is likely to remain latent under plausible external variation.

Observed risk patterns in both cases can be explained by the underlying configuration of dependency instability. Dependencies characterised by high structural vulnerability combined with strong uncertainty propagation form the primary loci of exposure. Then, external shocks do not create new risks, but trigger their activation, causing uncertainty to materialise as observable risks.

This indicates that risk articulation reflects underlying dependency structures rather than forming an independent layer of analysis.

### Concluding statement

Taken together, the research shows that assessing stage-gate readiness requires complementing risk-based evaluation with a structural perspective on how uncertainty connects to, and potentially impacts, the project. The MDF provides this perspective by making visible which dependencies carry the greatest exposure and how their development shapes the conditions under which commitment becomes defensible.

## 10.2. Key findings

The analyses of the Aramis and Porthos CCS projects reveal a consistent structural logic in how uncertainty manifests and develops in sustainability-oriented megaprojects. Rather than being dispersed or event-driven, project exposure is organised through the dependency architecture of the project. The findings can be synthesised into four interrelated insights.

### 1. Exposure concentrates in dependencies with high instability

The analysis suggests that project exposure is not diffusely distributed across the dependency landscape, but tends to concentrate in a limited set of dependencies characterised by high dependency instability - that is, where structural vulnerability coincides with strong uncertainty propagation.

These dependencies emerge as critical within the analysis. Their position in the system - low autonomy, limited bandwidth, and strong feedback effects - makes them particularly sensitive to external variation while simultaneously enabling disturbances to propagate across multiple parts of the project.

This pattern is not directly visible without structured analysis, but becomes apparent when dependencies are examined in terms of vulnerability and propagation. Dependencies identified as highly unstable consistently correspond to areas where high-impact risks are later observed. This suggests that dependency instability provides an indication of where exposure is most likely to materialise under changing external conditions.

### 2. Risks are manifestations of dependency instability

Observed risk patterns in both cases are not independent phenomena, but expressions of underlying dependency instability.

Dependencies combining high structural vulnerability with strong uncertainty propagation capacity are consistently the points at which uncertainty becomes operational. External shocks do not introduce new uncertainty, but activate instability that is already present within the dependency structure.

The analysis indicates that many articulated risks correspond to underlying external dependencies. What is framed as a discrete risk event often reflects a dependency on external conditions that has become sufficiently specified to be expressed in risk terms.

By linking risks to their underlying dependencies and associated development conditions, the framework enables a more detailed decomposition of these risks. Epistemic development conditions reveal how uncertainty is structured across multiple interacting elements, rather than being confined to a single event description. This exposes dimensions of uncertainty that are difficult to capture within conventional risk statements.

This demonstrates that risk articulation often compresses a broader field of uncertainty into simplified event-based representations. The MDF extends this by showing how such risks originate, how they are structurally connected, and why they recur.

Risk articulation can therefore be interpreted as a surface-level representation of underlying dependency structures, particularly where uncertainty is difficult to articulate due to its ambiguous or poorly defined nature, rather than a fully independent analytical layer.

### 3. Dependencies structure the transition from ontological to epistemic uncertainty

A key finding is that certain forms of uncertainty in these projects are difficult to clearly define or articulate within conventional frameworks. The dependency perspective provides a more structured way to interpret and engage with this uncertainty by analysing how it connects to the project through dependencies.

Dependencies act as interfaces through which external uncertainty becomes connected to specific elements of the project. Through this interaction, uncertainty becomes more bounded, more localised, and eventually expressible in epistemic terms.

Interventions aimed at stabilising dependencies - such as testing feasibility or clarifying system conditions - do not eliminate uncertainty, but transform it into a form that can be articulated, assessed, and governed.

The framework therefore provides a way to move from undefined uncertainty toward structured understanding, without assuming that uncertainty must first be reducible to risk.

### 4. Stage-gates formalise dependency configurations and enable governance shifts

Stage-gates - most notably FID - do not eliminate uncertainty, but formalise a specific configuration of dependency instability.

At the moment of commitment, the project fixes how vulnerability and robustness are embedded in its technical, contractual, and organisational structure. Dependencies that remain only partially stabilised continue to carry exposure into subsequent phases.

Whether this exposure remains latent or becomes operational depends on how these dependencies interact with future external conditions. External shocks play a decisive role in this process by activating vulnerability that was already present at the time of commitment.

This has direct implications for project governance. If risks are the manifestation of underlying dependency instability, then managing uncertainty cannot rely solely on mitigating articulated risks. Instead, governance must shift its focus toward stabilising the dependencies that generate those risks.

This implies a move:

- From mitigation to stabilisation;
- From planning based on assumed alignment to more active alignment of dependencies;
- From controlling events to designing for structural robustness, and;
- From execution control to structural control.

In this sense, stage-gates become moments at which not only risks are evaluated, but the stability of the dependency architecture itself is assessed. Readiness is therefore less about whether the project is “on track”, and more about whether its dependency structure is sufficiently stabilised to support commitment under changing external conditions.

### Synthesis

Taken together, these findings show that uncertainty in sustainability-oriented megaprojects is not primarily event-driven, but structurally conditioned.

- Exposure concentrates in dependencies with high instability;

- High-impact risks correspond to these zones of instability;
- Risks are manifestations of dependency-level exposure rather than independent events;
- Dependencies structure how uncertainty becomes interpretable and actionable; and
- Stage-gates formalise how this exposure is carried forward and governed.

The central implication is that project behaviour under uncertainty is determined less by individual risk events than by the configuration of the dependency structure through which uncertainty propagates.

## 10.3. Theoretical contribution

This research contributes to megaproject governance and uncertainty research by introducing a dependency-based perspective on how uncertainty connects to projects, becomes structured, and ultimately manifests in governance-visible forms such as risks.

Rather than treating uncertainty primarily as a set of discrete events, the research shows that a substantial part of project exposure originates in external conditions that are difficult to fully specify in advance, but become relevant through the dependencies that link these conditions to the project's viability. In doing so, the study shifts the analytical focus from risk events to the structures through which uncertainty is translated, propagated, and made actionable.

### Dependencies as interfaces between external uncertainty and project structure

A first contribution lies in conceptualising dependencies not merely as constraints, but as interfaces through which external uncertainty becomes connected to the project. This reframing moves beyond viewing uncertainty as something that "enters" the project through events, and instead shows how it is structurally coupled to the project through specific conditions and mechanisms.

This provides a more stable analytical basis for examining uncertainty across project phases, as it allows uncertainty to be analysed even when it cannot yet be expressed as a risk.

### From uncertainty to risk: a structuring perspective

A second contribution is the explicit articulation of how uncertainty becomes structured through dependencies.

The findings show that certain forms of uncertainty are difficult to clearly define or articulate within conventional risk-based frameworks. By tracing how this uncertainty connects to specific dependencies and their underlying development conditions, the MDF provides a way to progressively structure and interpret it.

In this sense, the framework does not replace risk articulation, but precedes it. It enables a transition from less defined forms of uncertainty toward more bounded, interpretable forms, without assuming that uncertainty must first be reducible to discrete events.

### Dependency instability as a structural concept of exposure

A third contribution is the introduction of dependency instability as a way to conceptualise structural exposure.

By combining structural vulnerability with uncertainty propagation, the framework identifies configurations in which uncertainty is both likely to affect the project and capable of spreading across it. This moves beyond treating vulnerability as a static property and instead links it to the dynamic behaviour of uncertainty within the system.

Dependency instability thereby provides a way to interpret why certain parts of a project repeatedly become critical under changing external conditions, and why high-impact risks tend to emerge in these areas.

### Repositioning risk within the project system

A fourth contribution concerns the role of risk within megaproject analysis.

The research shows that risks can often be understood as partial and simplified representations of underlying dependency structures, particularly in cases where uncertainty is difficult to fully articulate. By linking risks back to dependencies and their development conditions, the framework enables a more detailed interpretation of how uncertainty is structured and where it originates.

This does not invalidate risk-based approaches, but situates them within a broader analytical context. Risks are not treated as the primary source of uncertainty, but as governance-visible expressions of how uncertainty interacts with the project's dependency architecture.

### Propagation as the missing link in uncertainty analysis

A fifth contribution is the explicit incorporation of propagation into the analysis of project uncertainty.

Where existing frameworks often focus on identifying sources or domains of uncertainty, the MDF shows how disturbances move through the project via dependency structures. This makes it possible to understand not only where uncertainty is located, but how it can affect other parts of the system once activated.

By linking structural configuration to dynamic behaviour, the framework connects static descriptions of uncertainty with its operational consequences in project execution.

### Bridging uncertainty theory and stage-gate governance

Taken together, these contributions establish a link between theories of uncertainty and the governance of megaprojects at stage-gates.

The findings suggest that stage-gate decision-making cannot be understood solely in terms of risk mitigation or performance readiness. Instead, it involves assessing how uncertainty is structurally embedded in the project and how it may develop after commitment.

The MDF contributes to this by providing a way to analyse how exposure is carried forward through the dependency structure, how it may become activated under external change, and how its development can be influenced through the stabilisation of critical dependencies.

In this way, the research does not propose an alternative to existing governance approaches, but extends them. It provides a complementary perspective that allows projects to engage with forms of uncertainty that precede risk articulation and cannot yet be fully captured within probabilistic models.

## 10.4. Implications for project governance practice

The findings of this research suggest that the governance of sustainability-oriented megaprojects requires a shift in how uncertainty is interpreted and acted upon. Rather than being primarily event-driven, project exposure is structurally conditioned by the dependency architecture through which uncertainty connects to and propagates within the project.

This does not imply replacing existing governance practices such as risk management, planning, or control. These remain essential where uncertainty can be articulated in terms of identifiable risks. However, the findings indicate that a substantial part of project exposure originates in forms of uncertainty that precede risk articulation. Engaging with this type of uncertainty requires a complementary perspective focused on the stability of dependencies.

### From risk mitigation to dependency stabilisation

Traditional governance approaches focus on mitigating risks once they have been articulated. The findings of this research indicate that this addresses uncertainty relatively late in its development.

A dependency-based perspective shifts attention one level earlier: toward stabilising the dependencies through which uncertainty may propagate. This involves strengthening the underlying conditions that determine whether uncertainty remains bounded or becomes operational.

In practice, this means that interventions are not only aimed at reducing the likelihood or impact of events, but at reducing structural vulnerability and limiting propagation potential within the dependency architecture.

### From planning based on assumptions to alignment of dependencies

Project planning often assumes that external conditions and actors will align with internally defined schedules. The findings show that this assumption is structurally fragile in projects that depend on multiple external actors and evolving conditions.

A dependency perspective implies that sequencing should be more aligned with the maturation of critical dependencies, rather than imposed independently of them. External conditions do not simply constrain the project; they shape its feasible trajectory.

This shifts the role of planning from coordinating internal activities to actively aligning interdependent commitments across the project's external environment.

### From control of events to design for robustness

Conventional governance emphasises anticipating and controlling discrete disturbance events. The findings suggest that this approach is limited in situations where uncertainty cannot yet be meaningfully articulated.

A dependency-based view shifts attention toward the structural capacity of the project to absorb disturbances. Rather than attempting to control all possible events, governance focuses on designing configurations that are less sensitive to external variation and less prone to cascading effects.

This involves reducing reliance on single critical dependencies, increasing flexibility in system design, and limiting the pathways through which disturbances can propagate.

### From execution control to structural control

Underlying these shifts is a more fundamental reorientation of project governance.

Traditional governance is organised around execution control - managing time, budget, and performance against predefined targets. The findings of this research suggest that, in projects characterised by high dependency complexity, these dimensions are conditional on the stability of underlying dependencies.

A dependency-based perspective reframes governance around structural control: the deliberate configuration and stabilisation of the dependency architecture that determines whether project objectives remain achievable under changing external conditions.

From this perspective:

- Time becomes a function of dependency maturation rather than a fixed target;
- Sequencing follows the readiness of critical dependencies; and
- Stage-gates become moments at which the stability of the dependency structure is assessed.

A project is therefore not ready because it is “on schedule”, but because its critical dependencies have reached a level of stability that makes further commitment structurally defensible.

### Monitoring and decision-making beyond risk registers

Risk registers remain important instruments for tracking governance-visible uncertainty. However, the findings show that they provide only a partial representation of underlying exposure.

A dependency-informed approach complements this by monitoring the development of critical dependencies and their associated conditions. This provides insight into how uncertainty is evolving before it becomes articulated as risk.

At the level of governance, this perspective supports more focused discussions by identifying a limited set of structurally critical dependencies that carry disproportionate exposure. Rather than aggregating uncertainty into risk metrics alone, governance can consider how exposure is configured and how it may evolve under changing conditions.

### Implications for stage-gates and FID

Stage-gates should be understood not only as decision points based on risk evaluation, but as moments at which a specific configuration of dependency instability is formalised.

Crossing a gate does not eliminate uncertainty, but fixes how it is embedded in contracts, system design, and organisational arrangements. Dependencies that remain only partially stabilised continue to carry exposure into subsequent phases, where external changes may activate them.

Assessing readiness at FID therefore involves examining whether critical dependencies are sufficiently stabilised such that their vulnerability is likely to remain latent under foreseeable external variation.

### Summary

Taken together, these implications point to a shift in governance logic.

- From mitigating risks to stabilising dependencies;
- From planning based on assumed alignment to actively aligning dependencies;
- From controlling events to designing for robustness; and
- From execution control to structural control.

This shift is not about replacing existing practices, but about extending them. It enables governance to engage with forms of uncertainty that cannot yet be fully articulated.

## 10.5. Limitations and directions for future research

This research has three main limitations that define the scope of its findings: empirical context, interpretative modelling, and data perspective.

### Empirical scope

The analysis is based on two CCS megaprojects within the Dutch institutional context. These cases provide a relevant setting for studying dependency-driven uncertainty, but share similar regulatory and policy conditions.

The findings should therefore be understood as context-sensitive. At the same time, the underlying logic of dependency instability is not inherently sector-specific. Testing the framework in other sectors and institutional environments is required to assess its broader applicability.

### Interpretative modelling

The dependency mapping and analysis are inherently interpretative. The identification of dependencies, their classification, and the assessment of propagation paths required analytical judgement.

This introduces a degree of subjectivity, particularly in:

- Linking sub-dependencies to enabling mechanisms;
- Connecting enabling dependencies to higher-level viability conditions, and;
- Assigning dependencies to governance contexts.

These ambiguities reflect not only modelling choices, but also the fact that real-world uncertainty does not align neatly with analytical categories.

Further methodological refinement - especially in standardising dependency classification and linkage criteria - would improve consistency and comparability across cases.

### Data and governance perspective

The analysis is based on risk registers, governance documents, and expert interpretation. As a result, it reflects how uncertainty becomes visible within formal governance structures.

This implies that forms of uncertainty that remain unarticulated or informally managed may be under-represented. The findings should therefore be understood as an analysis of governance-visible uncertainty, rather than a complete representation of all underlying dynamics.

### Directions for future research

Two priorities follow directly from these limitations.

First, applying the MDF in other sectors and project types - particularly beyond CCS and the Dutch context - would test whether similar patterns of dependency instability and propagation emerge under different conditions.

Second, further development of the modelling approach is needed. Refining how dependencies are identified, linked, and evaluated would strengthen the robustness of the framework and support its use in comparative analysis.

# References

Adger, W. N. (2006). Vulnerability. *Global Environmental Change*, *16*(3), 268–281. https://doi.org/10.1016/j.gloenvcha.2006.02.006

Ashkanani, S. H., & Kerbache, L. (2023). Enhanced megaproject management systems in the lng industry. *Energy Reports*, *9*, 1062–1076. https://doi.org/10.1016/j.egyr.2022.12.030

Bates, A. (2018). After the dust settles—exploring common causes and cures of mega-project failures. https://www.troutman.com/insights/after-the-dustsettles-exploring-common-causes-and-cures-of-mega-project-failures.html

Benjaminsen, H., & Sørnes, J. O. (2025). Epistemic uncertainty in megaprojects. *Production Planning & Control*. https://doi.org/10.1080/09537287.2025.2548560

Damayanti, R. W., Hartono, B., & Wijaya, A. R. (2021). Clarifying megaproject complexity in developing countries. *International Journal of Engineering Business Management*, *13*. https://doi.org/10.1177/18479790211027414

Danilovic, M., & Sandkull, B. (2005). The use of dependence structure matrix in managing uncertainty. *International Journal of Project Management*, *23*, 193–203. https://doi.org/10.1016/j.ijproman.2004.11.001

Denicol, J., Davies, A., & Krystallis, I. (2020). What are the causes and cures of poor megaproject performance? *Project Management Journal*, *51*(3), 328–345. https://doi.org/10.1177/8756972819896113

Dewulf, A., & Biesbroek, R. (2018). Nine lives of uncertainty in decision-making. *Policy and Society*, *37*(4), 441–458. https://doi.org/10.1080/14494035.2018.1504484

DiMaggio, P. J., & Powell, W. W. (1983). The iron cage revisited. *American Sociological Review*, *48*(2), 147–160.

Esposito, G., & Terlizzi, A. (2023). Governing wickedness in megaprojects. *Policy and Society*. https://doi.org/10.1093/polsoc/puad002

Floricel, S., Michela, J., & Piperca, S. (2016). Complexity and uncertainty-reduction strategies. *International Journal of Project Management*, *34*, 1360–1383. https://doi.org/10.1016/j.ijproman.2015.11.007

Flyvbjerg, B. (2006). From nobel prize to project management. *Project Management Journal*, *37*. https://doi.org/10.1177/875697280603700302

Flyvbjerg, B. (2014). What you should know about megaprojects. *Project Management Journal*, *45*(2), 6–19.

Flyvbjerg, B. (2017). *The oxford handbook of megaproject management*. Oxford University Press.

Flyvbjerg, B., Bruzelius, N., & Rothengatter, W. (2003). *Megaprojects and risk*. Cambridge University Press.

Flyvbjerg, B., & Gardner, D. (2023). *How big things get done*. Crown Currency.

Geraldi, J., Maylor, H., & Williams, T. (2011). Now, let's make it really complex. *International Journal of Operations & Production Management*, *31*(9), 966–990. https://doi.org/10.1108/01443571111165848

Guan, L., Abbasi, A., & Ryan, M. (2021). Simulation-based risk interdependency network model. *Decision Support Systems*, *148*, 113602. https://doi.org/10.1016/j.dss.2021.113602

Jensen, C., Johansson, S., & Löfström, M. (2006). Project relationships. *International Journal of Project Management*, *24*, 4–12. https://doi.org/10.1016/j.ijproman.2005.06.004

Klinke, A., & Renn, O. (2002). A new approach to risk evaluation and management. *Risk Analysis*, *22*, 1071–1094.

Lempert, R. J. (2003). *Shaping the next one hundred years*.

Lenfle, S., & Loch, C. (2010). Lost roots. *California Management Review*, *53*(1), 32–55. https://doi.org/10.1525/cmr.2010.53.1.32

Li, Y., Wang, M., Locatelli, G., & Zhang, Y. (2024). Navigating the future of megaproject sustainability. *International Journal of Managing Projects in Business*, *17*(3), 533–561. https://doi.org/10.1108/IJMPB-02-2024-0027

McDermott, T. A., Nadolski, M., & Clifford, M. (2024). The future of megaproject management. *Proceedings of the International Annual Conference of the American Society for Engineering Management*, 1–11.

Miller, R., & Lessard, D. (2000). *The strategic management of large engineering projects*. MIT Press.

Nachbagauer, A. G., & Schirl-Boeck, I. (2019). Managing the unexpected in megaprojects. *International Journal of Managing Projects in Business*, *12*(3), 694–715. https://doi.org/10.1108/IJMPB-08-2018-0169

Perminova, O., Gustafsson, M., & Wikström, K. (2008). Defining uncertainty in projects. *International Journal of Project Management*, *26*(1), 73–79.

Power, M. (2007). *Organized uncertainty*. Oxford University Press.

Qazi, A., Dikmen, I., & Birgonul, T. (2020). Prioritization of interdependent uncertainties. *International Journal of Managing Projects in Business*. https://doi.org/10.1108/IJMPB-10-2019-0253

Qazi, A., Quigley, J., Dickson, A., & Kirytopoulos, K. (2016). Project complexity and risk management. *International Journal of Project Management*, *34*, 1183–1198. https://doi.org/10.1016/j.ijproman.2016.05.008

Rose, A., Shrimali, G., & Halttunen, K. (2025). A framework for assessing and managing dependencies. *iScience*, *28*(7).

Ross, J., & Staw, B. (1986). Expo 86: An escalation prototype. *Administrative Science Quarterly*, *31*(2), 274–297.

Samset, K., & Volden, G. H. (2016). Front-end definition of projects. *International Journal of Project Management*, *34*(2), 297–313.

Stevens, R. (2011). *Engineering mega-systems*. Auerbach Publications. https://doi.org/10.1201/EBK1420076660

Too, E. G., & Weaver, P. (2014). The management of project management. *International Journal of Project Management*, *32*(8), 1382–1394.

Turner, B. e. a. (2003). A framework for vulnerability analysis. *PNAS*, *100*, 8074–8079. https://doi.org/10.1073/pnas.1231335100

Urruty, N., Tailliez-Lefebvre, D., & Huyghe, C. (2016). Stability, robustness, vulnerability and resilience. *Agronomy for Sustainable Development*, *36*. https://doi.org/10.1007/s13593-015-0347-5

Van Marrewijk, A. e. a. (2008). Managing public-private megaprojects. *International Journal of Project Management*, *26*(6), 591–600.

Williams, T. e. a. (2019). The front-end of projects. *Production Planning & Control*, *30*(14), 1137–1169. https://doi.org/10.1080/09537287.2019.1594429

# A

# Model Application

## A.1. Autonomy, Bandwidth, Feedback Strength - Definitions & Scoring

### Autonomy

| **Autonomy** | |
|---|---|
| *The extent to which a dependency remains viable without continued alignment from external systems.* | |
| **Score** | **Criteria** |
| Low | Dependency is highly contingent on continued alignment by external actors or systems; no effective buffers or fallback arrangements exist. Loss of alignment directly undermines viability. |
| Medium | Dependency relies on multiple external systems or actors with partial redundancy; limited buffering or substitution is possible, but sustained misalignment would compromise viability. |
| High | Dependency can remain viable despite significant external misalignment due to internal buffers, alternative arrangements, or diversified reliance on external systems. |

**Table A.1:** Operationalisation of Autonomy in dependency assessment.

### Bandwidth

| **Bandwidth** | |
|---|---|
| **Score** | **Criteria** |
| Low | Very narrow tolerance window; minor deviations from assumed conditions push the dependency outside its viable corridor. |
| High | Broad tolerance window; dependency can absorb substantial variation without undermining overall project viability. |

**Table A.2:** Operationalisation of Bandwidth in dependency assessment.

### Feedback Strength

| **Feedback Strength** | |
|---|---|
| *The extent to which changes in a dependency propagate and amplify effects across the value chain.* | |
| **Score** | **Criteria** |
| Low | Effects remain largely localized; changes have limited knock-on effects and do not meaningfully propagate across contexts or value-chain segments. |
| Medium | Changes propagate to adjacent dependencies or contexts, but effects are partially dampened and do not escalate system-wide. |
| High | Small changes trigger strong propagation and amplification across multiple value-chain segments or contexts, potentially affecting overall project viability. |

**Table A.3:** Operationalisation of Feedback Strength in dependency assessment.

## A.2. Vulnerability classification

The combination of autonomy, bandwidth and feedback strength together determines the vulnerability. Ranging from outer, which is the most vulnerable assesement, then middle, and inner; which is more resilient.

| Autonomy | Bandwidth | Feedback Strength | Ring |
|---|---|---|---|
| Low | Low | Low | **OUTER** |
| Low | Low | Medium | **OUTER** |
| Low | Low | High | **OUTER** |
| Low | High | Low | **MIDDLE** |
| Low | High | Medium | **MIDDLE** |
| Low | High | High | **OUTER** |
| Medium | Low | Low | **MIDDLE** |
| Medium | Low | Medium | **MIDDLE** |
| Medium | Low | High | **OUTER** |
| Medium | High | Low | **MIDDLE** |
| Medium | High | Medium | **MIDDLE** |
| Medium | High | High | **MIDDLE** |
| High | Low | Low | **MIDDLE** |
| High | Low | Medium | **MIDDLE** |
| High | Low | High | **OUTER** |
| High | High | Low | **INNER** |
| High | High | Medium | **INNER** |
| High | High | High | **MIDDLE** |

**Table A.4:** Ring classification as a function of autonomy, bandwidth, and feedback strength.

## A.3. Sub-dependency development indicator - Status

| Status | Interpretation |
|---|---|
| **Green** | External developments are demonstrably and consistently moving towards the required condition, with visible progress and a degree of consolidation (e.g., commitments, institutional safeguards, or repeatable signals). |
| **Orange** | External developments show promising signs, but are still conditional, fragile, or dependent on additional decisions, timing, or external circumstances. |
| **Red** | External developments are moving away from the required condition, or are developing in a direction that structurally undermines its realization. Alternatively, developments are not yet sufficiently formed or visible to indicate direction or status. |

**Table A.5:** Interpretation of development status (Green–Orange–Red) for sub-dependency condition.

## A.4. Megaproject Dependency Framework - Contexts

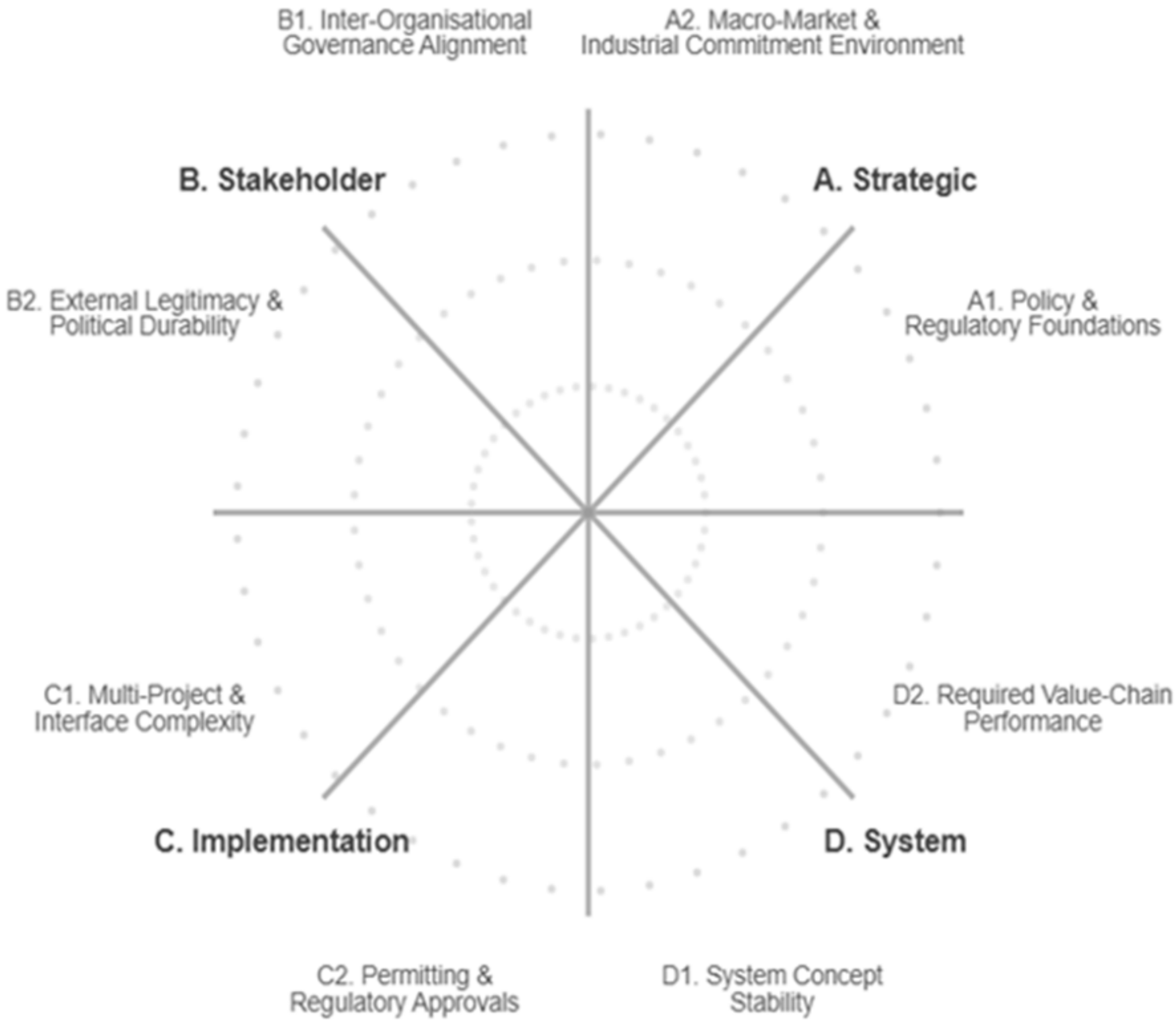


**Figure 1:** Contexts & Sub-Contexts

**(A) Strategic: Uncertainty about policy, market and investment decisions that determine the project framework.**

- A1 – Policy & Regulatory Foundations:

  Uncertainty about legislation, subsidies and policy frameworks with which the project must comply.
- A2 – Macro-Market & Industrial Commitment:

  Uncertainty about investment willingness, volumes and commitment of market parties.

**(B) Stakeholder: Uncertainty about cooperation, governance and public support.**

- B1 – Inter-Organisational Governance Alignment:

  Uncertainty about roles, responsibilities and decision-making between parties.
- B2 – External Legitimacy & Political Durability:

  Uncertainty about political and public support for the project.

**(C) Implementation: Uncertainty about coordination, interfaces and permits during implementation.**

- C1 – Multi-Project & Interface Complexity:

  Uncertainty due to coordination between sub-projects, systems and schedules.
- C2 – Permitting & Regulatory Approvals:

  Uncertainty about obtaining and retaining the necessary permits.

**(D) System: Uncertainty about the design, system concept and required performance.**

- D1 – System Concept Stability:

  Uncertainty about exactly what kind of system will be built and what design principles will apply.
- D2 – Required Value-Chain Performance:

  Uncertainty about the performance required to make the project viable

**Context placement criteria:**

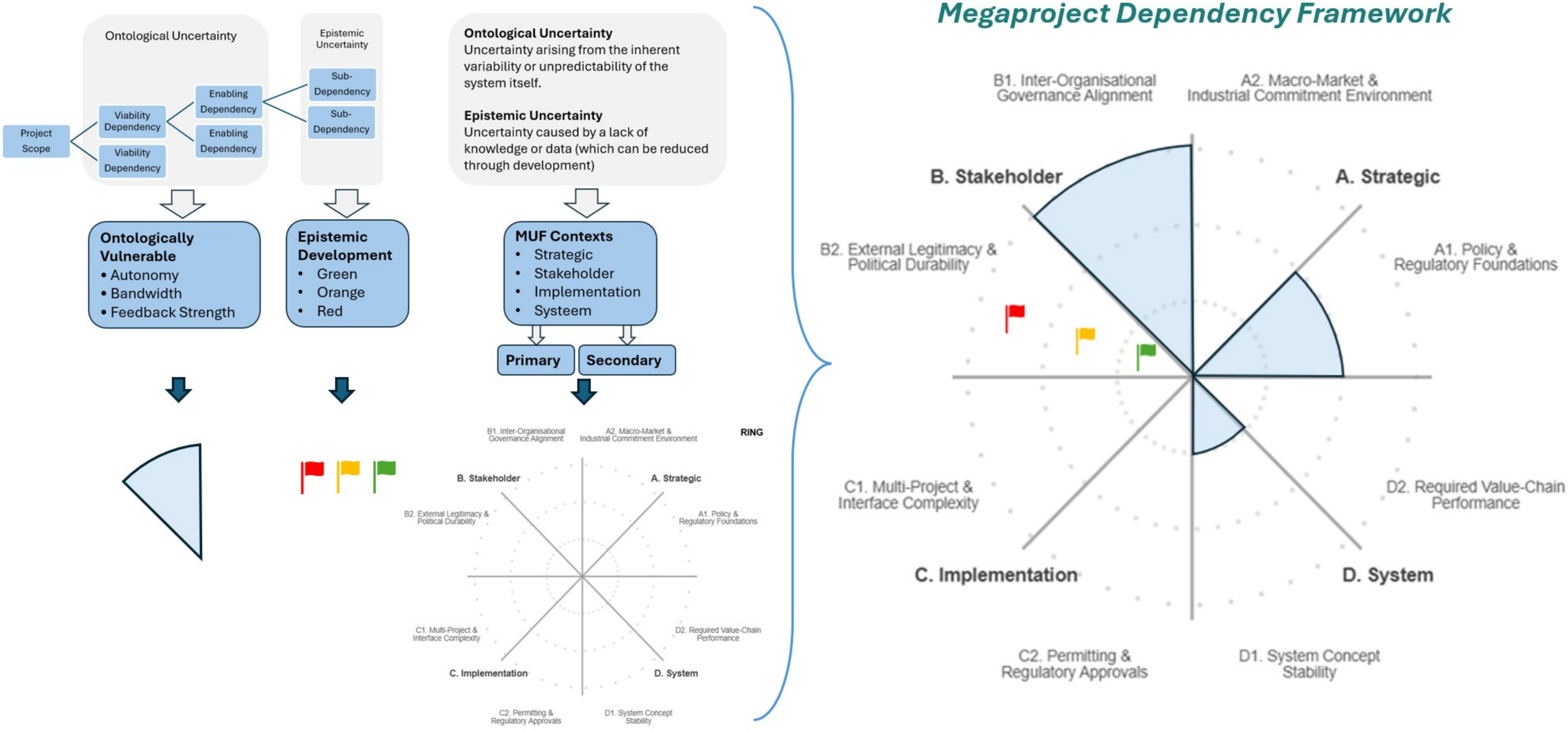
**Primary context:** In which context does uncertainty about this dependency primarily originate or become interpretable as a question of viability?

**Secondary contex:t** If uncertainty in this dependency develops, in which context does it logically become governance-relevant next?

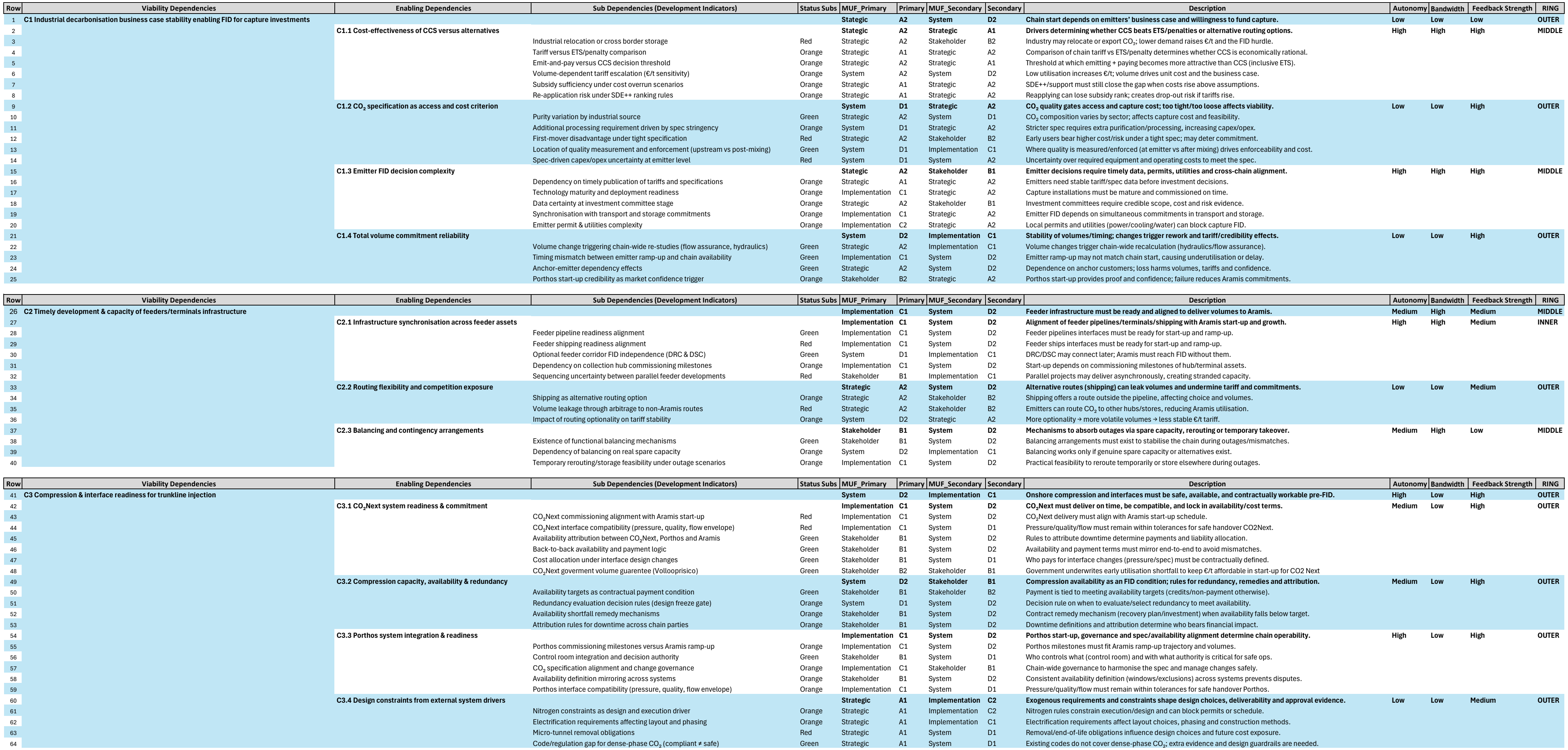
# B

# Aramis Dependencies Information

## B.1. Graphical Abstract



# B.2. Total data overview - Aramis dependencies



| Row | Viability Dependencies | Enabling Dependencies | Sub Dependencies (Development Indicators) | Status Subs | MUF_Primary | Primary | MUF_Secondary | Secondary | Description | Autonomy | Bandwidth | Feedback Strength | RING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | **C1 Industrial decarbonisation business case stability enabling FID for capture investments** | | | | **Stategic** | **A2** | **System** | **D2** | **Chain start depends on emitters' business case and willingness to fund capture.** | **Low** | **Low** | **Low** | **OUTER** |
| 2 | | **C1.1 Cost-effectiveness of CCS versus alternatives** | | | **Stategic** | **A2** | **Strategic** | **A1** | **Drivers determining whether CCS beats ETS/penalties or alternative routing options.** | **High** | **High** | **High** | **MIDDLE** |
| 3 | | | Industrial relocation or cross border storage | Red | Strategic | A2 | Stakeholder | B2 | Industry may relocate or export $CO_2$; lower demand raises €/t and the FID hurdle. | | | | |
| 4 | | | Tariff versus ETS/penalty comparison | Orange | Strategic | A1 | Strategic | A2 | Comparison of chain tariff vs ETS/penalty determines whether CCS is economically rational. | | | | |
| 5 | | | Emit-and-pay versus CCS decision threshold | Orange | Strategic | A2 | Strategic | A1 | Threshold at which emitting + paying becomes more attractive than CCS (inclusive ETS). | | | | |
| 6 | | | Volume-dependent tariff escalation (€/t sensitivity) | Orange | System | A2 | System | D2 | Low utilisation increases €/t; volume drives unit cost and the business case. | | | | |
| 7 | | | Subsidy sufficiency under cost overrun scenarios | Orange | Strategic | A1 | Strategic | A2 | SDE++/support must still close the gap when costs rise above assumptions. | | | | |
| 8 | | | Re-application risk under SDE++ ranking rules | Orange | Strategic | A1 | Strategic | A2 | Reapplying can lose subsidy rank; creates drop-out risk if tariffs rise. | | | | |
| 9 | | **C1.2 $CO_2$ specification as access and cost criterion** | | | **System** | **D1** | **Strategic** | **A2** | **$CO_2$ quality gates access and capture cost; too tight/too loose affects viability.** | **Low** | **Low** | **High** | **OUTER** |
| 10 | | | Purity variation by industrial source | Green | Strategic | A2 | System | D1 | $CO_2$ composition varies by sector; affects capture cost and feasibility. | | | | |
| 11 | | | Additional processing requirement driven by spec stringency | Orange | System | D1 | Strategic | A2 | Stricter spec requires extra purification/processing, increasing capex/opex. | | | | |
| 12 | | | First-mover disadvantage under tight specification | Red | Strategic | A2 | Stakeholder | B2 | Early users bear higher cost/risk under a tight spec; may deter commitment. | | | | |
| 13 | | | Location of quality measurement and enforcement (upstream vs post-mixing) | Green | System | D1 | Implementation | C1 | Where quality is measured/enforced (at emitter vs after mixing) drives enforceability and cost. | | | | |
| 14 | | | Spec-driven capex/opex uncertainty at emitter level | Red | System | D1 | System | A2 | Uncertainty over required equipment and operating costs to meet the spec. | | | | |
| 15 | | **C1.3 Emitter FID decision complexity** | | | **Stategic** | **A2** | **Stakeholder** | **B1** | **Emitter decisions require timely data, permits, utilities and cross-chain alignment.** | **High** | **High** | **High** | **MIDDLE** |
| 16 | | | Dependency on timely publication of tariffs and specifications | Orange | Strategic | A1 | Strategic | A2 | Emitters need stable tariff/spec data before investment decisions. | | | | |
| 17 | | | Technology maturity and deployment readiness | Orange | Implementation | C1 | Strategic | A2 | Capture installations must be mature and commissioned on time. | | | | |
| 18 | | | Data certainty at investment committee stage | Orange | Strategic | A2 | Stakeholder | B1 | Investment committees require credible scope, cost and risk evidence. | | | | |
| 19 | | | Synchronisation with transport and storage commitments | Orange | Implementation | C1 | Strategic | A2 | Emitter FID depends on simultaneous commitments in transport and storage. | | | | |
| 20 | | | Emitter permit & utilities complexity | Orange | Implementation | C2 | Strategic | A2 | Local permits and utilities (power/cooling/water) can block capture FID. | | | | |
| 21 | | **C1.4 Total volume commitment reliability** | | | **System** | **D2** | **Implementation** | **C1** | **Stability of volumes/timing; changes trigger rework and tariff/credibility effects.** | **Low** | **Low** | **High** | **OUTER** |
| 22 | | | Volume change triggering chain-wide re-studies (flow assurance, hydraulics) | Green | Strategic | A2 | Implementation | C1 | Volume changes trigger chain-wide recalculation (hydraulics/flow assurance). | | | | |
| 23 | | | Timing mismatch between emitter ramp-up and chain availability | Green | Implementation | C1 | System | D2 | Emitter ramp-up may not match chain start, causing underutilisation or delay. | | | | |
| 24 | | | Anchor-emitter dependency effects | Green | Strategic | A2 | System | D2 | Dependence on anchor customers; loss harms volumes, tariffs and confidence. | | | | |
| 25 | | | Porthos start-up credibility as market confidence trigger | Orange | Stakeholder | B2 | Strategic | A2 | Porthos start-up provides proof and confidence; failure reduces Aramis commitments. | | | | |

| Row | Viability Dependencies | Enabling Dependencies | Sub Dependencies (Development Indicators) | Status Subs | MUF_Primary | Primary | MUF_Secondary | Secondary | Description | Autonomy | Bandwidth | Feedback Strength | RING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 26 | **C2 Timely development & capacity of feeders/terminals infrastructure** | | | | **Implementation** | **C1** | **System** | **D2** | **Feeder infrastructure must be ready and aligned to deliver volumes to Aramis.** | **Medium** | **High** | **Medium** | **MIDDLE** |
| 27 | | **C2.1 Infrastructure synchronisation across feeder assets** | | | **Implementation** | **C1** | **System** | **D2** | **Alignment of feeder pipelines/terminals/shipping with Aramis start-up and growth.** | **High** | **High** | **Medium** | **INNER** |
| 28 | | | Feeder pipeline readiness alignment | Green | Implementation | C1 | System | D2 | Feeder pipelines interfaces must be ready for start-up and ramp-up. | | | | |
| 29 | | | Feeder shipping readiness alignment | Red | Implementation | C1 | System | D2 | Feeder ships interfaces must be ready for start-up and ramp-up. | | | | |
| 30 | | | Optional feeder corridor FID independence (DRC & DSC) | Green | System | D1 | Implementation | C1 | DRC/DSC may connect later; Aramis must reach FID without them. | | | | |
| 31 | | | Dependency on collection hub commissioning milestones | Orange | Implementation | C1 | System | D2 | Start-up depends on commissioning milestones of hub/terminal assets. | | | | |
| 32 | | | Sequencing uncertainty between parallel feeder developments | Red | Stakeholder | B1 | Implementation | C1 | Parallel projects may deliver asynchronously, creating stranded capacity. | | | | |
| 33 | | **C2.2 Routing flexibility and competition exposure** | | | **Strategic** | **A2** | **System** | **D2** | **Alternative routes (shipping) can leak volumes and undermine tariff and commitments.** | **Low** | **Low** | **Medium** | **OUTER** |
| 34 | | | Shipping as alternative routing option | Orange | Strategic | A2 | Stakeholder | B2 | Shipping offers a route outside the pipeline, affecting choice and volumes. | | | | |
| 35 | | | Volume leakage through arbitrage to non-Aramis routes | Red | Strategic | A2 | Stakeholder | B2 | Emitters can route $CO_2$ to other hubs/stores, reducing Aramis utilisation. | | | | |
| 36 | | | Impact of routing optionality on tariff stability | Orange | System | D2 | Strategic | A2 | More optionality → more volatile volumes → less stable €/t tariff. | | | | |
| 37 | | **C2.3 Balancing and contingency arrangements** | | | **Stakeholder** | **B1** | **System** | **D2** | **Mechanisms to absorb outages via spare capacity, rerouting or temporary takeover.** | **Medium** | **High** | **Low** | **MIDDLE** |
| 38 | | | Existence of functional balancing mechanisms | Green | Stakeholder | B1 | System | D2 | Balancing arrangements must exist to stabilise the chain during outages/mismatches. | | | | |
| 39 | | | Dependency of balancing on real spare capacity | Orange | System | D2 | Implementation | C1 | Balancing works only if genuine spare capacity or alternatives exist. | | | | |
| 40 | | | Temporary rerouting/storage feasibility under outage scenarios | Orange | Implementation | C1 | System | D2 | Practical feasibility to reroute temporarily or store elsewhere during outages. | | | | |

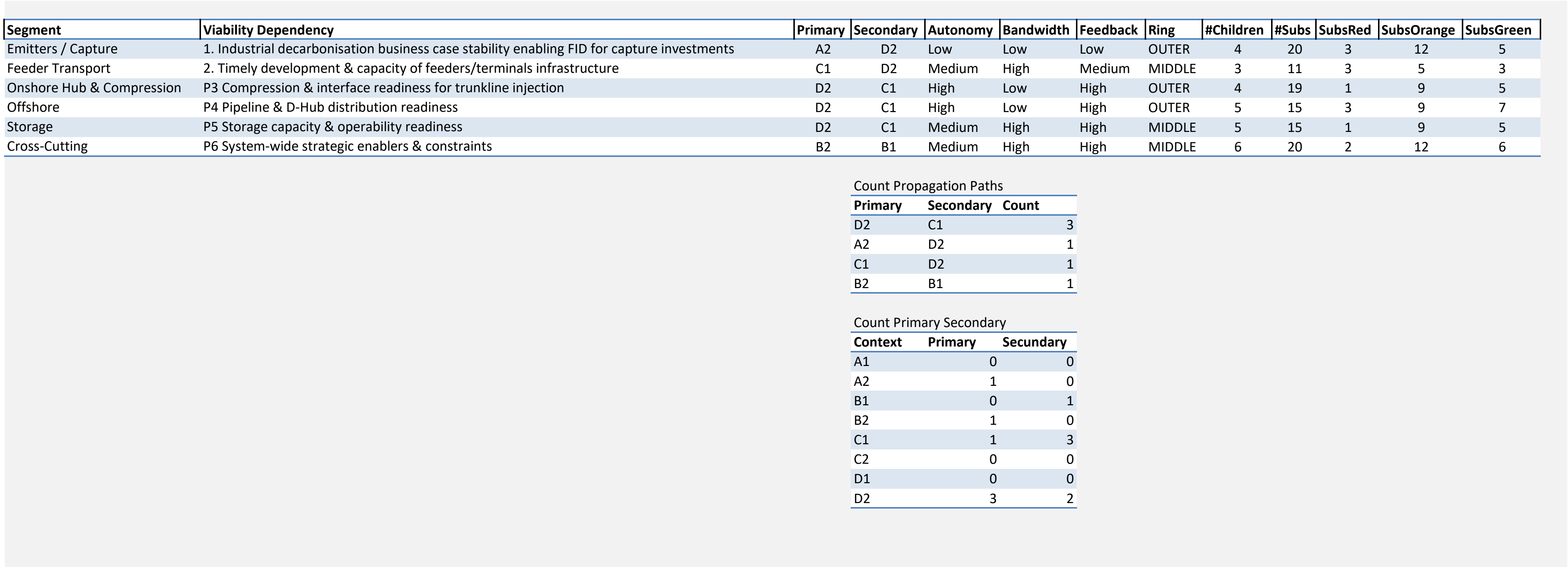
| Row | Viability Dependencies | Enabling Dependencies | Sub Dependencies (Development Indicators) | Status Subs | MUF_Primary | Primary | MUF_Secondary | Secondary | Description | Autonomy | Bandwidth | Feedback Strength | RING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 41 | **C3 Compression & interface readiness for trunkline injection** | | | | **System** | **D2** | **Implementation** | **C1** | **Onshore compression and interfaces must be safe, available, and contractually workable pre-FID.** | **High** | **Low** | **High** | **OUTER** |
| 42 | | **C3.1 $CO_2$Next system readiness & commitment** | | | **Implementation** | **C1** | **System** | **D2** | **$CO_2$Next must deliver on time, be compatible, and lock in availability/cost terms.** | **Medium** | **Low** | **High** | **OUTER** |
| 43 | | | $CO_2$Next commissioning alignment with Aramis start-up | Red | Implementation | C1 | System | D2 | $CO_2$Next delivery must align with Aramis start-up schedule. | | | | |
| 44 | | | $CO_2$Next interface compatibility (pressure, quality, flow envelope) | Red | Implementation | C1 | System | D1 | Pressure/quality/flow must remain within tolerances for safe handover CO2Next. | | | | |
| 45 | | | Availability attribution between $CO_2$Next, Porthos and Aramis | Green | Stakeholder | B1 | System | D2 | Rules to attribute downtime determine payments and liability allocation. | | | | |
| 46 | | | Back-to-back availability and payment logic | Green | Stakeholder | B1 | System | D2 | Availability and payment terms must mirror end-to-end to avoid mismatches. | | | | |
| 47 | | | Cost allocation under interface design changes | Green | Stakeholder | B1 | System | D1 | Who pays for interface changes (pressure/spec) must be contractually defined. | | | | |
| 48 | | | $CO_2$Next goverment volume guarentee (Vollooprisico) | Green | Stakeholder | B2 | Stakeholder | B1 | Government underwrites early utilisation shortfall to keep €/t affordable in start-up for CO2 Next | | | | |
| 49 | | **C3.2 Compression capacity, availability & redundancy** | | | **System** | **D2** | **Stakeholder** | **B1** | **Compression availability as an FID condition; rules for redundancy, remedies and attribution.** | **Medium** | **Low** | **High** | **OUTER** |
| 50 | | | Availability targets as contractual payment condition | Green | Stakeholder | B1 | Stakeholder | B2 | Payment is tied to meeting availability targets (credits/non-payment otherwise). | | | | |
| 51 | | | Redundancy evaluation decision rules (design freeze gate) | Orange | System | D1 | System | D2 | Decision rule on when to evaluate/select redundancy to meet availability. | | | | |
| 52 | | | Availability shortfall remedy mechanisms | Orange | Stakeholder | B1 | System | D2 | Contract remedy mechanism (recovery plan/investment) when availability falls below target. | | | | |
| 53 | | | Attribution rules for downtime across chain parties | Orange | Stakeholder | B1 | System | D2 | Downtime definitions and attribution determine who bears financial impact. | | | | |
| 54 | | **C3.3 Porthos system integration & readiness** | | | **Implementation** | **C1** | **System** | **D2** | **Porthos start-up, governance and spec/availability alignment determine chain operability.** | **High** | **Low** | **High** | **OUTER** |
| 55 | | | Porthos commissioning milestones versus Aramis ramp-up | Orange | Implementation | C1 | System | D2 | Porthos milestones must fit Aramis ramp-up trajectory and volumes. | | | | |
| 56 | | | Control room integration and decision authority | Green | Stakeholder | B1 | System | D1 | Who controls what (control room) and with what authority is critical for safe ops. | | | | |
| 57 | | | $CO_2$ specification alignment and change governance | Orange | Implementation | C1 | Stakeholder | B1 | Chain-wide governance to harmonise the spec and manage changes safely. | | | | |
| 58 | | | Availability definition mirroring across systems | Orange | Stakeholder | B1 | System | D2 | Consistent availability definition (windows/exclusions) across systems prevents disputes. | | | | |
| 59 | | | Porthos interface compatibility (pressure, quality, flow envelope) | Orange | Implementation | C1 | System | D1 | Pressure/quality/flow must remain within tolerances for safe handover Porthos. | | | | |
| 60 | | **C3.4 Design constraints from external system drivers** | | | **Strategic** | **A1** | **Implementation** | **C2** | **Exogenous requirements and constraints shape design choices, deliverability and approval evidence.** | **Low** | **Low** | **Medium** | **OUTER** |
| 61 | | | Nitrogen constraints as design and execution driver | Orange | Strategic | A1 | Implementation | C2 | Nitrogen rules constrain execution/design and can block permits or schedule. | | | | |
| 62 | | | Electrification requirements affecting layout and phasing | Orange | Strategic | A1 | Implementation | C1 | Electrification requirements affect layout choices, phasing and construction methods. | | | | |
| 63 | | | Micro-tunnel removal obligations | Red | Strategic | A1 | System | D1 | Removal/end-of-life obligations influence design choices and future cost exposure. | | | | |
| 64 | | | Code/regulation gap for dense-phase $CO_2$ (compliant ≠ safe) | Green | Strategic | A1 | System | D1 | Existing codes do not cover dense-phase $CO_2$; extra evidence and design guardrails are needed. | | | | |

| Row | Viability Dependencies | Enabling Dependencies | Sub Dependencies (Development Indicators) | Status Subs | MUF_Primary | Primary | MUF_Secondary | Secondary | Description | Autonomy | Bandwidth | Feedback Strength | RING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 65 | **C4 Pipeline & D-Hub distribution readiness** | | | | **System** | **D2** | **Implementation** | **C1** | **Offshore backbone must operate safely as an integrated system and remain robust under disturbances.** | **High** | **Low** | **High** | **OUTER** |
| 66 | | **C4.1 Offshore transport integrity and operability** | | | **System** | **D1** | **Implementation** | **C1** | **Dense-phase integrity: T/p envelope, corrosion behavior and upset response govern operability.** | **High** | **Low** | **High** | **OUTER** |
| 67 | | | Pressure and temperature envelope stability | Orange | System | D2 | Implementation | C1 | Maintaining T/p to avoid phase change and preserve integrity. | | | | |
| 68 | | | Dense-phase $CO_2$ transport behaviour under upset conditions | Green | System | D2 | System | D1 | Transient/upset behaviour drives two-phase risk and operating limits. | | | | |
| 69 | | | Corrosion and impurity sensitivity | Orange | System | D1 | Implementation | C1 | Impurities/water can trigger corrosion/acid formation, shaping design and spec. | | | | |
| 70 | | **C4.2 Safety-by-interface dependency** | | | **Implementation** | **C1** | **System** | **D2** | **Safety depends on partner interfaces, safeguards, trip logic and safety-case evidence.** | **High** | **High** | **High** | **MIDDLE** |
| 71 | | | Reliance on upstream/downstream protection systems | Green | Implementation | C1 | System | D2 | Aramis relies on partner protections (trips/PSVs) to protect integrity. | | | | |
| 72 | | | Interface-defined safeguards and trip logic | Green | Implementation | C1 | Stakeholder | B1 | Safeguards and trip logic must be consistent and tested across parties. | | | | |
| 73 | | | Loss of autonomy under partner system failures | Orange | System | D2 | Implementation | C1 | Partner failures can stop Aramis; limited control increases vulnerability. | | | | |
| 74 | | | Failure-mode discovery & safety case evidence (no rulebook) | Orange | Stakeholder | B2 | Implementation | C2 | New failure modes must be identified and evidenced in the safety case for authorities. | | | | |
| 75 | | **C4.3 Single-point-of-failure exposure** | | | **System** | **D2** | **Stakeholder** | **B2** | **Trunkline failure halts the chain; recovery/mean time to repair drives commercial and operational impact.** | **Low** | **Low** | **High** | **OUTER** |
| 76 | | | Trunkline downtime halting entire value chain | Green | System | D2 | Strategic | A2 | Backbone outage stops all flows; the key single-point-of-failure exposure. | | | | |
| 77 | | | Recovery and restart complexity after offshore outage | Green | Implementation | C1 | System | D2 | Recovery/restart requires time and procedures, affecting mean time to repair and payments. | | | | |
| 78 | | **C4.4 Off-spec and upset handling procedures** | | | **Implementation** | **C1** | **System** | **D2** | **Decision logic for stop/vent/recovery and controls to manage off-spec events.** | **Low** | **Low** | **High** | **OUTER** |
| 79 | | | Integrated control philosofy | Orange | System | D1 | Stakeholder | B1 | A unified control philosophy (setpoints/handoffs) is needed for safe integrated operation. | | | | |
| 80 | | | Stop / vent / recovery decision logic | Orange | Implementation | C1 | System | D2 | Predefined decision logic for isolation, depressurisation/venting and recovery. | | | | |
| 81 | | | Impact of upset handling on chain availability | Orange | System | D2 | Strategic | A2 | Upset handling choices determine downtime, payments and confidence. | | | | |
| 82 | | **C4.5 Execution capacity & schedule realism** | | | **Implementation** | **C1** | **Strategic** | **A2** | **Contractor/vessel availability and realistic P-curves determine pre-FID deliverability.** | **Low** | **Low** | **Medium** | **OUTER** |
| 83 | | | Contractor scarcity & schedule gate | Red | Implementation | C1 | Strategic | A2 | Scarce contractors/vessels gate the schedule and can drive cost and FID timing. | | | | |
| 84 | | | Safety & schedule compromise mechansim | Orange | Stakeholder | B2 | Implementation | C1 | Schedule pressure can trigger shortcuts, raising safety and incident risk. | | | | |
| 85 | | | EU tender/procurement rules fit with execution | Orange | Strategic | A1 | Implementation | C1 | Tender frameworks fit poorly with offshore risk; shapes contracting strategy. | | | | |

| Row | Viability Dependencies | Enabling Dependencies | Sub Dependencies (Development Indicators) | Status Subs | MUF_Primary | Primary | MUF_Secondary | Secondary | Description | Autonomy | Bandwidth | Feedback Strength | RING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 86 | **C5 Storage capacity & operability readiness** | | - | | **System** | **D2** | **Implementation** | **C1** | **Storage must provide sufficient capacity, injectivity and investment commitment for ramp-up.** | **Medium** | **High** | **High** | **MIDDLE** |
| 87 | | **C5.1 Storage capacity portfolio adequacy** | - | | **System** | **D2** | **Implementation** | **C1** | **A portfolio of fields must be sufficient and timely developed for throughput/ramp-up.** | **Low** | **High** | **High** | **OUTER** |
| 88 | | | Availability of sufficient developed fields | Green | System | D2 | Strategic | A2 | Enough fields must be developed in time to accept injection volumes. | | | | |
| 89 | | | Phasing of field development versus Aramis throughput | Green | Implementation | C1 | System | D2 | Field phasing must match pipeline throughput and ramp-up to avoid bottlenecks. | | | | |
| 90 | | | Ramp-up mismatch risks | Green | Implementation | C1 | System | D2 | Asynchronous field start-up vs transport causes underutilisation or constraints. | | | | |
| 91 | | | Opportunity funnel potential prospects | Red | Strategic | A2 | Stakeholder | B2 | The funnel of storage opportunities determines future capacity confidence. | | | | |
| 92 | | **C5.2 Injectivity and operational flexibility** | - | | **System** | **D2** | **System** | **D1** | **Injectivity and flexibility determine whether volumes can be injected safely without buffer.** | **Low** | **Low** | **Low** | **OUTER** |
| 93 | | | Injectivity as hard system constraint | Orange | System | D2 | System | D1 | Injectivity limits offtake; without injection the chain stops. | | | | |
| 94 | | | Limited buffering capacity (what goes in must go down) | Green | System | D2 | Implementation | C1 | Limited buffering creates immediate dependence on continuous injection. | | | | |
| 95 | | | Upset tolerance and contingency injection options | Orange | Implementation | C1 | Stakeholder | B2 | Options to pause, reroute or inject contingently during anomalies. | | | | |
| 96 | | **C5.3 Store availability and revenue exposure** | - | | **System** | **D2** | **Strategic** | **A1** | **Injection downtime hits revenues/subsidies and pressures chain payments.** | **High** | **High** | **Medium** | **INNER** |
| 97 | | | Store availability dependency on subsidy and revenue streams | Orange | System | D2 | Strategic | A1 | Storage downtime can hit subsidy/revenue, affecting willingness to commit. | | | | |
| 98 | | | Financial exposure under injection downtime | Orange | Stakeholder | B1 | Stakeholder | B2 | Who pays/compensates during downtime drives risk and contract design. | | | | |
| 99 | | **C5.4 $CO_2$ specification bounded by storage constraints** | - | | **Implementation** | **C2** | **System** | **D1** | **Storage license bounds composition; acceptance criteria propagate upstream into design.** | **Low** | **Low** | **Low** | **OUTER** |
| 100 | | | License-driven composition limits | Orange | Implementation | C2 | Stakeholder | B2 | Storage permits bound composition; off-spec can cause rejection/shutdown. | | | | |
| 101 | | | Upstream design implications of storage acceptance criteria | Green | Implementation | C1 | System | D1 | Storage acceptance criteria propagate upstream into spec, compression and operations. | | | | |
| 102 | | **C5.5 Stores investment commitment logic** | | | **Strategic** | **A2** | **System** | **D2** | **Storage parties invest only if risk reward balance is acceptable and chain FID alignment exists.** | **Medium** | **High** | **Low** | **MIDDLE** |
| 103 | | | Risk reward balance thresholds for private storage parties | Orange | Strategic | A2 | System | D2 | Private stores invest only if risk reward balance is sufficient. | | | | |
| 104 | | | Launch stores 5 megaton start-up momentum | Orange | Implementation | C1 | Stakeholder | B1 | Store FID depends on Aramis/emitters; mutual dependence delays decisions. | | | | |
| 105 | | | Stores competition by design to enhance the market for CCS commitment | Orange | Stakeholder | B1 | Implementation | C1 | Stores may coordinate to avoid destructive competition and increase investment certainty. | | | | |
| 106 | | | Implementation requirements Net Zero Industry Act | Orange | Implementation | C2 | Stakeholder | B2 | Oil and Gas companies may be obliged to develop storages facilities | | | | |

| Row | Viability Dependencies | Enabling Dependencies | Sub Dependencies (Development Indicators) | Status Subs | MUF_Primary | Primary | MUF_Secondary | Secondary | Description | Autonomy | Bandwidth | Feedback Strength | RING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 107 | **C6 System-wide strategic enablers & constraints** | | | | **Stakeholder** | **B2** | **Stakeholder** | **B1** | **Chain-wide conditions: synchronisation, contract coherence, optionality and permitting status.** | **Medium** | **High** | **High** | **MIDDLE** |
| 108 | | **C6.1 Tariff formation and cost discovery** | | | **System** | **D1** | **Strategic** | **A2** | **Tariff emerges from market prices and chain costs; uncertain until cost estimates mature.** | **Medium** | **Low** | **High** | **OUTER** |
| 109 | | | BAFO / CFT timing and late tariff certainty | Green | Strategic | A2 | System | D2 | Tariff certainty arrives only after tender/BAFO, affecting FID timing. | | | | |
| 110 | | | Emerging tariff build-up from chain-wide capex/opex excluding emitters | Orange | System | D1 | Strategic | A2 | Tariff is the sum of costs across the chain; each part affects the final price. | | | | |
| 111 | | | Sensitivity of tariff to downstream and upstream costs towards emitters | Orange | System | D1 | Strategic | A2 | Cost swings in stores/terminals/pipeline directly shift €/t. | | | | |
| 112 | | | Aramis government volume guarentee (Vollooprisico) | Green | Stakeholder | B2 | Strategic | A1 | Government underwrites early utilisation shortfall to keep €/t affordable in start-up for Aramis | | | | |
| 113 | | **C6.2 System integration governance** | | | **Stakeholder** | **B1** | **System** | **D1** | **Integrator mandate and change control determine safe and workable system integration.** | **Medium** | **High** | **High** | **MIDDLE** |
| 114 | | | System integrator role definition and mandate | Orange | Stakeholder | B1 | Stakeholder | B2 | Integrator scope/mandate defines who carries integration responsibility and risk. | | | | |
| 115 | | | Change-control authority across interfaces | Green | Stakeholder | B1 | Implementation | C1 | Who can approve interface changes; prevents uncontrolled scope churn. | | | | |
| 116 | | | Back-to-back mirroring of operational and financial logic | Green | Stakeholder | B1 | System | D2 | Operational rules must be financially and legally consistent across the chain. | | | | |
| 117 | | **C6.3 FID synchronisation across the value chain** | | | **Implementation** | **C1** | **Stakeholder** | **B2** | **Decisions are interdependent; simultaneous start/commitment is critical pre-FID.** | **Medium** | **Low** | **High** | **OUTER** |
| 118 | | | Chicken-and-egg dynamics across >20 parties | Orange | Implementation | C1 | Stakeholder | B1 | Many parties wait for each other; without simultaneous commits, no FID breakthrough. | | | | |
| 119 | | | Mutual dependence of investment commitments | Red | Implementation | C1 | Strategic | A2 | FIDs are coupled (emitters–transport–stores); delay by one affects the whole chain. | | | | |
| 120 | | **C6.4 Chain-wide contractual coherence** | | | **Stakeholder** | **B1** | **System** | **D2** | **End-to-end contract coherence avoids residual exposure via back-to-back and IOA terms.** | **Low** | **High** | **High** | **OUTER** |
| 121 | | | Back-to-back payment logic | Orange | Stakeholder | B1 | System | D2 | Payment terms must flow end-to-end or risk remains stranded with one party. | | | | |
| 122 | | | Availability mirroring across contracts | Orange | Stakeholder | B1 | System | D2 | One availability definition prevents disputes and residual payment exposure. | | | | |
| 123 | | | Residual exposure prevention | Green | Stakeholder | B1 | System | D2 | Mechanisms to prevent risk gaps (carve-outs, caps, true-ups). | | | | |
| 124 | | | IOA/IOE/TOI/EMOC | Orange | Stakeholder | B1 | Implementation | C1 | Integrated operating agreements and open items steer safe operations and change control. | | | | |
| 125 | | **C6.5 Strategic optionality and system rigidity** | | | **System** | **D1** | **Strategic** | **A1** | **Degree of lock-in; once committed, decoupling is hard and politically sensitive.** | **Low** | **High** | **High** | **OUTER** |
| 126 | | | Limited ability to phase or decouple investments | Orange | System | D1 | Implementation | C1 | The system is hard to split; phasing can create stranded assets and tariff stress. | | | | |
| 127 | | | Lock-in effects once trunk infrastructure is committed | Green | System | D1 | Strategic | A2 | After backbone commitment, changes are costly; re-FEED hurts schedule and support. | | | | |
| 128 | | | Policy-defined investment horizon | Orange | Strategic | A1 | System | D1 | Policy caps/horizons (e.g., subsidy duration) constrain long-term business cases. | | | | |
| 129 | | **C6.6 Permitting finality and legal operability** | | | **Implementation** | **C2** | **Stakeholder** | **B2** | **Legal finality and approval acceptance determine whether FID is legally and practically feasible.** | **Low** | **Low** | **High** | **OUTER** |
| 130 | | | Legal finality versus reversibility | Red | Implementation | C2 | Stakeholder | B2 | Definitive vs irrevocable; legal challengeability shapes FID willingness. | | | | |
| 131 | | | Interdependency of permits across chain segments | Orange | Implementation | C2 | Stakeholder | B1 | Permits act as a bundle; one weak link can block the whole set. | | | | |
| 132 | | | Temporal alignment of permit validity with FID and execution | Orange | Stakeholder | B2 | Implementation | C1 | Permit timing/validity must match FID and execution windows. | | | | |
| 133 | | | Authority acceptance of integrated system safety cases | Orange | Stakeholder | B2 | System | D1 | Authorities must accept the integrated safety case before FID/commissioning. | | | | |

### B.2.1. Viability Dependencies - Data



| Segment | Viability Dependency | Primary | Secondary | Autonomy | Bandwidth | Feedback | Ring | #Children | #Subs | SubsRed | SubsOrange | SubsGreen |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Emitters / Capture | 1. Industrial decarbonisation business case stability enabling FID for capture investments | A2 | D2 | Low | Low | Low | OUTER | 4 | 20 | 3 | 12 | 5 |
| Feeder Transport | 2. Timely development & capacity of feeders/terminals infrastructure | C1 | D2 | Medium | High | Medium | MIDDLE | 3 | 11 | 3 | 5 | 3 |
| Onshore Hub & Compression | P3 Compression & interface readiness for trunkline injection | D2 | C1 | High | Low | High | OUTER | 4 | 19 | 1 | 9 | 5 |
| Offshore | P4 Pipeline & D-Hub distribution readiness | D2 | C1 | High | Low | High | OUTER | 5 | 15 | 3 | 9 | 7 |
| Storage | P5 Storage capacity & operability readiness | D2 | C1 | Medium | High | High | MIDDLE | 5 | 15 | 1 | 9 | 5 |
| Cross-Cutting | P6 System-wide strategic enablers & constraints | B2 | B1 | Medium | High | High | MIDDLE | 6 | 20 | 2 | 12 | 6 |

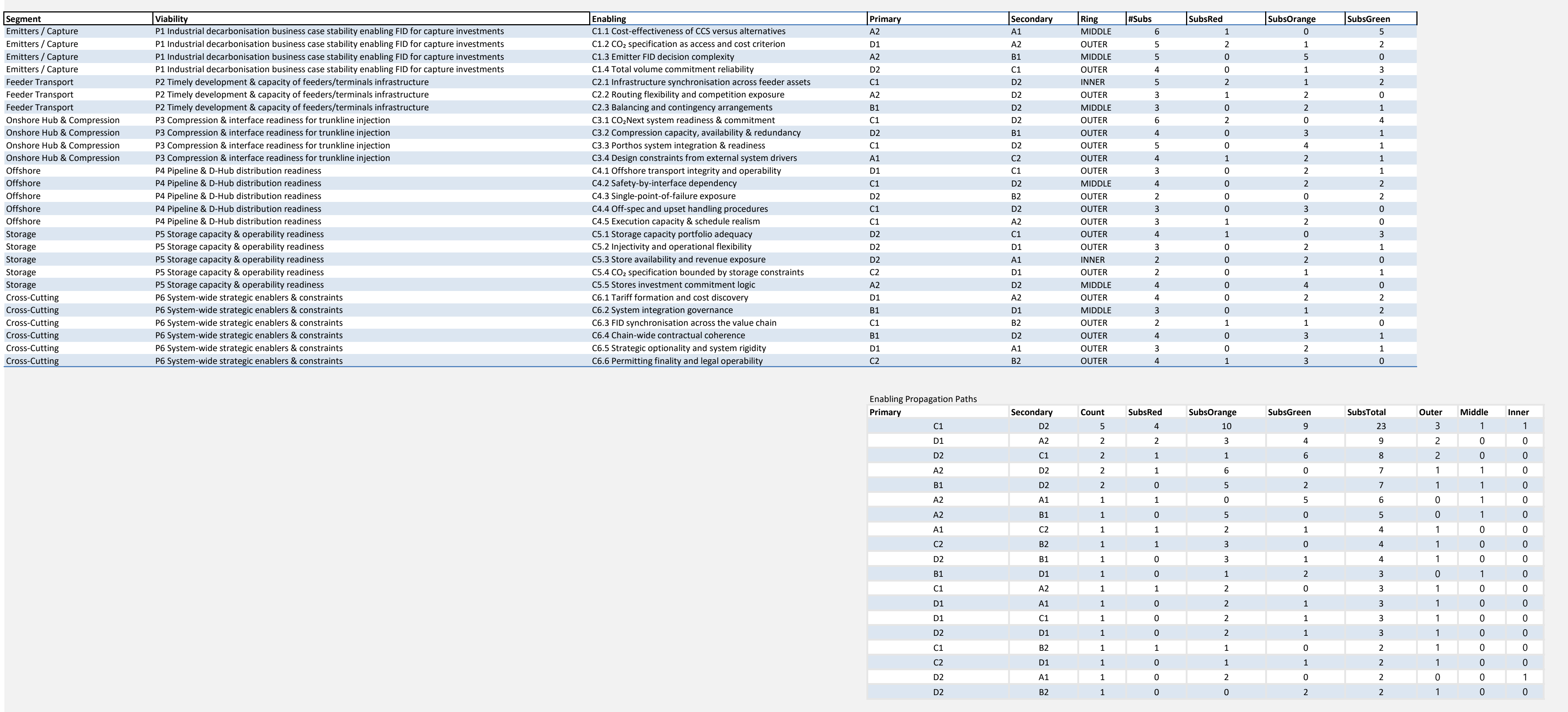
Count Propagation Paths

| Primary | Secondary | Count |
|---|---|---|
| D2 | C1 | 3 |
| A2 | D2 | 1 |
| C1 | D2 | 1 |
| B2 | B1 | 1 |

Count Primary Secondary

| Context | Primary | Secundary |
|---|---|---|
| A1 | 0 | 0 |
| A2 | 1 | 0 |
| B1 | 0 | 1 |
| B2 | 1 | 0 |
| C1 | 1 | 3 |
| C2 | 0 | 0 |
| D1 | 0 | 0 |
| D2 | 3 | 2 |

## B.2.2. Enabling Dependencies - Data

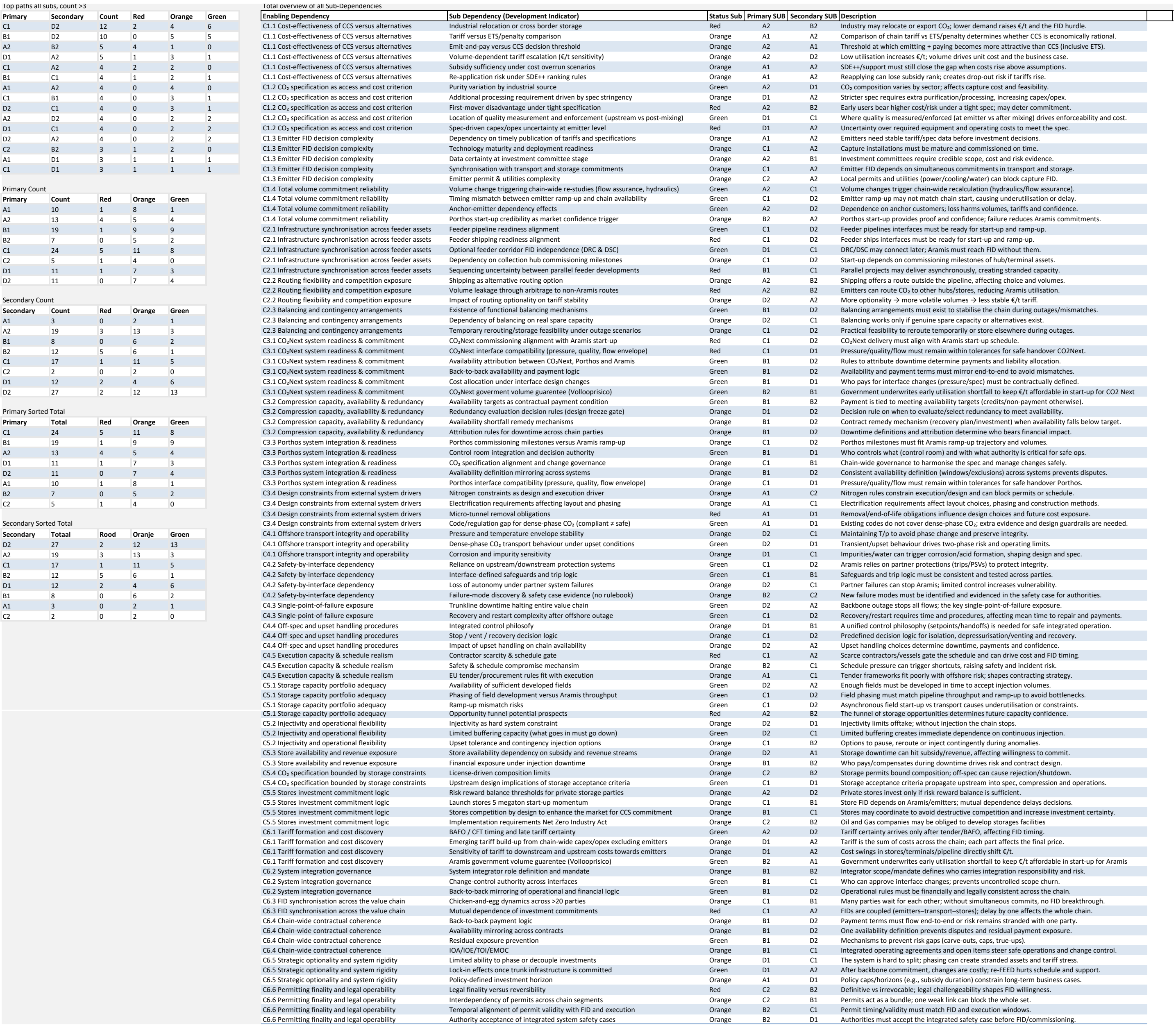


| Segment | Viability | Enabling | Primary | Secondary | Ring | #Subs | SubsRed | SubsOrange | SubsGreen |
|---|---|---|---|---|---|---|---|---|---|
| Emitters / Capture | P1 Industrial decarbonisation business case stability enabling FID for capture investments | C1.1 Cost-effectiveness of CCS versus alternatives | A2 | A1 | MIDDLE | 6 | 1 | 0 | 5 |
| Emitters / Capture | P1 Industrial decarbonisation business case stability enabling FID for capture investments | C1.2 $CO_2$ specification as access and cost criterion | D1 | A2 | OUTER | 5 | 2 | 1 | 2 |
| Emitters / Capture | P1 Industrial decarbonisation business case stability enabling FID for capture investments | C1.3 Emitter FID decision complexity | A2 | B1 | MIDDLE | 5 | 0 | 5 | 0 |
| Emitters / Capture | P1 Industrial decarbonisation business case stability enabling FID for capture investments | C1.4 Total volume commitment reliability | D2 | C1 | OUTER | 4 | 0 | 1 | 3 |
| Feeder Transport | P2 Timely development & capacity of feeders/terminals infrastructure | C2.1 Infrastructure synchronisation across feeder assets | C1 | D2 | INNER | 5 | 2 | 1 | 2 |
| Feeder Transport | P2 Timely development & capacity of feeders/terminals infrastructure | C2.2 Routing flexibility and competition exposure | A2 | D2 | OUTER | 3 | 1 | 2 | 0 |
| Feeder Transport | P2 Timely development & capacity of feeders/terminals infrastructure | C2.3 Balancing and contingency arrangements | B1 | D2 | MIDDLE | 3 | 0 | 2 | 1 |
| Onshore Hub & Compression | P3 Compression & interface readiness for trunkline injection | C3.1 $CO_2$Next system readiness & commitment | C1 | D2 | OUTER | 6 | 2 | 0 | 4 |
| Onshore Hub & Compression | P3 Compression & interface readiness for trunkline injection | C3.2 Compression capacity, availability & redundancy | D2 | B1 | OUTER | 4 | 0 | 3 | 1 |
| Onshore Hub & Compression | P3 Compression & interface readiness for trunkline injection | C3.3 Porthos system integration & readiness | C1 | D2 | OUTER | 5 | 0 | 4 | 1 |
| Onshore Hub & Compression | P3 Compression & interface readiness for trunkline injection | C3.4 Design constraints from external system drivers | A1 | C2 | OUTER | 4 | 1 | 2 | 1 |
| Offshore | P4 Pipeline & D-Hub distribution readiness | C4.1 Offshore transport integrity and operability | D1 | C1 | OUTER | 3 | 0 | 2 | 1 |
| Offshore | P4 Pipeline & D-Hub distribution readiness | C4.2 Safety-by-interface dependency | C1 | D2 | MIDDLE | 4 | 0 | 2 | 2 |
| Offshore | P4 Pipeline & D-Hub distribution readiness | C4.3 Single-point-of-failure exposure | D2 | B2 | OUTER | 2 | 0 | 0 | 2 |
| Offshore | P4 Pipeline & D-Hub distribution readiness | C4.4 Off-spec and upset handling procedures | C1 | D2 | OUTER | 3 | 0 | 3 | 0 |
| Offshore | P4 Pipeline & D-Hub distribution readiness | C4.5 Execution capacity & schedule realism | C1 | A2 | OUTER | 3 | 1 | 2 | 0 |
| Storage | P5 Storage capacity & operability readiness | C5.1 Storage capacity portfolio adequacy | D2 | C1 | OUTER | 4 | 1 | 0 | 3 |
| Storage | P5 Storage capacity & operability readiness | C5.2 Injectivity and operational flexibility | D2 | D1 | OUTER | 3 | 0 | 2 | 1 |
| Storage | P5 Storage capacity & operability readiness | C5.3 Store availability and revenue exposure | D2 | A1 | INNER | 2 | 0 | 2 | 0 |
| Storage | P5 Storage capacity & operability readiness | C5.4 $CO_2$ specification bounded by storage constraints | C2 | D1 | OUTER | 2 | 0 | 1 | 1 |
| Storage | P5 Storage capacity & operability readiness | C5.5 Stores investment commitment logic | A2 | D2 | MIDDLE | 4 | 0 | 4 | 0 |
| Cross-Cutting | P6 System-wide strategic enablers & constraints | C6.1 Tariff formation and cost discovery | D1 | A2 | OUTER | 4 | 0 | 2 | 2 |
| Cross-Cutting | P6 System-wide strategic enablers & constraints | C6.2 System integration governance | B1 | D1 | MIDDLE | 3 | 0 | 1 | 2 |
| Cross-Cutting | P6 System-wide strategic enablers & constraints | C6.3 FID synchronisation across the value chain | C1 | B2 | OUTER | 2 | 1 | 1 | 0 |
| Cross-Cutting | P6 System-wide strategic enablers & constraints | C6.4 Chain-wide contractual coherence | B1 | D2 | OUTER | 4 | 0 | 3 | 1 |
| Cross-Cutting | P6 System-wide strategic enablers & constraints | C6.5 Strategic optionality and system rigidity | D1 | A1 | OUTER | 3 | 0 | 2 | 1 |
| Cross-Cutting | P6 System-wide strategic enablers & constraints | C6.6 Permitting finality and legal operability | C2 | B2 | OUTER | 4 | 1 | 3 | 0 |

Enabling Propagation Paths

| Primary | Secondary | Count | SubsRed | SubsOrange | SubsGreen | SubsTotal | Outer | Middle | Inner |
|---|---|---|---|---|---|---|---|---|---|
| C1 | D2 | 5 | 4 | 10 | 9 | 23 | 3 | 1 | 1 |
| D1 | A2 | 2 | 2 | 3 | 4 | 9 | 2 | 0 | 0 |
| D2 | C1 | 2 | 1 | 1 | 6 | 8 | 2 | 0 | 0 |
| A2 | D2 | 2 | 1 | 6 | 0 | 7 | 1 | 1 | 0 |
| B1 | D2 | 2 | 0 | 5 | 2 | 7 | 1 | 1 | 0 |
| A2 | A1 | 1 | 1 | 0 | 5 | 6 | 0 | 1 | 0 |
| A2 | B1 | 1 | 0 | 5 | 0 | 5 | 0 | 1 | 0 |
| A1 | C2 | 1 | 1 | 2 | 1 | 4 | 1 | 0 | 0 |
| C2 | B2 | 1 | 1 | 3 | 0 | 4 | 1 | 0 | 0 |
| D2 | B1 | 1 | 0 | 3 | 1 | 4 | 1 | 0 | 0 |
| B1 | D1 | 1 | 0 | 1 | 2 | 3 | 0 | 1 | 0 |
| C1 | A2 | 1 | 1 | 2 | 0 | 3 | 1 | 0 | 0 |
| D1 | A1 | 1 | 0 | 2 | 1 | 3 | 1 | 0 | 0 |
| D1 | C1 | 1 | 0 | 2 | 1 | 3 | 1 | 0 | 0 |
| D2 | D1 | 1 | 0 | 2 | 1 | 3 | 1 | 0 | 0 |
| C1 | B2 | 1 | 1 | 1 | 0 | 2 | 1 | 0 | 0 |
| C2 | D1 | 1 | 0 | 1 | 1 | 2 | 1 | 0 | 0 |
| D2 | A1 | 1 | 0 | 2 | 0 | 2 | 0 | 0 | 1 |
| D2 | B2 | 1 | 0 | 0 | 2 | 2 | 1 | 0 | 0 |

### B.2.3. Sub Dependencies / Development indicators - Data



Top paths all subs, count >3

| Primary | Secondary | Count | Red | Orange | Green |
|---|---|---|---|---|---|
| C1 | D2 | 12 | 2 | 4 | 6 |
| B1 | D2 | 10 | 0 | 5 | 5 |
| A2 | B2 | 5 | 4 | 1 | 0 |
| D1 | A2 | 5 | 1 | 3 | 1 |
| C1 | A2 | 4 | 2 | 2 | 0 |
| B1 | C1 | 4 | 1 | 2 | 1 |
| A1 | A2 | 4 | 0 | 4 | 0 |
| C1 | B1 | 4 | 0 | 3 | 1 |
| D2 | C1 | 4 | 0 | 3 | 1 |
| A2 | D2 | 4 | 0 | 2 | 2 |
| D1 | C1 | 4 | 0 | 2 | 2 |
| D2 | A2 | 4 | 0 | 2 | 2 |
| C2 | B2 | 3 | 1 | 2 | 0 |
| A1 | D1 | 3 | 1 | 1 | 1 |
| C1 | D1 | 3 | 1 | 1 | 1 |

Primary Count

| Primary | Count | Red | Orange | Green |
|---|---|---|---|---|
| A1 | 10 | 1 | 8 | 1 |
| A2 | 13 | 4 | 5 | 4 |
| B1 | 19 | 1 | 9 | 9 |
| B2 | 7 | 0 | 5 | 2 |
| C1 | 24 | 5 | 11 | 8 |
| C2 | 5 | 1 | 4 | 0 |
| D1 | 11 | 1 | 7 | 3 |
| D2 | 11 | 0 | 7 | 4 |

Secondary Count

| Secondary | Count | Red | Orange | Green |
|---|---|---|---|---|
| A1 | 3 | 0 | 2 | 1 |
| A2 | 19 | 3 | 13 | 3 |
| B1 | 8 | 0 | 6 | 2 |
| B2 | 12 | 5 | 6 | 1 |
| C1 | 17 | 1 | 11 | 5 |
| C2 | 2 | 0 | 2 | 0 |
| D1 | 12 | 2 | 4 | 6 |
| D2 | 27 | 2 | 12 | 13 |

Primary Sorted Total

| Primary | Total | Red | Orange | Green |
|---|---|---|---|---|
| C1 | 24 | 5 | 11 | 8 |
| B1 | 19 | 1 | 9 | 9 |
| A2 | 13 | 4 | 5 | 4 |
| D1 | 11 | 1 | 7 | 3 |
| D2 | 11 | 0 | 7 | 4 |
| A1 | 10 | 1 | 8 | 1 |
| B2 | 7 | 0 | 5 | 2 |
| C2 | 5 | 1 | 4 | 0 |

Secondary Sorted Total

| Secondary | Totaal | Rood | Oranje | Groen |
|---|---|---|---|---|
| D2 | 27 | 2 | 12 | 13 |
| A2 | 19 | 3 | 13 | 3 |
| C1 | 17 | 1 | 11 | 5 |
| B2 | 12 | 5 | 6 | 1 |
| D1 | 12 | 2 | 4 | 6 |
| B1 | 8 | 0 | 6 | 2 |
| A1 | 3 | 0 | 2 | 1 |
| C2 | 2 | 0 | 2 | 0 |

Total overview of all Sub-Dependencies

| Enabling Dependency | Sub Dependency (Development Indicator) | Status Sub | Primary SUB | Secondary SUB | Description |
|---|---|---|---|---|---|
| C1.1 Cost-effectiveness of CCS versus alternatives | Industrial relocation or cross border storage | Red | A2 | B2 | Industry may relocate or export $CO_2$; lower demand raises €/t and the FID hurdle. |
| C1.1 Cost-effectiveness of CCS versus alternatives | Tariff versus ETS/penalty comparison | Orange | A1 | A2 | Comparison of chain tariff vs ETS/penalty determines whether CCS is economically rational. |
| C1.1 Cost-effectiveness of CCS versus alternatives | Emit-and-pay versus CCS decision threshold | Orange | A2 | A1 | Threshold at which emitting + paying becomes more attractive than CCS (inclusive ETS). |
| C1.1 Cost-effectiveness of CCS versus alternatives | Volume-dependent tariff escalation (€/t sensitivity) | Orange | A2 | D2 | Low utilisation increases €/t; volume drives unit cost and the business case. |
| C1.1 Cost-effectiveness of CCS versus alternatives | Subsidy sufficiency under cost overrun scenarios | Orange | A1 | A2 | SDE++/support must still close the gap when costs rise above assumptions. |
| C1.1 Cost-effectiveness of CCS versus alternatives | Re-application risk under SDE++ ranking rules | Orange | A1 | A2 | Reapplying can lose subsidy rank; creates drop-out risk if tariffs rise. |
| C1.2 $CO_2$ specification as access and cost criterion | Purity variation by industrial source | Green | A2 | D1 | $CO_2$ composition varies by sector; affects capture cost and feasibility. |
| C1.2 $CO_2$ specification as access and cost criterion | Additional processing requirement driven by spec stringency | Orange | D1 | A2 | Stricter spec requires extra purification/processing, increasing capex/opex. |
| C1.2 $CO_2$ specification as access and cost criterion | First-mover disadvantage under tight specification | Red | A2 | B2 | Early users bear higher cost/risk under a tight spec; may deter commitment. |
| C1.2 $CO_2$ specification as access and cost criterion | Location of quality measurement and enforcement (upstream vs post-mixing) | Green | D1 | C1 | Where quality is measured/enforced (at emitter vs after mixing) drives enforceability and cost. |
| C1.2 $CO_2$ specification as access and cost criterion | Spec-driven capex/opex uncertainty at emitter level | Red | D1 | A2 | Uncertainty over required equipment and operating costs to meet the spec. |
| C1.3 Emitter FID decision complexity | Dependency on timely publication of tariffs and specifications | Orange | A1 | A2 | Emitters need stable tariff/spec data before investment decisions. |
| C1.3 Emitter FID decision complexity | Technology maturity and deployment readiness | Orange | C1 | A2 | Capture installations must be mature and commissioned on time. |
| C1.3 Emitter FID decision complexity | Data certainty at investment committee stage | Orange | A2 | B1 | Investment committees require credible scope, cost and risk evidence. |
| C1.3 Emitter FID decision complexity | Synchronisation with transport and storage commitments | Orange | C1 | A2 | Emitter FID depends on simultaneous commitments in transport and storage. |
| C1.3 Emitter FID decision complexity | Emitter permit & utilities complexity | Orange | C2 | A2 | Local permits and utilities (power/cooling/water) can block capture FID. |
| C1.4 Total volume commitment reliability | Volume change triggering chain-wide re-studies (flow assurance, hydraulics) | Green | A2 | C1 | Volume changes trigger chain-wide recalculation (hydraulics/flow assurance). |
| C1.4 Total volume commitment reliability | Timing mismatch between emitter ramp-up and chain availability | Green | C1 | D2 | Emitter ramp-up may not match chain start, causing underutilisation or delay. |
| C1.4 Total volume commitment reliability | Anchor-emitter dependency effects | Green | A2 | D2 | Dependence on anchor customers; loss harms volumes, tariffs and confidence. |
| C1.4 Total volume commitment reliability | Porthos start-up credibility as market confidence trigger | Orange | B2 | A2 | Porthos start-up provides proof and confidence; failure reduces Aramis commitments. |
| C2.1 Infrastructure synchronisation across feeder assets | Feeder pipeline readiness alignment | Green | C1 | D2 | Feeder pipelines interfaces must be ready for start-up and ramp-up. |
| C2.1 Infrastructure synchronisation across feeder assets | Feeder shipping readiness alignment | Red | C1 | D2 | Feeder ships interfaces must be ready for start-up and ramp-up. |
| C2.1 Infrastructure synchronisation across feeder assets | Optional feeder corridor FID independence (DRC & DSC) | Green | D1 | C1 | DRC/DSC may connect later; Aramis must reach FID without them. |
| C2.1 Infrastructure synchronisation across feeder assets | Dependency on collection hub commissioning milestones | Orange | C1 | D2 | Start-up depends on commissioning milestones of hub/terminal assets. |
| C2.1 Infrastructure synchronisation across feeder assets | Sequencing uncertainty between parallel feeder developments | Red | B1 | C1 | Parallel projects may deliver asynchronously, creating stranded capacity. |
| C2.2 Routing flexibility and competition exposure | Shipping as alternative routing option | Orange | A2 | B2 | Shipping offers a route outside the pipeline, affecting choice and volumes. |
| C2.2 Routing flexibility and competition exposure | Volume leakage through arbitrage to non-Aramis routes | Red | A2 | B2 | Emitters can route $CO_2$ to other hubs/stores, reducing Aramis utilisation. |
| C2.2 Routing flexibility and competition exposure | Impact of routing optionality on tariff stability | Orange | D2 | A2 | More optionality → more volatile volumes → less stable €/t tariff. |
| C2.3 Balancing and contingency arrangements | Existence of functional balancing mechanisms | Green | B1 | D2 | Balancing arrangements must exist to stabilise the chain during outages/mismatches. |
| C2.3 Balancing and contingency arrangements | Dependency of balancing on real spare capacity | Orange | D2 | C1 | Balancing works only if genuine spare capacity or alternatives exist. |
| C2.3 Balancing and contingency arrangements | Temporary rerouting/storage feasibility under outage scenarios | Orange | C1 | D2 | Practical feasibility to reroute temporarily or store elsewhere during outages. |
| C3.1 $CO_2$Next system readiness & commitment | $CO_2$Next commissioning alignment with Aramis start-up | Red | C1 | D2 | $CO_2$Next delivery must align with Aramis start-up schedule. |
| C3.1 $CO_2$Next system readiness & commitment | $CO_2$Next interface compatibility (pressure, quality, flow envelope) | Red | C1 | D1 | Pressure/quality/flow must remain within tolerances for safe handover CO2Next. |
| C3.1 $CO_2$Next system readiness & commitment | Availability attribution between $CO_2$Next, Porthos and Aramis | Green | B1 | D2 | Rules to attribute downtime determine payments and liability allocation. |
| C3.1 $CO_2$Next system readiness & commitment | Back-to-back availability and payment logic | Green | B1 | D2 | Availability and payment terms must mirror end-to-end to avoid mismatches. |
| C3.1 $CO_2$Next system readiness & commitment | Cost allocation under interface design changes | Green | B1 | D1 | Who pays for interface changes (pressure/spec) must be contractually defined. |
| C3.1 $CO_2$Next system readiness & commitment | $CO_2$Next goverment volume guarentee (Vollooprisico) | Green | B2 | B1 | Government underwrites early utilisation shortfall to keep €/t affordable in start-up for CO2 Next |
| C3.2 Compression capacity, availability & redundancy | Availability targets as contractual payment condition | Green | B1 | B2 | Payment is tied to meeting availability targets (credits/non-payment otherwise). |
| C3.2 Compression capacity, availability & redundancy | Redundancy evaluation decision rules (design freeze gate) | Orange | D1 | D2 | Decision rule on when to evaluate/select redundancy to meet availability. |
| C3.2 Compression capacity, availability & redundancy | Availability shortfall remedy mechanisms | Orange | B1 | D2 | Contract remedy mechanism (recovery plan/investment) when availability falls below target. |
| C3.2 Compression capacity, availability & redundancy | Attribution rules for downtime across chain parties | Orange | B1 | D2 | Downtime definitions and attribution determine who bears financial impact. |
| C3.3 Porthos system integration & readiness | Porthos commissioning milestones versus Aramis ramp-up | Orange | C1 | D2 | Porthos milestones must fit Aramis ramp-up trajectory and volumes. |
| C3.3 Porthos system integration & readiness | Control room integration and decision authority | Green | B1 | D1 | Who controls what (control room) and with what authority is critical for safe ops. |
| C3.3 Porthos system integration & readiness | $CO_2$ specification alignment and change governance | Orange | C1 | B1 | Chain-wide governance to harmonise the spec and manage changes safely. |
| C3.3 Porthos system integration & readiness | Availability definition mirroring across systems | Orange | B1 | D2 | Consistent availability definition (windows/exclusions) across systems prevents disputes. |
| C3.3 Porthos system integration & readiness | Porthos interface compatibility (pressure, quality, flow envelope) | Orange | C1 | D1 | Pressure/quality/flow must remain within tolerances for safe handover Porthos. |
| C3.4 Design constraints from external system drivers | Nitrogen constraints as design and execution driver | Orange | A1 | C2 | Nitrogen rules constrain execution/design and can block permits or schedule. |
| C3.4 Design constraints from external system drivers | Electrification requirements affecting layout and phasing | Orange | A1 | C1 | Electrification requirements affect layout choices, phasing and construction methods. |
| C3.4 Design constraints from external system drivers | Micro-tunnel removal obligations | Red | A1 | D1 | Removal/end-of-life obligations influence design choices and future cost exposure. |
| C3.4 Design constraints from external system drivers | Code/regulation gap for dense-phase $CO_2$ (compliant ≠ safe) | Green | A1 | D1 | Existing codes do not cover dense-phase $CO_2$; extra evidence and design guardrails are needed. |
| C4.1 Offshore transport integrity and operability | Pressure and temperature envelope stability | Orange | D2 | C1 | Maintaining T/p to avoid phase change and preserve integrity. |
| C4.1 Offshore transport integrity and operability | Dense-phase $CO_2$ transport behaviour under upset conditions | Green | D2 | D1 | Transient/upset behaviour drives two-phase risk and operating limits. |
| C4.1 Offshore transport integrity and operability | Corrosion and impurity sensitivity | Orange | D1 | C1 | Impurities/water can trigger corrosion/acid formation, shaping design and spec. |
| C4.2 Safety-by-interface dependency | Reliance on upstream/downstream protection systems | Green | C1 | D2 | Aramis relies on partner protections (trips/PSVs) to protect integrity. |
| C4.2 Safety-by-interface dependency | Interface-defined safeguards and trip logic | Green | C1 | B1 | Safeguards and trip logic must be consistent and tested across parties. |
| C4.2 Safety-by-interface dependency | Loss of autonomy under partner system failures | Orange | D2 | C1 | Partner failures can stop Aramis; limited control increases vulnerability. |
| C4.2 Safety-by-interface dependency | Failure-mode discovery & safety case evidence (no rulebook) | Orange | B2 | C2 | New failure modes must be identified and evidenced in the safety case for authorities. |
| C4.3 Single-point-of-failure exposure | Trunkline downtime halting entire value chain | Green | D2 | A2 | Backbone outage stops all flows; the key single-point-of-failure exposure. |
| C4.3 Single-point-of-failure exposure | Recovery and restart complexity after offshore outage | Green | C1 | D2 | Recovery/restart requires time and procedures, affecting mean time to repair and payments. |
| C4.4 Off-spec and upset handling procedures | Integrated control philosofy | Orange | D1 | B1 | A unified control philosophy (setpoints/handoffs) is needed for safe integrated operation. |
| C4.4 Off-spec and upset handling procedures | Stop / vent / recovery decision logic | Orange | C1 | D2 | Predefined decision logic for isolation, depressurisation/venting and recovery. |
| C4.4 Off-spec and upset handling procedures | Impact of upset handling on chain availability | Orange | D2 | A2 | Upset handling choices determine downtime, payments and confidence. |
| C4.5 Execution capacity & schedule realism | Contractor scarcity & schedule gate | Red | C1 | A2 | Scarce contractors/vessels gate the schedule and can drive cost and FID timing. |
| C4.5 Execution capacity & schedule realism | Safety & schedule compromise mechansim | Orange | B2 | C1 | Schedule pressure can trigger shortcuts, raising safety and incident risk. |
| C4.5 Execution capacity & schedule realism | EU tender/procurement rules fit with execution | Orange | A1 | C1 | Tender frameworks fit poorly with offshore risk; shapes contracting strategy. |
| C5.1 Storage capacity portfolio adequacy | Availability of sufficient developed fields | Green | D2 | A2 | Enough fields must be developed in time to accept injection volumes. |
| C5.1 Storage capacity portfolio adequacy | Phasing of field development versus Aramis throughput | Green | C1 | D2 | Field phasing must match pipeline throughput and ramp-up to avoid bottlenecks. |
| C5.1 Storage capacity portfolio adequacy | Ramp-up mismatch risks | Green | C1 | D2 | Asynchronous field start-up vs transport causes underutilisation or constraints. |
| C5.1 Storage capacity portfolio adequacy | Opportunity funnel potential prospects | Red | A2 | B2 | The funnel of storage opportunities determines future capacity confidence. |
| C5.2 Injectivity and operational flexibility | Injectivity as hard system constraint | Orange | D2 | D1 | Injectivity limits offtake; without injection the chain stops. |
| C5.2 Injectivity and operational flexibility | Limited buffering capacity (what goes in must go down) | Green | D2 | C1 | Limited buffering creates immediate dependence on continuous injection. |
| C5.2 Injectivity and operational flexibility | Upset tolerance and contingency injection options | Orange | C1 | B2 | Options to pause, reroute or inject contingently during anomalies. |
| C5.3 Store availability and revenue exposure | Store availability dependency on subsidy and revenue streams | Orange | D2 | A1 | Storage downtime can hit subsidy/revenue, affecting willingness to commit. |
| C5.3 Store availability and revenue exposure | Financial exposure under injection downtime | Orange | B1 | B2 | Who pays/compensates during downtime drives risk and contract design. |
| C5.4 $CO_2$ specification bounded by storage constraints | License-driven composition limits | Orange | C2 | B2 | Storage permits bound composition; off-spec can cause rejection/shutdown. |
| C5.4 $CO_2$ specification bounded by storage constraints | Upstream design implications of storage acceptance criteria | Green | C1 | D1 | Storage acceptance criteria propagate upstream into spec, compression and operations. |
| C5.5 Stores investment commitment logic | Risk reward balance thresholds for private storage parties | Orange | A2 | D2 | Private stores invest only if risk reward balance is sufficient. |
| C5.5 Stores investment commitment logic | Launch stores 5 megaton start-up momentum | Orange | C1 | B1 | Store FID depends on Aramis/emitters; mutual dependence delays decisions. |
| C5.5 Stores investment commitment logic | Stores competition by design to enhance the market for CCS commitment | Orange | B1 | C1 | Stores may coordinate to avoid destructive competition and increase investment certainty. |
| C5.5 Stores investment commitment logic | Implementation requirements Net Zero Industry Act | Orange | C2 | B2 | Oil and Gas companies may be obliged to develop storages facilities |
| C6.1 Tariff formation and cost discovery | BAFO / CFT timing and late tariff certainty | Green | A2 | D2 | Tariff certainty arrives only after tender/BAFO, affecting FID timing. |
| C6.1 Tariff formation and cost discovery | Emerging tariff build-up from chain-wide capex/opex excluding emitters | Orange | D1 | A2 | Tariff is the sum of costs across the chain; each part affects the final price. |
| C6.1 Tariff formation and cost discovery | Sensitivity of tariff to downstream and upstream costs towards emitters | Orange | D1 | A2 | Cost swings in stores/terminals/pipeline directly shift €/t. |
| C6.1 Tariff formation and cost discovery | Aramis government volume guarentee (Vollooprisico) | Green | B2 | A1 | Government underwrites early utilisation shortfall to keep €/t affordable in start-up for Aramis |
| C6.2 System integration governance | System integrator role definition and mandate | Orange | B1 | B2 | Integrator scope/mandate defines who carries integration responsibility and risk. |
| C6.2 System integration governance | Change-control authority across interfaces | Green | B1 | C1 | Who can approve interface changes; prevents uncontrolled scope churn. |
| C6.2 System integration governance | Back-to-back mirroring of operational and financial logic | Green | B1 | D2 | Operational rules must be financially and legally consistent across the chain. |
| C6.3 FID synchronisation across the value chain | Chicken-and-egg dynamics across >20 parties | Orange | C1 | B1 | Many parties wait for each other; without simultaneous commits, no FID breakthrough. |
| C6.3 FID synchronisation across the value chain | Mutual dependence of investment commitments | Red | C1 | A2 | FIDs are coupled (emitters–transport–stores); delay by one affects the whole chain. |
| C6.4 Chain-wide contractual coherence | Back-to-back payment logic | Orange | B1 | D2 | Payment terms must flow end-to-end or risk remains stranded with one party. |
| C6.4 Chain-wide contractual coherence | Availability mirroring across contracts | Orange | B1 | D2 | One availability definition prevents disputes and residual payment exposure. |
| C6.4 Chain-wide contractual coherence | Residual exposure prevention | Green | B1 | D2 | Mechanisms to prevent risk gaps (carve-outs, caps, true-ups). |
| C6.4 Chain-wide contractual coherence | IOA/IOE/TOI/EMOC | Orange | B1 | C1 | Integrated operating agreements and open items steer safe operations and change control. |
| C6.5 Strategic optionality and system rigidity | Limited ability to phase or decouple investments | Orange | D1 | C1 | The system is hard to split; phasing can create stranded assets and tariff stress. |
| C6.5 Strategic optionality and system rigidity | Lock-in effects once trunk infrastructure is committed | Green | D1 | A2 | After backbone commitment, changes are costly; re-FEED hurts schedule and support. |
| C6.5 Strategic optionality and system rigidity | Policy-defined investment horizon | Orange | A1 | D1 | Policy caps/horizons (e.g., subsidy duration) constrain long-term business cases. |
| C6.6 Permitting finality and legal operability | Legal finality versus reversibility | Red | C2 | B2 | Definitive vs irrevocable; legal challengeability shapes FID willingness. |
| C6.6 Permitting finality and legal operability | Interdependency of permits across chain segments | Orange | C2 | B1 | Permits act as a bundle; one weak link can block the whole set. |
| C6.6 Permitting finality and legal operability | Temporal alignment of permit validity with FID and execution | Orange | B2 | C1 | Permit timing/validity must match FID and execution windows. |
| C6.6 Permitting finality and legal operability | Authority acceptance of integrated system safety cases | Orange | B2 | D1 | Authorities must accept the integrated safety case before FID/commissioning. |

## B.3. Overview Viability & Enabling Dependencies - Aramis

1. Industrial decarbonisation business case stability enabling FID for capture investments
   - C1.1 Cost-effectiveness of CCS versus alternatives
   - C1.2 $CO_2$ specification as access and cost criterion
   - C1.3 Emitter FID decision complexity
   - C1.4 Total volume commitment reliability
2. Timely development & capacity of feeders/terminals infrastructure
   - C2.1 Infrastructure synchronisation across feeder assets
   - C2.2 Routing flexibility and competition exposure
   - C2.3 Balancing and contingency arrangements
3. Compression and interface readiness for $CO_2$ flow to the trunkline
   - C3.2 Compression capacity, availability & redundancy
   - C3.3 Porthos system integration & readiness
   - C3.4 Design constraints from external system drivers
4. Pipeline & D-Hub distribution readiness
   - C4.1 Offshore transport integrity and operability
   - C4.2 Safety-by-interface dependency
   - C4.3 Single-point-of-failure exposure
   - C4.4 Off-spec and upset handling procedures
   - C4.5 Execution capacity & schedule realism
5. Storage capacity & operability readiness
   - C5.1 Storage capacity portfolio adequacy
   - C5.2 Injectivity and operational flexibility
   - C5.3 Store availability and revenue exposure
   - C5.4 $CO_2$ specification bounded by storage constraints
   - C5.5 Stores investment commitment logic
6. System-wide strategic enablers & constraints
   - C6.1 Tariff formation and cost discovery
   - C6.2 System integration governance
   - C6.3 FID synchronisation across the value chain
   - C6.4 Chain-wide contractual coherence
   - C6.5 Strategic optionality and system rigidity
   - C6.6 Permitting finality and legal operability

## B.4. Context Information - Viability Dependencies Aramis

### C1 Industrial decarbonisation business-case stability enabling FID for capture investments

**Value-chain segment:** Emitters / Capture

**Definition (viability dependency):** Industrial decarbonisation business-case stability enabling FID for capture investments.

**Description & context:** Chain initiation depends on emitters' business cases and their willingness to fund capture and contract volumes; without timely capture FIDs and committed throughput, the open-access system risks under-utilisation and tariff pressure.

**MDF placement:** Primary A2 (Market & Industrial Commitment); Secondary D2 (Required Value-Chain Performance)

- **Primary context:** A2 (Market & Industrial Commitment) This dependency is primarily placed here because uncertainty concerns emitters' investment and commitment behaviour (business cases, willingness-to-pay, capture FID cadence), which frames how viability is interpreted at governance level.
- **Secondary context:** D2 (Required Value-Chain Performance) If uncertainty in this dependency develops, it becomes governance-relevant in D2, where implications for throughput, utilisation and service performance must be addressed.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: Low
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Enabling: 4
- # Sub-dependencies (total): 20
- # Subs Red / Orange / Green: 3 / 12 / 5

### C2 Timely development & capacity of feeders/terminals infrastructure

**Value-chain segment:** Feeder Transport

**Definition (viability dependency):** Timely development & capacity of feeders/terminals infrastructure.

**Description & context:** Feeder pipelines, terminals, and shipping must be ready on time and aligned to deliver the planned volumes to the Aramis hub; desynchronisation causes volume dips that propagate into hub/offshore operability and performance.

**MDF placement:** Primary C1 (Multi-Project & Interface Complexity); Secondary D2 (Required Value-Chain Performance)

- **Primary context:** C1 (Mult-Project & Interface Complexity) Uncertainty is primarily about synchronisation and compatibility across feeder assets, terminals and shipping windows, which determines interpretation of readiness at governance level.
- **Secondary context:** D2 (Required Value-Chain Performance) If this uncertainty develops, it becomes governance-relevant in D2, where volume delivery, operability and ramp-up performance are managed.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: High
- Feedback strength: Medium
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Enabling: 3
- # Sub-dependencies (total): 11
- # Subs Red / Orange / Green: 3 / 5 / 3

### C3 Compression and interface readiness for $CO_2$ flow to the trunkline

**Value-chain segment:** Onshore Hub & Compression

**Definition (viability dependency):** Compression & interface readiness for trunkline injection.

**Description & context:** Onshore compression and interfaces (pressure/temperature/$CO_2$ specification/controls) must be demonstrably safe, available, and contractually workable pre-FID to ensure stable $CO_2$ flow into the trunkline.

**MDF placement:** Primary D2 (Required Value-Chain Performance); Secondary C1 (Multi-Project & Interface Complexity)

- **Primary context:** D2 (Required Value-Chain Performance) Uncertainty is primarily about injection availability and operating envelopes (pressure, temperature, specification, controls) that define the performance requirement.
- **Secondary context:** C1 (Mult-Project & Interface Complexity) If this uncertainty develops, it becomes governance-relevant in C1, where interface alignment, interlocking procedures and commissioning coordination are addressed.

**Vulnerability characteristics:**

- Autonomy: High
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Enabling: 5
- # Sub-dependencies (total): 15
- # Subs Red / Orange / Green: 3 / 9 / 7

### C4 Pipeline & Distribution-Hub distribution readiness

**Value-chain segment:** Offshore

**Definition (viability dependency):** Pipeline & Distribution-Hub distribution readiness.

**Description & context:** The offshore backbone (trunkline + Distribution Hub + distribution lines) must operate safely as an integrated system and stay robust under disturbances and off-spec conditions; failures directly affect availability, contractual performance, and chain-wide confidence.

**MDF placement:** Primary D2 (Required Value-Chain Performance); Secondary C1 (Multi-Project & Interface Complexity)

- **Primary context:** D2 (Required Value-Chain Performance) Uncertainty is primarily about the offshore backbone's ability to deliver the contracted service as an integrated system.
- **Secondary context:** C1 (Mult-Project & Interface Complexity) If uncertainty develops, it becomes governance-relevant in C1, where distribution interfaces, routing logic and operational coordination are interpreted.

**Vulnerability characteristics:**

- Autonomy: High
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Enabling: 4
- # Sub-dependencies (total): 19
- # Subs Red / Orange / Green: 1 / 9 / 5

### C5 Storage capacity & operability readiness

**Value-chain segment:** Storage

**Definition (viability dependency):** Storage capacity & operability readiness.

**Description & context:** Storage must provide sufficient capacity and injectivity, with timely investment commitment and operational flexibility for ramp-up; uncertainty here is chain-critical for continuity and revenue structures.

**MDF placement:** Primary D2 (Required Value-Chain Performance); Secondary C1 (Multi-Project & Interface Complexity)

- **Primary context:** D2 (Required Value-Chain Performance) Uncertainty is primarily about available storage capacity, injectivity, and operational flexibility that underpin required performance.
- **Secondary context:** C1 (Multi-Project & Interface Complexity) If uncertainty develops, it becomes governance-relevant in C1, where connection timing, integration steps and operational hand-offs are addressed.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: High
- Feedback strength: High
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Enabling: 5
- # Sub-dependencies (total): 15
- # Subs Red / Orange / Green: 1 / 9 / 5

### C6 System-wide strategic enablers & constraints

**Value-chain segment:** Cross-Cutting

**Definition (viability dependency):** System-wide strategic enablers & constraints.

**Description & context:** Chain-wide conditions—synchronisation of FIDs, end-to-end contractual coherence (back-to-back/IOA), strategic optionality versus rigidity, and permitting status/legal operability—shape the strategic resilience of the open-access system up to 22 Mtpa.

**MDF placement:** Primary B2 (External Legitimacy & Political Durability); Secondary B1 (Inter-Organisational Governance Alignment)

- **Primary context:** B2 (External Legitimacy & Political Durability) Uncertainty is primarily about societal/political defensibility, regulatory durability and public value framing of the open-access system.
- **Secondary context:** B1 (Inter-Organisational Governance Alignment) If uncertainty develops, it becomes governance-relevant in B1, where mandates, decision rights and chain-wide coordination are interpreted.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: High
- Feedback strength: High
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Enabling: 6
- # Sub-dependencies (total): 20
- # Subs Red / Orange / Green: 2 / 12 / 6

## B.5. Context Information - Enabling Dependencies Aramis

**C1 - Industrial decarbonisation business-case stability enabling FID for capture investments**

### C1.1 Cost-effectiveness of CCS versus alternatives

**Value-chain segment:** Emitters / Capture
**Definition (enabling dependency):** Cost-effectiveness of CCS versus alternatives.
**Description & context:** Drivers determining whether CCS outperforms paying EU-ETS costs/penalties or pursuing alternative routing and abatement options.
**MDF placement:** Primary A2; Secondary A1
**Context interpretation:**

- **Primary context:** A2. Uncertainty is primarily about relative economics and market choice (CCS versus ETS costs/penalties or alternative routes), which frames viability at governance level.
- **Secondary context:** A1. If uncertainty develops, it becomes governance-relevant in A1, where policy, pricing frameworks and incentive design are interpreted.

**Vulnerability characteristics:**

- Autonomy: High
- Bandwidth: High
- Feedback strength: High
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Sub-dependencies (total): 6
- # Subs Red / Orange / Green: 1 / 0 / 5

### C1.2 $CO_2$ specification as access and cost criterion

**Value-chain segment:** Emitters / Capture
**Definition (enabling dependency):** $CO_2$ specification as access and cost criterion.
**Description & context:** $CO_2$ quality gates both system access and capture cost; specifications that are too tight or too loose affect viability.
**MDF placement:** Primary D1; Secondary A2
**Context interpretation:**

- **Primary context:** D1. Uncertainty is primarily about the conceptual specification and design envelope that defines admissible $CO_2$ quality and access.
- **Secondary context:** A2. If uncertainty develops, it becomes governance-relevant in A2, where capture economics and emitter participation conditions are discussed.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 5
- # Subs Red / Orange / Green: 2 / 1 / 2

### C1.3 Emitter FID decision complexity

**Value-chain segment:** Emitters / Capture

**Definition (enabling dependency):** Emitter FID decision complexity.

**Description & context:** Emitter investment decisions require timely data, permits, utilities, and cross-chain alignment to reduce decision friction and delay.

**MDF placement:** Primary A2; Secondary B1

**Context interpretation:**

- **Primary context:** A2. Uncertainty is primarily about the decision environment of emitters (information, permits, utilities, investment criteria).
- **Secondary context:** B1. If uncertainty develops, it becomes governance-relevant in B1, where alignment, roles and decision rights across organisations are addressed.

**Vulnerability characteristics:**

- Autonomy: High
- Bandwidth: High
- Feedback strength: High
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Sub-dependencies (total): 5
- # Subs Red / Orange / Green: 0 / 5 / 0

### C1.4 Total volume commitment reliability

**Value-chain segment:** Emitters / Capture

**Definition (enabling dependency):** Total volume commitment reliability.

**Description & context:** Stability of contracted volumes and timing; changes trigger re-work and tariff/credibility effects across the chain.

**MDF placement:** Primary D2; Secondary C1

**Context interpretation:**

- **Primary context:** D2. Uncertainty is primarily about volume delivery and timing that underpin required service levels.
- **Secondary context:** C1. If uncertainty develops, it becomes governance-relevant in C1, where interfaces, dispatch logic and operational arrangements are handled.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 4
- # Subs Red / Orange / Green: 0 / 1 / 3

#### C2 - Timely development & capacity of feeders/terminals infrastructure

### C2.1 Infrastructure synchronisation across feeder assets

**Value-chain segment:** Feeder Transport

**Definition (enabling dependency):** Infrastructure synchronisation across feeder assets.

**Description & context:** Alignment of feeder pipelines, terminals and shipping windows with Aramis start-up and growth trajectories.

**MDF placement:** Primary C1; Secondary D2

**Context interpretation:**

- **Primary context:** C1. Uncertainty is primarily about multi-asset alignment (feeder pipelines, terminals, shipping windows).
- **Secondary context:** D2. If uncertainty develops, it becomes governance-relevant in D2, where throughput and operability are interpreted.

**Vulnerability characteristics:**

- Autonomy: High
- Bandwidth: High
- Feedback strength: Medium
- Resulting vulnerability (ring): INNER

**Other information:**

- # Sub-dependencies (total): 5
- # Subs Red / Orange / Green: 2 / 1 / 2

### C2.2 Routing flexibility and competition exposure

**Value-chain segment:** Feeder Transport

**Definition (enabling dependency):** Routing flexibility and competition exposure.

**Description & context:** Alternative routes—particularly shipping to other sinks—can leak volumes and undermine tariff formation and commitments.

**MDF placement:** Primary A2; Secondary D2

**Context interpretation:**

- **Primary context:** A2. Uncertainty is primarily about market routing choices and competition for volumes.
- **Secondary context:** D2. If uncertainty develops, it becomes governance-relevant in D2, where utilisation and tariff resilience are discussed.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: Medium
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 3
- # Subs Red / Orange / Green: 1 / 2 / 0

### C2.3 Balancing and contingency arrangements

**Value-chain segment:** Feeder Transport

**Definition (enabling dependency):** Balancing and contingency arrangements.

**Description & context:** Mechanisms to absorb outages via spare capacity, re-routing, or temporary takeover.

**MDF placement:** Primary B1; Secondary D2

**Context interpretation:**

- **Primary context:** B1. Uncertainty is primarily about inter-organisational rules for balancing, fallback and temporary takeover.
- **Secondary context:** D2. If uncertainty develops, it becomes governance-relevant in D2, where service continuity and KPIs are interpreted.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: High
- Feedback strength: Low
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Sub-dependencies (total): 3
- # Subs Red / Orange / Green: 0 / 2 / 1

**C3 - Compression and interface readiness for $CO_2$ flow to the trunkline**

### C3.1 $CO_2$Next system readiness & commitment

**Value-chain segment:** Onshore Hub & Compression

**Definition (enabling dependency):** $CO_2$Next system readiness & commitment.

**Description & context:** $CO_2$Next must be delivered on time, be technically compatible, and lock in availability/cost terms to underpin chain operability.

**MDF placement:** Primary C1; Secondary D2

**Context interpretation:**

- **Primary context:** C1. Uncertainty is primarily about interface readiness and schedule coordination with $CO_2$Next as an external system.
- **Secondary context:** D2. If uncertainty develops, it becomes governance-relevant in D2, where injection availability and stable flow are addressed.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 6
- # Subs Red / Orange / Green: 2 / 0 / 4

### C3.2 Compression capacity, availability & redundancy

**Value-chain segment:** Onshore Hub & Compression

**Definition (enabling dependency):** Compression capacity, availability & redundancy.

**Description & context:** Compression availability is an FID condition; rules for redundancy, remedies and attribution must be explicit.

**MDF placement:** Primary D2; Secondary B1

**Context interpretation:**

- **Primary context:** D2. Uncertainty is primarily about availability and redundancy rules required to meet performance obligations.
- **Secondary context:** B1. If uncertainty develops, it becomes governance-relevant in B1, where accountabilities, remedies and escalation paths are considered.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: Low
- Feedback strength: High

- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 4
- # Subs Red / Orange / Green: 0 / 3 / 1

### C3.3 Porthos system integration & readiness

**Value-chain segment:** Onshore Hub & Compression

**Definition (enabling dependency):** Porthos system integration & readiness.

**Description & context:** Porthos start-up, governance, and specification/availability alignment determine chain operability at the onshore hub.

**MDF placement:** Primary C1; Secondary D2

**Context interpretation:**

- **Primary context:** C1. Uncertainty is primarily about integration governance and interface alignment with Porthos at the hub.
- **Secondary context:** D2. If uncertainty develops, it becomes governance-relevant in D2, where operability and availability are interpreted.

**Vulnerability characteristics:**

- Autonomy: High
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 5
- # Subs Red / Orange / Green: 0 / 4 / 1

### C3.4 Design constraints from external system drivers

**Value-chain segment:** Onshore Hub & Compression

**Definition (enabling dependency):** Design constraints from external system drivers.

**Description & context:** Exogenous requirements and constraints shape design choices, deliverability, and approval evidence.

**MDF placement:** Primary A1; Secondary C2

**Context interpretation:**

- **Primary context:** A1. Uncertainty is primarily about exogenous regulatory/system requirements that constrain design choices.
- **Secondary context:** C2. If uncertainty develops, it becomes governance-relevant in C2, where project-specific approvals and evidence are discussed.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: Medium
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 4
- # Subs Red / Orange / Green: 1 / 2 / 1

**C4 - Pipeline & Distribution-Hub distribution readiness**

### C4.1 Offshore transport integrity and operability

**Value-chain segment:** Offshore

**Definition (enabling dependency):** Offshore transport integrity and operability.

**Description & context:** Dense-phase integrity—temperature/pressure envelope, corrosion behaviour, and upset response—governs steady-state operability.

**MDF placement:** Primary D1; Secondary C1

**Context interpretation:**

- **Primary context:** D1. Uncertainty is primarily about the system concept and material/operating envelope governing transport integrity.
- **Secondary context:** C1. If uncertainty develops, it becomes governance-relevant in C1, where operational coordination and interface procedures are handled.

**Vulnerability characteristics:**

- Autonomy: High
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 3
- # Subs Red / Orange / Green: 0 / 2 / 1

### C4.2 Safety-by-interface dependency

**Value-chain segment:** Offshore

**Definition (enabling dependency):** Safety-by-interface dependency.

**Description & context:** Safety depends on partner interfaces, safeguards, trip logic and safety-case evidence across assets.

**MDF placement:** Primary C1; Secondary D2

**Context interpretation:**

- **Primary context:** C1. Uncertainty is primarily about safety arrangements at interfaces (safeguards, trip logic, coordination across assets).
- **Secondary context:** D2. If uncertainty develops, it becomes governance-relevant in D2, where availability and safe service delivery are interpreted.

**Vulnerability characteristics:**

- Autonomy: High
- Bandwidth: High
- Feedback strength: High
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Sub-dependencies (total): 4
- # Subs Red / Orange / Green: 0 / 2 / 2

### C4.3 Single-point-of-failure exposure

**Value-chain segment:** Offshore

**Definition (enabling dependency):** Single-point-of-failure exposure.

**Description & context:** Trunkline failure halts the chain; recovery and mean-time-to-repair drive commercial and operational impacts.

**MDF placement:** Primary D2; Secondary B2

**Context interpretation:**

- **Primary context:** D2. Uncertainty is primarily about service continuity under component or node failure on the offshore backbone.
- **Secondary context:** B2. If uncertainty develops, it becomes governance-relevant in B2, where external assurance, accountability and public/political scrutiny are addressed.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 2
- # Subs Red / Orange / Green: 0 / 0 / 2

### C4.4 Off-spec and upset handling procedures

**Value-chain segment:** Offshore

**Definition (enabling dependency):** Off-spec and upset handling procedures.

**Description & context:** Clear decision logic for stop/vent/recovery and control responses is required to manage off-spec events.

**MDF placement:** Primary C1; Secondary D2

**Context interpretation:**

- **Primary context:** C1. Uncertainty is primarily about cross-party decision logic and procedures for managing off-spec and upset conditions.
- **Secondary context:** D2. If uncertainty develops, it becomes governance-relevant in D2, where service KPIs and recovery expectations are interpreted.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 3
- # Subs Red / Orange / Green: 0 / 3 / 0

### C4.5 Execution capacity & schedule realism

**Value-chain segment:** Offshore

**Definition (enabling dependency):** Execution capacity & schedule realism.

**Description & context:** Contractor/vessel availability and realistic P-curves determine pre-FID deliverability and schedule credibility.

**MDF placement:** Primary C1; Secondary A2

**Context interpretation:**

- **Primary context:** C1. Uncertainty is primarily about contractor resources, vessels and execution windows across projects.

- **Secondary context:** A2. If uncertainty develops, it becomes governance-relevant in A2, where cost exposure, market confidence and commitment timing are discussed.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: Medium
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 3
- # Subs Red / Orange / Green: 1 / 2 / 0

**C5 - Storage capacity & operability readiness**

### C5.1 Storage capacity portfolio adequacy

**Value-chain segment:** Storage

**Definition (enabling dependency):** Storage capacity portfolio adequacy.

**Description & context:** A portfolio of fields must be sufficient and timely developed to support throughput and ramp-up.

**MDF placement:** Primary D2; Secondary C1

**Context interpretation:**

- **Primary context:** D2. Uncertainty is primarily about portfolio sufficiency and delivery timing to support throughput and ramp-up.
- **Secondary context:** C1. If uncertainty develops, it becomes governance-relevant in C1, where connection schedules and operational integration are addressed.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: High
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 4
- # Subs Red / Orange / Green: 1 / 0 / 3

### C5.2 Injectivity and operational flexibility

**Value-chain segment:** Storage

**Definition (enabling dependency):** Injectivity and operational flexibility.

**Description & context:** Injectivity and flexibility determine whether volumes can be injected safely without excessive buffering.

**MDF placement:** Primary D2; Secondary D1

**Context interpretation:**

- **Primary context:** D2. Uncertainty is primarily about injectivity performance and operational flexibility required by the chain.
- **Secondary context:** D1. If uncertainty develops, it becomes governance-relevant in D1, where operating windows and concept tolerances are interpreted.

**Vulnerability characteristics:**

- Autonomy: Low

- Bandwidth: Low
- Feedback strength: Low
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 3
- # Subs Red / Orange / Green: 0 / 2 / 1

### C5.3 Store availability and revenue exposure

**Value-chain segment:** Storage
**Definition (enabling dependency):** Store availability and revenue exposure.
**Description & context:** Injection downtime affects revenues/subsidies and pressures chain payments under send-or-pay logics.
**MDF placement:** Primary D2; Secondary A1
**Context interpretation:**

- **Primary context:** D2. Uncertainty is primarily about availability obligations and revenue exposure under send-or-pay arrangements.
- **Secondary context:** A1. If uncertainty develops, it becomes governance-relevant in A1, where subsidy/compliance and regulatory accounting are discussed.

**Vulnerability characteristics:**

- Autonomy: High
- Bandwidth: High
- Feedback strength: Medium
- Resulting vulnerability (ring): INNER

**Other information:**

- # Sub-dependencies (total): 2
- # Subs Red / Orange / Green: 0 / 2 / 0

### C5.4 $CO_2$ specification bounded by storage constraints

**Value-chain segment:** Storage
**Definition (enabling dependency):** $CO_2$ specification bounded by storage constraints.
**Description & context:** Storage licences bound composition; acceptance criteria propagate upstream into hub and capture design.
**MDF placement:** Primary C2; Secondary D1
**Context interpretation:**

- **Primary context:** C2. Uncertainty is primarily about project-specific approval of storage acceptance criteria.
- **Secondary context:** D1. If uncertainty develops, it becomes governance-relevant in D1, where upstream design/specification choices are interpreted.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: Low
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 2

- # Subs Red / Orange / Green: 0 / 1 / 1

### C5.5 Stores investment commitment logic

**Value-chain segment:** Storage

**Definition (enabling dependency):** Stores investment commitment logic.

**Description & context:** Storage parties invest only if risk-reward balance is acceptable and chain-wide FID alignment exists.

**MDF placement:** Primary A2; Secondary D2

**Context interpretation:**

- **Primary context:** A2. Uncertainty is primarily about storage investors' commitment logic (risk-reward, alignment of FIDs).
- **Secondary context:** D2. If uncertainty develops, it becomes governance-relevant in D2, where capacity ramp-up and service deliverability are addressed.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: High
- Feedback strength: Low
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Sub-dependencies (total): 4
- # Subs Red / Orange / Green: 0 / 4 / 0

**C6 - System-wide strategic enablers & constraints**

### C6.1 Tariff formation and cost discovery

**Value-chain segment:** Cross-Cutting

**Definition (enabling dependency):** Tariff formation and cost discovery.

**Description & context:** Tariff emerges from market prices and chain costs; it remains uncertain until cost estimates mature and volumes stabilise.

**MDF placement:** Primary D1; Secondary A2

**Context interpretation:**

- **Primary context:** D1. Uncertainty is primarily about tariff formation as a function of concept scope, cost structure and configuration.
- **Secondary context:** A2. If uncertainty develops, it becomes governance-relevant in A2, where willingness-to-pay and commitment conditions are interpreted.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 4
- # Subs Red / Orange / Green: 0 / 2 / 2

### C6.2 System integration governance

**Value-chain segment:** Cross-Cutting
**Definition (enabling dependency):** System integration governance.
**Description & context:** The integrator's mandate and change control determine safe and workable system integration across parties.
**MDF placement:** Primary B1; Secondary D1
**Context interpretation:**

- **Primary context:** B1. Uncertainty is primarily about integrator mandate, change control and decision rights across organisations.
- **Secondary context:** D1. If uncertainty develops, it becomes governance-relevant in D1, where system architecture and baselines are interpreted.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: High
- Feedback strength: High
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Sub-dependencies (total): 3
- # Subs Red / Orange / Green: 0 / 1 / 2

### C6.3 FID synchronisation across the value chain

**Value-chain segment:** Cross-Cutting
**Definition (enabling dependency):** FID synchronisation across the value chain.
**Description & context:** Decisions are interdependent; simultaneous start/commitment is critical pre-FID to avoid stranded exposure.
**MDF placement:** Primary C1; Secondary B2
**Context interpretation:**

- **Primary context:** C1. Uncertainty is primarily about multi-project milestone alignment needed for simultaneous commitment.
- **Secondary context:** B2. If uncertainty develops, it becomes governance-relevant in B2, where programme credibility and public sponsor commitments are interpreted.

**Vulnerability characteristics:**

- Autonomy: Medium
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 2
- # Subs Red / Orange / Green: 1 / 1 / 0

### C6.4 Chain-wide contractual coherence

**Value-chain segment:** Cross-Cutting
**Definition (enabling dependency):** Chain-wide contractual coherence.
**Description & context:** End-to-end contract coherence (back-to-back and IOA terms) avoids residual exposure at interfaces.
**MDF placement:** Primary B1; Secondary D2

**Context interpretation:**

- **Primary context:** B1. Uncertainty is primarily about end-to-end contractual consistency (back-to-back, IOA terms, remedies).
- **Secondary context:** D2. If uncertainty develops, it becomes governance-relevant in D2, where service levels, performance allocation and accountability are interpreted.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: High
- Feedback strength: High
- Resulting vulnerability (ring): MIDDLE

**Other information:**

- # Sub-dependencies (total): 4
- # Subs Red / Orange / Green: 0 / 3 / 1

### C6.5 Strategic optionality and system rigidity

**Value-chain segment:** Cross-Cutting
**Definition (enabling dependency):** Strategic optionality and system rigidity.
**Description & context:** Degree of lock-in; once committed, decoupling is difficult and politically sensitive.
**MDF placement:** Primary D1; Secondary A1
**Context interpretation:**

- **Primary context:** D1. Uncertainty is primarily about option sets embedded in the system concept (scalability, configurational choices) and the degree of lock-in.
- **Secondary context:** A1. If uncertainty develops, it becomes governance-relevant in A1, where policy, subsidy and regulatory constraints are interpreted.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: High
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 3
- # Subs Red / Orange / Green: 0 / 2 / 1

### C6.6 Permitting finality and legal operability

**Value-chain segment:** Cross-Cutting
**Definition (enabling dependency):** Permitting finality and legal operability.
**Description & context:** Legal finality and approval acceptance determine whether FID is both legally and practically feasible.
**MDF placement:** Primary C2; Secondary B2
**Context interpretation:**

- **Primary context:** C2. Uncertainty is primarily about project-specific legal finality and approval acceptance.
- **Secondary context:** B2. If uncertainty develops, it becomes governance-relevant in B2, where political durability and external legitimacy are interpreted.

**Vulnerability characteristics:**

- Autonomy: Low
- Bandwidth: Low
- Feedback strength: High
- Resulting vulnerability (ring): OUTER

**Other information:**

- # Sub-dependencies (total): 4
- # Subs Red / Orange / Green: 1 / 3 / 0

# C

# Risk Analysis Aramis

## C.1. From Risks in register to dependencies for analysis - Aramis

CONFIDENTIAL

## C.2. Reframing risks to dependencies

CONFIDENTIAL

## C.3. Unfolding of the risks with the sub-dependencies / development indicators

### C.3.1. R-X Example Elaborated per Sub-Dependency

In this section Risk-X wil be elaborated on how the sub-dependencies relate to the risk. After the example of R-X al risks and there unfolding of sub-dependencies is shown.

- **Industrial relocation or cross-border storage**
  Emitters may relocate or use foreign storage if Aramis is economically unattractive.
  *Stabilisation required:* Aramis must remain competitive relative to alternative jurisdictions.

- **Tariff versus ETS/penalty comparison**
  A low ETS price weakens the economic incentive for CCS, undermining commitment.
  *Stabilisation required:* A predictable ETS trajectory ensuring CCS retains economic advantage.

- **Volume-dependent tariff escalation (€/t sensitivity)**
  Lower committed volumes increase tariffs, creating a self-reinforcing instability.
  *Stabilisation required:* A tariff structure robust across multiple volume scenarios.

- **Subsidy sufficiency under cost-overrun scenarios**
  SDE++ directly shapes emitter business cases; cost overruns may erode subsidy adequacy.
  *Stabilisation required:* A stable subsidy regime resilient to cost escalation.

- **Re-application risk under SDE++ ranking rules**
  Loss of subsidy ranking undermines emitter commitment.
  *Stabilisation required:* Predictable ranking criteria and timely allocation outcomes.

- **Purity variation by industrial source**
  $CO_2$ source variability affects early commitment and cost structures.
  *Stabilisation required:* A workable specification envelope across emitter types.

- **Additional processing driven by specification stringency**
  High purification requirements increase emitter CAPEX and OPEX.
  *Stabilisation required:* Technically feasible and economically affordable specifications.

- **Specification-driven CAPEX/OPEX uncertainty**
  Technical uncertainty directly affects emitter FID feasibility.
  *Stabilisation required:* Transparency on cost impacts and upgrade requirements.

- **Dependency on timely publication of tariffs and specifications**
  Investment committees require clear commercial and technical parameters.
  *Stabilisation required:* Definitive and timely tariff and specification publication.

- **Technology maturity and deployment readiness**
  Low technology readiness deters capture FID decisions.
  *Stabilisation required:* Demonstrated technical reliability and operational track record.

- **Data certainty at investment committee stage**
  Uncertain cost-benefit assessments block FID approval.
  *Stabilisation required:* Higher pre-FID data certainty and validation.

- **Emitter permit and utilities complexity**
  Local permitting and infrastructure constraints delay or prevent FID.
  *Stabilisation required:* Predictable permitting and utility integration pathways.

- **Volume change triggering chain-wide restudies**
  Volume instability cascades into system redesign requirements.
  *Stabilisation required:* System design robust under volume variability.

- **Anchor-emitter dependency effects**
  Loss of a major emitter destabilises the business case.
  *Stabilisation required:* Diversified volume base and multiple anchor commitments.

- **Trunkline downtime halting the value chain**
  System reliability affects emitter willingness to commit.
  *Stabilisation required:* High trunkline reliability and fallback mechanisms.

- **Launch store 5 Mt start-up momentum**
  Storage readiness shapes emitter confidence.
  *Stabilisation required:* Definitive storage capacity and deployment schedule.

- **Store competition shaping commitment**
  Competing storage routes influence emitter choice.
  *Stabilisation required:* Consistent and competitive commercial conditions.

- **Chicken-and-egg dynamics across >20 parties**
  Coordination failure delays simultaneous commitment.
  *Stabilisation required:* Structured synchronisation of FIDs.

- **Mutual dependence of investment commitments**
  Interlinked investment decisions create strategic waiting behaviour.
  *Stabilisation required:* Sequenced commitment logic or volume guarantees.

- **Interdependency of permits across chain segments**
  Legal instability in one segment blocks chain-wide commitment.
  *Stabilisation required:* Comprehensive legal finality across segments.

- **Shipping as alternative routing option**
  Maritime transport competes directly with Aramis infrastructure.
  *Stabilisation required:* Competitive commercial terms.

- **Volume leakage through arbitrage to non-Aramis routes**
  Emitters may choose lower-cost alternatives.
  *Stabilisation required:* Economic parity or advantage.

- **BAFO/CFT timing and late tariff certainty**
  Delayed commercial clarity postpones FID decisions.
  *Stabilisation required:* Timely and transparent commercial negotiation process.

### C.3.2. All risks unfolded in sub-dependencies

In this section an overview can be found how all selected nine risks are unfolded in the 174 sub-dependencies.

| Risk-ID | Viability | Viab.Vuln | Enabl_ID | Enabling_title | Sub dependency (development indicator) | Sub (G/O/R) | Primary | Secondary | Value chain segment |
|---|---|---|---|---|---|---|---|---|---|
| **R-X:** | **R-X reframe: ....** | | | | | | | | |
| R-X | P1 | OUTER | C1.1 | Cost-effectiveness of CCS versus alternatives | Industrial relocation or cross border storage | Red | A2 | B2 | Emitters / Capture |
| R-X | P1 | OUTER | C1.1 | Cost-effectiveness of CCS versus alternatives | Tariff versus ETS/penalty comparison | Orange | A1 | A2 | Emitters / Capture |
| R-X | P1 | OUTER | C1.1 | Cost-effectiveness of CCS versus alternatives | Volume-dependent tariff escalation (€/t sensitivity) | Orange | A2 | D2 | Emitters / Capture |
| R-X | P1 | OUTER | C1.1 | Cost-effectiveness of CCS versus alternatives | Subsidy sufficiency under cost overrun scenarios | Orange | A1 | A2 | Emitters / Capture |
| R-X | P1 | OUTER | C1.1 | Cost-effectiveness of CCS versus alternatives | Re-application risk under SDE++ ranking rules | Orange | A1 | A2 | Emitters / Capture |
| R-X | P1 | OUTER | C1.2 | CO$_2$ specification as access and cost criterion | Purity variation by industrial source | Green | A2 | D1 | Emitters / Capture |
| R-X | P1 | OUTER | C1.2 | CO$_2$ specification as access and cost criterion | Additional processing requirement driven by spec stringency | Orange | D1 | A2 | Emitters / Capture |
| R-X | P1 | OUTER | C1.2 | CO$_2$ specification as access and cost criterion | Spec-driven capex/opex uncertainty at emitter level | Red | D1 | A2 | Emitters / Capture |
| R-X | P1 | MIDDLE | C1.3 | Emitter FID decision complexity | Dependency on timely publication of tariffs and specifications | Orange | A1 | A2 | Emitters / Capture |
| R-X | P1 | MIDDLE | C1.3 | Emitter FID decision complexity | Technology maturity and deployment readiness | Orange | C1 | A2 | Emitters / Capture |
| R-X | P1 | MIDDLE | C1.3 | Emitter FID decision complexity | Data certainty at investment committee stage | Orange | A2 | B1 | Emitters / Capture |
| R-X | P1 | MIDDLE | C1.3 | Emitter FID decision complexity | Emitter permit & utilities complexity | Orange | C2 | A2 | Emitters / Capture |
| R-X | P1 | OUTER | C1.4 | Total volume commitment stability | Volume change triggering chain-wide re-studies (flow assurance, hydraulics) | Green | A2 | C1 | Emitters / Capture |
| R-X | P1 | OUTER | C1.4 | Total volume commitment stability | Anchor-emitter dependency effects | Green | A2 | D2 | Emitters / Capture |
| R-X | P2 | OUTER | C2.2 | Routing flexibility and leakage exposure | Shipping as alternative routing option | Orange | A2 | B2 | Feeder Transport |
| R-X | P2 | OUTER | C2.2 | Routing flexibility and leakage exposure | Volume leakage through arbitrage to non-Aramis routes | Red | A2 | B2 | Feeder Transport |
| R-X | P4 | OUTER | C4.3 | Single-point-of-failure exposure | Trunkline downtime halting entire value chain | Green | D2 | A2 | Offshore |
| R-X | P5 | MIDDLE | C5.5 | Stores investment commitment logic | Launch stores 5 megaton start-up momentum | Orange | C1 | B1 | Stores |
| R-X | P5 | MIDDLE | C5.5 | Stores investment commitment logic | Stores competition by design to enhance the market for CCS commitment | Orange | B1 | C1 | Stores |
| R-X | P6 | OUTER | C6.1 | Tariff formation and cost discovery | BAFO / CFT timing and late tariff certainty | Green | A2 | D2 | Cross-Cutting |
| R-X | P6 | OUTER | C6.1 | Tariff formation and cost discovery | Aramis government volume guarentee (Vollooprisico) | Green | B2 | A1 | Cross-Cutting |
| R-X | P6 | OUTER | C6.3 | FID synchronisation across the value chain | Chicken-and-egg dynamics across >20 parties | Orange | C1 | B1 | Cross-Cutting |
| R-X | P6 | OUTER | C6.3 | FID synchronisation across the value chain | Mutual dependence of investment commitments | Red | C1 | A2 | Cross-Cutting |
| R-X | P6 | OUTER | C6.6 | Permitting finality and legal operability | Interdependency of permits across chain segments | Orange | C2 | B1 | Cross-Cutting |

| Risk-ID | Viability | V_VULN | Enabl_ID | Enabling_title | Sub (mechanisme) | ub_Status (G/O/ | Primary | Secondary | Value chain segment |
|---|---|---|---|---|---|---|---|---|---|
| **R-Y:** | **R-Y reframe:...** | | | | | | | | |
| R-Y: | P1 | OUTER | C1.2 | CO$_2$ specification as access and cost criterion | Purity variation by industrial source | Green | A2 | D1 | Emitter / Capture |
| R-Y: | P1 | OUTER | C1.2 | CO$_2$ specification as access and cost criterion | Additional processing requirement driven by spec stringency | Orange | D1 | A2 | Emitter / Capture |
| R-Y: | P1 | OUTER | C1.2 | CO$_2$ specification as access and cost criterion | Spec-driven capex/opex uncertainty at emitter level | Red | D1 | A2 | Emitter / Capture |
| R-Y: | P1 | OUTER | C1.4 | Total volume commitment reliability | Timing mismatch between emitter ramp-up and chain availability | Green | C1 | D2 | Emitter / Capture |
| R-Y: | P1 | OUTER | C1.4 | Total volume commitment reliability | Porthos start-up credibility as market confidence trigger | Orange | B2 | A2 | Emitter / Capture |
| R-Y: | P2 | INNER | C2.1 | Infrastructure synchronisation across feeder assets | Dependency on collection hub commissioning milestones | Orange | C1 | D2 | Feeder Transport |
| R-Y: | P2 | INNER | C2.1 | Infrastructure synchronisation across feeder assets | Sequencing uncertainty between parallel feeder developments | Red | B1 | C1 | Feeder Transport |
| R-Y: | P2 | OUTER | C2.2 | Routing flexibility and leakage exposure | Shipping as alternative routing option | Orange | A2 | B2 | Feeder Transport |
| R-Y: | P2 | OUTER | C2.2 | Routing flexibility and leakage exposure | Sequencing uncertainty between parallel feeder developments | Red | B1 | C1 | Feeder Transport |
| R-Y: | P2 | OUTER | C2.2 | Routing flexibility and leakage exposure | Impact of routing optionality on tariff stability | Orange | D2 | A2 | Feeder Transport |
| R-Y: | P2 | MIDDLE | C2.3 | Balancing and contingency arrangements | Existence of functional balancing mechanisms | Orange | B1 | D2 | Feeder Transport |
| R-Y: | P2 | MIDDLE | C2.3 | Balancing and contingency arrangements | Dependency of balancing on real spare capacity | Orange | D2 | C1 | Feeder Transport |
| R-Y: | P3 | OUTER | C3.1 | CO$_2$Next system readiness & commitment | CO$_2$Next commissioning alignment with Aramis start-up | Red | C1 | D2 | Onshore Hub & Compression |
| R-Y: | P3 | OUTER | C3.1 | CO$_2$Next system readiness & commitment | Availability attribution between CO$_2$Next, Porthos and Aramis | Green | B1 | D2 | Onshore Hub & Compression |
| R-Y: | P3 | OUTER | C3.1 | CO$_2$Next system readiness & commitment | Back-to-back availability and payment logic | Green | B1 | D2 | Onshore Hub & Compression |
| R-Y: | P3 | OUTER | C3.1 | CO$_2$Next system readiness & commitment | Cost allocation under interface design changes | Green | B1 | D1 | Onshore Hub & Compression |
| R-Y: | P3 | OUTER | C3.1 | CO$_2$Next system readiness & commitment | CO$_2$Next goverment volume guarentee (Vollooprisico) | Green | B2 | B1 | Onshore Hub & Compression |
| R-Y: | P3 | OUTER | C3.3 | Porthos system integration & readiness | CO$_2$ specification alignment and change governance | Orange | C1 | B1 | Onshore Hub & Compression |
| R-Y: | P3 | OUTER | C3.3 | Porthos system integration & readiness | Availability definition mirroring across systems | Orange | B1 | D2 | Onshore Hub & Compression |
| R-Y: | P3 | OUTER | C3.4 | Design constraints from external system drivers | Nitrogen constraints as design and execution driver | Orange | A1 | C2 | Onshore Hub & Compression |
| R-Y: | P4 | OUTER | C4.1 | Offshore transport integrity and operability | Corrosion and impurity sensitivity | Orange | D1 | C1 | Onshore Hub & Compression |
| R-Y: | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Reliance on upstream/downstream protection systems | Green | C1 | D2 | Offshore |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| R-Y: | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Interface-defined safeguards and trip logic | Green | C1 | B1 | Offshore |
| R-Y: | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Loss of autonomy under partner system failures | Orange | D2 | C1 | Offshore |
| R-Y: | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Failure-mode discovery & safety case evidence (no rulebook) | Orange | B2 | C2 | Offshore |
| R-Y: | P5 | MIDDLE | C5.5 | Stores investment commitment logic | Launch stores 5 megaton start-up momentum | Orange | C1 | B1 | Storage |
| R-Y: | P5 | MIDDLE | C5.5 | Stores investment commitment logic | Risk reward balance thresholds for private storage parties | Orange | A2 | D2 | Storage |
| R-Y: | P5 | MIDDLE | C5.5 | Stores investment commitment logic | Implementation requirements Net Zero Industry Act | Orange | C2 | B2 | Storage |
| R-Y: | P5 | MIDDLE | C5.5 | Stores investment commitment logic | Stores competition by design to enhance the market for CCS commitment | Orange | B1 | B2 | Storage |
| R-Y: | P6 | OUTER | C6.1 | Tariff formation and cost discovery | BAFO / CFT timing and late tariff certainty | Green | A2 | D2 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.1 | Tariff formation and cost discovery | Emergent tariff build-up from chain-wide capex/opex | Orange | D1 | A2 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.1 | Tariff formation and cost discovery | Sensitivity of tariff to downstream and upstream costs | Orange | D1 | A2 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.1 | Tariff formation and cost discovery | Aramis government volume guarentee (Vollooprisico) | Green | B2 | A1 | Cross-Cutting |
| R-Y: | P6 | MIDDLE | C6.2 | System integration governance | Back-to-back mirroring of operational and financial logic | Green | B1 | D2 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.3 | FID synchronisation across the value chain | Chicken-and-egg dynamics across >20 parties | Orange | C1 | B1 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.3 | FID synchronisation across the value chain | Mutual dependence of investment commitments | Red | C1 | A2 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.4 | Chain-wide contractual coherence | Back-to-back payment logic | Orange | B1 | D2 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.4 | Chain-wide contractual coherence | Availability mirroring across contracts | Orange | B1 | D2 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.4 | Chain-wide contractual coherence | Residual exposure prevention | Green | B1 | D2 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.4 | Chain-wide contractual coherence | IOA/IOE/TOI/EMOC | Orange | B1 | C1 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.5 | Strategic optionality and system rigidity | Policy-defined investment horizon | Orange | A1 | D1 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.6 | Permitting finality and legal operability | Legal finality versus reversibility | Red | C2 | B2 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.6 | Permitting finality and legal operability | Interdependency of permits across chain segments | Orange | C2 | B1 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.6 | Permitting finality and legal operability | Temporal alignment of permit validity with FID and execution | Orange | B2 | C1 | Cross-Cutting |
| R-Y: | P6 | OUTER | C6.6 | Permitting finality and legal operability | Authority acceptance of integrated system safety cases | Orange | B2 | D1 | Cross-Cutting |

| Risk-ID | Viability | V_VULN | Enabl_ID | Enabling_title | Sub (mechanisme) | ub_Status (G/O/F | Primary | Secondary | Value chain segment |
|---|---|---|---|---|---|---|---|---|---|
| **R-Z:** | | | | | | | | | |
| R-Z | P3 | OUTER | C3.1 | $CO_2$Next system readiness & commitment | $CO_2$Next commissioning alignment with Aramis start-up | Red | C1 | D2 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.1 | $CO_2$Next system readiness & commitment | $CO_2$Next interface compatibility (pressure, quality, flow envelope) | Red | C1 | D1 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.1 | $CO_2$Next system readiness & commitment | Availability attribution between $CO_2$Next, Porthos and Aramis | Green | B1 | D2 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.1 | $CO_2$Next system readiness & commitment | Back-to-back availability and payment logic | Green | B1 | D2 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.1 | $CO_2$Next system readiness & commitment | Cost allocation under interface design changes | Green | B1 | D1 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.1 | $CO_2$Next system readiness & commitment | $CO_2$Next goverment volume guarentee (Vollooprisico) | Green | B2 | B1 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.2 | Compression capacity, availability & redundancy | Availability targets as contractual payment condition | Green | B1 | B2 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.2 | Compression capacity, availability & redundancy | Redundancy evaluation decision rules (design freeze gate) | Orange | D1 | D2 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.2 | Compression capacity, availability & redundancy | Availability shortfall remedy mechanisms | Orange | B1 | D2 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.2 | Compression capacity, availability & redundancy | Attribution rules for downtime across chain parties | Orange | B1 | D2 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.3 | Porthos system integration & readiness | Porthos commissioning milestones versus Aramis ramp-up | Orange | C1 | D2 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.3 | Porthos system integration & readiness | Control room integration and decision authority | Green | B1 | D1 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.3 | Porthos system integration & readiness | $CO_2$ specification alignment and change governance | Orange | C1 | B1 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.3 | Porthos system integration & readiness | Availability definition mirroring across systems | Orange | B1 | D2 | Onshore Hub & Compression |
| R-Z | P3 | OUTER | C3.3 | Porthos system integration & readiness | Porthos interface compatibility (pressure, quality, flow envelope) | Orange | C1 | D1 | Onshore Hub & Compression |
| R-Z | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Loss of autonomy under partner system failures | Orange | D2 | C1 | Offshore |
| R-Z | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Failure-mode discovery & safety case evidence (no rulebook) | Orange | B2 | C2 | Offshore |
| R-Z | P4 | OUTER | C4.3 | Single-point-of-failure exposure | Recovery and restart complexity after offshore outage | Green | C1 | D2 | Offshore |
| R-Z | P4 | OUTER | C4.4 | Off-spec and upset handling procedures | Integrated control philosofy | Orange | D1 | B1 | Offshore |
| R-Z | P4 | OUTER | C4.4 | Off-spec and upset handling procedures | Stop / vent / recovery decision logic | Orange | C1 | D2 | Offshore |
| R-Z | P4 | OUTER | C4.4 | Off-spec and upset handling procedures | Impact of upset handling on chain availability | Orange | D2 | A2 | Offshore |
| R-Z | P5 | INNER | C5.3 | Store availability and revenue exposure | Store availability dependency on subsidy and revenue streams | Orange | D2 | A1 | Stores |
| R-Z | P5 | INNER | C5.3 | Store availability and revenue exposure | Financial exposure under injection downtime | Orange | B1 | B2 | Stores |
| R-Z | P5 | MIDDLE | C5.5 | Stores investment commitment logic | Launch stores 5 megaton start-up momentum | Orange | C1 | B1 | Stores |
| R-Z | P5 | MIDDLE | C5.5 | Stores investment commitment logic | Stores competition by design to enhance the market for CCS commitment | Orange | B1 | C1 | Stores |

| Risk-ID | Viability | V_VULN | Enabl_ID | Enabling_title | Sub (mechanisme) | ub_Status (G/O/ | Primary | Secondary | Value chain segment |
|---|---|---|---|---|---|---|---|---|---|
| R-Z | P6 | OUTER | C6.1 | Tariff formation and cost discovery | BAFO / CFT timing and late tariff certainty | Green | A2 | D2 | Cross-Cutting |
| R-Z | P6 | OUTER | C6.1 | Tariff formation and cost discovery | Emergent tariff build-up from chain-wide capex/opex | Orange | D1 | A2 | Cross-Cutting |
| R-Z | P6 | OUTER | C6.1 | Tariff formation and cost discovery | Sensitivity of tariff to downstream and upstream costs | Orange | D1 | A2 | Cross-Cutting |
| R-Z | P6 | OUTER | C6.1 | Tariff formation and cost discovery | Aramis government volume guarentee (Vollooprisico) | Green | B2 | A1 | Cross-Cutting |
| R-Z | P6 | OUTER | C6.4 | Chain-wide contractual coherence | Back-to-back payment logic | Orange | B1 | D2 | Cross-Cutting |
| R-Z | P6 | OUTER | C6.4 | Chain-wide contractual coherence | Availability mirroring across contracts | Orange | B1 | D2 | Cross-Cutting |
| R-Z | P6 | OUTER | C6.4 | Chain-wide contractual coherence | Residual exposure prevention | Green | B1 | D2 | Cross-Cutting |
| R-Z | P6 | OUTER | C6.4 | Chain-wide contractual coherence | IOA/IOE/TOI/EMOC | Orange | B1 | C1 | Cross-Cutting |
| R-Z | P6 | OUTER | C6.5 | Strategic optionality and system rigidity | Lock-in effects once trunk infrastructure is committed | Green | D1 | A2 | Cross-Cutting |
| **Risk-ID** | **Viability** | **V_VULN** | **Enabl_ID** | **Enabling_title** | **Sub (mechanisme)** | **ub_Status (G/O/** | **Primary** | **Secondary** | **Value chain segment** |
| **R-A** | | | | | | | | | |
| R-A | P1 | OUTER | C1.2 | $CO_2$ specification as access and cost criterion | Purity variation by industrial source | Green | A2 | D1 | Emitters / Capture |
| R-A | P1 | OUTER | C1.2 | $CO_2$ specification as access and cost criterion | Location of quality measurement and enforcement (upstream vs post-mixing) | Green | D1 | C1 | Emitters / Capture |
| R-A | P1 | OUTER | C1.2 | $CO_2$ specification as access and cost criterion | Spec-driven capex/opex uncertainty at emitter level | Red | D1 | A2 | Emitters / Capture |
| R-A | P3 | OUTER | C3.4 | Design constraints from external system drivers | Code/regulation gap for dense-phase $CO_2$ (compliant ≠ safe) | Green | A1 | D1 | Onshore Hub & Compression |
| R-A | P4 | OUTER | C4.1 | Offshore transport integrity and operability | Pressure and temperature envelope stability | Orange | D2 | C1 | Offshore |
| R-A | P4 | OUTER | C4.1 | Offshore transport integrity and operability | Dense-phase $CO_2$ transport behaviour under upset conditions | Green | D2 | D1 | Offshore |
| R-A | P4 | OUTER | C4.1 | Offshore transport integrity and operability | Corrosion and impurity sensitivity | Orange | D1 | C1 | Offshore |
| R-A | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Failure-mode discovery & safety case evidence (no rulebook) | Orange | B2 | C2 | Offshore |
| R-A | P5 | OUTER | C5.4 | $CO_2$ specification bounded by storage constraints | License-driven composition limits | Orange | C2 | B2 | Stores |
| **Risk-ID** | **Viability** | **V_VULN** | **Enabl_ID** | **Enabling_title** | **Sub (mechanisme)** | **ub_Status (G/O/** | **Primary** | **Secondary** | **Value chain segment** |
| **R-B** | | | | | | | | | |
| R-B | P1 | OUTER | C1.2 | $CO_2$ specification as access and cost criterion | Additional processing requirement driven by spec stringency | Orange | D1 | A2 | Emitter / Capture |
| R-B | P1 | MIDDLE | C1.3 | Emitter FID decision complexity | Emitter permit & utilities complexity | Orange | C2 | A2 | Emitter / Capture |
| R-B | P1 | OUTER | C1.4 | Total volume commitment reliability | Volume change triggering chain-wide re-studies (flow assurance, hydraulics) | Green | A2 | C1 | Emitter / Capture |
| R-B | P3 | OUTER | C3.4 | Design constraints from external system drivers | Nitrogen constraints as design and execution driver | Orange | A1 | C2 | Onshore Hub & Compression |
| R-B | P3 | OUTER | C3.4 | Design constraints from external system drivers | Electrification requirements affecting layout and phasing | Orange | A1 | C1 | Cross-Cutting |
| R-B | P6 | OUTER | C6.6 | Permitting finality and legal operability | Legal finality versus reversibility | Red | C2 | B2 | Cross-Cutting |
| R-B | P6 | OUTER | C6.6 | Permitting finality and legal operability | Interdependency of permits across chain segments | Orange | C2 | B1 | Cross-Cutting |
| R-B | P6 | OUTER | C6.6 | Permitting finality and legal operability | Temporal alignment of permit validity with FID and execution | Orange | B2 | C1 | Cross-Cutting |
| **Risk-ID** | **Viability** | **V_VULN** | **Enabl_ID** | **Enabling_title** | **Sub (mechanisme)** | **ub_Status (G/O/** | **Primary** | **Secondary** | **Value chain segment** |
| **R-C** | | | | | | | | | |
| R-C | P1 | OUTER | C1.4 | Total volume commitment reliability | Porthos start-up credibility as market confidence trigger | Orange | B2 | A2 | Emitter / Capture |
| R-C | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Interface-defined safeguards and trip logic | Green | C1 | B1 | Offshore |
| R-C | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Failure-mode discovery & safety case evidence (no rulebook) | Orange | B2 | C2 | Offshore |
| R-C | P4 | OUTER | C4.4 | Off-spec and upset handling procedures | Integrated control philosofy | Orange | D1 | B1 | Offshore |
| R-C | P6 | OUTER | C6.6 | Permitting finality and legal operability | Legal finality versus reversibility | Red | C2 | B2 | Cross-Cutting |
| R-C | P6 | OUTER | C6.6 | Permitting finality and legal operability | Authority acceptance of integrated system safety cases | Orange | B2 | D1 | Cross-Cutting |
| **Risk-ID** | **Viability** | **V_VULN** | **Enabl_ID** | **Enabling_title** | **Sub (mechanisme)** | **ub_Status (G/O/** | **Primary** | **Secondary** | **Value chain segment** |
| **R-D** | | | | | | | | | |
| R-D | P1 | OUTER | C1.4 | Total volume commitment reliability | Porthos start-up credibility as market confidence trigger | Orange | B2 | A2 | Emitter / Capture |
| R-D | P2 | INNER | C2.1 | Infrastructure synchronisation across feeder assets | Sequencing uncertainty between parallel feeder developments | Red | B1 | C1 | Feeder Transport |
| R-D | P3 | OUTER | C3.1 | $CO_2$Next system readiness & commitment | $CO_2$Next interface compatibility (pressure, quality, flow envelope) | Red | C1 | D1 | Onshore Hub & Compression |
| R-D | P3 | OUTER | C3.1 | $CO_2$Next system readiness & commitment | Cost allocation under interface design changes | Green | B1 | D1 | Onshore Hub & Compression |
| R-D | P3 | OUTER | C3.2 | Compression capacity, availability & redundancy | Availability targets as contractual payment condition | Green | B1 | B2 | Onshore Hub & Compression |
| R-D | P3 | OUTER | C3.2 | Compression capacity, availability & redundancy | Attribution rules for downtime across chain parties | Orange | B1 | D2 | Onshore Hub & Compression |
| R-D | P3 | OUTER | C3.3 | Porthos system integration & readiness | Control room integration and decision authority | Green | B1 | D1 | Onshore Hub & Compression |
| R-D | P3 | OUTER | C3.3 | Porthos system integration & readiness | Porthos interface compatibility (pressure, quality, flow envelope) | Orange | C1 | D1 | Onshore Hub & Compression |
| R-D | P3 | OUTER | C3.4 | Design constraints from external system drivers | Code/regulation gap for dense-phase $CO_2$ (compliant ≠ safe) | Green | A1 | D1 | Onshore Hub & Compression |
| R-D | P4 | OUTER | C4.1 | Offshore transport integrity and operability | Pressure and temperature envelope stability | Orange | D2 | C1 | Offshore |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| R-D | P4 | OUTER | C4.1 | Offshore transport integrity and operability | Dense-phase CO$_2$ transport behaviour under upset conditions | Green | D2 | D1 | Offshore |
| R-D | P4 | OUTER | C4.1 | Offshore transport integrity and operability | Corrosion and impurity sensitivity | Orange | D1 | C1 | Offshore |
| R-D | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Reliance on upstream/downstream protection systems | Green | C1 | D2 | Offshore |
| R-D | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Interface-defined safeguards and trip logic | Green | C1 | B1 | Offshore |
| R-D | P4 | OUTER | C4.4 | Off-spec and upset handling procedures | Integrated control philosofy | Orange | D1 | B1 | Offshore |
| R-D | P6 | MIDDLE | C6.2 | System integration governance | System integrator role definition and mandate | Orange | B1 | B2 | Cross-Cutting |
| R-D | P6 | MIDDLE | C6.2 | System integration governance | Change-control authority across interfaces | Green | B1 | C1 | Cross-Cutting |
| R-D | P6 | MIDDLE | C6.2 | System integration governance | Back-to-back mirroring of operational and financial logic | Green | B1 | D2 | Cross-Cutting |
| R-D | P6 | OUTER | C6.5 | Strategic optionality and system rigidity | Limited ability to phase or decouple investments | Orange | D1 | C1 | Cross-Cutting |

| Risk-ID | Viability | V_VULN | Enabl_ID | Enabling_title | Sub (mechanisme) | ub_Status (G/O/F | Primary | Secondary | Value chain segment |
|---|---|---|---|---|---|---|---|---|---|
| **R-E** | | | | | | | | | |
| R-E | P1 | OUTER | C1.2 | CO$_2$ specification as access and cost criterion | Purity variation by industrial source | Green | A2 | D1 | Emitter / Capture |
| R-E | P1 | OUTER | C1.2 | CO$_2$ specification as access and cost criterion | Location of quality measurement and enforcement (upstream vs post-mixing) | Green | D1 | C1 | Emitter / Capture |
| R-E | P1 | OUTER | C1.4 | Total volume commitment reliability | Volume change triggering chain-wide re-studies (flow assurance, hydraulics) | Green | A2 | C1 | Emitter / Capture |
| R-E | P2 | MIDDLE | C2.3 | Balancing and contingency arrangements | Temporary rerouting/storage feasibility under outage scenarios | Orange | C1 | D2 | Transport Feeder |
| R-E | P3 | OUTER | C3.1 | CO$_2$Next system readiness & commitment | CO$_2$Next interface compatibility (pressure, quality, flow envelope) | Red | C1 | D1 | Onshore Hub & Compression |
| R-E | P3 | OUTER | C3.3 | Porthos system integration & readiness | CO$_2$ specification alignment and change governance | Orange | C1 | B1 | Onshore Hub & Compression |
| R-E | P3 | OUTER | C3.3 | Porthos system integration & readiness | Porthos interface compatibility (pressure, quality, flow envelope) | Orange | C1 | D1 | Onshore Hub & Compression |
| R-E | P3 | OUTER | C3.4 | Design constraints from external system drivers | Nitrogen constraints as design and execution driver | Orange | A1 | C2 | Onshore Hub & Compression |
| R-E | P3 | OUTER | C3.4 | Design constraints from external system drivers | Electrification requirements affecting layout and phasing | Orange | A1 | C1 | Onshore Hub & Compression |
| R-E | P3 | OUTER | C3.4 | Design constraints from external system drivers | Micro-tunnel removal obligations | Red | A1 | D1 | Onshore Hub & Compression |
| R-E | P3 | OUTER | C3.4 | Design constraints from external system drivers | Code/regulation gap for dense-phase CO$_2$ (compliant ≠ safe) | Green | A1 | D1 | Onshore Hub & Compression |
| R-E | P4 | OUTER | C4.1 | Offshore transport integrity and operability | Pressure and temperature envelope stability | Orange | D2 | C1 | Offshore |
| R-E | P4 | OUTER | C4.1 | Offshore transport integrity and operability | Corrosion and impurity sensitivity | Orange | D1 | C1 | Offshore |
| R-E | P4 | MIDDLE | C4.2 | Safety-by-interface dependency | Failure-mode discovery & safety case evidence (no rulebook) | Orange | B2 | C2 | Offshore |
| R-E | P4 | OUTER | C4.5 | Execution capacity & schedule realism | EU tender/procurement rules fit with execution | Orange | A1 | C1 | Offshore |
| R-E | P4 | OUTER | C4.5 | Execution capacity & schedule realism | Contractor scarcity & schedule gate | Red | C1 | A2 | Offshore |
| R-E | P5 | OUTER | C5.4 | CO$_2$ specification bounded by storage constraints | License-driven composition limits | Orange | C2 | B2 | Stores |
| R-E | P6 | OUTER | C6.1 | Tariff formation and cost discovery | BAFO / CFT timing and late tariff certainty | Green | A2 | D2 | Cross-Cutting |
| R-E | P6 | MIDDLE | C6.2 | System integration governance | System integrator role definition and mandate | Orange | B1 | B2 | Cross-Cutting |
| R-E | P6 | OUTER | C6.6 | Permitting finality and legal operability | Legal finality versus reversibility | Red | C2 | B2 | Cross-Cutting |
| R-E | P6 | OUTER | C6.6 | Permitting finality and legal operability | Authority acceptance of integrated system safety cases | Orange | B2 | D1 | Cross-Cutting |

| Risk-ID | Viability | V_VULN | Enabl_ID | Enabling_title | Sub (mechanisme) | ub_Status (G/O/F | Primary | Secondary | Value chain segment |
|---|---|---|---|---|---|---|---|---|---|
| **R-F** | | | | | | | | | |
| R-F | P1 | MIDDLE | C1.1 | Cost-effectiveness of CCS versus alternatives | Tariff versus ETS/penalty comparison | Orange | A1 | A2 | Emitter / Capture |
| R-F | P1 | MIDDLE | C1.1 | Cost-effectiveness of CCS versus alternatives | Emit-and-pay versus CCS decision threshold | Orange | A2 | A1 | Emitter / Capture |
| R-F | P1 | MIDDLE | C1.1 | Cost-effectiveness of CCS versus alternatives | Volume-dependent tariff escalation (€/t sensitivity) | Orange | A2 | D2 | Emitter / Capture |
| R-F | P1 | MIDDLE | C1.1 | Cost-effectiveness of CCS versus alternatives | Subsidy sufficiency under cost overrun scenarios | Orange | A1 | A2 | Emitter / Capture |
| R-F | P1 | MIDDLE | C1.1 | Cost-effectiveness of CCS versus alternatives | Re-application risk under SDE++ ranking rules | Orange | A1 | A2 | Emitter / Capture |
| R-F | P3 | OUTER | C3.1 | CO$_2$Next system readiness & commitment | CO$_2$Next goverment volume guarentee (Vollooprisico) | Green | B2 | B1 | Onshore Hub & Compression |
| R-F | P6 | OUTER | C6.1 | Tariff formation and cost discovery | Aramis government volume guarentee (Vollooprisico) | Green | B2 | A1 | Cross-Cutting |
| R-F | P6 | OUTER | C6.5 | Strategic optionality and system rigidity | Policy-defined investment horizon | Orange | A1 | D1 | Cross-Cutting |

## C.3.3. Sub-dependencies count underlying the nine risks

This section shows the frequency with which all sub-dependencies appeared across the nine risks. The second table on the page shows only the critical sub-dependencies: those red sub-dependencies associated with an enabling dependency that has an outer vulnerability. It also shows the specific risks in which they appear.

**Sub patterns (N_rows=174, N_risks=9, N_children=25) in Risks Sorted on amount in Risks (Red Status if Red in combination with Outer)**

| Enabling_ID | Enabling_SCORE | Enabling_label | Sub (mechanism) | Appears in #/9 risks | Status |
|---|---|---|---|---|---|
| C4.2 | MIDDLE | Safety-by-interface dependency | Failure-mode discovery & safety case evidence (no rulebook) | 5 | Orange |
| C1.2 | OUTER | $CO_2$ specification as access and cost criterion | Purity variation by industrial source | 4 | Green |
| C4.1 | OUTER | Offshore transport integrity and operability | Corrosion and impurity sensitivity | 4 | Orange |
| C6.1 | OUTER | Tariff formation and cost discovery | BAFO / CFT timing and late tariff certainty | 4 | Green |
| C6.1 | OUTER | Tariff formation and cost discovery | Aramis government volume guarentee (Vollooprisico) | 4 | Green |
| C6.6 | OUTER | Permitting finality and legal operability | Legal finality versus reversibility | 4 | Red |
| C1.2 | OUTER | $CO_2$ specification as access and cost criterion | Additional processing requirement driven by spec stringency | 3 | Orange |
| C1.2 | OUTER | $CO_2$ specification as access and cost criterion | Spec-driven capex/opex uncertainty at emitter level | 3 | Red |
| C1.4 | OUTER | Total volume commitment reliability | Volume change triggering chain-wide re-studies (flow assurance, hydraulics) | 3 | Green |
| C1.4 | OUTER | Total volume commitment reliability | Porthos start-up credibility as market confidence trigger | 3 | Orange |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | $CO_2$Next goverment volume guarentee (Vollooprisico) | 3 | Green |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | Cost allocation under interface design changes | 3 | Green |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | $CO_2$Next interface compatibility (pressure, quality, flow envelope) | 3 | Red |
| C3.3 | OUTER | Porthos system integration & readiness | $CO_2$ specification alignment and change governance | 3 | Orange |
| C3.3 | OUTER | Porthos system integration & readiness | Porthos interface compatibility (pressure, quality, flow envelope) | 3 | Orange |
| C3.4 | OUTER | Design constraints from external system drivers | Nitrogen constraints as design and execution driver | 3 | Orange |
| C3.4 | OUTER | Design constraints from external system drivers | Code/regulation gap for dense-phase $CO_2$ (compliant ≠ safe) | 3 | Green |
| C4.1 | OUTER | Offshore transport integrity and operability | Pressure and temperature envelope stability | 3 | Orange |
| C4.2 | MIDDLE | Safety-by-interface dependency | Interface-defined safeguards and trip logic | 3 | Green |
| C4.4 | OUTER | Off-spec and upset handling procedures | Integrated control philosofy | 3 | Orange |
| C5.5 | MIDDLE | Stores investment commitment logic | Launch stores 5 megaton start-up momentum | 3 | Orange |
| C5.5 | MIDDLE | Stores investment commitment logic | Stores competition by design to enhance the market for CCS commitment | 3 | Orange |
| C6.6 | OUTER | Permitting finality and legal operability | Interdependency of permits across chain segments | 3 | Orange |
| C6.6 | OUTER | Permitting finality and legal operability | Authority acceptance of integrated system safety cases | 3 | Orange |
| C1.1 | MIDDLE | Cost-effectiveness of CCS versus alternatives | Tariff versus ETS/penalty comparison | 2 | Orange |
| C1.1 | MIDDLE | Cost-effectiveness of CCS versus alternatives | Volume-dependent tariff escalation (€/t sensitivity) | 2 | Orange |
| C1.1 | MIDDLE | Cost-effectiveness of CCS versus alternatives | Re-application risk under SDE++ ranking rules | 2 | Orange |
| C1.1 | MIDDLE | Cost-effectiveness of CCS versus alternatives | Subsidy sufficiency under cost overrun scenarios | 2 | Orange |
| C1.2 | OUTER | $CO_2$ specification as access and cost criterion | Location of quality measurement and enforcement (upstream vs post-mixing) | 2 | Green |
| C1.3 | MIDDLE | Emitter FID decision complexity | Emitter permit & utilities complexity | 2 | Orange |
| C2.1 | INNER | Infrastructure synchronisation across feeder assets | Sequencing uncertainty between parallel feeder developments | 2 | Red |
| C2.2 | OUTER | Routing flexibility and leakage exposure | Shipping as alternative routing option | 2 | Orange |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | $CO_2$Next commissioning alignment with Aramis start-up | 2 | Red |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | Availability attribution between $CO_2$Next, Porthos and Aramis | 2 | Green |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | Back-to-back availability and payment logic | 2 | Green |
| C3.2 | OUTER | Compression capacity, availability & redundancy | Availability targets as contractual payment condition | 2 | Green |
| C3.2 | OUTER | Compression capacity, availability & redundancy | Attribution rules for downtime across chain parties | 2 | Orange |
| C3.3 | OUTER | Porthos system integration & readiness | Availability definition mirroring across systems | 2 | Orange |
| C3.3 | OUTER | Porthos system integration & readiness | Control room integration and decision authority | 2 | Green |
| C3.4 | OUTER | Design constraints from external system drivers | Electrification requirements affecting layout and phasing | 2 | Orange |
| C4.1 | OUTER | Offshore transport integrity and operability | Dense-phase $CO_2$ transport behaviour under upset conditions | 2 | Green |
| C4.2 | MIDDLE | Safety-by-interface dependency | Reliance on upstream/downstream protection systems | 2 | Green |
| C4.2 | MIDDLE | Safety-by-interface dependency | Loss of autonomy under partner system failures | 2 | Orange |
| C5.4 | OUTER | $CO_2$ specification bounded by storage constraints | License-driven composition limits | 2 | Orange |
| C6.1 | OUTER | Tariff formation and cost discovery | Emergent tariff build-up from chain-wide capex/opex | 2 | Orange |
| C6.1 | OUTER | Tariff formation and cost discovery | Sensitivity of tariff to downstream and upstream costs | 2 | Orange |

| | | | | | |
|---|---|---|---|---|---|
| C6.2 | MIDDLE | System integration governance | Back-to-back mirroring of operational and financial logic | 2 | Green |
| C6.2 | MIDDLE | System integration governance | System integrator role definition and mandate | 2 | Orange |
| C6.3 | OUTER | FID synchronisation across the value chain | Chicken-and-egg dynamics across >20 parties | 2 | Orange |
| C6.3 | OUTER | FID synchronisation across the value chain | Mutual dependence of investment commitments | 2 | Red |
| C6.4 | OUTER | Chain-wide contractual coherence | Back-to-back payment logic | 2 | Orange |
| C6.4 | OUTER | Chain-wide contractual coherence | Availability mirroring across contracts | 2 | Orange |
| C6.4 | OUTER | Chain-wide contractual coherence | Residual exposure prevention | 2 | Green |
| C6.4 | OUTER | Chain-wide contractual coherence | IOA/IOE/TOI/EMOC | 2 | Orange |
| C6.5 | OUTER | Strategic optionality and system rigidity | Policy-defined investment horizon | 2 | Orange |
| C6.6 | OUTER | Permitting finality and legal operability | Temporal alignment of permit validity with FID and execution | 2 | Orange |
| C1.3 | MIDDLE | Emitter FID decision complexity | Dependency on timely publication of tariffs and specifications | 1 | Orange |
| C2.1 | INNER | Infrastructure synchronisation across feeder assets | Dependency on collection hub commissioning milestones | 1 | Orange |
| C2.2 | OUTER | Routing flexibility and leakage exposure | Volume leakage through arbitrage to non-Aramis routes | 1 | Red |
| C2.3 | MIDDLE | Balancing and contingency arrangements | Existence of functional balancing mechanisms | 1 | Orange |
| C2.3 | MIDDLE | Balancing and contingency arrangements | Dependency of balancing on real spare capacity | 1 | Orange |
| C4.3 | OUTER | Single-point-of-failure exposure | Trunkline downtime halting entire value chain | 1 | Green |
| C4.3 | OUTER | Single-point-of-failure exposure | Recovery and restart complexity after offshore outage | 1 | Green |
| C4.4 | OUTER | Off-spec and upset handling procedures | Stop / vent / recovery decision logic | 1 | Orange |
| C4.5 | OUTER | Execution capacity & schedule realism | EU tender/procurement rules fit with execution | 1 | Orange |
| C4.5 | OUTER | Execution capacity & schedule realism | Contractor scarcity & schedule gate | 1 | Red |
| C5.3 | INNER | Store availability and revenue exposure | Store availability dependency on subsidy and revenue streams | 1 | Orange |
| C5.3 | INNER | Store availability and revenue exposure | Financial exposure under injection downtime | 1 | Orange |
| C6.5 | OUTER | Strategic optionality and system rigidity | Lock-in effects once trunk infrastructure is committed | 1 | Green |

**Critical SUB View Analysis - OUTER vulnerability & Red Development - Sub View**

| Enabling ID | Enabling Vulnerability | Enabling label | Sub (mechanisme) | Sub Appearance in #/9 risks | Status | Containing Risks # |
|---|---|---|---|---|---|---|
| C6.6 | OUTER | Permitting finality and legal operability | Legal finality versus reversibility | 4 | Red | 4 |
| C1.2 | OUTER | $CO_2$ specification as access and cost criterion | Spec-driven capex/opex uncertainty at emitter level | 3 | Red | 3 |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | $CO_2$Next interface compatibility (pressure, quality, flow envelope) | 3 | Red | 3 |
| C3.1 | OUTER | $CO_2$Next system readiness & commitment | $CO_2$Next commissioning alignment with Aramis start-up | 2 | Red | 2 |
| C6.3 | OUTER | FID synchronisation across the value chain | Mutual dependence of investment commitments | 2 | Red | 2 |
| C2.2 | OUTER | Routing flexibility and leakage exposure | Volume leakage through arbitrage to non-Aramis routes | 1 | Red | 1 |
| C4.5 | OUTER | Execution capacity & schedule realism | Contractor scarcity & schedule gate | 1 | Red | 1 |

### C.3.4. Aggregated sub-dependencies to enabling dependencies count underlying the nine risks

This section shows all enabling dependencies with their associated sub-dependencies aggregated. The second table on the page only shows the critical enabling dependencies: outer vulnerability + red subs.

**Aggregated Subs to their Associated Enabling Dependency - Count in Risks**

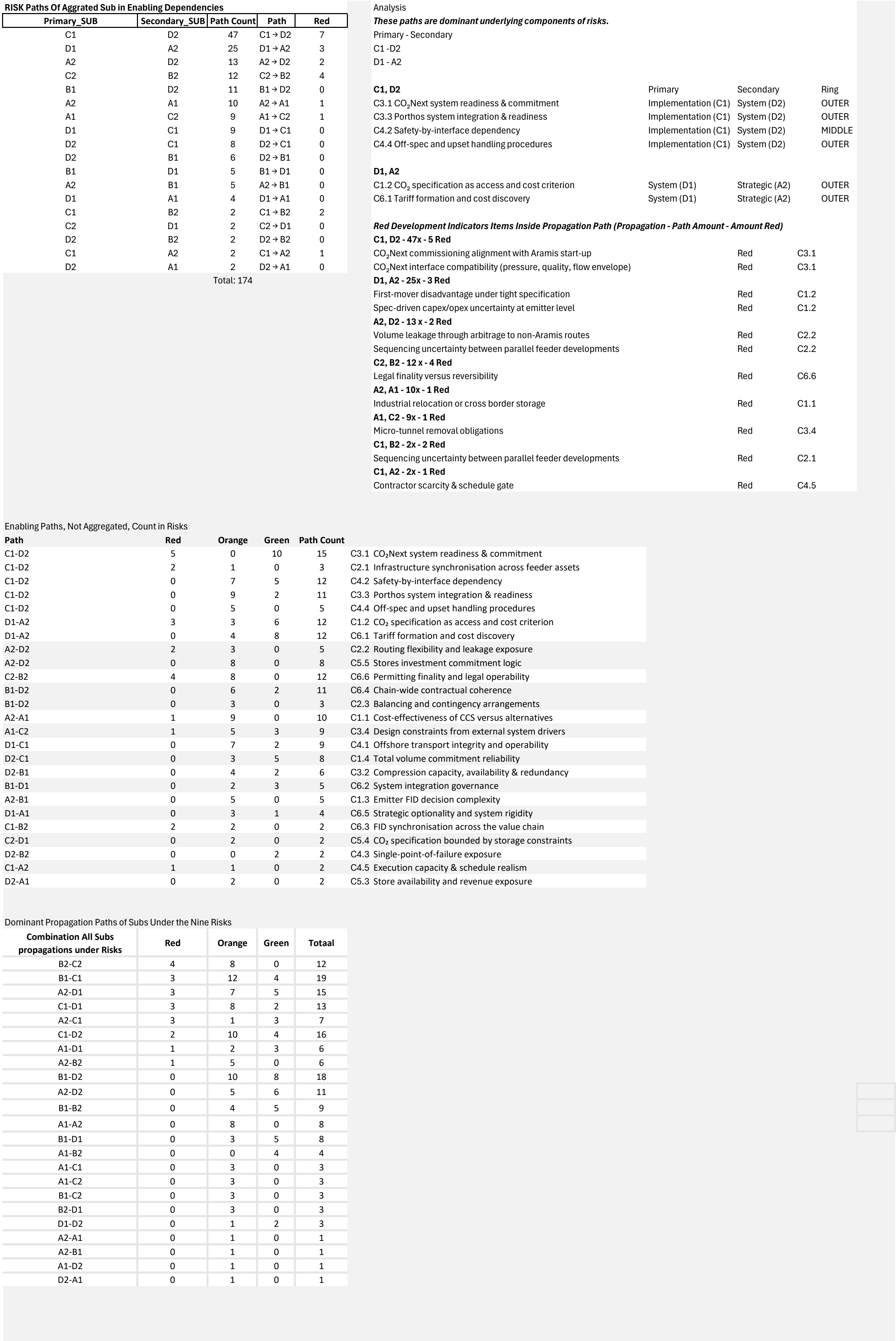
| Enabling ID | Enabling Label | Appears in #/9 risks | Score_vulnerability | Enabling count of used subs in total #/174 (rows) | Red | Orange | Green |
|---|---|---|---|---|---|---|---|
| C4.2 | Safety-by-interface dependency | 6 | MIDDLE | 12 | 0 | 7 | 5 |
| C1.4 | Total volume commitment reliability | 6 | OUTER | 8 | 0 | 3 | 5 |
| C3.1 | $CO_2$Next system readiness & commitment | 5 | OUTER | 15 | 5 | 0 | 10 |
| C1.2 | $CO_2$ specification as access and cost criterion | 5 | OUTER | 12 | 3 | 3 | 6 |
| C6.1 | Tariff formation and cost discovery | 5 | OUTER | 12 | 0 | 4 | 8 |
| C6.6 | Permitting finality and legal operability | 5 | OUTER | 12 | 4 | 8 | 0 |
| C3.4 | Design constraints from external system drivers | 5 | OUTER | 9 | 1 | 5 | 3 |
| C3.3 | Porthos system integration & readiness | 4 | OUTER | 11 | 0 | 9 | 2 |
| C4.1 | Offshore transport integrity and operability | 4 | OUTER | 9 | 0 | 7 | 2 |
| C6.5 | Strategic optionality and system rigidity | 4 | OUTER | 4 | 0 | 3 | 1 |
| C5.5 | Stores investment commitment logic | 3 | MIDDLE | 8 | 0 | 8 | 0 |
| C4.4 | Off-spec and upset handling procedures | 3 | OUTER | 5 | 0 | 5 | 0 |
| C6.2 | System integration governance | 3 | MIDDLE | 5 | 0 | 2 | 3 |
| C1.1 | Cost-effectiveness of CCS versus alternatives | 2 | MIDDLE | 10 | 1 | 9 | 0 |
| C6.4 | Chain-wide contractual coherence | 2 | OUTER | 8 | 0 | 6 | 2 |
| C3.2 | Compression capacity, availability & redundancy | 2 | OUTER | 6 | 0 | 4 | 2 |
| C1.3 | Emitter FID decision complexity | 2 | MIDDLE | 5 | 0 | 5 | 0 |
| C2.2 | Routing flexibility and leakage exposure | 2 | OUTER | 5 | 2 | 3 | 0 |
| C6.3 | FID synchronisation across the value chain | 2 | OUTER | 4 | 2 | 2 | 0 |
| C2.1 | Infrastructure synchronisation across feeder assets | 2 | INNER | 3 | 2 | 1 | 0 |
| C2.3 | Balancing and contingency arrangements | 2 | MIDDLE | 3 | 0 | 3 | 0 |
| C4.3 | Single-point-of-failure exposure | 2 | OUTER | 2 | 0 | 0 | 2 |
| C5.4 | $CO_2$ specification bounded by storage constraints | 2 | OUTER | 2 | 0 | 2 | 0 |
| C4.5 | Execution capacity & schedule realism | 1 | OUTER | 2 | 1 | 1 | 0 |
| C5.3 | Store availability and revenue exposure | 1 | INNER | 2 | 0 | 2 | 0 |

**Critical Enabling View Analysis - Aggregated subs to their enabling dependency, with OUTER vulnerability and amount of Sub-Dependencies scattered over all risks**

| Enabling ID | Enabling Vulnerability | Appears in #/9 risks | Score_vulnerability | Scattered amount of subs in #9 risks | Red | Containing Risks # |
|---|---|---|---|---|---|---|
| C3.1 | $CO_2$Next system readiness & commitment | 5 | OUTER | 15 | 5 | 5 |
| C1.2 | $CO_2$ specification as access and cost criterion | 5 | OUTER | 12 | 3 | 5 |
| C6.6 | Permitting finality and legal operability | 5 | OUTER | 12 | 4 | 5 |
| C3.4 | Design constraints from external system drivers | 5 | OUTER | 9 | 1 | 5 |
| C2.2 | Routing flexibility and leakage exposure | 2 | OUTER | 5 | 2 | 2 |
| C6.3 | FID synchronisation across the value chain | 2 | OUTER | 4 | 2 | 2 |

## C.3.5. Risk analysis on propagation paths

This section shows the dominant propagations of all sub-dependencies that are unfolded under the nine risks of Aramis.



**RISK Paths Of Aggrated Sub in Enabling Dependencies**

| Primary_SUB | Secondary_SUB | Path Count | Path | Red |
|---|---|---|---|---|
| C1 | D2 | 47 | C1 → D2 | 7 |
| D1 | A2 | 25 | D1 → A2 | 3 |
| A2 | D2 | 13 | A2 → D2 | 2 |
| C2 | B2 | 12 | C2 → B2 | 4 |
| B1 | D2 | 11 | B1 → D2 | 0 |
| A2 | A1 | 10 | A2 → A1 | 1 |
| A1 | C2 | 9 | A1 → C2 | 1 |
| D1 | C1 | 9 | D1 → C1 | 0 |
| D2 | C1 | 8 | D2 → C1 | 0 |
| D2 | B1 | 6 | D2 → B1 | 0 |
| B1 | D1 | 5 | B1 → D1 | 0 |
| A2 | B1 | 5 | A2 → B1 | 0 |
| D1 | A1 | 4 | D1 → A1 | 0 |
| C1 | B2 | 2 | C1 → B2 | 2 |
| C2 | D1 | 2 | C2 → D1 | 0 |
| D2 | B2 | 2 | D2 → B2 | 0 |
| C1 | A2 | 2 | C1 → A2 | 1 |
| D2 | A1 | 2 | D2 → A1 | 0 |
|  |  | Total: 174 |  |  |

Analysis

***These paths are dominant underlying components of risks.***

Primary - Secondary
C1 -D2
D1 - A2

| **C1, D2** | Primary | Secondary | Ring |
|---|---|---|---|
| C3.1 $CO_2$Next system readiness & commitment | Implementation (C1) | System (D2) | OUTER |
| C3.3 Porthos system integration & readiness | Implementation (C1) | System (D2) | OUTER |
| C4.2 Safety-by-interface dependency | Implementation (C1) | System (D2) | MIDDLE |
| C4.4 Off-spec and upset handling procedures | Implementation (C1) | System (D2) | OUTER |
|  |  |  |  |
| **D1, A2** |  |  |  |
| C1.2 $CO_2$ specification as access and cost criterion | System (D1) | Strategic (A2) | OUTER |
| C6.1 Tariff formation and cost discovery | System (D1) | Strategic (A2) | OUTER |

***Red Development Indicators Items Inside Propagation Path (Propagation - Path Amount - Amount Red)***

| **C1, D2 - 47x - 5 Red** |  |  |
|---|---|---|
| $CO_2$Next commissioning alignment with Aramis start-up | Red | C3.1 |
| $CO_2$Next interface compatibility (pressure, quality, flow envelope) | Red | C3.1 |
| **D1, A2 - 25x - 3 Red** |  |  |
| First-mover disadvantage under tight specification | Red | C1.2 |
| Spec-driven capex/opex uncertainty at emitter level | Red | C1.2 |
| **A2, D2 - 13 x - 2 Red** |  |  |
| Volume leakage through arbitrage to non-Aramis routes | Red | C2.2 |
| Sequencing uncertainty between parallel feeder developments | Red | C2.2 |
| **C2, B2 - 12 x - 4 Red** |  |  |
| Legal finality versus reversibility | Red | C6.6 |
| **A2, A1 - 10x - 1 Red** |  |  |
| Industrial relocation or cross border storage | Red | C1.1 |
| **A1, C2 - 9x - 1 Red** |  |  |
| Micro-tunnel removal obligations | Red | C3.4 |
| **C1, B2 - 2x - 2 Red** |  |  |
| Sequencing uncertainty between parallel feeder developments | Red | C2.1 |
| **C1, A2 - 2x - 1 Red** |  |  |
| Contractor scarcity & schedule gate | Red | C4.5 |

Enabling Paths, Not Aggregated, Count in Risks

| **Path** | **Red** | **Orange** | **Green** | **Path Count** |  |  |
|---|---|---|---|---|---|---|
| C1-D2 | 5 | 0 | 10 | 15 | C3.1 | $CO_2$Next system readiness & commitment |
| C1-D2 | 2 | 1 | 0 | 3 | C2.1 | Infrastructure synchronisation across feeder assets |
| C1-D2 | 0 | 7 | 5 | 12 | C4.2 | Safety-by-interface dependency |
| C1-D2 | 0 | 9 | 2 | 11 | C3.3 | Porthos system integration & readiness |
| C1-D2 | 0 | 5 | 0 | 5 | C4.4 | Off-spec and upset handling procedures |
| D1-A2 | 3 | 3 | 6 | 12 | C1.2 | $CO_2$ specification as access and cost criterion |
| D1-A2 | 0 | 4 | 8 | 12 | C6.1 | Tariff formation and cost discovery |
| A2-D2 | 2 | 3 | 0 | 5 | C2.2 | Routing flexibility and leakage exposure |
| A2-D2 | 0 | 8 | 0 | 8 | C5.5 | Stores investment commitment logic |
| C2-B2 | 4 | 8 | 0 | 12 | C6.6 | Permitting finality and legal operability |
| B1-D2 | 0 | 6 | 2 | 11 | C6.4 | Chain-wide contractual coherence |
| B1-D2 | 0 | 3 | 0 | 3 | C2.3 | Balancing and contingency arrangements |
| A2-A1 | 1 | 9 | 0 | 10 | C1.1 | Cost-effectiveness of CCS versus alternatives |
| A1-C2 | 1 | 5 | 3 | 9 | C3.4 | Design constraints from external system drivers |
| D1-C1 | 0 | 7 | 2 | 9 | C4.1 | Offshore transport integrity and operability |
| D2-C1 | 0 | 3 | 5 | 8 | C1.4 | Total volume commitment reliability |
| D2-B1 | 0 | 4 | 2 | 6 | C3.2 | Compression capacity, availability & redundancy |
| B1-D1 | 0 | 2 | 3 | 5 | C6.2 | System integration governance |
| A2-B1 | 0 | 5 | 0 | 5 | C1.3 | Emitter FID decision complexity |
| D1-A1 | 0 | 3 | 1 | 4 | C6.5 | Strategic optionality and system rigidity |
| C1-B2 | 2 | 2 | 0 | 2 | C6.3 | FID synchronisation across the value chain |
| C2-D1 | 0 | 2 | 0 | 2 | C5.4 | $CO_2$ specification bounded by storage constraints |
| D2-B2 | 0 | 0 | 2 | 2 | C4.3 | Single-point-of-failure exposure |
| C1-A2 | 1 | 1 | 0 | 2 | C4.5 | Execution capacity & schedule realism |
| D2-A1 | 0 | 2 | 0 | 2 | C5.3 | Store availability and revenue exposure |

Dominant Propagation Paths of Subs Under the Nine Risks

| **Combination All Subs propagations under Risks** | **Red** | **Orange** | **Green** | **Totaal** |
|---|---|---|---|---|
| B2-C2 | 4 | 8 | 0 | 12 |
| B1-C1 | 3 | 12 | 4 | 19 |
| A2-D1 | 3 | 7 | 5 | 15 |
| C1-D1 | 3 | 8 | 2 | 13 |
| A2-C1 | 3 | 1 | 3 | 7 |
| C1-D2 | 2 | 10 | 4 | 16 |
| A1-D1 | 1 | 2 | 3 | 6 |
| A2-B2 | 1 | 5 | 0 | 6 |
| B1-D2 | 0 | 10 | 8 | 18 |
| A2-D2 | 0 | 5 | 6 | 11 |
| B1-B2 | 0 | 4 | 5 | 9 |
| A1-A2 | 0 | 8 | 0 | 8 |
| B1-D1 | 0 | 3 | 5 | 8 |
| A1-B2 | 0 | 0 | 4 | 4 |
| A1-C1 | 0 | 3 | 0 | 3 |
| A1-C2 | 0 | 3 | 0 | 3 |
| B1-C2 | 0 | 3 | 0 | 3 |
| B2-D1 | 0 | 3 | 0 | 3 |
| D1-D2 | 0 | 1 | 2 | 3 |
| A2-A1 | 0 | 1 | 0 | 1 |
| A2-B1 | 0 | 1 | 0 | 1 |
| A1-D2 | 0 | 1 | 0 | 1 |
| D2-A1 | 0 | 1 | 0 | 1 |

# D

# Porthos

## D.1. Extracted Top-Risks from the Porthos risk-register

CONFIDENTIAL

## D.2. Section linking the risks to enabling dependencies with influence destabilising the depedency

CONFIDENTIAL

## D.3. Porthos propagation paths with associated risks in path and explanation

CONFIDENTIAL

## D.4. Section showing specific analysis of Porthos propagation paths

CONFIDENTIAL

## D.5. Porthos quotes ordered in MDF contexts

CONFIDENTIAL

### D.5.1. Uncertainty quotes derived from interviews

CONFIDENTIAL

## D.6. Porthos quotes ordered in MDF contexts

CONFIDENTIAL

## D.7. Porthos derived dependencies information & elaboration

CONFIDENTIAL