From the isolated twin to the ecosystem

Almost every canonical account of a digital twin describes a pair. There is a physical thing, there is a virtual counterpart, and there is a data connection that keeps the second faithful to the first. The earliest formulation named three constituents – a physical entity, a virtual entity and a connection between them – and a widely used extension names five: physical entity, virtual entity, services, data and connections [1]. A second, orthogonal distinction sharpens what "faithful" means: Kritzinger et al. classify a physical-virtual pairing as a *digital model* when data flows manually in both directions, a *digital shadow* when data flows automatically from the physical object to the virtual one but not back, and a *digital twin* proper only when the flow is automated in both directions [2]. That classification matters throughout this chapter: a construct that merely observes its physical counterpart, without the ability to act back on it, is not yet a twin in this strict sense, however much the surrounding literature calls it one. Between them, the five-dimension model and the digital-model/shadow/twin distinction are the two pieces of background this chapter assumes a reader needs before anything else makes sense, and the digital twin literature that has grown up around this single-pair picture is by now large, cross-domain and reasonably well mapped by systematic reviews [3], [4].

The pair, however, is not how twins are encountered in practice once more than one of them exists. A production machine has a twin; so does the line it sits in, the factory that contains the line, the product moving through it, and the building that houses it. Each of these was very likely built by a different team, using a different toolchain, against a different metamodel, for a different purpose and – in the cases that matter most – under a different owner. The interesting questions stop being about how well one twin mirrors one asset and start being about what happens when many twins have to coexist, exchange information, depend on one another, and be assembled into something larger. Several research communities have converged on this observation independently. Franke and Koch define a digital twin ecosystem plainly as "a conglomerate of multiple interconnected Digital Twins"[5], distinct from a single twin regardless of that twin's internal complexity. Giulianelli et al. put the engineering consequence directly: existing digital twin work has focused on virtualising individual assets in vertical application stacks, whereas an ecosystem is a distinct abstraction representing multiple connected, heterogeneous twins [6]. Kuruppuarachchi et al. approach it from the industrial-collaboration side, contrasting the *discrete* twin that "can create value without depending on other systems" with the *composite* twin that spans systems and business boundaries, and listing trust, interoperability, governance, ownership, security and privacy as the obstacles to the latter [7]. Michael et al. frame the same shift as a systems-of-systems problem and observe that most research on twins as systems-of-systems largely ignores reusing existing digital twins [8]. Healthcare arrives at the same place by a different road: the Virtual Human Twin is described as "a distributed and collaborative infrastructure" assembled from technologies, data and models contributed by many parties [9].

The economic pressure behind this shift is not subtle. Building a single twin already requires marshalling a heterogeneous collection of models, data sources, tools and services, and coordinating them is itself a substantial engineering task [10]. If every twin is built from scratch, the cost per twin does not fall, and the technology stays what Niederer et al. call artisanal rather than industrial [11]. Reuse is the only route to scale, and reuse across teams and organisations is precisely what an ecosystem is for.

Research question

This chapter is organised around one question:

> **RQ:** When digital twins cease to be isolated artefacts and become members of an ecosystem, what capabilities must the surrounding infrastructure provide – with respect to *interoperability*, *networking*, *collaboration* and *hierarchical composition* – and where do current approaches fall short of providing them?

The four terms are not an arbitrary partition. They correspond to four distinct relations that can hold between twins, and the literature treats them as four largely separate research programmes with only partial membership overlap. Interoperability concerns whether two twins can *understand* each other. Networking concerns whether they can *reach* each other, and at what cost in latency, bandwidth and fidelity. Collaboration concerns whether they *may* exchange anything at all, given that they answer to different owners with different interests. Hierarchical composition concerns whether a set of twins can be treated as a single twin at a coarser granularity, and what that new twin is entitled to claim about the system it represents.

A recurring argument of this chapter is that these four are not independent. Interoperability is usually posed as a data-representation problem, but the choice of representation determines what can later be composed hierarchically; networking is usually posed as a performance problem, but placement decisions across the edge-cloud continuum decide which collaborations are even feasible; and collaboration is usually posed as a trust problem, but the mechanisms that establish trust – policy-governed exchange, decentralised identity, provenance – are exactly the mechanisms that make a composed twin auditable. Treating the four separately is how ecosystems end up with four partial solutions that do not fit together.

A running example

Abstract arguments about ecosystems are readily agreed with and hard to test, so this chapter uses one scenario throughout. It is deliberately mundane, and deliberately spans ownership boundaries.

A contract manufacturer operates a production cell. The cell contains a six-axis robot supplied by a robot vendor, a conveyor supplied by a second vendor, and a vision system supplied by a third. The robot vendor ships a twin of the robot as part of its service contract: it carries a kinematic and dynamic model, wear estimates and a maintenance-scheduling service, and the vendor considers the model itself commercially sensitive. The manufacturer has built its own twin of the cell for cycle-time optimisation, and its customer – an automotive tier-one supplier – maintains a twin of the product being assembled, for quality traceability. The building the cell sits in has a twin too, operated by the facility owner, which prices energy and manages heating, ventilation and air conditioning; the grid operator would like a twin-level view of the building's flexible load.

Nothing in this arrangement is exotic. Every question this chapter asks has a concrete form in it. Can the manufacturer's cell twin interpret the robot twin's wear estimate, given that the two were described in different formats (Section 3)? What happens to the cell twin's cycle-time predictions when the plant network degrades and the robot twin's state goes stale (Section 4)? On what basis does the robot vendor allow its twin to answer the manufacturer's

queries without disclosing the model behind them, and what does the tier-one supplier's audit require (Section 5)? And is the manufacturer's cell twin – which aggregates the robot, conveyor and vision twins – entitled to claim it represents the cell, when the robot twin was validated only for the vendor's own duty cycle (Section 6)?

The last question is the one that recurs. It is the easiest to state, the hardest to answer, and the one on which the practical value of the other three depends.

Structure of the chapter

Section 2 takes an inventory of the vocabulary, because the same phenomenon currently travels under several names, and the differences between them are sometimes substantive and sometimes not. Sections 3–6 treat interoperability, networking, collaboration and hierarchical composition in turn, each closing with an assessment of what is settled and what is not. Section 7 synthesises the four into a capability model that answers the research question directly, and Section 8 states the open problems that synthesis exposes, including places where the corpus consulted for this chapter offers no answer at all.

What is a digital twin ecosystem?

An inventory of names

The phenomenon described above is not short of labels, and the differences between them repay attention.

Ecosystem of digital twins. The most common term, used for a set of twins that are connected, heterogeneous, and not centrally designed. It carries a deliberate biological connotation: members join and leave, and are not all built by the same party. Burattini et al.'s operating-room system uses the word in this sense [12], as does Martinelli et al.'s hierarchical industrial ecosystem [13].

Web of Digital Twins (WoDT). A stronger claim than "ecosystem": it commits to openness, to hypermedia-style discovery, and to twins forming a service-oriented layer that applications – including multi-agent systems – run on top of. Ricci et al. argue that the dominant view in the literature had been "about the virtualisation of individual physical assets in a closed-system perspective", proposing instead an open, distributed and dynamic ecosystem of connected twins [14]. The commitment is architectural, not merely metaphorical, and has been carried into a concrete open-source toolchain [6], [15].

Digital twin federation. Emphasises that members retain autonomy and local authority while participating in a larger whole. Marah and Challenger define federation as "a unified, integrated, interconnected, interoperable, and evolving ecosystem that couples and comprises a group of digital twins" [16]. The term travels well into settings with real organisational boundaries, such as cross-municipal urban mobility assessment [17] and federated scientific computing infrastructure [18].

Composite digital twin. Used where the emphasis is on crossing business boundaries to produce a single logical twin from parts belonging to different stakeholders, with the attendant trust and governance requirements [7].

Digital twin system-of-systems. The software-engineering framing, in which twins of smaller systems become parts of, or communicate with, twins of larger systems, and the classical systems-of-systems concerns about independently developed and independently evolving constituents apply [8].

System of digital twins and services. Human et al.'s term for an aggregated set of twins coupled with a services network, chosen to keep the services – not only the twins – in view as first-class design objects [19].

Digital twin continuum / fluid computing. Framings that foreground the deployment substrate: twins spread across a device-edge-cloud continuum, migrating as conditions change [20], [21].

Consortium glossaries have attempted to fix some of this vocabulary, and lean on the older notions of the digital thread and of system-of-systems to do so [22], [23]. The attempt has not converged, and the disagreement is not purely verbal: "federation" implies retained sovereignty, "composite" implies a single resulting twin, and "web" implies open discovery. A design that satisfies one does not automatically satisfy the others.

Table 1: Terms in current use for collections of digital twins, and the commitment each one carries beyond "more than one twin".

Term	Distinctive commitment	Representative source
Ecosystem of DTs	Heterogeneous, not centrally designed, members join and leave	[12]
Web of Digital Twins	Open discovery, hypermedia, consumed by agents	[14]
DT federation	Members retain local authority and sovereignty	[16]
Composite DT	Parts from several owners yield one logical twin	[7]
DT system-of-systems	Independently developed, independently evolving constituents	[8]
System of DTs and services	Services kept as first-class design objects	[19]
DT continuum / fluid computing	Deployment substrate spans device, edge and cloud	[20]

A working definition

For the purposes of this chapter, a *digital twin ecosystem* is a set of digital twins, built and operated by more than one party, that are able to discover one another, exchange information

with defined semantics, and be combined into larger twins or twin-backed services without being redesigned.

Three clauses in that definition are doing work. ”More than one party” rules out the case where a single vendor’s platform hosts many twins under one authority – that is a large deployment, not an ecosystem, and it does not face the governance problems that dominate Section 5. ”Defined semantics” rules out the case where twins merely exchange bytes: Combemale et al. warn that if plug-and-play capability “stays at the technical level” then the semantics of what was exchanged is lost [24]. ”Without being redesigned” is the economic clause: an assembly that requires each participating twin to be rewritten is an integration project, not an ecosystem.

Four relations between twins

Given that definition, four relations can hold between two twins A and B, and they map onto the four sections that follow.

1. A **can interpret** B**’s description and state.** This is interoperability (Section 3).
2. A **can reach** B **within a bounded time.** This is the network relation (Section 4), and it is not a given: twins are real-time-coupled artefacts whose usefulness degrades, sometimes discontinuously, when the coupling does [25].
3. A **is permitted to obtain what it obtains from** B**, and** B **is willing to provide it.** This is the collaboration relation (Section 5), and it is where ownership, intellectual property, privacy and liability enter [7], [8].
4. A **and** B**, together, constitute a twin** C **of the composite physical system.** This is the composition relation (Section 6), and it is by some distance the least well understood of the four [8], [26].

A disagreement worth stating early

Before proceeding it is worth naming a fault line that runs through the whole of what follows. One camp treats ecosystem members as essentially passive: a twin is a queryable, subscribable representation of an asset, and the intelligence lives in the applications above the twin layer. The hierarchical industrial ecosystem of Martinelli et al. is explicit about this, describing the ecosystem as an abstraction layer that decouples applications from the management of physical assets [13]. The other camp treats ecosystem members as agents: Esterle et al. describe autonomous cyber-physical systems that meet spontaneously and use digital twin models to improve collaboration and mutual safety [27], and later extend the idea to autonomous reconfiguration at runtime [28]; the Web of Digital Twins is designed explicitly to be consumed by multi-agent systems [14].

This is not a difference of emphasis. If twins are passive, the ecosystem’s hard problems are representation and plumbing. If twins are agents, the hard problems include negotiation, conflict resolution and mutual verification of claims, and none of the standardisation work discussed in Section 3 addresses those. The chapter returns to the consequence in Section 7.

Interoperability

Levels, not a level

Interoperability is routinely spoken of as though it were a single property, and this corpus's evidence does not support that. Combemale et al.'s model-based framework of digital twin integration challenges treats interoperability as a stratified problem, and warns specifically that a plug-and-play capability that "stays at the technical level" loses the semantics of what was exchanged [24]. At the *technical* level, two twins share a transport – MQTT, HTTP, OPC UA – and this is broadly solved; it is not where ecosystems fail. At the *syntactic* level, they share a serialisation and a schema, so a message can be parsed. At the *semantic* level, they agree on what the parsed fields mean. Beyond that, *organisational* interoperability – whether an exchange is permitted, accountable and contractually meaningful – is a distinct requirement that the Digital Twin Consortium's interoperability framework names directly, observing that information may originate in multiple domains and so may carry "disparate syntactic and semantic standards that must be correlated within a digital thread" [23]. The bulk of the digital twin standards effort, as the rest of this section shows, operates at the syntactic and semantic levels; Sections 4 and 5 argue that the dynamic and organisational levels are where ecosystems actually break.

The standards landscape

The Asset Administration Shell (AAS) is the most institutionally committed of the available standards. Its specification adopts the design principle "keep it simple but do not simplify if it affects interoperability" [29], and an ecosystem of open-source implementations and submodel templates has grown around it [30]. Zhang et al. examine the three AAS types against requirements extracted from popular digital twin definitions and conclude that each type contributes to engineering specific twin components, while leaving open challenges for implementing complete twins [31]. Ellwein et al.'s systematic mapping goes further: even the interaction semantics of the AAS – what an "active" shell is entitled to do, whether it makes decisions, exercises skills or negotiates – differ substantially across the sources that describe it [32]. A standard whose own community disagrees about its interaction model is not yet a settled basis for pragmatic-level interoperability.

Alongside the AAS sit the Digital Twins Definition Language (DTDL), OPC UA, the W3C Web of Things Thing Description, and, in manufacturing, the ISO 23247 series [33], [34]. Surveys of the standards relevant to digital twins reach catalogue length, evidencing that the landscape is diverging rather than converging.

Three strategies, and the disagreement between them

Faced with plurality, three strategies are visible in the corpus, and they are genuinely in tension.

Converge. Pick one metamodel and make everything speak it. This is the implicit programme of the AAS community. Its attraction is obvious; its cost is that a single metamodel must serve manufacturing, healthcare, civil infrastructure and energy, and this corpus's evidence that it can do so is thin.

Bridge. Accept plurality and build transformations between metamodels. Schmidt et al. implement and test a generic DTDL-to-AAS mapping, observing that existing standards "are incompatible with each other" in both syntax and semantics [35]. Cavalieri and Gambadoro map DTDL to OPC UA on the same reasoning [36]. Mattila et al. go a step further and work backwards from the task: they identify the data fields actually required for machine-to-machine collaboration, check which description formats carry them, and demonstrate the resulting mapping by controlling an industrial machine described simultaneously as an AAS document and a WoT Thing Description [37]. This is the most convincing evidence in the corpus that bridging is tractable, precisely because it is requirement-driven rather than schema-driven.

Leave. Abandon the industrial metamodels for Web architecture. Ricci et al. argue for hypermedia-style twins forming an open service layer [14]; Burattini et al. propose a core ontology and a Linked Data approach to state management [38]; and Giulianelli et al. turn this into an engineering method and a toolchain [6], [15]. The strategy inherits the Web's strengths in discovery and evolution – and, as the Web of Things survey documents, also inherits a fragmentation problem of its own that more than a decade of standardisation has not resolved [39].

The three strategies are not reconcilable by pursuing all of them at once. A bridge is only worth building if the endpoints are stable, which *converge* denies and *leave* makes irrelevant.

Does anyone follow the standards?

A finding that deserves more attention than it usually gets: Ferko et al. compared digital twin architectures documented in the literature against the ISO 23247 reference architecture and reported, from their own project experience, that "standards, and in particular the ISO 23247 standard, are not completely followed" [40]. This sits uncomfortably beside analyses that present the same standard as a generic framework with real international implementations, including its adoption by the Change2Twin project consortium [33], [34]. Both can be true – a standard can be well designed and poorly adopted – but for the ecosystem argument the adoption fact dominates the design fact. An ecosystem built on the assumption that members conform to a reference architecture will encounter members that do not.

The same gap between specification and practice appears at platform level. Pfeiffer et al.'s comparison of the modelling capabilities of commercial digital twin platforms finds substantial differences in what those platforms' metamodels can actually express, despite similar marketing claims [41]. A later comparison table across Kritzinger et al., Tao et al., ISO 23247 and the AAS/IDTA model shows the same underlying concepts – physical object, digital object, data, communication, service, twin – named differently across frameworks, which is the fragmentation problem restated at the conceptual level [42]. A case-study survey of open-source digital twin frameworks corroborates this from the tooling side, finding wide variation in whether a given framework supports syntactic-only or syntactic-and-semantic interoperability [43].

Semantics: ontologies and knowledge graphs

Semantic-level interoperability in this corpus almost always means ontologies. Kamburjan et al. use an OWL-based knowledge base not merely to describe a twin but to drive its

reconfiguration – detecting lifecycle changes and planning architecture changes – making the ontology load-bearing rather than documentary [44], and demonstrate the approach concretely in the GreenhouseDT exemplar [45]. Semantic reflection lifts runtime state into a knowledge graph connected to a domain ontology, giving other parties a principled way to interrogate a twin's current condition rather than a bespoke API [46]. Ontology also underpins the cognitive-twin programme, where it is treated as the basis for unified knowledge description and for integrating twins across lifecycle phases [47], and service-level description has been given the same treatment so that twin services – not only twins – can be discovered and reused [48].

The limitation is worth stating plainly: ontologies solve semantic agreement conditional on the parties having agreed on the ontology, which relocates the coordination problem rather than dissolving it. This is why the organisational level, discussed in Section 5, cannot be treated as an afterthought.

The digital thread and interoperability across the lifecycle

There is a third axis besides "across twins" and "across levels", and it is the one industrial practice cares about most: interoperability *across the lifecycle*. The digital thread names the requirement that information about an asset – its design intent, as-built configuration, service history – remain linked and interpretable as the asset moves between the organisations responsible for each phase [23]. For the running example, the thread is what would let the tier-one supplier trace a quality defect back through the cell twin to the specific robot trajectory and batch of material. The obstacle is that each phase typically has its own authoritative representation, and the mappings between them are lossy. This is acute in civil infrastructure, where the model architectures used at design time and those used in operation come from different traditions entirely, and interoperability solutions have had to be retrofitted [49]. Lifecycle-spanning analytics architectures have been proposed as a partial answer [50].

The thread also has a governance face. Once information persists across lifecycle phases and organisations, provenance becomes a requirement rather than a nicety – one route by which digital product passport work and dataspace work converge on the same infrastructure [51], [52].

Behavioural interoperability

Everything above concerns the exchange of *state*. Ecosystems also need the exchange of *behaviour*: twin *A*'s simulation must be able to be stepped in concert with twin *B*'s. This is co-simulation, and it has its own mature standards discussion, including the treatment of hybrid continuous-time and discrete-event coupling and the research needs that remain open [53]. Tooling has begun to address the mechanics: CoFMPy provides unified coordination of co-simulation, algebraic-loop resolution and distributed data exchange for FMI-based twins [54], and TwinOps combines FMI-based co-simulation of engineering models with run-time system monitoring [55]. But the composition of *validity* claims, as opposed to the composition of solvers, remains open, and is taken up in Section 6.

Interim assessment

Technical and syntactic interoperability are close to solved problems dressed up as open ones. Semantic interoperability has good mechanisms and a coordination problem: an ontology only helps once the parties have agreed on it. Pragmatic interoperability – agreeing on what an exchange commits the parties to – is barely addressed outside the AAS interaction-type debate [32] and Combemale et al.'s warning about semantics lost to technical plug-and-play [24]. Dynamic interoperability, meaning continued agreement under independent evolution, is named as a challenge but has, in the corpus surveyed here, no worked solution.

Networked digital twins

Two readings of "network"

The phrase "digital twin network" is used for two different things in this corpus, and conflating them causes confusion. The first reading, dominant in the software-engineering literature, is *twins that are networked*: the communication fabric over which ecosystem members exchange state and invoke each other's services. The second, dominant in the communications literature, is *twins of networks*: using a twin to model, predict and optimise a telecommunications network itself. Wu et al.'s survey defines the digital twin network in the second sense, realising co-evolution between physical and virtual spaces through modelling, communication, computing and data-processing technologies [56], and the 6G literature follows suit [57], [58]. This chapter is primarily concerned with the first reading, but the second is not irrelevant, as the end of this section argues.

Vaezi et al. bridge the two readings usefully: accurate real-time synchronisation between a physical system and its twin is essential, from which it follows that appropriate networking support is a key, not incidental, component of digital twin development and applications [25]. The network is not infrastructure that a twin happens to sit on; it is a determinant of whether the artefact is a twin at all, in the strict Kritzinger et al. sense introduced in Section 1.

Connectivity topologies

The most useful structural vocabulary for networked twins in this corpus comes from Schroeder et al., who specify six connectivity topologies: disconnected, connected, embedded, aggregated, multi-device and combined [59]. Each carries different ecosystem consequences. A *disconnected* twin has no live link to its asset and is, in Kritzinger et al.'s terms, a digital model [2]; it can still participate in an ecosystem as a source of design-time knowledge, but its state claims are not current. An *embedded* twin runs on the physical device itself, which bounds its computational resources but makes it robust to network loss. The *aggregated* topology, where a twin is composed of other twins – "the digital twin is composed of other digital twins" [59] – is treated in Section 6.

An ecosystem will contain members in several of these topologies simultaneously, and a member's topology determines what interaction patterns it can support: an embedded twin on a constrained device cannot serve arbitrary semantic queries. The topology is a deployment property, not a modelling property, which means it can change at runtime – and an ecosystem's interoperability contracts must survive that change.

The network as a first-class design constraint

Ecosystem architectures written from the software side often assume reachable, low-latency, reliable communication. The literature that has measured this does not. Frasheri et al. address the situation directly: what should a twin do when it and its physical counterpart get out of sync due to network degradation or a temporary drop? Their answer is a best-effort protocol comprising user notification, deliberate degradation of the twin to a digital shadow, and explicit recovery mechanisms to re-establish synchronisation [60]. This is, on the evidence assembled here, the single most consequential idea in the networking part of the ecosystem literature. It says that fidelity is a runtime property that can be lost and regained, that a twin should know when it has lost it, and – crucially for ecosystems – that a twin's declared fidelity is something its peers must be able to observe. An ecosystem in which members silently serve stale state as though it were live is worse than one with no ecosystem at all, because the staleness propagates through every composition built on top of it. Picone et al.'s experimental study of cybersecurity attacks on digital twin metrics gives this a quantitative face: packet loss and jitter, replay attacks and node failure each degrade timeliness, reliability, availability and consistency by measurably different amounts, confirming that fidelity loss is not a uniform, all-or-nothing event [61].

Upstream of the network sits the sensing layer, whose quality bounds everything above it [62]. Downstream sits the messaging substrate, where publish/subscribe systems dominate and where benchmarking under realistic domain-based workloads is only beginning to be done systematically [63]. Between them, quantitative evidence about ecosystem middleware is scarce: a systematic mapping study of digital twin middleware confirms that middleware is required to mediate interactions across a twin's whole lifecycle, but finds comparatively little measurement of how well existing middleware does so [64]. Franke and Koch's benchmark of FIWARE as a scalable interface for decentralised twin ecosystems, complete with a candid account of its limitations, is one of the few studies in this corpus that measures rather than asserts [5].

Placement across the edge-cloud continuum

Where a twin runs is an ecosystem-level decision, not a local one. Bellavista et al. make the strongest version of this argument with an entanglement-aware middleware: a highly dynamic and distributed ecosystem where containerised twins co-evolve with an orchestration middleware that monitors and reconfigures deployments "in light of application constraints, available resources, and the quality of cyber-physical entanglement" [65]. The phrase "quality of cyber-physical entanglement" is the important one: it makes the tightness of the twin-asset coupling into a measurable quantity that the infrastructure optimises, rather than an assumption the architecture makes. The same group drives the twin lifecycle along the cloud-to-edge continuum using microservices and serverless techniques [66], and related work frames the whole arrangement as a continuum [20] or as fluid computing [21].

The performance literature supplies the optimisation problems that placement raises: twin-assisted task offloading with edge collaboration [67] and the broader digital-twin edge-network programme toward 6G [58]. These are worth reading with a caution: they typically assume a single optimising authority with global visibility, which is exactly what a multi-party ecosystem does not have.

Twins of networks, and why an ecosystem should want one

The second reading of ”digital twin network” deserves more than the dismissal it usually gets in software-engineering treatments. Wu et al. survey the modelling, communication, computing and data-processing technologies that make a digital twin network work, and the application scenarios – manufacturing, aviation, healthcare, 6G networks, intelligent transport and urban intelligence – in which it has been proposed [56]. The 6G literature treats the twinned network as an architectural component of the network itself, used for what-if evaluation of configurations before they are applied to live infrastructure [57], [68].

A twin of the ecosystem’s own network is one credible way to supply the observable-fidelity signal that Section 4 argues an ecosystem member’s peers need: it can predict, rather than merely report, that the path to a given twin is about to degrade – the difference between a composed twin that flags reduced confidence in advance and one that discovers the problem in its outputs. In the running example, a twin of the plant network would let the manufacturer’s cell twin anticipate a communication drop instead of reacting to it. No source consulted here closes this loop explicitly – the network-twin literature and the twin-ecosystem literature are largely disjoint, citing different venues and different foundational papers. Connecting them is a modest, and on this evidence unclaimed, research opportunity.

Interim assessment

Networking is the best-instrumented of the four concerns – it inherits decades of measurement culture – and the worst-integrated with the others. The concepts that matter most for ecosystems are observable fidelity under degradation [60] and entanglement quality as an orchestration signal [65]. Neither appears in the interoperability standards discussed in Section 3: no widely used twin description format in this corpus has a field for ”how stale is this, and how would you know”. That omission is a concrete, fixable gap, and it recurs in Section 8.

Collaborative digital twins

Two senses of collaboration

”Collaborative digital twin” covers two distinct ideas, and both belong in this chapter. The first is *twins that collaborate*: twins of separate assets cooperating at runtime to achieve something neither could alone. Esterle et al. describe autonomous cyber-physical systems that meet spontaneously and use twin models to improve collaboration and mutual safety, proposing an architecture for self-integration and self-improvement at runtime [27], and extend the idea to autonomous reconfiguration [28]. Gil et al. address the modelling problem for cooperative robotic systems, extending information modelling to include behaviour and using an ontology to express the semantic relationships between the cooperating parts, demonstrated on two cooperative robotic arms [69].

The second is *people and organisations collaborating through twins*: the twin as the shared artefact across an organisational boundary. This is Kuruppuarachchi et al.’s composite twin spanning “systems and business boundaries” [7].

The two senses interact badly in one specific way. Runtime collaboration between twins

requires low-friction, low-latency interaction. Cross-boundary collaboration requires policy checks, provenance and consent. An ecosystem that supports both has to reconcile a millisecond budget with something closer to a legal review, and the corpus consulted for this chapter has not produced a design that does so convincingly.

The non-functional obstacles

Kuruppuarachchi et al.'s list is the best short statement of what stops collaboration: trust, interoperability, governance, ownership, security and privacy [7]. Michael et al.'s enumeration of integration challenges reaches the same places from the engineering side, naming protection of intellectual property, privacy aspects of data, and rights and roles in the integrated twin as distinct problems alongside the purely technical ones [8]. Combemale et al. add prioritisation – whose objective wins when two twins disagree – and observe that composing digital twins may inadvertently expose information, including intellectual property and privacy data, even when the aggregate complies with regulation and ecosystem policies [24]. Alcaraz and Lopez's security survey corroborates this from the threat-modelling side: confidentiality threats have the greatest impact of any digital twin threat category precisely because a digital model is, by construction, close to an exact copy of a physical counterpart's intellectual property [70].

Note what this implies for engineering practice. Intellectual-property protection is not a matter of encrypting a channel; it is a matter of a twin being able to answer useful questions about an asset without revealing the model that answers them. That is a design constraint on the twin's interface, which means it belongs in the interoperability discussion of Section 3 and is largely absent from it.

Dataspaces and sovereignty

The most developed institutional answer to cross-organisational exchange in this corpus is the dataspace. Singh et al. describe International Data Spaces-family dataspaces as facilitating sovereign, policy-governed data exchange via identity management and dynamic attribute provisioning [71], and motivate the need from the observation that businesses must adopt an open collaboration approach, sharing data and services in a governed, secure manner rather than competing outright [72]. A companion line of work proposes distributed-ledger technology specifically to solve digital trust, traceability, transparency and governance for shared data and services among dataspace stakeholders [73]. Schmidt et al. address the integration of AAS-based twins into dataspaces directly, reporting that this integration "remains challenging due to manual processes, synchronization issues, and varying implementations" [52]. Gleich et al. give a worked artefact: an AAS-based Digital Product Passport delivered as a Gaia-X service, demonstrating safe and trustworthy data exchange across a value chain [51].

Dataspaces supply what Section 3 called organisational interoperability: identity, policy, usage control and auditability. What they do not yet supply, on this corpus's evidence, is the pragmatic level. Knowing that party B is permitted to read a temperature series does not establish what B is entitled to conclude from it, or what obligation A incurs if the series is wrong.

Trust, identity and the disappearing perimeter

An ecosystem has no defensible perimeter. Every twin is a networked service with a data path to a physical asset, which is why security surveys treat twins as a distinctive threat class rather than as ordinary software [70]. Kulik et al. ground this in the structural fact: twin-enabled systems "depend on communications and networking, and so face a range of threats" [74]. NIST's zero trust architecture, which treats network location as no longer central to security posture, is the natural fit for a perimeterless ecosystem [75], and its practical precondition in a multi-domain setting is decentralised identity: authentication and authorisation that work when resources are shared across administrative domains in a computing continuum [76]. Twins also cut the other way, as instruments *for* security rather than only as targets – Qureshi et al. survey their use for simulation-based attack testing and secure-by-design evaluation [77].

Collaborating without sharing

A recurring pattern deserves separate mention because it dissolves rather than manages the sharing problem: parties collaborate on a model without exchanging the data that trained it. Lu et al. combine communication-efficient federated learning with a permissioned blockchain for twin edge networks [78]. Kim et al. pose federated twins as a scheduling problem and solve it with temporal graph neural networks and deep reinforcement learning [79]. Marah and Challenger's federation roadmap positions this within a broader programme of integration, collaboration and coordination across an ecosystem of federated twins [16], and research infrastructures have adopted federated computing and data as the organising principle for scientific twins for exactly this reason: no single institution holds either [18].

The limitation is scope. Federated learning solves collaborative *statistical* model-building. It does nothing for the case where the shared artefact is a physics-based model, a control decision, or a claim about the current state of a shared asset – which is most of what twins do.

Humans in the ecosystem

Industry 5.0 framings insist that people are ecosystem participants rather than users of it. Villani et al. present a twin-driven human-centric ecosystem on exactly this premise, centring worker well-being in production planning [80], and Tóth et al. propose a collaboration architecture integrating human, AI, IoT and robot agents through a generic semantic definition backed by a knowledge graph [81]. Human twins are being built from multimodal clinical data to support nurse practitioners' decision-making [82], which raises the privacy and consent questions of this section in their sharpest form. At the institutional scale, the social and human dimensions of twin technologies are themselves treated as a research object: twin governance requires inclusive, equitable institutional design, and twins land inside formal and informal institutions whose existing decision rights they do not automatically respect [83].

Collaboration at city and sector scale

The manufacturing setting of the running example understates the governance problem, because a supply chain at least has contracts. Two settings in this corpus push further. The

first is the city. Modular, adaptive architectures have been developed to make integration across mobility, energy, water, buildings and emergency services – each separately owned – tractable for real-time urban management [84], and methods exist for accelerating twin development specifically in the smart-city setting [85]. Cross-municipal federation has been given a functional architecture for mobility assessment [17], and the value case has been worked through for flexible building loads coordinated across a network of participants, with a composability platform and a business marketplace for secure supply-chain ordering where governance and trust are paramount [86].

The second is the research infrastructure, where scientific twins are being built on federated computing and federated data because no single institution holds either [18], and where the health-domain equivalent is explicitly framed as a distributed and collaborative infrastructure assembled from contributed technologies, data and models [9]. What both settings share is that the ecosystem's governance predates its technology. The participants already have decision rights, funding lines and accountability structures, and the twin ecosystem must fit them rather than replace them – the strongest practical argument, on this evidence, for the federation framing over the composite framing: it is the only term in Table 1 that treats retained local authority as a design requirement rather than an obstacle.

The supply side: marketplaces and reuse

If ecosystems are justified by reuse, someone must supply the reusable parts. The Change2Twin project provides the most concrete evidence available here: a catalogue of digital twin enabling technologies documenting the fragmentation of the tool landscape [87], and a marketplace, reviewed in operation [88]. HUBCAP demonstrated the adjacent case for model-based design of cyber-physical systems, providing collaboration infrastructure aimed specifically at small and medium enterprises that cannot absorb the fixed costs alone [89]. Value-creation studies show the same logic at the application level, with buildings-as-batteries value realised through a mass-customisation network of participants rather than by any single twin [86].

Interim assessment

Collaboration has the clearest problem statement of the four concerns [7], [8] and the most institutional machinery pointed at it [51], [52], [72]. What it lacks, on this corpus's evidence, is a connection to the technical layers: dataspace policy is expressed over datasets, whereas twins expose services, simulations and control affordances, and no source consulted here expresses usage policy over those. Section 8 returns to this.

Hierarchical and composed digital twins

Aggregation as a topology

The hierarchical case has a clean statement in Schroeder et al.'s aggregated topology: "the digital twin is composed of other digital twins …It can be seen as a hierarchical system, where the parent transparently provides all functionalities of its children, while possibly adding more functionalities" [59]. Their worked example is the standard one: the twin of a production line aggregates the twins of its machines and products, and the aggregation of

production lines yields the twin of the factory. They also note a structural consequence: a one-to-one physical-cyber correspondence can, in principle, be maintained at every level of the hierarchy simultaneously.

Martinelli et al. build this out into a running industrial ecosystem, exploiting twin relationships and hierarchies to build a digitalised replica of the whole manufacturing system structure, with the stated capabilities of data augmentation, actionability, navigability and composability, and they measure the resource consumption their hierarchy imposes [13]. Human et al. approach the same object as a design problem, giving a six-step framework for a system of aggregated twins and services running from needs analysis through physical-system decomposition and service allocation to verification and validation [19]. Both are valuable, and both assume a single designing authority – which returns the problem to the governance concerns of Section 5 the moment more than one party is involved.

What composition must preserve

An early reference model, C2PS, already proposed hierarchy-based composition of subsystems as one of five constituent elements of a cloud-based cyber-physical system [90]: the composition relation has been recognised for as long as the field has had reference models, without the field agreeing on what it preserves. Gill et al., working on automating the composition of digital twins within systems-of-systems, state that digital twins of individual components must be composed "efficiently and in accordance with" requirements including domain-specific semantic expression [26] – naming automation of composition as a goal, without supplying a semantics for what the composed twin may validly claim about the system it now represents. Combemale et al. pose composability explicitly as an open question and note a compounding effect: composing digital twins also composes their gateways, so interoperability mechanisms accumulate rather than cancel out under composition [91]. Michael et al. argue from the organisational side that because subsystems of a system-of-systems are typically controlled by different organisations, a single centrally managed digital twin of the complete system is unrealistic, and federating digital twins must reflect the compositional construction of the system being twinned [92].

For the running example, this is exactly the unanswered question: the manufacturer's cell twin aggregates the robot, conveyor and vision twins, but the robot twin was validated only for the vendor's own duty cycle. Nothing in the corpus consulted for this chapter supplies a mechanism by which the composed cell twin could state – automatically, and in a way its own consumers could check – that its claim about the cell now extends beyond what any of its parts were individually validated for. This is treated as an open problem in Section 8, not glossed over with the nearest available citation.

Platform support for composition

Practical platforms have made real progress on the mechanics of composition, even without resolving its semantics. Van Schalkwyk and Isaacs argue that composability and a lean, agile development approach together are what let digital twin programmes achieve scale and early results, illustrated with fielded examples [93]. OpenTwins is an open-source framework explicitly targeting "next-gen compositional digital twins" that link individual entities or subsystems into a higher-degree twin, sharing knowledge and data relationships [94], and has since been extended to distribute twins across nodes within the same framework [95]

– evidencing that composition and distribution are handled together in practice, not as separate concerns. Talasila et al.'s reusable-component platform for realising digital twins addresses the same problem from the supply side: building twins from reusable parts requires the parts to compose without redesign [10].

Interim assessment

Hierarchical composition has structural vocabulary [59], running systems [13], platform support for asset reuse [10], [93] and a clear statement of what automation would require [26], but – on the evidence gathered for this chapter – no semantics for what a composed twin is entitled to claim. Of the four concerns this chapter examines, this is the one where the gap between engineering capability and conceptual foundation is widest.

Synthesis: what an ecosystem demands of its infrastructure

Answering the research question

Read together, the four sections above answer the research question as a short list of capabilities, unequally supplied by the current literature.

C1 – Identity and discovery. An ecosystem member must be findable and identifiable across organisational boundaries. Dataspace identity management [71] and decentralised identity for multi-domain continua [76] are the most developed mechanisms; the Web of Digital Twins' hypermedia-style discovery is the most developed alternative outside the dataspace tradition [14], [15].

C2 – Semantic description of state and capability. Members must describe what they are and what they offer in terms other members can parse. The standards landscape of Section 3 supplies this unevenly and with three competing strategies (§3); ontology-backed approaches supply it more consistently, at the cost of requiring prior agreement on the ontology [44], [46].

C3 – Declared and observable fidelity. A member's claim about its own currency and accuracy must be visible to its peers, and must be able to change at runtime. This is the single clearest and most under-served capability in the corpus: Frasheri et al. supply the mechanism [60], but no standard reviewed in Section 3 has a field for it.

C4 – Bounded, adaptive communication. The network must be treated as a design constraint, not an assumption, with explicit handling of degradation [25], [60].

C5 – Placement and orchestration across a continuum. Where a twin runs must be revisable at runtime against a measurable coupling quality [20], [65].

C6 – Policy, sovereignty and accountability. Dataspaces supply this at the organisational level for data [52], [71], but not yet for the services, simulations and control affordances a twin actually exposes (§5).

C7 – Behavioural composability. Co-simulation standards let models be stepped together mechanically [53], [54], but the composition of validity claims about the result remains open (§6).

C8 – Reusable assets and a supply side. A marketplace or catalogue of reusable components, aimed particularly at organisations that cannot build everything themselves [10], [87], [89].

The couplings between concerns

None of these eight capabilities stands alone. C2 (semantic description) constrains C7 (behavioural composability): a co-simulation cannot compose sensibly if the twins being stepped together disagree about what their state variables mean. C4 (bounded communication) supplies the observation that C3 (declared fidelity) needs to be meaningful at all – a twin cannot honestly declare its own currency without knowing something about the path to its asset. C5 (placement) determines which C6 (policy) constraints are even satisfiable: a twin cannot honour a data-residency requirement if its placement decision is made without reference to sovereignty. And C1 (identity) is the precondition for every other capability being attributable to a specific, accountable party rather than an anonymous service. Treating the four research programmes in Sections 3 – 6 as separate, as most of the individual sources cited in this chapter do, is exactly how an ecosystem ends up with four partial solutions that do not fit together.

Passive layer or agent society, revisited

Section 2 named a fault line between ecosystems whose members are passive representations and ecosystems whose members are autonomous agents. The capability list above is largely agnostic to which camp is right for C1–C2 and C4–C6: identity, description, communication and policy are needed whichever camp turns out to be correct. C3 (declared fidelity) and C7 (behavioural composability) are not agnostic. A passive twin can be told its fidelity has degraded by an external monitor; an agent twin is expected to know and declare this itself, which is a materially harder self-modelling requirement. Composing the behaviour of passive twins is a solver-coupling problem; composing the behaviour of agent twins that negotiate is closer to a multi-agent systems problem that this corpus's interoperability and networking literatures do not address at all.

Open problems

P1 – A fidelity field in twin descriptions. No standard surveyed in Section 3 – AAS, DTDL, the ISO 23247 series – carries a field expressing declared fidelity or staleness in the sense Frasheri et al. define it [60]. This is a concrete, implementable gap: adding such a field to an existing description language is a smaller undertaking than any of the other problems

in this list, and it would let C3 and C6 above be expressed in the same artefact rather than in separate tooling.

P2 – Usage policy over twin services, not only data. The dataspace mechanisms surveyed in Section 5 express policy over datasets. No source consulted for this chapter expresses usage policy over a twin's services, simulations or control affordances – the things a twin actually exposes beyond raw data.

P3 – Connecting the network-twin and ecosystem-twin literatures. Section 4 identified two disjoint bodies of work that would, on inspection, plausibly strengthen each other: twinning the communication substrate itself would supply exactly the predictive, observable-fidelity signal the ecosystem literature needs. Neither cites the other in this corpus.

P4 – Compositional validity (gap). Section 6 found structural vocabulary for aggregation, running systems that use it, and a stated requirement that composition happen "in accordance with" semantic-expression requirements [26], but no source in this corpus supplies a semantics for what a composed twin is entitled to claim about the system it represents when its parts were validated separately, under different conditions. This is the sharpest unanswered question the running example raises, and the corpus consulted for this chapter does not answer it.

P5 – Liability and contract across composed and federated twins (gap). A national research-gaps report names "guidelines for identifying liability in the case of misuse" as an open problem for scaling digital twins, explicitly not a solved one [96]. No source consulted here goes further than naming the gap.

P6 – Comparative ecosystem benchmarks (gap). Section 4 found one benchmark of a single ecosystem interface's latency under a moderate load [5], and Section 5 found one comparison of digital twin *platforms* using qualitative decision-analysis methods rather than measurement [97]. Neither is a comparative, quantitative benchmark of competing ecosystem architectures against one another. This corpus offers no such benchmark.

Summary

This chapter asked what changes when digital twins become members of an ecosystem rather than isolated artefacts, and specifically what interoperability, networking, collaboration and hierarchical composition demand of the surrounding infrastructure.

The answer developed here is that the four demands are unequally met and badly connected. Interoperability has a rich set of mechanisms – description standards, transformations between them, ontologies and Linked Data – and a coordination failure about which to use, with three mutually incompatible strategies (converge, bridge, leave) in simultaneous pursuit and evidence that a flagship reference architecture is not, in practice, followed [40]. Networking is the most mature concern and contributes the chapter's most underappreciated idea, that fidelity is a runtime property which can be lost, observed and recovered

[60]. Collaboration has the clearest problem statement [7], [8] and the most institutional machinery, aimed one level too low: at datasets rather than at the services, simulations and control affordances that twins actually expose. Hierarchical composition has structural vocabulary [59], running systems [13], platform support for asset reuse [10], [93] and a clear statement of what automation would require [26], but no semantics for what a composed twin is entitled to claim.

Two conclusions follow for the remainder of this thesis. First, the ecosystem problem is not solved by any single layer: a platform that gets reuse right but cannot express sovereignty, or a standard that gets semantics right but cannot express staleness, will not compose into a working ecosystem. Second, the capability that falls between all four concerns – declared, observable fidelity – is both the cheapest to fix and the one on which the credibility of every composition rests.

[1] F. Tao *et al.*, "Five-dimension digital twin model and its ten applications," *Comput. Integr. Manuf. Syst*, vol. 25, no. 1, pp. 1–18, 2019.

[2] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital Twin in manufacturing: A categorical literature review and classification," *Ifac-PapersOnline*, vol. 51, no. 11, pp. 1016–1022, 2018, Accessed: Dec. 11, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896318316021

[3] C. Semeraro, M. Lezoche, H. Panetto, and M. Dassisti, "Digital twin paradigm: A systematic literature review," *Computers in Industry*, vol. 130, p. 103469, Sep. 2021, doi: 10.1016/j.compind.2021.103469.

[4] M. Dalibor *et al.*, "A Cross-Domain Systematic Mapping Study on Software Engineering for Digital Twins," *Journal of Systems and Software*, vol. 193, p. 111361, Nov. 2022, doi: 10.1016/j.jss.2022.111361.

[5] K. Franke and T. Koch, "FIWARE as a Scalable Digital Twin Interface for DT Ecosystems: Benchmark and Limitations," in *2025 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, Mar. 2025, pp. 134–139. doi: 10.1109/PerComWorkshops65533.2025.00054.

[6] A. Giulianelli, S. Burattini, A. Ciortea, and A. Ricci, "Engineering Interoperable Ecosystems of Digital Twins: A Web-based Approach," in *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems*, in MODELS Companion '24. New York, NY, USA: Association for Computing Machinery, Oct. 2024, pp. 476–485. doi: 10.1145/3652620.3688263.

[7] P. Kuruppuarachchi, S. Rea, and A. McGibney, "An architecture for composite digital twin enabling collaborative digital ecosystems," in *2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, IEEE, 2022, pp. 980–985. Accessed: Feb. 25, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9776073/

[8] J. Michael, J. Pfeiffer, B. Rumpe, and A. Wortmann, "Integration challenges for digital twin systems-of-systems," in *Proceedings of the 10th IEEE/ACM International Workshop on Software Engineering for Systems-of-Systems and Software Ecosystems*, in SESoS '22. New York, NY, USA: Association for Computing Machinery, Nov. 2022, pp. 9–12. doi: 10.1145/3528229.3529384.

[9] M. Viceconti, M. De Vos, S. Mellone, and L. Geris, "Position Paper From the Digital Twins in Healthcare to the Virtual Human Twin: A Moon-Shot Project for Digital Health Research," *IEEE Journal of Biomedical and Health Informatics*, vol. 28, no. 1, pp. 491–501, Jan. 2024, doi: 10.1109/JBHI.2023.3323688.

[10] P. Talasila, P. H. Mikkelsen, S. Gil, and P. G. Larsen, "Realising Digital Twins," in *The Engineering of Digital Twins*, J. Fitzgerald, C. Gomes, and P. G. Larsen, Eds., Cham: Springer International Publishing, 2024, pp. 225–256. doi: 10.1007/978-3-031-66719-0_11.

[11] S. A. Niederer, M. S. Sacks, M. Girolami, and K. Willcox, "Scaling digital twins from the artisanal to the industrial," *Nature Computational Science*, vol. 1, no. 5, pp. 313–320, May 2021, doi: 10.1038/s43588-021-00072-5.

[12] S. Burattini *et al.*, "An Ecosystem of Digital Twins for Operating Room Management," in *2023 IEEE 36th International Symposium on Computer-Based Medical Systems (CBMS)*, Jun. 2023, pp. 770–775. doi: 10.1109/CBMS58004.2023.00317.

[13] M. Martinelli, J. Zhang, A.-K. Splettstoßer, M. Picone, M. Lippi, and A. Wortmann, "Hierarchical Digital Twin Ecosystem for Industrial Manufacturing Scenarios," in *2024 50th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, Aug. 2024, pp. 56–63. doi: 10.1109/SEAA64295.2024.00018.

[14] A. Ricci, A. Croatti, S. Mariani, S. Montagna, and M. Picone, "Web of Digital Twins," *ACM Transactions on Internet Technology*, vol. 22, no. 4, pp. 1–30, Nov. 2022, doi: 10.1145/3507909.

[15] A. Giulianelli, S. Burattini, A. Ciortea, and A. Ricci, "HWoDT Framework: A toolchain to build interoperable Digital Twin Ecosystems," *SoftwareX*, vol. 31, p. 102275, Sep. 2025, doi: 10.1016/j.softx.2025.102275.

[16] H. Marah and M. Challenger, "(Re-)Engineering Digital Twins Towards Federation: Vision and Roadmap," in *Leveraging Applications of Formal Methods, Verification and Validation. Software Engineering Methodologies*, T. Margaria and B. Steffen, Eds., Cham: Springer Nature Switzerland, 2025, pp. 60–81. doi: 10.1007/978-3-031-75387-9_5.

[17] J. Li, J. Grübel, A. Nadi, M. Snelder, B. van Arem, and J. Gao, "Digital Twin Federation for Urban Mobility Assessment: A Functional Architecture for Low-Car Transformations in the Netherlands," 2025, Accessed: Oct. 25, 2025. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5370630

[18] "interTwin: Advancing Scientific Digital Twins through AI, Federated Computing and Data."

[19] C. Human, A. Basson, and K. Kruger, "A design framework for a system of digital twins and services," *Computers in Industry*, vol. 144, p. 103796, Jan. 2023, doi: 10.1016/j.compind.2022.103796.

[20] A. Barbone, S. Burattini, M. Martinelli, M. Picone, A. Ricci, and A. Virdis, "Digital Twin Continuum: A Key Enabler for Pervasive Cyber-Physical Environments," in *2024 33rd International Conference on Computer Communications and Networks (ICCCN)*, Jul. 2024, pp. 1–9. doi: 10.1109/ICCCN61486.2024.10637565.

[21] L. Bedogni, M. Mamei, M. Picone, M. Pietri, and F. Zambonelli, "Fluid Computing & Digital Twins for intelligent interoperability in the IoT ecosystem," *Future Generation Computer Systems*, vol. 171, p. 107855, Oct. 2025, doi: 10.1016/j.future.2025.107855.

[22] "Glossary of Digital Twins by Digital Twin Consortium," *Digital Twin Consortium*. Accessed: May 27, 2024. [Online]. Available: https://www.digitaltwinconsortium.org/glossary/glossary/

[23] A. Budiardjo and D. Migliori, "Digital twin system interoperability framework," Tech. rep. Digital Twin Consortium, East Lansing, Michigan, 2021. Accessed: Feb. 25, 2025. [Online]. Available: https://www.digitaltwinconsortium.org/pdf/Digital-Twin-System-Interoperability-Framework-12072021.pdf

[24] B. Combemale *et al.*, "On the Challenges of Integrating Digital Twins," in *2nd International Conference on Engineering Digital Twins (EDTconf 2025)*, 2025. Accessed: Sep. 29, 2025. [Online]. Available: https://inria.hal.science/hal-05221809/

[25] M. Vaezi *et al.*, "Digital twins from a networking perspective," *IEEE Internet of Things Journal*, vol. 9, no. 23, pp. 23525–23544, 2022, Accessed: Jan. 08, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9863238/

[26] M. S. Gill, J. Zhang, A. Wortmann, and A. Fay, "Toward Automating the Composition of Digital Twins Within System-of-Systems," in *2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Sep. 2024, pp. 1–4. doi: 10.1109/ETFA61755.2024.10710740.

[27] L. Esterle, C. Gomes, M. Frasheri, H. Ejersbo, S. Tomforde, and P. G. Larsen, "Digital twins for collaboration and self-integration," in *2021 IEEE International Conference on Autonomic Computing and Self-Organizing Systems Companion (ACSOS-C)*, IEEE, 2021, pp. 172–177. Accessed: May 25, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9599295/

[28] L. Esterle, M. Frasheri, and P. G. Larsen, "Autonomous Reconfiguration Enabled by Digital Twins," in *The Engineering of Digital Twins*, J. Fitzgerald, C. Gomes, and P. G. Larsen, Eds., Cham: Springer International Publishing, 2024, pp. 345–362. doi: 10.1007/978-3-031-66719-0_14.

[29] "Asset Administration Shell Part - 1."

[30] M. Jacoby *et al.*, "Open-Source Implementations of the Reactive Asset Administration Shell: A Survey," *Sensors*, vol. 23, p. 5229, May 2023, doi: 10.3390/s23115229.

[31] J. Zhang, C. Ellwein, M. Heithoff, J. Michael, and A. Wortmann, "Digital twin and the asset administration shell," 2024, Accessed: Apr. 09, 2025. [Online]. Available: https://awortmann.github.io/downloads/paper/Digital_twin_and_the_asset_administration_shell.pdf

[32] C. Ellwein *et al.*, "Rethinking Asset Administration Shell Communication Types: A Systematic Mapping Study and Portfolio-Based Classification," *Production Engineering*, vol. 20, Dec. 2025, doi: 10.1007/s11740-025-01378-3.

[33] G. Shao, "Use Case Scenarios for Digital Twin Implementation Based on ISO 23247," *NIST*, May 2021, Accessed: Mar. 06, 2024. [Online]. Available: https://www.nist.gov/publications/use-case-scenarios-digital-twin-implementation-based-iso-23247

[34] G. Shao, S. Frechette, and V. Srinivasan, "An Analysis of the New ISO 23247 Series of Standards on Digital Twin Framework for Manufacturing," American Society of Mechanical Engineers Digital Collection, Sep. 2023. doi: 10.1115/MSEC2023-101127.

[35] C. Schmidt, F. Volz, L. Stojanovic, and G. Sutschet, "Increasing Interoperability between Digital Twin Standards and Specifications: Transformation of DTDL to AAS," *Sensors*, vol. 23, p. 7742, Sep. 2023, doi: 10.3390/s23187742.

[36] S. Cavalieri and S. Gambadoro, "Proposal of Mapping Digital Twins Definition Language to Open Platform Communications Unified Architecture," *Sensors*, vol. 23, p. 2349, Feb. 2023, doi: 10.3390/s23042349.

[37] J. Mattila, R. Ala-Laurinaho, J. Autiosalo, and K. Tammi, "Interoperability of Digital Twins for Automation With Digital Twin Schema," *IEEE Access*, vol. 13, pp. 200595–200608, 2025, doi: 10.1109/ACCESS.2025.3633736.

[38] S. Burattini, A. Zimmermann, M. Picone, and A. Ricci, "Towards Linked Data for Ecosystems of Digital Twins," in *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems*, in MODELS Companion '24. New York, NY, USA: Association for Computing Machinery, Oct. 2024, pp. 332–337. doi: 10.1145/3652620.3688245.

[39] L. Sciullo, L. Gigli, F. Montori, A. Trotta, and M. Felice, "A Survey on the Web of Things," *IEEE Access*, vol. 10, pp. 1–1, Jan. 2022, doi: 10.1109/ACCESS.2022.3171575.

[40] E. Ferko, A. Bucaioni, P. Pelliccione, and M. Behnam, "Standardisation in Digital Twin Architectures in Manufacturing," in *2023 IEEE 20th International Conference on Software Architecture (ICSA)*, Mar. 2023, pp. 70–81. doi: 10.1109/ICSA56044.2023.00015.

[41] J. Pfeiffer, D. Lehner, A. Wortmann, and M. Wimmer, "Modeling capabilities of digital twin platforms-old wine in new bottles?" *J. Object Technol.*, vol. 21, no. 3, pp. 3–1, 2022, Accessed: Sep. 04, 2024. [Online]. Available: https://raw.githubusercontent.com/awortmann/awortmann.github.io/master/downloads/paper/Modeling_Capabilities_of_Digital_Twin_Platforms_-_Old_Wine_in_New_Bottles.pdf

[42] J. Pfeiffer *et al.*, "Towards a Unifying Reference Model for Digital Twins of Cyber-Physical Systems." arXiv, Jul. 2025. doi: 10.48550/arXiv.2507.04871.

[43] S. Gil, P. H. Mikkelsen, C. Gomes, and P. G. Larsen, "Survey on open-source digital twin frameworks–A case study approach," *Software: Practice and Experience*, vol. 54, no. 6, pp. 929–960, Jun. 2024, doi: 10.1002/spe.3305.

[44] E. Kamburjan, N. Bencomo, S. L. Tapia Tarifa, and E. B. Johnsen, "Declarative Lifecycle Management in Digital Twins," in *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems*, Linz Austria: ACM, Sep. 2024, pp. 353–363. doi: 10.1145/3652620.3688248.

[45] E. Kamburjan *et al.*, "GreenhouseDT: An Exemplar for Digital Twins," in *Proceedings of the 19th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, in SEAMS '24. New York, NY, USA: Association for Computing Machinery, Jun. 2024, pp. 175–181. doi: 10.1145/3643915.3644108.

[46] E. Kamburjan, A. Pferscher, R. Schlatte, R. Sieve, S. L. T. Tarifa, and E. B. Johnsen, "Semantic Reflection and Digital Twins: A Comprehensive Overview," in *The Combined Power of Research, Education, and Dissemination*, vol. 15240, M. Hinchey and B. Steffen, Eds., Cham: Springer Nature Switzerland, 2025, pp. 129–145. doi: 10.1007/978-3-031-73887-6_11.

[47] L. Jinzhi, Y. Zhaorui, X. Zheng, W. Jian, and K. Dimitris, "Exploring the concept of Cognitive Digital Twin from model-based systems engineering perspective," *The International Journal of Advanced Manufacturing Technology*, vol. 121, Aug. 2022, doi: 10.1007/s00170-022-09610-5.

[48] B. Oakes, C. Gomes, E. Kamburjan, G. Abbiati, E. Ecem Bas, and S. Engelsgaard, "Towards Ontological Service-Driven Engineering of Digital Twins," in *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems*, in MODELS Companion '24. New York, NY, USA: Association for Computing Machinery, Oct. 2024, pp. 464–469. doi: 10.1145/3652620.3688261.

[49] H. Naderi and A. Shojaei, "Digital twinning of civil infrastructures: Current state of model architectures, interoperability solutions, and future prospects," *Automation in Construction*, vol. 149, p. 104785, May 2023, doi: 10.1016/j.autcon.2023.104785.

[50] A. C. Marosi *et al.*, "Interoperable Data Analytics Reference Architectures Empowering Digital-Twin-Aided Manufacturing," *Future Internet*, vol. 14, no. 4, p. 114, Apr. 2022, doi: 10.3390/fi14040114.

[51] K. Gleich, S. Behrendt, M. Hörger, M. Benfer, and G. Lanza, "An Asset Administration Shell-Based Digital Product Passport as a Gaia-X Service," *Procedia CIRP*, vol. 127, pp. 224–229, Jan. 2024, doi: 10.1016/j.procir.2024.07.039.

[52] C. Schmidt, F. Volz, L. Stojanovic, and H. Kett, "Integration Approaches for Digital Twins in Dataspaces," *Applied Sciences*, vol. 15, no. 21, p. 11623, Jan. 2025, doi: 10.3390/app152111623.

[53] G. Schweiger *et al.*, "An empirical survey on co-simulation: Promising standards, challenges and research needs," *Simulation Modelling Practice and Theory*, vol. 95, pp. 148–163, Sep. 2019, doi: 10.1016/j.simpat.2019.05.001.

[54] C. Friedrich, A. Lombana, J. Fasquel, C. Schlick, N. Bennani, and M. Mendil, "CoFMPy: A Python Framework for Rapid Prototyping of FMI-based Digital Twins," in *The 2nd International Conference on Engineering Digital Twins*, 2025. Accessed: Nov. 10, 2025. [Online]. Available: https://hal.science/hal-05326255/

[55] J. Hugues, A. Hristosov, J. J. Hudak, and J. Yankel, "TwinOps - DevOps meets model-based engineering and digital twins for the engineering of CPS," in *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings*, in MODELS '20. New York, NY, USA: Association for Computing Machinery, Oct. 2020, pp. 1–5. doi: 10.1145/3417990.3421446.

[56] Y. Wu, K. Zhang, and Y. Zhang, "Digital Twin Networks: A Survey," *IEEE Internet of Things Journal*, vol. 8, no. 18, pp. 13789–13804, Sep. 2021, doi: 10.1109/JIOT.2021.3079510.

[57] L. U. Khan, W. Saad, D. Niyato, Z. Han, and C. S. Hong, "Digital-Twin-Enabled 6G: Vision, Architectural Trends, and Future Directions," *IEEE Communications Magazine*, vol. 60, no. 1, pp. 74–80, Jan. 2022, doi: 10.1109/MCOM.001.21143.

[58] F. Tang, X. Chen, T. K. Rodrigues, M. Zhao, and N. Kato, "Survey on Digital Twin Edge Networks (DITEN) Toward 6G," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1360–1381, 2022, doi: 10.1109/OJCOMS.2022.3197811.

[59] G. N. Schroeder, C. Steinmetz, R. N. Rodrigues, A. Rettberg, and C. E. Pereira, "Digital Twin connectivity topologies," *IFAC-PapersOnLine*, vol. 54, no. 1, pp. 737–742, Jan. 2021, doi: 10.1016/j.ifacol.2021.08.086.

[60] M. Frasheri *et al.*, "Addressing time discrepancy between digital and physical twins," *Robotics and Autonomous Systems*, vol. 161, p. 104347, Mar. 2023, doi: 10.1016/j.robot.2022.104347.

[61] M. Picone, E. Bouquillon, M. Pietri, and M. Mamei, "Assessing the Impact of Cybersecurity Attacks on Digital Twin Metrics: An Experimental Study," in *2026 IEEE 23rd Consumer Communications & Networking Conference (CCNC)*, IEEE, 2026, pp. 1–4. Accessed: Mar. 27, 2026. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/11366535/

[62] C. Gomes, D. E. L. Rötter, A. Iosifidis, H. Feng, H. Ejersbo, and M. Frasheri, "Sensing and Communication of Data from the Physical Twin," in *The Engineering of Digital Twins*, J. Fitzgerald, C. Gomes, and P. G. Larsen, Eds., Cham: Springer International Publishing, 2024, pp. 147–171. doi: 10.1007/978-3-031-66719-0_7.

[63] C. Badolato *et al.*, "PSMark: A distributed IoT benchmark for publish/subscribe under domain-based workloads," in *PerCom 2026-24th IEEE International Conference on Pervasive Computing and Communications*, 2026. Accessed: Mar. 18, 2026. [Online]. Available: https://inria.hal.science/hal-05517145/

[64] A. Almeida, T. Batista, E. Cavalcante, F. Delicato, R. Motta, and M. Vieira, "Middleware for Digital Twins: A Systematic Mapping Study," in *Proceedings of the 1st International Workshop on Middleware for Digital Twin*, in Midd4DT '23. New York, NY, USA: Association for Computing Machinery, Dec. 2023, pp. 19–24. doi: 10.1145/3631319.3632302.

[65] P. Bellavista, N. Bicocchi, M. Fogli, C. Giannelli, M. Mamei, and M. Picone, "An Entanglement-Aware Middleware for Digital Twins," *ACM Trans. Internet Things*, Oct. 2024, doi: 10.1145/3699520.

[66] P. Bellavista, N. Bicocchi, M. Fogli, C. Giannelli, M. Mamei, and M. Picone, "Exploiting microservices and serverless for Digital Twins in the cloud-to-edge continuum," *Future Generation Computer Systems*, vol. 157, pp. 275–287, Aug. 2024, doi: 10.1016/j.future.2024.03.052.

[67] T. Liu, L. Tang, W. Wang, Q. Chen, and X. Zeng, "Digital-Twin-Assisted Task Offloading Based on Edge Collaboration in the Digital Twin Edge Network," *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 1427–1444, Jan. 2022, doi: 10.1109/JIOT.2021.3086961.

[68] N. Apostolakis, L. E. Chatzieleftheriou, D. Bega, M. Gramaglia, and A. Banchs, "Digital Twins for Next-Generation Mobile Networks: Applications and Solutions," *IEEE Communications Magazine*, vol. 61, no. 11, pp. 80–86, Nov. 2023, doi: 10.1109/MCOM.001.2200854.

[69] S. Gil, P. H. Mikkelsen, D. Tola, C. Schou, and P. G. Larsen, "A Modeling Approach for Composed Digital Twins in Cooperative Systems," in *2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Sep. 2023, pp. 1–8. doi: 10.1109/ETFA54631.2023.10275601.

[70] C. Alcaraz and J. Lopez, "Digital Twin: A Comprehensive Survey of Security Threats," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 3, pp. 1475–1503, 2022, doi: 10.1109/COMST.2022.3171465.

[71] P. Singh, N. Meratnia, M. Beliatis, and M. Presser, "Navigating the International Data Space To Build Edge-Driven Cross-Domain Dataspace Ecosystem: English," Aug. 2024.

[72] P. Singh,..N., M. Beliatis, and M. Presser, "Data-Driven IoT Ecosystem for Cross Business Growth: An Inspiration Future Internet Model with Dataspace at the Edge," *INTERNET 2024 :*, Mar. 2024.

[73] P. Singh, M. J. Beliatis, and M. Presser, "Enabling edge-driven Dataspace integration through convergence of distributed technologies," *Internet of Things*, vol. 25, p. 101087, Apr. 2024, doi: 10.1016/j.iot.2024.101087.

[74] T. Kulik, Z. Kazemi, and P. G. Larsen, "Security and Privacy-related Issues in a Digital Twin Context," in *The Engineering of Digital Twins*, J. Fitzgerald, C. Gomes, and P. G. Larsen, Eds., Cham: Springer International Publishing, 2024, pp. 313–344. doi: 10.1007/978-3-031-66719-0_13.

[75] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture," National Institute of Standards; Technology, Aug. 2020. doi: 10.6028/NIST.SP.800-207.

[76] J. M. Bernabé Murcia, E. Cánovas, J. García-Rodríguez, A. M. Zarca, and A. Skarmeta, "Decentralised Identity Management solution for zero-trust multi-domain Computing Continuum frameworks," *Future Generation Computer Systems*, vol. 162, p. 107479, Jan. 2025, doi: 10.1016/j.future.2024.08.003.

[77] A. R. Qureshi, A. Asensio, M. Imran, J. Garcia, and X. Masip-Bruin, "A survey on security enhancing Digital Twins: Models, applications and tools," *Computer Communications*, vol. 238, p. 108158, Jun. 2025, doi: 10.1016/j.comcom.2025.108158.

[78] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Communication-Efficient Federated Learning and Permissioned Blockchain for Digital Twin Edge Networks," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2276–2288, Feb. 2021, doi: 10.1109/JIOT.2020.3015772.

[79] Y.-J. Kim, H. Kim, B. Ha, and W.-T. Kim, "Federated Digital Twins: A scheduling approach based on Temporal Graph Neural Network and Deep Reinforcement Learning," *IEEE Access*, 2025, Accessed: Jan. 23, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10843696/

[80] V. Villani, M. Picone, M. Mamei, and L. Sabattini, "A Digital Twin Driven Human-Centric Ecosystem for Industry 5.0," *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 11291–11303, 2025, doi: 10.1109/TASE.2024.3410703.

[81] A. Tóth, L. Nagy, R. Kennedy, B. Bohuš, J. Abonyi, and T. Ruppert, "The human-centric Industry 5.0 collaboration architecture," *MethodsX*, vol. 11, p. 102260, Dec. 2023, doi: 10.1016/j.mex.2023.102260.

[82] R. Azevedo *et al.*, "Human Digital Twins to Support Nurse Practitioners' Clinical Decision-Making Using Multimodal Data: A Theoretical, Methodological, and Analytical Framework," in *Digital Twin: Fundamentals and Applications*, S. Sabri, K. Alexandridis, and N. Lee, Eds., Cham: Springer Nature Switzerland, 2024, pp. 149–172. doi: 10.1007/978-3-031-67778-6_7.

[83] K. Alexandridis and M. LaFontaine, "Social and Human Dimensions of Digital Twin Technologies in Formal and Informal Institutional Settings," in *Digital Twin: Fundamentals and Applications*, S. Sabri, K. Alexandridis, and N. Lee, Eds., Cham: Springer Nature Switzerland, 2024, pp. 211–236. doi: 10.1007/978-3-031-67778-6_10.

[84] M. Herath, M. Alvi, R. Minerva, H. Dutta, N. Crespi, and S. M. Raza, "Smart City Digital Twins: A Modular and Adaptive Architecture for Real-Time Data-Driven Urban Management," in *2024 20th International Conference on Network and Service Management (CNSM)*, IEEE, 2024, pp. 1–7. Accessed: Jul. 11, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10814627/?casa_token=b3lXuilhxVEAAAAA:WzY-AJNhNhitSev00P-EOsrAct0Rzn5kMII2--FmI-MWhs7zhfPeakDTrSPTTaTbbfDKj8c

[85] D. McKee and D. Dokter, "DISCS: An Approach for Accelerating the Development of Digital Twins for Smart Cities," in *Digital Twin: Fundamentals and Applications*, S. Sabri, K. Alexandridis, and N. Lee, Eds., Cham: Springer Nature Switzerland, 2024, pp. 31–58. doi: 10.1007/978-3-031-67778-6_3.

[86] J. Reynolds, S. Sabri, and B. Lee, "Digital Twins for Creating Value Through 'Buildings as Batteries' Using a Mass Customization Network," in *Digital Twin: Fundamentals and Applications*, S. Sabri, K. Alexandridis, and N. Lee, Eds., Cham: Springer Nature Switzerland, 2024, pp. 191–209. doi: 10.1007/978-3-031-67778-6_9.

[87] "Tools and libraries catalogue." Change2Twin project.

[88] "Change2Twin marketplace review." Change2Twin project. Available: https://www.change2twin.eu/about/deliverables/

[89] P. G. Larsen *et al.*, "HUBCAP: A Novel Collaborative Approach to Model-Based Design of Cyber-Physical Systems," in *Simulation and Modeling Methodologies, Technologies and Applications*, M. S. Obaidat, T. Oren, and F. D. Rango, Eds., Cham: Springer International Publishing, 2022, pp. 90–110. doi: 10.1007/978-3-030-84811-8_5.

[90] K. M. Alam and A. El Saddik, "C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems," *IEEE access*, vol. 5, pp. 2050–2062, 2017, Accessed: Jan. 08, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7829368/

[91] B. Combemale *et al.*, "Model-Based DevOps: Foundations and Challenges," in *2023 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, Oct. 2023, pp. 429–433. doi: 10.1109/MODELS-C59198.2023.00076.

[92] J. Michael *et al.*, "Model-Driven Engineering for Digital Twins: Opportunities and Challenges," *Systems Engineering*, vol. 28, no. 5, pp. 659–670, 2025, doi: 10.1002/sys.21815.

[93] P. van Schalkwyk and D. Isaacs, "Achieving Scale Through Composable and Lean Digital Twins," in *The Digital Twin*, N. Crespi, A. T. Drobot, and R. Minerva, Eds., Cham: Springer International Publishing, 2023, pp. 153–180. doi: 10.1007/978-3-031-21343-4_6.

[94] J. Robles, C. Martín, and M. Díaz, "OpenTwins: An open-source framework for the development of next-gen compositional digital twins," *Computers in Industry*, vol. 152, p. 104007, Aug. 2023, doi: 10.1016/j.compind.2023.104007.

[95] S. Infante, J. Robles, C. Martín, B. Rubio, and M. Díaz, "Distributed digital twins on the open-source OpenTwins framework," *Advanced Engineering Informatics*, vol. 64, p. 102970, Mar. 2025, doi: 10.1016/j.aei.2024.102970.

[96] *Foundational Research Gaps and Future Directions for Digital Twins*. Washington, D.C.: National Academies Press, 2024. doi: 10.17226/26894.

[97] D. Parle, G. Sharma, N. Anand, N. Padgaonkar, D. Stoddart, and D. Malley, "A Comparative Analysis for Harnessing Digital Twin Platforms for Net-Zero Manufacturing," *IEEE Access*, vol. PP, Aug. 2024, doi: 10.1109/ACCESS.2024.3447475.